CA3: Course Specification

Along with the Module Specifications, the Course Specification form the definitive description of any qualification awarded by the University. The Academic Quality and Standards Office (AQSO) is responsible for maintaining up to date records of all definitive documents (course and module specifications). **Any** amendments made to the Course Specification must be submitted to the AQSO via the formal Amendments process outlined in Section 4 of the Academic Quality and Standards Handbook – Course Amendments.

PART A: About the Course (See Part B for other key information)

1. **Qualification** (award and title):

   BSc (Hons) Computer Science

2. **Date of Approval** (month and year)**:**

   31/07/2020

3. **Delivery Partners, Sites and Recognition:** *who delivers this course, where? Is it accredited by any professional bodies?*

Campuses/Partners	**Recognised/accredited by**
UWL RAK Branch Campus	

4. **Course Description:** *a short descriptive statement used for publicity* (max. 250 words):

The BSc (Hons) Computer Science course provides a professional education in the field of Computing with well-balanced content covering the theoretical and the practical aspects of Computing.
The focus of the course is the design and development of computing and information systems to provide optimised solutions based on specific requirements. The course is designed to meet industry needs and job market demands and provide students with the essential skills required to pursue a career in the computing industry.
The Course balances practical skills with theory, to allow the student to develop an understanding of software and systems, coupled with the knowledge and ability to design, develop, construct and manage computer based solutions.

Working in partnership with leading industry such as Amazon, Enterprise/BCS, Fujitsu, Innovation - Heathrow, ORACLE, BIRST, SEGA, and the course integrates the latest advances and technologies and solutions into the curriculum and provides best-practice learning experiences to students. Computing Industry Consultative Committee (CICC) meetings are held annually, for this purpose.

5. Course Structure Diagram: *a visual overview of the programme of study*

Full-time September start. All modules are 20 credits except for L6 project which is 40 credits.

LEVEL 4

Semester 1	**Semester 2**
Core - Computer Architecture (CA) **CP4CS53E**	**Core -** Algorithms and Data Types (ADT) **CP4CS64E**
Core **-** Programming (PROG) **CP4CS61E**	**Core -** Information Systems and Databases (ISDB) **CP4CS54E**
Core - Mathematics for Computing (MATH) **CP4CS63E**	**Core -** Data Communications (DC) **CP4CS58E**

LEVEL 5

Semester 1	**Semester 2**
Core – Theory of Computation (TC) **CP5CS04E**	**Core -** Functional Programming (FP) **CP5CS65E**
Core - Mobile Web App Development (MWAD) **CP5CS93E**	**Core -** Group Research Project (GPRJ) **CP5CS05E**
Core - Artificial Intelligence (AI) **CP5CS06E**	**Core -** Human Centred Computing (HCC) **CP5CS73E**

LEVEL 6

Semester 1	**Semester 2**
Core - Applied Software Engineering (ASE) **CP6CS19E** **Core -** Machine Learning (ML) **CP6CS57E**	Choose **two** from: **Option –** Cyber Security (CYS) **CP6CS59E**

	Option - Distributed Systems (DS) **CP6CS60E** **Option -** Databases and Analytics (DBA) **CP6CS50E** **Option –** IT Consultancy (ITC) **CP6CS58E**
Core - Project (PRJ, 40 credits – year long) **CP6CS46E**	

Full-time January starters will join the September cohort and start with semester 2.

6. Course Aims and Content by Level: *what is this course all about and how does the programme of study build and develop over time?*

This course aims to:

- Meet the employer/industry needs for well-trained, competent adaptable computing professionals who are able to effectively develop, implement and/or manage ICT systems.
- Provide students with a blend of key generic underpinning knowledge in the Computing and IS field including professional and ethical issues in ICT development and implementation
- Develop students' ability to design, develop and construct computer-based solutions.
- Develop an understanding and appreciation of IT technologies applied to client needs, objectives, development, operations and maintenance.
- Empower students from a variety of academic and occupational backgrounds to fulfil their own academic potential.
- Provide students with a learning environment that will develop interpersonal skills, raise self-awareness, and encourage personal and career growth as well as the ability to gain from lifelong learning.
- Provide an academic base for further study, research and / or other professional / vocational training.

