Course Specification

Course Summary Information		
1	**Course Title**	BSc (Hons) Software Engineering (with Foundation Year)
2	**Course Code**	US1551F
3	**Awarding Institution**	Birmingham City University
4	**Teaching Institution(s)** (if different from point 3)	N/A
5	**Professional Statutory or Regulatory Body (PSRB) accreditation** (if applicable)	Not applicable

6	**Course Description**
	This Software Engineering degree with Foundation Year will provide you with the knowledge and skills required to design and manage complex software systems. The Foundation Year course option enables you to study for our BSc (Hons) degree over an extended full-time duration of four years by including a Foundation Certificate (year one of four). The Foundation Certificate provides a broad study programme that underpins the follow-on degree. In order to progress to the next year of your degree, it is necessary to achieve a pass in all of the modules of the Foundation Certificate. The course has been designed to provide you with practical experience of models, techniques and tools that drive innovation in the design of software-based systems. In addition, the course also focuses on crucial professional considerations, such as users and organisations, cyber security and computer ethics, as well as the latest developments in the ever-changing computing field, such as machine learning and cloud-based data storage. You will graduate with skills in software design, artificial intelligence, data structures and algorithms and programming ready for a successful career. You could go into roles such as applications developer, full stack developer, software engineer (design, development and testing), site reliability engineer, game developer etc. The course covers the fundamental concepts of Software Engineering and Computer Science in the first year, including computer programming, computer systems, data structures & algorithms, software process, requirements analysis. In the second year, you will consolidate your first-year learning by studying advanced subjects that emphasise software engineering skills including software design, social issues & professional practice, project management. In final year, you will broaden your understanding of software engineering by studying specialised subjects like user experience design, and software quality verification & validation. You will also gain insight into artificial intelligence & machine learning, and modern data store. With its emphasis on computer laboratory based practical work and project delivery, the course provides you with ample opportunity to acquire both the technical and transferable skills desired by industry. The philosophy of the course is to promote lifelong self-learning by providing a curriculum that promotes a variety of approaches to study. The curriculum is designed to develop not only the academic knowledge, but also the engineering ethos and practical skills required by the modern Software Engineering professional. Academic members of staff within the Department of Computer Science will teach topics and supervise projects according to their personal research

	interests. This gives you a cutting-edge experience, one that opens up new opportunities for career advancement in industry. For students who show aptitude for research, there is also the possibility of continuing their studies on a PhD programme. The course structure and its content have been developed by taking into account inputs from various sources including the curriculum philosophy of the School of Architecture, Built Environment, Computing and Engineering where this course is located, the suggestions from industrial partners, the benchmark statements of the British Computer Society (BCS), and the computer science curriculum guidelines released by The Joint Task Force on Computing Curricula comprising Association for Computing Machinery (ACM) and IEEE Computer Society. The course also offers you with the opportunity to take a sandwich placement year between your second and third year. This is something that is highly recommended as it will give you an invaluable opportunity to hone your expertise, try out a potential career path and get relevant workplace experience that is valued by so many employers. The course is offered by the Department of computer Science, which has successfully delivered a wide range of computing courses over the years. The Department has strong links with organisations such as Oracle, Linux Professional Institute (LPI), Microsoft, and Amazon Web Services (AWS), which ensure that the course is relevant and respected by employers. Moreover, the Department is home to a Cisco Systems, and a Microsoft Academy Centre – one of Microsoft's top UK university-based academies. The Department is also a Cisco ASC (Academy Support Centre), Cisco Instructor Training Centre (ITC) and an AWS Academy. All these ensure that you can attain a range of vendor certificates, including LPI 1 Beta certification, Cisco networking certificates and AWS Cloud certificates, while studying for your degree. In addition, the course actively endeavours to enable you to interact with potential employers by collaborating with industrial partners and participating in various events throughout the academic year.

