



OTHM Level 7 Diploma in Immersive Software Engineering

Qualification Number: 610/3058/2

Specification | 2023

TABLE OF CONTENTS

QUALIFICATION OBJECTIVES.....	3
QUALITY, STANDARDS AND RECOGNITIONS	4
REGULATORY INFORMATION.....	4
EQUIVALENCES	4
QUALIFICATION STRUCTURE	4
DEFINITIONS.....	5
ENTRY REQUIREMENTS.....	5
PROGRESSION.....	6
DELIVERY OF OTHM QUALIFICATIONS	6
CENTRE RESOURCE REQUIREMENTS	6
ASSESSMENT AND VERIFICATION.....	7
OPPORTUNITIES FOR LEARNERS TO PASS	7
RECOGNITION OF PRIOR LEARNING AND ACHIEVEMENT	7
EQUALITY AND DIVERSITY	8
UNIT SPECIFICATIONS.....	9
Security Engineering.....	10
Software Programming Principles and Practices in Java I	17
Agile Project Management.....	22
Cloud Computing & DevOps	27
Database & SQL Programming	31
Web Designing	36
IMPORTANT NOTE.....	40

QUALIFICATION OBJECTIVES

The principal objective of the OTHM Level 7 Diploma in Immersive Software Engineering is to develop the knowledge skills, critical understanding necessary to design, develop, and evaluate immersive software engineering systems, which can be used to support research and innovation.

The qualification aims to equip learners with advanced-level knowledge and skills in immersive software engineering that can be applied to various domains such as Web & Cloud technologies, Security, Automation, Data analytics and Project methodologies. It also aims to develop learners' critical thinking and research skills to identify and address complex problems in the software engineering field.

OTHM has developed a suite of Level 7 Diploma qualifications. The qualifications provide learners with industry-specific and practical skills, enabling them to successfully apply their knowledge in the workplace, enhance their career prospects and allow progression to further study.

Key features of OTHM Level 7 qualifications.

- A stimulating and challenging programme of study that will be both engaging and informative for learners.
- Learners can gain the essential subject knowledge needed to progress successfully into further study or the world of work.
- Refreshed content that is closely aligned with employer and higher education needs,
- Assessments that consider cognitive skills along with affective and applied skills.
- Learners will develop knowledge and academic study skills including active research skills, effective writing skills, analytical skills, creative problem-solving, decision-making, and digital literacy.

Upon successfully completing the OTHM Level 7 Diploma in Immersive Software Engineering learners will be able to:

Develop a critical understanding of immersive software engineering and design software systems that meet industry standards and user requirements. They potentially can evaluate software development methodologies, analyse and solve complex problems, and apply advanced programming skills using object-oriented programming languages and development tools. Learners will also critically evaluate the ethical and social implications of immersive software engineering and propose solutions that promote ethical and socially responsible practices. They can conduct independent research and communicate complex technical information effectively using appropriate media and presentation techniques. Learners will work collaboratively in a team environment, demonstrating effective leadership, communication, and interpersonal skills.

QUALITY, STANDARDS AND RECOGNITIONS

OTHM Qualifications are approved and regulated by Ofqual (Office of Qualifications and Examinations Regulation). Visit the [Register of Regulated Qualifications](#).

OTHM has progression arrangements with several UK universities that acknowledges the ability of learners after studying level 7 qualifications to be considered for advanced entry into corresponding master's programmes.

REGULATORY INFORMATION

Qualification Title	OTHM Level 7 Diploma in Immersive Software Engineering
Qualification Ref. Number	610/3058/2
Regulation Start Date	27/7/2023
Operational Start Date	27/7/2023
Duration	1 Year
Total Credit Value	120 Credits
Total Qualification Time (TQT)	1200 Hours
Guided Learning Hours (GLH)	600 Hours
Sector Subject Area (SSA)	6.1 - ICT practitioners
Overall Grading Type	Pass / Fail
Assessment Methods	Coursework
Language of Assessment	English

EQUIVALENCES

OTHM level 7 diplomas are located on the Regulated Qualifications Framework (RQF) and are recognised as being at the same level as master's degrees. However, they are shorter (120 credits) qualifications which means learners will have to proceed to the dissertation stage (60 credits) with an appropriate university to achieve a full master's qualification.

QUALIFICATION STRUCTURE

The OTHM Level 7 Diploma in Immersive Software Engineering consists of 6 mandatory units for a combined total of 120 credits, 1200 hours Total Qualification Time (TQT) and 600 Guided Learning Hours (GLH) for the completed qualification.

Unit Ref.	Mandatory units	Credits	GLH	TQT
F/650/7994	Security Engineering	20	100	200
H/650/7995	Software programming principles and practices in Java I	20	100	200
J/650/7996	Agile Project Management	20	100	200
K/650/7997	Cloud Computing & DevOps	20	100	200
L/650/7998	Database & SQL Programming	20	100	200
M/650/7999	Web Designing	20	100	200

DEFINITIONS

Total Qualification Time (TQT) is the number of notional hours which represents an estimate of the total amount of time that could reasonably be expected to be required in order for a Learner to achieve and demonstrate the achievement of the level of attainment necessary for the award of a qualification.

Total Qualification Time is comprised of the following two elements –

- a) *the number of hours which an awarding organisation has assigned to a qualification for Guided Learning, and*
- b) *an estimate of the number of hours a Learner will reasonably be likely to spend in preparation, study or any other form of participation in education or training, including assessment, which takes place as directed by – but, unlike Guided Learning, not under the Immediate Guidance or Supervision of – a lecturer, supervisor, tutor or other appropriate provider of education or training.*

(Ofqual 15/5775 September 2015)

Guided Learning Hours (GLH) is defined as the hours that a teacher, lecturer or other member of staff is available to provide immediate teaching support or supervision to a learner working towards a qualification.

Credit value is defined as being the number of credits that may be awarded to a Learner for the successful achievement of the learning outcomes of a unit. One credit is equal to 10 hours of TQT.

ENTRY REQUIREMENTS

For entry onto the OTHM Level 7 Diploma in Immersive Software Engineering qualification, learners must possess:

- An honours degree in related subject or UK level 6 diploma or an equivalent overseas qualification
- Mature learners with management experience (learners must check with the delivery centre regarding this experience prior to registering for the programme)
- Learner must be 21 years old or older at the beginning of the course.