In addition to the above the BSc Computer Science programme aims to:

- Develop students' understanding of the theoretical issues that underpin a range of software development paradigms
- Develop students' ability to relate knowledge and techniques concerned with the design and exploitation of computer technology to the development of computing systems.
- Develop students' ability to design, develop, construct and manage computer-based solutions for applications that involve the consideration of complex technical alternatives particularly in distributed environments.

LEVEL 4 At level 4, the course comprises of six modules covering the fundamentals of computing laying strong foundations for the rest of the course. The course aims to give the essential introductory topics in programming and algorithms, databases, mathematics, data communications and computer architectures. In addition, this level aims to develop more generic skills such as awareness of academic terms, approaches and methods and confidence in using them.	**Reference Points**: • The Frameworks for Higher Education Qualifications of UK Degree-Awarding Bodies • Subject Benchmark Statements • Characteristics Statements • All available at: http://qaa.ac.uk/quality-code/the-existing-uk-quality-code# Apprenticeship standards Available at: https://www.gov.uk/topic/further-education-skills/apprenticeships/latest • PSRB standards (as appropriate)
LEVEL 5 At this level, the course aims to build on level 4 foundations, with modules relating more specifically to computer science, and to developing skills in analysis, design and development. The course covers the necessary concepts and topics in Computer Science including the Theory of Computation and Algorithm Complexities, Artificial Intelligence, Functional Programming, and Human Centred Computing. As part of the Group Research Project module students will focus on research methodologies. The module will also prepare students for their final year project. At level 5 the course aims to develop skills to distinguish and evaluate different theoretical frameworks to demonstrate knowledge and understanding in the computing industry.	
LEVEL 6 The final year of the course, will focus on a range of specialised and advanced computing topics, and develop students' skills in research and critical reflection. Students will be able to choose some optional modules which will allow them to shape their degree to their interests and ambitions. As part of the final year individual major project students will design and develop an artefact. Topic areas covered include Machine Learning, Software Engineering, Cyber Security, Databases and Analytics, IT Consultancy, and Distributed Systems and Cloud Computing. Level 6 aims to develop skills to exhibit in-depth knowledge of the computing industry and critically evaluate own skills and knowledge in this context for future career. Students to gain knowledge and skills to select and apply a range of robust procedures and	

techniques to model, design, and implement computer oriented solutions to practical problems. BSc (Hons) Computer Science with Industrial Placement aims to provide skills to demonstrate the understanding of group work and processes in a work environment; and ability to critically evaluate the practical experiences and knowledge gained.	

7. Course Contact Hours: *how much time should I commit to this course?*

Learning hours are determined by credits. One credit is worth 10 learning hours, so a 20 credit module is 200 learning hours, a 30 credit module is 300 hours etc. This is the amount of time you should be prepared to commit to each module.

Learning hours are divided into: taught or 'contact' hours, i.e. the amount of time students spend in contact with academic staff, whether through face-to-face classes or online learning; and independent study, i.e. the amount of time students are expected to spend on their own study and assessment preparation. Some kinds of learning mix contact time and independent study, for instance presentations or workshops by invited experts, or sessions where you are working in groups on a project but can call on academic staff for advice or feedback on your work so far. You also have one-to-one time with academic staff in personal tutorials.

8. Academic Staff:

Staff employed on UWL Academic contracts at Lecturer level have a minimum requirement to have a higher degree in an appropriate discipline and a teaching qualification (PG Cert or Academic Professional Apprenticeship) and/or HEA Fellowship. Senior Lecturers have a similar minimum level and in addition should either hold a PhD or be registered on a doctorate programme. Associate Professor and Professor levels are required to have a PhD. All staff on Academic contracts at UWL are required to undertake teaching. Hourly paid teaching staff are also used across UWL and these colleagues bring a wide range of professional, specialist and industry experience to the teaching of our students. The University has made an explicit commitment to supporting the professional development of its staff through the programme of continuing professional development (CPD) managed and delivered by the ExPERT Academy.

