THE TANZANIA COMMISSION FOR UNIVERSITIES

Benchmarks for Bachelor Degree Programmes in Information and Communication Technology

February 2024

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ACM	Association for Computing Machinery
AI	Artificial Intelligence
CBS	Cybersecurity
CE	Computer Engineering
CS	Computer Science
DS	Data Science
EE	Electronics Engineering
ELOs	Expected Learning Outcomes
EQF	European Qualification Framework
ERP	Enterprise Resource Planning
ICT	Information and Communication Technology
IEEE	Institute of Electrical and Electronics Engineers
IS	Information Systems
IT	Information Technology
IUCEA	Inter-University Council for East Africa
MoCU	Moshi Cooperative University
MU	Mzumbe University
NACTVET	National Council for Technical and Vocational Education and Training
RUCU	Ruaha Catholic University
SE	Software Engineering
SUZA	State University of Zanzibar
TCU	Tanzania Commission for Universities
TE	Telecommunications Engineering
UDOM	University of Dodoma
UDSM	University of Dar es Salaam
ZU	Zanzibar University

EXECUTIVE SUMMARY

These Benchmarks for programmes in the cluster of Information and Communication Technology (ICT) have been developed to serve as a point of inference for university institutions when developing or reviewing existing programmes in the same areas at the University Qualifications Framework (UQF) Level 8. That is, programmes at the Bachelor degree level. They include ten (10) UQF Level 8 programmes, namely, Computer Science (CS), Information Systems (IS), Information Technology (IT), Computer Engineering (CE), Software Engineering (SE), Telecommunications Engineering (TE), Electronics Engineering (EE), Cybersecurity (CBS), Data Science (DS), and Artificial Intelligence (AI).

The Benchmarks build on the foundation of the earlier works by different institutions including the Inter-University Council for East Africa (IUCEA), joint benchmarks by the Association for Computing Machinery (ACM), the Institute of Electrical and Electronics Engineers (IEEE) and the Association for Information Systems (AIS). They have also been developed while making reference to other benchmarks developed by institutions such as Nigeria National Universities Commission, Uganda National Council for Higher Education and South Africa National Commission on Higher Education. Reference was also made to existing ICT-related curricula currently being offered in various universities in Tanzania such as the University of Dar es Salaam (UDSM), University of Dodoma (UDOM), St. Joseph University in Tanzania (SJUIT), Mzumbe University (MU), Mbeya University of Science and Technology (MUST), and Sokoine University of Agriculture (SUA).

As mentioned hitherto, these Benchmarks are not directly linked to a degree structure in any specific context, but they provide guidance to those universities, which aspire to develop/review programmes in the ICT cluster in specific knowledge areas. That is, the expected minimum competencies required to be gained by a student in a particular discipline before he/she exist the university.

One of the reasons for developing these Benchmarks is to harmonise competencies among graduates from different university institutions who studied the same degree programme. Further, the Benchmarks have been developed taking into consideration of the fact that the ICT discipline is characterised by rapid changes in technology and industry practices. This includes the globalisation of ICT development processes, the advent of digital crimes and the associated digital threats, emergence of new digital applications such as mobile applications, introduction of Web technologies, e-learning, e-banking, e-governance, blockchain technologies, AI, etc. Also, the changes are cognisant of the emergence of new architectural paradigms, widespread utilization of large-scale Enterprise Resource Planning (ERP) systems, open source technologies, ubiquitous availability of mobile computing, and broad use of IT control and infrastructure frameworks.

The Tanzania Commission for Universities (TCU) has developed these Benchmarks after realising the increasing interest in the ICT discipline in the country. They represent an effort by the Commission to establish a guide that will enable universities across the country to prepare ICT-related curricula that will produce graduates with comparable learning outcomes to enable them (graduates), employers and the society as whole to reap the maximum benefits of the tertiary education system. Moreover, the Benchmarks will enable reviewers of academic programmes to objectively judge various curricula proposals that will be developed by different university institutions in the country while complying with the minimum core knowledge areas for each discipline, with avenues for the students to specialise in some subdomains of their choice with sufficient skills that will enable them to be employable.

It is important to note that these Benchmarks have taken into account the current entire ICT discipline i.e., Computing and Communication as opposed to other Benchmarks, which are exclusively confined to computing only. Further, the Association for Computing Machinery - Institute of Electrical and Electronics Engineers (ACM-IEEE-2017) identified CBS, AI and DS as three emerging technologies, hence their inclusion in the Benchmarks. Let it be noted further that in late 2017 and early 2019, ACM-IEEE recognized CBS and DS as independent programmes, respectively. These programmes are sufficiently represented in these Benchmarks.

CHAPTER 1

GENERAL INTRODUCTION

1.1 Background

The Standards and Guidelines for University Education in Tanzania, 2019 recognises that Programme Benchmarks are important tools for harmonisation of academic programmes in the same disciplinary area for the purpose of comparability of the quality of graduates who pursue similar programmes in different universities. Cognizant of this fact, the Commission has been developing Programme Benchmarks in different clusters. This is also reflected in the TCU Rolling Strategic Plan (2020/21-2025/26). Therefore, the Benchmarks will enable university institutions in Tanzania to develop and make review academic programmes in the ICT domain according to their institutional mission and programme goals.

After performing environmental scanning using various national, reginal and international benchmarks, ten (10) programmes in the ICT cluster were developed. These are, namely: Computer Science (CS); Information Systems (IS); Information Technology (IT); Computer Engineering (CE); Software Engineering (SE); Telecommunications Engineering (TE); Electronic Engineering (EE); Cybersecurity (CBS); Data Science (DS); and Artificial Intelligence (AI).

1.2 Objectives and Justification of the Formulated Benchmarks

These Benchmarks aim at fulfilling several objectives, including to:

i) Enable the Commission to assess the quality of curricula in the ICT cluster;

ii) Promote harmonisation of the specific programmes;

iii) Support staff and student mobility;

iv) Produce employable graduates and graduates with ability for self-employment;

v) Guide the labour market in judging the quality of graduates; and

vi) Produce graduates fit for regional and global ICT labour market.

1.3 Development Process

The Benchmarks were developed by a team of four (04) experts from university institutions. Prior to commencement of the assignment, the team had a one-day training session on Benchmarks, their components and development process. Some of the key issues covered in the training include current national/global trends in higher education, higher education scenario in Tanzania, important elements for harmonization of education, speaking the same language about quality, basic elements for harmonisation of higher education, the context of quality and quality assurance in higher education, the context of quality and harmonization of higher education, Benchmarks for academic programmes, and Qualifications Framework.

After the training, the team embarked on the development of the Benchmarks. The process involved different methods, including review and analysis of various related documents from different units within and outside the country such as East Africa, Africa, Europe, America and Asia, and consultations with various stakeholders. Among the documents reviewed include the ACM Benchmarks for computing cluster, which was based on curricula from more than 85 universities worldwide, where all continents were included.

All UQF Level 8 programmes in the ICT cluster currently being offered by different universities in Tanzania were equally used for inputs. These include programmes offered by the University of Dar es Salaam (UDSM), University of Dodoma (UDOM), St. Joseph University in Tanzania (SJUIT), Ardhi University (ARU), State University of Zanzibar (SUZA), Dar es Salaam Tumaini University (DarTU) [formerly, Tumaini University Dar es Salaam College – TUDARCo], Kampala International University in Tanzania (KIUT), Sokoine University of Agriculture (SUA), Mzumbe University (MU), Abdulrahman Al-Sumait University (SUMAIT), Moshi Co-operative University (MoCU), United African University of Tanzania (UAUT), Zanzibar University (ZU), Ruaha Catholic University (RUCU), and MUST Mbeya University of Science and Technology (MUST).

Involvement of stakeholders in the development process was essential as it ensured that actors including employers, regulatory agencies, and university institutions were given the opportunity to contribute to the development of the Benchmarks. Table 1 provides a detailed list of stakeholders who participated in the development of the Benchmarks.

The draft Benchmarks were sent to two (02) independent peer reviewers and provided comments and suggestions. Comments and suggestions received were incorporated by the team. The revised version was sent to various stakeholders in the ICT cluster for them to provide inputs. Inputs received were incorporated accordingly.

The revised version of the Benchmarks was discussed at a one-day stakeholders meeting that was convened by Commission to deliberate of the draft Benchmarks. Comments and suggestions received were incorporated. The revised version was sent to the Committee of Vice Chancellor and Principals/Provosts of Universities and University College in Tanzania (CVCPT) for deliberation and endorsement.

Table 1: List of stakeholders for programmes in the ICT cluster

SN	Category	Stakeholder identified	Programme(s)	Role (s)
1.	Professional Bodies	ICT Commission	All programmes under the ICT cluster	Registers ICT professionals, promotes and facilitates implementation of national ICT initiatives countrywide
		Engineers Registration Board (ERB)	Computer Engineering, Telecommunications Engineering, Software and Electronics Engineering	Registers engineering professionals and regulates engineering practices countrywide
2.	University institutions	UDSM, UDOM, SJUIT, ARU, SUZA, TUDARCo, KIUT, SUA, MU, SUMAIT, MoCU, UAUT, ZU, RUCU, MUST, etc.	All programmes under the ICT cluster	Implementation of the Benchmarks in their respective institutions at programme level
3.	Regulatory Bodies	Tanzania Commission for Universities (TCU)	All programmes under the ICT cluster	University education regulatory Agency mandated to accredit programmes being offered by registered/accredited

SN	Category	Stakeholder identified	Programme(s)	Role (s)
				university institutions in Tanzania
		National Council for Technical and Vocational Education and Training (NACTVET)	All programmes under the ICT cluster	Ensures that programmes in the technical education meet labour market demands by guiding and monitoring their adherence to the regulatory framework
		Commission for Science and Technologies (COSTECH)	All programmes under the ICT cluster	Coordinates and promotes research and technology development activities in the country
		Tanzania Communications Regulatory Authority (TCRA)	All programmes under the ICT cluster	Regulates the use of communication countrywide
4.	ICT Industry	Confederation of Tanzania Industries (CTI)	All programmes under the ICT cluster	Employers of graduates
		Association of Tanzania Employers (ATE)		
5.	Government	Ministry of Education, Science and Technology, and Public Service Recruitment Secretariat (PSRS)	All programmes under the ICT cluster	Employers and/or implementers
6.	Private Sector	Small and medium scale ICT startups, Alumni	To participate in the respective programmes under ICT cluster	Employers and beneficiaries for consultancy services

CHAPTER 2

USE OF THE BENCHMARKS

2.1 The Benchmarks and the Qualifications Framework

A benchmark refers to a standard point of reference against which things may be compared or assessed. The developed Benchmarks must be in line with various qualifications frameworks. As per the Tanzanian University Qualifications Framework (UQF), the developed Benchmarks are at level 8 of the defined awards (i.e., UQF level 8), which is Bachelor's degree.

2.2 Bachelor's Degree

The holder of the qualification will be able to apply knowledge, skills and understanding in a wide and unpredictable variety of contexts with substantial personal responsibility for the work of others and responsibility for the allocation of resources, policy, planning, execution and evaluation. Normally, the description for any bachelor's degree programme is very general, thus each degree programme needs to be filled in and elaborated using expected learning outcomes which are formulated by the discipline. Each discipline in turn, must translate the generic learning outcomes into specific course units within specific knowledge.

2.3 The Benchmarks and Curriculum Design

One of the purposes of formulating Benchmarks is to support university institutions to design or redesign their curricula. The Benchmarks are needed because learning outcomes guarantee:

i) Comparable quality levels of the graduates within and outside the country;

ii) Comparable chances for the graduates in the labour market;

iii) Labor market understands the competencies that ICT graduates possess;

iv) Increased national and international mobility of students; and

v) Increased national and international mobility of lecturers and students.

2.4 Guiding Principles

Any programme that is designed under the ICT cluster, must be guided by the following principles:

i) The programme must be sensitive to changes in technology, new developments in pedagogy, and the importance of lifelong learning. ICT education must seek to prepare students for lifelong learning that will enable them to move beyond today's technology to meet the challenges of the future;

ii) The ICT programme must include appropriate and necessary design and laboratory experiences. The programme should include "hands-on" experience in designing, building, and testing both hardware and software systems;

iii) The ICT programme must include preparation for professional practice as an integral component. These practices encompass a wide range of activities including; management, ethics and values, written and oral communication, working as part of a team, etc.;

iv) The development of a programme in the ICT industry must include participation from different stakeholders, including the ICT industry, Government, and the full range of higher educational institutions involved in ICT education;

v) The programme must strive to be international in scope. Even though curricula requirements may differ from one country to another depending on the socio-economic development of the country, the developed programmes should make every effort to ensure that the curriculum recommendations are sensitive to international differences so that they will be widely applicable throughout the world;

vi) Learning outcomes are a necessary component in undergraduate professional education. Learning outcomes contain statements of the knowledge, skills and attitudes that students must have acquired at the end of each course; and

vii) Relevant tools and standards must appear throughout the body of knowledge.

2.5 Formulating the Expected Learning Outcomes

The first step in designing or redesigning a programme is the formulation of the learning outcomes. The purpose of the learning outcomes is to describe clearly what a student is expected to demonstrate after completing the whole programme, a module or a course. University institutions are expected to compare their formulated learning outcomes with the Benchmarks and discern what is missing or what should be rephrased. For each learning outcome, one should describe how the learning outcome would be measured and assessed.

There are various definitions of learning outcomes or competences. The European Qualifications Framework (EQF) describes learning outcomes as *statements of what a learner knows, understands, and can do on completion of a learning process, which are defined in terms of knowledge, skills and competence*. Further, a *competence is the proven ability to use knowledge, skills and personal, social and/or methodologic abilities, in work or study situations and in professional and personal development*.

On the other hand, IUCEA describes learning outcomes as what a *learner is expected to know and understand, and be able to do/use or demonstrate, on completion of a learning process within a recognised qualifications framework*.

Defining learning outcomes would create an opportunity to:

i) Enhance students' learning and mobility;

ii) Provide guidance to instructors;

iii) Identify and overcome barriers to effective teaching;

iv) Facilitate collaboration among university institutions in the region and beyond;

v) Improve students' learning, retention and completion;

vi) Produce quality graduates; and

vii) Increase students´ chances for employability.

Because of this, formulating the learning outcomes becomes the first step in designing any degree programme. For each learning outcome, one should describe how the outcome would be measured and assessed. Since benchmarks are based on the formulated learning outcomes, these can be divided into three domains: Cognitive learning (Knowledge and its application), Psychomotor learning (Skills), and Affective learning (Attitude, feelings, values, and its appreciations).

In Bloom's taxonomy, the teaching and learning hierarchy is important for the correct and consistent building of the knowledge side of the learning outcomes. Bloom's Taxonomy, which has been well explored within the ICT cluster, specifies the cognitive skill level as Familiarity, Usage and Assessment (FUA). However, for the mastery levels to be indicative and not to impose theoretical constraint on users of this document, mastery level has been categorized into Knowledge, Skills and Attitude:

i) **Knowledge:** Knowledge means the outcome of the assimilation of information through learning. It is the body of facts, principles, theories and practices that is related to a field of work or study.

ii) **Skills:** Skills means ability to apply knowledge and use know-how to complete tasks and solve problems. Skills are categorized as:

- Cognitive skills (involve the use of logical, intuitive and creative thinking);
- Practical skills (involve manual dexterity and the use of methods, materials, tools and instruments); and
- Interpersonal skills (the way of communication, cooperation, etc.).

iii) **Attitude:** Attitude means a settled way of thinking or feeling about something. It is divided into four major components:

- Affective (emotions or feelings);
- Cognitive (belief or opinions held consciously);
- Conative (inclination for action); and
- Evaluative (positive or negative response to stimuli).

2.6 Translating Learning Outcomes in the Programme

The next step after the formulation of learning outcomes is to identify what courses are needed to achieve the learning outcomes. A distinction must be made between the core and the supporting knowledge. Establishment of what is already present in the programme (may be with another name) and what knowledge should be beaded is essential.

To check if a planned course covers the learning outcomes, it is important to develop a curriculum alignment matrix, an example is shown in Table 2**Error! Reference source not found.**. For each course, the specific learning outcomes have to be formulated and one must check how far the course contributes to the achievement of the programme learning outcomes.

Table 2: Example of curriculum alignment matrix

Learning outcomes	Course 1	Course 2	Course 3	Course 4
Learning outcomes 1	x		x	
Learning outcomes 2		x	x	x
Learning outcomes 3	x			x

2.7 Course Description

Course descriptions provide students with basic information to evaluate and enrol in courses. University institutions should provide course descriptions for students in all degree programmes to orient them in the course rationale, frame a brief overview of the key content, knowledge and skills to be learned, and state the major learning strategies and activities that students will experience. Course descriptions should include course title, course aim, course expected learning outcomes, course status, credit rating, total hours spent, course content, teaching and learning activities, assessment methods and reading list. Course descriptions should:

i) Be student-centred rather than teacher-centred or course-centred;

ii) Use brief, outcomes-based, descriptive phrases that begin with an imperative or active verb (e.g., design, demonstrate, plan, analyse);

iii) Be clear, concise, and easy to understand;

iv) Indicate significant learning outcomes of the course; and

v) Align the outcomes with the course contents.

2.8 The Benchmarks and Quality Assurance

It is envisaged that the Benchmarks will play a significant role in quality assurance of the programmes. It is therefore expected that university institutions will ideally align their standards with the Benchmarks. The Benchmarks also offer external assessment teams a frame of reference in assessing the quality of a programme.

2.9 Implementation of the ICT Benchmarks

All university institutions offering or are planning to offer UQF Level 8 programmes will be required to comply with the minimum standards provided in these Benchmarks. However, universities may surpass the minimum quality assurance standards provided in the Benchmarks.

2.10 Review of the Benchmarks

Due to the rapid changes in ICT and emerging technologies, the Benchmarks will be regularly reviewed and evaluated. However, from the preceding presentations, Benchmarks are not curricula. Thus, the development and review of curricula should be carried out in line with the existing guiding instruments.

CHAPTER 3

BENCHMARKS FOR A BACHELOR OF SCIENCE IN COMPUTER SCIENCE PROGRAMME

3.1 Computer Science Programme

Computer Science (CS) is sometimes called Computing Science, but it should not be confused with Computational Science or Software Engineering. It is a discipline that is concerned with theoretical concepts, principles, innovation, logic-based problem-solving techniques, design algorithms and software systems based on sound mathematical foundations, engineering, and scientific procedures. Some institutions provide courses with different nomenclature, but there are minor differences amongst them which are related to the coverage given to certain knowledge or topics. The course curriculum includes theory sessions, tutorials and laboratory sessions based on computing and mathematical theoretical foundations.

Computer Science can be divided into several sub-clusters such as computational complexity theory which is highly abstract, and computer graphics which emphasises real-world applications. Likewise, programming language theory considers approaches to the description of computational processes, while computer programming itself involves the use of programming languages and complex systems. Human–computer interaction considers the challenges in making computers useful, usable, and accessible. Internet of Things, Artificial Intelligence, Data Science, Blockchain Technologies and Virtual Reality are some of the emerging sub clusters/tracks of CS. Several different tracks and specializations are offered as unique options based on a programme's or institution's mission and vision.

Graduates of this programme will take up positions such as computer and information scientists, programmers, business analysts, systems analysts, information systems security experts, database administrators, information assurance experts, data modelers, computer specialists, game developers, mobile application developers, and network engineers.

3.2 Computer Science Programme Objectives

Bachelor of Computer Science programme should be designed in such a way that it produces graduates that meet academia, government, industry, and society needs. This can be achieved by focusing on the following group of programme objectives.

3.2.1 Academic Ability

The programme objectives under this category are to equip students with knowledge/skills/ability to:

i) Analyse complex problems and synthesize solutions to those problems

ii) Develop an effective way to solve computing problems;

iii) Design and implement software and software technologies;

iv) Adapt and adopt emerging/evolving computing technologies;

v) Undertake research and to progress to higher levels of studies;

vi) Plan, design, deploy and document appropriate security for computer systems;

vii) Design, implement, and evaluate a computer-based system, process, component, or programme to meet desired needs;

viii) Verify, troubleshoot, test and analyse an existing computer-based system, process, component or programme; and

ix) Demonstrate proficiency in problem-solving techniques using the computer.

3.2.2 Academia and Industrial Linkage/Employability

The programme objectives under this category are to equip students with knowledge and skills that match with industrial needs as follows:

i) Problem-solving skills for computer related tasks;

ii) Up-to-date computing skills for the industry;

iii) Analytical skills to understand impacts of computing on individuals, organizations and society;

iv) Ability to integrate theory and practice to work effectively and efficiently in organizations;

v) Knowledge and skills that enable creativity, innovativeness, and entrepreneurship in the field of computing.

3.2.3 Personal Development

The programme objectives under this category are to:

i) Prepare learners for lifelong learning and research;

ii) Empower students to progress in their personal career; impart professional ethics to the learner; equip the learner with skills and attitude to work in multicultural and global environments;

iii) Equip the learner with knowledge and skills to work as a team in the computing field; and

iv) Enable the learner to develop skills to perform effectively in technical and nontechnical environments.

3.3 Expected Learning Outcomes for Computer Science Programme

Expected Learning Outcomes (ELOs) provide a mechanism for describing not just knowledge and relevant practical skills but also personal and transferable skills. ELOs are not of equal size and do not have a uniform mapping to curriculum hours. Topics with the same number of hours may have quite different numbers of associated learning outcomes. Each learning outcome has an associated level of mastery. Some literature defines the levels based on Bloom's Taxonomy, which has been well explored within the computing field. The Bloom's Taxonomy specify the cognitive skill level as follows:

i) **Familiarity:** The student understands the concepts and its meaning. This level of mastery concerns a basic awareness of a concept as opposed to expecting real facility with its application. It provides an answer to the question "What do you know about this?"

ii) **Usage:** The student is able to use or apply a concept in a concrete way. For example, using a concept may include the use of a specific concept in a programme, using a particular proof technique or performing a particular analysis. It provides an answer to the question "How is this done? /How do you do this?"

iii) **Assessment**: The student is able to consider a concept from multiple viewpoints and/or justify the selection of a particular approach to solve a problem. This level of mastery implies more than using a concept; it involves the ability to select an appropriate approach from understanding alternatives. It provides an answer to the question "Why would you do that?"

The formulated ELOs as stipulated in Table 3 are the threshold: all graduates of the bachelor's degree programme in CS must achieve them.

Table 3: Expected Learning Outcomes for Computer Science programme

Category	**Expected Learning Outcomes**	
	Graduates should be able to:	
Knowledge (K)	K1:	Demonstrate knowledge and an understanding of essential concepts, principles, and theories relating to computer science and software applications.
	K2:	Demonstrate ability to apply knowledge and understanding of Mathematics and Natural Sciences appropriate to computer science.
	K3:	Demonstrate knowledge and understanding of the impact of computing on the society and the environment.
	K4:	Demonstrate understanding of quality standards and benchmarks in computer software development.
	K5:	Demonstrate an ability to use current techniques, skills, and tools necessary for computing practice.
	K6:	Demonstrate an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modelling and design of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices.
	K7:	Demonstrate an ability to apply design and development principles in the construction of software systems of varying complexity.
	K8:	Demonstrate understanding on institutional management, structure and governance.
Skills		
Cognitive Skills (SC)	SC1:	Ability to model, design, implement and evaluate computer-based systems.
	SC2:	Ability to develop computer software/ applications using modern platforms.

Category	Expected Learning Outcomes	
	SC3:	Ability to analyse a problem, and identify and define the computing requirements appropriate to its solution.
	SC4:	Ability to analyse the local and global impact of computing on individuals, organizations, and society.
	SC5:	Demonstrate creativity and innovativeness in developing computing solutions to real world problems.
	SC6:	Ability to assess risk related to computing activities.
	SC7:	Ability to evaluate the extent to which a computer-based system meets the criteria defined for its current use and scale up to future development.
Practical Skills (SP)	SP1:	Ability to deploy appropriate tools for the specification, design and implementation, of computer-based systems.
	SP2:	Ability to specify, plan, manage, conduct and report on a computer science research project.
	SP3:	Ability to prepare technical reports and deliver technical presentations.
	SP4:	Ability to plan, design, deploy and document appropriate security for computer systems.
	SP5:	Ability to design, implement, and evaluate a computer-based system, process, component, or programme to meet desired needs.
	SP6:	Ability to evaluate, verify, trouble-shoot, test and analyse an existing computer-based system, process, component or programme.
Interpersonal Skills (SI)	SI1:	Ability to function effectively in a team to accomplish a common goal.
	SI2:	Ability to communicate effectively with a range of audiences i.e. experts and non-experts.
	SI3:	Demonstrate an understanding of professional, ethical, legal, security, social issues and responsibilities in computing.
Attitude (A)	A1:	Ability to adapt to, and work in a multi-cultural and

Category	Expected Learning Outcomes
	global computing environment. A2: Show awareness and understanding of the ethical standards of the profession. A3: Act professionally in the work environment. A4: Demonstrate commitment to lifelong learning self and professional development. A5: Show self-awareness and ability to adapt to new situations. A6: Be creative and innovative in developing computing solutions to real world problems.

3.4 Translating Learning Outcomes in the Basic Phase of Computer Science Programme

Learning outcomes need to be translated into the programme which is in this case considered as a coherent set of courses leading to a bachelor's degree in Computer Science. This bachelor's degree programme is commonly divided into foundation and specialization phases as:

i) **The Basic/Foundation Phase:** This phase is common for all Computer Science programme students and consists of core and supporting courses.

ii) **The Specialization Phase:** This allows students to choose certain specializations according to their interests.

In the foundation phase, three areas can be distinguished as follows:

- **Core Courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline and they are mandatory for every student. The core courses are the backbone of the discipline, and they are typical Computer Science courses.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses, it will be difficult to understand the core courses. For example, Linear algebra is compulsory for all CS students.

- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student, to deepen or broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 4 presents the core and supporting knowledge areas in the basic phase of a Bachelor of Science in Computer Science programme. However, the following remarks apply:

i) The titles of the courses may differ from one university to another. The emphasis should be on the content rather than title.

ii) The autonomy and the uniqueness of the university institution will be taken into consideration in formulating the core courses for the basic phase. The university institution will have the choice to add their own courses beyond the core and supporting courses provided in the Benchmarks.

Table 4: Core and supporting knowledge areas for Computer Science programme

Core Knowledge Areas	Supporting Knowledge Areas
• Algorithms Complexity • Systems Security • Computational Science • Systems Analysis and Design • Software Engineering • Computer Networks and Communication • Parallel and Distributed Computing • Information Management • Computer Programming • Preparation for Professional Practice • Platforms Technologies • Emerging Technologies	• Mathematics • General Studies

It is important to note that the core and supporting courses can be implemented in a specific local context as independent courses or as components within fewer courses. It important to note that there is a core content that every undergraduate CS programme should incorporate, and that the list of core courses captures this content.

3.5 Credit Framework for Computer Science Programme

TCU's UQF defines a credit as a numerical value on a Qualifications Framework standard that represents the estimated time needed for a learner to achieve required specific learning outcomes. A credit within the UQF equates to learning outcomes achievable in 10 hours of learning time. It is a measurement unit for 'notional' or 'average learning' time which includes all the activities which the learner is expected to undertake in order to achieve the learning outcomes. Such activities for UQF level 8 conventional mode of learning include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training as provided in Table 5.

Table 5: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminars and Tutorials	% on Assignments	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 360 as the minimum cumulative credits required for a learner to graduate with a three-year UQF level 8 programme such as the Bachelor of Science in Computer Science. It should be noted that the curriculum designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn programming, more weight should be assigned to the practical activity.

3.6 Normal Learning Matrix for Computer Science Programme

The Normal Learning Matrix in Table 6 is a result of environmental scanning of Computer Science (CS) programme curricula from various institutions, including IUCEA, ACM/ IEEE Computer Society-IEEE2013-CC, ACM/IEEE2010-IS, ACM/2017 and other benchmarks. The proposed minimum standards consist of twelve (12) core knowledge areas that form the foundation for a Bachelor of Science in Computer Science programme.

Table 6: Normal Learning Matrix with minimum standards core credits

Core Knowledge Areas	
Algorithms and Complexity [27] • Data Structures and Algorithms • Algorithms Strategies • Automata, Computability and Complexity	**Systems Security [21]** • Network Security • Information Security • Software Security • Digital Forensic • Organization Security • Human/personnel and Societal Security
Computational Science [18] • Image Processing • Modelling and Simulation • Graphics and Visualization • Digital Electronics	**System Analysis and Design [18]** • Object Oriented Analysis and Design • Structured Systems Analysis and Design
Software Engineering [27] • Software Development Fundamentals • Software Project Management and Control • Requirements Engineering • Software Testing and Software Management	**Computer Networks and Communication [27]** • Data Communication and Networks • Computer Administration and Management • Computer Networks • Routers and Routing • Wireless Communication
Parallel and Distributed Computing [21] • Cloud Computing • Human Computer Interaction • Intelligent Systems • Distributed Systems	**Information Management [24]** • Information Management Concepts • Database Systems • Data Mining and Warehousing • Database Management Systems

Computer Programming [36] • Programming Fundamentals • Procedural Programming • Object-Oriented Programming • Logic Programming • Internet Programming • Scripting Concepts • Assembly Programming	**Preparation for Professional Practice [36]** • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and Organizational Behaviour • Final Year Project • Project Management • Intellectual Property and Legal Issues
Platforms Technologies [16] • Open-Source Technologies • Operating Systems • Computer Organization and Architecture • Computer hardware and maintenance • Mobile Technologies	**Emerging Technologies [16]** • Internet of Things • Artificial Intelligence • Data Science • Blockchain • Virtual Reality
Supporting Knowledge Areas	
Mathematics [21] • Discrete Mathematics • Mathematics Logic and Formal Semantics • Linear Algebra • Numerical Analysis • Probability and Statistics • Complex Analysis	**General Studies [21]** • Communication Skills • Development Perspectives

An institution should design a CS programme such that a student covers a minimum of 360 credit hours in three years for core, supporting and elective knowledge areas. For example, the minimum standard to deliver lectures in Modelling and Simulation of the Knowledge Area of

Computational Science is proposed to be a minimum of two (02) hours per week. It should be noted that an equal learning weight of two (02) hours has been placed on delivering practical. In a semester of 15 weeks, it is appropriate to say that a course in Modelling and Simulation would take 30 hours for lectures and 30 hours for practical. The minimum credits to deliver knowledge in Computational Science is 18 hours, as stipulated in Table 7.