English requirements: If a learner is not from a majority English-speaking country must provide evidence of English language competency. For more information visit [English Language Expectations](#) page on our website www.othm.org.uk.

Alternative professional qualifications with at least three years' relevant work experience in the public service field may also be considered. This could be in roles in local or national government, or in non-governmental and inter-governmental organisations, the voluntary and charitable sector, and private sector roles which support or deliver public services.

PROGRESSION

The OTHM Level 7 Diploma in Immersive Software Engineering enables learners to progress into or within employment and/or continue their further study.

As this qualification is approved and regulated by Ofqual (Office of the Qualifications and Examinations Regulation), learners may be eligible to progress to master's top-up at many universities in the UK and overseas with advanced standing. For more information visit the [University Progressions](#) page on the OTHM website.

DELIVERY OF OTHM QUALIFICATIONS

OTHM do not specify the mode of delivery for its qualifications, therefore OTHM Centres are free to deliver this qualification using any mode of delivery that meets the needs of their Learners. However, OTHM Centres should consider the Learners' complete learning experience when designing the delivery of programmes.

OTHM Centres must ensure that the chosen mode of delivery does not unlawfully or unfairly discriminate, whether directly or indirectly, and that equality of opportunity is promoted. Where it is reasonable and practicable to do so, it will take steps to address identified inequalities or barriers that may arise.

Guided Learning Hours (GLH) which are listed in each unit gives the Centres the number of hours of teacher-supervised or direct study time likely to be required to teach that unit.

The qualification has been designed to take learners on a structured learning pathway. The sequencing of units is likely to encourage proactive engagement due to the nature of the subjects and topics therein, whilst also supporting learners to develop the learning and assessment skills required to be successful at level 7.

CENTRE RESOURCE REQUIREMENTS

Tutor / Assessor Requirements

- Tutors/Assessors must be appropriately qualified and occupationally competent in the areas in which they are training.
- They must hold a Level 6 qualification or equivalent.
- They should hold or be working towards a Level 3 qualification in Assessing Vocationally Related Achievement such as the OTHM Level 3 Award in Assessing Vocationally Related Achievement.

Internal Verifier Requirements

- Internal quality assurers or verifiers must be appropriately qualified and occupationally competent in the areas in which they are moderating.
- They must hold or be working towards a Level 4 Award in the Internal Quality Assurance of Assessment Processes and Practice and/or a Level 4 Certificate in Leading the Internal Quality Assurance of Assessment Processes and Practice such as the OTHM Level 4 Certificate in Leading the Internal Quality Assurance of Assessment Processes and Practice.

- They must demonstrate that they have undertaken Continued Professional Development (CPD) activities relating to occupational health and safety or auditing quality assurance to maintain and update their skills and knowledge within the last year.

OTHM will request to see copies of relevant qualifications from assessors and verifiers.

ASSESSMENT AND VERIFICATION

The units in this qualification are internally assessed by the centre and externally verified by OTHM. The qualifications are criterion referenced, based on the achievement of all the specified learning outcomes.

To achieve a 'pass' for a unit, learners must provide evidence to demonstrate that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria. Judgement that the learners have successfully fulfilled the assessment criteria is made by the Assessor.

The Assessor should provide an audit trail showing how the judgement of the learners' overall achievement has been arrived at.

Specific assessment guidance and relevant marking criteria for each unit are made available in the Assignment Brief document. These are made available to centres immediately after registration of one or more learners.

OPPORTUNITIES FOR LEARNERS TO PASS

Centres are responsible for managing learners who have not achieved a Pass for the qualification having completed the assessment. However, OTHM expects at a minimum that centres must have in place a clear feedback mechanism to learners by which they can effectively retrain the learner in all the areas required before re-assessing the learner.

RECOGNITION OF PRIOR LEARNING AND ACHIEVEMENT

Recognition of Prior Learning (RPL) is a method of assessment that considers whether learners can demonstrate that they can meet the assessment requirements for a unit through knowledge, understanding or skills they already possess and do not need to develop through a course of learning.

RPL policies and procedures have been developed over time, which has led to the use of a number of terms to describe the process. Among the most common are:

- Accreditation of Prior Learning (APL)
- Accreditation of Prior Experiential Learning (APEL)
- Accreditation of Prior Achievement (APA)
- Accreditation of Prior Learning and Achievement (APLA).

All evidence must be evaluated with reference to the stipulated learning outcomes and assessment criteria against the respective unit(s). The assessor must be satisfied that the evidence produced by the learner meets the assessment standard established by the learning outcome and its related assessment criteria at that particular level.

Most often RPL will be used for units. It is not acceptable to claim for an entire qualification through RPL. Where evidence is assessed to be only sufficient to cover one or more learning outcomes, or to partly meet the need of a learning outcome, then additional assessment methods should be used to generate sufficient evidence to be able to award the learning outcome(s) for the whole unit. This may include a combination of units where applicable.

EQUALITY AND DIVERSITY

OTHM provides equality and diversity training to staff and consultants. This makes clear that staff and consultants must comply with the requirements of the Equality Act 2010, and all other related equality and diversity legislation, in relation to our qualifications.

We develop and revise our qualifications to avoid, where possible, any feature that might disadvantage learners because of their age, disability, gender, pregnancy or maternity, race, religion or belief, and sexual orientation.

If a specific qualification requires a feature that might disadvantage a particular group (e.g. a legal requirement regarding health and safety in the workplace), we will clarify this explicitly in the qualification specification.

UNIT SPECIFICATIONS

Security Engineering

Unit Reference Number	F/650/7994
Unit Title	Security Engineering
Unit Level	7
Number of Credits	20
Total Qualification Time (TQT)	200
Guided Learning Hours (GLH)	100
Mandatory / Optional	Mandatory
Sector Subject Area (SSA)	6.1 - ICT practitioners
Unit Grading type	Pass / Fail

Unit Aims

The aim of this unit is to develop learners understanding and skills in secure software development life cycle, system hardening techniques, authentication and authorisation processes, encryption techniques, network security measures to analyse, evaluate security of systems and design web applications architecture. The learner can critically evaluate various security tools and standards for security engineering in a range of data security measures, social engineering attacks, to explore and implement system hardening techniques and acquire the knowledge and skills to be able to design a secure system.