9. Course Learning Outcomes: *what can I expect to achieve on this course?*

	Level 4	Level 5	Level 6
Knowledge and understanding	A4.1 - Demonstrate key factual and conceptual knowledge in the field of computing. A4.2 - Apply concepts and techniques for developing and architecting software and hardware**.** A4.3 - Apply given techniques to analyse and test data with an appreciation of fundamental models and concepts. A4.4 - Show knowledge and understanding of fundamental models and concepts underpinning computing and software programming and computer architectures. A4.5 – Apply and evaluate the principles of algorithms and data structures that underpin software development	A5.1 – Select and apply a range of procedures and techniques to model, design, and implement computer oriented solutions to practical problems A5.2 – Distinguish and evaluate different theoretical frameworks to demonstrate knowledge and understanding in the computing industry A5.3 – Demonstrate and construct different paradigms for software development A5.4 – Implement the principles of object oriented programming and design patterns and that underpin software development and ubiquitous computing	A6.1 – Exhibit in-depth knowledge of the computing industry and critically evaluate own skills and knowledge in this context for future career A6.2 – After critical analysis, select and apply a range of robust procedures and techniques to model, design, and implement computer oriented solutions to practical problems A6.3 – Demonstrate the understanding of concepts and infrastructure underpinning security and artificial systems. A6.4 – Demonstrate the deployment of enterprise solutions A6.5 – Specify, design and critically evaluate different programming paradigms for appropriate contextual deployment and software engineering.
	ALL = CA, MATH, PROG, DC, ADT, ISDB A4.1 – CA, MATH, PROG, DC, ADT, ISDB A4.2 – CA, PROG, DC, ADT, ISDB A4.3 – ADT, ISDB, MATH	ALL = FP, TC, AI, MWAD, HCC, GPRJ A5.1 – FP, TC, AI, MWAD, HCC, GPRJ A5.2 – FP, TC, AI, MWAD, HCC, GPRJ A5.3 – FP, TC, AI, MWAD, HCC, GPRJ A5.4 – MWAD	ALL = ASE, ML, DBA, CYS, DS, PRJ, ITC A6.1 – DS, DBA, CYS, ML, ITC, ASE, PRJ A6.2 – DS, DBA, CYS, ML, ASE, PRJ A6.3 – CYS, ML

	A4.4 – CA, PROG, DC, ADT, ISDB A4.5 – PROG, ADT, ISDB	A5.5 – PLC	A6.4 – ASE, DS, PRJ A6.5 – DS, CYS, ML, ASE, PRJ, ITC
Intellectual / Cognitive skills	B4.1 - Manipulate and use various tools and techniques for architecting given requirements B4.2 - Analyse a simple system in terms of given principles B4.3 - Investigate a problem, formulate solutions with justification of conclusions B4.4 - Specify a computing system in terms of basic hardware and software requirements	B5.1 – Evaluate alternative solutions and apply appropriate criteria in a variety of contexts B5.2 – Design, by the selection and application of appropriate techniques, a computer artefact B5.3 – Assess alternative approaches to the programming of software solutions B5.4 – Evaluate and select appropriate research methods in order to formulate a credible research proposal. B5.5 – Critically apply Object Oriented theory to the practical design of elements of distributed computing and services	B6.1 Employ the cognitive skills of critical thinking, analysis and synthesis including the capability to identify assumptions, evaluate statements in terms of evidence, to identify implicit values, to define terms adequately and to generalise appropriately. B6.2 – Manifest a critical awareness of current ethical, legal and quality frameworks that apply to the development of systems by incorporation of these concepts across a range of business issues. B6.3 – Devise and apply concepts for Machine Learning and artificial intelligence. B6.4 – Design software systems based on best practices for component based enterprise systems. B6.5 – Critically analyse and apply concepts in software engineering and computer security to the practical design of elements of business solutions.
	B4.1 – CA, ISDB, DC B4.2 - CA, MATH, PROG, DC, ADT, ISDB B4.3 - CA, MATH, PROG, DC, ADT, ISDB B4.4 - CA, PROG, DC, ADT, ISDB	B5.1 – FP, TC, AI, MWAD, HCC, GPRJ B5.2 – FP, TC, AI, MWAD, HCC, GPRJ B5.3 – FP, TC, AI, MWAD, GPRJ B5.4 – GPRJ B5.5 – MWAD	ALL = ASE, ML, DBA, CYS, DS, PRJ, ITC B6.1 – ASE, ML, DBA, CYS, DS, PRJ, ITC B6.2 – ASE, DBA, CYS, DS, PRJ B6.3 – ML, ASE, PRJ

