SECTION 1: KEY PROGRAMME DETAILS

PART A: PROGRAMME INFORMATION	
Highest Award	BSc (Hons) Software Engineering for Business
Interim Award	BSc Software Engineering for Business
Interim Award	DipHE Software Engineering for Business
Interim Award	CertHE Software Engineering for Business

Awarding Institution	UWE Bristol
Teaching Institution	UWE Bristol
Delivery Location	Frenchay Campus
Study Abroad / Exchange / Credit Recognition	Placement X Sandwich Year X Credit Recognition X Year Abroad X
Faculty Responsible For Programme	Faculty of Environment & Technology
Department Responsible For Programme	FET Dept of Computer Sci & Creative Tech
Professional Statutory or Regulatory Body (PSRB) Links	Tech Partnership
Apprenticeships	
Mode of Delivery	Full-time

ENTRY REQUIREMENTS	UCAS Tariff Points: For the current entry requirements see the UWE public website.
For Implementation From	1 Sep 2020

ISIS Code/s	Programme Code I3NC13-SEP-FT-FR-I3N1 Other codes: JACS Software engineering HECoS 100374: Software Engineering UCAS SLC

SECTION 2: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

1. (Programme) Overview (c. 400 words)

Central and crucial to the SEfB programme will be an emphasis on partnership, student centred engagement and an appropriate balance of education, training and practice-based experience. Employers actively supporting the programmes include BT, CA Technologies, The Cabinet Office, Capgemini, CGI (was Logica), Cisco, Enternships, General Dynamics, IBM, Kinetic, Ministry of Justice, MOD, NHS Wales and TCS (Tata Consultancy Services). UWE has developed close links with a number of the members of the e-skills UK employers strategy forum e.g. IBM, Capgemini, GSK, and Credit Suisse, who regularly provide placements and internships for our students and who have employed a good number of our graduates.

All partner universities create their own version of a balanced curriculum that engages students in various learning and practical activities. Consistency across all programmes is achieved through an interaction of aims/objectives, assessment techniques, teaching methods and content approved by both the partner employers and the university. Partner universities are expected to design a curriculum that meets all of the requirements of the Learning Outcomes Skills Requirement (LOSR) document and proof of this is a requirement for endorsement.

An employer-led design team, informed by wider employer consultation, has worked in partnership with curriculum experts from partner universities (including UWE) to define the learning outcomes of most value to the sector. Strategies for sustainability and for employer involvement have been designed in, informed by our collective experience of ITMB, including ‘guru’ lectures, awards, project challenges and business placements for students.

UWE has been a member of the e-skills Programme Development Group since the start of the design process. The goal was to produce the blueprint for a programme which sets high standards re:

Relevance of content.
Effective employer engagement.
Attracting the highest quality students.
Appeal to women as well as men.
Degree completion rates.
Progression into IT careers.

2. Educational Aims (c. 4-6 aims)

The specific educational aims of the programme are to:

Provide students with a broad background understanding of business operations, procedures and culture, as applicable to a career in an IT environment.

Enable students to recognise the nature, role and importance of information systems within business

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

organisations and the importance of information security.

Develop students' knowledge and practical skills to select and employ appropriate technologies, tools, techniques and methods for understanding and developing information systems in business contexts.

Develop deep technical competence in the software development process as applied in business.

Equip students to play a key role in software project management.

Develop both personal and inter-personal skills to enable the students to work closely and communicate with others in all aspects of the software development process.

Provide students with a set of problem-solving and modelling skills appropriate to IT related business systems development and operations.

Enable the students to play a central role in an IT project; and gain business experience in a project oriented environment.

Develop the students' critical, evaluative team working and problem-solving abilities that will be valuable to them in any career.

Continue the development of those general study skills that will enable students to become independent, lifelong learners including career and employability awareness.

