L-Università ta' Malta

MATSEC
Examinations Board

AM 19 SYLLABUS

Information Technology

2025

Table of Contents

Introduction 2
List of Subject Foci 2
List of Learning Outcomes 2
Learning Outcomes and Assessment Criteria 3
Scheme of Assessment 48
Coursework 51
Databases and Programming Coursework 51
Marking Criteria: Databases and Programming 57
Web Design Coursework 59
Marking Criteria: Web Design 63

Introduction

The area of Information Technology (IT) is an actual, pervasive, and critical area of study that is impossible to avoid in the context of modern business and other processes. It is important to create a knowledge and skills workforce targeted at supporting and boosting the modern digital economy. This course intends to prepare candidates for applicative business-related and computer information-related courses as well as preparing candidates for work in industry in the information management fields.

The content of the course includes insights into the role and types of information systems in organisations as well as the techniques, technologies, and strategies used in supporting modern organisations and their processes. In this respect, this course aims to equip candidates with an appreciation, understanding, and necessary programming skills, to implement solutions that satisfy industry requirements. Candidates will also be provided with practical hands-on skills through their participation in projects involving programming, data management and web design.

The syllabus for IT at this level does not assume or require any prior knowledge or certification in the field of IT, and the specific intent of this course can be gleaned through the learning outcomes as listed under LO1 to LO4 in this syllabus.

List of Subject Foci

1. Information Systems
2. ICT in Organisations
3. Software
4. Programming Techniques and System Development

List of Learning Outcomes

At the end of the programme, I can:

LO 1. understand the various parts of a computing system in terms of hardware and Information Systems (IS) in organisations and the types of IS.

LO 2. understand the different types of technologies, the systems used in organisations, and the different organisational structures.

LO 3. understand software (system, online and application), and the modelling and use of data.

LO 4. understand programming paradigms and a representative programming language, as well as the phases of software solution development.

Learning Outcomes and Assessment Criteria

Subject Focus:	**Information Systems**
Learning Outcome 1:	**I can understand the various parts of a computer system in terms of hardware and Information Systems (IS) in organisations and the types of IS.**

Topic	Sub-Topic	Assessment Criteria
1.1 Information systems in organisations	1.1.1 Role of information systems	1) Define Information Systems (IS). 2) Outline the main functionality of IS for an organisation. *Limited to IS as a means of extracting the required information by the organisations.* 3) Define the types of IS found in organisations. *Limited to:* *i) Data Processing Systems;* *ii) Knowledge-Based Systems;* *iii) Management Information Systems (MIS).* 4) Compare and contrast the main differences in section 1.1.1 (3). 5) Define what constitutes internal and external information with respect to an organisation. 6) Distinguish between the two types of information listed in section 1.1.1 (5). 7) Define Strategic, Tactical and Operational Information Systems.
	1.1.2 Management information systems	8) Outline MIS to include a listing of their types. *Limited to:* *i) Decision Support Systems (DSS);* *ii) Expert Systems (ES);* *iii) Executive Information Systems (EIS) or Executive Support Systems (ESS).*

Topic	Sub-Topic	Assessment Criteria
		9) List the characteristics of MIS. 10) Describe the role of MIS. *Limited to how managers are supported in their decision-making.* 11) Outline the factors affecting the adoption success or failure of MIS. 12) Define EIS. 13) Define DSS. 14) Define Data Warehousing. 15) Define Online Analytical Processing (OLAP). 16) Define Data Mining. 17) Describe the purpose of "drill-down" in data mining.
	1.1.3 Data capture	18) Define the methods commonly associated with data capture. *Limited to:* *i) Keying in the data;* *ii) Bar codes;* *iii) Magnetic stripe cards;* *iv) Radio Frequency Identifier (RFID) Near Field Communication (NFC);* *v) Optical Mark Recognition (OMR);* *vi) Optical Character Recognition (OCR);* *vii) Quick Response (QR) codes;* *viii) Smart cards.* 19) Compare and contrast the main differences between the data capture methods listed in section 1.1.3 (18). 20) Give usage examples of the data capture methods listed in section 1.1.3 (18).

Topic	Sub-Topic	Assessment Criteria
		21) Describe Electronic Data Interchange (EDI). *Limited to the use of EDI to link suppliers and retailers through a fast electronic information exchange.* 22) Define the Internet of Things (IoT). 23) Give examples of the IoT. *Limited to physical devices that are interconnected and managed by an Internet connected device.* *Example - a lightbulb that can be switched on using a smartphone app is an IoT device, an Air Conditioning unit that is temperature-controlled from a smartphone app is also an IoT device.*
	1.1.4 - Data loss, disaster planning, and recovery methods	24) Define negligence in an IS environment. 25) Discuss the effects of negligence in an IS environment. *Examples should reflect organisations and the impact suffered through data loss due to non-existent or inefficient back-up plans, security breaches, and natural disasters.* 26) List the main stages in disaster planning and the drawing up of a disaster recovery plan. *Limited to:* *i) preparedness;* *ii) response;* *iii) recovery;* *iv) mitigation.* 27) Describe the stages as listed in section 1.1.4 (26). 28) Outline a security plan including physical security and software related security. 29) Define a contingency plan. 30) Outline the criteria used to select a contingency plan.

Topic	Sub-Topic	Assessment Criteria
	1.1.5 Training issues	31) Discuss the need for Information and Communication Technology (ICT) related training in modern organisations. 32) Compare and contrast the training requirements for managerial, technical, and administrative staff. 33) Distinguish between unsupervised computer-based training and instructor-led training. 34) Outline the main components of a corporate training strategy in terms of: i) analysing training needs and goals; ii) identifying skill gaps; iii) prioritising training needs and goals; iv) planning and delivery of training. 35) Outline the main reasons for a corporate training strategy in terms of: i) training for new employees; ii) training in response to technological update/changes; iii) training for skill retention.
1.2 Tools required to handle information	1.2.1 Computer system	1) Define the main hardware components of a computer system. *Limited to:* *i) Central Processing Unit (CPU);* *ii) Main memory;* *iii) Auxiliary storage;* *iv) Input/Output (I/O).* 2) Draw a simple Von Neumann architecture based on the components mentioned in section 1.2.1 (1).

Topic	Sub-Topic	Assessment Criteria
		3) Compare and contrast the main categories of computers currently in existence in terms of: i) relative computational speed; ii) relative data storage capacity; iii) typical I/O devices used. *Limited to:* *i) supercomputers;* *ii) servers (physical);* *iii) desktops (office, gaming, design, media);* *iv) laptops (in terms of mobility and processing power);* *v) smart computing devices, including smartphones, tablets, smartwatches, and embedded systems (e.g. vehicles and household appliances).* 4) Give examples of typical applications of the main categories of computers as listed in section 1.2.1 (3).
	1.2.2 Input devices	5) Outline the function of the main input devices. *Limited to:* *i) mouse;* *ii) trackball;* *iii) keyboard;* *iv) trackpad / touchpad;* *v) touchscreen;* *vi) microphone;* *vii) webcam;* *viii) game controller.*

Topic	Sub-Topic	Assessment Criteria
		6) Compare and contrast input devices listed in section 1.2.2 (5) in terms of suitability of use for a given activity. *No internal structural detail is required. Brand names should not be used.*
	1.2.3 Storage devices	7) Describe primary and secondary storage. 8) Define Random Access Memory (RAM). 9) Define Read Only Memory (ROM). 10) Outline the top-level concepts of magnetic storage technology. *Limited to the hard drive.* 11) Outline the top-level concepts of optical storage technology. *Limited to DVD-ROM and Blu-Ray.* 12) Define Solid State storage technology. 13) Give examples of devices using Solid State storage technology. 14) Compare and contrast the types of storage technologies in sections 1.2.3 (10) to (12). *Limited to advantages and disadvantages, performance, relative costs, durability, and robustness.* 15) Give examples of uses and applications for sections 1.2.3 (10) to (12).
	1.2.4 Output devices	16) Outline the main function of output devices. *Limited to:* *i) monitors;* *ii) built-in/integrated displays;* *iii) touchscreens;* *iv) digital projectors;*

Topic	Sub-Topic	Assessment Criteria
		v) Virtual Reality (VR)/ Augmented Reality (AR) headsets; *vi) speakers/Headphones (wired/wireless);* *vii) printers (inkjet, thermal, laser, 3D printer) and plotters;* *viii) haptic feedback devices (game controllers, wearables).* 17) Compare and contrast output devices listed in section 1.2.4 (16) in terms of suitability of use for a given activity. *No internal structural detail is required. Brand names should not be used.* 18) Define the following terms in relation to visual output devices: i) resolution (in terms of pixel count); ii) display size (measured diagonally); iii) aspect ratio. 19) Define the following terms in relation to printing devices: i) resolution (dots per inch (dpi)); ii) print speed (pages per minute (ppm)).
	1.2.5 Processing modes	20) Define the following processing modes: i) Batch (e.g. Payroll at end of month); ii) Online (e.g. Web Applications); iii) Real-Time (e.g. aircraft autopilot, manufacturing control, etc.). *Limited to systems that are purely one type or another. Systems that exhibit hybrid behaviour i.e. behaviour spanning more than one type of system should not be included.* 21) Give examples of real-world applications for each of the processing modes listed in section 1.2.5 (20). 22) Compare and contrast each of the processing modes listed in section 1.2.5 (20).

Topic	Sub-Topic	Assessment Criteria
		23) Outline the criteria upon which the processing modes listed in section 1.2.5 (20) would be chosen to support real-world processes.
	1.2.6 User Interface (UI) and Human Computer Interaction (HCI) in IS	24) Outline the basic psychological factors that are needed to design good software access. *Limited to: short-term memory (STM) and long-term memory (LTM).* 25) Compare and contrast text versus graphics-based UIs in terms of the psychological factors mentioned in section 1.2.6 (24). 26) Define the following interface technologies: i) Command Line Interface (CLI); ii) Graphical User Interface (GUI); iii) virtual and augmented reality interaction; iv) interaction through voice recognition; v) haptic interaction; vi) virtual surfaces; vii) eye movement tracking; viii) gesture recognition. 27) Compare and contrast the advantages and disadvantages of the interface styles mentioned in section 1.2.6 (26). 28) Give examples where the technologies mentioned in section 1.2.6 (26) (vi-viii) may be used.
	1.2.7 Number base systems	29) Define the following number base systems: i) Binary; ii) Decimal (denary); iii) Hexadecimal.

