Agència per a la Qualitat del Sistema Universitari de Catalunya

SUBJECT BENCHMARK STATEMENT COMPUTING

Generalitat de Catalunya

ENQA eqar ISO 9001 ISO 27001

C. dels Vergós, 36-42

08017 Barcelona

First edition: September 2017

Guide approved by the Institutional and Programme Assessment Committee on 3 July 2017.

SUMARI

1. INTRODUCTION AND OBJECTIVES

The number of bachelor and master's degrees in Computing offered by universities in Catalonia in 2016 is increasing at a faster rate than ever before. New programmes currently undergo an ex-ante accreditation procedure, known as validation, and modifications may be proposed in subsequent annual progress reports. The final stage in programme accreditation is a site visit by an external panel of experts. The fact that there are three different stages in the review process has become increasingly complex and this is perhaps not absolutely necessary. During 2015, a total of 246 degrees were reviewed, with 55 sites visits to institutions (a 57% increase compared to 2014) involving 59 external panels of experts. The outcome was favourable in 93% of all validations and 98% of all modification proposals presented in 2015, which implies acceptance. In this context, this document serves as a self-validation instrument for autonomous use by institutions to enable and ensure that greater emphasis in the Agency's work is placed on the accreditation stage.

This document is the Subject Benchmark Statement by AQU Catalunya on Computing that defines what can be expected of graduates in the subject, in terms of what they might know, do and understand on completing their studies. It refers to bachelor and master's degrees in Computing.

Subject Benchmark Statements are used as reference points in the design, delivery and review of academic programmes. They provide general guidance for articulating the learning outcomes associated with the programme but are not intended to represent a set curriculum in a subject or to prescribe set approaches to teaching, learning or assessment. Instead, they allow for flexibility and innovation in programme design within a framework agreed by the subject community.

This document was inspired by the QAA Subject Benchmark Statement on Computing 2016 (hereinafter QAA-SBS). This is particularly the case in sections 2 and 3 where the initial text was complete and customised according to the specific context here. Sections 6 and 7, together with Annex I, are specific to the university framework in Catalonia. Issued in 2016, this is the first edition of this Benchmark Statement by AQU Catalunya.

You may want to read this document if you are:

- Involved in the design, delivery, and review of programmes of study in Computing and/or related subjects.
- A prospective student thinking about studying Computing or if you are currently a student of the subject, to know what is expected of a graduate in Computing.
- An employer, to find out about the knowledge and skills generally expected of a graduate in Computing.

Computing is concerned with the understanding, design and exploitation of computation and computer technology, one of the most significant advances in recent decades. It is a discipline [QAA-SBS] that:

- Blends elegant theories (including those derived from a range of other disciplines such as mathematics, engineering, psychology, graphic design or well founded experimental insight) with the solution of immediate practical problems.
- Underpins the development of small- and large-scale, secure, reliable, and usable systems that support organisational goals.
- Helps organisations and individuals in their everyday lives.
- Is pervasive, ubiquitous, and diversely applied to a range of applications, in a way that important components are often invisible to the naked eye.

Computing promotes innovation and creativity. It requires a disciplined approach to problem solving. It approaches design and development through selection from alternative possibilities justified by carefully crafted arguments. It controls complexity first through abstraction and simplification, and then by the integration of components.

The diversity of Computing is reflected in the varied titles and curricula that higher education providers have used for Computing-related degree courses worldwide.

Computing degrees will continue to evolve in response to developments in the subject and future human needs. This Benchmark Statement therefore concentrates on general graduate outcomes and neither specifies a core curriculum for Computer Science nor includes a syllabus or body of knowledge. The curriculum documents from the Association of Computing Machinery (ACM), IEEE CS, ANECA and QAA can be used as a source of guidance on possible curriculum content. See for instance:

- http://www.acm.org/education/curricula-recommendations
- http://www.computer.org/web/peb/curricula
- http://www.aneca.es/media/150388/libroblanco_jun05_informatica.pdf
- http://www.qaa.ac.uk/en/Publications/Documents/Subject-benchmark-statement-Computing.aspx.pdf

This document is for use in the ex-ante accreditation (validation) of degrees in Computing. It should also serve as reference for the ex-ante accreditation of the Computing part of interdisciplinary curricula (as defined in Section 2) at both undergraduate and master's level.