3. Programme and Stage Learning Outcomes (c. 6-8 outcomes)

Programme (Learning) Outcomes (POs)

Knowledge and Understanding

A1	IT in Business Students will be able to assess the contribution and impact of technology in the global business environment; demonstrate an understanding of basic business functions and organisational structures in different sectors (including an international dimension) and possess a sound understanding of contemporary working practices. Students will understand why a business case is needed and be able to create a business case for a technology-enabled business solution, including an assessment of business benefits, impacts, risks and return on investment and design a small-scale system suitable for business use, adopting open-source software, operating systems, development tools and materials as appropriate.
A2	Custom Software Development Essentials Students will understand software development processes that support the design and construction of software products, including the transformation of a design into an implementation, the tools used during this process, formal software construction methods and designing for maintainability; Students will understand software design and development methodology (e.g., structured or object-oriented), be able to apply appropriate industry standard design notation, select, with justification, an appropriate set of tools to support the development of software products; explain the potential benefits and drawbacks of using formal

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

	specification languages and understand the software development process as aligned to industry practice.
A3	Data and Algorithms. Students will be able to write programs that use a wide range of data types and data structures; use pseudo code and an appropriate programming language to implement, test, and debug algorithms for solving problems; confidently design and apply algorithms for manipulating data in programming solutions for a variety of computational problems and understand search techniques and memory management.
A4	Software Modelling and Analysis. Students will understand the role and purpose of modelling and the distinction between analysis and design models; Students will be able to apply a wide variety of industry standard modelling techniques, analyse the problem domain to establish a basis for the creation of a software design and describe what the customer requires through solution design modelling.
A5	Software Architecture. Students will understand the differences between multi tiered (1/2/3) architectures and layers and their merits, be able to formulate a system according to a multi-tier architecture, so that the presentation, the application processing and the data management can be separated into different logical processes and understand concepts such as separation of concerns, loose coupling and cohesion. Students will understand the role of software architecture in software design.
A6	Software Requirements Management. Students will understand the requirements development process: elicitation, specification, analysis, and management and the use of tools for managing requirements, and how requirements, design, implementation and verification are linked together to control development and ensure quality. Students will be able to elicit and negotiate requirements using a variety of techniques, be competent in writing clear and concise functional and non-functional requirements, organize and prioritize and validate requirements according to criteria such as feasibility, clarity, testability and freedom from ambiguity.
A7	Software Design. Students will be able to discuss the properties of good software design including the nature and the role of associated documentation, appreciating that design increasingly covers use of existing code and 3rd party elements, select and apply appropriate design patterns in the construction of software, understand the different contexts for HCI (mobile devices, consumer devices, business applications, web, business applications, collaboration systems, games, etc.) and be able to define a user-centred design that explicitly recognises the user and is DDA compliant (Disability Discrimination Act), create and specify a software design for a medium-size software product using a software requirement specification, an accepted program design methodology (e.g., structured or object-oriented), and appropriate design notation.
A8	Software Verification and Testing. Students will be able to describe in detail the purpose of, and distinguish between the different types and levels of verification (analysis, demonstration, test, formal proof, inspection etc.) and testing (unit, integration, systems, and acceptance) including the role and value of test driven development techniques, analyse requirements to determine appropriate testing strategies, create, evaluate, and