Topic	Sub-Topic	Assessment Criteria
		30) Convert between the number base systems listed in section 1.2.7 (29). *Limited to unsigned 8-bit binary integer number.*
	1.2.8 Data validation and verification in IS	31) Outline the importance of accuracy and validity of data. 32) Define data validation and data verification. 33) Outline the main uses of validation checks. *Limited to:* *i) presence check;* *ii) format check;* *iii) range check;* *iv) file look-up check;* *v) look-up list check.* 34) Outline the main uses of verification checks. *Limited to:* *i) parity (even and odd);* *ii) checksum;* *iii) proof-reading data;* *iv) double entry.* 35) Outline the common data errors. *Limited to transcription and formatting errors.*
	1.2.9 Networks	36) Define the term network. 37) List the advantages and disadvantages of a network.

Topic	Sub-Topic	Assessment Criteria
		38) Compare the types of networks commonly used. *Limited to top-level understanding of Local Area Network (LAN), Wide Area Network (WAN), and the Internet.* 39) Define the common network topologies. *Limited to Bus, Star, Ring and Mesh topologies.* 40) Compare and contrast the common network topologies (as mentioned in section 1.2.9 (39)). *Limited to layout, and their respective advantages and disadvantages.* 41) Define server-based and peer-to-peer networks. 42) Compare the networks mentioned in section 1.2.9 (41). *Limited to their respective advantages and disadvantages.* 43) Define the functions of modems, routers (including mesh routers), hubs, switches, and network extenders. *Limited to their top-level functions.* 44) Distinguish between the devices listed in section 1.2.9 (43). 45) Compare the use of the following wired media: i) coaxial; ii) twisted pair; iii) power lines; iv) fibre-optic. *Limited to their application and performance. No technical details are required.*

Topic	Sub-Topic	Assessment Criteria
		46) Compare the use of the following wireless technologies: i) Wi-Fi; ii) Bluetooth; iii) Near-field Communication (NFC). *Limited to their application and performance. No technical details are required.* 47) Define cross-talk and spurious noise as examples of interference in data transmission. 48) Define Cloud Computing Technology. 49) Compare and contrast different Cloud-based services. *Limited to PaaS (Platform as a Service), SaaS (Software as a Service), and IaaS (Infrastructure as a Service).*
1.3 Information policy, strategy and systems	1.3.1 Data and information	1) Give examples of the transformational effect of digitisation on information management, from the pre-digital era to the digital economy. 2) Distinguish between the value of data, information and knowledge and how they relate to each other. 3) Define the input-process-output cycle in terms of an information system. *No reference to the input-process-output cycle at hardware level is expected.* 4) Explain the importance of quality of information. *Limited to keeping information up-to-date, accurate and complete.* 5) Discuss the challenges of information management in modern organisations. In terms of information used in organisations. *Limited to the concepts of identification of information needs; acquisition and creation of information; analysis and interpretation of information; organisation and storage of information; information access and dissemination, and information overload.*

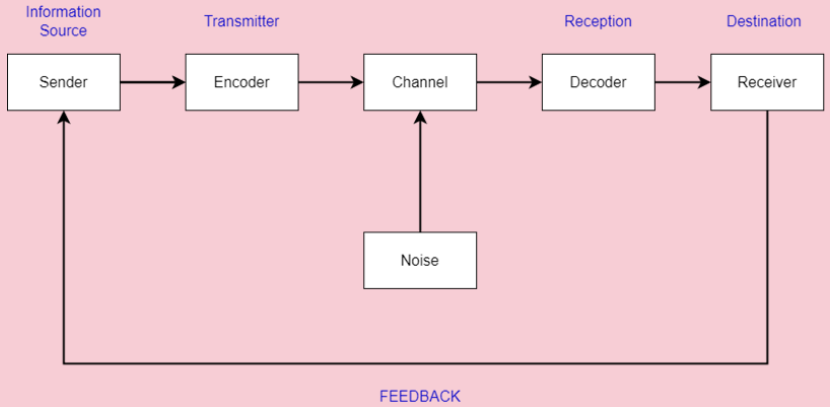
Topic	Sub-Topic	Assessment Criteria
		6) Explain the significance of data and information to modern organisations. *Limited to the information's ability to help take effective decisions for the benefit of organisations.* 7) Outline the concept of 'Garbage in – Garbage out'. 8) Differentiate between direct and indirect sources of data. 9) Compare centralised and decentralised information approaches with respect to managing information across an organisation. *Reference to distributed ledger technologies is not expected.* 10) Discuss the different information needs of different user groups, and how these are addressed through a given solution. *Limited to users, system administrators, and managers.* 11) Outline the concepts behind a knowledge-based society. *Limited to concepts of a society based on data, information and knowledge, and adequately knowledgeable and skilled human resources.* 12) Discuss the specific uses of ICT technologies and tools to facilitate the search of information. *Limited to the use of the Internet as a search tool using:* *i) search engines;* *ii) on-line libraries (digital libraries, electronic journals, and academic resources).*
	1.3.2 Information security policies	13) Outline the following policies related to information security: i) Remote access; ii) Password creation and management; iii) Portable media (USB devices);

Topic	Sub-Topic	Assessment Criteria
		iv) Acceptable use of hardware and Internet access.
	1.3.3 Data protection legislation through compliance with the General Data Protection Regulation (GDPR)	14) Explain the need for Data Protection and its legislation. 15) Define the main aim of the GDPR. 16) Compare and contrast a data subject and data controller. 17) List the roles of the Data Protection Officer. *Limited to a local (Maltese) context.*
	1.3.4 IS strategies within organisations	18) Define an IS strategy. 19) Outline the importance of an IS strategy in an organisation. 20) Outline key aspects of IS strategy in terms of: i) resources (non-human and human); ii) accessibility; iii) communication. 21) Describe the techniques used to achieve IS strategies including: i) Strengths, Weaknesses, Opportunities, and Threats (SWOT) analysis; ii) resource analysis (human and financial); iii) demand analysis (clients and suppliers).
	1.3.5 Backing up data	22) Explain the importance of a backup strategy and the potential risks of not having one. 23) Compare backup types. *Limited to full, differential, and incremental backups.* 24) Compare backup technologies. *Limited to RAID system (levels 0 and 1), Grandfather-father-son (up to three generations),*

Topic	Sub-Topic	Assessment Criteria
		Cloud-based backups.
	1.3.6 Security of data	25) Justify the importance of data integrity. 26) Distinguish between Authentication and Authorisation. 27) Give examples of authentication technologies. *Limited to:* *i) User ID/Password/PIN codes;* *ii) Access cards (using magnetic or Near Field Communication - NFC);* *iii) Biometrics (fingerprint, face, palm print, voice recognition and retina/iris scanning).* 28) Define n-factor authentication. *In terms of section 1.3.6 (27).* 29) Give examples of the use of n-factor authentication. *Limited to 2-factor authentication.* 30) Give examples of authorisation. *Limited to:* *i) physical access control;* *ii) access rights management and permissions - no access, read, write and execute.* 31) Explain the main issues involved in securing data against intentional and unintentional loss. *Limited to:* *i) malicious practices (through use of malware or other illegal practices);* *ii) natural causes (e.g., floods, fires, hardware failure).* 32) Explain the issue of security vs usability.

Topic	Sub-Topic	Assessment Criteria
		With reference to the balancing act between additional security, its impact on users, and the setting of security policies (refer to section 1.3.2). 33) Define Cybercrime. 34) Outline the effects that Cybercrime can have on organisations. 35) Distinguish between the following malpractices: i) hacking; ii) phishing; iii) malware; *Limited to spyware, ransomware, viruses, worms, Trojans and rootkits.* iv) social engineering to manipulate human behaviour; v) physical eavesdropping and break ins; vi) skimming. *Technical explanation and details are not required.*

Subject Focus:	ICT in Organisations
Learning Outcome 2:	I can understand the different types of technologies, the systems used in organisations, and the different organisational structures.

Topic	Sub-Topic	Assessment Criteria
2.1 Forms of communication	2.1.1 Introduction to communication	1) Describe the transmission model of communication.  2) Distinguish between the different types of communication: i) voice; ii) written; iii) visual. *To include advantages and disadvantages of each communication.*
	2.1.2 Voice communication	3) Outline the application of the following voice communication methods: i) fixed-line; ii) mobile; iii) VoIP; iv) satellite links.

Topic	Sub-Topic	Assessment Criteria
		4) Outline the technologies used in the voice communication methods listed in section 2.1.2 (3). *Limited to the medium of communication used.*
	2.1.3 Written communication	5) Outline the application of the following written communication methods: i) word processors; ii) e-mail clients; iii) Portable Document Format (PDF); iv) groupware and workflow systems (e.g., Microsoft Teams, Trello); v) code management systems (e.g., Github). 6) Compare and contrast the methods discussed in section 2.1.3 (5). 7) Give examples of where and why the methods listed in section 2.1.3 (5) are used.
	2.1.4 Visual communication	8) Outline the application of the following visual communication methods: i) digital image processing; ii) scanning; iii) video and presentation platforms; iv) virtual reality; v) augmented reality. 9) Give examples of where and why the methods in section 2.1.4 (8) are used. 10) Define some terms in respect to digital image processing. *Limited to resizing, aspect ratio, compressing images (JPG, GIF, PNG).*
	2.1.5 The internet as a communication medium	11) Distinguish between synchronous and asynchronous media communication media.