7	**Course Awards**		
7a	**Name of Final Award**	**Level**	**Credits Awarded**
	Bachelor of Science with Honours Software Engineering	6	480
7b	**Exit Awards and Credits Awarded**		
	Foundation Certificate in Computing	3	120
	Certificate of Higher Education Software Engineering	4	240
	Diploma of Higher Education Software Engineering	5	360
	Bachelor of Science Software Engineering	6	420

8	**Variation from the University Regulations**
	None

9	**Delivery Patterns**		
Mode(s) of Study	**Location(s) of Study**	**Duration of Study**	**Code(s)**
Full Time	City Centre	3 years	US1551
Full Time with Foundation Year	City Centre	4 years	US1551F

10	Entry Requirements	
	Home:	The admission requirements for this course are stated on the course page of the BCU website at https://www.bcu.ac.uk or may be found by searching for the course entry profile located on the UCAS website.
	EU:	
	International:	
	Access:	

11	Course Aims
	Through the integration of academic theory and practice the students will develop a strong conceptual and theoretical understanding of fundamental theories, methods, techniques and technologies related to Software Engineering that will lead the students to become an autonomous professional in industry or research and be able to work independently and in groups. Throughout the course students will be supported by expert teaching staff, all of whom have a wide range of research and industrial experience in different sub-fields of Software Engineering. Students will develop technical, research, analytical, planning, design and organisational skills, which are highly transferable that will enable them to make effective and professional contribution beyond the boundaries of traditional Software Engineering. Computer Science has significantly influenced the technological advances over the last few decades and software engineering has played the central role in the radical transformation of businesses and industry practices within this period. As the trend continues, software is needed in almost every industry, in every business, and for every function nowadays; hence the importance of software engineering becomes more evident as time goes on. Examples of career destinations include: • applications developer • full stack developer • web app design/development • software engineer (design, development and testing) • site reliability engineer • game developer • multimedia programming • technical support role

12	Course Learning Outcomes
	Knowledge & Understanding
1	Demonstrate knowledge and understanding of essential facts, concepts, theories and principles of software development and software engineering.
2	Knowledge and understanding of contemporary tools and technologies to produce solutions relevant to the domain of software engineering to meet a set of agreed requirements.
3	Understand the roles and responsibilities of a professional working within the computing profession.
	Cognitive & Intellectual Skills
4	Apply the modelling and design of software-based systems for the purposes of comprehension, communication, prediction and the understanding of trade-offs.
5	Specify the requirements and practical constraints of software-based systems (including computer systems, information systems, and distributed systems) considering a wide range of aspects including commercial, economic, legal, ethical and social issues.

6	Evaluate systems in terms of general quality attributes and possible trade-offs presented within the given problem the ability to recognise any risks or safety aspects that may be involved in the operation of computing equipment within a given context.
	Practical & Professional Skills
7	Deploy effectively the tools, theories and methodologies used for the construction, design & implementation and documentation of computer applications, with particular emphasis on understanding the whole process involved in the effective deployment of computers to solve practical problems.
8	Work as a member of a development team, recognising the different roles within a team and different ways of organising teams.
	Key Transferable Skills
9	Structure and communicate ideas effectively, both orally and in writing to broad range of audience including specialist and no specialist audiences.
10	Show awareness of the rapid rate of change in the IT industry and the need for practitioners continually to update their skills and knowledge.

13	**Level Learning Outcomes**
	Upon completion of Level 4 / the Certificate of Higher Education, students will be able to:
1	Apply problem solving skills to design, implement and test efficient algorithmic solutions to computing problems.
2	Collaboratively produce an innovative and opportunistic Minimum Viable Product.
3	Demonstrate knowledge of computing systems foundations and appraise relevant techniques.
4	Understand the fundamental concepts and principles of software engineering and software process models.
5	Understands the requirements engineering process including various techniques for effective requirements elicitation and documentation.
	Upon completion of Level 5 / the Diploma of Higher Education, students will be able to:
1	Apply tools, theories and methodologies to design and implement computer applications, web applications connected to databases for storing and retrieving of information.
2	Evaluate the quality, security and technical issues in large, complex software engineering and web engineering projects.
3	Appreciate the human, social, environmental, ethical, economic and commercial considerations that impact on the processes of developing computer systems.
4	Apply the modelling and design of computer-based systems for the purposes of comprehension, communication, prediction and the understanding of trade-offs.
5	Investigate, design, and implement computing solutions based on modern theory and practice of Artificial Intelligence.
	Upon completion of 60 credits at Level 6 / the Bachelors Degree, students will be able to:
1	Critically evaluate and defend principal theories and technologies associated with the current trend in the IT industry.
2	Demonstrate a systematic understanding of software security principles and secure software development process.
3	Deploy effectively the tools, theories and methodologies to assure software quality.
4	Demonstrate theoretical concepts and practical skills in user centred software design process.
5	Demonstrate ability to plan, design and implement research informed solutions to computing problems.