		B5.6 – PLC	B6.4 – ASE, DS, DBA B6.5 – ASE, DBA, CYS
Subject practical skills	C4.1 - Design, implement and test a simple computer based solution and present results appropriately C4.2 - Use appropriate techniques to analyse model, and test data, and to produce structured reports C4.3 – Apply knowledge on computer software and systems architectures C4.4 - Use basic tools to analyse and model software programs	C5.1 – Develop a software artefact with an appropriate design for best practices as part of software services C5.2 – Analyse the technologies used to support requirements and construct software related to specified requirements C5.3 – Develop skills in the development of systems using object oriented and functional programming techniques C5.4 – Implement procedures and concepts for the development of software systems for distributed systems C5.5 – Critically evaluate and select recommended needed technologies needed to develop software artefacts.	C6.1 – Plan, implement, monitor and complete a significant independent computing practical project under only limited guidance from academic supervisor C6.2 – Account for their professional conduct, particularly with respect to current ethical, legal and quality frameworks that apply within the computing industry C6.3 – Critically evaluate and recommend the technologies needed to develop secure artefacts C6.4 – Analyse requirements for, and evaluate and develop solutions to knowledge based systems C6.5 – Critically evaluate applications of intelligent systems
	C4.1 - CA, MATH, PROG, DC, ADT, ISDB C4.2 - CA, MATH, PROG, ADT, ISDB C4.3 - CA, MATH, PROG, DC, ADT, ISDB C4.4 - CA, MATH, PROG, DC, ADT, ISDB	C5.1 – MWAD, GPRJ C5.2 – FP, MWAD, GPRJ C5.3 – FP, MWAD, C5.4 – MWAD C5.5 – FP, FC, AI, MWAD, GPRJ C5.6 – PLC	C6.1 – DS, CYS, ML, ASE, DBA, PRJ C6.2 – ASE, ICT, PRJ C6.3 – CYS,ASE,PRJ C6.4 – DBA, CYS, ML, ASE, DS, PRJ C6.4 – DBA, CYS, ML, ASE, DS, PRJ
Transferable skills	D4.1 - Adopt a flexible, adaptable and professional attitude towards learning and the development of skills	D5.1 – Interact effectively in a variety of different learning groups in both real and virtual contexts	D6.1 – Exercise initiative and personal responsibility for the management of own learning

	D4.1 - Manage time and resources to meet deadlines D4.3 - Present information in effectively in a clear and concise manner using a variety of formats. D4.4 - Recognise their own, and others, strengths and weaknesses	D5.2 – Efficiently apply appropriate technology to facilitate and manage own learning D5.3 – Communicate in a clear and concise manner using appropriate technical approach D5.4 – Work effectively with others	D6.2 – Demonstrate the ability to manage independent learning necessary for continued professional development D6.3 – Reflect on personal attainment and appropriately apply learning experiences to inform and enhance subsequent professional practice.
	D4.1 - CA, MATH, PROG, DC, ADT, ISDB D4.2 - CA, MATH, PROG, DC, ADT, ISDB D4.3 - CA, MATH, PROG, DC, ADT, ISDB D4.4 - CA, MATH, PROG, DC, ADT, ISDB	D5.1 – FP, TC, AI, MWAD, HCC, GPRJ D5.2 – FP, TC, AI, MWAD, HCC, GPRJ D5.3 – FP, TC, AI, MWAD, HCC, GPRJ D5.4 – GPRJ D5.5 – PLC	D6.1 – ML,ASE,DBA,DS, CYS, ICT, PRJ D6.2 – ML,ASE,DBA,DS, CYB, PRJ D6.3 – ML,ASE,DBA,DS, CYS, ICT, PRJ

10. Learning, Teaching and Assessment Strategies: *how will I learn, how will my learning be assessed, and why are these the most appropriate methods?*

Teaching and Learning Approaches

Design of the modules making up the course is assessment driven, in that they are designed from learning outcomes and assessment upwards. Students on the course will be encouraged to take responsibility for their own learning whilst still being supported by subject tutors, and to immerse themselves in the subject area.
There is an appropriate balance of theory and practice, and in order to be successful students will need to demonstrate appropriate levels of analytical, critical and reflective skills alongside a professional level of practical skills and knowledge. Problem-based learning approach is used as a means to link theory and practice.
In addition, teaching and learning on the course is underpinned by the research and development activities of the course team.
Teaching methods include lectures, whole group information-giving sessions, workshops, tutorials, practical work, and blended e-learning and group critiques.
The practical work, as part of the course, will reflect real-world techniques that practitioners would encounter in industry.