Topic	Sub-Topic	Assessment Criteria
		12) Describe the following Internet communication tools. i) e-mail; ii) video and audio (streamed and pre-recorded); iii) chatting; iv) forums; v) collaborative tools; vi) web conferencing; vii) blogging; viii) wikis; ix) micro-blogging; x) social networks. 13) Give examples of how each of the communication tools in section 2.1.5 (12) can be used in the context of a business, both in an internal and an external communications context.
2.2 Role of ICT in organisations	2.2.1 Organisational structure	1) Define an organisation. 2) Distinguish between the different structures that would subdivide an organisation by: i) function; ii) product; iii) project. 3) Differentiate between the three main different organisational charts: i) hierarchical; ii) horizontal; iii) matrix. 4) Draw any organisational chart from the ones mentioned in section 2.2.1 (3).

Topic	Sub-Topic	Assessment Criteria
		5) Compare and contrast the roles of the following top management posts: i) Chief Executive Officer (CEO); ii) Chief Operations Officer (COO); iii) Chief Technical Officer (CTO); iv) Chief Information Officer (CIO); v) Chief Financial Officer (CFO); vi) Chief Security Officer (CSO).
	2.2.2 E-business	6) Define e-Commerce. *To include mobile e-Commerce.* 7) Define e-Business. 8) Differentiate between e-Commerce and e-Business. 9) List the advantages and disadvantages of e-Commerce for businesses and customers. 10) List the advantages and disadvantages of e-Business for businesses and customers. 11) Compare and contrast the following models: i) business-to-business (B2B); ii) business-to-consumer (B2C); iii) consumer-to-consumer (C2C). *In terms of their functionality, their application, and examples of platforms used.* 12) Define e-Marketing. 13) Define e-Markets (e.g., eBay). 14) Define a web portal. 15) Give examples of a web portal.

Topic	Sub-Topic	Assessment Criteria
	2.2.3 E-government	16) Define E-government. 17) Outline of the following e-Services: i) admin to admin; *Limited to Inter-departmental data requests.* ii) admin to business; *Limited to e-Procurement.* iii) admin to citizen (e.g., requesting birth certificate). 18) Distinguish between Informational and Transactional e-Services. *Limited to applications and payments.* 19) Give examples of Informational and Transactional e-Services (e.g., gov.mt). 20) Define an Electronic Identity. 21) List the uses of an Electronic Identity.
	2.2.4 E-Learning tools in organisations	22) Define e-Learning. 23) Compare and contrast synchronous and asynchronous e-Learning. 24) Define Learning Management Systems. *In terms of general structure, functionality, and properties.* 25) Outline the advantages and disadvantages of e-Learning. 26) Justify the application of Massive Open Online Courses (MOOCs) in today's educational environment.
	2.2.5 User support systems	27) Define user support systems. 28) List the aspects of user support.

Topic	Sub-Topic	Assessment Criteria
		Limited to sales and technical. 29) Explain why an organisation and external customers may need user support. 30) Outline the use of a help desk and a help desk software. 31) Describe the various types of user support systems. *Limited to:* *i) forums;* *ii) user manuals;* *iii) live-chat and chat-bots;* *iv) FAQs and knowledge bases;* *v) remote support (e.g., Teamviewer).* 32) Give examples of the user support systems in section 2.2.5 (31). 33) State the purpose of each example of support systems given in section 2.2.5 (31). 34) Compare and contrast the aspects of customer support in section 2.2.5 (31). 35) Compare and contrast the three types of manuals: i) user; ii) technical; iii) operational.
	2.2.6 Customer Relationship Management (CRM)	36) Define CRM. 37) Discuss the benefits of CRM. *In terms of sales, marketing and support to select, acquire, retain, and extend the customer-base.* 38) Define the types of CRMs. *Limited to collaborative, operational, and analytical.* 39) Outline the applications of CRM within an organisation.

Topic	Sub-Topic	Assessment Criteria
		In terms of marketing, sales, support, and customer retention.
	2.2.7 ICT tools in organisations relating to Science and Engineering	40) Define the concepts: i) Computer Aided Design (CAD) and Computer Aided Manufacturing (CAM); ii) experimentation and simulation; *Limited to traffic and piloting (air and sea).* iii) navigation, data tracking systems, telemetry; iv) statistical packages (e.g., SPSS). 41) Give examples of the concepts mentioned in section 2.2.7 (40).
	2.2.8 Artificial Intelligence (AI)	42) Define AI. 43) Give examples of uses of AI. *Limited to:* *i) entertainment;* *ii) financial services;* *iii) communications;* *iv) healthcare;* *v) travel and tourism;* *vi) education;* *vii) transportation networks and vehicles;* *viii) logistics.* 44) Outline advantages and disadvantages of AI in the areas mentioned in section 2.2.8 (43).
	2.2.9 Big Data	45) Define Big Data. 46) Define the three Vs of Big data (Volume, Velocity, and Variety).

Topic	Sub-Topic	Assessment Criteria
		47) Give examples of Big Data in the following areas: i) logistics; ii) retail; iii) healthcare; iv) telecommunications; v) financial services. 48) Outline advantages and disadvantages of the use of Big Data as outlined in 2.2.9 (47).
	2.2.10 Distributed Ledger Technology (DLT)	49) Define DLT. 50) Outline the uses of DLT. *Limited to cryptocurrencies and blockchain.* 51) Give examples of the usage of cryptocurrencies and blockchain technologies.
	2.2.11 Sustainable technologies in ICT	52) Define sustainability in terms of ICT. 53) Outline the following domains in terms of ICT sustainability: i) storage techniques; ii) on-demand computing; iii) Internet of Things (IoT); iv) Data Analytics. 54) Give examples of systems under the domains listed in section 2.2.11 (53). 55) Outline the concepts behind the use of automation to leverage sustainability. *Limited to:* *i) manufacturing;* *ii) transport;* *iii) public administration;* *iv) domestic smartness.*

Topic	Sub-Topic	Assessment Criteria
	2.2.12 Health and safety	56) List the advantages and disadvantages of automation. 57) Explain the following concepts with reference to Health and Safety: i) computer usage stress and digital detox; ii) Repetitive Strain Injury (RSI); iii) eyestrain (font and icon sizes, themes, dark and light modes). 58) Give examples of health and safety issues relating to an ICT work environment. *Limited to the following aspects:* *i) monitor positioning;* *ii) seating conditions;* *iii) ergonomic keyboards and mice;* *iv) size of monitor;* *v) ambient lighting;* *vi) climatic conditions;* *vii) ownership of workspace by employees;* *viii) taking regular breaks from the working area.*

Subject Focus:	Software
Learning Outcome 3:	I can understand software (system, online, and application), and the use and modelling of data.

Topic	Sub-Topic	Assessment Criteria
3.1 Aspects and categories of software	3.1.1 Software	1) Define software. 2) Distinguish between open source, free, and proprietary software in terms of software usage and licensing. *Licencing limited to group and individual licencing.* 3) Define the notion of Intellectual Property and the right of owning it.
	3.1.2 System software	4) Define system software. 5) Define the following types of system software: i) operating system; ii) utility applications; *Limited to antivirus, archivers (compression), defragmenters, computing resource monitors, and file managers.* iii) translators; *Limited to compilers, interpreters, and assemblers.* iv) Monitoring software for both hardware and software. 6) Recommend a suitable utility application from the list in section 3.1.2 (5) (ii), for a given situation. 7) Give examples of situations where utility applications can be of benefit to a computer system's maintenance and performance.
	3.1.3 Application software	8) Define application software.

Topic	Sub-Topic	Assessment Criteria
		9) Define the types of application software. *Limited to:* *i) specific;* *ii) general purpose (generic);* *iii) application generators;* *iv) software suites;* *v) online application environments.* 10) Compare and contrast the types of application software listed in section 3.1.3 (9). 11) Give examples of situations where the types of application software listed in section 3.1.3 (9), may be applied.
	3.1.4 Operating System (OS)	12) Define an OS. 13) Define the components of an OS. *Limited to:* *i) kernel (supervisor or control program);* *ii) memory manager;* *iii) input/output manager;* *iv) file system manager;* *v) backing store manager;* *vi) resource allocation and process scheduler;* *vii) accounting (Event logging);* *viii) error handling;* *ix) security.*

Topic	Sub-Topic	Assessment Criteria
		14) Define the types of OS. *Limited to:* *i) batch;* *ii) on-line (e.g. cloud based OSs);* *iii) real-time;* *iv) mobile;* *v) network (distributed).* 15) Give examples of situations where the types of OS listed in section 3.1.4 (14), may be deployed.
	3.1.5 Software properties	16) Define software portability. 17) Justify the need for software portability. 18) Give examples of portable software applications. 19) Define downward compatibility. 20) Justify the need for downward compatibility. 21) Give examples of downward compatibility. 22) Compare and contrast Object Linking and Embedding (OLE) within the context of objects within documents. 23) Define software reliability. *Limited to Mean Time To Failure (MTTF) and Mean Time Between Failure (MTBF).*

Topic	Sub-Topic	Assessment Criteria
	3.1.6 Software evaluation	24) Define software evaluation criteria, which an organisation or an individual might consider when selecting a software solution. *Limited to:* *i) hardware needs;* *ii) quality of documentation;* *iii) compatibility with existing software (interoperability);* *iv) ease of use of the software;* *v) technical support;* *vi) cost;* *vii) compliance in regulated sectors;* *viii) provider reputation.*
3.2 Database systems	3.2.1 Introduction	1) Define a file. 2) Define the four types of file access. *Limited to serial, sequential, direct, and indexed-sequential files.* 3) Define a database. 4) Describe the importance of databases. *In terms of usability and data organisation.* 5) Outline the concept of the Three-level database architecture (External, Conceptual, and Internal/Physical). 6) Define a flat-file database structure. 7) Describe the issues with a flat-file database structure. *Limited to data isolation, data duplication (redundancy), and program/data dependence.* 8) Define a relational database.