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

	implement a test plan, develop and execute accurate and clear test scripts, be competent in developing automated tests within the build phase for applications at a transaction and method level.
A9	Software Development Process. Students will be able to understand the importance of a software process for governing software development both technically, and in terms of cost control, quality, adherence of technical strategy and IPR identification. They will understand the software life cycle, its phases and deliverables, and have a critical awareness of both predictive and adaptive methods and approaches so as to be able to select, with justification software development models and process elements most appropriate for the development and maintenance of a diverse range of software products. Students will have an understanding of distributed development work (e.g. onshore, near shore and offshore) and understand the processes, benefits and drawbacks of each.
A10	Software Development in Context. Students will understand the fundamental components of technology solutions in a range of typical modern business environments and explain their interactions for applicable target systems – e.g. games console, smart-phone, embedded system understanding the differences that these environments bring. Students will understand the contexts of real time and embedded systems development as well as component software within hardware systems and be able to demonstrate the capability to justify, select, and apply an appropriate set of tools to support the development of a range of software products and to specify and implement user interfaces for a range of different contexts.
A11	Software Configuration and Release Management. Students will be able to define configuration management processes for use throughout the product development life cycle in storing software deliverables and controlling and tracking changes to software both at component and release level; modify software designs and deliverables following sound change control approaches and change control tools; understand the importance of version control, select and apply configuration management and change control tools for use within software development projects, and be able to apply them.
A12	Software Deployment. Students will understand the organisational context into which software is deployed and the human and business issues associated with deployment, recognise the challenges of deploying software releases which form part of a software or hardware system, perhaps with other software elements from a 3rd party, and embedded systems, Be able to interact to agree dependencies on interfaces, processing speed, resource utilisation and staged deployment in to maturing prototypes and systems, develop and apply user documentation and training materials as part of software development and deployment activities and design and develop training materials and plan end user training following software deployment.
A13	Software Maintenance. Students will understand the impact of developing software for systems which need to be maintained for extended operational periods and be aware of the importance of documentation rigour in those circumstances, read and Analyse existing software behaviour in order to improve its efficiency, reliability, and maintainability, maintain and update software as required to ensure continued effectiveness and in response to external factors and understand the role and purpose of refactoring in improving programming solutions efficiency, scalability, maintainability and extensibility.

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

A14	Legacy Systems. Students will have an understanding of legacy architectures and technologies (e.g. mainframe/COBOL etc) and be equipped to identify, review and understand legacy system documentation, source code and system data architectures distributed across files with incompatible structures. Students will understand the reasons for, and risks associated with, replacing or keeping and maintaining legacy systems.
A15	Software Quality. Students will be able to specify, design and build high-quality software components, understanding the differences and interactions between safety and quality, compare and contrast quality methods and techniques and be aware of industry standard static and dynamic code analysis frameworks.
A16	Data Modelling, Database Development and Data Analysis Students will understand the basic principles of the relational data model and the issues of scale and management of large data or big data; be competent at developing a range of industry standard database models; Students will be able to: create a relational database schema that incorporates key, entity integrity, and referential integrity constraints; implement a database-driven web site, explaining the relevant technologies involved in each tier of the architecture and the accompanying performance tradeoffs; write stored procedure queries; understand the role of data mining, the algorithms developed to address different data mining goals and the application of these algorithms to real-world problems including big data.
A17	Risk and Information Security. Students will understand the nature of risk to information and information systems; be able to define what cyber security is, and explain its importance when developing software solutions and mitigating risk; appreciate the importance of determining and managing risk for threats and vulnerabilities to information systems on an ongoing basis; understand human aspects of information security including client data protection and the data protection act; understand how to make software more resilient to threats.
A18	Software Project Management. Students will be able to: interpret and use standards in software project management, including PRINCE2; prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, change management, and project risk identification and management; be able to identify the range of software development resources required and allocate them to a project and demonstrate through involvement in a team project the central elements of team building and team management during software development; manage project progress, productivity and other aspects of software development processes, against plan; apply the concepts of earned value management to IT projects and programs; identify risks to a software project and indicate an approach to managing risk that will help to secure the on-time on-budget on-quality delivery of software; document project progress recording risks, actions, issues and decisions.
A19	Professional Practice Students will: understand the principles of leadership and be able to work effectively as a member or lead of a small development team and adopt best practices for developing software in teams, recognising challenges and approaches taken to resolve them; understand the challenges of working in a distributed team, and mechanisms to address the challenges; be able to make

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

concise, engaging and well-structured verbal presentations, arguments and explanations, to a range of audiences about technical problems and their solutions, taking into account the audience viewpoint at all times; understand the wide variety of IT professional and managerial roles that exist and understand the role and membership benefits of professional bodies in promoting IT professionalism.