2. NATURE AND EXTENT OF COMPUTING

This section is based on a summary of Section 2 of the document [QAA-SBS], which gives a precise definition of the discipline.

Computing delivers innovative solutions to problems and drives technological, economic and social progress. Computing is evolving at a rapid rate and touches upon all aspects of life as it does so. Computing as a discipline consists of several central elements: mathematics, fundamentals of computation, and realisation of computing systems in both hardware and software.

Computing graduates apply their understanding, skills, knowledge, and experience to create social, scientific and economic value by building secure, reliable, and usable systems. Computing includes aspects that overlap with areas of interest to a number of adjacent subjects. These areas include engineering (problem solving, constrained optimisation, simulation), electronics and physics (device-level development of computing components), mathematics (boolean logic for reasoning about software, and theoretical models of computation, numeric algorithms), business (information services), linguistics, and even philosophy, physiology, and psychology (human computer interaction and aspects of artificial intelligence).

Computing is a rigorous academic discipline in its own right and also facilitates and supports a wide range of other disciplines from computational biology to computational social science.

The concept of **computational thinking** is central to the discipline. According to Jeannette M Wing (2008), "Computational thinking is using abstraction and decomposition when attacking a large complex task or designing a large complex system. It is separation of concerns. It is choosing an appropriate representation for a problem or modelling the relevant aspects of a problem to make it tractable. It is using invariants to describe a system's behaviour succinctly and declaratively".

The term computing applies to an increasingly diverse set of degree programmes all based on the foundations of Computer Science. This Benchmark Statement identifies Computer Science, Computer Engineering, Software Engineering, Information Technology, and Information Systems as discipline areas and outlines the content covered by these.

- Computer Science provides the necessary knowledge to understand, analyse, and build computational systems. Among other things, it includes mathematical tools, algorithmic thinking on recursive, distributed, and parallel systems, the relationships between the concepts of specification, programmes and data as well as the power of transformation and proof, with simplicity and elegance being recognised as useful concepts and principles. Simplicity is normally expressed through the ability to specify, design, write, and verify elegant algorithms and programmes.
- Computer Engineering is concerned with the realisation of computer science fundamentals in computer hardware, including the use of hardware design, software tools, simulation tools, and the ability to understand the construction of, and make best use of, computational devices, interfaces and protocols.
- Software Engineering is concerned with the building and realisation of software systems. It includes problem definition, specification (including formal specification), design, simulation, implementation, verification, maintenance and software testing, the understanding of the range of possible options and an appreciation of design trade-offs, and the study of security and safety-critical systems. Normally these should result in the ability to create fit-for-purpose software in a variety of application domains.
- Information Technology is concerned with the application of computing technologies to other domains. It includes the selection and application of software and hardware and the integration of components to provide solutions in a variety of application domains. These

are normally expressed in the ability to deliver a computer-based system as a solution to desired needs.

- Information Systems is concerned with the modelling, codification and storage of data and information for later retrieval and analysis, including data management, data architecture, databases, information modelling, data mining, indexing and searching, and machine learning, with an emphasis on usability and man-machine interfaces. Normally these are expressed in the ability to construct usable systems that acquire, codify, store and transmit information, and are able to reveal previously hidden patterns and relationships in large data sets.

The need for interdisciplinary skills is ever greater. Computing has become closely connected with other areas of engineering in a myriad of new computer-intensive instruments and systems that are computer-intensive and call for new IT solutions such as microcontrollers, data analysis and machine learning, including cars and transport systems, mobile device and phone apps, domotics, photography, appliances like vacuum cleaners, drones, etc. These are defining a new context in which requirements are constantly evolving. Many higher education providers deliver degrees that focus on specific aspects of computing in society, for example computer networking, games, multimedia, health informatics, Biocomputing. Many new degrees are and will be computing-related. These curricula are defined as Computing when their content is informed and inspired by the discipline areas listed above. In addition, computing will be taught in many other joint and interdisciplinary programmes. This Benchmark Statement is for use as the reference point for the Computing component of such undergraduate and master's programmes.

