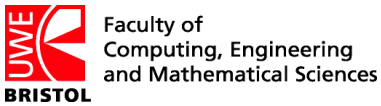

Faculty of Computing, Engineering & Mathematical Sciences

BSc (Hons) Computing

May 2006

Modified 23 May 2006
Modified 31 July 2003
Modified 3 September 2003
Modified 9 September 2003

Contents Page

Programme Specification

Section 1: Basic Data

Awarding institution/body	University of the West of England
Teaching institution	University of the West of England
Faculty responsible for programme	Computing, Engineering and Mathematical Sciences
Programme accredited by	N/A
Highest award title	BSc (Hons) Computing
Default award title	
Interim award title	BSc Computing, Dip HE Computing, Cert HE Computing
Modular Scheme title (if different)	
UCAS code (or other coding system if relevant)	
Relevant QAA subject benchmarking group(s)	Computing
On-going/valid until* (*delete as appropriate/insert end date)	
Valid from (insert date if appropriate)	1st September 2003

Authorised by… **Date:…**

Version Code

For coding purposes, a numerical sequence (1, 2, 3 etc.) should be used for successive programme specifications where 2 replaces 1, and where there are no concurrent specifications. A sequential decimal numbering (1.1; 1.2, 2.1; 2.2 etc) should be used where there are different and concurrent programme specifications

Section 2: Educational Aims of the Programme

The aims of the programme are:

1. to provide a broad-based treatment of the fundamental aspects of computing, the development of computer systems and the application of computing to practical problems;
2. to develop sufficient experience, knowledge and understanding to enable students to analyse, model and develop applications in a diverse application areas such as internet systems, database applications and modern component-based construction;
3. To provide a diversity of routes to a honours degree, in order to enable students from a variety of backgrounds to progress successfully;
4. To prepare students for computing careers in business, industry, and commerce, or in organisations with a significant in-house IT management culture.
5. To develop problem-solving and other transferable skills that will be valuable to students in any career.

Section 3: Learning Outcomes of the Programme

The award route provides opportunities for students to develop and demonstrate knowledge and understanding, intellectual skills, subject-specific skills and transferable skills. The individual student's choice of modules will determine the range of skills developed; however for the 'recommended pathway' the following set of skills is developed. The teaching and learning strategies and the modes of assessment are as in the relevant module specifications, and are shown here for indicative purposes.

A. Knowledge and Understanding

Knowledge and Understanding of:	***Teaching/Learning Methods and Strategies***	***Assessment***
1. Object-oriented programming language concepts; other programming paradigms; syntax and semantics; top-down development; programming to satisfy designs. 2. Program design concepts, methods, and notations; object-oriented design and other design paradigms; algorithms; design	On all modules, at all levels, the learner is encouraged to undertake independent reading both to supplement and consolidate what is being taught/learnt and to broaden their individual knowledge of the subject. The programme of study is designed to introduce the knowledge and understanding necessary to engage, from the beginning, in appreciating and solving small-scale problems. At level 1, the	Testing of the knowledge base is through: Assessed coursework (topics: 2, 4, 5, 8, 9); Assessed practical work (topics: 1, 2); Examination (topics: 1, 2, 3, 4, 5, 6, 7, 8, 9); Peer and tutor evaluation (topics: 2); Group coursework/project (topics: 2); Portfolio of exercises (topics: 1, 7).