Learning Outcomes, Assessment Criteria, and Indicative Content

Learning Outcomes – the learner will:	Assessment Criteria – the learner can:	Indicative content
1. Be able to analyse and evaluate the security of systems and web applications architecture.	1.1 Analyse the security of web application architecture and provide recommendations for improvement.	<ul style="list-style-type: none"> ▪ Web application architecture to identify potential security risks. ▪ Basic architecture of web applications and their components. ▪ OWASP Top 10 web application security risks and their implications.

	<p>1.2 Classify the types of security models that are applicable to different system architectures.</p> <p>1.3 Critique the security models and threat models used in each given system architecture, identifying any potential vulnerabilities, and suggesting improvements.</p> <p>1.4 Demonstrate the ability to evaluate the attack surfaces of a given system architecture and recommend measures to mitigate potential attacks.</p>	<ul style="list-style-type: none"> ▪ Existing security measures and recommended improvements. ▪ Recommended improvements to enhance the security of the web application. ▪ Different security models and their features. ▪ Factors that influence the choice of a security model for a given system architecture. ▪ Different types of security models according to their characteristics and suitability for specific system architectures. ▪ Advantages and disadvantages of different security models in different contexts. ▪ Security models and creating threat models used in each system architecture to identify potential vulnerabilities. ▪ Critical assessment of existing security models and threat models in mitigating security risks. ▪ Potential risks associated with different attack surfaces. ▪ Measures to mitigate potential attacks and enhance the security of the system architecture.
--	---	---

<p>2. Be able to implement system hardening techniques.</p>	<p>2.1 Implement system hardening techniques effectively.</p> <p>2.2 Understand and implement authentication and authorization processes that meet industry standards.</p> <p>2.3 Review the system configurations of a given network architecture and propose hardening techniques that comply with industry standards.</p> <p>2.4 Design a plan for implementing system hardening techniques that addresses the security requirements of a given network architecture.</p>	<ul style="list-style-type: none"> ▪ System hardening and the purpose of reviewing system configurations. ▪ Common system configurations and their potential security vulnerabilities. ▪ System configurations of a given network architecture. ▪ Various system hardening techniques and recommendations that comply with industry standards. ▪ Authentication and authorization processes. ▪ Security requirements of a given network architecture and assess authentication and authorization mechanisms. ▪ Implementing authentication and authorization processes that meet industry standards. ▪ Testing and validating SSO (Single sign on) security controls.
<p>3. Be able to apply encryption techniques to secure data.</p>	<p>3.1 Apply encryption techniques effectively to secure data.</p> <p>3.2 Devise and implement network security measures that meet industry standards.</p> <p>3.3 Examine and implement network security measures.</p>	<ul style="list-style-type: none"> ▪ Symmetric and asymmetric encryption techniques and encryption algorithms. ▪ Encryption key management principles and best practices. ▪ Encryption techniques to secure data in different contexts. ▪ Encryption techniques in securing data. ▪ Encryption tools and software. ▪ Critique network security models, firewalls, and standards.

		<ul style="list-style-type: none"> ▪ Different types of intrusion detection and prevention systems (IDS/IPS), and virtual private networks (VPNs). ▪ Network security measures in mitigating different types of attacks.
<p>4. Be able to understand and implement data security measures including identifying and mitigating social engineering attacks.</p>	<p>4.1 Analyse malware and develop countermeasures that meet industry standards.</p> <p>4.2 Recognise and implement data security measures that meet industry standards.</p> <p>4.3 Identify and mitigate social engineering attacks effectively.</p>	<ul style="list-style-type: none"> ▪ Types of malwares and their characteristics ▪ Techniques for analyzing malware behavior. ▪ Countermeasures for malware, including detection, removal, and prevention. ▪ Industry standards for malware protection ▪ Types of data and their security requirements ▪ Data security models and their features ▪ Encryption algorithms and key management techniques ▪ Data backup and recovery strategies ▪ Access control mechanisms and policies ▪ Types of social engineering attacks and their methods ▪ Techniques for identifying social engineering attempts. ▪ Strategies for mitigating social engineering attacks. ▪ Best practices for user awareness and training

		<ul style="list-style-type: none"> ▪ Industry standards and guidelines for preventing social engineering attacks.
<p>5. Be able to use tools and standards for security engineering.</p>	<p>5.1 Identify the relevant security engineering tools and explain their functions and features.</p> <p>5.2 Interpret the security engineering standards and frameworks (e.g., NIST Cybersecurity Framework, ISO 27001) and apply them to a given scenario.</p>	<ul style="list-style-type: none"> ▪ Different security engineering tools such as network scanners, vulnerability scanners, firewall management tools, intrusion detection and prevention systems, forensic analysis tools, encryption tools, etc. ▪ Various functions and features of each tool, such as the types of vulnerabilities it can detect, its scanning capabilities, its reporting and alerting mechanisms, etc. ▪ Security engineering tools based on specific requirements and constraints. ▪ Critical comparison of different security engineering tools based on their effectiveness, ease of use, cost, compatibility with different systems, etc. ▪ Various security engineering standards and frameworks and their objectives, such as the NIST Cybersecurity Framework, ISO 27001, COBIT, etc. ▪ Key components of each standard or framework, such as risk management, security controls, governance, compliance, etc. ▪ Key requirements of each standard or framework and their applicability to a given scenario or system.

		<ul style="list-style-type: none"> ▪ Gaps and deficiencies in each system with respect to the requirements of a specific security engineering standard or framework. ▪ Privacy, compliance, and governance for securing software systems.
--	--	---

Assessment

To achieve a 'pass' for this unit, learners must provide evidence to demonstrate that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria.

Learning Outcomes to be met	Assessment criteria to be covered	Type of assessment	Word count (approx. length)
LO1	All AC's under LO1	Essay	1500 words
LO4	All AC's under LO4	Written Report	2500 – 3000 Words
LO3 and LO5	All AC's under LO3 and L05	Practical Assessments	N/A
LO2	All AC's under LO2	Presentations	2500-3000 Words

Indicative Reading List

Goutam, R. K. (2021). *Cybersecurity Fundamentals: Understand the Role of Cybersecurity, Its Importance and Modern Techniques Used by Cybersecurity Professionals (English Edition)*. BPB Publications.