Table 7: Sample minimum credits for modelling and simulation knowledge area

Knowledge Units	**# of Courses**	**Lectures Hrs.**	**Seminars and Tutorials Hrs.**	**Assignments. Hrs.**	**Individual Studies and Research Hrs.**	**Practical Training Hrs.**	**Total Hours**	**Credits**
% of Learning Activity		30	20	10	10	30	100	
Modelling and Simulation	1	30	15	7.5	7.5	30	90	9
Graphics and Visualization	1	30	15	7.5	7.5	30	90	9
Total Minimum Credit for Computational Science							**180**	**18**

3.7 Computer Science Course Description

All courses to be taught and learnt within the programme should follow the outline in Table 8. Universities may use numbering for listing purposes.

Table 8: Course outline for Computer Science programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, “What is the purpose of the course?”, “What is the course trying to achieve?”
Course Expected Learning Outcome (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit

	framework
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

3.8 Learning Outcomes and Curriculum Alignment Matrix for Computer Science Programme

It is important to develop a curriculum alignment matrix to check if the planned courses cover the learning outcomes. For each course, one has to formulate the specific learning outcomes for that course and check how far the course contributes to the programme learning outcomes. Table 9 shows an example of a curriculum alignment matrix for the Expected Learning Outcomes for Bachelor of Science in Computer Science programme where K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal and Attitude, respectively.

Table 9: Curriculum alignment matrix for Computer Science programme

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**K5**	**K6**	**K7**	**SC1**	**SC2**	**SC3**	**SC4**	**SC5**	**SC6**	**SC7**	**SP1**	**SP2**	**SP3**	**SP4**	**SP5**	**SP6**	**SI1**	**SI2**	**SI3**	**A1**	**A2**	**A3**	**A4**	**A5**	**A6**
Algorithms and Complexity	x	x		x	x	X	x	x	x	x	x	x	x		x	x	x	x	x	x	x	x	x						
Systems Security			x	x	x								x	x				x					x						
Computational Science	x	x			x	x	x	x				x			x														
System Analysis and Design	x			x	x	x	x	x		x				x	x		x	x	x										
Software Engineering				x		x	x		x	x		x	x	x	x	x		x	x	x									
Computer Networks and Communication	x				x		x								x			x	x										
Parallel and Distributed Computing	x	x					x					x	x	x									x						

Core Knowledge Areas	K1	K2	K3	K4	K5	K6	K7	SC1	SC2	SC3	SC4	SC5	SC6	SC7	SP1	SP2	SP3	SP4	SP5	SP6	SI1	SI2	SI3	A1	A2	A3	A4	A5	A6
Information Management	x			x		x	x	x	x		x	x	x	x	x				x	x									x
Computer Programming	x			x		x	x	x	x	x		x			x			x	x	x									
Preparation for Professional Practice			x								x			x		x	x				x	x	x	x	x	x	x	x	x
Platforms Technologies	x			x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x									
Emerging Technologies	x	x	x	x		x	x	x	x	x	x	x	x		x	x	x		x	x	x	x	x						
Supporting Knowledge Areas																													
General Studies			x																					x	x	x	x	x	x
Mathematics	x	x	x			x		x		x		x	x											x			x		

3.9 Benchmarks for Specialization of Computer Science Programme

After the basic phase, students may choose to deepen their knowledge in one of the following specializations:

i) Multimedia Systems, Game Programming and Animation;

ii) Artificial Intelligence and Knowledge Based Systems;

iii) Parallel, networking and Distributed Systems;

iv) Software Engineering;

v) Systems and Information Security; and

vi) Mobile Computing.

3.10 Minimum Entry Qualification for CS Programme

Entry requirements for admission into the Bachelor of Science in Information Technology programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have sufficient background in Mathematics, and for equivalent qualifications, a first-class Diploma in CS from a recognised institution.

3.11 Computer Science Programme Duration

The duration for Bachelor of Science in Computer Science programme should be three (03) years or six (06) semesters, two (02) semesters per year.

3.12 Graduation Requirement for Computer Science Programme

A student is required to accumulate a minimum of 360 credits to graduate.

3.13 Evaluation and Course Assessment for Computer Science Programme

3.13.1 Evaluation

University institutions should offer up-to-date courses to learners in the CS programme. Due to the rapid changes in the CS field and its emerging technologies, the design and implementation of curricula should be regularly reviewed and evaluated. The assessment can be done in several ways including written student evaluations, in-class observations, industrial symposium, and personal interviews with various stakeholders such as academia, alumni, employers and experts from the ICT industries. Once the course has been taught for three years, it is a good practice to interview graduates regarding the value of such a course to their professional work environment.

During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of CS graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

3.13.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

3.14 Resource Requirements for Computer Science Programme

3.14.1 Human Resource Requirements

Competent staff members are vital to the strength of the CS programme. For each CS programme, academic staff need both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete the degree in a timely manner. The interests and qualifications of the staff members must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

Staff must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual staff members. The school/department should support continuing staff development. Given the rapidly changing technology, it is particularly critical that staff members have sufficient time for professional development and scholarly activities. Resources should be provided for staff to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organisations. The CS programme is enhanced significantly when staff acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. Staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed by the programme is influenced by factors such as the number of students in the programme, the number of required courses, the number of supporting and elective courses offered, and the teaching load of the faculty/department. A CS programme should have a minimum number of full-time staff members with primary commitment to the CS programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty/department expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the staff members should span a range of interests in Computer Science. Additional staff will be needed to teach the supporting courses that provide foundation-level knowledge across the campus.

3.14.2 Computing Infrastructure Requirements

Computing infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the CS programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most freshmen now enter college with computer resources, so access plays a much more significant role (Lee, 2009). Therefore, network access should be available for staff and students to use with their own computers. Students at different levels in the curriculum have different needs. Substantial resources must be provided to support the courses targeted to all students. More sophisticated resources are necessary for Information Systems minors and majors who are developing skills in computing and CS fundamentals. Specialized laboratories or access to specialized simulation software is needed for advanced students where group and individual projects are developed. Contemporary and emerging software development tools should be available to create the most current enterprise solutions.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of staff duties, a waste of precious staff resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital in a campus environment.

3.14.3 Laboratory Requirements

Departments require hardware and software for structured, open/public, and specialized laboratories. Students must have an opportunity to use learning materials in both structured and unstructured laboratories. Hardware and software are rapidly changing and improving. It is critical that staff and students have access to facilities reflecting an environment that graduates will be expected to use professionally. All computing systems should be kept current. A plan should exist to continuously upgrade and/or replace software and equipment in a timely manner. The rate of change in technology suggests a rapid replacement cycle, with some technologies reaching obsolescence in less than 12 months.

In addition, simulation software is becoming more prevalent for teaching advanced CS topics. This can include simulations for using applications to manage single workstations to complex enterprise-level networks. Many companies including Microsoft, Cisco, and even the textbook companies have developed sophisticated simulation software which do not require the latest equipment.

Various courses and areas of study have their own specialized requirements, such as the large database with realistic sample data that are needed for effective work in the area of data management. Students should be provided opportunities to work together on team-oriented projects. The group skills developed in this mode are critical to a successful Information Systems professional. Technological support, such as groupware, is expected for group and team activities. All laboratories must have adequate technical support in terms of professional staff to provide for installation and maintenance of the equipment. The staff should be proficient in both the hardware and software applications. Complete documentation must also be available. Laboratories should be able to support the following types of functions:

i) Structured Laboratories

A structured laboratory is a closed, scheduled, supervised experience in which students complete specified exercises. An instructor who is qualified to provide necessary support and feedback to the students provides supervision. Exercises are designed to reinforce and complement the lecture material.

ii) Open/Public Laboratories

Student ownership of computers has continued to increase. However, laboratories remain essential for those students who do not own a computer and for providing additional resources not available on personal machines.

iii) Specialized Laboratories/Studies

Laboratory facilities are necessary to support team projects and special computing environments. Special facilities may be needed for systems development, network infrastructure, and other advanced technologies.

3.14.4 Classrooms

Suitable classroom facilities, equipped with information technology teaching resources, should be provided. A computing system with multimedia facilities is necessary for demonstrating the development, implementation, and application of information technology as well as conducting walkthroughs and making presentations. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

3.14.5 Library

Library support is an important part of any academic programme. It is especially important for disciplines with rapid development of knowledge such as the ICT field. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the computing professional societies.

CHAPTER 4

BENCHMARKS FOR A BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY PROGRAMME

4.1 Information Technology Programme

Information Technology (IT) signifies all aspects of computing and its integration into all aspects of today's society and digital economy. Organizations of every kind are dependent on IT and computing systems that must work properly and efficiently, be secure, and scaled with organizational objectives and customer needs. IT professionals select computing products and services, integrate them to enhance supported environments, and develop, adapt, and manage computing technologies to meet the organization's goals and business objectives. Generally, IT is the study of systemic approaches to select, develop, apply, integrate, and administer secure computing technologies to enable users to accomplish their personal, organizational, and societal goals.

The development of an IT minimum standards is one of the vital tasks that defines the competencies IT students should possess at the time of graduation. The IT minimum standard framework will enable IT departments to implement, evaluate, and revise baccalaureate IT degree programmes according to their institutional or university mission and programme goals.

4.2 Information Technology Programme Objectives

IT programmes aim to provide IT graduates with the skills and knowledge to take on appropriate professional positions in the IT field upon graduation and grow into leadership positions in the IT field or to practice IT profession in institution/independence or pursue research or graduate studies in the field. The Bachelor of Science in IT programme should be designed in such a way that it addresses the concerns of different stakeholders. This is achieved by focusing on the following grouped programme objectives.

4.2.1 Academic Ability

The programme objectives under this category are to:

i) Enable learners to understand and, in some cases, contribute to the scientific, mathematical and theoretical foundations on which information technologies are built;

ii) Equip learners with IT knowledge, skills, and competences;

iii) Enable learners to be problem solvers, skilled practitioners, or applied research investigators who enjoy getting technology to work effectively and meet user needs in a variety of settings;

iv) Enable learners to be innovative in the development and application of IT in a dynamic environment;

v) Enable the learners to adapt and adopt emerging/evolving ICT technologies; and

vi) Prepare and develop the learners to undertake research and to progress to higher levels of studies.

4.2.2 Academia and Industrial Linkage/Employability

The programme objectives under this category are to:

i) Educate and graduate professional IT people for organisations of every kind dependent on information technology and computing systems;

ii) Equip learners with IT technical skills, entrepreneurial and managerial skills; and

iii) Prepare learners to meet the requirements of the labour market and have a competitive advantage.

4.2.3 Personal Development

The programme objectives under this category are to:

i) Manage the IT resources of an individual or organization;

ii) Anticipate the changing direction of IT and evaluate and communicate the likely utility of new technologies to an individual or organization;

iii) Impart professional ethics to learners;

iv) Prepare learners for lifelong learning and research;

v) Empower learners to progress in their personal career;

vi) Equip the learners with skills and attitude to work in multicultural and global environments;

vii) Equip the learners with knowledge and skills to work as a team in the IT field; and

viii) Enable the learners to develop skills to perform effectively in technical and nontechnical environments.

4.3 Expected Learning Outcomes for Information Technology Programme

To harmonise competencies in the Bachelor's degree programme in IT, the following ELOs have been formulated to be used as benchmarks. The formulated ELOs for IT as stipulated in Table 10 are the minimum threshold. That is, all graduates of the bachelor's programme in IT must achieve them. Besides these, a graduate also has to achieve the ELOs for their chosen specialization. University institutions may consider adding more ELOs as and when necessary, in line with their mission and vision or other identified need(s).

Table 10: Expected Learning Outcomes for Information Technology programme

Category	Expected Learning Outcomes
	The graduate should be able to:
Knowledge (K)	K1. Establish knowledge in core concepts, principle, theories and content of IT and application of learning to new situations. K2. Establish knowledge in professional, ethical, legal, security and social issues and responsibilities as well as the best practices and standards and their application. K3. Establish knowledge of existing and any emerging IT technologies. K4. Establish knowledge of current techniques, skills, and tools necessary for computing practice as well as to use and apply current technical concepts and practices in the core information technologies. K5. Establish knowledge on local and global impact of computing on individuals, organizations, environment, and society. K6. Establish knowledge to Identify and analyse user needs and take them into account in the selection, creation,

Category	Expected Learning Outcomes
	evaluation, and administration of computer-based systems. K7. Establish knowledge to Identify and analyse user needs and take them in-to account in the selection, creation, evaluation, and administration of computer-based systems.
Skills	The graduate should be able to;
Cognitive Skills (SC)	SC1:Use knowledge and understanding in continuing professional development, modelling and design of computer-based systems. SC2:Identify and analyse criteria and specifications appropriate to specific problems, and plan strategies for their solution. SC3:Evaluate critically and analyse the extent to which a different computer system meets the criteria defined for its current use and future development. SC4:Deploy appropriate theory, practices, and tools for the specification, analysis, design, implementation, and maintenance of IT systems. SC5:Evaluate systems in terms of general quality attributes and possible trade-offs among alternative systems. SC6:Identify and analyse user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems.
Practical Skills (SP)	SP1:Apply the principles of effective management and retrieval of information of various types. SP2:Apply the principles of human-computer interaction to the evaluation and construction of a wide range of applications including user interfaces, web pages, multimedia systems, and mobile systems. SP3:Deploy effectively the tools used for the development and documentation of software. SP4:Manage computing equipment, software systems and projects effectively and efficiently. SP5:Integrate IT-based solutions into user environment. SP6:Make clear presentations to a range of audiences about technical problems and their solutions.

Category	Expected Learning Outcomes
Interpersonal Skills (SI)	SI1:Act professionally, legally and ethically in the work environment. SI2:Display effective team work.
Attitude (A)	A1: Adapt to and work in a multicultural and global IT environment. A2: Work effectively as a member of a team. A3: Demonstrate commitment to lifelong learning self and professional development. A4: Demonstrate self-confidence and ability to adapt to new situations.

4.4 Translating Learning Outcomes in the Basic Phase of Information Technology Programme

A Bachelor degree programme in IT will be divided into:

i) **The Basic/Foundation Phase:** This phase is common for all IT programme students and consists of core and supporting knowledge areas.

ii) **The Specialization Phase:** This allows students to choose certain specializations according to their interests. The programme will be organized in courses.

In the basic/ foundation phase, three types of areas can be distinguished as follows:

- **Core Courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline. The core courses are the backbone of the discipline. They are the typical IT courses mandatory for every student.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses, students will face difficulties in acquiring required competencies in the core knowledge areas.
- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student, to deepen or broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 11 presents the core knowledge areas and supporting knowledge areas in the basic phase of a Bachelor of Science in IT programme. Based on the core and supporting knowledge areas in 11, the following remarks apply:

i) The core knowledge areas refer to the knowledge and skills that every student in all IT degree programmes should attain;

ii) The titles of the courses may differ from one university institution to another. The emphasis should be on the content rather than title;

iii) The autonomy and the uniqueness of the university institution will be taken into consideration in formulating the core courses for the basic phase. The university institutions will have the choice to add their own courses beyond the core and supporting courses; and

iv) The knowledge areas are not courses and the core components do not constitute a complete curriculum. Each IT programme may choose to cover the core knowledge units in a variety of ways.

The goal of designing the IT programme benchmarks is to keep the required component of the body of knowledge as small as possible. This is done to allow programmes in IT to be as flexible as possible since programme goals or objectives vary widely from programme to programme. Therefore, IT curriculum designers are flexible to formulate knowledge units under the corresponding knowledge areas.

Table 11: Core and supporting knowledge areas for Information Technology programme

Core Knowledge Areas	Supporting Knowledge Areas
• Information Technology Fundamentals • Systems Security • Human Computer Interaction • Computer Programming • Information Management • Computer Networks and Communication • Platforms Technologies	• Mathematics and Statistics • General Studies

• Systems Integration and Architecture • Web Systems and Technologies • Preparation for Professional Practice • Emerging Technologies	

4.5 Credit Framework for Information Technology Programme

The credit framework for the Bachelor of Science in IT will be as presented in Table 12. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme.

Table 12: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminars and Tutorials	% on Assignment	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 360 as minimum cumulative credits required for a learner to graduate with a three-year UQF level 8 programme such as the Bachelor of Science in IT. It should be noted that, for a knowledge area, IT programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn programming, more weight should be assigned to the practical activity.

4.6 Normal Learning Matrix for Information Technology Programme

The Normal Learning Matrix in Table 13 is a result of environmental scanning of IT for higher education minimum standards for Bachelor of Science in IT. The matrix depicts varying credit weights to each knowledge area. A university institution should design an IT programme such that a student covers a minimum of credit hours in three years for core knowledge areas as indicated in Table 13.

Table 13: Normal Learning Matrix with minimum standards core credits

Core Knowledge Areas	
Information Technology Fundamentals [18] • Information Technology • Systems Organization	**Systems Security [21]** • Network Security • Information Security • Software Security • Digital Forensic • Organization Security • Human/personnel and Societal Security
Human Computer Interaction [16] • Human Computer Interaction • Multimedia Systems Development • Interactive Systems Development • Human Cognitive Skills	**Computer Programming [27]** • Programming fundamentals • Procedural Programming • Object Oriented Programming • Logic Programming • Scripting Concepts
Information Management [32] • Information Systems Fundamentals • Database Management Systems • Distributed Systems • Data Mining • Data Warehousing • System Analysis and Design • Multimedia Information Systems • Digital Libraries • Managing Information and Systems	**Computer Networks and Communication [40]** • System Administration • Data Communication • Mobile Computing • Wireless Communication • Database Administration • Network Management • Computer Networks • Networking Router and Routing • Protocol of Enterprise Networks
Platforms Technologies [21] • Computer Organization and Architecture • Hardware Implementation	**System Integration and Architecture [27]** • Software Engineering • Software Acquisition and

Technologies • Computer Hardware and Maintenance • Computing Techniques • Operating Systems • Open-Source Technologies	Implementation • Software Testing • Cloud Computing
Web Systems and Technologies [18] • Web Design • E-commerce • Data-Driven Websites • Web Software Tools • E-governance **Emerging Technologies [16]** • Internet of Things • Artificial Intelligence • Data Science • Blockchain • Virtual Reality	**Preparation for Professional Practice [36]** • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and Organizational Behaviour • Final Year Project • Project Management • Intellectual property and legal issues
Supporting Knowledge Areas	
Mathematics and Statistics [21] • Probability and Statistics • Discrete Mathematics • Computer Simulation and Modelling	**General Studies [21]** • Communication Skills • Development Perspectives

The minimum standard to deliver lectures in the knowledge area of IT is proposed to be two (02) hours per week. It should be noted that an equal learning weight of two (02) hours has been placed on delivering practical sessions in the knowledge area of IT. Therefore, for a typical 15-week semester, a course in IT would take 30 hours for lectures and 30 hours for practical sessions per semester. Again, a course can have one (01) hour for tutorial and the other for assignment and self-study. In a course of three years, university institutions may prepare a minimum of four (04) courses in Information Management (IM) to ensure all theories, concepts, and practical sessions are covered. For instance, an IT curriculum with four (04) courses in the Information Management knowledge area would require a minimum of 320 hours as shown in Table 14.

Where:

IM1 represents Relational Database

IM2 represents Information Systems Fundamentals

IM3 represents Multimedia Information Systems

IM4 represents Systems Analysis and Design

Table 14: Sample minimum credits for Information Management as the knowledge area

Knowledge Units	**# of Courses**	**Lectures Hrs.**	**Seminars and Tutorials Hrs.**	**Assignment Hrs.**	**Independent Studies and Research Hrs.**	**Practical Training Hrs.**	**Total Hours**	**Credits**
% of Learning Activity		30	20	10	10	30		
IM1	1	30	15	2.5	2.5	30	80	8
IM2	1	30	15	2.5	2.5	30	80	8
IM3	1	30	15	2.5	2.5	30	80	8
IM4	1	30	15	2.5	2.5	30	80	8
Total minimum credit for Information Management							**320**	**32**

The proposed minimum standard consists of 11 core knowledge areas that form the foundation for a bachelor in IT. Table 13 outlines the minimum credits for a bachelor in IT. The minimum credits for each of the 11 knowledge areas have been formulated following the same process as for IT foundation in Table 14. A minimum of at least 46 credits have been left open for learners to pursue several electives in the span of three years. The credit distributed as 320 minimum core credits and at least 46 minimum elective credits make up 360 credits which is the minimum cumulative credits requirement to graduate with a UQF level 8 award in three years such as the Bachelor of Science in IT.

While an IT programme learning matrix may differ from one university to another in terms of number of courses offered and weight of learning activities, more credit weight must be directed towards knowledge units and courses that distinguish IT graduates from other graduates.

4.7 Information Technology Course Description

All courses to be taught and learnt within the programme should follow the outline in Table 15. Universities may use numbering for listing purposes.

Table 15: Course outline for Information Technology programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer questions such as, "What is the purpose of the course?", "What is the course trying to achieve?"
Course Expected Learning Outcome (s)	Course Learning Outcomes should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit framework
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

4.8 Learning Outcomes and Curriculum Alignment Matrix for Information Technology Programme

To check if the knowledge area covers the learning outcomes, it is important to develop a curriculum alignment matrix. For each knowledge area, one must formulate the specific learning outcomes for that course and have to check how much the course contributes to the programme learning outcomes. Table 16 highlights an example of a curriculum alignment matrix for the ELOs of a Bachelor degree in IT whereby K, SC, SP, SI and A stands for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal, and Attitude, respectively.

Table 16: Curriculum alignment matrix for Information Technology programme

Core Knowledge Areas	K1	K2	K3	K4	K5	K6	SC1	SC2	SC3	SC4	SC5	SC6	SP1	SP2	SP3	SP4	SP5	SP6	SI1	SI2	A1	A2	A3	A4
Information Technology Fundamentals	x	x	x	x						x														
Human Computer Interaction	x	x	x	x	x	x	x		x	x	x	x		x										
Systems Security	x	x	x	x						x	x													
Information Management	x	x	x	x	x			x		x	x		x	x				x						
Computer Networks and Communica tion	x	x		x		x						x	x	x		x	x							
Computer Programming	x	x		x	x	x	x	x	x	x	x	x			x	x								
Platform Technologies	x	x	x	x	x				x	x						x								
System Integration and Architecture	x	x		x				x		x		x				x	x	x						

Core Knowledge Areas	K1	K2	K3	K4	K5	K6	SC1	SC2	SC3	SC4	SC5	SC6	SP1	SP2	SP3	SP4	SP5	SP6	SI1	SI2	A1	A2	A3	A4
Web Systems and Technologies	x	x	x	x						x			x	x			x							
Preparation for Professional Practice	x	x	x	x	x	x	x	x	x	x		x		x	x		x	x	x	x	x	x	x	x
Emerging Technologies	x	x	x	x	x	x	x	x	x	x		x		x	x	x	x	x	x					
Supporting Knowledge Areas																								
General Studies	x				x													x	x	x	x	x	x	x
Mathematics and Statistics	x	x																						

4.9 Benchmarks for Specialization of Information Technology Programme

After the basic phase, students may choose to deepen their knowledge in one of the following specializations:

i) Programming and Software Development;

ii) Information Systems Engineering;

iii) Networking;

iv) Information Security;

v) Web Systems and Technologies;

vi) Multimedia and Computer Graphics;

vii) IT Audit Business Information Technology; and

viii) Content Management.

4.10 Minimum Entry Qualification for Information Technology Programme

Entry requirements for admission into the Bachelor of Science in Information Technology programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have sufficient background in Mathematics and Physics. For equivalent qualifications, a first-class Diploma in IT from a recognised institution.

4.11 Information Technology Programme Duration

The BSc. IT programme is a three-year programme where all courses require the equivalent of five to seven weekly contact hours for a semester of fifteen weeks. The fact that the focus of the programme is on computing in general and on Information Technology in particular, does not exempt it from exposing the students to general education courses from other fields of knowledge such as communication, entrepreneurship, law, organizational behaviour, etc.

4.12 Graduation Requirements for Information Technology Programme

At the end of the programme, an IT learner is required to complete the minimum required credits for a UQF level 8 three-year programme which should be not less than 360. An IT learner is required to accumulate a minimum of 360 credits which accounts to 3,600 notional hours of learning to graduate. University institutions should ensure that learners undertake one practical training within each academic year except their last year where they will take the final year project.

4.13 Evaluation and Course Assessment for Information Technology Programme

4.13.1 Evaluation

For implementation, a department should be ready to offer new general IT courses to the university or college and the community. After the course has been offered, its design and implementation should be carefully reviewed and evaluated. The data needed for assessment can be collected in a number of ways: written student evaluations, in-class observations, and personal interviews with students and faculty from the client departments. Once the course has been taught for a few years, it is also a good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of the graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

4.13.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes, quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

4.14 Resource Requirements for Information Technology Programme

4.14.1 Human Resource Requirements

Competent staff members are vital to the strength of the IT programme. For each IT programme, academic staff need both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete the degree in a timely manner. The interests and qualifications of the staff members must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

Staff must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual staff members. The school/department should support continuing staff development. Given the rapidly changing technology, it is particularly critical that staff members have sufficient time for professional development and scholarly activities. Resources should be provided for staff to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organizations. The programme is enhanced significantly when staff acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. Staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed by the programme is influenced by factors such as the number of students in the programme, the number of required courses, the number of supporting and elective courses offered, and the teaching load of the faculty/department. A programme should have a minimum number of full-time staff members with primary commitment to the Information Technology programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty/department expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the staff members should span a range of interests in Information Technology. Additional staff will be needed to teach the supporting courses that provide foundation-level knowledge across the campus.

4.14.2 Computing Infrastructure Requirements

Computing infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the IT programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most freshmen enter college with computer resources, so network access plays a much more significant role (Lee, 2009). Therefore, network access should be available for faculty and students to use with their own computers. Students at different levels in the curriculum have different needs. Substantial resources must be provided to support the courses targeted to all students. More sophisticated resources are necessary for Information Systems minors and majors who are developing skills in computing and IS fundamentals. Specialized laboratories or access to specialized simulation software is needed for advanced students where group and individual projects are developed. Contemporary and emerging software development tools should be available to create the most current enterprise solutions.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital in a campus environment.

4.14.3 Laboratory Requirements

Systems require hardware and software for structured, open/public, and specialized laboratories. Students must have an opportunity to use learning materials in both structured and unstructured laboratories.

Hardware and software are rapidly changing and improving. It is critical that faculty and students have access to facilities reflecting an environment that graduates will be expected to use professionally. All computing systems should be kept current. A plan should exist to continuously upgrade and/or replace software and equipment in a timely manner. The rate of change in technology suggests a rapid replacement cycle, with some technologies reaching obsolescence in less than 12 months.

Simulation software is becoming more prevalent for teaching advanced IT topics. This can include simulations for using applications to manage single workstations to complex enterprise-level networks. Many companies including Microsoft, Cisco, and even the textbook companies have developed sophisticated simulation software that does not require the latest equipment. Various courses and areas of study have their own specialized requirements, such as the large database with realistic sample data that are needed for effective work in the area of data management.

Students should be provided opportunities to work together on team-oriented projects. The group skills developed in this mode are critical to a successful Information Technology professional. Technological support, such as groupware, is expected for group and team activities.

All laboratories must possess adequate technical support in terms of professional staff to provide for installation and maintenance of the equipment. The staff should be proficient in both the hardware and software applications. Complete documentation must also be available. Laboratories should be able to support the following types of functions:

i) Structured Laboratories

A structured laboratory is a closed, scheduled, supervised experience in which student's complete specified exercises. An instructor who is qualified to provide necessary support and feedback to the students provides supervision. Exercises are designed to reinforce and complement the lecture material.

ii) Open/Public Laboratories

Student ownership of computers has continued to increase. However, laboratories remain essential for those students who do not own a computer and for providing additional resources not available on personal machines.

iii) Specialized Laboratories/Studios

Laboratory facilities are necessary to support team projects and special computing environments. Special facilities may be needed for systems development, network infrastructure, and other advanced technologies.

4.14.4 Classrooms

Suitable classroom facilities, equipped with information technology teaching resources, should be provided. A computing system with multimedia facilities is necessary for demonstrating the development, implementation, and application of information technology as well as conducting walkthroughs and making presentations. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

4.14.5 Library

Library support is an important part of an academic programme. It is especially important for disciplines with the rapid development of knowledge such as the Information Technology field. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the computing professional societies.

CHAPTER 5

BENCHMARKS FOR A BACHELOR OF SCIENCE IN INFORMATION SYSTEMS PROGRAMME

5.1 Information Systems Programme

Professionals in the Information Systems (IS) discipline are primarily concerned with the information that computer systems can provide to aid an enterprise in defining and achieving its goals, and the processes that an enterprise can implement or improve using information technology. Information Systems focuses on the information aspects of information technology. Information technology is the complement of that perspective: its emphasis is on the technology itself more than on the information it conveys. IT programmes *exist to produce graduates who possess the right combination of knowledge and practical, hands-on expertise to take care of both an organization's information technology infrastructure and the people who use it* (CC 2005, p. 14).

5.2 Information Systems Programme Objectives

The Bachelor of Science in IS programme should be designed in such a way that it produces graduates that meet the concern of the academia, government, industry and society as a whole. This can be achieved by focusing on the following grouped programme objectives.

5.2.1 Academic Ability

The programme objectives under this category are to equip students with knowledge/skills/ability to:

i) Analyse complex problems and synthesize solutions to those problems, develop an effective way to solve IS problems;

ii) Design and implement software and software technologies;

iii) Adapt and adopt emerging/evolving IS technologies;

iv) Undertake research and to progress to higher levels of studies;

v) Plan, design, deploy and document appropriate security for computer systems;

vi) Design, implement, and evaluate a computer-based system, process, component, or programme to meet desired needs;

vii) Verify, troubleshoot, test and analyse an existing computer-based system, process, component or programme; and

viii) Demonstrate proficiency in problem-solving techniques using the computer.

5.2.2 Academia Industrial Linkage/Employability

The programme objectives under this category are to equip students with knowledge and skills that match with the industrial needs as follows:

i) Problem-solving skills for computer related works;

ii) Up-to-date Information Systems skills for multiple industries (public as well as private sector);

iii) Analytical skills to understand impacts of information and Information Systems on individuals, organizations and society;

iv) Ability to integrate theory and practice to work effectively and efficiently in organizations; and

v) Knowledge and skills that enable creativity, innovativeness and entrepreneurship in the field of Information Systems.