patterns. 3. Object-oriented and related databases; logical and physical database design; database query languages. 4. The concepts underpinning distributed systems and networks. 5. The concepts underpinning World-Wide Web technology and web-based application development. 6. Electronic commerce; architectures and components of commercial applications based upon www technology; technical and management issues. 7. The concepts underlying the reuse of components and framework in software development; related research issues. 8. The architecture and main components of computers. 9. The concepts underpinning user interfaces; good design practice; notation issues; user interface evaluation.	context in which these issues reside is introduced but the in-depth understanding of large, complex, real-world problems essentially starts with level 2 study. At level 3, we continue to increase in-depth knowledge and understanding of in-depth to technical solutions of real-world problems for topics pertinent to the present state of the industry. At level 1, knowledge and understanding of topics 1-6 (Object-oriented programming language concepts; Program design concepts; Object-oriented and related databases; Concepts underpinning distributed systems and networks; Concepts underpinning World-Wide-Web technology; and electronic commerce are introduced on two modules which explore the general concepts, components and issues, positioning them in the computing environment. The general understanding of topics 1-6 is built on with more in-depth knowledge and specific understanding of application in further levels. Topic 8, "The architecture and main components of computers." is taught only at level 1 providing the delimiters of a sufficient technical knowledge and understanding. Topics 7 & 9 ("Concepts underlying the reuse of components and framework in software development", and "Concepts underpinning user interfaces; good design practice; notation issues; user interface evaluation.") have only a cursory mention at level 1 although the more astute learner will find consistent references to relevant knowledge. At level 2 the knowledge and understanding of computing continues with an expansion into broader and larger issues, such as, the design of, and methods of building, large systems. The complexity and design of such systems is addressed in all level 2 modules. Moreover, in-depth knowledge and understanding of topics 2-4 (Program design concepts; Object-oriented and related databases; Concepts underpinning distributed systems and networks.) is delivered in these modules. At level 2, knowledge of topic 1 is assumed but will be consolidated by constant review and usage. The development of specialised and more specific knowledge and understanding emerges level 3 where half-modules, seen for the first time, allow in-depth focus on advanced topics. In particular, topics 3 (Object-oriented and related databases), 5 (Concepts underpinning World-Wide Web technology and web-based application development.), 6 (Electronic commerce) and 9 (The	

B. Intellectual Skills

Intellectual Skills	*Teaching/Learning Methods and Strategies*	*Assessment*
1. Critical Thinking 2. Analysis 3. Synthesis of different types of information 4. Evaluation 5. Problem Solving 6. Appreciate problem contexts 7. Balance conflicting objectives	At all levels students are required to bring together knowledge and skills acquired in several modules and hence determine new ways of working. As the student progresses, the need to synthesise (3) ever-greater volumes of information and approaches into a coherent approach is developed and consequently so is their critical thinking (1). At level 1 Analysis (2), Evaluation (4) and Problem Solving (5) are developed on small-scale problems in various programming activities in a number of modules. Here the focus is on understanding the problem and then solving it free from the environmental implications of real-world problems and without the need to examine alternatives and to balance conflicting goals. At level 2 there is a move away from small-scale problems to the design of larger scale systems. With this comes the need to evaluate (4) alternative methods and designs and to balance conflicting objectives (7). Level 3 sees the move to specific application examples and with it the need to appreciate problem contexts (6) is developed as well as striking the right balance when facing conflicting objectives (7).	Programming of complex software requires demonstration of all of the intellectual skills. At level 1 the focus in programming coursework assessment, undertaken in a number of modules, is on the skills of Analysis (2), Evaluation (4) and Problem Solving (5). At levels 2 and 3 this branches out to include all the remaining skills. Many of the coursework assessments and exam papers include elements of programming work. Independent reading is used to enable students to focus on their own areas of interest and in the process asses skills 1-4 in the submitted reports, essays and exam answers. Design-work, even when not implemented in a programming language, requires demonstration of skills 1,2,5,6,7 and a number of coursework assessments and exam questions are devoted to such work. Finally, all of the examinations assess skills 1-4 whist skills 5-7 are covered in many exams.

C. Subject, Professional and Practical Skills

Subject/Professional/Practical Skills	*Teaching/Learning Methods and Strategies*	*Assessment*
Students will be able to: 1. Write programs that conform to designs 2. Create high-level and low-level designs that correspond to stated requirements 3. Design databases to meet application requirements 4. Create user interfaces for a variety of applications 5. Perform adequate tests on programs 6. Know how to use existing components and frameworks to build new applications 7. Build web-based systems 8. Employ a range of tools and notations to support the activities listed above: e.g. editors, compilers, design workbenches, HTML, CGI, Java etc..	Throughout the program, the skills listed are developed through a combination of theoretical discussion, practical laboratory based work, classroom based tutorial exercises and directed self-study. Many of the skills listed (1,2,3,5,6,8) are introduced at level 1 and then drawn into sharper focus at levels 2 and 3. The general teaching/learning method is therefore to impart these practical/professional skills by a process of moving from an overview of what is required to a specific application of an individual skill at a higher level. Some very specific skills (4, 7) are introduced at level 3. These are underpinned by the more generalised capabilities (1, 8) that are practised throughout the levels in most of the modules that contribute to the award.	The possession of these skills is demonstrated both by the development of a practical piece of coursework (software) and by examination. The practical nature of the skills to be acquired means that some are specifically addressed by particular modules (3, 4, 6, 7). The more generic skills (1,2,5,8) are assessed across the modules. Skills such as conformance to design and requirements (1, 2) and the construction of adequate testing strategies (5) are fundamental to professional software development of any sort and thus contribute to the assessment of all the practical work produced.