Topic	Sub-Topic	Assessment Criteria
		9) Compare and contrast relational and flat-file database systems. *Limited to data consistency and integrity, data redundancy, design (structural) complexity, and cost.*
	3.2.2 Centralised and distributed database systems	10) Define centralised and distributed database systems. 11) Draw representations, using simple block diagrams (using blocks and lines only), of the centralised and distributed database structures. 12) Compare and contrast centralised to distributed databases. 13) Justify the usage of a centralised or distributed databases, for a given scenario.
	3.2.3 Relational data model	14) Define the following database terms: i) table (relation); ii) entity; iii) field; iv) key field (primary key); v) composite key; vi) foreign key; vii) secondary key; viii) record (tuple); ix) links (relationships). 15) Describe a given set of tables using the database notation as listed hereunder: i) the name of each table (relation) is followed by a list of all the fields in brackets; ii) key fields are underlined; iii) foreign keys are in italic if printed or over-line if hand-written; iv) entity names in upper case whereas field names are in lower case. 16) Identify the best data types to represent data attributes for a given scenario. *Limited to text, numeric, Boolean, and date.*

Topic	Sub-Topic	Assessment Criteria
	3.2.4 DBMS	17) Define a Database Management System (DBMS). 18) Outline the difference between a database and a DBMS. 19) Define a Data Dictionary. 20) Compare and contrast Data Description Language (DDL) and Data Manipulation Language (DML). 21) Define database views. 22) Outline the role of the Database Administrator (DBA). 23) Outline the role of Access Rights in terms of database security.
	3.2.5 Entity Relationship Diagrams (ERD) *Crow's Foot notation is to be used.*	24) Outline the use of an Entity Relationship Diagram (ERD). 25) Define cardinality. 26) Define the three cardinalities: i) one-to-one; ii) one-to-many; iii) many-to-many. 27) Give examples of the cardinalities listed in section 3.2.5 (26). 28) Interpret an ERD. *Limited to a maximum of four data entities.* 29) Construct an ERD for a given scenario. *Limited to a maximum of four data entities.*
	3.2.6 Normalisation	30) Define normalisation. 31) Explain the need for normalisation.

Topic	Sub-Topic	Assessment Criteria
	3.2.7 Structured query language	32) Define the 1st, 2nd and 3rd normal forms. 33) Identify normalisation issues for a given scenario. 34) Recommend a normalised solution for a given scenario. *Limited up to the 3rd normal form.* 35) Construct SQL statements from given scenarios. *All queries are to be limited to one table only.* *Queries are to be limited to:* *i) SELECT… FROM...;* *ii) SELECT… FROM…WHERE;* *iii) SELECT… FROM…WHERE…AND;* *iv) SELECT… FROM…WHERE…ORDER BY… (using ascending or descending order).* 36) Interpret the outcome of DDL statements. *Limited to CREATE and DROP.* 37) Interpret the outcome of DML statements. *Limited to INSERT, UPDATE, and DELETE.*
3.3 Internet technologies and applications	3.3.1 Introduction to the Internet	1) Define Domain Names and Internet Protocol (IP) addresses. 2) Outline the function of the Domain Name System in terms of structure, meaning, and translation to IP addresses. 3) Define Internet registries in terms of registration of domain names and IPs on a global level. 4) Define Internet Service Provider (ISP). 5) Compare and contrast Intranets and Extranets.

Topic	Sub-Topic	Assessment Criteria
	3.3.2 Basics of HTML and Cascading Style Sheets (CSS)	6) Define IPv4 and IPv6. *Limited to address space capabilities and number system used.* 7) Define HTML. 8) Explain the use of the following HTML tags: i) <html> ii) <body> iii) <head> iv) <style> v) <title> vi) <p> vii) <h1>, <h2>, <h3> viii) <a> (internal/external) ix) Lists and x) <table> xi) <img> xii) <nav> xiii) <footer> xiv) <header> xv) <main> xvi) <figure> xvii) xviii) <a> (anchors/mailto) xix) comments in HTML <!-- … --> xx) <meta> 9) Correct erroneous use of the HTML tags as listed in section 3.3.2 (8) i) to xi).

Topic	Sub-Topic	Assessment Criteria
		10) Write HTML code snippets. *Limited to not more than 15 lines of HTML code using tags in their default mode only and limited to those listed in section 3.3.2 (8) i) to xi).* 11) Define CSS. 12) Justify the need for external CSS files rather than embedding CSS within HTML. 13) Outline the use of *class* and *ID* in CSS. 14) Give examples of basic CSS uses. *Limited to the following properties:* *i) color (limited to predefined colour names for plain text);* *ii) text-align (limited to left, right, centre and justify);* *iii) font-family (limited to exercising the knowledge of candidates in setting types of font faces);* *iv) text-decoration (limited to underline, overline and line-through);* *v) font-weight (limited to bold or normal);* *vi) font-style (limited to italic and normal);* *vii) font-size (limited to pixels).* *Limited to formatting using the HTML tags in section 3.3.2 (8) (vi), (vii) and (ix) and the properties listed under section 3.3.2 (14).* 15) Correct erroneous use of the CSS selectors as listed in section 3.3.2 (14). 16) Define form validation. *Limited to the context of online forms and client-side validation.* 17) Justify the use of different types of validation on an online form. *Limited to range, presence, format, length, and data type.* 18) Define sitemaps. 19) Outline the use of sitemaps.

Topic	Sub-Topic	Assessment Criteria
	3.3.3 World Wide Web (WWW)	20) Define the following: i) WWW; ii) website; iii) web browser. 21) Outline the use of a web server. 22) Outline the features of a web browser. *Limited to the search/address bar, bookmarks, and browsing history.* 23) Explain the use of a Uniform Resource Locator (URL). 24) Identify the various parts of a URL. *Limited to protocol, domain, top level domain (TLD), and file path.*
	3.3.4 Internet protocols	25) Define the following protocols: i) TCP/IP; ii) SMTP; iii) POP and IMAP; iv) FTP and SFTP; v) HTTP and HTTPS.
	3.3.5 Internet client applications	26) Define the following applications: i) email client; ii) file sharing client; iii) video conferencing applications. 27) Outline the advantages and disadvantages of the applications listed in section 3.3.5 (26).

Topic	Sub-Topic	Assessment Criteria
	3.3.6 Internet security problems	28) Define the following types of Internet-based fraud: i) phishing; ii) password hacking; iii) brute force attacks; iv) DoS attacks; v) social engineering. 29) Define the following types of malware: i) adware; ii) spyware; iii) worms; iv) trojans; v) viruses; vi) keyloggers; vii) rootkit; viii) ransomware. 30) Define the following countermeasures to the problems listed in section 3.3.6 (29). i) encryption (private and public); ii) firewalls; iii) antivirus software; iv) digital signatures and certificates.

Subject Focus:	Programming Techniques and System Development
Learning Outcome 4:	I can understand programming paradigms and a representative programming language, and the phases of software solution development.

Topic	Sub-Topic	Assessment Criteria
4.1 Programming techniques	4.1.1 Programming languages	1) Define natural and programming languages. 2) Compare and contrast natural languages with programming languages. 3) Distinguish between the following programming language levels: i) low-level (Assembly); ii) high-level (to mean any language from levels 3 and 4). 4) Outline the purpose of language translators. 5) Define compilers, interpreters and assemblers.
	4.1.2 Programming paradigms	6) Define the main programming paradigms: i) imperative; ii) functional; iii) declarative; iv) object-oriented. *Limited to classes/objects, inheritance.* 7) Give examples of real-world processes to which the programming paradigms listed in section 4.1.2 (6) can be applied.
	4.1.3 Introduction to basic algorithms	8) Define an algorithm. 9) Draw a flowchart from given Python code or pseudocode. 10) Construct pseudocode statements from given Python code or from a flowchart. 11) Write simple algorithms in pseudocode.

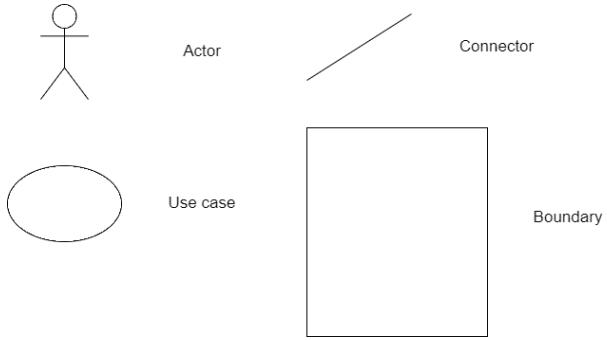
Topic	Sub-Topic	Assessment Criteria
	4.1.4 Foundations of programming in Python	*Limited to search algorithms (linear and binary), the bubble sort, and validation check algorithms (range and data types).* 12) Justify the use of correct indentation and comments (inline/block). 13) Explain the use of the following concepts: i) variable; *To include naming and scope of variable.* ii) basic data types; *Limited to Integer, Float, String, Boolean.* iii) type casting; *Limited to 'int()', 'float()', 'str()'.* iv) data collections; *Limited to Lists, Tuples and Dictionaries.* v) control structures; *To include selection and iteration.* vi) functions; vii) modules.