5.2.3 Personal Development

The programme objectives under this category are to:

i) Prepare learners for lifelong learning and research;

ii) Empower students to progress in their personal career;

iii) Impart professional ethics to the learner;

iv) Equip the learner with skills and attitude to work in multicultural and global environments;

v) Equip the learner with knowledge and skills to work as a team in the computing field; and

vi) Enable the learner to develop skills to perform effectively in technical and nontechnical environments.

5.3 Expected Learning Outcomes for Information Systems Programme

IS specific knowledge and skills are divided into four main categories (and sub-categories) as follows:

i) Identifying and designing opportunities for IS-enabled organizational improvement such as ensuring alignment between IS strategy and organizational strategy and improving organizational processes with information technology solutions;

ii) Analysing trade-offs such as identifying and designing high-level solution and sourcing options and analysing and documenting the feasibility of various options;

iii) Designing and implementing Information Systems solutions, including identifying, evaluating, and procuring detailed solution and sourcing options; configuring and integrating organizational solutions using packaged solutions; and

iv) Managing ongoing information technology operations including managing the use of enterprise technology resources.

IS learning outcomes are itemized in Table 17.

Table 17: Expected Learning Outcomes for Information Systems programme

Category	Expected Learning Outcomes
	Graduates should be able to:
Knowledge (K)	K1. Demonstrate knowledge and understanding of concepts, principles and theories of Information Systems and software applications, computer science and information and digital technology. K2. Demonstrate knowledge and understanding of concepts and principles of leadership, management, communication, critical thinking and mathematics. K3. Demonstrate knowledge and understanding of concepts, principles and theories of business models, business process design and management and organizational theory. K4. Demonstrate knowledge and understanding of analytical and critical thinking.

Category	Expected Learning Outcomes	
Skills		
Cognitive Skills (SC)	SC1	Design opportunities for IS-enabled organizational improvement.
	SC2	Design the role of Information Systems in managing organizational risks and establishing controls.
	SC3	Identify and exploit opportunities created by emerging technology innovations.
	SC4	Design enterprise architectures and solutions that provide a high-quality user experience.
	SC5	Identify and evaluate detailed solution and sourcing options.
	SC6	Design secure systems, data infrastructures, and information models.
	SC7	Design applications, application architectures and integrated systems.
Practical Skills (SP)	SP1:	Implement Information Systems in managing organizational risks and establishing controls.
	SP2:	Implement enterprise architectures and solutions that provide a high-quality user experience.
	SP3:	Implement secure systems and infrastructures application architectures and integrated systems.
	SP4:	Implement data and information models.
	SP5:	Manage and exploit organizational data and information and systems development/procurement resources.
Interpersonal Skills (SI)	SI1:	Lead, ensure collaboration and working in teams including leading cross-functional global teams, managing globally distributed systems and projects, working effectively in diverse teams and Structuring organizations effectively.
	SI2:	Communicate proficiently with eloquence and in multiple languages with ability to listening, observing, interviewing, and analysing archival materials, Writing memos, reports, and documentation.
	SI3:	Negotiate with skills mindful of national interests, users, funding, resources of time, staff, and features.

Category	Expected Learning Outcomes	
		Negotiate with providers about service levels, about quality and performance of deliverables.
Attitude (A)	A1:	Adapt to and work in a multicultural and global IS environment.
	A2:	Demonstrate awareness and understanding of the ethical standards of IS profession.
	A3:	Demonstrate commitment to lifelong learning self and professional development.
	A4:	Demonstrate self-awareness and ability to adapt to new situations.

5.4 Translating Learning Outcomes in the Basic Phase of Information Systems Programme

A Bachelor degree programme in IS will be divided into:

i) **The Basic/Foundation Phase:** This phase is common for all ICT programme students and consists of core and supporting knowledge areas.

ii) **The Specialization Phase:** This allows students to choose certain specializations according to their interests. The programme will be organized in courses.

In the basic/ foundation phase, three (03) types of areas can be distinguished as follows:

- **Core Courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline. The core courses are the backbone of the discipline. They are the typical IS courses mandatory for every student.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses, it will be difficult to understand the core knowledge areas.
- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student, to deepen or broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 18 shows the core and the supporting knowledge in the basic phase of the Bachelor of Science in Information Systems programme. Based on the core and supporting courses, the following should be noted:

i) The titles of the courses may differ from one university to another. The emphasis should be in the content rather than in the title.

ii) The autonomy and the uniqueness of the university will be taken into consideration in formulating the core courses for the basic phase. A specific university institution will have a choice to add its own courses beyond the core and supporting courses.

iii) The core and supporting courses may be designed in the form of courses.

Table 18: Core and supporting knowledge areas for Information Systems programme

Core Knowledge Areas	Supporting Knowledge Areas
• Computer Programming • Computer Networks and Communication • Information Management • IS Project Management • Systems Analysis and Design • Preparation for Professional Practice • Platforms Technologies • Systems Security • Human Computer Interaction • Emerging Technologies	• General Studies • Mathematics and Statistics

5.5 Credit Framework for Information Systems Programme

The credit framework for the Bachelor of Science in IS programme will be as presented in Table 19. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should, normally, be designed as presented in Table 19.

Table 19: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminar and Tutorial	% on Assignment	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

Minimum cumulative credit required for a learner to graduate with a Bachelor of Science in IS in three years is 360. It should be noted that for a particular knowledge area, IS programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn programming, more weight should be assigned to the practical activity. This is recommended for every knowledge area.

5.6 Normal Learning Matrix for Information Systems Programme

The following sample of learning matrix is a result of environmental scanning of IS for higher education minimum standards for Bachelor of Science in IS. The matrix depicts varying credit weight to each knowledge area. A university should design an IS programme such that a student covers a minimum of 110 credits per year or 320 credit hours in three years for core knowledge areas. The proposed minimum standard consists of 10 core knowledge areas that form the foundation for a Bachelor in IS. Table 20 outlines the minimum credits for a Bachelor in IS. The minimum credits for each of the 10 knowledge areas have been formulated following the same process as for Systems Security as the knowledge area in Table 21. A minimum of at least 40 credits have been left open for learners to pursue several electives in the span of three years. The credit distributed as 317 minimum core credits and at least 40 minimum elective credits make up 360 credits which is the minimum cumulative credits requirement to graduate in a three-year UQF level 8 programmes such as the Bachelor of Science in IS.

While the IS programme learning matrix may differ from one university to another, in terms of number of courses offered and weight of learning activities, more credit weight must be directed towards knowledge areas and courses that distinguish IS graduates from other graduates.

Table 20: Normal Learning Matrix with minimum standards core credits

Core Knowledge Areas	
Computer Programming [34] • Programming fundamentals • Procedural/OOP/logical • Object-oriented Programming • Logical Programming • Internet and Application Programming • Scripting Concepts	**Computer Networks and Communication [21]** • Computer Networks • Enterprise Networks • Computer Administration and Management
Information Management [32] • Data and Data Structures • Database Management Systems • Data and Information Modelling Conceptual and Logical Level • Distributed Systems • Decision Support Systems • Data Analysis	**IS Project Management [40]** • Project Implementation • Enterprise Architecture Frameworks • Systems Integration • Content Management • Information Systems Management • Strategic Alignment • IT Risk Management • Information Systems Economics
Systems Analysis and Design [28] • Systems Philosophy and Approaches • Business Process Design and Management • Systems Requirements • Configuration and Change Management • High level System Design Issues	**Preparation for Professional Practice [36]** • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and Organizational Behaviour

• System Deployment and Implementation	• Final Year Project • Business Project Management • Intellectual property and legal issues
Platforms Technologies [21] • Open-Source Technologies • Operating Systems • Computer Architecture and Organization • Mobile Technologies	**Systems Security [27]** • Network Security • Information Security • Software Security • Digital Forensics • Organization Security • Human/Personnel and Societal Security
Human Computer Interaction [21] • Imaging Technologies • Interactive Computing • Human Cognitive Skills	**Emerging Technologies [21]** • Internet of Things • Artificial Intelligence • Data Science • Blockchain • Virtual Reality
Supporting Knowledge Areas	
General Studies [21] • Communications Skills • Development Perspectives	**Mathematics and Statistics [15]** • Linear Algebra • Probability and Statistics

The minimum standard to deliver lectures in the knowledge area of IS is proposed to be two (02) hours per week. It should be noted that equal learning weight of two (02) hours has been placed on delivering practical knowledge in the knowledge area of IS. Therefore, for a typical 15-week semester, it can be said that a course in IS would take 30 hours for lectures and 30 hours for practical per semester. A course can have one (01) hour for tutorial and another for assignment and self-study. In a course of three years, universities may prepare a minimum of 10 core knowledge areas in Systems Security to ensure all theories, concepts and practical are covered. For instance, an IS curriculum with three (03) courses in the Systems Security knowledge area would require 270 hours. As shown in Table 21, the minimum credits to deliver knowledge in the Systems Security (SS) is 27.

Where:

SS1 Data Security

SS2 Software Security

SS3 Digital Forensics

Table 21: Sample minimum credits for Systems Security as the knowledge area

Knowledge Units	# of Courses	Lectures Hrs.	Seminars and Tutorials Hrs.	Assignment Hrs.	Independent Studies and Research Hrs.	Practical Training Hrs.	Total Hours	Credits
% of Learning Activity		30	20	10	10	30		
SS1	1	30	15	7.5	7.5	30	90	9
SS2	1	30	15	7.5	7.5	30	90	9
SS3	1	30	15	7.5	7.5	30	90	9
Total minimum credit for Systems Security							**270**	**27**

5.7 Information Systems Course Description

The IS course description outlines all courses or modules to be taught and learnt within the programme in the entire period of three years. This will be aligned with the course matrix specified in section 5.6. Each course should follow the outline shown below. All courses or modules to be taught and learnt within the programme should follow the outline in Table 22.

Table 22: Course outline for Information Systems programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, "What is the purpose of the course?", "What is the course trying to achieve?"
Course Expected Learning Outcome (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit framework
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

5.8 Learning Outcomes and Curriculum Alignment Matrix for Information Systems Programme

To check if the knowledge area covers the learning outcomes, it is important to develop a curriculum alignment matrix. For each knowledge area, one must formulate the specific learning outcomes for that course and have to check how far the course contributes to the programme learning outcomes. Table 23 gives an example of a curriculum alignment matrix for the expected learning outcomes of the Bachelor of Science in IS. For each knowledge area, the contribution to the expected learning outcomes is given. As shown in Table 23, K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal and Attitude, respectively.

Table 23: Curriculum alignment matrix for Information Systems programme

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**SC1**	**SC2**	**SC3**	**SC4**	**SC5**	**SC6**	**SC7**	**SP1**	**SP2**	**SP3**	**SP4**	**SP5**	**SI1**	**SI2**	**SI3**	**A1**	**A2**	**A3**	**A4**
Computer Programming	x			x	x		x																
Computer Networks and Communication	x								x					x									
Information Management	x	x	x	x	x	x	x		x		x	x	x		x	x							
IS Project Management	x	x	x	x		x		x			x	x	x	x		x	x	x	x	x			
Systems Analysis and Design	x	x	x	x		x			x		x		x	x	x	x							
Platform Technologies	x		x	x	x		x	x	x	x	x	x	x		x	x							
Systems Security	x				x		x					x	x		x	x							
Emerging Technologies	x	x	x	x	x		x	x	x	x	x	x	x		x	x							
Preparation for Professional Practice	x	x	x	x		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Human Computer Interaction																							
Supporting Knowledge Areas																							
General Studies	x																x	x	x	x	x	x	x
Mathematics and Statistics	x	x		x																			

5.9 Benchmarks for Specialization of Information Systems Programme

After the basic phase, students may choose to deepen their knowledge in one of the following specializations:

i) Information Security and Assurance;

ii) Management Information Systems;

iii) IT Systems Audit/Analysis;

iv) Multimedia Technologies and Animations;

v) Content Development;

vi) Health Information Systems; and

vii) Business Information Systems.

5.10 Minimum Entry Qualification for Information Systems Programme

Entry requirements for admission into the Bachelor of Science in Information Systems programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have background in Mathematics or Physics or Chemistry or Economics, or Computer Science and for equivalent qualification, a Diploma in CS or IT or IS, or TE, or EE or ICT from a recognised institution.

5.11 Information Systems Programme Duration

Information Systems is a three (03) years programme where all modules require the equivalent of five to seven weekly contact hours for a semester of fifteen weeks. The fact that the focus of the programme is on computing in general and on information systems in specific does not exempt it from exposing its students to general education modules from other fields of knowledge such as communication, entrepreneurship, law, organizational behaviour, etc.

5.12 Graduation Requirements for Information Systems Programme

At the end of the programme, an IS learner is required to complete the minimum required credits for a UQF level 8 three-year programme which should be not less than 360. An IS learner is required to accumulate a minimum of 360 credits which accounts to 3,600 notional hours of learning to graduate. University institutions should ensure that learners undertake one practical training within each academic year except their last year where they will take the final year project

5.13 Evaluation and Course Assessment for Information Systems Programme

5.13.1 Evaluation

For implementation, a department should be ready to offer new general IS courses to the university or college and the community. This leaves only the final step in the course development process and assessment. After the course has been offered, its design and implementation should be carefully reviewed and evaluated. The data needed for assessment can be collected in a number of ways: written student evaluations, in-class observations, and personal interviews with students and faculty from the client departments. Once the course has been taught for a few years, it is good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of the graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

5.13.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, projects, etc. The summative mode involves tests, and examinations.

5.14 Resource Requirements for Information Systems Programme

The human resource for the IS degree programmes keeps changing with time. In addition to human resource, other resources needed for an IS degree programme include internet access, laboratories and library services. In a rapidly changing technical environment, students should be exposed to a variety of up-to-date hardware and software systems that adequately represent the professional setting in which they will be employed.

5.14.1 Human Resource Requirements

Competent staff members are vital to the strength of the Information Systems programme. For each IS programme, academic staff need both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete the degree in a timely manner. The interests and qualifications of the staff members must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

Staff must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual staff members. The school/department should support continuing staff development. Given the rapidly changing technology, it is particularly critical that staff members have sufficient time for professional development and scholarly activities. Resources should be provided for staff to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organizations. The programme is enhanced significantly when staff acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. Staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed by the programme is influenced by factors such as the number of students in the programme, the number of required courses, the number of supporting and elective courses offered, and the teaching load of the faculty/department. A programme should have a minimum number of full-time staff members with primary commitment to the IS programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty/department expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the staff members should span a range of interests in IS. Additional staff will be needed to teach the supporting courses that provide foundation-level knowledge across the campus.

5.14.2 Computing Infrastructure Requirements

Computing infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the IS programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most freshmen enter college with computer resources, so access plays a much more significant role (Lee, 2009). Therefore, network access should be available for faculty and students to use with their own computers. Students at different levels in the curriculum have different needs. Substantial resources must be provided to support the courses targeted to all students. More sophisticated resources are necessary for Information Systems minors and majors who are developing skills in computing and IS fundamentals. Specialized laboratories or access to specialized simulation software is needed for advanced students where group and individual projects are developed. Contemporary and emerging software development tools should be available to create the most current enterprise solutions. In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital necessity in a campus environment.

5.14.3 Laboratory Requirements

Systems require hardware and software for structured, open/public, and specialized laboratories. Students must have an opportunity to use learning materials in both structured and unstructured laboratories.

Hardware and software are rapidly changing and improving. It is critical that staff and students have access to facilities reflecting an environment that graduates will be expected to use professionally. All computing systems should be kept current. A plan should exist to continuously upgrade and/or replace software and equipment in a timely manner. The rate of change in technology suggests a rapid replacement cycle, with some technologies reaching obsolescence in less than 12 months. In some cases, simulation software is becoming more prevalent for teaching advanced IS topics. This can include simulations for using applications to manage single workstations to complex enterprise-level networks. Many companies including Microsoft, Cisco, and even the textbook companies have developed sophisticated simulation software that does not require the latest equipment. Various courses and areas of study have their own specialized requirements, such as the large database with realistic sample data that are needed for effective work in the area of data management.

Students should be provided opportunities to work together on team-oriented projects. The group skills developed in this mode are critical to a successful Information Systems professional. Technological support, such as groupware, is expected for group and team activities. All laboratories must have adequate technical support in terms of professional staff to provide for installation and maintenance of the equipment. The staff should be proficient in both the hardware and software applications. Complete documentation must also be available. Laboratories should be able to support the following types of functions:

Structured Laboratories

A structured laboratory is a closed, scheduled, supervised experience in which student's complete specified exercises. An instructor who is qualified to provide necessary support and feedback to the students provides supervision. Exercises are designed to reinforce and complement the lecture material.

i) Open/Public Laboratories

Student ownership of computers has continued to increase. However, laboratories remain essential for those students who do not own a computer and for providing additional resources not available on personal machines.

ii) Specialized Laboratories

Laboratory facilities are necessary to support team projects and special computing environments. Special facilities may be needed for systems development, network infrastructure, and other advanced technologies.

5.14.4 Classrooms

Suitable classroom facilities, equipped with information technology teaching resources, should be provided. A computing system with multimedia facilities is necessary for demonstrating the development, implementation, and application of information technology as well as conducting walkthroughs and making presentations. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

5.14.5 Library

Library support is an important part of an academic programme. It is especially important for disciplines with rapid development of knowledge such as the Information Systems field. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the computing professional societies.

CHAPTER 6
BENCHMARKS FOR A BACHELOR OF SCIENCE IN COMPUTER ENGINEERING PROGRAMME

6.1 Computer Engineering Programme

Computer Engineering (CE) as an academic field encompasses the broad areas of electrical or electronics engineering and computer science. CE is a discipline that embodies the science and technology of design, construction, implementation, and maintenance of hardware and software components of modern computing systems and computer-controlled equipment.

6.1.1 Computer Engineering in the ICT Cluster

CE evolved from the disciplines of electrical engineering and computer science. Initial curricula efforts in CE commonly occurred as a specialization within electrical engineering programmes, extending digital logic design to the creation of small-scale digital systems and, eventually, to the design of microprocessors and computer systems. Later, curricula in CE increasingly began to include and finally evolved to integrate relevant knowledge areas from computer science. Today, that trend is diminishing, and CE programmes reflect their own knowledge areas.

One expects that the growth trend in CE will increase as computing and electronic technologies become more complex. The evolution may take many forms, including but not limited to:

i) Tighter integration with computer science and ICT;

ii) Collaboration with the software engineering programme on application-focused projects and embedded systems with a greater emphasis on design and analysis tools; and

iii) Re-integration with electrical engineering, as computer-based systems become dominant in areas such as control systems and telecommunications.

6.2 Computer Engineering Programme Objectives

The primary objective of the CE programme should be to prepare graduates for professional and supervisory positions in the field of computer engineering. CE programmes must include opportunities to develop the skills for engineering supporting high-tech computing environments, both in Tanzania and internationally. An important distinction must be made between computer engineers, electrical engineers, other computer professionals, and engineering technologists. In order to ensure distinction, graduate versatility and competence, every CE programme design should aim to produce graduates that meet the following specific objectives.

6.2.1 Academic Ability

Every CE programme should aim to equip learners with academic knowledge/skills/ability to:

i) Model, design, construct, operate, evaluate and maintain computers, computer-engineered systems, their hardware, software, network components as well as their integration to solve engineering problems; and

ii) Analyse a problem and then: a) identify and define the computing and engineering requirements appropriate to the problem solution; b) evaluate the extent to which the computer-engineered solution meets the criteria defined for its current use and scale up to future development.

6.2.2 Academia and Industrial Linkage / Employability

The programme objectives under this category are to equip learners with knowledge and skills that match with the industrial needs as follows:

i) Ability to integrate theory and practice to work effectively and efficiently in organizations;

ii) Knowledge and skills that enable creativity, innovativeness and entrepreneurship in the field of computing;

iii) Assess risk related and analyse the impact of local and global trends of Computer Engineered solutions on individuals, organizations, and society; and

iv) Demonstrate critical thinking, creativity and innovativeness in developing computer-engineered solutions to real world problems in societies and industries.

6.2.3 Personal Development

The programme objectives under this category are to:

i) Empower learners to progress in their personal career, impart professional ethics to the learner, equip the learner with skills and attitude to work in multicultural and global environments;

ii) Enable the learner to develop skills to perform effectively in technical and nontechnical environments, as a team member and at individual level; and

iii) Communicate effectively with experts and non-experts.

6.3 Expected Learning Outcomes for Computer Engineering Programme

Expected Learning Outcomes (ELOs) provide a mechanism for describing not just knowledge and relevant practical skills, but also personal and transferable skills. They describe what we expect a student will know or can do by the time of graduation. To harmonize the training of the Bachelor programme in CE, the following learning outcomes have been formulated to be used as benchmarks. The formulated ELOs as stipulated in Table 24 are the threshold: all graduates of the bachelor's programme in CE must achieve them. Universities may consider adding ELOs as and when necessary, in line with their mission and vision or other identified need(s).

Table 24: Expected Learning Outcomes in Computer Engineering programme

Category	Expected Learning Outcomes
	The graduate should be able to:
Knowledge (K)	K1. Demonstrate knowledge and understanding of concepts, theories and operations of a computer system, the design of the hardware and software for that system, and the processes involved in constructing, analysing, and maintaining it over the lifetime of the system. K2. Demonstrate knowledge and understanding of fundamental mathematics and engineering science consistent with problem solving abilities of a degreed professional in the CE field. K3. Demonstrate understanding of quality standards and

Category	Expected Learning Outcomes
	benchmarks in design and development of computers, computer-based systems and networks that involve both hardware and software.
Skills	
Cognitive Skills (SC)	SC1: Analyse the impact of local and global trends of CE solutions on individuals, organizations, and society. SC2: Demonstrate critical thinking, creativity and innovativeness in developing computer-engineered solutions to real world problems. SC3: Assess risk related to computer-engineered activities. SC4: Analyse a problem and then identify and define the computing and engineering requirements appropriate to the problem solution. SC5: Evaluate the extent to which a computer-engineered system meets the criteria defined for its current use and scale up to future development.
Practical Skills (SP)	SP1: Model, design, implement and evaluate computers, computer-engineered systems, their hardware, software, network components as well as their integration to solve engineering problems. SP2: Deploy a variety of computer-based and laboratory tools and properly interpret and report experimental results. SP3: Complete a sequence of design experiences, encompassing hardware and software elements and their integration, building on prior work, and including at least one major project. In this context, design refers to a level of ability beyond "assembling" or "configuring" systems. SP4: Evaluate, verify, trouble-shoot, test and analyse existing computer engineered systems, their hardware, software and network components. SP5: Identify and address engineering problems by extending the concepts of simple building blocks to system level design.

Category	Expected Learning Outcomes
Interpersonal Skills (SI)	SI1: Work effectively in a team. SI2: Communicate effectively with experts and non-experts. SI3: Demonstrate an understanding of professionalism, ethics, legal, security, social issues and responsibilities in CE.
Attitude (A)	A1: Adapt to, and work in a multicultural and global CE environment. A2: Work effectively as a team member as well as at individual level. A3: Demonstrate commitment to lifelong learning, self and professional development. A4: Demonstrate self-confidence and ability to adapt to new situations.

6.4 Translating Learning Outcomes in the Basic Phase of Computer Engineering Programme

In the basic/ foundation phase, three (03) types of areas can be distinguished as follows:

- **Core Courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline. The core courses are the backbone of the discipline. They are the typical CE courses mandatory for every student.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses, it will be difficult to understand the core knowledge areas.
- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student, to deepen or broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 25 shows the core and supporting knowledge areas in the basic phase of a Bachelor degree in CE programme. Based on these core and supporting knowledge areas, the following remarks apply:

i) The core knowledge areas refer to the knowledge and skills that every student in all CE degree programmes should attain;

ii) The titles of the courses may differ from one university to another. The emphasis should be on the content rather than title;

iii) The autonomy and the uniqueness of a university will be taken into consideration in formulating the core courses for the basic phase. Universities will have the choice to add their own courses beyond the core and supporting courses; and

iv) The knowledge areas are not courses and the core components do not constitute a complete curriculum. Each CE programme may choose to cover the core knowledge areas in a variety of ways.

The goal of designing this CE programme benchmark is to keep the required component of the body of knowledge as small as possible. This is done to allow CE programme to be as flexible as possible since programme goals or objectives vary widely from programme to programme. Therefore, CE curriculum designers are flexible to formulate data structures and algorithms as subject units under corresponding knowledge areas such as programming.

Table 25: Core and supporting knowledge areas for Computer Engineering programme

Core Knowledge Areas	Supporting Knowledge Areas
• Computer Engineering Foundation • Digital Design • Electrical and Electronics Measurements • Electronics Foundation • Computer Networks and Communication • Preparation for Professional Practice • Embedded Systems • Systems and Project Engineering • Software Design and Development • Computer Programming • Hardware Design and Implementation	• Mathematics • General Studies

Core Knowledge Areas	Supporting Knowledge Areas
• Control Systems • Information Management • Platforms Technologies • Systems Security • Emerging Technologies	

i) Applied Emerging Technologies

The identification and effects of applied emerging technologies on computer engineering is useful in producing a competent engineer who can contribute to the profession in a changing world. CE programmes should allow the exploration of applied emerging technologies. For example, CE instructors may encourage examination of ways in which 3D printers might produce artifacts that are harmful to the society or describe the challenges one would face in designing and producing integrated circuits. As another example, students should be able to explain ways in which nanotechnology or the internet of things (IoT) can transform the technological workplace. Computer engineers are already or will soon be interacting with optical, biological, or quantum computers or they will be designing a new-age robotic system for manufacturing. These newly emergent and modern technologies present challenges to computer engineering students and practitioners that could involve financial and ethical trade-offs affecting professional practice in a changing world.

ii) Conceptual Emerging Technologies

The CE programme should allow exploration of new inventions that are yet to emerge as viable technologies. For example, teachers might encourage exploration of ways in which a computer engineer would design environments involving augmented reality and virtual worlds or ways in which big data and data analytics might affect the work of a computer engineer. These activities would affect design constraints needed to address emerging areas such as computational biology and bioinformatics. Additionally, it would be useful to have CE learners explore the role of a computer engineer in an era of machine learning and intelligent systems or discuss engineering strategies needed in developing a culture of green computing and sustainability. New technologies might even expose safety issues affecting the field of computer engineering. Awareness of these and other issues are important in developing a well-rounded and social conscious computer engineer.

6.5 Credit Framework for Computer Engineering Programme

The credit framework for the Bachelor of Science in CE programme will be as presented in Table 26. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should, normally, be designed as presented in Table 26.

Table 26: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminars and Tutorials	% on Assignments	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 480 as minimum cumulative credits required for a learner to graduate with a four-year UQF level 8 programme such as the Bachelor of Science in CE. It should be noted that, for a knowledge area, CE programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn programming, more weight should be assigned to the practical activity. Universities offering the Bachelor of Science in Engineering should assign more weight on practical and industrial field activities as compared to theoretical learning.

6.6 Normal Learning Matrix for Computer Engineering Programme

The following sample learning matrix (Table 27) is a result of environmental scanning of CE programme curricula for higher education minimum standards for bachelor of CE. The matrix depicts varying credit weight to each knowledge area. A university should design a CE programme such that a student covers a minimum of credit hours in four years for core knowledge areas as shown in Table 28.

The minimum standard to deliver lectures in the knowledge area of CE is proposed to be two (02) hours per week. Equal learning weight of two (02) hours has been placed on delivering practical in the knowledge area of CE. Therefore, for a typical 15-week semester, a course in CE would take 30 hours for lectures and 30 hours for practical per semester. In a course of four years, universities may prepare a maximum of four (04) courses in Computer Organization and Architecture to ensure all theories, concepts and practical are covered and a firm CE foundation is built. For instance, a CE curriculum with two (02) courses in Computer Organization and Architecture knowledge area would require 60 hours for lectures. As shown in Table 27, the minimum credits to deliver knowledge in the Computer Engineering foundation area is 54.

Where:

CE1 represents Introduction to CE

CE2 represents Computer Organisation and Architecture

CE3 represents Computer Hardware and Maintenance

CE4 represents Microcomputer Systems

Table 27: Sample minimum credits for Computer Engineering Foundation as the knowledge area

Knowledge Units	# of Courses	Lectures Hrs.	Seminars and Tutorials Hrs.	Assign. Hrs.	Independent Studies and Research Hrs.	Practical Training Hrs.	Total Hours	Credits
% of Learning Activity		30	20	10	10	30		
CE1	1	30	15	7.5	7.5	30	90	9
CE2	2	60	30	15	15	60	180	18
CE3	1	30	15	7.5	7.5	30	90	9
CE4	2	60	30	15	15	60	180	18
Total minimum credit for Computer Engineering foundation							**540**	**54**

The proposed minimum standard consists of 15 core knowledge areas that form the foundation for a Bachelor in CE. Table 28 outlines the minimum credits for a Bachelor in CE. Minimum credits for each of the 15 knowledge areas have been formulated following the same process as for CE foundation in Table 27. At least 71 credits have been left open for learners to pursue several electives in the span of four years. The

409 minimum core credits and 71 minimum elective credits make up 480 credits which is the minimum cumulative credits requirement to graduate in a four-year UQF level programme such as the bachelor in CE.

While a CE programme learning matrix may differ from one university to another university in terms of number of courses offered and weight of learning activities, more credit weight must be directed towards knowledge areas and courses that distinguish CE graduates from other graduates.