D. Transferable Skills and Other Attributes

Transferable Skills and Other Attributes	*Teaching/Learning Methods and Strategies*	*Assessment*
1. Communication skills**:** to communicate orally or in writing, including, for instance, the results of technical investigations, to peers and/or to "problem owners".	1. Skill one is developed through a variety of methods and strategies including the following: ♦ Students maintain laboratory log books ♦ Students participate in electronic conferences, workshops, and groupwork sessions. ♦ Students participate in discussion tutorials ♦ Students present research topic findings in tutorials ♦ Students participate in individual tutorials	These skills are demonstrated in a variety of contexts including • examination • poster presentation. individual and group projects • Practical assignments • Portfolio of exercises In addition skill two is assessed by both peers and tutors.
2. 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.	2. Skill two is developed through a variety of methods and strategies including the following: ♦ Students conduct self-managed practical work ♦ Students participate in practically-oriented tutorial laboratory sessions ♦ Students work through practical work-sheets in teams ♦ Students practice design and programming	
3. IT Skills in Context (to use software in the context of problem-solving investigations, and to interpret findings)	3. Skill three is developed widely throughout the programme.	
4. Problem formulation: To express problems in appropriate notations.	4. Skill four is developed through a variety of methods and strategies including the following: ♦ Students develop problem solving programs ♦ Students practice design and programming ♦ Students sketch designs of larger systems	

5. 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.	5. Skill five is developed through a variety of methods and strategies including the following: ♦ Students are encouraged to practice programming to extend their skills ♦ Students develop problem-solving programs ♦ Students are encouraged to research relevant topics ♦ Students are encouraged to use online facilities to discover information	
6. Comprehension of professional literature: to read and to use literature sources appropriate to the discipline to support learning activities.	6. Skill six is developed through a variety of methods and strategies including the following: ♦ Students are encouraged to access online material	
7. 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.	7. Skill seven is developed through a variety of methods and strategies including the following: ♦ Students develop a database system in laboratory sessions	

Section 4: Programme Structure

The programme is structured to allow students to select as wide as possible a choice of modules. Subject to the requirements of the Modular Assessment Regulations, students may select any combination of modules from a wide range, as shown below. However, in order to assist students entering at level one to select an appropriate programme of study, they are strongly advised to adopt the 'recommended pathway' as shown. At level 1 it introduces fundamental computing skills and contextual considerations, at level 2 these are developed providing techniques for and practice in the construction of large software applications. Finally, at level 3, a series of modules are prescribed that provide a thorough coverage of techniques required in the construction of modern component-based and internet applications. Adoption of the recommended pathway provides a coherent core of fundamental software development knowledge and techniques yet still leaves plenty of scope for choosing modules to capitalise on an individual's strengths and interests.

BSc (Hons) Computing
Optional Pathway

Year 1
Option: 1 120 CREDITS
Year 2
Option: 2 120 CREDITS
Year 2P
Option: Placement 0 - 120 CREDITS
Year 3
Option: 3 120 CREDITS

Option: 1 choose from:	
ILP	Institution Language Programme
UFCE46-20-1	Introduction to Program Development
UFCE47-20-1	Systems Development
UFIE8W-20-1	Information Technology
UFIE8Q-20-1	Information Systems Application Contexts
UFCE48-20-1	Computer Science Concepts
UFQEFY-20-1	Analytical Modelling
UFIE8T-20-1	Information Systems Development 1
UFCEKN-20-1	Data Modelling and Databases
UFCEMQ-20-1	Computer Crime and Digital Evidence
UFCEKR-20-1	Media Technologies
UFCE3H-20-1	Computational Intelligence
UPSNL3-20-1	Work, Organisations and Society 1
UFIE7W-20-1	Introduction to Web-based Information Systems
UFIE96-20-1	Information Systems Development and Practice 1
UMOCA8-20-1	Management and Organisational Behaviour
UFQEGC-20-1	Data Analysis
UFIEQQ-20-1	Informing & Communicating in Practice