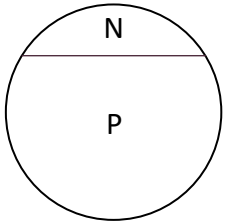
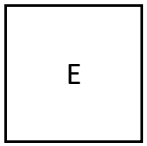
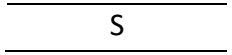
Topic	Sub-Topic	Assessment Criteria
		14) Write code snippets (not to exceed 10 lines of code) using the following constructs, and the concepts listed in section 4.1.4 (13) (i) to (vii): i) variable initialisation; *Limited to Integer, Float, String, Boolean.* ii) type casting; *Limited to 'int()', 'float()', 'str()'.* iii) program i/o; *To include input() and print().* *Limited to new line, end= and sep=.* iv) lists; *Limited to the following actions: list creation, list item access, change list item, list traversal, join lists.* *Using the following in-built functions: len(), append(), clear(), count(), insert(), pop(), remove(), extend().* v) operators; *To include:* *(a) assignment;* *Limited to '=', '+=', '-=', '*=', '/=' and '%='* *(b) arithmetic;* *Limited to '+', '-', '*', '/' and '%'* *(c) comparison;* *Limited to '==', '!=', '>', '<', '<=', '>='* *(d) logical.* *Limited to 'and', 'or', 'not'* vi) control statements; *Limited to 'if', 'elif', 'else'* vii) loops. *Limited to 'for-in', 'while', 'while-else', use of 'break' and 'continue', nested loops (limited to one level).*

Topic	Sub-Topic	Assessment Criteria
		15) Write code snippets using the following concepts: i) tuples; *Limited to the following actions: tuple creation, tuple item access, change tuple item, tuple traversal, and join tuples.* *Using the following in-built functions: map(), append(), clear(), count(), insert(), pop(), remove().* ii) sets; *Limited to the following actions: creation of a set, access set element, add a set element, remove a set element, set traversal, update sets, and distinguish between set elements.* *Using the following in-built functions: add(), clear(), copy(), discard(), difference(), update(), union(), pop(), remove().* iii) dictionaries; *Limited to the following actions: create a dictionary, access dictionary items, remove dictionary items, traversing a dictionary, copy a dictionary.* *Using the following in-built functions: clear(), items(), keys(), get(), update(), pop().* iv) DBMS connection; *Limited to the connection of a MySQL database using the 'MySQL Connector Python' module. Such methods include connect(), cursor(), execute(), close().* v) Tkinter GUI objects. *Limited to the development of a GUI interface using the Tkinter ready-made module to create a window, button, textfield, textarea, label, and proper use of the grid or place layout managers.* *Assessment criterion 4.1.4 (15) should be considered within the context of the coursework only.*

Topic	Sub-Topic	Assessment Criteria
		16) Interpret the blackbox use of: i) functions; *Examples to include in-built functions (limited to the ones listed in section 4.1.4 (14)), and custom ones.* ii) modules; *Limited to import syntax.* iii) File Handling *Limited to 'open()', 'read()', 'write(), 'close()'.* *Limited to a maximum code size of 20 lines.*
	4.1.5 Error types and debugging	17) Compare and contrast the different program error types. *Limited to syntax, logical and run-time errors.* 18) Identify the program error type in given Python code. *Limited to Python concepts listed in sections 4.1.4 (13) and 4.1.4 (14).* 19) Correct errors in given Python code. *Limited to Python concepts listed in sections 4.1.4 (13) and 4.1.4 (14).* 20) Write code snippets to give examples of the different program error types as per section 4.1.5 (17). 21) Outline the use of debugging tools. *Limited to breakpoints, inspecting variables, and code stepping.*
4.2 System development	4.2.1 Stages of system development	1) Define a computer-based system in terms of information handling. 2) Give examples of computer-based system types. *Limited to transactional-based systems, real-time systems, and embedded systems.*

Topic	Sub-Topic	Assessment Criteria
		3) Define the main stages of development of computer-based systems. *Limited to:* *i) problem definition;* *ii) feasibility study;* *iii) requirements elicitation;* *iv) analysis and design;* *v) testing;* *vi) implementation;* *vii) maintenance;* *viii) retirement.*
	4.2.2 Problem definition	4) List the activities involved in the problem definition stage. *In terms of the objectives of the investigation, the resources available, the time scale involved, and any constraints.* 5) List possible reasons why an organisation decides to develop a new system. *Limited to:* *i) the current system being no longer suitable for its original purpose;* *ii) the current system being too inflexible or expensive to maintain;* *iii) competitive advantage required.*
	4.2.3 Feasibility study	6) Explain the purpose of a feasibility study. 7) Outline the different feasibility aspects that a system analyst would need to consider. *Limited to technical, operational, timeliness (scheduling), economic, legal, and social feasibility.* 8) Justify the feasibility aspects to consider for a given scenario.

Topic	Sub-Topic	Assessment Criteria
	4.2.4 Requirements elicitation	9) Discuss the methods of how a system analyst is able to gather data about the current system. *Limited to interviews, questionnaires, inspection of documents, observation of the existing system and the work processes it supports.* 10) List advantages and disadvantages for the methods mentioned in section 4.2.4 (9). 11) Outline reasons why one would consider the use of "off-the-shelf" solutions and bespoke development. 12) Define a Use-Case Diagram (UCD). 13) Illustrate the difference between the symbols used in UCDs. *Limited to the following:*  14) Explain how UCDs help to describe and determine system requirements. 15) Construct a UCD with no more than 5 Use-Cases. *All diagrams are to exclude generalisation and specialisation notation as well as stereotyped relationships.* 16) Interpret a given UCD of not more than 5 Use-Cases. *All diagrams are to exclude generalisation and specialisation notation as well as stereotyped relationships.*

Topic	Sub-Topic	Assessment Criteria
	4.2.5 System analysis and design – Data flow modelling	*The De Marco notation hereunder is to be used throughout section 4.2.5* **Data Process:**  "N" is the process number "P" is the process name **External Entity:**  "E" is the entity name **Data Flow:** "D" is the data flow name **Data Store:**  "S" is the data store name 17) Define data flow modelling. *Limited to the concepts of data flow, data transformation, and external entities.* 18) Outline the use of Data Flow Diagrams (DFDs) in data flow modelling. 19) Illustrate the difference between the symbols used in DFDs. 20) Construct a DFD. *Limited to a Context diagram (level 0) and/or a level 1 diagram of not more than 5 processes.* 21) Interpret a given DFD. *Limited to Context (level 0) and/or level 1 diagrams.* 22) Distinguish between DFDs and UCDs. *In terms of the meaning of the notation and the application of the model.*

Topic	Sub-Topic	Assessment Criteria
	4.2.6 System analysis and design – Data modelling	23) Compare and contrast the top-down and bottom-up approaches to system design. 24) Define a module and modular design. 25) List the advantages and disadvantages of modular design. 26) Outline the main design aspects of a new system. *Limited to:* *i) the design of a user interface;* *In terms of layout, usability, and menu style and structure.* *ii) the design of data inputs and outputs;* *In terms of form and presentation.* *iii) the importance of a conversion plan (content of plan itself not necessary);* *iv) the importance of a testing.* *In terms of strategy, test plan, and test data.* 27) Distinguish between the following diagrammatic algorithm representation forms. *Limited to Flowcharts and DFDs.* 28) Distinguish between the textual algorithm representation forms. *Limited to structured English and pseudo-code.*
	4.2.7 Coding and testing	29) Define the different types of testing approaches: i) module testing; ii) integration testing; iii) system testing; iv) acceptance testing. 30) Compare and contrast black box testing and white box testing. 31) Compare and contrast alpha testing and beta testing.

Topic	Sub-Topic	Assessment Criteria
		32) Identify which testing technique (as listed in in section 4.2.7 (29)) is the most appropriate given a system's stage of development.
	4.2.8 Implementation	33) Outline the tasks that need to be completed before the implementation stage is deemed complete. *Limited to installing and/or configuring any hardware, and staff training.* 34) Compare and contrast the different types of changeover techniques: direct, parallel, phased, and pilot. 35) Justify the selection of an adequate changeover technique for a given scenario. 36) Compare and contrast the different types of software solution documentation: technical, and user.
	4.2.9 Maintenance and retirement	37) Compare and contrast the different types of maintenance: adaptive, corrective, perfective, and predictive. 38) Identify the type of maintenance (as listed in section 4.2.9 (37)) for specific situations. 39) Discuss the factors that lead to retirement. *Limited to maintenance costs, technology shift, and changes in business processes.*
	4.2.10 Project management	40) Define project management. *In terms of the impact it may have on the quality of the solution.* 41) Outline the aspects of project management. *Limited to time and resource management.* 42) Outline how development progress is measured and any corrective action which may be required to keep development on track. 43) List the characteristics of an efficient project team. *Limited to any three characteristics.* 44) List the characteristics of an effective project manager. *Limited to any three characteristics.*

Scheme of Assessment

AM Information Technology is assessed by means of three components:

i) Paper I - 3 hour written paper assessing Subject foci 1 and 2;

ii) Paper II - 3 hour written paper assessing Subject foci 3 and 4;

iii) Coursework - 2 assignments.

The following table shows the percentage weighting of each component and subsequent sections.

Component	Section	Module		% Weighting
Paper I (3 hours)	Section A	Subject Focus 1:	Information Systems	20
	Section B	Subject Focus 2:	ICT in Organisations	15
Paper II (3 hours)	Section A	Subject Focus 3:	Software	15
	Section B	Subject Focus 4:	Programming Techniques and System Development	20
Coursework	Assignment 1: Web Design			8
	Assignment 2: Programming + Database			22

The examination consists of two written papers, Paper I and Paper II and two assignments. Each of the written papers is of three hours duration.

Paper 1

- a written paper of 3 hours duration, out of 100, which carries 35% of the total score;
- candidates will write their answers on a separate booklet provided by MATSEC;
- consists of two sections:

Section A: Information Systems

- Questions are set on the syllabus content of Module 1 with a maximum mark of 60.
- Four questions are set:
  - one compulsory question, consisting of short questions and carrying 20 marks;
  - three questions, to choose two, each carrying 20 marks.

Section B: Human Communication and Business Organisations

- Questions are set on the syllabus content of Module 2 with a maximum mark of 40.
- Four questions are set:
  - one compulsory question, consisting of short questions and carrying 20 marks;
  - three questions, to choose two, each carrying 10 marks.

Paper 2

- a written paper of 3 hours duration, out of 100, which carries 35% of the total score;
- candidates will write their answers on a separate booklet provided by MATSEC;
- consists of two sections:

Section A: Software

- Questions are set on the syllabus content of Module 3 with a maximum mark of 45.
- Four questions are set:
  - one compulsory question, consisting of short questions and carrying 15 marks;
  - three questions, to choose two, each carrying 15 marks.