A20 Innovation.

Students will: understand technology innovation, including disruptive technologies and be able to describe current and past examples (e.g., cloud, apps, the hard disk drive, text messaging, GPS, the smart phone, Linux, Social networking); be equipped to explore and describe how businesses innovate through technology, considering a range of organisations in different sectors; be able to assist in the assessment of the commercial viability of new technology based ideas and in transforming research based ideas into feasibility and business plans.

Intellectual Skills

B1 Critical Thinking.
B2 Analysis.
B3 Synthesis of different types of information.
B4 Evaluation .
B5 Problem Solving.
B6 Appreciate problem contexts.
B7 Balance conflicting objectives.

Subject/Professional Practice Skills

C1 Show competence in software development processes, including the knowledge, skills, and professional competences necessary to begin practice as a software engineer in a business environment.

Students, through learning, regular reinforcement and practice, and exposure to real world software development paradigms need to gain confidence in their ability to perform software development practice. Through a structured program of study, the knowledge, understanding and skills are acquired through a staged approach with increasing levels of competence being achieved as the degree progresses. Graduates need to develop an understanding and appreciation of professional issues in software development related to ethics and professional conduct, economics, and the needs of employers.

C2 Be able to confidently work as an individual and as part of a team to develop and deliver quality software deliverables.

Students need to develop the ability to perform software development tasks that involve work both as an individual and also working in a team. Developing an appreciation and experience of effective team working is essential to operating effectively in a business environment. The degree programme must include an emphasis on the importance of team working as a disciplined approach, the need to adhere to deadlines, communication, and team as well as individual performance evaluation.

C3 Design appropriate solutions in a range of application contexts/domains using software development approaches that deliver business value.

Students need to be exposed to a variety of software design contexts, including desktop applications, internet and mobile device software and apps. They need to understand and develop appropriate approaches to software design in the general sense, and to problem solving for specific application domains. They

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES	
	need to be able to understand the strengths and the weaknesses of the various options available and the implications of the selection of appropriate approaches for a given context. They must develop competence in designing solutions within business constraints of time, cost and quality as well as addressing security. Having a thorough understanding of software requirements (including non-functional requirements), how to interpret and prioritise requirements is essential.
C4	Build and test software solutions for a range of application contexts/domains. Students need to be able to apply software development to a wide range of contexts and environments. These include operating systems, internet and fixed platforms, apps, games, business solutions etc. They need to be able to understand how to transfer development principles to new and different technologies. They need to be able to assure the functionality of their solutions through testing and embed secure development principles to all stages of development. Students also need to understand that a significant amount of software development work, involves updating, maintaining and refactoring existing solutions. They need to develop and demonstrate confidence in reading and reviewing existing software and the need to increase efficiency through refactoring.
C5	Reconcile conflicting project objectives, finding acceptable compromises recognising the limitations of capability, capacity, cost, and time. Students will develop a professional approach to managing their contribution to project deliverables. In the business environment they will need to understand how to estimate activities and balance meeting the specification, but delivering on time and to the right quality level. Students need to understand how to approach conflicting demands and engage in exercises that expose them to conflicting and changing requirements. There will be a strong real world emphasis in the study. Students will appreciate when to question deliverables – e.g. is an activity to fix an error (bug) to achieve the stated quality, or is it an enhancement request that needs to be considered as an additional requirement with impact assessment, planning, resourcing and costing decisions to be made on if and when to consider the additional request. The degree programme will address these issues, with the aim of ensuring high quality requirements and a feasible software design.
C6	Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for problem identification and analysis, software design, development, implementation, verification, and documentation. The presence of the Capstone project, an important final activity at the end of a software development program of study, is of considerable importance in this regard. It offers students the opportunity to tackle a major project and demonstrate their ability to bring together topics from a variety of courses and apply them effectively. This mechanism allows students to demonstrate their appreciation of the broad range of software development topics and their ability to apply their skills to genuine effect. This will also include the ability to offer reflections on their achievements.
C7	Demonstrate an understanding and appreciation for the importance of negotiation, effective work habits, leadership, and good communication with stakeholders in a typical software development business environment. It is important that students gain an insight and understanding of range of professional interpersonal skills required to be effective in business when producing a solution for a problem domain (requirements, design, solution development etc). Software engineers must recognise that effective software development is achieved as a result of a dynamic and interactive team based approach. Employer visits, guest lectures from practicing software engineers etc