Programme designers, students and employers will need to be aware of this spectrum of programme identities, but behind such variation there are **key ideas that can be expected to characterise any degree programme in Computing**:

- The concept of computational thinking, the recognition of its main elements and the relevance of these to everyday life.
- The balance of practice and theory in such a way that a clear understanding of its underlying principles supports any practical activity.
- The ability to analyse (and develop, if necessary) computing and information systems. A clear understanding of the system and its operation gives a deeper knowledge of what the system does or the way(s) in which it is used.
- Programmes should not be based on short-life knowledge. Instead of focusing on current industrial needs, graduates should be able to solve future and unknown practical problems. Adaptability is an essential requirement in Computing degrees.

3. GENERIC AND SPECIFIC SKILLS

This section includes a summary of concepts from Section 3 of the document [QAA-SBS], which presents a precise and structured view of computing-related skills. It is also based on the Qualifications Framework for Spain (MECES)[1]. Computing students are expected to develop a wide range of skills, which can be divided into three broad categories:

- Computing-related cognitive skills.
- Computing-related practical skills.
- Generic skills for employability.

Cognitive, practical, and generic skills need to be placed in the context of the programme of study as designed by the higher education provider and/or the possible pathways selected by the individual student. There is an implicit interplay between these identified skills within and across these three categories. The extent to which students acquire these skills depends on the emphasis of individual degree programmes. Skills at undergraduate level include the acquisition of a general training in several disciplines, aimed at preparing students for professional roles. At master's level, skills include the acquisition by the student of an advanced training (either specialized or multidisciplinary) aimed at professional specialisation and/or at fundamental aspects to start research activities.

Computing-related cognitive skills include (parentheses with the letter M imply that the content applies only for a master's programme):

- Computational thinking, including its relevance to everyday life.
- Knowledge and understanding: demonstrate (M: advanced) knowledge and understanding of:
  - Essential facts, concepts, principles (M: and scientific theories) related to computing,
  - Technology constraints, orders of magnitude for the main computing system parameters, and
  - Computer applications.
- Requirements, practical constraints and computer-based systems. These include embedded and distributed computer systems, data, and security in their context: recognise and analyse criteria and specifications appropriate to specific problems, and plan strategies for their solutions.
- Modelling: use such knowledge and understanding in the modelling and design of (M: advanced and complex) computer-based systems. Prediction and the understanding of trade-offs.

[1] Qualifications Framework for Spain (MECES):
http://www.cualificacionesprofesionales.info/sites/default/files/legislacion/Marco%20Espa%C3%B1ol%20de%20Cualificaciones.pdf

- Critical evaluation and testing: analyse the extent to which a computer-based or computer-intensive system meets the criteria defined for its current use and future development.
- Methods and tools: deploy (M: appropriate theory,) practices and tools for the specification, design, implementation, verification, and evaluation of computer-based systems.
- Professional considerations: recognise the professional, economic, social, environmental, and ethical issues involved in the sustainable exploitation of computer technology and be guided by the adoption of appropriate professional, ethical and legal practices.

Computing-related practical skills include:

- The ability to critically evaluate and analyse complex problems, including those with incomplete information, and devise appropriate solutions.
- The ability to specify, design and construct reliable, secure, and usable computer systems.
- The ability to (M: predict and) evaluate systems in terms of quality attributes and possible trade-offs presented within the given problem.
- The ability to plan and manage projects to deliver computing systems within constraints or a user's incomplete requirements, timescale and budget. (M: The ability to predict and control the evolution of complex situations by developing new and innovative working methodologies adapted to the specific scientific, technological, or professional level, generally within multidisciplinary teams.)
- The ability to recognise any risks as well as security and safety aspects that may be involved in the deployment of computing systems within a given context. (M: The ability to reflect upon social responsibility and ethical issues). Security is becoming crucial everywhere.
- The ability to deploy effectively the tools used for the construction and documentation of computer applications, (M: and even to develop new and innovative tools) with particular emphasis on understanding the whole process involved in the effective deployment of computers to solve practical problems.