Table 28: Normal Learning Matrix with minimum standard core credits

Core Knowledge Areas	
Computer Engineering Foundation [48] • Introduction to Computer Engineering • Computer Organization and Architecture • Computer Hardware and Maintenance • Microcomputer systems	**Digital Design [36]** • Digital Image Processing • Digital Signal Processing • Hardware Description Languages and Program Logic • Robotics and Automation
Electrical and Electronics Measurements [18] • Measurements and Instrumentation • Intelligent Instrumentation	**Electronics Foundation [18]** • Analogue Electronics • Digital Electronics • Very Large-Scale Integrated Circuits
Computer Networks and Communication [27] • Introduction to Computer Networks • Computer Network Design and Administration. • Network routing and Switching • Wide Area Networking • Optical Networks	**Preparation for Professional Practice [36]** • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and

<table>
<tr><td></td><td>Organizational Behaviour
Final Year Project
Engineering Project Management
Intellectual property and legal issues
Workshop Training
General Engineering Procedure and Ethics</td></tr>
<tr><td>Embedded Systems [18]
Fundamentals of Embedded Systems
Mobile and Networked Embedded Systems
Computing platforms for embedded systems</td><td>Systems Security [16]
Network Security
Information Security
Software Security
Digital Forensic
Organization Security
Human/Personnel and Societal Security
System and Component Security</td></tr>
<tr><td>Software Design and Development [36]
Object Oriented Analysis and Design
Mobile Applications Design and Development
Intelligent Systems Design
Software Engineering
Software Testing and Quality</td><td>Computer Programming [28]
Programming and Compiler Technology
Object Oriented Programming
Procedural Programming
Data Structures and Algorithms
Logical Programming
Assembly Programming</td></tr>
<tr><td>Hardware Design and Implementation [18]
Engineering Drawing
Computer Aided Drafting and Design</td><td>Control Systems [18]
Control Systems Engineering
Control Theory</td></tr>
</table>

Information Management [18] • Database Management Systems • Data Mining and warehousing	**Platform Technologies [16]** • Operating Systems • Open-Source Technology • Computer Architecture and Organization • Mobile Technologies
Emerging Technologies [16] • Internet of Things • Artificial Intelligence • Data Science • Blockchain • Virtual Reality	
Supporting Knowledge Areas	
Mathematics [21] • Calculus • Discrete Math • Linear Algebra • Probability and Statistics	**General Studies [21]** • Communication Skills for Engineers • Perspectives in Development

6.7 Computer Engineering Course Description

All courses or modules to be taught and learnt within the programme should follow the outline in Table 29.

Table 29: Course outline for Computer Engineering programme

Course Title	Refers to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, "What is the purpose of the course?", "What is the course trying to achieve?"
Course Expected Learning Outcome (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency

Course Status	Core or Elective
Credit Rating	Refers to the UQF credit framework
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

6.8 Learning Outcomes and Curriculum Alignment Matrix for Computer Engineering Programme

To check if the knowledge area covers the learning outcomes, it is important to develop a curriculum alignment matrix. For each knowledge area, one must formulate the specific learning outcomes for that course and must check how far the course contributes to the programme learning outcomes. Table 30 is an example of a curriculum alignment matrix for the ELOs of the Bachelor in CE. For each knowledge area, the contribution to the expected learning outcomes is given. As shown in Table 30, K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal and Attitude, respectively.

Table 30: Curriculum Alignment Matrix for Computer Engineering programme

Core Knowledge Areas	K1	K2	K3	SC1	SC2	SC3	SC4	SC5	SP1	SP2	SP3	SP4	SP5	SI1	SI2	SI3	A1	A2	A3	A4
Electronics Foundation	X		X						X	X		X								
Computer Engineering Foundation	X		X			X			X	X		X								
Computer Networks and Communication	X		X						X	X		X								
Preparation for Professional Practice			X	X	X	X						X	X	X		X	X	X	X	
Embedded Systems	X		X						X	X		X								
Systems Security	X					X				X		X								
Software Design and Development	X		X			X	X	X	X	X										
Computer Programming	X		X						X			X								
Information Management	X		X						X			X								
Platforms Technologies	X		X						X	X		X								
Systems and Project Engineering			X	X	X	X	X	X	X	X	X	X	X	X	X	X		X		

Core Knowledge Areas	**K1**	**K2**	**K3**	**SC1**	**SC2**	**SC3**	**SC4**	**SC5**	**SP1**	**SP2**	**SP3**	**SP4**	**SP5**	**SI1**	**SI2**	**SI3**	**A1**	**A2**	**A3**	**A4**
Digital Design	X		X				X	X	X	X										
Emerging Technologies	X		X				X	X	X	X	X	X								
Hardware Design and Development	X		X						X	X	X	X								
Supporting Knowledge Areas																				
Mathematics		X			X	X														
General Studies														X	X		X	X		X

6.9 Minimum Entry Qualification for Computer Engineering Programme

Entry requirements for admission into the Bachelor of Science in Computer Engineering programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have background in Physics and Mathematics and for equivalent qualification, a first-class Diploma in CE from a recognised institution.

6.10 Computer Engineering Programme Duration

A CE program typically reflects Electrical Engineering and Computer Science knowledge areas in varying weight coverage depending on the vision of the CE department. Due to the intensity of knowledge area coverage, four years of CE programme duration are recommended, with a minimum of two (02) semesters per academic year.

6.11 Graduation Requirements for Computer Engineering Programme

A CE learner is required to accumulate a minimum of 409 core credits which accounts to 4,200 notional hours of learning to graduate. At least 71 credits have been left open for the learner to pursue elective knowledge units. Universities should ensure learners undertake one (01) Practical Training within each academic year except the last academic year.

6.12 Evaluation and Course Assessment for Computer Engineering Programme

6.12.1 Evaluation

Universities should offer up-to-date courses to learners in the CE programme. Due to the rapid changes in the CE field and its emerging technologies, the design and implementation of curricula should be regularly reviewed and evaluated. The assessment can be done in several ways: written student evaluations, in-class observations, industrial symposium, and personal interviews with various stakeholders such as academia, alumni, and experts from ICT industries. Once the course has been taught for four 4 years, it is a good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of CE graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

6.12.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

6.13 Resource Requirements for Computer Engineering Programme

6.13.1 Laboratory Experience

Laboratory experiences are an essential part of the CE curriculum, and they serve multiple functions. It is important that CE students have many opportunities to observe, explore, and manipulate characteristics and behaviours of actual devices, systems, and processes. This includes designing, implementing, testing, and documenting hardware and software, designing experiments to acquire data, analysing and interpreting that data, and using that data to correct or improve the design and to verify that it meets specifications. Universities should seek options that best suit their needs based on available space, the subject unit objectives, and the resources available.

6.13.2 CE Laboratories

Many courses in CE should contain laboratory experiences. Typically, a laboratory experience lasts one to two hours and it occurs in a location configured with specialized equipment. The depth and breadth of these experiences will vary among institutions. The variation often depends on the time allocated within the curriculum, physical space, and resources. Table 31 illustrates the benchmarks recommendation on the types of laboratories students studying CE should experience. Some laboratories involve practices that all CE students must or should do. The laboratory type marked "**Must"** represents experience that are considered compulsory/necessary for every computer engineer while "**Should**" refers to experiences that are considered the best for every computer engineer. The one marked "**Supplemental**" is considered as additional experience for computer engineers.

Table 31: Types of Computer Engineering laboratories

Laboratory Type	Must	Should	Supplemental
Circuits and Electronics	√		
Computer Architecture Design			√
Digital Signal Processing			√
Digital Logic and System Design	√		
Embedded Systems	√		
Networking		√	
Software Design		√	
Final Year Project Design	√		
Computers in Manufacturing			√
Audio Engineering			√
Electricity Energy Systems			√
Graphics			√
Mechatronics			√
Microwave Measurements			√
Operating Systems			√
Robotics			√
Specialized Electronics Lab			√
Teaching Enhancement			√

Laboratories should include some physical implementation of designs such as electronic and digital circuits, bread-boarding, FPGAs/CPLDs, microcontroller-based systems, prototyping, and implementation of firmware. Laboratories should also include application and simulation software to design computer systems including digital systems. Simulation tools present intrinsic value as part of professional CE practice. They are useful in modelling real systems, and they are often desirable and necessary to allow students to study systems that are impractical to design and implement given available time and resources.

6.13.3 Software Considerations

Software tools and packages related to CE will vary based on the philosophy and needs of each programme. Table 32 suggests some software that could appear on all machines within specific laboratory settings. Products mentioned in the table are included for illustrative purposes only and that no endorsement of a specific product is implied. Additionally, it is not envisioned that any programme will incorporate every one of these software applications. Each programme should determine its own needs and consider including the most current version of appropriate applications.

Table 32: Suggestions for possible software applications

• Design Modelling and Simulation • Circuit-level (e.g., SPICE) • Gate-level (schematic entry) • Digital systems (e.g., VHDL, Verilog) • Analog/mixed-signal circuits • (e.g., VHDL-AMS, Verilog-AMS) • System-level design (e.g., System • Verilog, System C) • Software Development • Microcontroller Systems	• Integrated Circuits/ASIC Design • Systems Engineering Tools • General Computing/Productivity • Digital Hardware Prototyping • Microcontroller Systems Design • Mathematics Packages • Lab Automation and Instrumentation • Computer Aided Design and Modelling (CAD tools) • Printed Circuit Board (PCB) Design • Other tools

i) Open-ended Labs

The CE curriculum often contains open-ended experiences where true research and development takes place. One might view this as the "ultimate lab experience." A culminating or capstone design experience usually embodies this open-ended flavour. In such situations, an instructor and a team of students decide on an exploration area and, once decided, the student team begins the research and design process. Programmes usually provide a dedicated space where teams can meet and work. These spaces generally contain modern facilities and provide enough space for electronic devices (e.g., robots) and other equipment required by the project at hand.

ii) Embedded Labs

The traditional laboratory experience normally takes place in a room separate from the lecture and at a different time. For example, a graphics course might have lectures on Tuesday and Thursday, with a three-hour laboratory on Wednesday. It is increasingly common in modern educational settings to have laboratories embedded within ordinary courses. One such practice is to have lectures take place within the laboratory itself. Another is to partition a room in some manner and have the lecture take place in one part of the room with the laboratory and its associated equipment in another part of the same room. There is also an emerging trend to have "flipped classrooms" in which lectures are recorded for students to view in advance. Then class time is used to engage the students in active learning, which could include students bringing their own breadboards and instrumentation and engaging in laboratory exercise.

6.13.4 Human Resource Requirements

Competent staff members are vital to the strength of the CE programme. For each CE programme, academic staff need both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete the degree in a timely manner. The interests and qualifications of the staff members must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

Staff must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual staff members. The school/department should support continuing staff development. Given the rapidly changing technology, it is particularly critical that staff members have sufficient time for professional development and scholarly activities. Resources should be provided for staff to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organizations. The programme is enhanced significantly when staff acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. Staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed by the programme is influenced by factors such as the number of students in the programme, the number of required courses, the number of supporting and elective courses offered, and the teaching load of the faculty/department. A programme should have a minimum number of full-time staff members with primary commitment to the CE programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty/department expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the staff members should span a range of interests in CE. Additional staff will be needed to teach the supporting courses that provide foundation-level knowledge across the campus.

6.13.5 ICT Infrastructure Requirements

ICT infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the CE programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most incoming undergraduates enter college with computer resources, so access plays a much more significant role. Therefore, network access should be available for faculty and students to use with their own computers. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital necessity in a campus environment.

6.13.6 Library

Library support is an important part of an academic programme. It is especially important for disciplines with rapid changes of knowledge such as CE. Libraries should provide both traditional and digital access for journals, proceedings, monographs, and reference books wherever possible. The holdings should include access to digital journals and proceedings of the ICT professional societies.

CHAPTER 7

BENCHMARKS FOR A BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING PROGRAMME

7.1 Software Engineering Programme

Software Engineering (SE) is the study and application of a systematic, disciplined, quantifiable approach to the design, development, operation, and maintenance of software, and the study of these approaches. The systematic development and application of techniques normally leads to the creation of correct and reliable computer software systems for business processes automation. Knowledge and principles from other disciplines, such as computer science, computer engineering, and mathematical analysis, are employed by computer software engineers for designing, developing, testing, and evaluating the software and the systems that computers use to carry out various applications. Software engineers are engaged in analysing user requirements, and designing, developing, testing, and maintaining software. Graduates with such software engineering skills have a wide range of secure and financially rewarding careers awaiting in Tanzania and the global as a whole. The careers in this programme include Software developer, Software engineers, Data processing manager, Software quality assurance engineer, IT consultant, Internet engineer, Business analyst, Systems analyst, among others. Graduates of this programme will be able to understand and demonstrate the entire software engineering project process, which consists of object-oriented analysis, design, programming and testing.

7.2 Software Engineering Programme Objectives

The Bachelor of Software Engineering programme should be designed to build human capacity in the software engineering discipline in the region so as to accelerate the development of innovations in ICT that will address human challenges. It should be designed in such a way that it produces graduates who meet the concern of the academia, Government, industry and society as a whole. This can be achieved by focusing on the following groups of programme objectives:

7.2.1 Academic Ability

The programme objectives under this category are to equip students with:

i) Ability to analyse software engineering problems and synthesize solutions using appropriate methods of software analysis and design;

ii) Ability to pursue research careers;

iii) Ability to plan, design, deploy and document appropriate security for software and software technologies;

iv) Skills to design, implement, and evaluate a computer-based system, process, component, or programme to meet desired needs;

v) Ability to verify, troubleshoot, test and analyse an existing computer-based system, process, component or programme;

vi) Understanding of the principles of large-scale software systems, and the processes that are used to build them;

vii) Understanding of and ability to apply current theories, models, and techniques that provide a basis for the software lifecycle;

viii) Ability to develop an appreciation of the cost, quality, and management issues involved in software construction;

ix) Ability to apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment;

x) Acquisition of strong fundamental knowledge in science, mathematics, fundamentals of computer science, software engineering and multidisciplinary engineering to begin in practice as a software engineer; and

xi) Ability to design applicable solutions in one or more application domains using software engineering approaches that integrate ethical, social, legal and economic concerns.

7.2.2 Academia and Industrial Linkage / Employability

The programme objectives under this category are to equip students with knowledge and skills that match with the industrial needs as follows:

i) Ability to use tools and techniques necessary for software engineering practice;

ii) Problem-solving skills for software engineering related works;

iii) Ability to apply new software models, techniques and technologies to bring out creativity, innovative and novelistic solutions for the growth of the society in all aspects and evolving into their continuous professional development;

iv) Ability to deliver quality software products by possessing the leadership skills and demonstrating effective and modern working strategies by applying both communication and negotiation management skills;

v) Ability to integrate theory and practice to work effectively and efficiently in organizations; and

vi) Ability to design and communicate ideas about software system solutions at different levels of abstraction and have the opportunity to transfer such skills across a wide range of industrial and commercial domains.

7.2.3 Personal Development

The programme objectives under this category are to:

i) Enable the learner to develop skills to perform effectively in technical and nontechnical environments;

ii) Prepare learners for lifelong learning and research;

iii) Empower students to progress in their personal career; impart professional ethics to the learner; equip the learner with skills and attitude to work in multicultural and global environments; and

iv) Enable learners to work as individuals and as part of a multidisciplinary team to develop and deliver quality software.

7.3 Expected Learning Outcomes for Software Engineering Programme

Expected Learning Outcomes (ELOs) provide a mechanism for describing not just knowledge and relevant practical skills but also personal and transferable skills. ELOs are not of equal size and do not have a uniform mapping to curriculum hours; topics with the same number of hours may have quite different numbers of associated learning outcomes. Each learning outcome has an associated level of mastery. Some literature defines the levels based on Bloom's Taxonomy, which has been well explored within the computing field. The Bloom's Taxonomy specify the cognitive skill level as follows:

i) **Knowledge:** Remembering previously learned material. Test observation and recall of information; that is, "bring to mind the appropriate information".

ii) **Comprehension**: Understanding information and the meaning of material presented. For example, being able to translate knowledge to a new context, interpret facts, compare, contrast, order, and group, infer causes, predict consequences, and so forth.

iii) **Application**: Using learned material in new and concrete situations. For example, using information, methods, concepts, and theories to solve problems requiring the skills or knowledge presented.

However, for the mastery levels to be indicative and not to impose theoretical constraint on users of benchmarks, we categorize mastery level into Knowledge, Cognitive Skills, Practical Skills, Interpersonal Skills and Attitude. The outcomes for an undergraduate curriculum in SE are summarized in Table 33.

Table 33: Expected Learning Outcomes for Software Engineering programme

Categories	**Expected Learning Outcome**
	The graduate should be able to:
Knowledge (K)	K1: Demonstrate knowledge and understanding of a wide range of design and development principles and tools used in the construction of software systems of varying complexity. K2: Demonstrate knowledge and understanding of the software-development process, including requirements analysis, design, programming, testing and maintenance. K3: Demonstrate an understanding of appropriate theories, models, and techniques that provide a basis for problem identification and analysis, software design, development, implementation, verification, and documentation. K4: Demonstrate an ability to apply mathematical and software engineering foundations in the modelling and designing of software. K5: Demonstrate understanding of quality standards and benchmarks in computer software development. K6: Demonstrate an understanding of the main risks of software development and use.
Skills	
Cognitive Skills (SC)	SC1: Model, design, implement and evaluate object-oriented software systems. SC2: Identify, design and plan software solutions to problems using an object-oriented strategy, and critically evaluate and justify the proposed design solutions. SC3: Demonstrate creativity and innovativeness in developing computing solutions to real world problems. SC4: Investigate and improve the specification of a software system. SC5: Use logic and discrete mathematics to specify software elements.

Categories		Expected Learning Outcome
Practical Skills (SP)	SP1:	Design appropriate solutions in one or more application domains using software engineering approaches that integrate ethical, social, legal, and economic concerns.
	SP2:	Analyse system requirements and the production of system specifications, ensuring high-quality functional and nonfunctional requirements and a feasible software design.
	SP3:	Specify, design, construct and use CASE tools and application software.
	SP4:	develop and apply testing strategies for software applications.
	SP5:	Plan, design, deploy and document appropriate security for computer systems.
	SP6:	Design, implement, and evaluate a computer-based system, process, component, or programme to meet desired needs.
	SP7:	Prepare and deliver coherent and structured technical reports.
Interpersonal Skills (SI)	SC1:	Work both individually and as part of a team to develop and deliver quality software artifacts.
	SC2:	Demonstrate an understanding and appreciation of the importance of negotiation, effective work habits, leadership, and good communication with stakeholders in a typical software development environment.
	SC3:	an ability to communicate effectively with a range of audiences.
	SC4:	Reconcile conflicting project objectives, finding acceptable compromises within the limitations of cost, time, knowledge, existing systems, and organizations. Students should be engaged in exercises that expose them to conflicting and changing requirements.
	SC5:	Demonstrate an understanding of professional, ethical, legal, security, social issues and responsibilities in computing.

Categories	Expected Learning Outcome
Attitude (A)	A1: Creative and innovative in developing computing solutions to real world problems. A2: Able to act professionally in a work environment. A3: Able to work in a multicultural and global computing environment. A4: Able to demonstrate commitment to lifelong learning self and professional development. A5: Able to demonstrate self-awareness and ability to adapt to new situations. A6: Able to manage learning and self-development, including time management and the development of organizational skills.

7.4 Translating Learning Outcomes in the Basic Phase of Software Engineering Programme

Learning outcomes need to be translated into the programme which is in this case considered as a coherent set of courses leading to a bachelor's degree in Software Engineering. This bachelor's degree programme is commonly divided into foundation and specialization phases. The former is common for all Software Engineering programme students and consists of core and supporting courses while the latter allows students to choose certain specializations according to their interests.

In the foundation phase, three (03) types of areas can be distinguished as follows:

- **Core Courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline and they are mandatory for every student. The core courses are the backbone of the discipline and they are the typical Software Engineering courses.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses, it will be difficult to understand the core courses. For example, Mathematics is compulsory for all students.

- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student, to deepen or to broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 34 summarizes the core knowledge and supporting knowledge areas in the basic phase of a Bachelor of Science in SE programme. The following should be noted:

i) The titles of the courses may differ from one university to another. The emphasis should be on the content rather than title;

ii) The autonomy and the uniqueness of the university will be taken into consideration in formulating the core courses for the basic phase. Universities will have the choice to add their own courses beyond the core and supporting courses; and

iii) The core and supporting courses may be designed in the form of courses.

Table 34: Core and supporting knowledge areas for Software Engineering programme

Core Knowledge Areas	Supporting Knowledge Areas
• Engineering Foundation • Systems Analysis and Design • Data Management • Computer Programming • Computer Networks and Communication • Platforms Technologies • Software Development • Systems and Project Engineering • Preparation for Professional Practice • Software Management • Software Maintenance • Software Testing and Quality Assurance • Emerging Technologies • Systems Security	• Mathematics • General Studies

7.5 Credit Framework for Software Engineering Programme

The credit framework for the Bachelor of Science in SE programme will be as presented in Table 35. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should, normally, be designed as presented in Table 35.

Table 35: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminars and Tutorials	% on Assignments	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 480 as minimum cumulative credits required for a learner to graduate with a four-year UQF level 8 programme such as the Bachelor of Science in SE. It should be noted that, for a knowledge area, SE programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn databases, more weight should be assigned to the practical activity.

7.6 Normal Learning Matrix for Software Engineering Programme

The sample learning matrix in Table 36 is a result of environmental scanning of SE programme curricula from various institutions. The proposed minimum benchmarks consist of 15 core knowledge areas that form the foundation for a Bachelor in SE.

Table 36: Normal Learning Matrix with minimum standards core credits

Core Knowledge Areas	
Software Engineering Foundation [28] • Measurement and metrics • Software Engineering fundamentals • Computer Organization and Architecture • Engineering Design	**System Analysis and Design [36]** • Requirement Engineering • Object Oriented Analysis and Design • Structured Systems Analysis and Design
Data Management [21] • Databases • Data Mining & Knowledge Discovery in Databases	**Computer Programming [54]** • Procedural Programming • Object-Oriented Programming • Logical Programming • Internet and Application Programming • Compiler Technology • Data structures and Algorithms • Assembly Programming
Computer Networks and Communication [16] • Data Communication and Networks • Computer Network Design and Administration	**Platforms Technologies [16]** • Open-source Technologies • Operating Systems • Computer Architecture and Organization • Mobile Technologies
Software Development [36] • Software Development Tools and Technologies • Business Intelligent Systems • Network Application Development • Real time and Embedded Applications • Mobile Computing and Applications	**Preparation for Professional Practice [36]** • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and Organizational Behaviour • Final Year Project • Engineering Project Management • Intellectual property and legal

	issues • Workshop Training • General Engineering Procedure and Ethics
Software Management [28] • Software project management • Problem Analysis and Reporting • Risk Management • Software Configuration Management	**Software Maintenance [27]** • Maintenance issues • Software Evolution • Distribution and backup
Software Testing and Quality Assurance [24] • Software Quality Assurance principles • Software Metrics • Process and Product quality planning • Software verification and validation • Software reliability and testing	**Emerging Technologies [16]** • Internet of Things • Artificial Intelligence • Data Science • Block Chain • Virtual Reality
Systems Security [28] • Network Security • Information Security • Software Security • Digital Forensic • Organization Security • Human/personnel and Societal Security	
Supporting Knowledge Areas	
General Studies [21] • Life Skills • Communication Skills • Development Perspective	**Mathematics [36]** • Discrete Mathematics • Mathematics Logic and Formal Semantics • Linear Algebra • Calculus Numerical Analysis • Probability and Statistics

A university should design a SE programme such that a student covers a minimum of 480 credit hours in four years for core, supporting and elective knowledge areas. For example, the minimum standard to deliver lectures in Requirement Engineering of the knowledge area of Systems Analysis and Design is proposed to be a minimum of two (02) hours per week. It should be noted that an equal learning weight of two (02) hours has been placed on delivering practical. In a semester of 15 weeks, a course in Requirement Engineering would take 30 hours for lectures and 30 hours for practical. As shown in Table 37, the minimum credits to deliver knowledge in the System Analysis and Design is 36.

Table 37: Sample minimum credits for Software Engineering Foundation as the knowledge area

Knowledge Units	**# of Courses**	**L/H**	**S H**	**A/H**	**I/H**	**P/H**	**T/H**	**Credits**
% of Learning Activity		30	20	10	10	30	100	
Requirement Engineering	1	30	15	7.5	7.5	30	90	9
Object Oriented Analysis and Design	2	60	30	15	15	60	180	18
Structured Systems Analysis and Design	1	30	15	7.5	7.5	30	90	9
Total minimum credit for Software Engineering Foundation							**360**	**36**

7.7 Course Description for Software Engineering Programme

All courses to be taught and learnt within the programme should follow the outline in Table 38.

Table 38: Course outline for Software Engineering programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course aims are statements that describe the overarching intentions of a course. They should try to answer, “What is the purpose of the course?”, “What is the course trying to achieve?”
Course Expected Learning Outcome (s)	Course Learning Outcome(s) should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency.
Course Status	Core or Elective

Credit Rating	Reference should be made to the UQF credit framework.
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books)

7.8 Learning Outcomes and Curriculum Alignment Matrix for Software Engineering Programme

It is important to develop a curriculum alignment matrix to check if the planned courses cover the learning outcomes. For each course, one has to formulate the specific learning outcomes for that course and have to check how far the course contributes to the programme learning outcomes. Table 39 shows an example of a curriculum alignment matrix for the ELOs of Software Engineering programme. For each knowledge area, the contribution to the ELOs is given.

Table 39: Curriculum Alignment Matrix for Software Engineering programme

Core Knowledge Areas	K1	K2	K3	K4	K5	K6	SC1	SC2	SC3	SC4	SC5	SP1	SP2	SP3	SP4	SP5	SP6	SP7	SI1	SI2	SI3	SI4	SI5	A1	A2	A3	A4	A5	A6
Software Engineering Foundation	x		x	x								x																	
System Analysis and Design	x	x	x				x			x		x	x	x		x	x	x				x							
Data Management	x		x				x																						
Computer Programming			x				x							x	x		x												
Computer Networks and Data Communication	x		x	x												x													
Platforms Technologies	x	x	x	x			x	x									x	x											
Software Development	x	x	x				x					x	x	x			x					x							x
Preparation for Professional Practice								x	x			x						x		x	x	x		x	x	x	x	x	x
Software Management	x	x	x				x					x	x	x			x					x							x
Software Maintenance	x	x	x				x					x	x	x			x					x							x
Software Testing and Quality Assurance	x	x	x				x					x	x	x			x					x							x

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**K5**	**K6**	**SC1**	**SC2**	**SC3**	**SC4**	**SC5**	**SP1**	**SP2**	**SP3**	**SP4**	**SP5**	**SP6**	**SP7**	**SI1**	**SI2**	**SI3**	**SI4**	**SI5**	**A1**	**A2**	**A3**	**A4**	**A5**	**A6**
Emerging Technologies	x	x	x	x	x					x							x												
Systems Security												x				x													
Supporting Knowledge Areas																													
Mathematics				x							x																x		x
General Studies	X																							X	X	X	X	X	X

7.9 Benchmarks for Specialization of Software Engineering Programme

After the basic phase, students may choose to deepen their knowledge in one of the following specializations:

i) Modelling and Simulation;

ii) Computer Applications Development;

iii) Systems Development;

iv) Web Development;

v) Embedded Systems Development;

vi) Mobile Applications Development;

vii) Network Centric Systems;

viii) Digital and embedded systems;

ix) Geographic Information Systems; and

x) Games and Entertainment Systems.

7.10 Minimum Entry Qualification for Software Engineering Programme

Entry requirements for admission into the Bachelor of Science in Software Engineering programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have a background in Mathematics and Physics and for equivalent qualification, a first-class Diploma in SE or CE or CS or IT from a recognised institution.

7.11 Software Engineering Programme Duration

The duration for Bachelor of Science in Software Engineering should be four (04) years, with a minimum of two (02) semesters per academic year.

7.12 Graduation Requirement for Software Engineering Programme

Students are required to accumulate a minimum of 480 credits which account to 4,800 notional hours of learning to graduate.

7.13 Evaluation and Course Assessment for Software Engineering Programme

7.13.1 Evaluation

Universities should offer new software engineering courses to the learners. Due to the rapid changes in ICT and emerging technologies, the design and implementation of curricula should be regularly reviewed and evaluated. The data needed for assessment can be collected in a number of ways: written student evaluations, in-class observations, and personal interviews with students and faculty members from the client departments. Once the course has been taught, it is a good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

7.13.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

7.14 Resource Requirements for Software Engineering Programme

Human resources for the SE degree programmes keep changing with time. In addition to human resources, the resources needed for the SE degree programme are lecture rooms, internet access, laboratories and library resources. In a rapidly changing technical environment, students should be exposed to a variety of up-to-date hardware and software systems that adequately represent the professional setting in which they will be employed.

7.14.1 Human Resource Requirements

Staff of the offering department are vital to the strength and delivery of the Software Engineering programme. Its faculty needs both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete a degree in a timely manner. The interests and qualifications of the staff must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

The staff must remain up-to-date in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual staff. Given the rapidly changing technology, it is particularly critical that staff have sufficient time for professional development and scholarly activities. Resources should be provided for faculty to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organizations. The programme is enhanced significantly when faculty acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Faculty must also be equipped to develop teaching materials for their students. Staff must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. The staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed for the programme is influenced by such factors as the number of students in the program, the number of required courses, the number of service and elective courses offered, and the teaching load of the faculty. A programme should have a minimum number of full-time staff members with primary commitment to the Software Engineering programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the staff members should span a range of interests in ICT including computer systems concepts, information systems concepts, data management, telecommunications and networks, systems design and development, systems integration, and Information Systems management and policy. Additional staff will be needed to teach the service courses that provide foundation-level knowledge across the campus.

7.14.2 Computing Infrastructure Requirements

Computing infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the SE programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most freshmen enter college with computer resources, so access plays a much more significant role (Lee, 2009). Therefore, network access should be available for faculty and students to use with their own computers. Students at different levels in the curriculum have different needs. Substantial resources must be provided to support the courses targeted to all students. Specialized laboratories or access to specialized simulation software is needed for advanced students where group and individual projects are developed. Contemporary and emerging software development tools should be available to create the most current enterprise solutions.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital necessity in a campus environment.

7.14.3 Laboratory Requirements

Systems require hardware and software for structured, open/public, and specialized laboratories. Students must have an opportunity to use learning materials in both structured and unstructured laboratories.

Hardware and software are rapidly changing and improving. It is critical that faculty and students have access to facilities reflecting an environment that graduates will be expected to use professionally. All computing systems should be kept current. A plan should exist to continuously upgrade and/or replace software and equipment in a timely manner. The rate of change in technology suggests a rapid replacement cycle, with some technologies reaching obsolescence in less than 12 months.