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES

will aid students in better comprehending the business team environment, and the importance of professionalism.

C8 Learn new models, techniques, and technologies as they emerge and appreciate the necessity of such continuing professional development.

On graduating, students will clearly demonstrate that they are self-motivated life-long learners. They will understand the continuous nature of emerging technologies, and how and when is best to harness and exploit their benefits for business gain. Students will be capable of identifying, selecting and applying new technologies, tools and methods and be able to assess their contribution to software development. The final year project allows students to apply critical thinking and use context driven software development to plan and complete a significant project (not simply a software deliverable).

C9 Embedding security in the software development life cycle.

Security has become a widespread and significant issue in the development and ongoing safeguarding of software systems. Students will have a robust understanding of software security issues, and the principle models of security in relation to information systems. They will understand the nature of risk, threats and vulnerabilities and be able to incorporate preventative measures into software design. They will be aware of current approaches to safeguarding the systems that they are involved in developing, including security testing. Throughout the degree they will be thinking about security implications and adopt secure architecture, design and development practices.

Transferable Skills and other attributes

D1 Communication skills: to communicate orally or in writing, including, for instance, the results of technical investigations, to peers and/or to "problem owners".

Developed through a variety of methods and strategies including the following:

Participation in tutorials and other discussion forums
Negotiation of work plans and requirements with team members and clients.
Presentation of work to peers, staff, and clients.
Writing essays, reports, and examination answers.
Students participate in electronic conferences, workshops, and group work sessions.

D2 Self-management skills: to manage one's own time; to meet deadlines; to work with others having gained insights into the problems of team based systems development.

Developed through a variety of methods and strategies including the following:

Self-managed practical work.
Effective participation in tutorial and laboratory sessions.
Methodical execution of analysis and design tasks.
Synchronising with others in team work.
Scheduling assignment work and revision.
Scheduling and attending meetings with clients.

D3 IT Skills in Context (to use software in the context of problem-solving investigations, and to interpret findings).

Developed widely throughout the programme including:

Use of range of system development tools, methods and packages.

PART A: PROGRAMME OVERVIEW, AIMS and LEARNING OUTCOMES	
	Regular involvement in systems analysis and design activity. Cumulative mastery of tools and methods. Use of online teaching materials. Sustained use of the Internet. Emphasis on user-centred and accessible systems design work. Building systems to a user-focused specification.
D4	Problem formulation: To investigate and express problems in appropriate forms. Developed through a variety of methods and strategies including the following: Students develop problem solving systems. Students practice systems design and development using a variety of tools and methods.
D5	Progression to independent learning: To gain experience of, and to develop skills in, learning independently of structured class work. For example, to develop the ability to use on-line facilities to further self-study. Developed through a variety of methods and strategies including the following: Students are encouraged to research relevant topics in order to complete tutorial task and project based work. Students are encouraged to use online facilities to discover information.
D6	Comprehension of professional literature: to read and to use literature sources appropriate to the discipline to support learning activities. Developed through a variety of methods and strategies including the following: Students are encouraged to access online material.
D7	Working with Others: to be able to work as a member of a team; to be aware of the benefits and problems which teamwork can bring. Developed through a variety of methods and strategies including the group work undertaken in The Information Practitioner modules.