Section B: Programming Techniques and System Development

- Questions are set on the syllabus content of Subject Focus 4 with a maximum mark of 55.
- Four questions are set:
  - compulsory question on Python + Database and Web Design (coursework) carrying 25 marks. This question will be sub-divided into three sections:
    - 15 Marks for Python assignment related questions;
      - code snippet of 15-20 lines (inline numbering for each line of code);
      - questions of a simple nature, such as the following:
        - describe the algorithm between lines x and y;
        - give line numbers where functions are declared;
        - identify mistakes in lines x to y;
        - find missing parts in lines x to y;
        - fill in missing parts;
        - fix the errors in lines x to y;
        - the output in line x;
        - the purpose of the code in lines x to y.
    - 5 Marks for Database assignment related questions;
      - SQL snippets or Entity Relationship Diagram (ERD);
      - questions of a simple nature, such as the following:
        - what is the expected output of the code in lines x to y?
        - find mistakes in code;
        - fill in the missing code;
        - from the ERD given, write an SQL statement to output the following;
        - underline Primary Keys;
        - draw an ERD from a given scenario.
    - 5 Marks for Web Design assignment related questions.
      - HTML or CSS snippets (as indicated in syllabus);
      - questions of a simple nature, such as the following:
        - describe what tags on lines x, y and z mean;
        - turn text on line x as an ordered list;
        - make text on lines y a link pointing to URL u;
        - make text on line x a main header.
  - three questions, to choose two, each carrying 15 marks.

Coursework

- The coursework is assessed through **TWO** assignments carried out by the candidate during the course of study, monitored and assessed by the tutor and moderated by the Markers' panel. All marks are to be submitted to MATSEC by not later than the date stipulated by the MATSEC support unit.

- **TWO** compulsory assignments will be set during the course. Assignment 1 is related to Web Design and has a weighting of 8%. Assignment 2 is related to Programming and Database and has a weighting of 22%. Candidates must submit and meet the criteria for a pass, as set by the examiners in both assignments to be able to get a grade between A and E.

- It is up to the tutor to schedule the completion date of each assignment.

- All candidates may be called for an interview on their coursework.

- Candidates may re-submit both assignments as specified in the MATSEC Examination Regulations.

- An authentication form is required and is available for download from the MATSEC website.

Note for Private candidates:

- Private Candidates are to submit both projects to the MATSEC Support Unit for assessment by the Markers Panel, by the date stipulated by the unit. Candidates may be called for an interview about their work.

- An authentication form is required and is available for download from the MATSEC website.

Grading

- The final grade will be based on an overall aggregate score.

- The candidates have to meet the criteria for a pass, as set by the examiners in all components of the assessment (Paper I, Paper II, Coursework), to qualify for Grade A to E.

Re-sit

- Candidates who fail to meet the criteria for a pass will have to re-sit the entire examination. Coursework marks may be carried forward for subsequent sessions based on this syllabus.

Coursework

Guidelines

Problems chosen by candidates should be realistic and reasonable in the sense that the objectives planned may be implemented in the time-frame available. Candidates should be encouraged to use different sources of information – textbooks, Internet, other scientific and peer-reviewed resources.

In marking the assignments, credit will be given to the inclusion of the appropriate features as described in the following sections. All the required templates for coursework are provided within this document (including front-page and document structure).

Software Requirements

For the sake of the documentation, it is recommended that candidates should become conversant with the nature and capabilities of the most common generic reporting and presenting software as required to successfully describe and explain one's work and reasoning.

Thus, the documentation should be presented as a word-processed document that includes all the presentation and layout features as listed in the marking scheme below. Furthermore, the document is to be sectioned and titled as indicated in the marking scheme for readability purposes.

Databases and Programming Coursework

Databases and Programming Coursework	
200 marks	This coursework aims at introducing the candidate to the notion of front-end and back-end development. In other words, the practical aspects of implementing a database system whose inner complexities are then abstracted from the user through a GUI (Graphical User Interface). Insertion, deletion, editing, and querying of data needs to be done through the Python GUI and not through the DBMS. **Software Requirements** In the case of the Programming part of the assignment, the language to be used is Python. It is recommended that the candidate become conversant with any IDE (Integrated Development Environment) that would be set to manage and implement the required implementation. The choice of an IDE is ultimately left to the discretion of the individual educational entity, however, for the sake of guidance, the following IDEs can be indicated: • PyCharm; • IDLE; • Thonny. The choice of a DBMS (Database Management System) is ultimately left to the discretion of the individual educational entity, however, for the sake of guidance, MySQL Workbench can be indicated. If a candidate opts for a different DBMS, the candidate should check beforehand that Python supports the creation of a connection with the chosen DBMS.

Section	Details
System Analysis and Problem Formulation	**Problem Definition + Background of the problem** The assignment should be thought of as a solution within a context of the candidate's choice, for example, recording sales in a car dealership or keeping track of a football club's nursery applications. Therefore, in this section, the candidate is expected to provide a description of the context within which the system is to be built. This should include a description of any relevant data management problems currently faced by the client which can be solved by a computerised solution. **Scope** The candidate is expected to briefly explain the reasons behind the project.
Client Requirements	**Use-Case Diagram (UCD)** A UCD should be drawn up using UML notation as described in section 4.2.4 (13). The listed minimum in the marking sheet is to be taken as the obligatory required functionality/features of the solution. This does not preclude additional functionality/features being arbitrarily added to the solution as the tackled scenario requires. Relations between Use Cases are not expected to go beyond simple associations unless the candidate wishes to do so. **Database requirements** The scope of this section is to ensure that candidates outline a top-level description of the requirements. This should include a list of the entities and their data attributes. • For example, a candidate could describe a shop system in the following terms: the system keeps track of the customers that purchase the product and which product is supplied by which supplier. Therefore, Customers, Purchases, Products, and Suppliers would be identified as the entities to be designed and developed. • For example, the Customer entity would include name, surname, address, email and mobile number as the required data attributes. **Programming requirements** By functions one should understand the internal functions that the system is to provide to satisfy the features as illustrated in the Use-Case Diagram. • An example of search and sort parameters could be a facility to search for all the sales records pertaining to the previous week and then returning those records sorted by customer name in ascending order. • Description of each report type and structure - reports should be the result of the queries which will be tackled in the implementation part (e.g., description of the report that will list all the Job Orders for the preceding month). **General description of data validation requirements** In this section candidates are required to write a brief overview of how the different data items in their chosen scenario will be validated. Note that validation needs to be handled from the programming part only.
Database Design	**Database Entity Relationship Diagram (ERD)** The ERD should follow the Crow's Foot notation as listed in section 3.2.5, with a minimum of four entities, their relationships (including cardinality), and a clear indication of primary and foreign keys. **Database normalisation** The database should be normalized up to, and including, the third normal form. Candidates need to describe all three normal forms within the context of the assignment chosen. **Description and justification of field types** For each table, candidates should present a detailed description of every field, clearly indicating primary and foreign keys. The data types used must include varchar, int and Date. Refer to the table below as an example:

Customers Table:

Field name	Data type	Description	Default value
Customer ID (Primary key)	Varchar	The Maltese ID card number system is to be used for this field. Input mask or validation rule to be created.	-------(X)
Name	Varchar	Customer's name.	
Surname	Varchar	Customer's surname. This is separate from the name to allow searching and sorting on surname as per client's requirements.	
Contact Number	Int	Only the mobile phone number is to be collected here. Input mask or validation rule to be created.	(+356)00000000
Date of Birth	Date	The DOB of the customer.	
… etc …			

Programming Design

Program design shown through HIPO chart(s)

This is to be understood as the architectural (i.e., top-level) view of the functionality of the solution as a whole. This is not to be confused with any windows/forms design layout models that will also be used. In practice, this implies a minimum of one HIPO chart which depicts the top-level inputs, processes and outputs of the Python part of the assignment. It is suggested that the candidate look at the UCD created beforehand to extract from it the functionalities to be represented in the HIPO chart and subsequently implemented.

GUI prototype

In this part candidates will be required to produce an outline of every window or form making sure to point out properly the layout of buttons and other objects (e.g., text fields, text areas). These windows/forms should be produced in wireframe form and can be drawn either using pen and paper or through the use of any specific computer-based drawing tool.

Description of data input validation

It is suggested that these descriptions are presented in tabular form, however other forms of presentation such as free text, are also acceptable.
Example:

Field to be validated	Description of validation
Customer's ID	When the user tries to submit the customer's details, this field will be checked for a null value, in which case the submission is refused.
Customer Contact number	This field will be checked to make sure that only numeric characters are submitted to the database.
Product Price	Only prices >0 will be allowed in this field.
... etc...	

Implementation	The candidate is only required to describe one instance of each of the points listed under this section as applicable. It is obligatory that snippets of code or relevant screenshots accompany every description to highlight the use of each feature in the assignment. **Database tables and their relationship** The candidate should briefly describe the steps taken to create one database table using the DBMS. The candidate should also describe how the DBMS was used to create at least one of the relationships. **Type casting** The candidate should provide an example of the use of type casting in the assignment. For example, a String received from the DBMS is cast to an integer in the programming part to make a calculation. **The connection of Python with the underlying database** The connection should be made using any readily available Python module which allows successful connection to the DBMS. For guidance purposes, the following code can be considered. `mydb = mysql.connector.connect(host="localhost", user="chris", passwd="1234", database="person")mycursor = mydb.cursor()` **The separation of the database manipulation code from the interface code** The candidate should strive to separate the code which deals with the GUI aspects of the program from the code which sends and receives the data to the database itself. One could place the functions which manage records in a separate file from the functions which display the GUI forms. Then the 'database handling' file can be linked in the same way as any other module in Python. **Exception handling** At least one instance of exception handling through a 'try and except' block must be briefly described. **Construction of functions with parameters and return values** The candidate should select any function with return values from the assignment and through the selected function describe the concept of input parameters, processing of such parameters to produce the required result, and the returning of the result to the caller function. The idea is to show an appreciation for the use of functional usage in Python where each defined function is a self-contained entity. **Use of queries** The candidate is to provide one example from the assignment of each of the listed commands, as per the marking sheet, describing briefly how the desired result was achieved through their use. These queries must be implemented through the Python code and not from the DBMS. **Usage of lists/tuples/sets/dictionaries** The candidate can opt to make use of any one or more of the above data structures. At least one example is to be chosen and described in the documentation in terms of how it was created and what its use is in general and specifically within the scenario of the assignment. **Use of loops** The candidate should briefly describe any one loop as used in the assignment in terms of how it was created and what its use is in general and specifically within the scenario of the assignment. **Use of conditional statements** The candidate should briefly describe any one conditional statement as used in the assignment in terms of how it was created and what its use is in general and specifically within the scenario of the assignment. **Tkinter GUI objects** The candidate should provide brief descriptions outlining how to create and effectively use each of the GUI objects listed in the marking sheet. Special attention should be given to the use of layout managers as a tool to help create organised user interfaces. **Data input validations** The candidate should briefly describe how range, data type and presence checks were implemented and used in the assignment.