14	Course Learning, Teaching and Assessment Strategy
	Knowledge and understanding are acquired though a wide range of teaching and learning strategies that may differ module to module depending on the content of the module. Some of the common strategies are formal lectures, tutor-led seminars and practical activities and other independent learning activities at all stages. Emphasis is placed on guided, self-directed and student-centred learning with increasing independence of approach, thought and process. **Lectures:** Lectures are usually held in a lecture theatre and delivered to large groups of students. The purpose of the lectures is to introduce themes, theories and concepts within a particular module. Technology enhanced learning is used, where appropriate, through the provision of online resources, discussion forums and other activities. **Practical Sessions:** Practical sessions are usually held in various laboratories of the university and delivered to small group of students. These sessions are used throughout the course to develop practical skills and to place theory in a work-related context and most often based upon the topics covered in lectures. **Directed Study:** In this mode of study academic staff may direct students to undertake individual or group work that is relevant to the module that students study and the assessment that is part of the module. Quite often students will be required to carry out the work/study outside of the module's normal contact time. Virtual Learning Environment, Moodle, will be heavily used in the delivery of the course. Each module that students study will have its own Moodle site which will contain important documents and information pertaining to that module such as lecture notes, lab handouts, links to external resources. Each student will undertake a major project in the final year that involves research and application of that research in solving appropriate systems problems. Student will develop the key skills of research, academic writing and time management required for study at degree level throughout years 1 and 2. These skills are further developed and placed into context through the major project during the final year. The course provides access to effective commercial development environments and ensures students have practical awareness of computer systems requirements. Learners are required to meet strict deadlines and to manage and plan their overall workload. Assessments are designed to be appropriate to the needs of demonstrating achievement of module learning outcomes. Knowledge is assessed formatively and summatively, by a number of methods, including seminars, course-works, examinations (seen and unseen, open and closed-book), viva, presentation, interactive automated assessment, and project work. Assessment criteria are published both at a generic course level and to provide guidance for individual items of assessment. In accordance with the University Regulations, students will be permitted two attempts at assessments. After first sit, if a student is unsuccessful in any of assessments, there will be a final resit opportunity for which mark, provided that you pass the assessment, will be capped at 40%.

15	Course Requirements

15a

Level 3:

In order to complete this programme a student must successfully complete all the following CORE modules (totalling 120 credits):

Module Code	Module Name	Credit Value
CMP3010	Fundamental Mathematics	20
CMP3014	Fundamentals of Digital Technology	20
CMP3012	Web Application Design	20
CMP3016	Emerging Technologies	20
CMP3015	Independent Practice	20
CMP3009	Foundations of Programming	20

Level 4:

In order to complete this course a student must successfully complete all the following CORE modules (totalling 120 credits):

Module Code	Module Name	Credit Value
CMP4266	Computer Programming	20
CMP4267	Computer Systems	20
CMP4XXX	Introduction to Software Engineering	20
CMP4272	Data Structures and Algorithms	20
CMP4XXX	Requirements Engineering	20
CMP4285	Innovation Project	20

Level 5:

In order to complete this course a student must successfully complete all the following CORE modules (totalling 120 credits):