PART B: Programme Structure
1. Structure

Year 1

Year 1 Compulsory Modules

Code	Module Title	Credit	Type
UFCFFS-30-1	Foundations of Computing 2020-21	30	Compulsory
UFCF9F-30-1	Information Systems Development 1 2020-21	30	Compulsory
UFCFC3-30-1	Introduction to OO Systems Development 2020-21	30	Compulsory
UFCF83-30-1	IT Practice: Skills, Models and Methods 2020-21	30	Compulsory

Year 2

Year 2 Compulsory Modules

Code	Module Title	Credit	Type
UFCFV4-30-2	Data, Schemas and Applications 2021-22	30	Compulsory
UFCFW4-30-2	Design and Analysis of Data Structures and Algorithms 2021-22	30	Compulsory
UFCFN6-30-2	IT Practice: Collaborative Project 2021-22	30	Compulsory
UFCFB6-30-2	Object-Oriented Systems Development 2021-22	30	Compulsory

Year 3

Year 3 Compulsory Modules

Code	Module Title	Credit	Type
UFCFAF-30-3	Development of Information Systems Projects (disp) 2022-23	30	Compulsory

UFCFB5-15-3	Ethical and Professional Issues in Computing and Digital Media 2022-23	15	Compulsory
UFCFP6-30-3	IT Practice: Consultancy Project 2022-23	30	Compulsory
UFCFFF-30-3	Software Development Project 2022-23	30	Compulsory

Year 3 Optional Modules
Students choose 15 credits from the modules below:

Code	Module Title	Credit	Type
UFCFU3-15-3	Advanced Databases 2022-23	15	Optional
UFCFX3-15-3	Advanced Topics in Web Development I 2022-23	15	Optional
UFCFT4-15-3	Cryptography 2022-23	15	Optional
UFCF95-15-3	Entrepreneurial Skills 2022-23	15	Optional
UFCFA5-15-3	Information, Networks and Society 2022-23	15	Optional
UFCFVJ-15-3	Professional Development 2022-23	15	Optional
UFCFM6-15-3	Requirements Engineering 2022-23	15	Optional
UFCFD5-15-3	Technical Writing and Editing 2022-23	15	Optional

PART C: Higher Education Achievement Record (HEAR) Synopsis

Employer-designed and backed and e-skills UK endorsed, this programme provides graduates with the mix of skills and capabilities required by UK business for the specification, design and delivery of ICT systems, services and solutions in a range of business contexts and application domains.

It develops technically competent individuals who think and communicate effectively and who can conduct inquiry, solve problems, undertake critical analysis and deliver effective software systems solutions in a constantly changing business context.

It provides a solid foundation for lifelong learning, emphasising the development of knowledge, skills and professional values essential to the practice of systems development.

PART D: EXTERNAL REFERENCE POINTS AND BENCHMARKS

In designing this programme, the Faculty has drawn upon the following external reference points:

QAA UK Quality Code for HE: National qualification framework

QAA Subject benchmark statements for Computing and Business and Management

UWE Bristol Strategy 2020

E-skills UK Endorsement Document for the BSc Information Technology Management for Business

UWE's Technology Enhanced Learning Strategy (2012-2017).

The QAA UK quality code for Higher Education Qualifications describes the attributes and skills expected of Honours graduates. It is our view that the learning outcomes of the programme are fully consistent with the qualification descriptor in the Framework, and hence that graduates are able to demonstrate that they meet the expectations of the Framework.

The curriculum for the programme draws on the QAA Subject Benchmark Statements for Computing and (to a lesser extent) those for Business and Management. The QAA Computing Benchmarking document recognizes that computing awards may be placed on a spectrum, with those covering a broad range of computing topics at one end, and those focusing on specialist areas, e.g. safety-critical systems, at the other. This award lies between the two extremes in that it provides a reasonably broad coverage of the main areas of Information Systems development applicable in the business context. The specified aims, objectives and philosophy lead to an award which conforms to the principles of course design in the benchmark statement. The QAA Business and Management benchmark explicitly recognises the important role of the study of Information Systems in the context of business, and the design of the programme reflects this.