Lectures are used to introduce new material impart information and launch and guide individual study activities. Lectures are used on all modules on the course.
Seminars are plenary sessions that aim to promote student centred learning via group discussion with feedback aimed at developing self-confidence in communication and presentation. They are also designed to encourage self-criticism and the enhancement of interpersonal skills. Seminars are used on most modules.
Labs seek to build confidence in the use of tools and technologies and apply them to real or simulated problems. Frequently they allow students to engage in practical activities individual and in teamwork.
The use of virtual learning (Blackboard) allows students to extend their studies beyond the confines of the classroom. Lecture notes, video links, supplementary reading and assessment guidelines are incorporated on Blackboard.
Guest speakers seek to provide students with the opportunity to experience and debate current industrial practices. Students also have at least one visit to an external organisation to learn about contemporary projects within the field.
The use of professional tools and technologies seek to provide the knowledge and skills required in computing industry. These tools are thought on most modules and available in all computer labs on the campus including the library lab rooms.

As part of the course students will be able to access a variety of learning materials via the University's e-learning system, Blackboard. Tutors will encourage students to use and access it for such purposes as:

- Obtaining the learning materials such as lecture notes, supplementary videos and reading lists.

- General or specific notices or announcement relating to the course as a whole or to a specific group. Here posts will be made available via the course community site and individual posts by module tutors.
- Submitting assignments and received feedback from tutors.
- Networking with tutors and other students via the use of tools such as e-discussions, e-tutorials and e-forum.

Assessment Approaches

Coursework based assessment is designed, where possible to simulate the variety of tasks that graduates from the course may encounter in relevant employment. Where necessary other academic assessment devices, such as a formal examination are also used.
During the foundation year, and at level 4, much of the assessment is by means of portfolio development which may take the form of class tests, presentations, or practical activities. As the year develops and the confidence of the student increases, the assessment will have a more conventional output e.g. formal reports and other documentation.
In subsequent years, assessment types include the following:
•Analysis and design and the production of appropriate artefacts
•Portfolio of work
•Presentations to tutors and peers
•Development of design specifications
•Critiques of own and peer work
•Major implementation project
•Examinations and class tests
Wherever possible the students will be encouraged to relate theory to practice and to reflect on practice experienced. This challenge will be addressed by developing a student-centred approach to learning and teaching. This has involved the introduction of a range of specific techniques including problem-based learning and case studies, which will inform the development of the assessment strategy. Its intension is to make the curriculum interesting and relevant to diverse groups of students. Social constructivism emphasises the importance of student learning through interaction with teachers and other students.
The assessment material for each module will be distributed to the students at the commencement of the module. This material will include a complete set of requirements and dates for completion as well as a marking scheme and associated performance criteria.
During the delivery of each module students will receive feedback on their performance on class exercises. These exercises will not normally be graded for assessment purposes. However, the work completed during these exercises may form part of the work to be handed in for the module assessment at the completion of the module.
There are a variety of mechanisms by which feedback on assessments is provided to students. Where assessments are portfolio based, formative assessment on elements of draft submissions is provided during class or via email.
Additionally, portfolio based assessments allow work to be submitted in stages where each element submitted receives summative feedback.

Student feedback will also be provided via the Blackboard E-Learning Platform. This is particularly useful where there are trends in student work and formative feedback can be provided to class groups. Formative feedback is available during dedicated class sessions and via email and pre-arranged meetings with course tutors.

Written feedback is provided on assessed work via Rubrics on Blackboard. Additional feedback is also provided in a general form as part of the module leader's report or during course committee meetings.

Level 4

Level 4 modules cover the basic theoretical concepts, programming, and Mathematics. Appropriate software tools are introduced and incorporated as part of the teaching materials to provide the skills needed to design artefacts, and solve Computing specific tasks, and for students to be able to evaluate relevant information and ideas.

Assessments are designed to simulate a variety of tasks including problem solving activities, and to apply concepts and techniques for developing and architecting software and hardware.

All of the assessment is by means of written assignment reports and portfolio development, and in-class assessment which may take the form of class tests, presentations, or practical activities. Assignments are designed with elements of programming exercises and problem solving. They provide an ability, for students, to present and evaluate results and solutions and to make judgements in accordance with the theories and concepts presented in the modules.