Generic Skills for Employability: students are expected to develop a wide range of generic skills to ensure they become effective in the workplace, to the benefit of themselves, their employers and the wider economy. Students who develop generic skills and are able to evidence and demonstrate such skills will gain significant advantage when seeking employment. It is the responsibility of higher education providers to provide every student the opportunity to acquire and evidence generic skills; it is the responsibility of the student to make the most of that opportunity. These skills include:

- Intellectual skills: critical thinking; making a case; numeracy and literacy; information literacy. The ability to construct well argued and grammatically correct documents. The ability to locate and retrieve relevant ideas, and ensure these are correctly and accurately referenced and attributed.

- Self-management: self-awareness and reflection; goal setting and action planning; independence and adaptability; acting on initiative; innovation and creativity. The ability to work unsupervised, plan effectively and meet deadlines, and respond readily to changing situations and priorities.
- Oral skills in writing and electronic interaction: reflection and communication: the ability to succinctly present rational and reasoned arguments that address a given problem or opportunity, to a range of audiences, and also in English. (M: The ability to present scientific, technology and innovative results).
- Team working and management: the ability to recognise and make best use of the skills and knowledge of individuals, to collaborate, and to be able to identify problems and desired outcomes and negotiate to mutually acceptable conclusions. The ability to understand the role of a leader in setting direction and taking responsibility for actions and decisions. (M: Leadership ability).
- Contextual awareness: the ability to understand and meet the needs and ways of thinking and acting of individuals, business and the community, and to understand how workplaces and organisations are governed. The ability to collect and interpret data and information to argument results and conclusions in a solid way. When necessary and appropriate, these data will include the analysis of social, scientific, and ethical aspects.
- The ability to discuss with people who have different points of view and come from different domains and traditions.
- The ability to listen and understand how people of other scientific and engineering disciplines think and talk, and adapt accordingly. This is essential in multi-disciplinary teams.
- (M: Scientific approach: The ability to evaluate and select appropriate scientific theories and precise methodologies in order to make judgments based on incomplete or limited information.)

4. TEACHING, LEARNING AND ASSESSMENT

Students achieve an understanding of computing through:

- A solid base that allows them to build mathematical and computational models with a good structure and to implement such models in practice.
- A significant exposure to practical coursework, substantial individual and group-project work, and problems and projects taken from the real world.
- Learning to be innovative in problem solving and to evaluate, select, justify, and communicate.

- Self-learning. For instance, using and deploying solutions by using similar but not the same tools (i.e. different programming languages or development environments), and access to existing high-quality Internet courses produced by other prestigious universities and senior academic staff of standing, under the advice and supervision of tutors and faculty staff.

Curriculum design should be based and informed on current developments and trends, reflecting appropriate recent research and industrial and business practices, together with an understanding of potential graduate destinations [QAA-SBS].

Student projects should include design-and-build, consultancy, and research-led tasks to develop both independence of thought and the ability to work effectively in a team. Projects should include computing-related case studies, linking them to applications that students will encounter after graduation. Teaching and learning needs to be placed within the context of social, ethical, legal, professional and economic factors relevant to computing.

The curriculum should include a major activity allowing students to demonstrate their ability in applying practical and analytical skills as a whole: this should take the form of a final-year project.

Students should evolve during their studies from classical learning practices to computer-based problem solving, using optimisation and related tools to solve complex problems.

Work-based learning, including activities such as industrial internships, is an essential part of the curriculum. These should be properly integrated in terms of the preparation of students before the activity, debriefing, building on the experience afterwards and assessment. Internships offer the opportunity for students to apply and validate their learning and skills in the context of the real world and provide early exposure to professional skills and competences.

Access to software and communication facilities enables students to extend their horizons. Apart from the usual range of programming languages, these include software libraries, graphics packages, computer-aided software engineering tools, integrated development environments, theorem provers, project management software and planning systems. These may be provided as appropriate to the programme of study [QAA-SBS].