Barnett, R. C. And Grossman, J. (2012) *Web Application Defender's Cookbook*. O'Reilly.

Bidgoli, H. (2006). *Handbook of information security, information warfare, social, legal, and international issues and security foundations (Vol. 2)*. John Wiley & Sons.

Stallings, W., Brown, L., Bauer, M. D., & Howard, M. (2012). *Computer security: principles and practice (Vol. 3)*. Upper Saddle River: Pearson.

Schneier, B. (2007). *Applied cryptography: protocols, algorithms, and source code in C*. John Wiley & Sons.

Ferguson, N., Schneier, B., & Kohno, T. (2011). *Cryptography engineering: design principles and practical applications*. John Wiley & Sons.

Moore, R. (2014). *Cybercrime: Investigating high-technology computer crime*. Routledge.

Perlman, R., Kaufman, C., & Speciner, M. (2016). *Network security: private communication in a public world*. Pearson Education India.

Mitnick, K. D., & Simon, W. L. (2009). *The art of intrusion: the real stories behind the exploits of hackers, intruders and deceivers*. John Wiley & Sons.

Pfleeger, C. P., & Pfleeger, S. L. (2012). *Analyzing computer security: A threat/vulnerability/countermeasure approach*. Prentice Hall Professional.

Software Programming Principles and Practices in Java I

Unit Reference Number	H/650/7995
Unit Title	Software Programming Principles and Practices in Java I
Unit Level	7
Number of Credits	20
Total Qualification Time (TQT)	200
Guided Learning Hours (GLH)	100
Mandatory / Optional	Mandatory
Sector Subject Area (SSA)	6.1 - ICT practitioners
Unit Grading type	Pass / Fail

Unit Aims

The aim of this unit is to develop learners understanding on what a programming language is, how it works and how to interact with computers using a programming language. The learner will cover all the basic concepts of a programming language and is taught through real world coding examples, the learner will have regular coding assignments for better understanding through practice.

Learning Outcomes, Assessment Criteria, and Indicative Content

Learning Outcomes – the learner will:	Assessment Criteria – the learner can:	Indicative content
1. Understand the importance of programming in the current world	1.1 Analyse and evaluate the purpose of programming and see how we communicate with the computer.	Principles of programming, file extensions, knowledge about the compiler which is used for compiling code, and insights about byte code, which is the code understood by the computer.

<p>2. Understand why java is called a portable language</p>	<p>2.1 Identify why portability is important and why java is popular compared to its previous languages</p>	<p>Java Virtual Machine (JVM), Java Development Kit (JDK), Software Development Kit (SDK), class files(which contains byte code), javac(Command that is used to compile code)</p>
<p>3. Understand how data is stored and processed</p>	<p>3.1 Understand different data types and their uses int, float, long, boolean, String, double, long, char</p>	<p>Explain different data types to store data locally. int: To store smaller integers, float: To store smaller decimal numbers, long: To store large integers, boolean: To store TRUE or FALSE, char: To store a single character. string: To store collection of characters, double: To store large decimal numbers,</p>
<p>4. Understand about arithmetic and Boolean operators</p>	<p>4.1 Analyse and implement these operators with examples. 4.2 Analyse and implement these operators with examples</p>	<p>Arithmetic- +, -, *, /, *=, +=, /=, -=, ++, --, % Boolean- &&, , &, , >, <, >=, <=, ==, != Arithmetic and Boolean operators are used to make different type of calculations with the data types</p>
<p>5. Be able to understand multiple conditional statements and breaking out of them</p>	<p>5.1 Evaluate and implement conditional statements with examples. 5.2 Design and implement conditional statements with examples if, else, else if 5.3 Design and implement multi-conditional statements along with break and continue with examples</p>	<p>Conditional- if, else, else if Multi-conditional- switch, break, continue. Both these conditional statements will use boolean operators and will use boolean output to make a decision.</p>

	5.4 Evaluate and implement multi-conditional statements along with break and continue with examples	
6. Understand the usage and importance of loops and functions	<p>6.1 Formulate and Implement loops with examples for, while, dowhile loops with examples.</p> <p>6.2 Formulate different type of functions with examples</p>	<p>Loops are used to repeat a specific code for the required number of times based on condition and the different types of loops are</p> <p>Loops: - for, while, do while</p> <p>Functions are used for core reusability and code modularity.</p> <p>Functions- function prototype, body, return type and return statements,</p>
7.Introduction to multiple data storage types	7.1 Explain and design different type of arrays with all the above concepts	<p>Arrays are used to store multiple values.</p> <p>Arrays, index of arrays, arrays with conditions, functions, return type</p>
8. Introduction to classes and objects	8.1 Explain and Implement different type of classes and objects with examples-	<p>Classes are used for encapsulation of code and will represent real world entities through code.</p> <p>Class, object, constructor, getter, setter</p> <p>Arrays, index of arrays, arrays with conditions, functions, return type</p>

Assessment

To achieve a 'pass' for this unit, learners must provide evidence to demonstrate that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria.

Learning Outcomes to be met	Assessment criteria to be covered	Type of assessment	Word count (approx. length)
LO1- LO8	ACs under 1-8	Assignment Project	<ul style="list-style-type: none"> Code assignments to be submitted which can be executed on evaluator's machine. No word count- It will be measured by the quality of the code written and executed efficiently without errors ideally or with any minimal code config tweaks.

Indicative Reading List

Bloch, J. (2008). Effective java (the java series). Prentice Hall PTR.

Burd, B. (2017). Beginning programming with Java for Dummies. John Wiley & Sons.

Eckel, B. (1998). Thinking in Java, MindView. Inc.

Horstmann, S., Cornell, G., & Java, C. (2). Volume I-Fundamentals.

Schildt, H. (2022). Java: a beginner's guide. McGraw-Hill Education.

Sierra, K., & Bates, B. (2003). Headfirst Java; covers Java 5.0. A Brain Friendly Guide, O'REILLY Publication, 2nd Edition, ISBN-10, 596009208.