Level 5

Wherever possible the students will be encouraged to relate theory to practice and to reflect on practice experienced. This challenge will be addressed by developing a student-centred approach to learning and teaching. This has involved the introduction of a range of specific techniques including problem-based learning and case studies, which will inform the development of the assessment strategy. Its intention is to make the curriculum interesting and relevant to diverse groups of students.

Theoretical concepts covered in algorithm optimisation in terms of user experience and App development, best practices and design patterns for creating practical databases are examples of practical elements and experiences students gain.

As part of the Group Research Project module students design and develop an artefact, using case studies and requirements elicitation techniques and communicating and collaborating by means of appropriate software tools. They will also begin investigating a topic and produce a draft project proposal for their final year project.

Level 6

During the final year of the course, students engage on their final year-long project. Project titles and topics are discussed with supervisors, and aims and objectives are set. As part of the project work, the theory and concepts applied by designing and developing an artefact, with opportunity to exercise initiative and personal responsibility for the management of own learning.
As part of the project work, students will critically evaluate and recommend the technologies used and incorporated for the artefact.
Students taking the placement will critically evaluate the practical experiences and knowledge gained by producing a report as part of the Placement module.
Theories and concepts covered in Applied Software Engineering are case study based where the module aims to allow students to design appropriate applications with real world requirements. Models and diagrams at abstract level are implemented and software produced.

The research effort within the school contributes towards the curriculum, and at level 6, students engaged in providing solutions to given tasks, and to deploy accurately the established techniques of design and analysis. As an example by means of creating deployable distributed components and services to cloud platforms.

11. Formal and Informal Links with External Organisations/Industrial Partners:

what opportunities are there for me to interact with professional contacts?

The links and ongoing dialogues established by the team with the computing industry and other educational establishments form an essential and valued element in the design of the course material and the delivery of the course. Constant development and refinement of the modules will be a feature of the course in order to update and maintain its currency and a wholly contemporary relationship with the design industries the students aspire to enter on graduation. In addition to those links forged by staff with key industrial practitioners, students are expected as a matter of course to begin to make contact with relevant industrial links as part of the push to develop their presentation skills. A key element in this is the continuing professional practice by the delivery team which covers practical updating, individual research, theoretical enquiry in relevant areas as well as, most importantly, being creative and exhibiting practitioners in their chosen industry based and/or technical fields. This course will benefit from the work research centres and groups within the School. The School currently has a number of Research Centres and Research Groups. The school has a strong link with its industry partners through the Computing Industrial Consultative Committee panel with regular annual meetings arranged per each academic year where the content of courses and subjects are discussed. The aim is to ensure the course delivers skills required in the industry. Furthermore, guest lectures are delivered by our industry partners to present topics of current subject interest in the market.

12. Admissions Criteria: *what qualifications and experience do I need to get onto the course?*

• **UCAS Tariff (UG only)** • **Subject-specific requirements** • **Additional information**	• 112 UCAS tariff points at Level 4 • A Levels at grade B, B and C, or above or Equivalent • Higher Secondary School Leaving certificate (India) • BTEC Extended Diploma with Distinction, Merit, Merit • Access to HE Diploma You also need GCSE English and Maths (grade 9 – 4 / A* - C) or Level 2 equivalents. Mature students wishing to apply will be considered on an individual basis. Their professional/work/life experiences and their ability to engage with, and benefit from, the course will be taken into account. We also welcome applicants with no formal qualifications. These applicants will be considered on an individual basis taking into account their professional/work/life experiences and their ability to engage with, and benefit from, the course. All applicants will be interviewed and invited to discuss their portfolio of work. They will only be offered a place if they are able to demonstrate a level of intellectual curiosity and organisational ability such that they have the potential to be successful on the course.
Arrangements for Recognition of Prior Learning	
IELTS Score for International Students	*Standard L3 entry (ie, for UG Course with Foundation): 5.5 - minimum 5.5 for each of the 4 individual elements: reading, writing, speaking, listening* *Standard UG (L4 entry) : 6.0 - minimum 5.5 in each individual element* *Standard PG (L7 entry): 6.5 - minimum 5.5 in each individual element*

13. Student Support Arrangements: *what kinds of academic and pastoral support and advice are available?*

STUDENT SUPPORT

University-wide Support Services for *all* students:

- Careers and Employment Services
- Student Advice
- Information Team
- Accommodation Service
- Counselling
- Students' Union
- Mentoring

Student advice, help and support is further detailed in the Student Handbook with regard to University facilities, services and current policies

The Engagement Service is available to all UWL students working at any level and on any course. The Service offers easily accessible support so that students can reach their full potential.