Computer Engineering differs from the remaining discipline areas (Computing, Software Engineering, Information Technology and Information Systems) in that there is a significant difference between bachelor and master's degrees. A bachelor's degree in Computer Engineering should enable students to understand the construction of computer systems and make best use of them. A Master's degree in Computer Engineering should add the capability to design and verify these systems and their parts.

Master's degrees should include an initial selection of student candidates based on academic criteria and objectives. A reasonable student structure would usually consist of around 70% of undergraduate students and 30% of master students.

5. COMPUTING DEGREES AS PREPARATION FOR PROFESSIONAL PRACTICE

There are many different types of Computing degree programme, but all are designed to equip their graduates with knowledge, understanding and skills that will enable them to begin a professional career in some aspect of computing.

Not all graduates will proceed with a professional career in computing. The attributes of Computing graduates also make them attractive to many non-computing employers in sectors such as manufacturing, finance, consultancy, public services, creative industries and the arts, as well as entrepreneurs in their own right.

Computing programmes should prepare bachelor and master's students with an employability profile. On graduating, students should:

- Be able to work in complex projects and problems.
- Be able to cooperate with others in a team-working environment.
- Be able to innovate, design and develop practical and usable products and to achieve successful solutions.
- Have initiative and leadership.
- Be proactive.
- Have learned languages and social communication.
- Have interpersonal skills, including team working, negotiation, adaptive skills, and the ability to make public presentations.
- Have the ability to understand the many points of view coming from different domains, skills and traditions in a complex project. The ability to work in inter-disciplinary teams with engineers and graduates from other disciplines.
- Have learned entrepreneurship.

Master's graduates must have followed a specialised training with particular emphasis on innovation, research and leadership aspects. It is advisable for there to be connections between master's programmes and industry and other HEIs, and for programmes to be underpinned by current research being undertaken in research institutes in Computer Sciences and other areas of Engineering.

6. BENCHMARK STANDARDS AT UNDERGRADUATE AND MASTER'S LEVELS

Undergraduate degrees should preferably be generalist. They should ensure that students acquire a solid foundation that enables them to learn and understand advanced tourism-related concepts with a broad-minded perspective. The world is full of challenging and interesting engineering problems that require engineers to overcome real complications and constraints.

Undergraduate curricula should combine fundamental engineering concepts with practical examples and project-based design challenges related to computing.

Master's degrees should be specialist in nature, although some generalist Master's programmes may exist. Generalist master's degrees would usually include elements of the major areas of computing and include Computer Science, Computer Engineering, Software Engineering, Information Technology, and Information Systems.

Undergraduate level

On graduating with a degree in Computing, students should be able [QAA-SBS] to:

- Demonstrate a sound understanding of the main areas of the body of knowledge within their programme of study, with an ability to exercise critic judgement.
- Critically analyse and apply essential concepts, principles and practices of the subject in the context of loosely defined scenarios, showing effective judgement in the selection and use of tools and techniques.
- Produce work involving problem identification, the analysis, design and/or development of a system, with appropriate documentation, with recognition of the important relationships between these. The work will show problem solving and evaluation skills, draw upon supporting evidence, and demonstrate a good understanding of the need for a high quality solution.
- Demonstrate generic skills with an ability to show organised work with minimum guidance both as an individual and team member.
- Apply appropriate practices within a professional, legal and ethical framework and identify mechanisms for continuing professional development and lifelong learning.

OBJOBJOBJOBJ

Excellent students:

- Will be able to contribute and drive the analysis, design and/or development of systems that are complex and fit for purpose, and recognise the important relationships between these.
- Will be creative and innovative in their application of the principles covered in the curriculum.
- Will be able to exercise critical evaluation and review of both their own work and the work of others.

Master's level

All master's degrees should be based on a solid foundation and prepare students to undergo innovative and research activities in industries and other organisations, as well as preparing them for entrepreneurship and self-employment when establishing their own companies.

Master's degrees with a generalist perspective include the outcomes of undergraduate degrees and go beyond them to provide a greater range and depth of specialist knowledge, often within a research and industrial environment, as well as a broader and more general fundamental base. Such programmes provide a foundation for leadership.