UFQELG-20-1	Linear Algebra and Calculus
UFEEHV-20-1	Computer Systems
UMSCBQ-20-1	Global Business Context

Option: 2 choose from:	
ILP	Institution Language Programme
UFCE4B-20-2	Software Design
UFCE4A-20-2	Data Structures and Databases
UFEEHX-20-2	Computer Networks & O/S
UFCE49-20-2	Software Engineering
UFCE4C-20-2	Declarative Programming
UFCE4D-20-2	Symbolic Processing
UFCE4E-20-2	Subsymbolic Processing
UFCE4F-20-2	Graphics Programming
UFIE7U-20-2	eBusiness
UFIE84-20-2	Web Design
UFIE8P-20-2	Information Systems Practice 2
UFIE8R-20-2	Information Systems in the Human Context
UFIE8U-20-2	Information Systems Development 2
UFQEFW-20-2	Discrete Mathematics
UFIE97-20-2	Information Systems Development and Practice 2
UFIE9A-20-2	Human-Computer Interaction
UFIE9B-20-2	Project Management
UFIE9C-20-2	Information in Action
UFCEKP-20-2	Client-Server Programming
UFIEKG-20-2	Data, Schemas and Applications
UJQTD5-20-2	Science in Court
UFIEK4-20-2	Computing , Audio & Music
UFIEKE-20-2	Content Design
UFCEKX-20-2	Java for Sound and Music
UFCEKS-20-2	Multimedia Authoring
UFIEK3-20-2	Moving Image Technology
UFEEJ3-20-2	Introduction to Real time Systems Development

Option: Placement choose from:	
UFPEJH-120-P	Industrial Placement

Option: 3 **Not already taken level 1 or 2 options – subject to a maximum of 20 credits at level 1 or 2 choose from:**	
ILP	Institution Language Programme
UFCE4Y-20-3	Component Based Development
UFCE4P-10-3	Object Oriented Databases
UFCE4X-10-3	Software Technologies for the Web
UFCE4T-10-3	Interface Engineering
UFIE86-10-3	E-Business Special Interest Groups
UFCE3B-40-3	Computing Project

UFCE4J-20-3	Formal Language Processing
UFCE4K-20-3	Design and Analysis of Algorithms
UFCE4L-20-3	Software Engineering Project
UFCE4Q-10-3	Distributed and Parallel Databases
UFCE4R-10-3	Intelligent Systems
UFCE4S-10-3	Requirements Engineering
UFCE4W-10-3	Advanced Databases
UFIE85-20-3	Internet Systems Group Project
UFIE95-20-3	Information Systems in Society
UFIE8V-20-3	Information Systems Development 3
UFIE8Y-20-3	Information Systems Dissertation
UFIE98-20-3	Information Systems Development and Practice 3
UFIE99-40-3	Information Systems Practice 3
UFIEKV-10-3	Digital Stories
UFEEJ6-10-3	Advanced Distributed Systems
UFIE9H-10-3	Computing and Law
UFIE9K-10-3	Professional, Legal and Commercial Issues
UFIE9L-10-3	Information Technology Audit
UFIE9M-10-3	Technical Writing and Editing
UFIE9R-10-3	Knowledge in Organisations
UFIE9T-20-2	Creativity and Design
UFIEKJ-20-3	Professional, Ethical and Policy issues
UFIE9S-20-3	Multimedia Systems: Contexts & Applications
UFIE9G-10-3	Information Systems in Complex Organisations
UFCEMR-20-3	Forensic Computing Practice
UFEEJA-10-3	Advanced Operating Systems Programming
UFCE3K-20-3	Machine Learning
UFCEMU-20-3	Cryptography and Coding Systems
UFIE8M-20-3	Information Systems Management
UFIEMF-20-3	Data Mining Methodology
UFCEKQ-10-3	Internet Security
UFCEKW-20-3	Application Development Project
UFCEKT-20-3	3D Modelling and Animation
UFCEKU-20-3	Games Programming
UFIEKK-10-3	Text & Markup Languages
UFCEMV-20-3	Computer & Network Security
Choose modules from	Option: 1
Choose modules from	Option: 2