Academic Support Opportunities include:

- Daily Academic Drop-in Support
- Academic Skills workshops
- Peer Mentor Pairing
- Summer Workshops

Daily Academic Drop-in Support

The Drop-in Support Sessions offer daily opportunities for students to seek guidance and academic support with no appointment required. The Drop-in Support sessions provide students with an easily available space and opportunity to seek solutions and gain personal academic advice whenever needed. Students attend to both overcome challenges and to build on their success. Many report that attending a drop-in has helped to dramatically reduce their anxiety and raise their confidence.

Academic Skills Workshops

The Academic Skills workshops are delivered throughout the year, helping students to develop skills relevant to their degree. Some aspects of a student's course may be challenging, or the student may have been out of education for some time. The workshops have been developed so that they include a theoretical element followed by an hour of supported practical study, where the theory can be applied and questions asked. Examples of workshops include: Essay Writing, Time Management and Organising Your Studies, Report Writing, How to Write Critically, Group Work and Presentation Skills.

Peer Mentoring

Peer mentoring offers students the opportunity to be paired with another student, studying within the same academic school, who can support them by sharing their own experiences of the course and UWL. Students can request a mentor and it has been shown to be one of the best ways to help students acclimatise to university life and maximise their student experience, academically and socially. Students are paired with a peer mentor who can share their experience and

provide another perspective on the School, the subject area and the course through regular meetings.
As student's progress through their course they can also volunteer to be peer mentors and support other students. Peer mentoring is highly valued by many employers and is a highly rewarding experience for all involved.

Summer Schools
The Engagement Team offer One Day Summer School opportunities to allow new and returning students to enhance their study skills and build their confidence in their ability to demonstrate these skills.

Undergraduate and taught postgraduate courses
The University-wide support framework encompasses:

- Induction
- Course Leaders
- Module Leaders
- Personal Tutors
- VLE (Blackboard)
- In-course learning skills development*
- Personal Development Planning (PDP)**

*Learning skills include critical appraisal, reflection, literature searching, information technology, peer review, group work, presentation, research, practice/professional skills, note-taking, writing skills, electronic information retrieval, communication skills and independent study at home.
**PDP has been formalised on undergraduate degree courses via the Personal Tutorial system delivered through levels 4-6. These tutorials are designed to support the development of academic skills (at level 4) employability (at level 5) and personal reflection and research enquiry linked to career options (at level 6). PDP is developed informally in other areas of learning through students' development of personal skills such as time management, leadership, and teamwork. Guest speakers and field visits provide students with networking opportunities.

14. Assessment Matrix: *a list of all the assessments on the course, along with how much they count for and where they come in the year.*

Module Title and Code	Core /Optional (write C or O)	Credit	Assessment Type (choose from the dropdown list)	Weighting (%)	Overall pass mark	Minimum percentage (PSRBs and Apprenticeships only)	Apprenticeships Only: contributes to 'End-Point Assessment' (write YES or NO)	Submission: Week Number (indicative)
Level 4								
CP4CS53E Computer Architecture	C	20	*Portfolio*	**100**	40	**30**		**Week 16**
CP4CS61E Programming	C	20	*Portfolio*	**100**	40	**30**		**Week 16**
CP4CS63E Mathematics for Computing	C	20	*Portfolio*	**100**	**40**	**30**		**Week 16**
CP4CS54E Information Systems and Databases	C	20	*Portfolio*	**100**	**40**	**30**		**Week 16**
CP4CS64E Algorithms and Data Types	C	20	*Portfolio*	**100**	**40**	**30**		**Week 16**
CP4CS58E Data Communications	C	20	*Portfolio*	**100**	**40**	**30**		**Week 16**
Level 5								
CP5CS04E	**C**	**20**	*Portfolio*	**50**	**40**	**30**		**Week 12**