Through its constituent teaching and learning elements, assessment strategies, practical focus and learning outcomes, the programme intends to support the achievement of the workstreams defined in the UWE Bristol Strategy 2020. Specifically, the programme will contribute to the strategic ambitions of outstanding learning (Work-stream 1) and developing ready and able graduates (Work-stream 2).

UWE's Learning and Teaching Strategy has informed the Faculty's policy for the delivery of its programmes and this companion to an already well-established programme (ITMB) is proposed in the light of the recent CSCT curriculum refresh project. It also takes account of the policies and requirements set out in UWE's Technology Enhanced Learning Strategy and the current FET Teaching, Learning and Assessment Strategy. In particular, it seeks to maximise the efficiency of resource utilisation while promoting the achievement of high quality outcomes through, for example:

The utilization of existing core modules from both the computer science and information science clusters (the programme contains one new module at L1 and two at L2)

Consolidation of the existing emphasis on partnership, student-centred engagement and an appropriate balance of education, training and practice-based experience;

An increased focus on ethical and professional issues relating to computing and digital media.

The ethos, structure and content of the proposed programme has been very much shaped by the success of the ITMB programme, together with input from the e-skills employers strategy forum, feedback from employers of placement students and from graduates of both the ITMB and the BSc Business Information Systems.

The emphasis of the programme is to prepare students with a solid grounding in software engineering so as to enable them to take up a variety of technical positions in industry. The success of the approach been borne out by the wide variety of organisations in which ITMB students have been placed or employed and by the overwhelmingly positive feedback of their employers.

PART D: EXTERNAL REFERENCE POINTS AND BENCHMARKS

The development of the programme reflects UWE policies and is fully consistent with the University's commitment to 'make a positive difference to our students, business and society'.

UWE Bristol Strategy 2020, Outstanding Learning:

As previously stated, central and crucial to the SEfB programme will be an emphasis on partnership, student centred engagement and an appropriate balance of education, training and practice-based experience. As with the ITMB, this award is expected to set high standards with respect to:

Relevance of content

Effective employer engagement

Attracting the highest quality students

Appeal to women as well as men

Degree completion rates

Progression into IT careers.

UWE Bristol Strategy 2020, Ready and able graduates:

As part of the program design and consultation exercise the following features of the BSc 'ITMB' were identified as areas of good practice. These are central to and have been built upon for the' SEfB programme.

The high level of employer involvement helps to confirm and consolidate the relevance and importance of the topics and subjects covered by the award, to broaden the context of study for the students through exposure to contemporary applications, initiatives and issues and to inspire them to see their education as a powerful force in their personal and professional development.

e-skills UK organize an annual round of inter-university competitions. UWE's first year inter-university competition entries are developed in Peer Assisted Learning (PAL) sessions, under the guidance of the second-year-student PAL tutors. Participation in these competitions helps students to develop personal and interpersonal and project management core skills as well as enhancing their knowledge and understanding of emerging areas of importance. It also helps to instill a strong sense of self-worth in both the participating students and the wider cohort, and helps students to understand that academic achievement is only one part of the skills mix required for success in life.

The second year 'Power up' programme for CSCT students focuses on employability and consists of a mixture of department wide activities, jointly run (employer led) employability workshops and, additionally for ITMB/ SEfB, off-site, employer run mock selection centres and CV clinics.

The 3 level vertically integrated practice theme provides a focus for the integration of academic learning and the practical application of technical, personal and interpersonal skills. It is of great value in preparing students as reflective practitioners and provides high quality work experience (especially to those students who elect not to undertake a placement year).

PART E: REGULATIONS

Approved to University Regulations and Procedures.