Master's degrees in Computer Engineering may also provide a foundation for designing hardware components.

Some master's degrees may have a more specific and specialized focus for students and professionals wanting to enhance or upgrade their knowledge in specific areas.

Proposals for multi-disciplinary master´s programmes involving a certain level of computing knowledge (such as design, bioinformatics and others) will clearly become more commonplace in the future. The analysis of proposals will need to be made on a specific and individual basis, with Computing components complying with this benchmark statement.

A Master's thesis should serve to ensure that a student has acquired all of the programme's intended skills and competences. Master's theses should therefore include a series of developments with both practical as well as fundamental components.

7. SELF-BENCHMARKING

This Benchmark Statement is for use as a reference point for the design, delivery and review of courses and programmes of study. It offers general guidelines for establishing the learning outcomes associated with these programmes. Institutions that are either starting new programmes in Computing or in the process of modifying existing ones are asked to go through the checklist[2] in Annex 1 for self-benchmarking their curricular project. This list is divided into six main sections:

A. The environment: a programme's design should take into account all of the questions provided in this Section, together with an analysis of the answers provided in each specific case.

B. Developing and producing the curriculum for Computing: a programme's design should benefit from reflection on the items in this section.

C. The resources: based on the answers given to the questions in this section, curriculum designers should evaluate the suitability of the resources in relation to the intended learning outcomes. These are not necessarily equal for bachelor and master's degrees.

D. Employability: the assessment of students' skills should take into account what is said about the items in this section.

E. Quality control and assurance: a programme's design should include the necessary factors to elicit a positive response to the items in this section.

---

[2] For reference, see also: Developing a Competency-Based Curriculum for Higher Education Project ESF: http://ospe.utu.fi/english.php

F. Learning and teaching: a programme's design should include the necessary factors to elicit a positive response to the items in this section.

ANNEX 1: CHECKLIST QUESTIONS

1. Title
    a. Is the title of the programme appropriate and clear as regards the labour market?
2. Justification
    a. Are the areas of professional practice mainly considered for graduates present in industry and economic activities in your region? Do you prepare your graduates for the international labour market? Have you performed an analysis of potential graduate destinations in your region and worldwide?
3. Skills and competences
    a. Do you take into account the professional profiles sought after by business and industry in the region, as well as at national and international level?
    b. Are graduates able to recognize, analyse, and discuss criteria and specifications appropriate to specific problems, as well as designing appropriate strategies for their solution by team working? Are they able to configure and size computer and cluster configurations to process data at a given rate?
    c. Are graduates able to deploy appropriate theory, practices and tools for the specification, design, implementation, and evaluation of new products that respond to specific user (and market) needs?
    d. Are graduates able to understand, construct, and make the best use of computational devices, interfaces and protocols?
    e. Are graduates able to construct, acquire, code, store, organize, analyse, classify, and transmit information with the required level of safety?
    f. Are graduates able to build mathematical and computational models with a good structure and implement such models in practice? Are the students able to incorporate optimisation methods in their solutions?
    g. Do students acquire the ability to solve problems and analyse the influence of assumptions in the final result by using sensitivity analysis?
    h. Are students able to prepare well-argued and grammatically correct documents?
    i. Are students able to present succinctly to a range of audiences, not only orally but also in writing, solid arguments that address a given problem or opportunity together with an assessment of the impact of ICT solutions? Do students present the results of work to other students? How frequently?
    j. Are students able to detect situations in which computing- intensive systems can be relevant to solve everyday life problems?
    k. Are students able to recognize the professional, economic, social, environmental and ethical issues involved in the sustainable exploitation of computer technology?

l. Are students able to evaluate systems in terms of general quality attributes and possible trade-offs for a given problem? Are they capable of empirically assessing the quality of systems they have implemented?

m. Are students able to recognize any risks or safety aspects that may affect the operation of computing equipment or programmes in real practice?

4. Access and student admission

   a. Are mechanisms in place for the selection of master´s programme candidates?