Theory of Computation			*Written examination*	**50**	**40**	**30**		**Exam Week**
CP5CS93E Mobile Web App Development	**C**	**20**	*Portfolio*	**50**	**40**	**30**		**Week 14**
			Written assignment	**50**	**40**	**30**		**Week 16**
CP5CS06E Artificial Intelligence	**C**	**20**	*Written assignment*	**50**	**40**	**30**		**Week 11**
			Written examination	**50**	**40**	**30**		**Exam Week**
CP5CS65E Functional Programming	**C**	**20**	*Portfolio*	**50**	**40**	**30**		**Week 12**
			Written examination	**50**	**40**	**30**		**Exam Week**
CP5CS81E Group Research Project	**C**	**20**	*Written assignment*	**100**	**40**	**30**		**Week 14**
CP5CS73E Human Centred Computing	**C**	**20**	*Written assignment*	**100**	**40**	**30**		**Week 16**
Level 6								
CP6CS46E Project	**C**	**40**	*Written assignment*	**20**	**40**	**30**		**Semester 1, Week 14**
			Written assignment	**60**	**40**	**30**		**Semester 2, Week 14**
			Oral assignment	**20**	**40**	**30**		**Semester 2, Week 16**
CP6CS57E Machine Learning	**C**	**20**	*Written assignment*	**100**	**40**	**30**		**Week 13**
CP6CS19E Applied Software Engineering	**C**	**20**	*Written assignment*	**100**	**40**	**30**		**Week 14**

CP6CS59E Cyber Security	**O**	**20**	*Portfolio*	**100**	**40**	**30**		**Week 16**
CP6CS60E Distributed Systems	**O**	**20**	*Written assignment*	**50**	**40**	**30**		**Week 10**
			Written examination	**50**	**40**	**30**		**Exam Week**
CP6CS50E Databases and Analytics	**O**	**20**	*Written assignment*	**50**	**40**	**30**		**Week 9**
			Portfolio	**50**	**40**	**30**		**Week 14**
CP6CS58E IT Consultancy	**O**	**20**	*Written assignment*	**50**	**40**	**30**		**Week 8**
			Portfolio	**50**	**40**	**30**		**Week 14**

15. External Examiner Arrangements: *who checks the standards and quality of the course?*

External examiners are attached to all modules at level 5 and level 6 as per the university regulations. They are responsible for assessing the quality of the programme and the consistency of standards across all levels.
External examiners are selected on the basis of their subject expertise and are subject to scrutiny by a division of the University's Academic and Quality Control Department – External Examiner's Advisory Committee (EEAC). External examiners are proposed by the College and if accepted by EEAC are in position for four years.

PART B: Key Information

1.	**Awarding Institution**	University of West London
2.	**UWL School/College**	SCE
3.	**Subject Area within School/College**	
4.	**Academic Partners and type of arrangement**	UWL RAK Branch Campus
5.	**Course recognised by**	
6.	**Sites of delivery**	UWL RAK Branch Campus
7.	**Modes and duration of delivery**	Full time = 3 years - *BSc (Hons) Computer Science*
8.	**Sequencing**	*September and January start*
9.	**Final enrollable award(s)**	*BSc (Hons) Computer Science*
10.	**Level of final award**	6

11.	**Credit for final award (CATS and ECTS)**	BSc (Hons) Computer Science = 360 CATS/180 ECTS		
12.	**Exit awards and credits**	*BSc Computer Science = 360 credits at level 4, 5 and 60 at level 6* *Dip HE Computer Science = 240 credits at levels 4 and 5.* *Cert HE Computer Science = 120 credits at level 4*		
13.	**UCAS code(s) (UG programmes)**	I100, G401		
14.	**QAA Subject Benchmarking Statement**	Computing (2016) https://www.qaa.ac.uk/docs/qaa/subject-benchmark-statements/sbs-computing-16.pdf?sfvrsn=26e1f781_12 https://www.qaa.ac.uk/docs/qaa/subject-benchmark-statements/subject-benchmark-statements.pdf?sfvrsn=1656ff81_2		
15.	**Apprenticeship Standard title and code**	N/A		
16.	**Course-specific Regulations**	Standard UWL regulations apply. In addition 30% thresholds applied on assessments, as per accreditation body requirements.		
17.	**Language of study**	*English*		
18.	**Original approval Date**		**Last Revision Date**	

PART C: Record of Approved Amendments

Approved Amendments to Course Specification since original approval/last review					
Course Specification Title	**Module Level and title**	**Brief Outline of Amendment**	**Approval by School/College Quality Committee**	**Approval effective from**	**Student cohort affected**
Specify award titles/routes affected by change			*Date and meeting minute*		*e.g. students entering Level 5 from AY2018*