Programme Content – **Master of Information Technology**

<table>
<tr><td>Course Code</td><td colspan="3">IMIT 53203</td></tr>
<tr><td>Course Name</td><td colspan="3">Computer Systems</td></tr>
<tr><td>Prerequisites</td><td colspan="3">Entry qualifications</td></tr>
<tr><td>Credit Value</td><td colspan="3">3</td></tr>
<tr><td>Compulsory/Optional</td><td colspan="3">Compulsory</td></tr>
<tr><td rowspan="2">Hourly Breakdown
(Notional Hrs.)</td><td>Theory</td><td>Practical</td><td>Guided Self-study</td></tr>
<tr><td>150</td><td>0</td><td>0</td></tr>
<tr><td colspan="4">Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,
▪ describe the evolution and classification of computers
▪ explain how data and instructions are represented in computers
▪ explain how the combinational and sequential circuits perform computer operations
▪ compare and contrast different computer architectures
▪ describe functionality and working of the building blocks of computers
▪ explain the key aspects, functionalities and working of operating systems
▪ describe the concepts, models and approaches used in design of operating systems</td></tr>
<tr><td colspan="4">Course Content:
Evolution of computers, computer classification, parts of an information system, software classification, hardware components, connectivity and communication systems, data and instruction representation, combinational and sequential circuits, von Neumann and non von Neumann architectures, fetch-execute cycle, microprocessors, instruction set architectures, instruction pipelining, operand addressing, microcode, parallelism, static and dynamic RAM, byte and word addressing, caching, I/O and bus architectures, interrupt handling, evolution of operating systems, roles of an operating system, process models, processor scheduling, concurrency control, memory and file management, privileged modes and protection levels, notable computer architectures and operating systems, current trends and future insights</td></tr>
<tr><td colspan="4">Teaching /Learning Methods:
Lectures and guided self-study assignments</td></tr>
<tr><td colspan="4">Assessment Criteria:
End of course unit examination and continuous assessments</td></tr>
<tr><td colspan="2">Continuous Assessment
25 – 50%</td><td colspan="2">Final Assessment
50 – 75%</td></tr>
<tr><td>Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %</td><td>Theory (%)
100%</td><td>Practical (%)
0%</td><td>Other (%)(specify)</td></tr>
<tr><td colspan="4">Recommended Reading:
1. T. O'Leary, L. O'Leary and D. O'Leary, Computing Essentials 2017: Making IT work for you, 26th Edition, McGraw-Hill, 2017
2. D. E. Comer, Essentials of Computer Architecture, Chapman and Hal, 2nd Edition, 2017
3. W. Stallings, Operating Systems: Internals and Design Principles, 9th Edition, Pearson, 2017
4. W. Stallings, Computer Organization and Architecture: Designing for Performance, 11th Edition, Pearson, 2018
5. Material provided in CAL</td></tr>
</table>

Course Code	IMIT 53213		
Course Name	Programming Concepts		
Prerequisites	Entry qualifications		
Credit Value	3		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided self-study
	125	Within Lecture Hours	25

Course Aim/Intended Learning Outcomes:

After completing this module, the students should be able to,

- design an algorithm to solve a given practical problem
- specify the designed algorithm in flowcharts/pseudo code
- use programming constructs to implement an algorithm
- decide when to use selection and repetition constructs
- use structured data types in computer programs
- use functions as a fundamental program building block
- use file processing to work with data stored in secondary storage
- perform database access through simple GUIs
- develop and test efficient and reliable computer program for a given problem

Course Content:

Design of algorithms, program execution cycle, evolution of programming languages, low-level and high-level languages, language translation, imperative, declarative and object oriented programming languages, comments, elementary data types, variables and constants, expressions and statements, programming constructs, mathematical, relational and Boolean operators and operator precedence, string operations, type casting, functions, parameters and arguments, recursion, arrays, lists, dictionaries and tuples, files and file operations, use of libraries and services, searching and sorting techniques, exception handling, Graphical User Interfaces (GUIs), database programming, input validation, design of reports and data visualization, debugging techniques, programming standards and best practices

Teaching /Learning Methods:

Lectures, supervised practical and guided self-study assignments

Assessment Criteria:

End of course unit examination and continuous assessments

Continuous Assessment 25 – 50%	Final Assessment 50 – 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. K. A. Lambert, The Fundamentals of Python: First Programs, Cengage Learning, 2nd Edition, 2018
2. A. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, O'Reilly Media, 2015
3. Bhaskar Chaudhary, Tkinter GUI Application Development Blueprints, Packt Publishing, 2nd Edition, 2018
4. Materials provided in CAL