Testing

Requirements Testing

This type of testing should be seen as a simple checklist whereby it is ensured that all the client requirements, are implemented in the assignment. A guiding example is reproduced in this table.

Requirement	Completed?
Employee details properly stored.	Y
Car details properly stored.	Y
Form to capture all Employee details.	Y
... etc ...	

Blackbox testing

Blackbox tests are to be carried out on every validation check created to make sure that they work as expected.

- Range checks are to be restricted to boundary testing only. For example, if a field has the validation conditions >= 0 and <=20000, then the required testing includes the following values; -1, 0, 20000, 20001.
- Data type checks will need to be tested with at least one valid value and one incorrect value. For example, an integer field might be tested with the values 4,000 and "h".
- Presence checks will need to be tested by omitting values in the field under test.
- Tests on buttons are to ensure that when clicked they behave as expected, that is, the correct function is called.

All tests should be presented in tabular form as indicated in table n (the tests included are meant only to serve as examples).

Test number	Description	Expected Outcome	Actual Outcome	Comment
1	Checking that the ID field in the suppliers form by leaving the field empty.	A message box is displayed warning the user.	A message box is displayed warning the user.	Refer to Figure xx in Appendix Y.
2	Checking that the ID field in the suppliers form by inputting relevant data in the field.	Input accepted	Input accepted	
...etc...				

Evaluation

System Evaluation

The candidate is expected to show that the requirements set have been met and whether the proposed solution is suitable for the set task.

Conclusion

To include considerations arising from what has been built including any limitations, the context in which it is going to be applied and any skills acquired by the candidate.

Future Enhancements

This should include any possible improvements which, given more time, and experience, could have been included in the assignment.

Appendix

Well-structured user manual with step-by-step instructions (and guiding pictures)

The user manual, as a minimum, is to include an introduction, grouping of functionality, usage explanation, and guiding pictures. In other words, the candidate must create step-by-step lists accompanied by relevant screenshots which a user can follow to make use of the system's main features.

Screenshots of all test cases (as proof of testing)

A simple list of screenshots is to be provided by the candidate as proof that the testing was carried out. For ease of reference, each screenshot should be captioned with the relevant test case number and presented in sequential order. Only screenshots for tests that expose a validation error should be included in this appendix.

Quality assurance	**Overall presentation and layout of documentation** All the listed items must be included in word processed format. The candidate is encouraged to take advantage of the features available in most word processing programs to create an orderly table of contents including page numbers and an APA-style bibliography. A bibliography is taken to be a set of resources that have been overviewed to gain insight into the domain of the assignment. In the header or footer a candidate should include his name, surname and government I.D. card number (or equivalent identity document), and the school's name (if applicable). A page number, in Arabic form, should be included to the far right of the footer. It is suggested that font sizes of the main text should be 12pt for body text and 16pt for main headings and 14pt for sub-heading. Font styles should be either Arial or Calibri. The Courier New font type should be used for any code snippets. Line spacing should be 1.5. All tables, diagrams and screenshots must be properly captioned and the caption reference should be used within the text to indicate which table, diagram, screenshot is being referred to. **Tables contain a minimum of 10 valid records each** The reason for this limitation is to ensure that sensible results can be achieved when using aggregate functions such as SUM or AVG. **Proper and consistent naming of fields and tables** Names chosen for fields and tables should intuitively indicate to anyone unfamiliar with the database what they represent and which data they are expected to hold. E.g., a table which holds customer records should intuitively be named 'Customers'. **Queries implemented through the Python GUI** All the queries to the database should only be considered as valid if they are launched through the Python part of the system. Clause 'b' in the marking sheet specifies that both commands, that is GROUP BY and ORDER BY, have to be used at least once. **Use of comments in coding** Comments are an essential part of a program's readability and allow for easier future maintenance of the system. As such, it is expected that the candidate writes a comment wherever relevant. As a minimum there should be at least one comment per function. **Properly indented code** Beyond the indentations required by the programming language itself, the candidate should strive to indent the assignment's code appropriately to improve the readability of the code. E.g., the comments should be indented in line with the code they refer to. **Proper and consistent naming of variables and functions** The names given to variables and functions should reflect the purpose for which they were created. This not only helps to improve the readability of the code but also improves the maintainability of the same code. E.g., a function which returns the Fibonacci sequence for a number of terms, should be called something along the lines of 'fibonacci_seq' or 'fibonacci_func'. **User-friendly interface** The candidate should make sure to create neat and orderly designs for the window(s) created as part of the GUI. E.g, fields for a customer form should be in a logical order starting from top to bottom with, e.g., name, surname, e-mail address, contact number and the input text-boxes neatly aligned with each other and having a size appropriate for the expected length of the input.

Marking Criteria: Databases and Programming

MARKING CRITERIA – Databases and Programming	Marks
System analysis and problem formulation (15 marks)	
Problem Definition + Background of the problem [300 to 400 words]	10
Scope [100 to 200 words]	5
Client Requirements (25 marks)	
Use-Case Diagram. To include a minimum of: • Adding data • Removing data • Updating data • Searching and view report • Sorting	10
Database requirements: a) Description of data entities and their relationships b) List of the attributes for the entities identified in a)	 2 3
Programming requirements: • Explanation of the functions illustrated in the Use-Case Diagram • Description of search and sort parameters in relation to the chosen assignment • Description of each report type and structure	 4 2 2
General description of data validation requirements	2
Database Design (25 marks)	
Database Entity Relationship Diagram (ERD)	12
Database normalisation (up to 3NF)	8
Description and justification of field types	5
Programming Design (20 marks)	
Program design shown through HIPO chart(s)	8
GUI prototype for every form or window as applicable	4
Description of data input validation in terms of: • Range Check(s) • Data Type Check(s) • Presence Check(s)	 3 3 2
Implementation (50 marks) *Candidates are expected to describe the use of the following elements and sub-sections, and to provide a working solution for each.*	
Database tables and their relationship (minimum of four)	4
Type casting	2
The connection of Python with the underlying database	3
The separation of the database manipulation code from the interface code through the use of separate Python files for the logic of the GUI, from the logic of the database.	2
Exception handling	3
Construction of functions with parameters and return values	3
Use of Queries: • GROUP BY across a minimum of two tables (using any from COUNT, AVG, SUM, MIN, MAX) • Delete query • Update query • Insert query	 3 2 2 3
Usage of lists/tuples/set/dictionaries	3
Use of loops	3
Use of conditional statements	2
Tkinter GUI objects - to include a minimum of: • Buttons • Text Field • Text Area • Labels • Grid or place layout managers • Data input validations (as per design section)	 2 2 2 2 3 4

Testing (20 marks)	
Requirements Testing	4
Black box Testing, to include: • All range checks • All data type checks • All presence checks • All buttons	 4 4 4 4
Evaluation (10 marks)	
System evaluation (scope of assignment vs. what was implemented)	5
Conclusion	2
Future Enhancements	3
Appendix (10 marks)	
Well structured user manual with step-by-step instructions (and guiding pictures)	8
Screenshots of all test cases (as proof of testing)	2
Quality Assurance: Overall Objectives and Quality of Assignment (25 marks)	
Overall presentation and layout of documentation which must include: • Table of Contents • Header and footer • Page numbering • Good use of fonts and styles • Captioned images inserted within margins • Bibliography using APA citation style	 1 1 1 1 1 1
Tables contain a minimum of 10 valid records each	2
Proper and consistent naming of fields and tables	1
Queries implemented through the Python GUI: • Minimum of 3 simple working single-table queries • Minimum of 3 multi-table working queries which must include GROUP BY and ORDER BY (it is not necessary for each multi-table query to contain both commands)	 3 6
Use of comments in coding	2
Properly indented code	1
Proper and consistent naming of variables and functions	1
User-friendly interface: • Intuitive items (field, buttons and other components) • Field layout • Logical grouping of buttons	3
Total:	**200**

Web Design Coursework

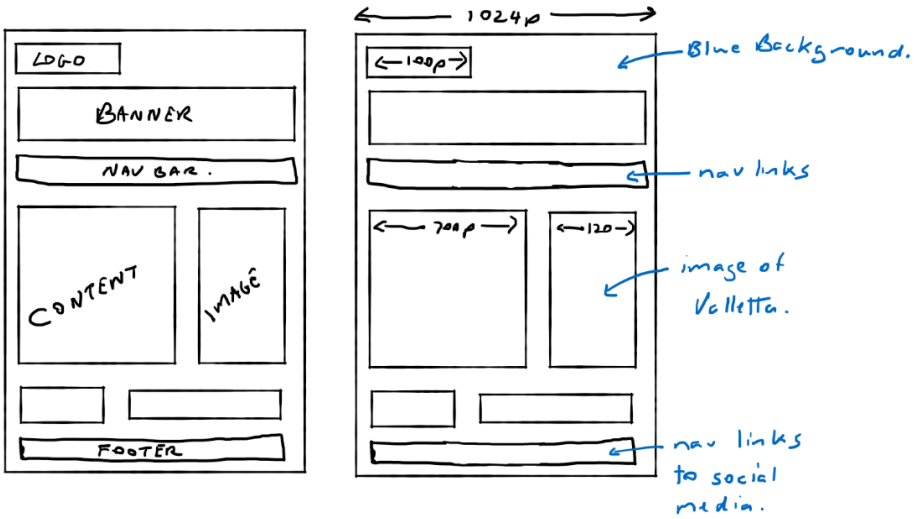
Web Design Coursework

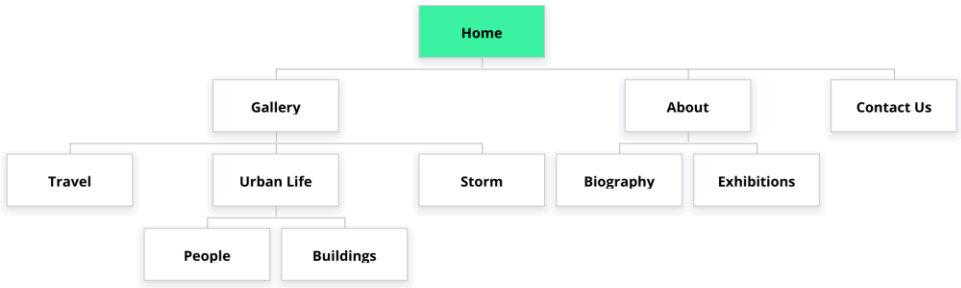
100 marks

The aim of this assignment is to introduce the candidates to HTML (Hyper Text Markup Language) and CSS (Cascade Style Sheets). The candidate should appreciate what goes on "behind the scenes" and is therefore not allowed to make use of any tools which auto-generate HTML or CSS code or make use of premade templates. The candidate's efforts should be aimed at learning how to use HTML tags to generate the content of simple web pages and applying style and presentation through a separate CSS file.