Module Code	Module Name	Credit Value
CMP5332	Object Oriented Programming	20
CMP5387	Backend Webservice Development and Database Engineering	20
CMP5XXX	Inclusive and Impactful Software Solutions	20
CMP5354	Software Design	20
CMP5XXX	Enterprise Project Management	20
CMP5367	Artificial Intelligence and Machine Learning	20

Level 6:

In order to complete this course a student must successfully complete all the following CORE modules (totalling 120 credits):

Module Code	Module Name	Credit Value
CMP6214	User Experience Design	20
CMP6XXX	Cybersecurity for Software Engineers	20
CMP6207	Modern Data Stores	20
CMP6XXX	Software Quality Assurance	20
CMP6200	Individual Honours Project	40

15b Structure Diagram

All modules are core 20 credits, unless otherwise stated.

Level 3

SEMESTER ONE	SEMESTER TWO
Core	**Core**
CMP3010 Fundamental Mathematics	CMP3016 Emerging Technologies
CMP3014 Fundamentals of Digital Technology	CMP3015 Independent Practice
CMP3012 Web Application Design	CMP3009 Foundations of Programming

Level 4

SEMESTER ONE	SEMESTER TWO
Core	**Core**
CMP4266 Computer Programming	CMP4272 Data Structures and Algorithms
CMP4267 Computer Systems	CMP4XXX Requirements Engineering
CMP4XXX Introduction to Software Engineering	CMP4285 Innovation Project

Level 5

SEMESTER ONE	SEMESTER TWO
Core CMP5332 Object Oriented Programming CMP5387 Backend Webservice Development and Database Engineering CMP5367 Artificial Intelligence and Machine Learning	**Core** CMP5354 Software Design CMP5XXX Enterprise Project Management CMP5XXX Inclusive and Impactful Software Solutions

Level 6

SEMESTER ONE	SEMESTER TWO
Core CMP6214 User Experience Design CMP6XXX Cybersecurity for Software Engineers	**Core** CMP6207 Modern Data Stores CMP6XXX Software Quality Assurance
Core CMP6200 Individual Honours Project (40 Credits)	

16 Overall Student Workload and Balance of Assessment

Overall student *workload* consists of class contact hours, independent learning and assessment activity, with each credit taken equating to a total study time of around 10 hours. While actual contact hours may depend on the optional modules selected, the following information gives an indication of how much time students will need to allocate to different activities at each level of the course.

- *Scheduled Learning* includes lectures, practical classes and workshops, contact time specified in timetable
- *Directed Learning* includes placements, work-based learning, external visits, on-line activity, Graduate+, peer learning
- *Private Study* includes preparation for exams

The *balance of assessment* by mode of assessment (e.g. coursework, exam and in-person) depends to some extent on the optional modules chosen by students. The approximate percentage of the course assessed by coursework, exam and in-person is shown below.

Level 3

Workload

37% time spent in timetabled teaching and learning activity

Activity	Number of Hours
Scheduled Learning	456
Directed Learning	400
Private Study	344
Total Hours	1200

Balance of Assessment

Assessment Mode	Percentage
Coursework	78%
Exam	0
In-Person	22%

Level 4

Workload

24% time spent in timetabled teaching and learning activity

Activity	Number of Hours
Scheduled Learning	294
Directed Learning	420
Private Study	486
Total Hours	1200

Balance of Assessment

Assessment Mode	Percentage
Coursework	80%
Exam	5%
In-Person	15%

Level 5

Workload

24% time spent in timetabled teaching and learning activity

Activity	Number of Hours
Scheduled Learning	288
Directed Learning	396
Private Study	516
Total Hours	1200

Balance of Assessment

Assessment Mode	Percentage
Coursework	96.7%
Exam	0%
In-Person	3.3%

Level 6

Workload

17% time spent in timetabled teaching and learning activity

Activity	Number of Hours
Scheduled Learning	202
Directed Learning	234
Private Study	752
Total Hours	1200

Balance of Assessment

Assessment Mode	Percentage
Coursework	80.83%
Exam	3.33%
In-Person	15.83%