Course Code	IMIT 53223		
Course Name	Database Concepts		
Prerequisites	Entry qualifications		
Credit Value	3		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	150	Within Lecture Hours	0

Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,

- identify data / information needs of an organization
- describe the techniques used in storing and retrieving data
- design local data models
- develop databases using relational model
- define and manipulate data using Structured Query Language (SQL)
- recognize new trends in databases

Course Content:
Data and information: structured, semi-structured and un-structured data, database approach, role of database management systems in organizations, different database architectures, relational database design and development: ER modeling, normalization, Structured Query Language (SQL): DDL, DML, data warehousing and dimensional data modeling, NoSQL databases, new trends in databases

Teaching /Learning Methods:
Lectures, supervised practical and guided self-study assignments

Assessment Criteria:
End of course unit examination and continuous assessments

Continuous Assessment 25 – 50%	Final Assessment 50 – 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, 7th Edition, Pearson Education, 2017
2. C. Coronel and S. Morris, Database Systems: Design, Implementation and Management, 13th Edition, Cengage Learning, 2018
3. T. Connolly and C. Begg, Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition, Pearson Education, 2014
4. G. Harrison, Next Generation Databases: NoSQL and Big Data, Apress (2015)
5. Materials provided in CAL

Course Code	IMIT 53233		
Course Name	Data Communication and Computer Networks		
Prerequisites	Entry qualifications		
Credit Value	3		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	150	Within Lecture Hours	0

Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,

- explain the purpose and methods of communication among computer systems
- compare and contrast different technologies related to data communication
- differentiate underlying functions of data communication in computer networks
- identify hardware and software for data communication
- describe different network types and network protocols
- explain the key concepts related to computer network administration and security
- configure and troubleshoot basic computer networks

Course Content:
Signal types and properties, wired and wireless transmission, data encoding and protocols, modems and Public Switched Telephone Networks (PSTN), network devices and topologies, Media Access Control, internetworking, IP addressing and routing, reliable and non-reliable end-to-end communication, Domain Name System, Open Systems Interconnection (OSI) Reference Model, TCP/IP Model, common threats and security measures, Internet Service Providers (ISP), Internet and its services

Teaching /Learning Methods:
Lectures, supervised practical and guided self-study assignments

Assessment Criteria:
End of course unit examination and continuous assessments

Continuous Assessment 25 – 50%	Final Assessment 50 – 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. C. White, Data Communications and Computer Networks: A Business User's Approach, 8th Edition, Cengage Learning, 2015
2. C. Meinel and H. Sack, Internetworking: Technological Foundations and Applications, Springer, 2013
3. W. Odom, CCNA Routing and Switching 200-125 Official Cert Guide Library, Cisco Press, 2016
4. Materials provided in the CAL

Course Code	IMIT 53243		
Course Name	Web and Multimedia Technologies		
Prerequisites	Entry qualifications		
Credit Value	3		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	125	Within Lecture Hours	25

Course Aim/Intended Learning Outcomes:

After completing this module, the students should be able to,

- explore the need of web applications
- analyze user requirements related to web application development
- describe the web application architecture
- explain how static and dynamic web sites are designed and engineered
- describe the core components of a web-based application
- apply multi-media technologies to develop content and web applications
- design and develop a dynamic, responsive web-based applications

Course Content:

The World Wide Web, types of web sites, web application architecture, client-server model, web servers, HTML, CSS, client-side scripting, Document Object Model, server-side development, working with databases, error handling and validation, managing state, asynchronous web applications, advanced CSS and responsive design, Model-View Controller (MVC) architecture, digital representation of images, color models, image concepts, file formats, animation, audio and video

Teaching /Learning Methods:

Lectures, supervised practical and guided self-study assignments

Assessment Criteria:

End of course unit examination and continuous assessments

Continuous Assessment 25 – 50%	Final Assessment 50 – 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. R. Connolly and R. Hoar, Fundamentals of Web Development, 2nd Edition, Pearson, 2018
2. P. J. Deitel and H. M. Deitel, Internet & World Wide Web How to Program, Pearson, 2012
3. L. Welling and L. Thomson, PHP & MySQL Web Development. Addison-Wesley, 2017
4. Material provided in CAL

Course Code	IMIT 53253		
Course Name	Software Engineering		
Prerequisites	Entry qualifications		
Credit Value	3		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	150	0	0