Agile Project Management

Unit Reference Number	J/650/7996
Unit Title	Agile Project Management
Unit Level	7
Number of Credits	20
Total Qualification Time (TQT)	200
Guided Learning Hours (GLH)	100
Mandatory / Optional	Mandatory
Sector Subject Area (SSA)	6.1 - ICT practitioners
Unit Grading type	Pass / Fail

Unit Aims

The primary aim of this unit is to equip learners with a comprehensive understanding of traditional Project Management with Waterfall SDLC, Business Agility, Scrum framework, and Kanban for building IT products. The learners will gain proficiency in the Scrum framework to build products iteratively and incrementally while learning about roles such as Scrum Master, Product Owner, and Developers. Moreover, they will explore how to utilize Kanban to enhance the flow of value and create customer-centric products.

Furthermore, learners will gain valuable insights into agile planning and estimation, as well as team dynamics in an agile environment. Lastly, they will develop knowledge in scaling agile for creating complex products with large teams using available frameworks. By participating in practical exercises, learners will acquire the ability to apply relevant tools and techniques in real-world situations.

Learning Outcomes, Assessment Criteria, and Indicative Content

Learning Outcomes – the learner will:	Assessment Criteria – the learner can:	Indicative content
1. Be able to understand the fundamentals of project management and the role of project manager	1.1 Understand the context of executing projects & role of project manager in delivering successful projects across industries	Role of project manager, Triple constraints (time, cost & scope), stakeholder engagement ,
2. Be able to understand traditional project management with Waterfall SDLC (software development life cycle)	2.1 Influence traditional project management tools & techniques to run IT projects using waterfall SDLC	Waterfall SDLC (software development life cycle), Phase gates
3. Be able to understand & apply business agility in the context dynamic market changes	3.1 Define business agility and apply customer centricity in the context of dynamic markets. 3.2 Apply design thinking to build the right products.	VUCA (volatile, uncertain, complex & ambiguous) environment, Customer centricity, Design thinking, Agile manifesto, Agile principles
4. Be able to understand & apply scrum framework to build IT products	4.1 Understand Scrum and able to work with a scrum team to build products iteratively & incrementally	Cone of uncertainty, Empiricism, Scrum – Roles, Events, Artifacts, Servant leadership, Product backlog & vision
5. Be able to attain knowledge required to perform roles like scrum master, product owner & developers	5.1 Understand Scrum and able to work in a scrum team to build products iteratively & incrementally	Scrum master, Leading change, Facilitation, Product owner, Product backlog management, Agile development, Extreme programming
6. Be able to understand & leverage kanban to optimize the flow of value	6.1 Build kanban systems to optimize flow of value	Visualizing work in progress, optimizing flow of value, Pull systems
7. Be able to understand & leverage product mindset to build customer centric products	7.1 Define, design, build, validate & measure customer centric products	Product mindset, Product ownership, Product strategy, Systems thinking, Agile metrics & KPIs with intent

8. Be able to understand & demonstrate knowledge of agile planning & estimation	8.1 Estimate Product backlog Items to plan iterations & releases	Daily planning, Iteration planning, Release planning, Product planning, Strategy planning, Relative estimation, User stories
9. Be able to understand & demonstrate team dynamics within an agile team	9.1 Build cross functional and self-manageable agile teams	Tuckman's ladder for agile teams, Self-organisation, Cross functionality, Agile maturity
10. Be able to understand & demonstrate scaling agile to build complex products with large teams	10.1 Define when and why scaling agile is required. 10.2 Gain foundational knowledge to leverage available scaling Frameworks to scale agile from team to organisation level	Scaling agile, Scaled agile framework (SAFe), Large scale scrum (LeSS), Nexus, Scrum at scale

Assessment

To achieve a 'pass' for this unit, learners must provide evidence to demonstrate that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria.

Learning Outcomes to be met	Assessment criteria to be covered	Type of assessment	Word count (approx. length)
LO1&2	ACs under LO1 & 2	Presentations	N/A
LO3	ACs under LO3	Essay	1000-1500 words
LO4 & 5	ACs under LO4 & 5	Practical Assessments	N/A
LO6	ACs under LO6	Practical Assessments	N/A
LO7	ACs under LO7	Written Report	1000-1500 words
LO8	ACs under LO8	Practical Assessments	N/A
LO9&10	ACs under 9,10	Presentations	N/A

Indicative Reading List

Joseph Philipps (2018) CAPM/PMP Project Management Certification All-In-One Exam Guide, Fourth Edition McGraw Hill

Ken Schwaber and Jeff Sutherland (2022). Scrum Guide This publication is offered for licence under the Attribution Share-Alike license of Creative Commons. <https://scrumguides.org/scrum-guide.html>

Kanban University (2022) The Official Guide to Kanban Method V2 <https://kanban.university/kanban-guide/#download>

Mike Cohn (2009) -Succeeding with Agile: Software Development Using Scrum Addison-Wesley Professional

Mike Cohn (2005)- Agile Estimating and Planning- Prentice Hall

Don McGreal & Ralph Jocham(2018)- Professional Product Owner, The: Leveraging Scrum as a Competitive Advantage - Addison-Wesley Professional

Eric Ries (2011) -The Lean Startup: How Constant Innovation Creates Radically Successful Businesses- Portfolio Penguin

David J. Anderson (2010) Kanban: Successful Evolutionary Change for Your Technology Business: Successful Evolutionary Change for your Technology Business: Successful Evolutionary Change for your Technology Business. Blue Hole Press.