5. Curriculum

   a. Are the contents of the Computing curriculum adapted to the needs of innovative industries and organisations in your region and/or worldwide?

   b. Have companies and economic agents in your region been involved in defining the Computing curriculum? In what way/s have they been involved?

   c. How have curriculum documents from ACM, IEEE CS, QAA, ANECA, and other agencies been taken into account?

   d. Have external and international experts been involved in producing the curriculum?

   e. Have you been able to prevent influences (lobbying) by faculty staff from resulting in teaching imbalances in the curriculum? Have specific procedures been established to keep these pressures and influences under control?

   f. Have the consistency and completeness of the contents been checked?

   g. Have you included specific mechanisms for a periodic review and update of the curriculum, based on new research findings and industrial practices?

   h. How is intra- and inter-university cooperation implemented? How does the curriculum describe and facilitate mobility, including at international level?

   i. Does the curriculum contain basic courses to introduce the main concepts of mathematics, electronics, etc. to students while showing their relevance in computing?

   j. How do you encourage, assess and evaluate peer-to-peer learning?

   k. Are gender perspective and diversity (multiculturalism) included in the curriculum?

   l. Does the curriculum include periods of time working in a company (or similar organisation) as placement students?

6. Human resources (teachers and staff)

   a. Is the number of teachers adequate for programme delivery and student support?

   b. Is the percentage of staff with PhD qualifications above 50% (bachelor’s)/ 70% (master´s)?

c. Is the percentage of staff with external recognition and PhD qualifications above 60%?

d. Do you have a strategy to increase this percentage (bachelor/master level)?

e. Is an annual report produced on the faculty's indexed publications?

f. Is an annual report produced on the (international, national) competitive projects led by your faculty?

g. What percentage of your teaching staff has been trained in educational skills?

7. Resources and services

a. How is student guidance embedded in the curriculum? How are the duties and responsibilities for student guidance defined? Who participates in student guidance?

b. Do students receive guidance regarding existing Internet courses offered by prestigious universities and senior academic staff of standing whereby such courses could be followed as an alternative to your institution's courses? Have academic staff in your institution been involved in selecting these courses, have they used them in a collaborative context in any way, and have any of your academic staff been involved in producing any of them?

c. What is the prospective total number of students over the next 10 years?

d. Do you have plans to upgrade resources and facilities in the future?

e. How many computers and facilities are there in student labs? What software is available for students? Is there a library for use by the students?

8. Intended learning outcomes

a. How are interpersonal skills (team working, leadership, communication, ability to make public presentations) assessed?

b. Do you assess the ability of students to work in complex projects and problems?

c. Do you assess their ability to develop practical and usable products and to achieve successful solutions?

d. Do you assess initiative and proactive skills?

e. Do you assess entrepreneurship? How?

9. Quality management system

a. Have you designed quality assurance procedures for the programme?

b. Are external agencies involved in quality assurance?

c. Is there an external ex-ante accreditation (validation) procedure to ensure that the curriculum, learning outcomes, assignments and grades are equivalent in terms of quality with Computing degrees run at other universities and higher education institutions in Catalonia?

d. Are external experts involved in quality assurance?

e. Are self-analysis and/or self-regulation techniques used to enhance quality assurance and the assessment of student work and activities?

f. Are mechanisms in place to detect excellence among students?

g. Are specific curriculum options planned for these students?

10. Delivery timeline

    a. Is the delivery of the programme realistic, given the actual and planned resources?

ANNEX 2: MEMBERSHIP OF THE AQU EXPERTS COMMITTEE

Membership of the panel of experts for the Subject Benchmark Statement for Computing (2016)

- Pere Brunet Crosa (Chair)
- Ricardo Baeza-Yates
- Gerard Berry
- Enrique Castillo
- José Duato Marín
- Ramon López de Mántaras
- Juan José Moreno Navarro
- Isidro Ramos Salavert
- Hans-Peter Seidel
- Concepción Herruzo (AQU Catalunya), Secretary

Agència per a la Qualitat del Sistema Universitari de Catalunya
September 2017 · AQU-19-2017

Agència
per a la Qualitat
del Sistema Universitari
de **Catalunya**

www.aqu.cat

@aqucatalunya