Course Aim/Intended Learning Outcomes:

After completing this module, the students should be able to,

- explain the evolution of system analysis and deign
- explain the stages and activities carried out in the system development life cycle
- discuss the different system development models and methodologies
- explain the different tools and techniques for systems analysis and design
- use appropriate methods and techniques to produce an analysis of a given scenario
- identify the requirements and prepare system requirement specification
- use appropriate modelling techniques to produce a system design for a given scenario
- explain development and testing approaches to implement a computerized system

Course Content:

Concept of systems, classification of systems, evolution of system analysis and design, system development life cycle, system development models, system development methodologies: structured and object oriented methodologies, architectural design, program specification, user interface design, database design, software testing, parallel, implementation strategies, support and maintenance, Computer Aided Software Engineering (CASE), software project management

Teaching /Learning Methods:

Lectures, supervised practical and guided self-study assignments

Assessment Criteria:

End of course unit examination and continuous assessments

Continuous Assessment 25%	Final Assessment 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. P. Weaver, N. Lambrou and M. Walkley, Practical Business Systems Development Using SSADM, 3rd Edition, Financial Times/Prentice Hall, 2002
2. J. A. Valacich, J. George and J. A. Hoffer, Essentials of Systems Analysis and Design, 6th Edition, Pearson, 2014
3. I. Sommerville, Software Engineering, 10th Edition, Pearson, 2015
4. R. Pressman and B. Maxim, Software Engineering: A Practitioner's Approach, 8th Edition, McGraw-Hill, 2014
5. Material provided in CAL

Course Code	IMIT 53263		
Course Name	Internet of Things		
Prerequisites	IMIT 53203, IMIT 53213, IMIT 53233, IMIT 53243		
Credit Value	3		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	125	Within Lecture Hours	25

Course Aim/Intended Learning Outcomes:

After completing this module, the students should be able to,

- discuss general purpose computer systems vs. embedded systems
- describe the components of an embedded system
- compare different implementations of embedded systems
- describe how embedded systems are programmed
- design and develop embedded systems for automating real world tasks
- explain the concept of Internet of Things (IoT)
- discuss the enabling technologies for IoT
- discuss the various applications of IoT
- design and develop IoT applications to make the day-to-day life smart
- discuss the social and security concerns of IoT

Course Content:

Introduction to embedded systems, requirements and constraints, single-chip and single-board computers, development platforms, hardware software co-design, microcontrollers, microprocessors, sensors, actuators, development boards, hardware programmers, bootloaders, operating systems and device drivers, analog to digital conversions, interrupts, pulse width modulation, inter device communication, data acquisition and logging, firmware development, debugging and simulating, concept of Internet-connected devices, enabling technologies, IoT applications, Internet connectivity, communication protocols, machine to human and machine to machine interactions, IoT platforms, design and implementation of IoT-based systems, security concerns of IoT and social impact of IoT devices

Teaching /Learning Methods:

Lectures, supervised practical sessions and guided self-study assignments

Assessment Criteria:

End of course unit examination and continuous assessments

Continuous Assessment 25 - 50%	Final Assessment 50 - 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. M. A. Mazidi, S. Naimi and S. Naimi, AVR Microcontroller and Embedded Systems: Using Assembly and C, 2nd Edition, Micro Digital Ed, 2018
2. S. Monk, Programming Arduino Next Steps: Going Further with Sketches, 2nd Edition, McGraw-Hill, 2018
3. P. Seneviratne, Internet of Things with Arduino Blueprints, Packt Publishing, 2015
4. J. M. Hughes, Arduino: A Technical Reference: A Handbook for Technicians, Engineers, and Makers (In a Nutshell), O'Reilly Media, 2016
5. Material provided in CAL

Course Code	IMIT 53272		
Course Name	Intelligent Systems		
Prerequisites	Entry qualifications		
Credit Value	2		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	100	0	0

Course Aim/Intended Learning Outcomes:

After completing this module, the students should be able to,

- compare and contrast natural intelligence and artificial intelligence
- appreciate man-machine and machine-machine coexistence
- explain key artificial intelligence techniques
- explain the role of agent and multi-agent systems in intelligent systems
- evaluate the applicability of intelligent techniques with respect to a given application
- explain the key ongoing developments pertaining to intelligent systems

Course Content:

Intelligent (and emotional) computing; man-machine and machine-machine coexistence, software agents, artificial intelligence techniques: search, knowledge representation, expert systems, artificial neural networks, genetic algorithms, fuzzy logic, multi-agent systems, natural language processing, machine learning, speech and vision-based system, emerging techniques of intelligent computing

Teaching /Learning Methods:

Lectures and self-study assignments

Assessment Criteria:

End of course unit examination and continuous assessments

Continuous Assessment 25%	Final Assessment 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall, 2015
2. G. Sarkar, R. Bali and T. Sharma, Practical Machine Learning with Python, APress, 2018
3. A. Ethem, Introduction to Machine Learning, MIT Press, 2010
4. Material provided in CAL

Course Code	IMIT 53282		
Course Name	Management Information Systems		
Prerequisites	IMIT 53203, IMIT 53213, IMIT 53223, IMIT 53233, IMIT 53243, IMIT 53253, IMIT 53263, IMIT 53272		
Credit Value	2		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	100	0	0

Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,

- describe business information systems and infrastructure
- describe different types of information systems used in business organizations
- explain how information systems enable organizations to be competitive
- explain the steps of information systems development process
- assess emerging technologies to develop competitive information systems
- explain different e-business strategies
- evaluate the ethical concerns and security measures

Course Content:
Components of information systems, role of information systems in a digital economy, key information systems infrastructure and emerging technologies, the strategic use of information systems, information system types and applications, electronic commerce and global e-business platforms, the importance of secure payment mechanisms and cryptocurrencies, information systems security, different models of building and acquiring business information systems, ethical use of information systems and their use on crime, war and terrorism

Teaching /Learning Methods:
Lectures and self-study assignments

Assessment Criteria:
End of course unit examination and continuous assessments

Continuous Assessment 40%	Final Assessment 60%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. C. L. Kenneth and P. L. Jane, Management Information Systems, 14th Edition, Pearson, 2016
2. A. O. James and M. M. George, Management Information Systems, 9th Edition, McGraw Hill, 2010
3. E. Turban, R. Sharda and D. Delen, Business Intelligence and Analytics: Systems for Decision Support, 10th Edition, Pearson, 2014
4. Material provided in CAL

Course Code	IMIT 53293		
Course Name	Advanced Programming		
Prerequisites	IMIT 53203, IMIT 53223		
Credit Value	3		
Compulsory/Optional	Optional		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	125	Within lecture hours	25

Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,

- describe the principles of object-oriented and event-driven programming
- compare and contrast object-oriented vs. structured programming and event-driven vs. command-line programming
- characterize the commonly used events
- describe the properties, methods and events of reusable objects
- explain the use of object-oriented programming concepts in event-driven programming
- choose appropriate widgets and their events to develop Graphical User Interfaces
- develop real-world software applications using object-oriented, event-driven programming languages

Course Content:
Principles of object oriented and event-driven programming, objects, classes and subclasses, interface and implementation, access modifiers, encapsulation and information hiding, inheritance, aggregation, associations, properties, methods and events, static binding and dynamic binding, polymorphism, overloading and overriding, event propagation, detection and handling, Integrated Development Environment (IDE), exception handling, class hierarchies, Graphical User Interfaces (GUIs), widgets and callbacks, canvases, buttons, labels, entries and texts, scrollbars, frames, packing widgets, menus and callables, database programming, input validation, design of reports and data visualization, custom components and reusable libraries.

Teaching /Learning Methods:
Lectures and guided self-study assignments

Assessment Criteria:
End of course unit examination and continuous assessments (including a group mini-project)

Continuous Assessment (including a group mini-project): 40 - 60%	Final Assessment 40 - 60%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. Allen Downey, Think Python: How to Think Like a Computer Scientist, 2nd edition, O'Reilly Media, 2015
2. Bhaskar Chaudhary, Tkinter GUI Application Development Blueprints, 2nd Edition, Packt Publishing, 2018
3. Albert Lukaszewski, MySQL for Python, Packt Publishing, 2010
4. Material provided in CAL

Course Code	IMIT 53302		
Course Name	Machine Learning		
Prerequisites	IMIT 53293		
Credit Value	2		
Compulsory/Optional	Optional		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	75	0	25

Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,

- explain different machine learning techniques
- develop programs to implement machine learning algorithms
- apply machine learning techniques to recognize patterns in data
- use available machine learning tools in Python, R and Octave for real applications