Jeff Sutherland (2015) Scrum: The Art of Doing Twice the Work in Half the Time . Random House Business

Richard Knaster & Dean Leffingwell (2020) SAFe 5.0 Distilled: Achieving Business Agility with the Scaled Agile Framework Addison-Wesley Professional

Craig Larman & Bas Vodde(2016)Large-Scale Scrum: More with Less. Addison-Wesley Professional

Craig Larman & Bas Vodde(2008) Scaling Lean & Agile Development Thinking and Organizational Tools for Large-Scale Scrum. Addison-Wesley Professional

Lysa Adkins(2010) - Coaching Agile Teams: A Companion for ScrumMasters, Agile Coaches, and Project Managers in Transition . Addison-Wesley Professional

John P.Kotter (2012)- Leading Change- Harvard Business Review Press

Jeff Gothelf(2016) - Lean UX- O'Reilly

Matthew Skelton & Manuel Pais(2019)- Team Topologies: Organizing Business and Technology Teams for Fast Flow - IT Revolution Press

Gene Kim & Kevin Behr (2018)- Phoenix Project: A Novel About It, Devops, And Helping Your Business Win -Trade Select

Tom DeMarco & Tim Lister (2016)- Peopleware: Productive Projects and Teams -Addison Wesley

Cloud Computing & DevOps

Unit Reference Number	K/650/7997
Unit Title	Cloud Computing & DevOps
Unit Level	7
Number of Credits	20
Total Qualification Time (TQT)	200
Guided Learning Hours (GLH)	100
Mandatory / Optional	Mandatory
Sector Subject Area (SSA)	6.1 - ICT practitioners
Unit Grading type	Pass / Fail

Unit Aims

The aim of this unit is to equip students with the knowledge and skills required to work with cloud computing and Devops. This includes learning about various cloud computing models such IaaS, PaaS and SaaS and complete knowledge of different services hosted in cloud computing. Students will learn about the key Devops principles and practises including continuous integration (CI) and continuous delivery (CD) automation with various cutting-edge technologies with practical experience. Upon completion of this course, students should have solid understanding of the role, benefits, and challenges of cloud computing and Devops, able to apply these principles and practices in real world scenarios.

Learning Outcomes, Assessment Criteria, and Indicative Content

Learning Outcomes – the learner will:	Assessment Criteria – the learner can:	Indicative content
1. Be able to understand and implement continuous integration and continuous delivery process and techniques	1.1 Define and integrate the version control process with CI/CD system like Jenkins. 1.2 Demonstrate git branching and review the code.	<ul style="list-style-type: none"> Understand the workflow of Secure software development life cycle (SDLC) Analyse different Source Code Management techniques and secure coding practices.

	<p>1.3 Design and implement CI/CD automation stages like building, testing and deploying the software to reduce the time/effort perform the tasks manually.</p> <p>1.4 Deploy the build code into cloud services.</p> <p>1.5 Be able to automate the manual jobs and process</p>	<ul style="list-style-type: none"> • Able to review the pull request for code review. • Understand and scan the source code for any vulnerabilities. • Understand the benefits of CI/CD and how it works. • Develop a plan to implement CI/CD pipelines with different tools Jenkins, Gitlab • Recommend different use cases of provisioning infrastructure with CI pipelines. • Evaluate the risks in automation by integrating security tools within the CI pipelines.
<p>2. Be able to understand, design and implement distributed systems on cloud.</p>	<p>2.1 Implement scalable solutions with AWS cloud services like ec2, s3, databases etc...</p> <p>2.2 Implement reliable and secure cloud services.</p> <p>2.3 Understand and implement cloud security architecture like VPC, Subnets, Nacls, Security groups etc.</p> <p>2.4 Understand and provision cloud infrastructure services using IaaC tools like Terraform.</p> <p>2.5 Demonstrate Linux administration</p>	<ul style="list-style-type: none"> • Understand the definition, benefits and how cloud computing work. • Understand the cloud computing models including IaaS, PaaS, and SaaS with popular public cloud service provider such as AWS. • Differences between data center, cloud, and hybrid cloud solutions • Evaluate the effectiveness of cloud networking and security best practices. • Demonstrate access management to the public cloud Demonstrate to provision the cloud services as IaaC using Terraform. • Demonstrate ability to implement different cloud services and its benefits.

<p>3. Be able to understand docker container and implement container orchestration using Kubernetes</p>	<p>3.1 Understand the fundamental of container and install docker.</p> <p>3.2 Able to build docker images from scratch and maintain the vulnerabilities using the scanning tools.</p> <p>3.3 Proficiently install Kubernetes in the cloud as-well as in the local systems.</p> <p>3.4 Understand and apply container management with scalable, high available, secure, automated, cost-effective techniques using Kubernetes.</p> <p>3.5 Understand different Kubernetes services and patching in cloud.</p>	<ul style="list-style-type: none"> • Certified as AWS Cloud Practitioner and AWS Solution architect associate. • Understand the benefits of containers and how it works. • Design, write Docker files, build docker images and manage Docker containers. • Understand the benefits of container orchestration using Kubernetes. • Understand Kubernetes architecture including using master nodes and interaction with Kubernetes API. • Demonstrate installation on Kubernetes cluster as a cloud service and in the local machine and manage. • Design and configure Kubernetes objects including pods, deployment, service, ingress, and network policies. • Demonstrate deploying an application with a docker image, securing, and accessing the application in the web browser like real life application.
<p>4. Be able to use tools and standards for Linux system fundamentals, command line interface, user/group management.</p>	<p>4.1 Able to provision Linux virtual machine in cloud.</p> <p>4.2 Demonstrate user management on a Linux platform and Linux administration.</p> <p>4.3 Proficiently demonstrate the Linux system troubleshooting.</p> <p>4.4 Able to write shell scripts for automation.</p>	<ul style="list-style-type: none"> • Understand the basics of Linux fundamentals. • Demonstrate Linux Administration by created directories, user management and upgrading the servers. • Red Hat Certified System Administrator Framework

Assessment

To achieve a 'pass' for this unit, learners must provide evidence to demonstrate that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria.

Learning Outcomes to be met	Assessment criteria to be covered	Type of assessment	Word count (approx. length)
LO1-LO4	All AC's under LO1 , LO2, LO3 and Lo4	1x Report 1x Project	2500-3500 N/A

Indicative Reading List

Jon Loeliger(2022). *Version Control with Git*, 3rd Edition. O'Reilly.

Trent R. Hein, Evi Nemeth, Garth Snyder, Ben Whaley, Dan Mackin(2017). *UNIX and Linux System Administration Handbook, 5th Edition*. Addison-Wesley Professional

Zaigham Mahmood, Ricardo Puttini, Thomas Erl (2013). *Cloud Computing: Concepts, Technology & Architecture*. Pearson

David Farley, Jez Humble (2010). *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*. Addison-Wesley Professional

Mikael Krief(2020). *Terraform Cookbook: Efficiently define, launch, and manage Infrastructure as Code across various cloud platforms*. Packt Publishing

Sean P. Kane, Karl Matthias (2018). *Docker: Up & Running, 2nd Edition*. O'Reilly Media, Inc

Justin Domingus, John Arundel (2022). *Cloud Native DevOps with Kubernetes, 2nd Edition*. O'Reilly Media, Inc.