Note: Structures are indicative and subject to change

PLEASE NOTE: REFER TO THE FACULTY ON-LINE INFORMATION SYSTEM FOR UP-TO-DATE STRUCTURE INFORMATION

http://www.cems.uwe.ac.uk/exist/index.xql

BSc (Hons) Computing
Recommended Pathway

Year 1				
UFCE46-20-1 Introduction to Program Development	UFCE47-20-1 Systems Development	UFIE8W-20-1 Information Technology	UFIE8Q-20-1 Information Systems Application Contexts	Option: 1 40 CREDITS

Year 2			
UFCE4B-20-2 Software Design	UFCE4A-20-2 Data Structures and Databases	UFEEHX-20-2 Computer Networks & O/S	Option: 2 60 CREDITS

Year 2P
Option: Placement 0 - 120 CREDITS

Year 3					
UFCE4Y-20-3 Component Based Development	UFCE4P-10-3 Object Oriented Databases	UFCE4X-10-3 Software Technologies for the Web	UFCE4T-10-3 Interface Engineering	UFIE86-10-3 E-Business Special Interest Groups	Option: 3 60 CREDITS

Option: 1 choose from:	
ILP	Institution Language Programme
UFCE48-20-1	Computer Science Concepts
UFQEFY-20-1	Analytical Modelling
UFIE8T-20-1	Information Systems Development 1
UFCEKN-20-1	Data Modelling and Databases
UFCEMQ-20-1	Computer Crime and Digital Evidence
UFCEKR-20-1	Media Technologies
UFCE3H-20-1	Computational Intelligence
UPSNL3-20-1	Work, Organisations and Society 1
UFIE7W-20-1	Introduction to Web-based Information Systems
UFIE96-20-1	Information Systems Development and Practice 1
UMOCA8-20-1	Management and Organisational Behaviour
UFQEGC-20-1	Data Analysis
UFIEQQ-20-1	Informing & Communicating in Practice
UFQELG-20-1	Linear Algebra and Calculus
UFEEHV-20-1	Computer Systems
UMSCBQ-20-1	Global Business Context

Option: 2 choose from:	
ILP	Institution Language Programme
UFCE49-20-2	Software Engineering
UFCE4C-20-2	Declarative Programming

UFCE4D-20-2	Symbolic Processing
UFCE4E-20-2	Subsymbolic Processing
UFCE4F-20-2	Graphics Programming
UFIE7U-20-2	eBusiness
UFIE84-20-2	Web Design
UFIE8P-20-2	Information Systems Practice 2
UFIE8R-20-2	Information Systems in the Human Context
UFIE8U-20-2	Information Systems Development 2
UFQEFW-20-2	Discrete Mathematics
UFIE97-20-2	Information Systems Development and Practice 2
UFIE9A-20-2	Human-Computer Interaction
UFIE9B-20-2	Project Management
UFIE9C-20-2	Information in Action
UFCEKP-20-2	Client-Server Programming
UFIEKG-20-2	Data, Schemas and Applications
UJQTD5-20-2	Science in Court
UFIEK4-20-2	Computing , Audio & Music
UFIEKE-20-2	Content Design
UFCEKX-20-2	Java for Sound and Music
UFCEKS-20-2	Multimedia Authoring
UFIEK3-20-2	Moving Image Technology
UFEEJ3-20-2	Introduction to Real time Systems Development

Option: Placement choose from:	
UFPEJH-120-P	Industrial Placement

Option: 3 choose from:	
ILP	Institution Language Programme
UFCE3B-40-3	Computing Project
UFCE4J-20-3	Formal Language Processing
UFCE4K-20-3	Design and Analysis of Algorithms
UFCE4L-20-3	Software Engineering Project
UFCE4Q-10-3	Distributed and Parallel Databases
UFCE4R-10-3	Intelligent Systems
UFCE4S-10-3	Requirements Engineering
UFCE4W-10-3	Advanced Databases
UFIE85-20-3	Internet Systems Group Project
UFIE95-20-3	Information Systems in Society
UFIE8V-20-3	Information Systems Development 3
UFIE8Y-20-3	Information Systems Dissertation
UFIE98-20-3	Information Systems Development and Practice 3
UFIE99-40-3	Information Systems Practice 3