Course Content:
Machine Learning and Analytics, Supervised Learning, Discriminative Algorithms, and Generative Algorithms, Linear models for classification and regression, Logistic Regression, Bayesian Learning, Artificial Neural Networks, Support Vector Machines (SVM), Monte Carlo Methods, Unsupervised Learning, Expectation Maximization Algorithms, Factor Analysis, Principal Components Analysis, Independent Components Analysis, Reinforcement Learning and Control, Deep Learning

Teaching /Learning Methods:
Lectures and guided self-study assignments in R or Python

Assessment Criteria:
End of course unit examination and continuous assessments

Continuous Assessment 25%	Final Assessment 75%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. Bishop, C., Pattern Recognition and Machine Learning, Springer, 2006
2. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly Media, 2017
3. Nikhil Buduma, Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms, O'Reilly Media, 2017

Course Code	IMIT 53313		
Course Name	Big Data Analytics		
Prerequisites	IMIT 53302		
Credit Value	3		
Compulsory/Optional	Optional		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	125	Within lecture hours	25

Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,

- identify the basic characteristics of big data
- explain the business and technological opportunities/challenges imposed by big data
- appreciate the role of big data analytics in decision support systems
- explain key analytics techniques
- apply appropriate analytics techniques to solve practical problems in data science

Course Content:
Introduction to big data and analytics, Data understanding, Data preprocessing, Data warehousing and online analytical processing, Data cube technology, Association rule mining, Classification methods, Cluster analysis, Outlier analysis, Analysis of complex and unstructured data, Social implications of big data analytics

Teaching /Learning Methods:
Lectures and guided self-study assignments in R or Python

Assessment Criteria:
End of course unit examination and continuous assessments

Continuous Assessment 40 - 60%	Final Assessment 40 - 60%		
Details: quizzes %, mid-term %, Assignments% 0 - 50 %, 0 – 50 % 0 – 50 %	Theory (%) 100%	Practical (%) 0%	Other (%)(specify)

Recommended Reading:

1. Jiawei Han, Micheline Kamber, Jian Pei, Data Mining Concepts and Techniques, 3rd Edition, Morgan Kaufmann, 2015
2. Ian H. Witten, Data Mining: Practical Machine Learning Tools and Techniques, 3rd Edition, Morgan Kaufmann Series in Data Management Systems, 2011
3. Flash, P., Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, 2012
4. Bowles, M., Machine Learning in Python - Essential Techniques in Predictive Analysis, Wiley, 2015
5. Ronald Davis, BIG DATA and DATA ANALYTICS: The Beginner's Guide to Understanding the Analytical World, CreateSpace Independent Publishing Platform, 2017

Course Code	IMIT 53323		
Course Name	Seminar on Emerging Technologies in ICT		
Prerequisites	N/A		
Credit Value	3		
Compulsory/Optional	Compulsory		
Hourly Breakdown (Notional Hrs.)	Theory	Practical	Guided Self-study
	0	0	150

Course Aim/Intended Learning Outcomes:
After completing this module, the students should be able to,

- Examine and evaluate emerging technologies that support digital transformation
- Compare and contrast candidate technologies with respect to a given problem context
- Explain and appraise the likely impacts of emerging technologies on firms as well as the society in general
- Provide a feasible solution to a real-world problem with available and emerging technologies

Course Content:
Students are expected to select a real-world problem and critically examine available technologies to devise a practical solution to the problem. The technologies could be selected from a range of areas including, but not limited to, advanced broadband, cloud computing, virtual/augmented/mixed reality, blockchains, data visualization, internet of things, mobile communication, data mining, big data analytics, social media, robotics, wearables, online education, etc. The students are required to carry out their studies under the guidance of an academic member and, make presentations on their progress at seminar sessions arranged by the Department. A comprehensive report on the solution need to be submitted at the end.

Teaching /Learning Methods:
Guided self-study

Assessment Criteria:
Continuous assessments on the progress of the work and the final report

Continuous Assessment 50%	Final Assessment 50%		
Details: quizzes %, mid-term %, Presentations% : 0 %, 0 % , 50 %	Theory (%) 0%	Practical (%) 0%	Other (%) (Final report) 50%

Recommended Reading:

1. Follett, J. (ed.), Designing for emerging technologies: UX for genomics, robotics, and the Internet of things, O'Reilly Media, 2014
2. George Gilder, Life After Google: The Fall of Big Data and the Rise of the Blockchain Economy, Gateway Editions, 2018
3. Jørgen Randers, 2052: A Global Forecast for the Next Forty Years, Chelsea Green Publishing, 2012
4. Material provided in CAL