Simulation software is becoming more prevalent for teaching advanced SE topics. This can include simulations for using applications to manage single workstations to complex enterprise-level networks. Many companies including Microsoft, Cisco, and even the textbook companies have developed sophisticated simulation software that does not require the latest equipment.

Various courses and areas of study have their own specialized requirements such as the large database with realistic sample data that are needed for effective work in the area of data management. Students should be provided opportunities to work together on team-oriented projects. The group skills developed in this mode are critical to a successful Information Systems professional. Technological support, such as groupware, is expected for group and team activities. All laboratories must have adequate technical support in terms of professional staff to provide for installation and maintenance of the equipment. The staff should be proficient in both the hardware and software applications. Complete documentation must also be available. Laboratories should be able to support the following types of functions:

i) Structured Laboratories

A structured laboratory is a closed, scheduled, supervised experience in which students complete specified exercises. An instructor who is qualified to provide necessary support and feedback to the students provides supervision. Exercises are designed to reinforce and complement the lecture material.

ii) Open/Public Laboratories

Student ownership of computers has continued to increase. However, laboratories remain essential for those students who do not own a computer and for providing additional resources not available on personal machines.

iii) Specialized Laboratories/Studies

Laboratory facilities are necessary to support team projects and special computing environments. Special facilities may be needed for systems development, network infrastructure, and other advanced technologies.

7.14.4 Classrooms

Suitable classroom facilities, equipped with information technology teaching resources, should be provided. A computing system with multimedia facilities is necessary for demonstrating the development, implementation, and application of information technology as well as conducting walkthroughs and making presentations. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

7.14.5 Library

Library support is an important part of an academic programme. It is especially important for disciplines with rapid development of knowledge such as the Information Systems field. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the computing professional societies.

The objective of the project work is to give students an opportunity to apply the knowledge matter learnt to a practical problem under supervision of an academic staff. The project work should be compulsory and graded. The project can be done at industry or at the university. Interdisciplinary projects (involving Computer science and other disciplines) and teamwork are encouraged. At the end of the programme, students should be required to submit a final project report, make an oral presentation and demonstrate a working prototype.

CHAPTER 8

BENCHMARKS FOR A BACHELOR OF SCIENCE IN TELECOMMUNICATIONS ENGINEERING PROGRAMME

8.1 Telecommunications Engineering Programme

A Telecommunications Engineer is responsible for designing and overseeing the installation of telecommunications equipment and facilities, such as complex electronic switching systems, and other plain old telephone service facilities, optical fiber cabling, IP networks, and microwave transmission systems.

8.1.1 Telecommunications Engineering in the ICT Cluster

Telecommunications Engineering (TE) evolved from the disciplines of electrical engineering. Initial curricula efforts in the TE commonly occurred as a specialization within electrical engineering programmes. It began using smoke signals and drums in different parts of the world. In the 1790s, the first fixed semaphore system emerged in Europe. In the 1830s, electrical telecommunications systems started to appear, extending digital logic design to the creation of small-scale digital systems and, eventually, to the design of microprocessors and computer systems. Today, TE programmes reflect different areas such as communication, signal processing, electronics, electrical, networking, computer knowledge, programming, embedded systems, and other emerging technologies.

The BSc. TE degree is theoretical based, which means that most, if not all, of the courses needed to complete the programme involve the use of research, and the proper understanding and analysis of theories and principles that are presented in textbooks, journal reports, and past experiments. This includes engineering courses. While there are certain laboratory courses that the student needs to complete, these are often done in controlled environments, in order to prove and give the students a further understanding of the theories and principles discussed in class.

8.1.2 Telecommunications Engineering Programme Objectives

The overall objective of the programme is to train quality telecommunications engineering graduates of high intellectual standard and high caliber capable of managing and operating telecommunications systems and to meet current and future needs in the field of telecommunications engineering, both in Tanzania and internationally. In order to ensure distinction, graduate versatility, and competence, every TE programme design should aim to produce graduates who meet the following specific objectives.

8.1.3 Academic Ability

Every TE programme should aim to equip learners with academic knowledge/skills/ability to:

i) Model, design, construct, operate, evaluate and maintain computers, computer-engineered systems, their hardware, software, network components as well as their integration to solve engineering problems; and

ii) Analyse a problem and then: a) identify and define the computing and engineering requirements appropriate to the problem solution; b) evaluate the extent to which the telecommunications solution meets the criteria defined for its current use and scale up to future development.

8.1.4 Academia and Industrial Linkage / Employability

The programme objectives under this category are to equip learners with knowledge and skills that match with the industrial needs as follows:

i) Ability to integrate theory and practice to work effectively and efficiently in organizations;

ii) Knowledge and skills that enable creativity, innovativeness, and entrepreneurship in the field of computing;

iii) Ability to assess risk related and analyse the impact of local and global trends of Computer Engineered solutions on individuals, organizations, and society; and

iv) Demonstrate critical thinking, creativity, and innovativeness in developing computer-engineered solutions to real world problems in societies and industries.

8.1.5 Personal Development

The programme objectives under this category are to:

i) Empower learners to progress in their personal career, impart professional ethics to the learner, equip the learner with skills and attitude to work in multicultural and global environments;

ii) Enable the learner to develop skills to perform effectively in technical and nontechnical environments, as a team member and at an individual level; and

iii) Enable learners to communicate effectively with experts and non-experts.

8.2 Expected Learning Outcomes for Telecommunications Engineering Programme

Expected Learning Outcomes (ELOs) provide a mechanism for describing not just knowledge and relevant practical skills, but also personal and transferable skills. They describe what a student is expected to know or can do by the time of graduation. To harmonize the Bachelor programme in TE, the following ELOs have been formulated to be used as benchmarks. The formulated ELOs as stipulated in Table 40 are the threshold: all graduates of the bachelor's programme in TE must achieve them. Besides these, a graduate must also achieve the ELOs for their chosen specialization. Universities may consider adding ELOs as and when necessary, in line with their mission and vision or other identified need(s).

Table 40: Expected Learning Outcomes for Telecommunications Engineering programme

Category	Expected Learning Outcomes
	A graduate should be able to:
Knowledge (K)	K1: Demonstrate knowledge and understanding of concepts, theories and operations of technical skills to work professionally in one or more of the following areas: Telecommunications hardware and software design, computer network design, Telecommunications system design and integration in wire line, mobile and satellite systems. K2: Demonstrate knowledge and understanding of

Category	Expected Learning Outcomes
	fundamental mathematics and engineering science consistent with problem solving abilities of a degreed professional in the TE field. K3: Demonstrate understanding of quality standards and benchmarks in design and development of telecommunications systems and networks that involve both hardware and software and capable of entering and succeeding in an advanced degree programme in a field such as engineering, science, or business.
Skills	
Cognitive Skills (SC)	SC1: Analyse the impact of local and global trends of TE solutions on individuals, organizations, and society. SC2: Demonstrate critical thinking, creativity, and innovativeness in developing Telecommunications Engineered solutions to real world problems. SC3: Demonstrate hands-on experience on key Telecommunications tests and measurement equipment. SC4: Analyse a problem and then identify and define the computing and engineering requirements appropriate to the problem solution. SC5: Communicate clearly, demonstrate effective leadership traits and decision-making skills, commitment to ethical and social responsibilities, diversity, lifelong learning, and teamwork abilities.
Practical Skills (SP)	SP1: Model, design, implement and evaluate telecommunications systems as well as their integration to of different Telecommunications systems. SP2: Deploy a variety of computer-based and laboratory tools and properly interpret and report experimental results. SP3: Evaluate, verify, trouble-shoot, test and analyse existing telecommunications systems. SP4: Identify and address telecommunications problems.
Interpersonal Skills (SI)	SC1: Work effectively in a team. SC2: Communicate effectively with experts and non-

Category	Expected Learning Outcomes	
		experts.
	SC3:	Demonstrate an understanding of professional, ethical, legal, security, social issues and responsibilities in TE.
Attitude (A)	A1:	Adapt to, and work in a multicultural and global TE environment.
	A2:	Work effectively as a team member as well as at an individual level.
	A3:	Demonstrate commitment to lifelong learning, self and professional development.
	A4:	Demonstrate self-confidence and ability to adapt to new situations.

8.3 Translating Learning Outcomes in the Basic Phase of Telecommunications Engineering Programme

Learning outcomes need to be translated into the programme which is in this case considered as a coherent set of courses leading to a bachelor's degree in Telecommunications Engineering. This bachelor's degree programme is commonly divided into foundation and specialization phases as follows:

i) **The Basic/Foundation Phase:** This phase is common for all ICT programme students and consists of core and supporting knowledge areas. An engineering programme that falls within the ICT cluster must provide a base in the science area, engineering area, and general study area. Universities should consider these areas to fit their objectives and missions.

ii) **The Specialization Phase:** This allows students to choose certain specializations according to their interests. The programme may be organized in courses.

In the basic/ foundation phase, three (03) types of areas can be distinguished as follows:

- **Core courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline. The core courses are the backbone of the discipline. They are the typical TE courses mandatory for every student.

- **Supporting courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses, it will be difficult to understand the core knowledge areas.
- **Elective courses (or Knowledge Areas):** These are courses that can be taken by a student, to deepen or broaden the knowledge, but they are not compulsory. However, a student must make a choice to meet the minimum credit requirements for graduation.

Table 41 shows the core knowledge and supporting knowledge areas in the basic phase of a Bachelor of Science in TE programme. Based on these cores and supporting knowledge areas, the following remarks apply:

i) The core knowledge areas refer to the knowledge and skills that every student in all TE degree programme should attain;

ii) The titles of the courses may differ from one university to another. The emphasis should be on the content rather than title;

iii) The autonomy and the uniqueness of universities will be taken into consideration in formulating the core courses for the basic phase. Universities will have the choice to add their own courses beyond the core and supporting courses; and

iv) The knowledge areas are not courses and the core components do not constitute a complete curriculum. Each TE programme may choose to cover the core knowledge areas in a variety of ways.

The goal of designing this TE programme benchmarks is to keep the required component of the body of knowledge as small as possible. This is done to allow TE programme to be as flexible as possible since the programme goals or objectives vary widely from programme to programme. Therefore, TE curriculum designers are flexible to formulate data structures and algorithms as subject units under corresponding knowledge areas such as programming.

Table 41: Core and supporting knowledge areas for Telecommunications Engineering programme

Core Knowledge Areas	Supporting Knowledge Areas
• Electrical Foundation • Electronics Foundation • Signal Processing • Electrical and Electronic Measurement • Computer Networks and Communication • Communication Foundation • Embedded Systems • Control System • Hardware Design and Implementation • Project Engineering • Computer Knowledge • Computer Programming • Platforms Technologies • Emerging Technologies	• Mathematics • General Studies

8.4 Credit Framework for Telecommunications Engineering Programme

The credit framework for the Bachelor of Science in TE programme will be as presented in Table 42. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should, normally, be designed as presented in Table 42.

Table 42: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminar and Tutorial	% on Assignment	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

The minimum cumulative credit required for a learner to graduate with a bachelor of TE in four years is 480. It should be noted that, for a particular knowledge area, TE programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn programming, more weight should be assigned to the practical activity.

8.5 Normal Learning Matrix for Telecommunications Engineering Programme

The following sample learning matrix is a result of environmental scanning of TE programme curricula for higher education minimum standards for bachelor of TE. The matrix depicts varying credit weight to each knowledge area. Universities should design a TE programme such that a student covers a minimum of 120 credits per year and a minimum of 385 credit hours in four years for core knowledge areas as shown in Table 43.

Table 43: Normal Learning Matrix with minimum standards core credits

Core Knowledge Areas	
Electrical Foundation [27] • Fundamental of Electrical Engineering • Electrical Energy Systems • Electrical Material • Electrical Network Analysis and Design • Electrical Machine	**Communication Foundation [31]** • Introduction to Telecommunications Engineering • Mobile Communication • Microwave Engineering • Satellite Communication • Television Systems

Control Systems [18] • Control Theory • Control Systems Engineering	• Information Theory and Coding • Optical Communication • Engineering Electromagnetics • Tele-traffic Engineering • Radio Frequency Planning • Switching and Transmission
Electronics Foundation [27] • Analog Electronics • Analog Telecommunications • Digital Electronics • Digital Telecommunications • Very Large-Scale Integrated Circuits	**Signal Processing [27]** • Fundamental of Signal and System • Digital Signal Processing • Digital Image Processing • Digital Broadcasting Technology • Digital Broadcasting Technology
Electrical and Electronic Measurement [18] • Measurements and Instrumentation • Intelligent Instrumentation **Embedded Systems [21]** • Fundamentals of Embedded Systems • Microcontroller Systems • Microprocessor Systems	**Preparation for Professional Practice [36]** • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and Organizational Behaviour • Final Year Project • Engineering Project Management • Intellectual Property and Legal Issues • Workshop Training • General Engineering Procedure and Ethics
Computer Engineering Foundation [18] • Introduction to Computer Engineering • Computer Organization and Architecture	**Hardware Design and Implementation [21]** • Engineering Drawing • System Design and Implementation • Computer Aided Drafting and Design

Computer Networks and Communication [27] • Computer Networking Design and Admiration • Computer Network Protocol • Networking Router and Routing Protocol • LAN Switching • WAN Technology • Wireless Networking	**Computer Programming [18]** • Procedural Programming • Object Oriented Programming • Logical Programming
Emerging Technologies [16] • Internet of Things • Artificial Intelligence • Data Science • Blockchain • Virtual Reality	**Platforms Technologies [17]** • Open-Source Technologies • Operating Systems • Mobile Technologies • Computer Architecture and Organization
Systems Security [21] • Network Security • Information Security • Software security • Digital Forensic • Organization Security • Human/Personnel and Societal Security • System and Component Security	
Supporting Knowledge Areas	
Mathematics [21] • Linear Algebra • Probability and Statistics • Calculus • Discrete Mathematics	**General Studies [21]** • Communication skills • Perspectives in Development • Human Resource Management

The minimum standard to deliver lectures in the knowledge area of TE is proposed to be two (02) hours per week. It should be noted that equal learning weight of two (02) hours has been placed on delivering practical in the knowledge area of TE. Therefore, for a typical 15-week semester, a course in TE would take 30 hours for lectures and 30 hours for practical sessions per semester. Again, a course can have one (01) hour for tutorial and more hours for assignments and self-study. In a course of four years, universities may prepare a minimum of five (05) courses in the Communication Foundation to ensure all theories, concepts, and practical sessions are covered. As shown in Table 44, the minimum credits to deliver knowledge in the Communication Foundation (CF) area is 18.

Where:

CF1 represents Introduction to Telecommunications Engineering

CF2 represents Microwave Engineering

Table 44: Sample minimum credits for Telecommunications Engineering Foundation as the knowledge area

Knowledge Units	**# of Courses**	**Lectures Hrs.**	**Seminars and Tutorials Hrs.**	**Assignment Hrs.**	**Independent Studies and Research Hrs.**	**Practical Training Hrs.**	**Total Hours**	**Credits**
% of Learning Activity		30	20	10	10	30		
CF1	1	30	15	7.5	7.5	30	90	9
CF2	1	30	15	7.5	7.5	30	90	9
Total minimum credit for Communication Foundation							**180**	**18**

The proposed minimum standard consists of 15 core knowledge areas that form the foundation for a bachelor in TE. Table 43 outlines 385 as minimum credits for a Bachelor in TE. Minimum credits for each of the 15 knowledge areas have been formulated following the same process as for TE foundation in Table 44. A minimum of 95 credits have been left open for learners to pursue several electives in the span of four years. The 385 minimum core credits and 95 minimum elective credits make up 480 credits which is the minimum cumulative credits requirement to graduate in TE in four years.

While a TE programme learning matrix may differ from one university to another in terms of number of courses offered and weight of learning activities, more credit weight must be directed towards knowledge areas and courses that distinguish TE graduates from other graduates.

8.6 Course Description for Telecommunications Engineering Programme

All courses to be taught and learnt within the programme should follow the outline in Table 45.

Table 45: Course outline for Telecommunications Engineering programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, "What is the purpose of the course?", "What is the course trying to achieve?"
Course Expected Learning Outcome (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit framework.
Total Hours Spent	Total number of hours spent in the course
Course Content:	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods:	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

8.7 Learning Outcomes and Curriculum Alignment Matrix for Telecommunications Engineering Programme

To check if the knowledge area covers the learning outcomes, it is important to develop a curriculum alignment matrix. For each knowledge area, one has to formulate the specific learning outcomes for that course and have to check how far the course contributes to the programme learning outcomes. Table 46 gives an example of a curriculum alignment matrix for the ELOs of the Bachelor in TE. For each knowledge area, the contribution to the ELOs is given. As shown in Table 46, K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal, and Attitude, respectively.

Table 46: Curriculum Alignment Matrix for Telecommunications Engineering programme

Core Knowledge Areas	K1	K2	K3	SC1	SC2	SC3	SC4	SC5	SP1	SP2	SP3	SP4	SI1	SI2	SI3	A1	A2	A3	A4
Electrical Foundation	x				x														
Electronics Foundation	x					x				x									
Signal Processing	x				x					x									
Electrical and Electronic Measurement	x				x	x				x									
Computer Networks and Communication	x		x								x	x							
Communication Foundation	x		x	x	x						x	x							
Embedded Systems	x				x							x							
Control Systems	x									x		x							
Systems Security	x									x					x				x
Preparation for Professional Practice	x		x	x	x	x	x	x		x	x	x	x		x	x	x	x	x
Computer Programming		x	x		x					x							x		
Hardware Design and Implementation	x		x								x	x							
Platforms Technologies	x			x											x				x

Emerging technologies	x	x	x	x	x														
Supporting Knowledge Areas																			
Mathematics	x	x								x									
General Studies																x	x	x	x

8.8 Benchmarks for Specialization of Telecommunications Engineering Programme

The following are the specializations in Telecommunications Engineering:

i) Telecommunications systems;

ii) Telematics;

iii) Electronics; and

iv) Sound and Images.

8.9 Minimum Entry Qualification for Telecommunications Engineering Programme

Entry requirements for admission into the Bachelor of Science in Telecommunications Engineering programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have a background in Mathematics and Physics and for equivalent qualification, a first-class Diploma in TE from a recognised institution.

8.10 Telecommunications Engineering Programme Duration

Telecommunications Engineering is a four years programme where all courses require the equivalent of five to seven weekly contact hours for a semester of fifteen weeks. The fact that the focus of the programme is on electrical and computing in specific does not exempt it from exposing its students to general education modules from other fields of knowledge such as communication, entrepreneurship, law, organizational behaviour, etc.

8.11 Graduation Requirements for Telecommunications Engineering Programme

At the end of each academic year, a TE learner is required to complete a minimum of 120 credit score. Hence, the minimum required credits for a four-year programme should be not less than 480. A TE learner is required to accumulate a minimum of 480 credits which accounts to 4,800 notional hours of learning to graduate. Universities should ensure learners undertake one Practical Training within each academic year except their last year where they will take the final year project.

8.12 Evaluation and Course Assessment for Telecommunications Engineering Programme

8.13.1 Evaluation

Universities should offer new general TE courses to the students. After the course has been offered, its design and implementation should be carefully reviewed and evaluated. The data needed for assessment can be collected in a number of ways: written student evaluations, in-class observations, and personal interviews with students and faculty from the client departments. Once the course has been taught, it is also a good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of the graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

8.13.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

8.13 Resource Requirements for Telecommunications Engineering Programme

8.13.1 Human Resource

Competent staff members are vital to the strength of the TE programme. For each TE programme, academic staff need both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete the degree in a timely manner. The interests and qualifications of the staff members must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

Staff must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual staff members. The school/department should support continuing staff development. Given the rapidly changing technology, it is particularly critical that staff members have sufficient time for professional development and scholarly activities. Resources should be provided for staff to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organizations. The programme is enhanced significantly when staff acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. Staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed by the programme is influenced by factors such as the number of students in the programme, the number of required courses, the number of supporting and elective courses offered, and the teaching load of the faculty/department. A programme should have a minimum number of full-time staff members with primary commitment to the TE programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty/department expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the staff members should span a range of interests in TE. Additional staff will be needed to teach the supporting courses that provide foundation-level knowledge across the campus.

8.13.2 Laboratories

Laboratory experiences are an essential part of the TE curriculum, and they serve multiple functions. It is important that TE students have ample time to observe, explore, and manipulate the characteristics and behaviours of actual devices, systems, and processes. This includes designing, implementing, testing, and documenting hardware and software, designing experiments to acquire data, analysing and interpreting that data, and using that data to correct or improve the design and to verify that it meets specifications. Universities should seek options that best suit their needs based on available space, the subject unit objectives, and the resources available.

8.13.3 Telecommunications Engineering Laboratories

Many courses in TE should contain laboratory experiences. Typically, a laboratory experience lasts one to two hours and it occurs in a location configured with specialized equipment. The depth and breadth of these experiences will vary among institutions. The variation often depends on the time allocated within the curriculum, physical space, and resources. Table 47 illustrates types of laboratories that are considered compulsory/necessary for every telecommunication engineer.

Table 47: Suggested Telecommunications Engineering laboratories

• Transmission and Switching Lab • Microwave Engineering Lab • Electronics Lab • Electrical Lab • Microprocessor Lab	• Digital Electronics Lab • Measurement Instruments • Communication Lab • Network Lab

Laboratories should include some physical implementation of designs such as electronic and digital circuits, bread-boarding, FPGAs/CPLDs, microcontroller-based systems, prototyping, and implementation of firmware. Laboratories should also include application and simulation software to design computer systems including digital systems. Simulation tools present intrinsic value as part of professional TE practice. They are useful in modelling real systems, and they are often desirable and necessary to allow students to study systems that are impractical to design and implement given available time and resources.

i) Structured Laboratories

A structured laboratory is a closed, scheduled, supervised experience in which students complete specified exercises. An instructor who is qualified to provide necessary support and feedback to the students provides supervision. Exercises are designed to reinforce and complement the lecture material.

ii) Open/Public Laboratories

Student ownership of computers has continued to increase. However, laboratories remain essential for those students who do not own a computer and for providing additional resources not available on personal machines.

iii) Specialized Laboratories

Laboratory facilities are necessary to support team projects and special computing environments. Special facilities may be needed for systems development, network infrastructure, and other advanced technologies.

8.13.4 Classrooms

Suitable classroom facilities, equipped with information technology teaching resources, should be provided. A computing system with multimedia facilities is necessary for demonstrating the development, implementation, and application of information technology as well as conducting walkthroughs and making presentations. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

8.13.5 Library

Library support is an important part of an academic programme. It is especially important for disciplines with the rapid development of knowledge such as the Information Systems field. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the computing professional societies.

CHAPTER 9

BENCHMARKS FOR A BACHELOR OF SCIENCE IN ELECTRONICS ENGINEERING PROGRAMME

9.1 Electronics Engineering Programme

Electronics Engineering (EE) is separated as a branch of electrical engineering. EE is concerned with the use of electromagnetic spectrum and with such electronic devices as integrated circuits. EE is a field requiring the application of scientific and engineering knowledge and methods, combined with technical skills, in support of engineering activities. Apart from facilitating information sharing, electronic parts are used to create phones, automobiles, household appliances, automated industries and other most recent innovations.

An electronics engineer is a person who is knowledgeable in electronics theory and design and who understands state-of-the-art practices on digital and analog circuits and systems. Electronics Engineering graduates can expect to find career opportunities in the automotive industry, manufacturing industries, the world of computers, and more specific fields such as digital signal processing, optoelectronics, and mobile electronics. Electronics engineers are hired to design and create the systems and devices that society relies upon.

9.1.1 Electronics Engineering in ICT Cluster

EE evolved from the disciplines of electrical engineering. A global environmental scan reveals that most higher education institutions still offer electronics alongside electrical engineering, and the programme nomenclature is usually phrased as "Bachelor in Electrical and Electronics Engineering". The difference between electrical and electronics engineering is often blurred, but it is generally true to say that electrical engineers are concerned mainly with the large-scale production and distribution of electrical power, while electronics engineers focus on much smaller electronic circuits. In an EE programme, one intends to develop an expert understanding of the circuits used in computers and other modern technologies, and for this reason, electronics engineering is often taught alongside computer science.

Even though electronics may be taught alongside computer science, the benchmarks in this chapter are for programmes in electronics engineering as opposed to electronics science because of the following;

i) Even though electronics has evolved to include Computer Science disciplines, it is still tightly coupled with electrical engineering, hence an EE programme of four years will allow core coverage in both the electrical engineering and computer science fields;

ii) In Tanzania's operating environment, Electronics Science students face the problem of registration and recognition from existing Professional Bodies. Currently, Government agencies prefer graduates that are recognised by Professional Bodies, hence EE programmes in the ICT cluster improves employability.

9.2 Electronics Engineering Programme Objectives

The primary objective of the EE programme is to prepare graduates for professional and supervisory positions in the field of electronics engineering. EE programmes must include opportunities to develop the skills for engineering supporting high-tech electronic industry environments, both in Tanzania and internationally. Every EE programme should be designed to meet the following specific objectives.

9.2.1 Academic Ability

Every EE programme should aim to equip learners with academic knowledge/skills/ability to:

i) Model, design, construct, operate, evaluate and maintain electronic systems, their hardware, software, network components, measurement equipment as well as their integration to solve engineering problems; and

ii) Analyse a problem and then: a) identify and define the ICT and engineering requirements appropriate to the problem solution; b) evaluate the extent to which the electronic-engineered solution meets the criteria defined for its current use and scale up to future development.

9.2.2 Academia and Industrial Linkage / Employability

The programme objectives under this category are to equip learners with knowledge and skills that match with the industrial needs as follows:

i) Ability to integrate theory and practice to work effectively and efficiently in organizations;

ii) Knowledge and skills that enable creativity, innovativeness and entrepreneurship in the field of computing;

iii) Ability to assess risk related and analyse the impact of local and global trends of electronics engineered solutions on individuals, organizations, and society; and

iv) Demonstrate critical thinking, creativity and innovativeness in developing electronics-engineered solutions to real world problems in societies and industries.

9.2.3 Personal Development

The programme objectives under this category are to:

i) Empower learners to progress in their personal career, impart professional ethics to the learner, equip the learner with skills and attitude to work in multicultural and global environments;

ii) Enable the learner to develop skills to perform effectively in technical and nontechnical environments, as a team member and at individual level; and

iii) Enable the leaner to communicate effectively with experts and non-experts.

9.3 Expected Learning Outcomes for Electronics Engineering Programme

Expected Learning Outcomes (ELOs) provide a mechanism for describing not just knowledge and relevant practical skills, but also personal and transferable skills. They describe what a student is expected to know or can do by the time of graduation. To harmonize the Bachelor programme in EE, the following ELOs have been formulated to be used as benchmarks. The formulated ELOs as stipulated in Table 48 are the threshold: all graduates of the bachelor's programme in EE must achieve them. Besides these, a graduate also must achieve the ELOs for their chosen specialization. Universities may consider adding ELOs as and when necessary, in line with their mission and vision or other identified need(s).

Table 48: Expected Learning Outcomes for Electronics Engineering programme

Category	**Expected Learning Outcomes**	
Knowledge (K)	The graduate should be able to:	
	K1:	Demonstrate knowledge and understanding of concepts, theories, and operations of electronics theory and design, including state-of-the-art practices on digital and analogy circuits and systems.
	K2:	Enable students to develop appropriate electronics analytical models based on given specifications.
	K3:	Demonstrate knowledge and understanding of fundamental mathematics and engineering science consistent with problem solving abilities of a degreed professional in the EE field.
	K4:	Demonstrate understanding of quality standards and benchmarks in design and development of small-scale circuits, electronic-based systems and networks that involve both hardware and software.
Skills		
Cognitive Skills (SC)	SC1:	Analyse the impact of local and global trends of EE solutions on individuals, organizations, and society.
	SC2:	Assess risk related to electronics-engineered activities.
	SC3:	Analyse a problem and then identify and define the electronics engineering requirements appropriate to the problem solution.
	SC4:	Evaluate the extent to which an electronics-engineered system meets the criteria defined for its current use and scale up to future development.
Practical Skills (SP)	SP1:	Model, Design, implement and evaluate electronics analytical models based on given specifications.
	SP2:	Design, implement, test and use measurements equipment for electronic systems based on given specifications and properly interpret and report experimental results.

Category	Expected Learning Outcomes	
	SP3:	Complete a sequence of design experiences, encompassing electronic hardware and software elements and their integration, building on prior work, and including at least one major project. In this context, design refers to a level of ability beyond “assembling” or “configuring” systems.
	SP4:	Evaluate, verify, trouble-shoot, test and analyse existing electronics engineered systems, their hardware, software and network components.
	SP5:	Identify and address engineering problems by extending the concepts of simple building blocks to system level design.
Interpersonal Skills (SI)	SI1:	Work effectively in a team.
	SI2:	Communicate effectively with experts and non-experts.
	SI3:	Demonstrate an understanding of professional, ethical, legal, security, social issues and responsibilities in EE.
Attitude (A)	A1:	Adapt to, and work in a multicultural and global EE environment.
	A2:	Work effectively as a team member as well as at individual level.
	A3:	Demonstrate commitment to lifelong learning, self and professional development.
	A4:	Demonstrate self-confidence and ability to adapt to new situations.

9.4 Translating Learning Outcomes in the Basic Phase of Electronics Engineering Programme

Learning outcomes need to be translated into the programme which is in this case considered as a coherent set of courses leading to a bachelor’s degree in EE. Every Bachelor programme must include the basic phase. The basic phase, also known as the foundation phase, is common for all ICT programme students and consists of core and supporting knowledge areas. In the basic/ foundation phase, three (03) types of areas can be distinguished as follows:

- **Core Courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline. The core courses are the backbone of the discipline. They are the typical EE courses mandatory for every student.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses it will be difficult to understand the core knowledge areas.
- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student to deepen or broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 49 shows the core knowledge and supporting knowledge areas in the basic phase of a Bachelor of Science in EE programme. Based on these cores and supporting knowledge areas, the following remarks apply:

i) The core knowledge areas refer to the knowledge and skills that every student in all EE degree programmes should attain;

ii) The titles of the courses may differ from one university to another. The emphasis should be on the content rather than title;

iii) The autonomy and the uniqueness of a university will be taken into consideration in formulating the core courses for the basic phase. Universities will have the choice to add their own courses beyond the core and supporting courses; and

iv) The knowledge areas are not courses and the core components do not constitute a complete curriculum. Each EE programme may choose to cover the core knowledge units in a variety of ways.