Database & SQL Programming

Unit Reference Number	L/650/7998
Unit Title	Database & SQL Programming
Unit Level	7
Number of Credits	20
Total Qualification Time (TQT)	200
Guided Learning Hours (GLH)	100
Mandatory / Optional	Mandatory
Sector Subject Area (SSA)	6.1 - ICT practitioners
Unit Grading type	Pass / Fail

Unit Aims

The aim of this unit is to develop learner's ability to understand the shortfalls of traditional storage systems and how modern relational database systems overcomes the challenges. Learner will also develop the programming skills required to communicate with database systems for accessing, transforming, and persisting the data using Structured Query Language.

Learning Outcomes, Assessment Criteria, and Indicative Content

Learning Outcomes – the learner will:	Assessment Criteria – the learner can:	Indicative content
1. Understand data and its storage importance, traditional storage systems and its challenges, understand importance of modern relational database and management systems (RDBMS)	<p>1.1 Analyse and recognise the challenge in reading and writing data from/into traditional storage systems and identify the advantages of the modern database systems compared to traditional ones.</p> <p>1.2 Differentiate logical and physical data structures in relational database management systems.</p>	<ul style="list-style-type: none"> • Introduction to data, types of data, storage, storage types, challenges with the traditional storage practices • Importance of a relational database and management system, client and server architecture, relational database management system vendors/implementers in the market

	<p>1.3 Describe what is data modelling and types of data modelling.</p> <p>1.4 Define significance of metadata and its constituents.</p>	<ul style="list-style-type: none"> • RDBMS logical data structure (database, schema, and table) and physical data structure • Basics of data modelling concepts - conceptual, logical, and business model • Metadata and its constituents
<p>2. Understand the relational database installation and configuration, role of client and DBMS server. Understand basics of SQL programming language to communicate with relational database and management systems (RDBMS), understand relational table structures and SQL Data Definition Language (DDL)</p>	<p>2.1 Demonstrate installation and configuration of a relational database client on a local desktop machine.</p> <p>2.2 Differentiate the roles of client and DBMS server.</p> <p>2.3 Describe relational table anatomy and types of table relations.</p> <p>2.4 Identify various types of SQL commands/queries used for different purposes.</p> <p>2.5 Analyse and develop SQL queries to define, create and drop relational database table structures</p>	<ul style="list-style-type: none"> • Relational database client installation and configuration • DBMS client and server architecture • Relational table anatomy - columns, rows and datatypes, representing empty data. Types of table relations – One to One, One to Many or Many to One, Many to Many, table relation using primary and foreign keys • Introduction to Structured Query Language (SQL) • Types of SQL Queries – Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL), Data Query Language (DQL), Transaction Control Language (TCL) • Commands to define/re-define relational tables and destroy/drop the existing relational tables.
<p>3. Understand various data manipulation techniques, queries to manipulate data - SQL data manipulation language (DML), understand database transactions and use of SQL Transaction Control Language (TCL)</p>	<p>3.1 Analyse and write SQL queries to manipulate data in database tables and identify difference between data deletion and truncation.</p> <p>3.2 Develop SQL queries to selectively delete, update and merge data in relational tables.</p>	<ul style="list-style-type: none"> • Understand the necessity of data manipulation and various methods to manipulate the data. • Commands for data insertion, deletion, replacement, truncation. Internals of data deletion and truncation

	<p>3.3 Analyse and understand the importance of database transactions.</p> <p>3.4 Identify the scenarios when to fire commit/rollback commands and write SQL queries to perform data transactions</p>	<ul style="list-style-type: none"> • Selective data deletes vs full data deletes, selective data updates and deletes in a single step using upsert/merge operation. • Introduction to database transactions and its application in real-world scenarios • SQL queries to control database transactions using commands - commit, rollback and save point
<p>4. Understand data retrieval techniques, data transformations, SQL commands to query database (DQL), understand data summarisation and its support in SQL program</p>	<p>4.1 Develop writing SQL queries for various real-world problem scenarios to view the actual data persisted in tables and to view the transformed table data by applying operations like arithmetic, logical, filter, ordering, joins, set operations etc.</p> <p>4.2 Assess the quality of a given raw data and develop SQL queries to cleanse and improve the quality of data.</p> <p>4.3 Analyse the requirements for data copying or movement and develop a set of SQL queries to copy/move the data from one form to other and from one location to other.</p> <p>4.4 Analyse and solve real-world scenarios using grouping and group filtering operations.</p>	<ul style="list-style-type: none"> • Data searching techniques, select operation - select and its different clauses, retrieving subset of rows vs all rows, logical & arithmetic operators, types of data filters, data cleansing techniques - finding null data, transforming null data with real values, sorting data, challenge to combine similar data sets, Introduction to sets and operations, challenge to combine different data sets, introduction to joins and types of joins, issues with cartesian joins, solving the challenge of combining different datasets, introduction to complex queries. • Importance of data indexing in tables • Data definition and data copying and moving techniques. • Introduction to data summarization and its importance, grouping data, operations on data groups
<p>5. Understand how to query metadata, understand significance of data quality and data auditing, RDBMS techniques to enforce data quality and data auditing.</p>	<p>5.1 Describe what is metadata and its role in database system.</p>	<ul style="list-style-type: none"> • Introduction to data integrity & quality and its importance, enforce data integrity/quality using database keys & constraints in tables.

	<p>5.2 Define SQL queries to query database metadata to pull table structure definition.</p> <p>5.3 Develop SQL queries to enforce referential integrity and data quality rules like no duplicates, no empty data etc.</p> <p>5.4 Develop SQL triggers to audit change in table data</p>	<ul style="list-style-type: none"> • Understand various table data constraints – In Value, Not Null, Unique values, Referential. • Introduction to data auditing and its importance, implicit vs explicit data auditing techniques • Create and apply triggers on tables for auditing data changes
<p>6. Understand the importance of data access control, security of sensitive data and SQL commands to enforce data access control and security</p>	<p>6.1 Describe few examples for sensitive data and explain the significance of securing sensitive data.</p> <p>6.2 Define what are different user privileges – object and system privileges and roles in database systems.</p> <p>6.3 Develop SQL queries to control and secure the data access to certain users/roles.</p> <p>6.4 Develop SQL queries to define and amend the database view structures</p>	<ul style="list-style-type: none"> • Introduction to data security and its importance, challenges in securing the sensitive & valuable data. • Understanding types of user privileges – Object and System privileges and roles • SQL queries to create and manage users and roles • SQL queries to control and secure data access to roles or users • Significance of database views, use of views to control access to data to different users or group of users

Assessment

To achieve a 'pass' for this unit, learners must provide evidence to demonstrate that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria.