UFIEKV-10-3	Digital Stories
UFEEJ6-10-3	Advanced Distributed Systems
UFIE9H-10-3	Computing and Law
UFIE9K-10-3	Professional, Legal and Commercial Issues
UFIE9L-10-3	Information Technology Audit
UFIE9M-10-3	Technical Writing and Editing
UFIE9R-10-3	Knowledge in Organisations
UFIE9T-20-2	Creativity and Design
UFIEKJ-20-3	Professional, Ethical and Policy issues
UFIE9S-20-3	Multimedia Systems: Contexts & Applications
UFIE9G-10-3	Information Systems in Complex Organisations
UFCEMR-20-3	Forensic Computing Practice
UFEEJA-10-3	Advanced Operating Systems Programming
UFCE3K-20-3	Machine Learning
UFCEMU-20-3	Cryptography and Coding Systems
UFIE8M-20-3	Information Systems Management
UFIEMF-20-3	Data Mining Methodology
UFCEKQ-10-3	Internet Security
UFCEKW-20-3	Application Development Project
UFCEKT-20-3	3D Modelling and Animation
UFCEKU-20-3	Games Programming
UFIEKK-10-3	Text & Markup Languages
UFCEMV-20-3	Computer & Network Security

Note: Structures are indicative and subject to change

Section 5: Entry Requirements

The university's minimum requirements for entry to a degree apply to this programme. In addition entrants are required to have evidence of achievement in Mathematics at GCSE Grade C or equivalent.

Section 6: Assessment Regulations

The Modular Assessment Regulations apply to this programme

Section 7: Student Learning: Distinctive Features and Support

The defining characteristic of this degree is that it offers a highly flexible programme of study. The degree offers a recommended core set of subjects and a large number of options from the Computer Science and Information Systems Fields from which to choose the remainder of the programme. Because of the flexibility of the programme, students who might not quite achieve the entry grades required to follow a specialist degree may be admitted and they can tailor their programme to play to their strengths. When choosing options students may decide to follow a set of identified modules to allow them to focus on particular areas of interest such as Multimedia, Internet Technology, Computer Science, Information Systems, Software Engineering or Artificial Intelligence. Those performing particularly well in a specialist area may be eligible to transfer to a specialised degree programme.

Within the Faculty of Computing Engineering and Mathematical Sciences, student learning is supported in the following ways:

- through provision of a large Open Access Laboratory (3P10) containing 50 machines that provide students with access to a wide range of computer-based applications;
- through provision of a number of other, frequently available, computer laboratories that provide similar access;
- through provision of the CEMS System Support Helpdesk that provides a range of support for learning to students including:
  - support for a wide range of applications used by the students;
  - help in the form of Assistants who are trained to resolve many common student problems;
  - and help in the form of a large set of “Helpsheet Documents”, developed over a number of years, that cover a variety of common student requests for information.

Section 8 Reference Points/Benchmarks

The QAA Subject Benchmark Statement for Computing was published in 2000, and is applicable to this proposal. The design team has considered them in drawing up the structure of the proposed degree, and is of the view that the proposal falls clearly within the scope of the benchmarks, as regards curriculum, teaching and learning, and the benchmarking standards themselves.

The benchmarks (para 2.1) identify a range of types of degrees in computing, from (at one extreme) a programme which "*covers a wide range of topics spanning the entire area of computing"* to (at another extreme) programmes which "*take one very specific aspect of computing and covers it in great depth"*. This proposal is closer to the first of these extremes. Nevertheless it does allow students to recognise the importance of speciality areas, in particular through the choice of Level 2 and 3 modules.

The benchmarks recognise (para 3.3) that diversity of provision is to be encouraged. This programme meets the ambition of the faculty to provide programmes suitable for as wide as possible a range of entrants, by offering a very wide choice of options at Levels 2 and 3, and by enabling entry to applicants offering a wider range of entry qualifications than is the case for most other degrees in computing.

The benchmarks also contain (section 5) statements of the standards expected of graduates at both modal and threshold levels. The team is of the view that graduates of the proposed programme will be able to meet the required standards.