The goal of designing this EE programme benchmarks is to keep the required component of the body of knowledge as small as possible. This is done to allow programmes in EE to be as flexible as possible since programme goals or objectives vary widely from programme to programme.

Table 49: Core and supporting knowledge areas for Electronics Engineering programme

Core Knowledge Areas	Supporting Knowledge Areas
• Electronics Foundation • Electrical Foundation • Electronics Measurements • Embedded Electronics • Telecommunications Foundation • Control Systems • Computer Networks and Communication • Preparation for Professional Practice • Engineering Design • Systems and Project Engineering • Emerging Technologies • Signal Processing • Computer Programming	• Mathematics • General Studies

9.5 Emerging Technologies

The identification and effects of applied emerging technologies on the EE field is useful in producing a competent engineer who can contribute to the profession in a changing world. EE programmes should allow the exploration of emerging technologies. For example, EE instructors may encourage exploration of how OLEDs which are based on a flexible substrate can be applied to make flexible electronic displays in smartphones and laptops. Other emerging technologies include but not limited to, 3D biometrics, memristors and molecular electronics. These newly emergent and modern technologies present challenges to electronics engineering students and practitioners that could involve financial and ethical trade-offs affecting professional practice in a changing world.

9.6 Credit Framework for Bachelor of Science in Electronics Engineering

The credit framework for the Bachelor of Science in EE programme will be as presented in Table 50. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should, normally, be designed as presented in Table 50.

Table 50: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminar and Tutorial	% on Assignment	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 480 as minimum cumulative credits required for a learner to graduate with a four-year UQF level 8 programme such as the Bachelor of Science in EE. It should be noted that, for a knowledge area, EE programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn electronics measurements, more weight should be assigned to the practical activity. Universities offering BSc. Electronics Engineering programmes should assign more weight on practical and industrial field activities as compared to theoretical learning.

9.7 Normal Learning Matrix for Bachelor of Science in Electronics Engineering

The following sample learning matrix (Table 51) is a result of environmental scanning of EE programme curricula for higher education minimum academic standards for EE undergraduate programmes. The matrix depicts varying credit weight to each knowledge area. Universities should design an EE programme such that a student covers a minimum of 420 credits in four years for core knowledge areas as depicted in Table 52.

The minimum standard to deliver lectures in the knowledge area of EE is proposed to be two (02) hours per week. Equal learning weight of two (02) hours has been placed on delivering practical knowledge in the knowledge area of Electronics Foundation (EF). Therefore, for a typical 15 weeks semester, a course in Digital Electronics would take 30 hours for lectures and 30 hours for practical per semester. In a span of four years, universities may prepare a maximum of two (02) courses in Digital Electronics to ensure all theories, concepts and practical sessions are covered and a firm digital Electronics foundation is built. For instance, an EF curriculum with two (02) courses in analog electronics and two (02) courses in digital electronics knowledge areas would require 360 hours for lectures. As shown in Table 51, the minimum credits to deliver knowledge in the Electronics foundation area is 36.

Where:

EF1 represents Analog Electronics
EF2 represents Digital Electronics

Table 51: Sample minimum credits for Electronics Engineering Foundation as the knowledge area

Knowledge Units	# of Courses	Lectures Hrs.	Seminars and Tutorials Hrs.	Assignment Hrs.	Independent Studies and Research Hrs.	Practical Training Hrs.	Total Hours	Credits
% of Learning Activity		30	20	10	10	30		
EF1	2	60	30	15	15	60	180	18
EF2	2	60	30	15	15	60	180	18
Total minimum credit for Electronics foundation							**360**	**36**

The proposed minimum benchmarks consist of 15 core knowledge areas that form the foundation for a bachelor in EE. Table 52 outlines 396 as minimum credits for a bachelor in EE. Minimum credits for each of the 15 knowledge areas have been formulated following the same process as for Electronics foundation in Table 51. At least 84 credits have been left open for learners to pursue several electives in the span of four years. The 396 minimum core credits and 84 minimum elective credits make up 480 credits which is the minimum cumulative credits requirement to graduate in EE in four years.

While an EE programme learning matrix may differ from one university to another university in terms of number of courses offered and weight of learning activities, more credit weight must be directed towards knowledge units and courses that distinguish EE graduates from other graduates.

Table 52: Normal Learning Matrix with minimum standard core credits

Core Knowledge Areas	
Electronics Foundation [36] • Analogue Electronics • Digital Electronics • Analogue Electronics Practical • Digital Electronics Practical • Solid State Electronics • Optoelectronics • Very Large-Scale Integrated Circuits	**Electrical Foundation [32]** • Fundamentals of electrical Engineering • Electrical Network Analysis and Design • Electrical Materials • Electrical Energy Systems • Electrical Machines
Electrical and Electronics Measurements [27] • Measurements and Instrumentation • Intelligent Instrumentation	**Communication Foundation [27]** • Introduction to Telecommunications • Microwave electronics • Engineering Electromagnetics
Embedded Systems [36] • Microcontrollers • Microprocessors • Introduction to Robotics • Electromechanical systems • Hardware Interfacing Techniques	**Preparation for Professional Practice [36]** • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and Organizational Behaviour • Final Year Project • Engineering Project Management • Intellectual property and legal issues • Workshop Training • General Engineering Procedure and Ethics
Computer Engineering Foundation [8]	**Control Systems [18]** • Control Theory

• Computer Hardware and Maintenance	Control Systems Engineering
Computer Networks and Communication [18] • Introduction to Computer Networks • Computer Network Design and Administration	**Hardware Design and Implementation [27]** • Engineering Drawing • Computer Aided Drafting and Design • Industrial Electronics Design
Computer Programming [18] • Object Oriented Programming e.g., Java, Python, C++ • Procedural Programming e.g., C	**Emerging Technologies [16]** • Internet of Things • Artificial Intelligence • Data Science • Blockchain • Virtual Reality
Signal Processing [18] • Fundamentals of Signals and Systems • Signal Image Processing • Digital Image Processing	**Systems Security [21]** • Network Security • Information Security • Software security • Digital Forensic • Organization Security • Human/Personnel and Societal Security • System and Component Security
Platforms Technologies [16] • Open-Source Technologies • Operating Systems • Computer Organization and Architecture • Mobile Technology	
Supporting Knowledge Areas	
Mathematics [21] • Calculus • Statistics and Probability • Linear Algebra • Discrete Mathematics	**General Studies [21]** • Communication Skills for Engineers • Perspectives in Development

9.8 Course Description for Bachelor in Electronics Engineering

All courses to be taught and learnt within the programme should follow the outline in Table 53.

Table 53: Course outline for Electronics Engineering programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, “What is the purpose of the course?”, “What is the course trying to achieve?”
Course Expected Learning Outcome (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit framework
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

9.9 Learning Outcomes and Curriculum Alignment Matrix for Electronics Engineering Programme

To check if the knowledge area covers the learning outcomes, it is important to develop a curriculum alignment matrix. For each knowledge area, one must formulate the specific learning outcomes for that course and must check how far the course contributes to the programme learning outcomes. Table 54 gives an example of a curriculum alignment matrix for the ELOs of the Bachelor of Science in EE. For each knowledge area, the contribution to the ELOs is given. As reflected in Table 54, K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal, and Attitude, respectively.

Table 54: Curriculum Alignment Matrix for Electronics Engineering programme

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**SC1**	**SC2**	**SC3**	**SC4**	**SP1**	**SP2**	**SP3**	**SP4**	**SP5**	**SI1**	**SI2**	**SI3**	**A1**	**A2**	**A3**	**A4**
Electronics Foundation	x	x							x			x	x							
Electrical Foundation	x	x							x											
Electronics and Electrical Measurements	x	x		x		x			x	x		x								
Embedded Systems	x	x							x				x							
Communications Foundation	x								x											
Control Systems	x								x											
Computer Networks and Communication	x	x							x			x	x							
Preparation for Professional				x	x	x								x	x	x	x		x	

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**SC1**	**SC2**	**SC3**	**SC4**	**SP1**	**SP2**	**SP3**	**SP4**	**SP5**	**SI1**	**SI2**	**SI3**	**A1**	**A2**	**A3**	**A4**
Practice																				
Hardware Design and Implementation		x	x	x		x			x	x	x		x					x		
Computer Programming	x											x								
Computer Engineering Foundation	x											x								
Emerging Technologies	x	x	x	x					x	x					x	x				
Signal Processing	x	x							x											
Systems Security						x						x								
Platforms Technologies	x			x						x										x
Supporting																				

Core Knowledge Areas	K1	K2	K3	K4	SC1	SC2	SC3	SC4	SP1	SP2	SP3	SP4	SP5	SI1	SI2	SI3	A1	A2	A3	A4
Knowledge Areas																				
General Studies															x		x	x		x
Mathematics			x																	

9.10 Minimum Entry Qualification for Electronics Engineering Programme

Entry requirements for admission into the Bachelor of Science in Electronics Engineering programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have background in Mathematics and Physics and for equivalent qualification, a first-class Diploma in EE from a recognised institution.

9.11 Electronics Engineering Programme Duration

An EE programme typically reflects Electrical Engineering, Telecommunications and Computer Science knowledge areas in varying weight coverage depending on the vision of university. Due to the intensity of knowledge area coverage, four years of EE programme duration is recommended, with a minimum of two (02) semesters per academic year.

9.12 Graduation Requirements for Electronics Engineering Programme

At the end of each academic year, an EE learner is required to complete a minimum of 120 credit score. Hence, the minimum required credits for a four-year programme should be not less than 480. An EE learner is required to accumulate a minimum of 480 credits which accounts to 4,800 notional hours of learning to graduate.

9.13 Evaluation and Course Assessment for Electronics Engineering Programme

9.13.1 Evaluation

Universities should offer up-to-date courses to learners in the EE programme. Due to the rapid changes in the EE field and its emerging technologies, the design and implementation of curricula should be regularly reviewed and evaluated. The assessment can be done in several ways: written student evaluations, in-class observations, industrial symposium, and personal interviews with various stakeholders such as academia, alumni, and experts from ICT industries. Once the course has been taught, it is a good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of EE graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

9.13.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

9.14 Resources Requirements for Bachelor in Electronics Engineering programme

9.14.1 Human Resource

Competent staff members are vital to the strength of the EE programme. For each EE programme, academic staff need both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete the degree in a timely manner. The interests and qualifications of the staff members must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

Staff must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual staff members. The university should support continuing staff development. Given the rapidly changing technology, it is particularly critical that staff members have sufficient time for professional development and scholarly activities. Resources should be provided for staff to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organizations. The programme is enhanced significantly when staff acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. Staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed by the programme is influenced by factors such as the number of students in the programme, the number of required courses, the number of supporting and elective courses offered, and the teaching load of the faculty/department. A programme should have a minimum number of full-time staff members with primary commitment to the Electronics Engineering programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty/department expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the staff members should span a range of interests in EE. Additional staff will be needed to teach the supporting courses that provide foundation-level knowledge across the campus.

9.14.2 Laboratories for Electronics Engineering Programme

Laboratory experiences constitute an essential part of the EE curriculum, and they serve multiple functions. It is important that EE students have many opportunities to observe, explore, and manipulate characteristics and behaviours of actual devices, systems, and processes. This includes designing, implementing, testing, and documenting electronic devices and software. Laboratories should include some physical implementation of designs such as electronic and digital circuits, bread-boarding, FPGAs/CPLDs, microcontroller-based systems, prototyping, and implementation of firmware. Laboratories should also include application and simulation software to design digital systems. Simulation tools present intrinsic value as part of professional EE practice. They are useful in modelling real systems and they are often desirable and necessary to allow students to study systems that are impractical to design and implement given available time and resources.

Sample laboratory types include Circuits and Electronics, Embedded Systems, Digital Logic and System Design, and Systems and Project Engineering, Electrical Energy Systems, Mechatronics, Robotics and Specialized Electronics Labs. Telecommunication, Microwave measurements and teaching enhancement can be considered as additional EE laboratories. Universities should seek out options that best suit their needs based on available space, the knowledge unit objectives, and the resources available.

9.14.3 Electronics Engineering Laboratory Equipment

Many courses in EE should contain laboratory experiences. Typically, a laboratory experience lasts one to two hours and it occurs in a location configured with specialized equipment. The depth and breadth of these experiences will vary among institutions. The variation often depends on the time allocated within the curriculum, physical space, and resources. Table 55 illustrates the benchmarks recommended on the list of equipment for an EE laboratory.

Table 55: Sample list of Electronics Engineering laboratory equipment

SN	Equipment type	Quantity
1	Spectrum analysers	7
2	Oscilloscopes	25
3	Signal generators	18
4	Meters	20
4	Power supplies	15
5	Tester	12
6	Counter	2
7	Training boards	7
8	Electronics training boards	19
9	Telecommunications training boards	15
10	Microwave kits	4
11	Microprocessor kits	20

9.14.4 Software Considerations

Software tools and packages related to EE will vary based on the philosophy and needs of each programme. Table 56 suggests some software that could appear on all machines within specific laboratory settings. Products mentioned in the table are included for illustrative purposes only and that no endorsement of a specific product is implied. Additionally, it is not envisioned that any programme will incorporate every one of these software applications. Each programme should determine its own needs and consider including the most current version of appropriate applications.

Table 56: Suggestions for possible software applications

• Design Modelling and Simulation • Circuit-level (e.g., SPICE) • Gate-level (schematic entry) • Digital systems (e.g., VHDL, Verilog) • Analog/mixed-signal circuits • (e.g., VHDL-AMS, Verilog-AMS) • System-level design (e.g., System • Verilog, System C	• Software Development • Integrated Circuits/Asic Design • Systems Engineering Tools • Digital Hardware Prototyping • Microcontroller System Design • Mathematics Packages • Lab Automation and Instrumentation • Computer Aided Design and Modelling (Cad Tools) • Printed Circuit Board (PCB) Design

9.14.5 Open-ended Labs

The EE curriculum often contains open-ended experiences where true research and development takes place. One might view this as the “ultimate lab experience.” A culminating or capstone design experience usually embodies this open-ended flavour. In such situations, an instructor and a team of students decide on an exploration area and, once decided, the student team begins the research and design process. Programmes usually provide a dedicated space where teams can meet and work. These spaces generally contain modern facilities and provide sufficient space for electronic devices (e.g., robots) and other equipment required by the project at hand.

9.14.6 Embedded Labs

The traditional laboratory experience normally takes place in a room separate from the lecture and at a different time. For example, a graphics course might have lectures on Tuesday and Thursday, with a three-hour laboratory on Wednesday. It is increasingly common in modern educational settings to have laboratories embedded within ordinary courses. One such practice is to have lectures take place within the laboratory itself. Another is to partition a room in some manner and have the lecture take place in one part of the room with the laboratory and its associated equipment in another part of the same room. There is also an emerging trend to have "flipped classrooms" in which lectures are recorded for students to view in advance. Then class time is used to engage the students in active learning, which could include students bringing their own breadboards and instrumentation and engaging in laboratory exercise.

9.14.7 ICT Infrastructure Requirements

ICT infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the EE programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most incoming undergraduates enter college with computer resources, so access plays a much more significant role. Therefore, network access should be available for faculty and students to use with their own computers. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern ICT infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital necessity in a campus environment.

9.14.8 Library

Library support is an important part of an academic programme. It is especially important for disciplines with rapid development of knowledge such as EE. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the ICT professional societies.

CHAPTER 10

BENCHMARKS FOR A BACHELOR OF SCIENCE IN CYBERSECURITY PROGRAMME

10.1 Cybersecurity Programme

Computer security, also known as Cybersecurity (CBS) or IT security, is concerned with the protection of Information Systems from theft or damage to the hardware, the software, and the information on them, as well as from disruption or misdirection of the services they provide. It includes controlling physical access to the hardware, as well as protecting against harm that may come via network access, data and code injection, and due to malpractice by operators, whether intentional, accidental or due to them being tricked into deviating from secure procedures.

10.2 Cybersecurity Programme Objectives

The Bachelor of Science in CBS aims to prepare highly competent information technology specialists to deal with detection of digital vulnerabilities, secure design and development of software, secure design and implementation of computer networks ensuring that the Cyberspace is safe for business and the entire society. Every CBS programme should be designed to meet the following specific objectives.

10.2.1 Academic Ability

Every CBS programme should aim to equip learners with academic knowledge/skills/ability to:

i) Secure information systems;

ii) Detect vulnerabilities, devise approaches and techniques to remedy all digital vulnerabilities;

iii) Design, implement and maintain secure software systems, secure hardware systems and secure information networks; and

iv) Analyse digital incidents and provide diagnostics results.

10.2.2 Academia and Industrial Linkage / Employability

The programme objectives under this category are to equip learners with knowledge and skills that match with the societal needs as follows:

i) Ability to integrate theory and practice to work effectively and efficiently in organizations and in the community at large;

ii) Knowledge and skills that enable creativity, innovativeness and entrepreneurship in the field of CBS;

iii) Assess risk related and analyze the impact of local and global trends of information security solutions on individuals, organizations, and society; and

iv) Demonstrate critical thinking, creativity and innovativeness in developing secure systems to real world problems in societies and industries.

10.2.3 Personal Development

The programme objectives under this category are to:

i) Empower learners to progress in their personal career;

ii) Impart professional ethics to the learner, equip the learner with skills and attitude to work in multicultural and global environments;

iii) Enable the learner to develop skills to perform effectively in technical and nontechnical environments, as a team member and at individual level;

iv) Communicate effectively with experts and non-experts.

10.3 Expected Learning Outcomes for Cybersecurity Programme

Expected Learning Outcomes (ELOs) provide a mechanism for describing not just a knowledge and relevant practical skills but also personal and transferable skills. ELOs are not of equal size and do not have a uniform mapping to curriculum hours, topics with the same number of hours may have quite different numbers of associated learning outcomes. Each learning outcome has an associated level of mastery. Some literature defines the levels based on the Bloom's Taxonomy, which has been well explored within the computing field. The Bloom's Taxonomy specify the cognitive skill level as follows:

i) **Familiarity:** The student understands the concepts and its meaning. This level of mastery concerns a basic awareness of a concept as opposed to expecting real facility with its application. It provides an answer to the question "What do you know about this?"

ii) **Usage:** The student is able to use or apply a concept in a concrete way. Using a concept may include, for example, appropriately using a specific concept in a programme, using a particular proof technique, or performing a particular analysis. It provides an answer to the question "What do you know how to do?"

iii) **Assessment:** The student is able to consider a concept from multiple viewpoints and/or justify the selection of a particular approach to solve a problem. This level of mastery implies more than using a concept, it involves the ability to select an appropriate approach from understanding alternatives. It provides an answer to the question "Why would you do that?".

The formulated ELOs as stipulated in Table 57 are the threshold: all graduates of the bachelor's programme in CBS must achieve them.

Table 57: Expected Learning Outcomes for Cybersecurity programme

Category	Expected Learning Outcomes	
	Graduates should be able to:	
Knowledge (K)	K1:	Demonstrate knowledge and an understanding of essential concepts, principles, and theories relating to Cybersecurity.
	K2:	Demonstrate ability to apply knowledge and understanding of Mathematics and natural science as related to Cybersecurity.
	K3:	Demonstrate knowledge and understanding of the impact of Cybersecurity on society and the environment.
	K4:	Demonstrate understanding of quality standards and benchmarks in Cybersecurity.
	K5:	Demonstrate an ability to use current techniques, skills, and tools necessary for Cybersecurity practice.
	K6:	Demonstrate an ability to apply mathematical foundations, algorithmic principles, computer science

Category	Expected Learning Outcomes	
		theory in securing information, components, connections and systems.
	K7:	Demonstrate an ability to apply design and development principles in the identifying software and hardware vulnerabilities in various information systems complexity.
	K8:	Demonstrate understanding on institutional management, structure and governance.
	K9:	Demonstrate understanding in identifying systems vulnerabilities.
	K10:	Demonstrate understanding in remedying systems vulnerabilities.
Skills		
Cognitive Skills (SC)	SC1:	Model, design, implement and evaluate secure systems.
	SC2:	Develop secure software/ applications using modern platforms.
	SC3:	Analyze a problem and identify and define the security requirements appropriate to its solution.
	SC4:	Analyze the local and global impact of information security on individuals, organizations, and society.
	SC5:	Demonstrate creativity and innovativeness in developing secure solutions to real world problems.
	SC6:	Assess risk related to security activities.
	SC7:	Evaluate the extent to which a secure system meets the criteria defined for its current use and scale up to future development.
Practical Skills (SP)	SP1:	Deploy appropriate tools for identification of vulnerabilities in information systems.
	SP2:	Specify, plan, manage, conduct and report on security activities and projects.
	SP3:	Prepare technical reports and deliver technical presentations.
	SP4:	Plan, design, deploy and document appropriate security measures for any system.

Category	Expected Learning Outcomes	
	SP5:	Design, implement, and evaluate an AI system, process, component, or programme to meet desired needs.
	SP6:	Evaluate, verify, trouble-shoot, test and analyze existing security vulnerabilities at hardware and software level.
	SP7:	Perform digital forensics and provide evidence for various incidents.
Interpersonal Skills (SI)	SI1:	To function effectively in a team to accomplish a common goal.
	SI2:	Communicate effectively with a range of audiences i.e. experts and non-experts.
	SI3:	Demonstrate an understanding of professional, ethical, legal, security, social issues and responsibilities in computing.
Attitude	A1:	Adapt to, and work in a multi-cultural and global computing environment.
	A2:	Show awareness and understanding of the ethical standards of the profession.
	A3:	Act professionally in the work environment.
	A4:	Demonstrate commitment to lifelong learning self and professional development.
	A5:	Show self-awareness and ability to adapt to new situations.
	A6:	Be creative and innovative in developing security solutions to real world problem.

10.4 Translating Learning Outcomes in the Basic Phase of Cybersecurity Programme

Bachelor programmes are commonly divided into foundation and specialization. The foundation phase is concerned with learning the core and supporting courses. During specialization phase, learners are given an opportunity to choose certain specializations according to their interests. In addition, there are elective courses which allow students to improve their scope of knowledge as well as shaping their personalities.

- **Core Courses (or Knowledge Areas):** These are essential courses offering a thorough foundation of the discipline. These subjects are mandatory for every student;
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses it will be difficult to understand the core courses. For example, Mathematics and Basic Computer Studies are also compulsory to all students; and
- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student to deepen or broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 58 shows the core and the supporting knowledge areas in the basic phase of the Bachelor of Science in CBS. Based on the core and supporting courses, the following should be noted:

i) The titles of the courses may differ from one university to another. The emphasis should be on the content rather than the title;

ii) The autonomy and the uniqueness of the university will be taken into consideration in formulating the core courses for the basic phase. A university will have a choice to add its own courses beyond the core and supporting courses; and

iii) The core and supporting courses may be designed in form of courses.

Table 58: Core and supporting knowledge areas for Cybersecurity

Core Knowledge Areas	Supporting Knowledge Areas
• Data Security • Software Security • Connection Security • Component Security • Systems Security • Human Security • Organizational Security	• Mathematics and Statistics • General Studies

• Societal Security • Computer Programming • Platform Technologies • Digital Forensics • Emerging Technologies • Preparation for Professional Practice	

i) Applied Emerging Technologies

The identification and effects of applied emerging technologies on CBS security is useful in producing a competent engineer who can contribute to the profession in a changing world. CBS programmes should allow the exploration of applied emerging technologies. CBS engineers are already on high demand due to exponential expansion of digitization of human activities accompanied by the crimes that follow it.

ii) Conceptual Emerging Technologies

The CBS programme should allow exploration of new inventions to improve computer systems security, minimize vulnerabilities and ensure that business is conducted in a secure CBS environment.

10.5 Credit Framework for Cybersecurity Programme

The credit framework for the Bachelor of Science in CBS programme will be as presented in Table 59. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should, normally, be designed as presented in Table 59.

Table 59: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminars and Tutorials	% on Assignments	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 480 as minimum cumulative credits required for a learner to graduate with a four-year UQF level 8 programme such as the Bachelor of Science in CBS. It should be noted that, for a knowledge area, CBS programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn data security, more weight should be assigned to the practical activity to facilitate acquisition of practical skills in database security, authentication, access control, cryptography, etc. Universities offering a CBS programme should assign more weight on practical and industrial field activities as compared to theoretical learning.

10.6 Normal Learning Matrix for Cybersecurity Programme

The following sample learning matrix (Table 60) is a result of environmental scanning of CBS programme curricula for higher education minimum standards for bachelor of CBS. The matrix depicts varying credit weight to each knowledge area. Universities should design a CBS programme such that a student covers a minimum of 420 credit hours in four years for core knowledge areas as shown in Table 61.

The minimum standard to deliver lectures in the knowledge area of CBS is proposed to be two (02) hours per week. It should be noted that equal learning weight of two (02) hours has been placed on delivering practical in the knowledge area of CBS. Therefore, for a typical 15 weeks semester, a course in data security (DS) knowledge area would take 30 hours for lectures and 30 hours for practical per semester. In a course of four years, universities may prepare a maximum of four (04) courses in data security to ensure all theories, concepts and practical are covered and a firm data security foundation is built. As shown in Table 60, the minimum credits to deliver knowledge in data security area is 36.

Where:

DS1 represents cryptography and cryptanalysis,

DS2 represents Data integrity and authentication

DS3 represents Information storage security

DS4 represents Secure Communication Protocols

Table 60: Sample minimum credits for Cybersecurity Foundation as the knowledge area

Knowledge Units	**# of Courses**	**Lectures Hrs.**	**Seminars and Tutorials Hrs.**	**Assign. Hrs.**	**Indepen. Studies and Research Hrs.**	**Practical Training Hrs.**	**Total Hours**	**Credits**
% of Learning Activity		30	20	10	10	30		
DS1	1	30	15	7.5	7.5	30	90	9
DS2	1	30	15	7.5	7.5	30	90	9
DS3	2	30	15	7.5	7.5	30	90	9
DS4	2	30	15	7.5	7.5	30	90	9
Total minimum credit for Data Security Knowledge Area							**360**	**36**

The proposed minimum standard consists of 13 core knowledge areas that form the foundation for a bachelor in CBS. Table 61 outlines 420 as minimum credits for a bachelor in Cybersecurity. Minimum credits for each of the 13 knowledge areas have been formulated following the same process as for Data Security knowledge area in Table 60. Sixty (60) credits have been left open for learners to pursue several electives in the span of four years. The 420 minimum core credits and 60 minimum elective credits make up 480 credits which is the minimum cumulative credits requirement to graduate in bachelor in CBS in four years.

While a CBS programme learning matrix may differ from one university to another in terms of number of courses offered and weight of learning activities, more credit weight must be directed towards knowledge units and courses that distinguish CBS graduates from other graduates.

Table 61: Normal Learning Matrix with minimum standard core credits

Core Knowledge Areas	
Data Security [36] • Cryptography and cryptanalysis, • Data integrity and authentication • Information storage security • Secure communication protocols	**Software Security [36]** • Secure design principles, • Security requirements and their role in design • Secure software Implementation • Analysis and Testing • Secure Deployment and Maintenance • Ethics
Computer Programming [21] • Principles of secure programming • Secure Object-Oriented Programming e.g. in Java/Python/C++ • Secure procedural programming e.g. in C, C# • Secure data structures and algorithms	**Connection Security [36]** • Physical media characteristics and transmission • Systems architecture, models, and standards, • Network architecture and implementations • Network services and attacks • Network defense • End-to-end secure communications
System Security [36] • Secure systems analysis and design • Secure systems development and testing • Systems operation and management • Systems threat models and attacks • Systems audit and defenses • Security of common systems architecture • Systems Administration Security	**Component Security [36]** • Vulnerabilities of system components, • Secure component design principles, • Supply chain management security • Component security testing • Component reverse engineering • Secure network administration and management

<table>
<tr>
<td>Societal Security [21]
• Cyber law
• Cyber ethics
• Cyber policy
• Cyber Privacy</td>
<td>Organizational Security [36]
• Risk management
• Security program management
• Governance and policy
• Laws, ethics, and compliance
• Strategy and planning</td>
</tr>
<tr>
<td>Human Security [36]
• Identity management
• Social engineering
• Compliance with security policies and ethics
• Privacy and security
• Security awareness and understanding</td>
<td rowspan="2">Preparation for Professional Practice [36]
• ICT Innovation and Entrepreneurship
• Research Methods
• Practical/Industrial Training
• Social and Professional Issues in Computing
• Principles of Management and Organizational Behaviour
• Final Year Project
• Intellectual property and legal issues
• Project Management</td>
</tr>
<tr>
<td>Platform Technologies [16]
• Open-Source Technologies
• Operating Systems
• Computer Organization and Architecture
• Mobile Technologies</td>
</tr>
<tr>
<td>Emerging Technologies [16]
• Internet of Things
• Artificial Intelligence
• Blockchain
• Virtual Reality
• Security intelligence</td>
<td>Digital Forensic [16]
• Mobile Forensic
• Forensic Data Analysis
• Computer Forensic
• Network Forensic
• Database Forensic</td>
</tr>
<tr>
<td colspan="2">Supporting Knowledge Areas</td>
</tr>
<tr>
<td>Mathematics [21]
• Calculus
• Discrete Math
• Linear Algebra
• Probability and Statistics</td>
<td>General Studies [21]
• Communications Skills
• Development Perspectives
• Ethical issues
• Intellectual Property
• Change Management</td>
</tr>
</table>

10.7 Cybersecurity Course Description

All courses to be taught and learnt within the programme should follow the outline in Table 62.

Table 62: Course outline for Cybersecurity programme

Course Title	Refer to UQF qualification titles and nomenclatures.
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, "What is the purpose of the course?", "What is the course trying to achieve?"
Course Expected Learning Outcome (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit framework
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books)

10.8 Learning Outcomes and Curriculum Alignment Matrix for Cybersecurity Programme

To check if the knowledge area covers the learning outcomes, it is important to develop a curriculum alignment matrix. For each knowledge area, one must formulate the specific learning outcomes for that course and have to check how far the course contributes to the programme learning outcomes. Table 63 is an example of a curriculum alignment matrix for the ELOs of the Bachelor of Science in CBS. For each knowledge area, the contribution to the ELOs is given. As shown in Table 63, K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal and Attitude, respectively.