Learning Outcomes to be met	Assessment criteria to be covered	Type of assessment	Word count (approx. length)
LO1	All ACs under LOs 1	Essay	2000 words
LO2	ACs 2.1	Practical Assessment	N/A
LO2	ACs 2.2, 2.3, 2.4, 2.5	Essay	600 words
LO3	All ACs under LO 3	Project	N/A
LO4	All ACs under LO 4	Project	N/A
LO5	All ACs under LO5	Essay	200 words
LO5	All ACs under LO5	Practical Assessment	N/A
LO6	All ACs under LO6	Essay	200 words
LO6	All ACs under LO6	Practical Assessment	N/A

Indicative Reading List

Ramez Elmasri & Shamkant B. Navathe, Fundamentals of database systems, Pearson, 7th edition, 2021

Alan Beaulieu, Learning SQL, O'Reilly Media, Inc., 3rd edition, 2020

Thomas M. Connolly & Carolyn E. Begg, Database systems: a practical approach to design, implementation and management, Pearson, 6th edition, 2015

Chris Ruel, Michael Wessler, Oracle 12c For Dummies, John Wiley & Sons Inc, 1st edition, 2013

Bill Karwin, SQL Antipatterns: Avoiding the Pitfalls of Database Programming, Pragmatic Bookshelf, 1st edition, 2017

Anthony Molinaro, SQL Cookbook, O'Reilly Media, Inc., 1st edition, 200

Web Designing

Unit Reference Number	M/650/7999
Unit Title	Web Designing
Unit Level	7
Number of Credits	20
Total Qualification Time (TQT)	200
Guided Learning Hours (GLH)	100
Mandatory / Optional	Mandatory
Sector Subject Area (SSA)	6.1 - ICT practitioners
Unit Grading type	Pass / Fail

Unit Aims

This unit focuses to equip learners with the necessary knowledge and skills to design and implement basic web pages using HTML, CSS, and JavaScript. By the end of this course, learners should be able to design and implement web site and web pages using HTML and CSS, integrate CSS styles with HTML tags, understand the flow of overrides among CSS rules, and implement tables and lists in HTML. Learners will also gain an understanding of user input properties of HTML, Document Object Model (DOM) management, and the basics of JavaScript and its uses in web design.

Learning Outcomes, Assessment Criteria, and Indicative Content

Learning Outcomes – the learner will:	Assessment Criteria – the learner can:	Indicative content
1. Understand what the building blocks of a web page and basics needed to design it	1.1 Design and implement a basic webpage 1.2 Understand the building blocks of a html file. 1.3 Design and implement basic html tags	HTML, CSS and JS <head />, <body />, <footer />, <p />, , <a />

2. Understand the basics of CSS and different styles and how to integrate CSS with Html	2.1 Design and implement css styles. 2.2 Design and implement CSS styles on Html tags. Integrating CSS into Html	Color, font, position, background-color, height, width etc CSS from a different file, import into html, inline styling
3. Understand CSS rules and their order	3.1 Understand and implement CSS rules and to learn the flow of overrides among rules. 3.2 Advanced CSS concepts using all the above rule overrides, tags	Tag, class, id, inline Margins, padding, layouts, transparency etc
4. Understand and implement tables and lists in HTML with examples using CSS	4.1 HTML tables and lists with CSS props 4.2 Advanced CSS concepts using all the above rule overrides, tags 4.3 Understand and implement tables and lists in HTML with examples using CSS	<table >, , <ui>
5. Understand user input properties of HTML	5.1 Understand and implement HTML input fields with examples	Textfield, checkbox, radiobutton, image, video embed with CSS props
6. Introduction to DOM	6.1 Understand and implement DOM management and changing DOM	HTML DOM basic concepts
7. Introduction to Javascript and its uses	7.1 Introduction to JS, integration with HTML and CSS	Js, integrate with file and js inside html
8. Demonstrate basic JS properties and functions	8.1 Design and implement basic JS code, functions, and its dynamic nature	Alerts, document.getelementbyid, class etc

Assessment

To achieve a 'pass' for this unit, learners must provide evidence to demonstrate that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria.

Learning Outcomes to be met	Assessment criteria to be covered	Type of assessment	Word count (approx. length)
LO1-LO8	ACs under LO1, LO2, LO3, LO4, LO5, LO6, LO7 and LO8	Practical Assignment projects	<ul style="list-style-type: none"> Code assignments to be submitted which can be executed on evaluator's machine. No word count- It will be measured by the quality of the code written and executed efficiently without errors ideally or with any minimal code config tweaks.

Indicative Reading List

DuRocher, D. (2021). HTML & CSS QuickStart Guide: The Simplified Beginners Guide to Developing a Strong Coding Foundation, Building Responsive Websites, and Mastering the Fundamentals of Modern Web Design. ClydeBank Media LLC.

Duckett, J., & Schlüter, J. (2011). HTML and CSS. Wiley.

Robbins, J. N. (2012). Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics. " O'Reilly Media, Inc."

Meloni, J. C., & Kyrnin, J. (2018). HTML, CSS, and JavaScript All in One, Sams Teach Yourself. Sams Publishing.

Frain, B. (2020). Responsive Web Design with HTML5 and CSS: Develop future-proof responsive websites using the latest HTML5 and CSS techniques. Packt Publishing Ltd.

IMPORTANT NOTE

Whilst we make every effort to keep the information contained in programme specification up to date, some changes to procedures, regulations, fees matter, timetables, etc may occur during the course of your studies. You should, therefore, recognise that this booklet serves only as a useful guide to your learning experience.

For updated information please visit our website www.othm.org.uk