Software Requirements

The choice of a Web development tool must be limited to a text-based editor, e.g., Brackets, or any basic plain text editors, such as Notepad, VI editor, etc. Any tools or services that allow the automatic generation of HTML code, from diagrammatic or other forms, are not to be used.

Section	Details
Webpage Requirements	**Justification of name given to website and availability and appropriateness of choice of Internet domain name** The assignment should start with the selection of a context within which the website is to be created. For example, creating a website to donate and recycle toys for charity or an online computer shop. It is highly recommended that this is followed by the detailing of the web pages' requirements such as the selection of a domain name. For example, the domain name 'toys.com' is already taken up and it is unlikely that a voluntary organisation can afford buying it off its current owners should they even consider selling it. **Description of each web page** As detailed in the marking scheme, the candidate should briefly describe the planned content of a minimum of five web pages. The content should be relevant to the context chosen, and the colour schemes should reflect the theme of the same context. It is recommended that candidates check whether the colours chosen meet widely accepted accessibility standards. The 'Contact Us' page should contain a form whose fields are listed in the 'Implementation' section of this document
Design	**Paper prototype of each page** Paper prototypes should serve as a graphical equivalent to the textual description of each page in order to further clarify the proposed organisation and layout of each page. For guidance purposes one could use the following example to get an idea of what is expected in this section. 

	Sitemap A simple hierarchical diagram depicting all the pages of the site and their organisation is sufficient. For guidance purposes one could use the following example to get an idea of what is expected in this section.  **Cascading Style Sheet (CSS) elements** The candidates should specify details of font styles, sizes, alignment and colours for the main elements identified in the layouts in the paper prototype. As a minimum this should include the background colour for the pages as well as details for the <h1>, <h2>, <a>, <nav>, and <footer> elements.
Implementation	**Use of div tags with class or IDs** The candidate is to appreciate that HTML tags, or even tags within individual divisions, can be customised with their own CSS code. The web pages should be created using multiple divs layered together to form a single page. Formatting for divs and other tags should be done through the use of CSS rules attached to ID and class attributes or the tags themselves. E.g., The candidate might specify "default" CSS code for the <p> tag but then customise a particular <p> tag which is meant to look different from the higher level <p> tags by using an ID or class attribute. Similarly, if the candidate requires the enforcement of a rule for all <p> tags, the rule might be added to all <p> tags directly. **Minimum of five different pages** The candidate should present a minimum of five HTML pages including a "Homepage" and a "Contact Us" page. **Use of title tag for every page** Filling in the <title> tag is required since this text will appear in the title bar and tabs in a browser which helps the user to navigate the website when using multiple windows. **Use of meta tags** The meta tags specified in the marking scheme are to be included in every HTML page such that web crawlers can identify what the website is about. **Use of external CSS linked to all pages to ensure uniform presentation** The candidate is to ensure that all the CSS rules are placed in a CSS file separate from the HTML files. Individual tags which need separate modifications can always be addressed using IDs as mentioned earlier. **Limitations of a "Contact Us" page** The HTML form does not need to implement any Javascript. That is to say that the candidate is not expected to present a working form which can be submitted to a server since the steps involved do not fall within the scope of this assignment. **Use of at least one table** The candidate is expected to make use of at least one HTML table where appropriate. E.g., to display a list of products sold by a shop. **Use of Hyperlinks** By internal links it is understood that such links direct the website's user to another page within the same website. Whereas external links direct the user to other websites. Anchors should direct the user to other points within the same web page which is especially useful for web pages with lengthy content.

Ideally the mail-to link would be placed within the “Contact Us” page, however other placements should be accepted as long as they fit into the website context chosen by the candidate.

Use of images

The candidate should ensure that all images are relevant to the context chosen.

Use of embedded content

The candidate is encouraged to research the Web for the numerous educational materials that give step-by-step instructions on how to include responsive videos and online maps.

Page footer

The candidate is to create at least three social media icons. The candidate is not expected to create the social media pages themselves, but to just create a link to the homepage of the social media website.

Evaluation

Overall presentation and layout of documentation

All the listed items must be included in word processed format. The candidate is encouraged to take advantage of the features available in most word processing programs to create an orderly table of contents including page numbers and an APA-style bibliography. A bibliography is taken to be a set of resources that have been overviewed to gain insight into the domain of the assignment.

In the header or footer a candidate should include his name, surname and government I.D. card number (or equivalent identity document), and the school's name (if applicable). A page number, in Arabic form, should be included to the far right of the footer.

It is suggested that font sizes of the main text should be 12pt for body text and 16pt for main headings and 14pt for sub-heading. Font styles should be either Arial or Calibri. The Courier New font type should be used for any code snippets.

Line spacing should be 1.5. All tables, diagrams and screenshots must be properly captioned and the caption reference should be used within the text to indicate which table, diagram, screenshot is being referred to.

Requirements coverage

For this section the candidate should present a checklist of items which are to be verified as completed prior to delivering the website. The following tabular form can be taken as an example.

Requirement	Completed?
A Google map was included in the “Contact Us” page.	Y
The background colour of all the pages is fuchsia.	Y
The font style used for the <h1> tags is Arial.	Y
... etc…	

Testing that all links work

All tests should be presented in tabular format as indicated below (the tests included below are meant only to serve as examples).

Test number	Description	Expected Outcome	Actual Outcome
1	Test button Home in page Contact Us	Home page opens	Home page opens
2	Test mail-to link in page Contact Us	Email client opens	Email client opens
3	Test google map in page Contact Us	Google map opens	Google map opens
… etc …			

	Test under HTML validator The candidate must include at least one screenshot for every webpage as proof that all the webpages were tested. **Test under CSS validator:** The candidate must include at least one screenshot proving that the CSS file/s were tested. **Works without horizontal scrolling** Content in each page must fit a screen width of 1024 pixels resolution in order to avoid any horizontal scrolling. **Proper folder structure is to be used:** All website files must be located within the same folder. However, any images and downloadable resources should be paced in appropriately named separate subfolders. **Use of images which are not resized through browser:** Resizing of images must not be done through HTML or CSS code, but must be done beforehand using an image editor. **Legibility:** The colour scheme used throughout the site should be consistent. The proposed site should be readable. Readability refers to having sufficient contrast between the foreground colour and background colour such that the text content can be easily read.
Appendix	As a minimum, screenshots of all web pages in at least two different browsers and the test runs are expected to be included in the appendix.

Marking Criteria: Web Design

MARKING CRITERIA – Web Design	Marks
Web Page Requirements (10 marks)	
Justification of name given to website and availability and appropriateness of choice of Internet domain name	1
Description of each web page (up to a minimum of 5). One of the five web pages is to include an HTML contact form. Description should include: a) content and layout of the header, footer, and nav bar; b) the colour scheme used throughout the site in terms of readability and consistency; c) a list of images and video(s). The description could also include any other multimedia content (e.g. audio, slideshow, etc.).	4 3 2
Design (15 marks)	
Paper prototype of each page	5
Sitemap	2
Cascading Style Sheet (CSS) elements to modify: a) font properties; b) text properties; c) page properties.	2 3 3
Implementation (45 marks) *Candidates do not need to describe the steps in the following but need to provide a working solution for each website element below.*	
Use of div tags with class or IDs	5
Minimum of five different pages (including a Homepage and Contact us page)	5
Use of title tag for every page	2
Use of meta tags (to include author, keywords, charset, and description)	4
Use of an external style sheet (CSS) linked to all pages to ensure uniform presentation.	4
Use of a "Contact Us" page with a form containing fields: a) name; b) surname; c) email; d) comments; e) submit button; f) reset button.	3
Use of at least one table	3
Use of Hyperlinks (minimum of one of each type must be used): a) internal; b) external; c) anchor; d) mail-to.	8
Use of images	3
Use of embedded content: a) Online map; b) Video.	4
Page footer must contain: a) copyright notice (i.e. © Joe Borg YYYY); b) social media icons.	4
Evaluation: Overall Objectives of Assignment (25 marks)	
Overall presentation and layout of documentation which must include: a) Table of Contents b) Header and footer c) Page numbering d) Good use of fonts and styles e) Captioned images inserted within margins f) Bibliography using APA citation style	3
Requirements coverage (in the form of a checklist)	3
Testing that all links work (in tabular form - see annex)	3
Test under HTML validator: http://validator.w3.org (in the form of a screenshot)	2
Test under CSS validator: http://jigsaw.w3.org/css-validator (in the form of a screenshot)	2
Works without horizontal scrolling	2
Proper folder structure is to be used	3
Use of images which are not resized through the browser, but are properly resized using a photo editor prior to insertion into the HTML	3
Legibility (contrast between text and background, text size and font type considerations)	4
Appendix (5 marks)	
Screenshots of all web pages in at least two different browsers.	5
Total:	**100**