Table 63: Curriculum Alignment Matrix for Cybersecurity programme

Core Knowledge Areas	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	SC1	SC2	SC3	SC4	SC5	SC6	SC7	SP1	SP2	SP3	SP4	SP5	SP6	SI1	SI2	SI3	A1	A2	A3	A4	A5	A6
Data Security	x	x		x	x	x			x	x	x	x	x		x	x	x	x	x		x	x	x			x						
Software Security	x			x	x	x	x		x	x	x	x	x		x	x	x	x	x		x	x				x						
Connection Security	x	x		x	x	x			x	x	x		x		x	x	x	x	x		x	x										
Component Security	x			x	x	x	x		x	x	x		x		x	x	x	x	x		x	x										
Systems Security	x			x	x	x	x		x	x	x		x		x	x	x	x	x		x	x	x									

Human Security	x		x		x				x	x				x		x		x			x	x				x						
Organizational Security	x		x	x	x			x	x	x				x		x			x		x	x				x						
Societal Security	x		x	x										x												x						
Digital Forensics	x	x			x										x	x		x	x		x	x	x									
Emerging Technologies	x	x	x	x	x	x	x								x			x			x	x										
Computer Programming	x	x	x	x	x	x	x				x	x			x			x			x	x										

Preparation for Professional Practice					x			x	x	x	x	x	x		x		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Platforms Technologies	x			x	x	x	x								x			x			x	x										
Supporting Knowledge Areas																																
General Studies	x										x	x			x		x			x	x	x		x	x	x	x	x	x	x	x	x
Mathematics		x				x					x	x			x																	x

10.9 Minimum Entry Qualification for Cybersecurity Programme

Entry requirements for admission into the Bachelor of Science in Cybersecurity programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have background in Mathematics and Physics and for equivalent qualification, a first-class Diploma in CBS or CS or IT or EE or CE or SE or TE or DS from a recognised institution.

10.10 Cybersecurity Programme Duration

A CBS programme typically reflects Electrical Engineering and Computer Science knowledge areas in varying weight coverage depending on the vision of the CBS department. Due to the intensity of knowledge area coverage, it is recommended that CBS programmes duration be four years, with a minimum of two (02) semesters per academic year.

10.11 Graduation Requirements for Cybersecurity Programme

A student is required to accumulate a minimum of 480 credits to graduate.

10.12 Evaluation and Course Assessment for Cybersecurity Programme

10.12.1 Evaluation

Universities should offer up-to-date courses to learners in the CBS programme. Due to the rapid changes in SE field and its emerging technologies, the design and implementation of curricula should be regularly reviewed and evaluated. The assessment can be done in several ways: written student evaluations, in-class observations, industrial symposium, and personal interviews with various stakeholders such as academia, alumni, and experts from the ICT industries. Once the course has been taught, it is a good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultations should be conducted to assess the performance of CBS graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

10.12.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

10.13 Resources Requirements for Cybersecurity Programme

10.13.1 Human Resource Requirements

Competent faculty/department members are vital for the strength of a CBS programme. Staff members need both academic training and practical experience. There must be enough staff to provide course offerings that allow the students to complete a degree in a timely manner. The interests and qualifications of the faculty members must be enough not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand. Faculty members must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual faculty members. The programme is enhanced significantly when staff acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff members must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students.

The number of full-time staff needed by the programme is influenced by such factors as the number of students in the programme, the number of required courses, the number of service and elective courses offered, and the teaching load of the faculty. A programme should have a minimum number of full-time staff members with primary commitment to the CBS programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty expertise.

10.13.2 Cybersecurity Laboratories

Laboratory experiences are an essential part of the CBS curriculum, and they serve multiple functions. It is important that CBS students have many opportunities to observe, explore, and manipulate characteristics and behaviours of actual devices, systems, and processes. This includes designing, implementing, testing, and documenting hardware and software, designing experiments to acquire data, analysing and interpreting that data, and using that data to correct or improve the design and to verify that it meets specifications. Universities should seek out options that best suit their needs based on available space, the subject unit objectives, and the resources available.

Many courses in CBS should contain laboratory experiences. Typically, a laboratory experience lasts one to two hours and it occurs in a location configured with specialized equipment. The depth and breadth of these experiences will vary among institutions. The variation often depends on the time allocated within the curriculum, physical space, and resources. Table 64 illustrates the recommended types of laboratories students studying a CBS programme should experience. Some laboratories involve practices that all CBS students must or should do. The laboratory type marked "**Must**" represents experience that are considered compulsory/necessary for every Cybersecurity, while "**Should"** refers to experience that are considered the best for every Cybersecurity. The one marked "**Supplemental"** is considered as additional experience for Cybersecurity.

Table 64: Types of Cybersecurity laboratories

Laboratory type	Must	Should	Supplemental
Digital Vulnerabilities/Detection and Remedy	√		
Computer Architecture Design			√
Secure Hardware Design	√		
Secure Software System Design	√		
Secure Hardware Design	√		
Networking		√	
Cryptography		√	
Final Year Project Design	√		

10.13.3 Security Tools

Digital tools related to CBS will vary based on the philosophy and needs of each programme. Table 65 suggests some digital tools that could appear on all machines within specific laboratory settings. Products mentioned in this table are included for illustrative purposes only and that no endorsement of a specific product is implied. Additionally, it is not envisioned that any programme will incorporate every one of these tools. Each programme should determine its own needs and consider including the most current version of appropriate tools.

Table 65: Suggestions for Cybersecurity and Digital Forensic Tools

Security Tools	Digital Tools
• Scanning and penetration tools • Offensive and defensive tools • Sniffers, digital authenticators • Analytical tools, etc.	• Network tools • Threat modelling tools • Systems engineering Tools • General computing/productivity • Digital hardware prototyping • Lab automation and instrumentation • Computer aided design and modelling (CAD tools) • Other tools

10.13.4 Open-ended Labs

The CBS curriculum often contains open-ended experiences where true research and development takes place. One might view this as the "ultimate lab experience." A culminating or capstone design experience usually embodies this open-ended flavour. In such situations, an instructor and a team of students decide on an exploration area and, once decided, the student team begins the research and design process. Programmes usually provide a dedicated space where teams can meet and work. These spaces generally contain modern facilities and provide enough space for electronic devices (e.g., robots) and other equipment required by the project at hand.

10.13.5 Embedded Labs

The traditional laboratory experience normally takes place in a room separate from the lecture and at a different time. It is increasingly common in modern educational settings to have laboratories embedded within ordinary courses. One such practice is to have lectures take place within the laboratory itself. Another is to partition a room in some manner and have the lecture take place in one part of the room with the laboratory and its associated equipment in another part of the same room. There is also an emerging trend to have "flipped classrooms" in which lectures are recorded for students to view in advance. Then class time is used to engage the students in active learning, which could include students bringing their own breadboards and instrumentation and engaging in laboratory exercise.

10.13.6 ICT Infrastructure Requirements

ICT infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the CBS programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most incoming undergraduates enter college with computer resources, so access plays a much more significant role. Therefore, network access should be available for faculty and students to use with their own computers. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital necessity in a campus environment.

10.13.7 Library

Library support is an important part of an academic programme. It is especially important for disciplines with rapid development of knowledge such as CBS. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the ICT professional societies.

CHAPTER 11

BENCHMARKS FOR A BACHELOR OF SCIENCE IN DATA SCIENCE PROGRAMME

11.1 Programme Definition

Data Science (DS) is an applied field akin to engineering, with its emphasis on data and how it describes the world. A Data Science programme at the undergraduate level is designed to provide a synergistic approach to problem solving. Students learn new techniques and methods that may not exist today. They will need to work with increasingly varied forms of data. It emanates its root foundations from mathematics, computational and statistical thinking and practice while incorporating the practical and important Data Science skills.

11.2 Data Science Programme Objectives

The Bachelor of Science in DS aims at preparing highly competent information technology specialists, who can make use of data as a resource. Data scientists are professionals who employ data analytics techniques to generate knew knowledge for strategic decision making and complex problem solving in organizations and society at large. Every DS programme should be designed to meet the following specific objectives.

11.2.1 Academic Ability

The programme objectives under this category are to equip students with knowledge/skills/ ability to:

i) Analyse complex problems and synthesize solutions to those problems;

ii) Develop effective ways to solve DS computing problems;

iii) Design and implement software and software technologies based on DS;

iv) Undertake research and to progress to higher levels of studies;

v) Plan, design, deploy and document appropriate data analytics for computer systems;

vi) Design and implement solutions to meet desired needs; and

vii) Demonstrate proficiency in problem-solving techniques using DS technologies.

11.2.2 Academia and Industrial Linkage/Employability

The programme objectives under this category are to equip students with knowledge and skills that match with industrial needs as follows:

i) Problem-solving skills for DS related tasks;

ii) Up-to-date DS skills for the industry;

iii) Analytical skills to understand impacts of computing on individuals, organizations and society;

iv) Ability to integrate theory and practice to work effectively and efficiently in organizations; and

v) Knowledge and skills that enable creativity, innovativeness and entrepreneurship in the field of DS.

11.2.3 Personal Development

The programme objectives under this category are to:

i) Prepare learners for lifelong learning and research;

ii) Empower students to progress in their personal career; impart professional ethics to the learner; equip the learner with skills and attitude to work in multicultural and global environments;

iii) Equip the learner with knowledge and skills to work as a team in the DS field; and

iv) Enable the learner to develop skills to perform effectively in technical and nontechnical environments.

11.3 Expected Learning Outcomes for Data Science Programme

Expected Learning Outcomes (ELOs) provide a mechanism for describing not just a knowledge and relevant practical skills but also personal and transferable skills. ELOs are not of equal size and do not have a uniform mapping to curriculum hours, topics with the same number of hours may have quite different numbers of associated learning outcomes. Each learning outcome has an associated level of mastery. Some literature defines the levels based on Bloom's Taxonomy, which has been well explored within the computing field. The Bloom's Taxonomy specify the cognitive skill level as follows:

i) **Familiarity:** The student understands the concepts and its meaning. This level of mastery concerns a basic awareness of a concept as opposed to expecting real facility with its application. It provides an answer to the question "What do you know about this?"

ii) **Usage:** The student is able to use or apply a concept in a concrete way. Using a concept may include, for example, appropriately using a specific concept in a programme, using a particular proof technique, or performing a particular analysis. It provides an answer to the question "What do you know how to do?"

iii) **Assessment:** The student is able to consider a concept from multiple viewpoints and/or justify the selection of a particular approach to solve a problem. This level of mastery implies more than using a concept, it involves the ability to select an appropriate approach from understanding alternatives. It provides an answer to the question "Why would you do that?".

The formulated ELOs as stipulated in Table 66 are the threshold: all graduates of the bachelor's programme in DS must achieve them.

Table 66: Expected Learning Outcomes for Data Science programme

Category	**Expected Learning Outcomes**	
Knowledge (K)	Graduates should be able to:	
	K1:	Demonstrate knowledge and an understanding of essential concepts, principles, and theories relating to data and data science.
	K2:	Demonstrate ability to apply knowledge and understanding of Mathematics and natural science as related to data science.
	K3:	Demonstrate knowledge and understanding of the impact of data science on society and the environment.
	K4:	Demonstrate understanding of quality standards and benchmarks in data science.
	K5:	Demonstrate an ability to use current techniques, skills, and tools necessary for data analytics practice.
	K6:	Demonstrate an ability to apply mathematical foundations, algorithmic principles, computer science theory and in data analytics and usage.
	K7:	Demonstrate an ability to apply data science principles in spheres of life.

Category	Expected Learning Outcomes	
	K8:	Demonstrate understanding on institutional management, structure and governance.
Skills		
Cognitive Skills (SC)	SC1:	Model, design, perform and data analytics.
	SC2:	Develop/implement applications for data analytics.
	SC3:	Analyse a problem, and identify and define the requirements appropriate to its solution.
	SC4:	Analyse the local and global impact of data use and analytics on individuals, organizations, and society.
	SC5:	Demonstrate creativity and innovativeness in developing solutions requiring data science to real world problems.
	SC6:	Assess risk related to data science activities.
	SC7:	Evaluate the extent to which data science meets the criteria defined for its current use and scale up to future development.
Practical Skills (SP)	SP1:	Deploy appropriate tools for data analytics in information systems.
	SP2:	Specify, plan, manage, conduct and report on data analytics and projects.
	SP3:	Prepare technical reports and deliver technical presentations.
	SP4:	Plan, design, deploy and document appropriate data use and data analytics plan for any system.
	SP5:	Design, implement, and evaluate data exploitation meet desired needs.
	SP6:	Evaluate, verify, trouble-shoot, test and analyse an existing data use and data analytics potentials in various contexts.
Interpersonal Skills (SI)	SI1:	To function effectively in a team to accomplish a common goal.
	SI2:	Communicate effectively with a range of audiences i.e., experts and non-experts.
	SI3:	Demonstrate an understanding of professional, ethical, legal, security, social issues and responsibilities in computing.
Attitude (A)	A1:	Adapt to, and work in a multi-cultural and global

Category	Expected Learning Outcomes	
		computing environment.
	A2:	Show awareness and understanding of the ethical standards of the profession.
	A3:	Act professionally in the work environment.
	A4:	Demonstrate commitment to lifelong learning self and professional development.
	A5:	Show self-awareness and ability to adapt to new situations.
	A6:	Be creative and innovative in developing computing solutions to real world problem.

11.4 Translating Learning Outcomes in the Basic Phase of Data Science Programme

The ELOs need to be translated into the programme. A DS programme is seen as a coherent set of courses leading to a Bachelor degree in Data Science. The programme is commonly divided into foundation and specialization phases. The former is common for all Data Science programme students and consists of core and supporting courses while the latter allows students to choose certain specializations according to their interests. In the foundation phase, three (03) types of areas can be distinguished as follows:

- **Core Courses (or Knowledge Areas):** These are the essential courses offering a thorough foundation of the discipline and they are mandatory for every student. The core courses are the backbone of the discipline and they are the typical Data Science courses.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses it will be difficult to understand the core courses. For example, mathematics is compulsory for all students.
- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student, to deepen or to broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

It is important to note that the core and supporting courses can be implemented in a specific local context as independent courses or as components within fewer courses. There is a core content that every undergraduate DS programme should incorporate, and that the list of core courses captures this content.

Table 67 summarizes the core knowledge and supporting knowledge areas in the basic phase of a Bachelor of Science in DS programme. The following should be noted:

i) The titles of the courses may differ from one university to another. The emphasis should be on the content rather than the title;

ii) The autonomy and the uniqueness of the university will be taken into consideration in formulating the core courses for the basic phase. Universities will have the choice to add their own courses beyond the core and supporting courses; and

iii) The core and supporting courses may be designed in the form of courses.

Table 67: Core and supporting knowledge areas for Data Science programme

Core Knowledge Areas	Supporting Knowledge Areas
• Data Analysis and Presentation • Artificial Intelligence • Big Data Systems • Computing and Computer Fundamentals • Data Acquisition, Management, and Governance • Data Mining • Systems Security • Machine Learning • Preparation for Professional Practice • Computer Programming • Emerging Technologies • Platforms Technologies • Software Development and Maintenance • Algorithms and Complexity	• Mathematics • General Studies

11.5 Credit Framework for Data Science Programme

The credit framework for the Bachelor of Science in DS programme will be as presented in Table 68. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should normally be designed as presented in Table 68.

Table 68: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminars and Tutorials	% on Assignments	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 480 as minimum cumulative credits required for a learner to graduate with a four-year UQF level 8 programme such as the Bachelor of Science in Data Science. It should be noted that the curriculum designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn programming, more weight should be assigned to the practical activity.

11.6 Normal Learning Matrix for Data Science Programme

The learning matrix for the DS programme is a result of environmental scanning of Data Science programme curricula from various institutions. The matrix depicts varying credit weight to each knowledge area. Universities should design a DS programme such that a student covers a minimum of 480 credit hours in four years for core knowledge areas as shown in Table 69.

Table 69: Normal Learning Matrix with minimum standard core credits

Core Knowledge Areas	
Data Analysis and Presentation [21] • Foundational Considerations • Visualization • User-centered Design • Interaction Design • Interface Design and Development	**Artificial Intelligence [21]** • General Knowledge Representation and Logic based Reasoning • Knowledge Representation and Reasoning –Probability Based • Planning and Search Strategies
Big Data Systems [27] • Problems of Scale • Big Data Computing Architectures • Parallel Computing Frameworks • Distributed Data Storage • Parallel Programming • Techniques for Big Data Applications • Cloud Computing • Complexity Theory • Software Support for Big Data Applications	**Computing and Computer Fundamentals [21]** • Basic Computer Architecture • Storage Systems Fundamentals • Operating System Basics • File Systems • Networks • Web and Web Programming • Compilers and Interpreters
Data Acquisition, Management, and Governance [36] • Data Acquisition • Information Extraction • Working with Various Types of Data • Data Integration • Data Reduction and Compression • Data Transformation	**Data Mining [32]** • Proximity Measurement • Data Preparation • Information Extraction • Cluster Analysis • Classification and Regression • Pattern Mining • Outlier Detection • Time Series Data

Core Knowledge Areas	
• Data Cleaning • Data Privacy and Security	• Mining Web Data • Information Retrieval
Systems Security [21] • Data Privacy • Data Security • Data Integrity • Analysis For Security • Network Security • Software Security • Digital Forensic • Organization Security • Human/personnel and Societal Security	**Machine Learning [32]** • Machine Learning Foundation • Supervised Learning • Unsupervised Learning • Mixed Methods • Deep Learning
Preparation for Professional Practice [36] • ICT Innovation and Entrepreneurship • Research Methods • Practical/Industrial Training • Social and Professional Issues in Computing • Principles of Management and Organizational Behaviour • Final Year Project • Project Management • Intellectual Property and Legal Issues	**Computer Programming [21]** • Programming Fundamentals • Procedural Programming • Object-Oriented Programming • Logic Programming • Internet and Application Programming • Scripting Concepts • Assembly Programming
	Emerging Technologies [21] • Internet of Things • Blockchain • Virtual Reality
Platforms Technologies [16] • Open-Source Technologies • Operating Systems • Computer Organization and	**Software Development and Maintenance [16]** • Software Design and Development • Software Testing and Maintenance

Core Knowledge Areas	
Architecture • Mobile Technology	
Computer Networks and Communication [21] • Communication and Networks • Computer Networks	**Algorithms and Complexity [21]** • Data Structures and Algorithms • Algorithms Strategies • Automata, Computability and Complexity
Supporting Knowledge Areas	
Mathematics [21] • Calculus • Discrete Mathematics • Linear Algebra • Probability and Statistics	**General Studies [21]** • Communication Skills • Development Perspectives

A university should design a DS programme such that a student covers a minimum of 480 credit hours in four years for core and supporting knowledge areas. For example, the minimum standard to deliver lectures in Data Analysis and Presentation (DP) of the knowledge area of Data Science is proposed to be a minimum of two (02) hours per week. It should be noted that equal learning weight of two (02) hours has been placed on delivering practical. In a semester of 15 weeks, a course in User-centred Design would take 30 hours for lectures and 30 hours for practical. The minimum credits to deliver knowledge in the User-centred Design is 18 as shown in Table 70.

Where:

DP1 represents User-centred Design

DP2 represents Visualization

Table 70: Sample minimum credits for Data Analysis and Presentation as the knowledge area

Knowledge Units	# of Courses	Lectures Hrs.	Seminars and Tutorials Hrs.	Assign. Hrs.	Independent Studies and Research Hrs.	Practical Training Hrs.	Total Hours	Credits
% of Learning Activity		30	20	10	10	30		
DP 1	1	30	15	7.5	7.5	30	90	9
DP 2	1	30	15	7.5	7.5	30	90	9
Total minimum credit for Data Analysis and Presentation							**180**	**18**

11.7 Course Description for Data Science Programme

All courses to be taught and learnt within the programme should follow the outline in Table 71.

Table 71: Course outline for Data Science programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, "What is the purpose of the course?", "What is the course trying to achieve?"
Course Expected Learning Outcome (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit framework
Total Hours Spent	Total number of hours spent in the course
Course Content	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

11.8 Learning Outcomes and Curriculum Alignment Matrix for Data Science Programme

It is important to develop a curriculum alignment matrix to check if the planned courses cover the learning outcomes. For each course, one has to formulate the specific learning outcomes for that course and check how far this course contributes to the programme learning outcomes. Table 72 shows an example of a curriculum alignment matrix for the ELOs of Bachelor in Data Science whereby K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal and Attitude, respectively.

Table 72: Course Alignment Matrix for Data Science programme

Core Knowledge Areas	K1	K2	K3	K4	K5	K6	K7	K8	SC1	SC2	SC3	SC4	SC5	SC6	SC7	SP1	SP2	SP3	SP4	SP5	SP6	SI1	SI2	SI3	A1	A2	A3	A4	A5	A6
Data Analysis and Presentation	x	x		x	x	x			x	x	x	x	x	x	x	x	x	x	x	x	x									
Artificial Intelligence	x	x	x		x	x	x			x			x		x				x		x									x
Big Data Systems	x	x		x	x	x			x	x	x	x	x	x	x	x	x	x	x	x	x	x								x
Computing and Computer Fundamentals	x	x				x			x	x	x	x							x	x										
Data Acquisition, Management, and Governance	x		x	x	x			x	x		x	x	x	x		x	x				x	x		x	x					x
Data Mining	x	x		x	x	x			x	x	x	x	x	x	x	x	x	x	x	x	x	x								x
Systems Security														x										x						
Machine learning	x	x	x		x	x	x			x			x		x				x		x									x
Preparation for Professional Practice			x		x			x			x	x	x		x	x	x	x	x					x	x	x	x	x	x	x
Computer Programming						x				x		x			x	x	x		x	x		x		x	x		x	x		x
Emerging Technologies	x	x	x	x	x									x					x									x		
Platforms Technologies					x									x					x									x		
Software Development and Maintenance						x				x		x			x	x	x		x	x		x		x	x		x	x		x

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**K5**	**K6**	**K7**	**K8**	**SC1**	**SC2**	**SC3**	**SC4**	**SC5**	**SC6**	**SC7**	**SP1**	**SP2**	**SP3**	**SP4**	**SP5**	**SP6**	**SI1**	**SI2**	**SI3**	**A1**	**A2**	**A3**	**A4**	**A5**	**A6**
Computer Networks and Communication						x								x																x
Algorithms and Complexity	x	x	x		x	x	x			x			x		x				x		x									x
Supporting Knowledge Areas																														
Mathematics		x			x												x													
General Studies								x									x						x			x		x		

11.9 Minimum Entry Qualification for Data Science Programme

Entry requirements for admission into the Bachelor of Science in Data Science programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have sufficient background in Mathematics and for equivalent qualification, a first-class Diploma in DS or CS or IT, SE, CE or EE or TE from a recognised institution.

11.10 Data Science Programme Duration

The duration for Bachelor of Science in Data Science should be four (4) years with a minimum of number of two (2) semesters per year.

11.11 Graduation Requirement for Data Science Programme

At the end of each academic year, a student is required to complete a minimum of 120 credits. A student is required to accumulate a minimum of 480 credits to graduate.

11.12 Evaluation and Course Assessment for Data Science Programme

11.12.1 Evaluation

Universities should offer up-to-date courses to learners in the DS programme. Due to the rapid changes in DS field and its emerging technologies, the design and implementation of curricula should be regularly reviewed and evaluated. The assessment can be done in several ways: written student evaluations, in-class observations, industrial symposium, and personal interviews with various stakeholders such as academia, alumni, employers and experts from the ICT industries. Once the course has been taught, it is a good practice to interview graduates regarding the value of such course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultation should be conducted to assess the performance of DS graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

11.12.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

11.13 Resource Requirements for Data Science Programme

The human resources for the DS degree programmes keep changing with time. In addition to human resources the resources needed for DS degree programme are lecture rooms, access to the internet, laboratories and library resources. In a rapidly changing technical environment, students should be exposed to a variety of up-to-date hardware and software systems that adequately represent the professional setting in which they will be employed.

11.13.1 Human Resource Requirements

Competent staff members are vital to the strength of DS programme. Its faculty needs both academic training and practical experience (Looney et al., 2007). There must be enough staff to provide course offerings that allow the students to complete a degree in a timely manner. The interests and qualifications of the staff members must be sufficient not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand.

Staff members must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual faculty members. The school should support continuing faculty development. Given the rapidly changing technology, it is particularly critical that faculty members have sufficient time for professional development and scholarly activities. Resources should be provided for faculty to regularly attend conferences, workshops, and seminars, and to participate in academic and professional organizations. The programme is enhanced significantly when faculty acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Staff must also be equipped to develop teaching materials for their students. Staff members must be equipped with technologies at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students. Staff must be connected to the internet in order to have access to students and to the larger academic and professional community.

The number of full-time staff needed by the programme is influenced by such factors as the number of students in the programme, the number of required courses, the number of supporting and elective courses offered, and the teaching load of the faculty. A programme should have a minimum number of full-time staff members with primary commitment to the Data Science programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty expertise. Courses must be offered with sufficient frequency for students to complete the programme in a timely manner. The professional competence of the faculty members should span a range of interests in Data Science. Additional staff will be needed to teach the supporting courses that provide foundation-level knowledge across the campus.

11.13.2 Computing Infrastructure Requirements

Computing infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the DS programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most freshmen enter college with computer resources so access plays a much more significant role (Lee, 2009). Therefore, network access should be available for faculty and students to use with their own computers. Students at different levels in the curriculum have different needs. Substantial resources must be provided to support the courses targeted to all students. More sophisticated resources are necessary for Information Systems minors and majors who are developing skills in computing and DS fundamentals. Specialized laboratories or access to specialized simulation software is needed for advanced students where group and individual projects are developed. Contemporary and emerging software development tools should be available to create the most current enterprise solutions.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise.

Support staff who maintain hardware, software and communications resources rarely have overlapping skills and an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital necessity in a campus environment.

11.13.3 Laboratory Requirements

Hardware and software are rapidly changing and improving. It is critical that staff and students have access to facilities reflecting an environment that graduates will be expected to use professionally. All computing systems should be kept current. A plan should exist to continuously upgrade and/or replace software and equipment in a timely manner. The rate of change in technology suggests a rapid replacement cycle, with some technologies reaching obsolescence in less than 12 months.

In addition, simulation software is becoming more prevalent for teaching advanced ICT topics. This can include simulations for using applications to manage single workstations to complex enterprise-level networks. Many companies including Microsoft, Cisco, and even the textbook companies have developed sophisticated simulation software which do not require the latest equipment.

11.13.4 DS Laboratories

Many courses in DS should contain laboratory experiences. Typically, a laboratory experience lasts one to two hours and it occurs in a location configured with specialized equipment. The depth and breadth of these experiences will vary among institutions. The variation often depends on the time allocated within the curriculum, physical space, and resources. Table 73 illustrates the recommended types of laboratories students studying DS should experience. Some laboratories involve practices that all DS students must or should do. The laboratory type marked "**Must**" represents experience that are considered compulsory/necessary for every Data Science while "**Should**" refers to experience that are considered the best for every Data Science. The one marked "**Supplemental**" is considered as additional experience for Data Science.

Table 73: Types of Data Science laboratories

Laboratory Type	Must	Should	Supplemental
Data mining and warehousing	√		
Computer Architecture Design	√		
Data analytics	√		
Digital Logic and System Design			√
Embedded Systems			√
Networking		√	
Software Design		√	
Final Year Project Design	√		
Computers in Manufacturing			√
Block Chain Technology			√
Internet of things			√
Graphics			√
Mechatronics			√
Microwave Measurements			√
Operating Systems			√
Robotics			√
Specialized Electronics Lab			√
Teaching Enhancement			√

Laboratories should include some physical implementation of designs such as electronic and digital circuits, bread-boarding, FPGAs/CPLDs, microcontroller-based systems, prototyping, and implementation of firmware. Laboratories should also include application and simulation software to design computer systems including digital systems. Simulation tools present intrinsic value as part of professional DS practice. They are useful in modelling real systems and they are often desirable and necessary to allow students to study systems that are impractical to design and implement given available time and resources.

i) Structured Laboratories

A structured laboratory is a closed, scheduled, supervised experience in which students complete specified exercises. An instructor who is qualified to provide necessary support and feedback to the students provides supervision. Exercises are designed to reinforce and complement the lecture material.

ii) Open/Public Laboratories

Student ownership of computers has continued to increase. However, laboratories remain essential for those students who do not own a computer and for providing additional resources not available on personal machines.

iii) Specialized Laboratories/Studies

Laboratory facilities are necessary to support team projects and special computing environments. Special facilities may be needed for systems development, network infrastructure, and other advanced technologies.

11.13.5 Classrooms

Suitable classroom facilities, equipped with information technology teaching resources, should be provided. A computing system with multimedia facilities is necessary for demonstrating the development, implementation, and application of information technology as well as conducting walkthroughs and making presentations. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

11.13.6 Library

Library support is an important part of an academic programme. It is especially important for disciplines with rapid development of knowledge such as the Data Science field. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the computing professional societies.

CHAPTER 12

BENCHMARKS FOR A BACHELOR OF SCIENCE IN ARTIFICIAL INTELLIGENCE PROGRAMME

12.1 Artificial Intelligence Programme

Artificial Intelligence (AI) programme is designed for professionals who are interested in developing AI-based solutions. This study programme will provide learners not only with the skills and knowledge on how to apply AI algorithms but also its graduates will possess deep fundamental understanding of AI algorithms. This will enable graduates to determine algorithms efficiency, develop hybrid AI models using image and language processing and machine learning, as well as choosing appropriate architectural parameters for systems that aimed at implementing intelligent solutions.

AI is a discipline concerned with theoretical concepts, principles, and innovation in artificial intelligence, logic-based problem-solving techniques to design algorithms and software systems based on sound mathematical foundations, engineering and scientific procedures focused on artificial intelligence. The course curriculum includes theory sessions, tutorials and laboratory sessions based on computing, mathematical theoretical foundations and artificial intelligence.

AI emphasizes in real-world intelligent applications. Graduates of this programme will take up positions such as computer and information scientists, programmers and AI software engineers.

12.2 Artificial Intelligence Programme Objectives

The Bachelor of Science of Science in AI aims at preparing highly competent information technology specialists, who can develop intelligent software systems applying machine learning, image recognition, data processing and data analysis algorithms. Every AI programme should be designed to meet the following specific objectives.

12.2.1 Academic Ability

The programme objectives under this category are to equip students with knowledge/skills/ability to:

i) Analyse complex problems and synthesize solutions to those problems;

ii) Develop effective ways to solve computing problems employing AI;

iii) Design and implement software and software technologies base on AI;

iv) Undertake research and to progress to higher levels of studies;

v) Plan, design, deploy and document appropriate security for computer systems;

vi) Design, implement, and evaluate AI systems, process, component to meet desired needs;

vii) Verify, troubleshoot, test and analyse existing AI systems, process, component or programme; and

viii) Demonstrate proficiency in problem-solving techniques using AI technology.

12.2.2 Academia and Industrial Linkage/Employability

The programme objectives under this category are to equip students with knowledge and skills that match with industrial needs as follows:

i) Problem-solving skills for AI related tasks;

ii) Up-to-date AI skills for the industry;

iii) Analytical skills to understand impacts of computing on individuals, organizations and society;

iv) Ability to integrate theory and practice to work effectively and efficiently in organizations; and

v) Knowledge and skills that enable creativity, innovativeness and entrepreneurship in the field of AI.

12.2.3 Personal Development

The programme objectives under this category are to:

i) Prepare learners for lifelong learning and research;

ii) Empower students to progress in their personal career; impart professional ethics to the learner; equip the learner with skills and attitude to work in multicultural and global environments;

iii) Equip the learner with knowledge and skills to work as a team in the AI field; and

iv) Enable the learner to develop skills to perform effectively in technical and nontechnical environments.

12.3 Expected Learning Outcomes for Artificial Intelligence Programme

Expected Learning Outcomes (ELOs) provides a mechanism for describing not just a knowledge and relevant practical skills but also personal and transferable skills. ELOs are not of equal size and do not have a uniform mapping to curriculum hours, topics with the same number of hours may have quite different numbers of associated learning outcomes. Each learning outcome has an associated level of mastery. Some literature defines the levels based on Bloom's Taxonomy, which has been well explored within computing field. The Bloom's Taxonomy specify the cognitive skill level as follows:

i) **Familiarity:** The student understands the concepts and its meaning. This level of mastery concerns a basic awareness of a concept as opposed to expecting real facility with its application. It provides an answer to the question "What do you know about this?"

ii) **Usage**: The student is able to use or apply a concept in a concrete way. Using a concept may include, for example, appropriately using a specific concept in a programme, using a particular proof technique, or performing a particular analysis. It provides an answer to the question "What do you know how to do?"

iii) **Assessment**: The student is able to consider a concept from multiple viewpoints and/or justify the selection of a particular approach to solve a problem. This level of mastery implies more than using a concept; it involves the ability to select an appropriate approach from understanding alternatives. It provides an answer to the question "Why would you do that?"

The formulated ELOs as stipulated in Table 74 are the threshold: all graduates of the bachelor's programme in AI must achieve them.

Table 74: Expected Learning Outcome for Artificial Intelligence programme

Category	Expected Learning Outcomes	
Knowledge (K)	Graduate should be able to:	
	K1:	Demonstrate knowledge and understanding of concepts, theories and operations of a computer systems in relation to AI, the design of intelligent hardware and software for that systems.
	K2:	Demonstrate knowledge and understanding of fundamental mathematics and engineering science consistent with problem solving abilities of a degreed professional in the AI field.
	K3:	Demonstrate understanding of quality standards and benchmarks in design and development of AI computer-based systems and networks that involve both hardware and software.
	K4:	Demonstrate understanding of quality standards and benchmarks in computer software development.
	K5:	Demonstrate an ability to use current techniques, skills, and tools necessary for computing practice.
	K6:	Demonstrate an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modelling and design of AI-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices.
	K7:	Demonstrate an ability to apply design and development principles in the construction of AI software systems of varying complexity.
	K8:	Demonstrate understanding on institutional management, structure and governance.
Skills		
Cognitive Skills (SC)	SC1:	Analyse the impact of local and global trends of AI solutions on individuals, organizations, and society.
	SC2:	Demonstrate critical thinking, creativity and innovativeness in developing AI-based solutions to real world problems.

Category	Expected Learning Outcomes	
	SC3:	Assess risk related to AI-engineered activities.
	SC4:	Analyse a problem and then identify and define the computing and engineering requirements appropriate to the problem solution.
	SC5:	Evaluate the extent to which a AI-engineered system meets the criteria defined for its current use and scale up to future development.
Practical Skills (SP)	SP1:	Model, Design, implement and evaluate computers, AI-engineered systems, their hardware, software, network components as well as their integration to solve AI-engineering problems.
	SP2:	Deploy a variety of AI-based and laboratory tools and properly interpret and report experimental results.
	SP3:	Complete a sequence of design experiences, encompassing AI-hardware and AI-software elements and their integration, building on prior work, and including at least one major project. In this context, design refers to a level of ability beyond "assembling" or "configuring" systems.
	SP4:	Evaluate, verify, trouble-shoot, test and analyse existing AI engineered systems, their hardware, software and network components.
	SP5:	Identify and address engineering problems by extending the concepts of simple building blocks to system level design.
Interpersonal Skills (SI)	SI1:	Work effectively in a team.
	SI2:	Communicate effectively with experts and non-experts.
	SI3:	Demonstrate an understanding of professionalism, ethics, legal, security, social issues and responsibilities in CE.
Attitude (A)	A1:	Adapt to, and work in a multicultural and global AI environment.
	A2:	Work effectively as a team member as well as at individual level.

Category	Expected Learning Outcomes
	A3: Demonstrate commitment to lifelong learning, self and professional development. A4: Demonstrate self-confidence and ability to adapt to new situations.

12.4 Translating Learning outcome in the Basic Phase of Artificial Intelligence Programme

Bachelor programmes are commonly divided into foundation and specialization. The foundation phase is concerned with learning the core and supporting courses. During specialization phase, learners are given an opportunity to choose certain specializations according to their interests. In addition, there are elective courses which allow students to improve their scope of knowledge as well as shaping their personalities.

- **Core Courses (or Knowledge Areas):** These are essential courses offering a thorough foundation of the discipline. These subjects are mandatory for every student.
- **Supporting Courses (or Knowledge Areas):** These are courses for backing up the core courses. Without these courses it will be difficult to understand the core courses. For example, mathematics and basic computer studies are also compulsory to all students.
- **Elective Courses (or Knowledge Areas):** These are courses that can be taken by a student to deepen or broaden the knowledge, but they are not compulsory. However, a student has to make a choice to meet the minimum credit requirements for graduation.

Table 75**:** Core and supporting knowledge areas for Artificial Intelligence

shows the core knowledge and the supporting knowledge in the basic phase of Bachelor of Artificial Intelligence Engineering. Based on the core and supporting courses the following should be noted:

i) The titles of the courses may differ from one university to another. The emphasis should be in content rather than in the title;

ii) The autonomy and the uniqueness of the Higher Learning Institution (HLI) will be taken into consideration in formulating the core courses for the basic phase. A specific HLI will have a choice to add its own courses beyond the core and supporting courses; and

iii) The core and supporting courses may be designed in form of modules, course units as per the specific HLI's systems.

Table 75: Core and supporting knowledge areas for Artificial Intelligence

Core Knowledge Areas	**Supporting Knowledge Areas**
• Fundamentals of AI • Digital Design • Machine Learning • Deep Learning • Computer Programming • AI Engineering Fundamentals • Computer Networks and Communication • Systems Security • Platforms Technologies • Computer Networks and Communication • Human Computer Interaction • Algorithms and Complexity • Natural Language Processing • Decision Making and Robotics	• Mathematics and Statistics • General Studies

i) Applied Emerging Technologies

The identification and effects of applied emerging technologies on AI is useful in producing a competent engineer who can contribute to the profession in a changing world. AI programmes should allow the exploration of applied emerging technologies. AI are already on high demand due to exponential expansion of digitization of human activities.

ii) Conceptual Emerging Technologies

The artificial intelligence programme should allow exploration of new inventions to enable people and organizations to use machines to replace humans at work.

12.5 Credit Framework for Artificial Intelligence Programme

The credit framework for the Bachelor of Science in AI programme will be as presented in Table 76. This is consistent with the requirement in the University Qualifications Framework for a UQF level 8 programme. Activities for UQF level 8 programmes include but not limited to, lectures, seminars/tutorials, assignments, independent studies, and practical training. In this framework, these activities should normally be designed as presented in Table 76.

Table 76: Credit framework for conventional mode of learning

UQF Level	% on Lectures	% on Seminars and Tutorials	% on Assignments	% on Independent Studies and Research	% on Practical Training
8	40	20	10	20	10

UQF defines 480 as minimum cumulative credits required for a learner to graduate with a four-year UQF level 8 programme such as the Bachelor of Science in AI. It should be noted that, for a knowledge area, AI programme designers are flexible to assign more weight to an activity that heavily contributes to the student's learning. For instance, for a student to learn programming, more weight should be assigned to the practical activity. Universities offering the Bachelor of Science in AI should assign more weight on practical and industrial field activities as compared to theoretical learning.

12.6 Normal Learning Matrix for Artificial Intelligence Programme

The following sample learning matrix (Table 77) is a result of environmental scanning of programme curricula from various institutions for higher education minimum standards for Bachelor of AI. The matrix depicts varying credit weight to each knowledge area. A university should design an AI programme such that a student covers a minimum of 480 credit hours in four years for core knowledge areas as shown in Table 78.

The minimum standard to deliver lectures in the knowledge area of AI is proposed to be two (02) hours per week. Please note that equal learning weight of two (02) hours has been placed on delivering practical in the knowledge area of Fundamentals of AI. Therefore, for a typical 15 weeks semester, a course in Application of Cognitive Neuroscience would take 30 hours for lectures and 30 hours for practical per semester. The minimum credits to deliver knowledge in Fundamentals of AI area is 18.

Where:

AI1 represents Application of Cognitive Neuroscience

AI2 represents Artificial Intelligence Ecosystems

Table 77: Sample minimum credits for Artificial Intelligence Foundation as the knowledge area

Knowledge Units	# of Courses	Lectures Hrs.	Seminars and Tutorials Hrs.	Assign. Hrs.	Independent Studies and Research Hrs.	Practical Training Hrs.	Total Hours	Credits
% of Learning Activity		30	20	10	10	30		
AI1	1	30	15	7.5	7.5	30	90	9
AI2	1	30	15	7.5	7.5	30	90	9
Total minimum credit for Fundamentals of AI							**180**	**18**

Table 78 outlines the minimum credits for a Bachelor of Science in AI. Minimum credits for each of the 16 knowledge areas have been formulated following the same process as for AI foundation in Table 77. While an AI programme learning matrix may differ from one university to another, in terms of number of courses offered and weight of learning activities, more credit weight must be directed towards knowledge units and courses that distinguish AI graduates from other graduates.

Table 78: Normal Learning Matrix with minimum standard core credits

<table>
<tr><th colspan="2">Core Knowledge Areas</th></tr>
<tr><td>Fundamentals of AI [21]
• Application of Cognitive Neuroscience
• Artificial Intelligence Ecosystems
• Reactive Machines
• Data Science</td><td>Digital Design [27]
• The First Principles of Digital Logic
• Digital Image Processing
• Digital Signal Processing
• Hardware Description Languages
• Robotics and Automation</td></tr>
</table>

Core Knowledge Areas	
Machine Learning [32] • Machine Leaning for Structured Data • Machine Leaning for Text Mining • Advanced Data Analysis • Software Engineering for Machine Learning • Intermediate Machine Learning • Building Machine Learning Pipelines	**Deep Learning [32]** • Introduction to Deep Learning • Fundamentals of Multi-Agent Systems • Neural Networks • Intermediate Deep Learning • Deep Re-Enforcing Leaning and Control • Deep Learning Systems • Algorithms and Implementation
Data Science [27] • Data Analytics • Data Mining • Introduction to Big Data • Algorithms for Big Data Processing	**Computer Programming [27]** • Programming Fundamentals • Procedural Programming • Object-Oriented Programming • Logic Programming • Internet and Application Programming • Scripting Concepts • Assembly Programming
AI Engineering Fundamentals [32] • Design and Analysis of Computer Algorithms • Reactive Machines • Limited Memory Machines • Theory of Mind • Self-Aware Machines	**Systems Security [27]** • Network Security • Information Security • Software Security • Digital Forensic • Organization Security • Human/personnel and Societal Security
Platform Technologies [16] • Operating Systems • Open-Source Technologies • Computer Organization and Architecture	**Computer Networks and Communication [16]** • Communication and Networks • Computer Networks

Core Knowledge Areas	
• Mobile Technologies	
Human Computer Interaction [21] • Human Computer Interfacing • Identity Management • Social Engineering • Computer Vision	**Algorithms and Complexity [21]** • Data Structures and Algorithms • Algorithms Strategies • Automata, Computability and Complexity
Natural Language Processing [32] • Text Mining • Language Translations • Speech Processing • Computation Perception • Computation Photographs • Vision Sensors	**Decision Making and Robotics [32]** • Planning Techniques for Robotics • Neural Computations • Autonomous Agents • Cognitive Robotics • Mobile Robots Programming • Robotics Kinematics and Dynamics
Emerging Technologies [21] • Internet of Things • Data Science • Blockchain • Virtual Reality	
Supporting Knowledge Areas	
General Studies [21] • Communication Skills • Development Perspectives • Ethical Issues • Intellectual Property • Change Management	**Mathematics [21]** • Calculus • Discrete Mathematics • Linear Algebra • Probability and Statistics

12.7 Artificial Intelligence Course Description

All courses to be taught and learnt within the programme should follow the outline in Table 79.

Table 79: Course outline for Artificial Intelligence programme

Course Title	Refer to UQF qualification titles and nomenclatures
Course Aim	Course Aims are statements that describe the overarching intentions of a course. They should try to answer, "What is the purpose of the course?", "What is the course trying to achieve?"
Course Expected Learning Outcomes (s)	Course Learning Outcome should describe what students will be able to do by the end of the course in terms of knowledge, skills and competency
Course Status	Core or Elective
Credit Rating	Reference should be made to the UQF credit framework.
Total Hours Spent	Total number of hours spent in the course
Course Content:	Content of the course
Teaching and Learning Activities	Activities that would facilitate achievement of planned learning outcomes
Assessment Methods	An assessment criterion on how to achieve the outcomes in course expected learning outcome (s)
Reading List	Indicate up-to-date list of the required course textbooks, journals and the reference books

12.8 Learning Outcomes and Curriculum Alignment Matrix for Artificial Intelligence Programme

To check if the knowledge area covers the learning outcomes, it is important to develop a curriculum alignment matrix. For each knowledge area, one must formulate the specific learning outcomes for that course and have to check how far this course contributes to the programme learning outcomes. Table 80 is an example of a curriculum alignment matrix for the ELOs of the Bachelor of Science in AI. For each knowledge area, the contribution to the ELOs are given whereby K, SC, SP, SI and A stand for Knowledge, Skills-Cognitive, Skills-Practical, Skills-Interpersonal and Attitude, respectively.

Table 80: Curriculum Alignment Matrix for Artificial Intelligence programme

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**K5**	**K6**	**K7**	**K8**	**SC1**	**SC2**	**SC3**	**SC4**	**SC5**	**SC6**	**SC7**	**SP1**	**SP2**	**SP3**	**SP4**	**A3**	**A4**	**A5**	**A6**
Fundamentals of AI	x	x			x	x	x	x	x	x	x	x	x	x	x	x	x	x	x				
Digital Design	x	x	x	x	x	x	x	x	x	x			x		x	x	x	x	x				
Machine Learning	x	x							x	x	x	x	x	x	x	x	x		x				
Deep Learning	x	x							x	x	x	x	x										
Data Science	x	x	x	x																			
Computer Programming	x	x	x	x	x	x		x	x	x	x	x	x		x	x	x	x					
AI Engineering Fundamentals	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x					
Systems Security	x	x	x						x	x	x	x	x	x	x	x	x	x					
Platforms Technologies	x	x	x						x	x	x	x	x	x	x	x	x	x					
Computer Networks and Communication	x	x	x						x	x	x	x	x	x	x	x	x	x					
Human Computer Interaction	x		x						x		x												

Core Knowledge Areas	**K1**	**K2**	**K3**	**K4**	**K5**	**K6**	**K7**	**K8**	**SC1**	**SC2**	**SC3**	**SC4**	**SC5**	**SC6**	**SC7**	**SP1**	**SP2**	**SP3**	**SP4**	**A3**	**A4**	**A5**	**A6**
Algorithms and Complexity	x	x	x	x	x	x	x																
Natural Language Processing	x	x	x																				
Decision Making and Robotics	x	x	x						x	x	x	x	x			x	x	x	x				
Emerging Technologies	x	x	x	x					x	x	x	x	x										
Supporting Knowledge Areas																							
Mathematics and Statistics	x	x	x																				
General Studies				x																x	x	x	x

12.9 Minimum Entry Qualification for Artificial Intelligence Programme

Entry requirements for admission into the Bachelor of Science in Data Science programme should be in line with the Commission's minimum entry and specific admission requirements. Candidates with A-Level passes should have sufficient background in Physics and Mathematics and for equivalent qualification, a first-class Diploma in AI or CE or SE or EE, or TE from a recognised institution.

12.10 Artificial Intelligence Programme Duration

An AI programme typically reflects Computer Science and Neural Science knowledge areas in varying weights coverage depending on the vision of the AI department. Due to the intensity of knowledge area coverage, the programme duration is recommended to be four (4) years, with minimum of two (2) semesters per academic year.

12.11 Graduation Requirements for Artificial Intelligence Programme

An AI learner is required to accumulate a minimum of 420 core credits which accounts to 4,200 notional hours of learning in order to graduate. Sixty (60) credits have been left open for the learner to pursue elective knowledge units. Universities should ensure learners undertake one (1) Practical Training within each academic year.

12.12 Evaluation and Course Assessment for Artificial Intelligence Programme

12.12.1 Evaluation

Universities should offer up-to-date courses to learners in the AI programme. Due to the rapid changes in AI field and its emerging technologies, the design and implementation of curricula should be regularly reviewed and evaluated. The assessment can be done in several ways including: written student evaluations, in-class observations, industrial symposium, and personal interviews with various stakeholders such as academia, alumni, and experts from the ICT industries. Once the course has been taught, it is a good practice to interview graduates regarding the value of the course to their professional work environment. During the evaluation, tracer studies and stakeholders' consultation should be conducted to assess the performance of graduates. Also, feedback from students and examiners can be used to improve the delivery of the programme.

12.12.2 Assessment

Assessment mode should be both formative and summative. The formative mode includes; quizzes, assignments, lab presentations, and projects, etc. The summative mode involves tests, and examinations.

12.13 Resources Requirements for Artificial Intelligence Programme

12.12.3 Human Resource Requirements

Staff members are vital to the strength of an AI programme. Its faculty needs both academic training and practical experience. There must be enough staff to provide course offerings that allow the students to complete a degree in a timely manner. The interests and qualifications of the faculty members must be enough not only to teach the courses but also to plan and modify the courses and curriculum to meet the market demand. Faculty members must remain current in the discipline. Professional development and scholarly activities are a joint obligation of the institution and the individual faculty members. The programme is enhanced significantly when faculty acquire practical experience in the profession through activities such as training, consulting, sabbatical leaves, and industry exchange programmes. Faculty must also be equipped to develop teaching materials for their students. Faculty must have available technology at least equivalent to and compatible with that available to students so that they may prepare educational materials for use by students.

The number of full-time staff needed by the programme is influenced by such factors as the number of students in the programme, the number of required courses, the number of service and elective courses offered, and the teaching load of the faculty. A programme should have a minimum number of full-time staff members with primary commitment to the AI programme in order to meet the teaching and advising needs of the programme and to provide depth and breadth of faculty expertise.

Laboratory experiences are an essential part of the AI curriculum, and they serve multiple functions. It is important that AI students have many opportunities to observe, explore, and manipulate characteristics and behaviours of actual devices, systems, and processes. This includes designing, implementing, testing, and documenting hardware and software, designing experiments to acquire data, analysing and interpreting that data, and using that data to correct or improve the design and to verify that it meets specifications. Universities should seek out options that best suit their needs based on available space, the knowledge unit objectives, and the resources available.

12.12.4 Artificial Intelligence Laboratories

Any course in AI should be supported with adequate laboratory facilities. These facilities are vital for students to perform laboratory activities leading to some experience. Typically, a laboratory experience lasts one to two hours and it occurs in a location configured with specialized equipment. The depth and breadth of these experiences will vary among institutions. The variation often depends on the time allocated within the curriculum, physical space, and resources. Table 81 illustrates the recommended types of laboratories students studying AI should experience. Some laboratories involve practices that all AI students must or should do. The laboratory type marked "**Must**" represents experience that are considered compulsory/necessary for every AI while "**Should**" refers to experience that is considered the best for every AI. The one marked "**Supplemental**" is considered as additional experience for AI.

Table 81: Types of Artificial Intelligence laboratories

Laboratory Type	Must	Should	Supplemental
AI fundamentals	√		
Computer Architecture Design			√
Hardware Design and Implementation	√		
Reactive Machines	√		
Theory of Mind	√		
Self-Aware Machines		√	
Robotics		√	
Final Year Project Design	√		
Computers in Manufacturing			√
Audio Engineering			√
Electricity Energy Systems			√

Graphics			√
Mechatronics			√
Microwave Measurements			√
Operating Systems			√
Robotics			√
Specialized Electronics Lab			√
Teaching Enhancement			√

Laboratories should include some physical implementation of designs such as electronic and digital circuits, bread-boarding, FPGAs/CPLDs, microcontroller-based systems, prototyping, and implementation of firmware. Laboratories should also include application and simulation software to design computer systems including digital systems. Simulation tools present intrinsic value as part of professional AI practice. They are useful in modelling real systems, and they are often desirable and necessary to allow students to study systems that are impractical to design and implement given available time and resources.

12.12.5 Software Tools

Software tools and packages related to AI will vary based on the philosophy and needs of each programme. Table 82 suggests some software that could appear on all machines within specific laboratory settings. Products mentioned in this table are included for illustrative purposes only and that no endorsement of a specific product is implied. Additionally, it is not envisioned that any programme will incorporate every one of these software applications. Each programme should determine its own needs and consider including the most current version of appropriate applications.

Table 82: Suggestions for possible tools

AI Tools	Digital Tools
• Data analysis and Visualization Tools • Modelling and Simulation Tools • Algorithms implementation Tools	• Microcontroller Systems • Integrated Circuits/ASIC Design • Systems Engineering Tools • General Computing/Productivity • Digital Hardware Prototyping • Microcontroller system Design • Lab Automation and Instrumentation • Computer aided Design and Modelling (CAD tools) • Printed Circuit Board (PCB) Design • Other tools

12.12.6 Open-ended Labs

The AI curriculum often contains open-ended experiences where true research and development takes place. One might view this as the "ultimate lab experience." A culminating or capstone design experience usually embodies this open-ended flavour. In such situations, an instructor and a team of students decide on an exploration area and, once decided, the student team begins the research and design process. Programmes usually provide a dedicated space where teams can meet and work. These spaces generally contain modern facilities and provide enough space for electronic devices (e.g., robots) and other equipment required by the project at hand.

12.12.7 Embedded Labs

The traditional laboratory experience normally takes place in a room separate from the lecture and at a different time. For example, a graphics course might have lectures on Tuesday and Thursday, with a three-hour laboratory on Wednesday. It is increasingly common in modern educational settings to have laboratories embedded within ordinary courses. One such practice is to have lectures take place within the laboratory itself. Another is to partition a room in some manner and have the lecture take place in one part of the room with the laboratory and its associated equipment in another part of the same room. There is also an emerging trend to have "flipped classrooms" in which lectures are recorded for students to view in advance. Then class time is used to engage the students in active learning, which could include students bringing their own breadboards and instrumentation and engaging in laboratory exercise.

12.12.8 ICT Infrastructure Requirements

ICT infrastructure consists of hardware, software, and technical support. Adequate computing facilities are essential for effective delivery of the AI programme though the form in which this infrastructure is allocated has changed significantly. These formerly involved a blend of computer facilities of varying capabilities and complexity. Most incoming undergraduates enter college with computer resources, so access plays a much more significant role. Therefore, network access should be available for faculty and students to use with their own computers. Classrooms should have access to the internet and extranet networks, either with port per seat or wireless networking capabilities.

In addition to software and hardware, it is paramount that these tools have adequate technical support. Modern computing infrastructure is highly complex requiring technically trained support staff to maintain the equipment. This is beyond the scope of faculty duties, a waste of precious faculty resources, and often outside their individual expertise. Support staff who maintain hardware, software and communications resources rarely have overlapping skills an interest in teaching due to the focus on product design and provider relationships. These technical experts are a vital necessity in a campus environment.

12.12.9 Library

Library support is an important part of an academic programme. It is especially important for disciplines with rapid development of knowledge such as AI. Libraries should provide both traditional and digital access wherever possible to journals, proceedings, monographs, and reference books. The holdings should include access to digital journals and proceedings of the ICT professional societies.

REFERENCE

1. ACM/IEE. (2005). Computing Curricula 2005.
2. ACM/IEE. (2008). ACM Curriculum Guidelines for Undergraduate Degree Programmes in Information Technology, 1–139.
3. Technology, C. (n.d.). Information and Communication Technology for the School System.
4. ACM/IEE. (2019). ACM Computing Competencies for Undergraduate Data Science Curricula, (January).
5. ACM/IEE. (2016). ACM Information Technology Curricula 2017, 1–112.
6. ACM/IEE. (2017). Cybersecurity Curricula 2017.
7. Karsten, H., & Brown, S. A. (2016). MSIS 2016 Global Competency Model for Graduate Degree Programs in Information Systems.
8. Asmal, K. (2004). Creating Comprehensive Universities In South Africa: A Concept Document Department of Education January 2004, (January).
9. Inter-University Council for East Africa. (2015). Benchmarks For the Bachelor of Computer Science And The Bachelor Of Information Technology Benchmarks.
10. Education, U. N. C. F. H. (2014). Minimum Standards for Bachelor of Science in Computer Engineering Bachelor of Information Systems Bachelor of Information Technology Bachelor of Library and Information Science Bachelor of Science in Software Engineering, (January).
11. Africa, D. of E. R. of S. (n.d.). South Africa Formal Degree-Diploma-Certificate Programme classification structure manual.pdf.
12. ACM/IEE. (2016). Computer Engineering Curricula 2016.
13. ACM/IEE. (2013). Computer Science Curricula 2013.
14. Wright, R. T., Kaiser, K. M., & Nunamaker, J. F. (2010). IS 2010 Curriculum Guidelines for Undergraduate Degree Programs in Information Systems.
15. Minimum, B., Standards, A., & Universities, N. (2015). Nigeria National universities commission, (November 2014).

16. ACM/IEE. (2017). Information Technology Curricula 2017.
17. ACM/ IEEE (2020) Computing Curricula 2020, CC2020
18. ACM/IEEE Computing Competencies for Undergraduate Data Science Curricula, 2021.
19. Universities, T. C. for. (2012). University Qualifications Framework.
20. Callahan, D. & Pedigo, B. (2002). Educating experienced IT professionals by addressing industry's needs. IEEE Software,19, 5, 86–93
21. Brinkerhoff, Derick. June 1998. From Design to Implementation: Stakeholder Analysis in a PHC Project in India. Bethesda, MD: Abt Associates Inc.
22. Crosby, Benjamin L. March 1992. "Stakeholder Analysis: A Vital Tool for Strategic Managers." Technical Notes, no. 2. Washington, DC: Implementing Policy Change Project for the US Agency for International Development (USAID).
23. Schmeer, Kammi. September 1998. "Process for Developing an Interest Map in Ecuador," Technical Report no. 23. PHR Project. Bethesda, MD: Abt Associates Inc.
24. ARU, Bachelor of Science in Information Systems Management
25. MUST, Bachelor of Computer Engineering
26. MOCU, Bachelor Science in Business Communication and Information Technology
27. MU, Bachelor of Science in Information Communication Technology with Management (2018)
28. MU, Bachelor of Science in Information Technology Systems
29. MU, Bachelor of Science in Information Technology with Business
30. OUT, Bachelor of Science in ICT
31. RUCU, Bachelor of Computer Science
32. RUCU, Bachelor of Accounting and Finance with Information Technology
33. RUCU, Bachelor of Science in Computer Science in Software Engineering

34. SUA, Bachelor of Science in Informatics
35. SJUIT, Bachelor of Electrical & Electronics Engineering
36. SJUIT, Bachelor of Electronics & Communication Engineering
37. SJUIT, Bachelor of Engineering in Computer Science and Engineering
38. SUZA, Bachelor of IT Application & Management
39. SUZA, Bachelor of Science in Computer Science
40. SUZA, Bachelor Degree in Information Technology with Accounting
41. TURDA Co, Bachelor of Information Management
42. UAUT, Bachelor of Science in Computer Engineering and Information Technology
43. UDSM, Bachelor of Science in Computer Engineering and Information Technology
44. UDSM, Bachelor of Science in Telecommunications Engineering
45. UDSM, Bachelor of Science in Computer Science
46. UDSM, Bachelor of Science in Electronic Science and Communication
47. UDSM, Bachelor of Science with Computer Science
48. UDSM, Bachelor of Science in Electronics Engineering
49. UDSM, Bachelor of Science in Business Information Technology
50. UDOM, Bachelor of Science in Computer Science
51. UDOM, Bachelor of Science in Information Systems
52. UDOM, Bachelor of Science in Software Engineering
53. UDOM, Bachelor of Science in Business Information Systems
54. UDOM, Bachelor of Science in Computer Engineering
55. UDOM, Bachelor of Science in Telecommunications Engineering
56. UDOM, Bachelor of Education in Science with ICT
57. UDOM, Bachelor of Science in Health Information Systems

58. UDOM, Bachelor of Science in Computer Networks and Information Security Engineering
59. UDOM, Bachelor of Commerce in Information Systems Management
60. UDOM, Bachelor of Science in Cyber Security and Digital Forensics Engineering
61. UDOM, Bachelor of Science in ICT Mediated Content Development
62. UDOM, Bachelor of Science in Multimedia Technology & Animation
63. UDOM, Bachelor of Science in Instructional Design and Information Technology
64. UDOM, Bachelor of Science in Digital Content and Broadcasting Engineering
65. ZU, Bachelor of Science in Computer Engineering and Information Technology
66. ZU, Bachelor of Science in Telecommunications Engineering