

## Course Proposal Form

Please see Page 2 for instructions on which parts of this form to complete, whom to consult with to avoid unnecessary effort, and where to send the completed form.

**Proposer(s): Cristina Adriana Alexandru**

**Date: 14/02/2020**

### Cover page: Basic permanent course information

Unless otherwise noted, items in this section are entered into EUCLID and **cannot** be changed without creating an entirely new course.

<b>Course Name</b>	Informatics 2- Software Engineering and Professional Practice
<b>Course Acronym</b> <i>(used by the School only, e.g., for the Sortable Course List)</i>	Inf2-SEPP
<b>Course Level</b> If the course is <b>only</b> available to MSc students, then it must be classed as Postgraduate. All other courses, regardless of level, are Undergraduate.	<input checked="" type="checkbox"/> Undergraduate <input type="checkbox"/> Postgraduate
<b>Normal Year Taken</b>	<input type="checkbox"/> UG1 <input checked="" type="checkbox"/> UG2 <input type="checkbox"/> UG3 <input type="checkbox"/> UG4 <input type="checkbox"/> UG5 <input type="checkbox"/> MSc
<b>Also available in years</b> <i>[This can be changed later if need be.]</i>	<input type="checkbox"/> UG1 <input type="checkbox"/> UG2 <input type="checkbox"/> UG3 <input type="checkbox"/> UG4 <input type="checkbox"/> UG5 <input type="checkbox"/> MSc
<b>SCQF Credit Level</b> Level 8 should normally be used for pre-honours courses. Level 10 should normally be used for optional UG3 courses (so UG4 students may also take them) and for courses aimed mainly at UG4 students. Level 11 should be used for courses aimed mainly at MSc students, whether or not UG4 students can also take them.	<input type="checkbox"/> 7 <input checked="" type="checkbox"/> 8 <input type="checkbox"/> 9 <input type="checkbox"/> 10 <input type="checkbox"/> 11
<b>SCQF Credit Points</b>	<input type="checkbox"/> 10 <input checked="" type="checkbox"/> 20 <input type="checkbox"/> 40 <input type="checkbox"/> 60 <input type="checkbox"/> 80 <input type="checkbox"/> Other:
<b>Delivery Location</b>	<input checked="" type="checkbox"/> Campus <input type="checkbox"/> On-line Distance Learning
<b>Course Type</b>	<input checked="" type="checkbox"/> Standard (default) <input type="checkbox"/> Dissertation <input type="checkbox"/> Online Distance Learning <input type="checkbox"/> Other (specify: Placement, Student Led Individually Created Course, Year Abroad)
<b>Marking Scheme</b> By default, courses use a numerical marking scheme. If you wish to use a grade-only marking scheme, your course proposal below should justify this.	<input checked="" type="checkbox"/> Standard (numerical) <input type="checkbox"/> Letter grade only

## Guidance for remaining sections:

**For an initial course proposal**, please complete the **cover page and Section 1 (Case for Support)**, which asks you to describe the need for this course and to provide an overview of the course design, including the learning outcomes. **Please discuss your plans as early as possible with the head of Curriculum Review to avoid unnecessary effort.**

Send the form with these sections completed to the BoS Academic Secretary and head of Curriculum Review (listed on the BoS page) to obtain their comments before filling out the remainder of the form.

**If a full proposal is invited**, please complete the remaining sections and send to **iss-bos@inf.ed.ac.uk**.

### **2. Student-facing course description and additional feedback and assessment information.**

*This section provides most of the information students see in the DRPS entry for this course, as well as related details for BoS consideration.*

### **3. Further information for BoS consideration: sample materials.**

**4. Additional Course Details required for DRPS.** *[Administrative information such as delivery timing and prerequisites.]*

**5. Placement in degree programme tables.** *[Required for all level 9-11 courses; used to determine where the course will be added to existing degree programme tables.]*

**6. Comments from colleagues.** *[All course proposal should be sent to relevant colleagues in the area as well as to the appropriate year organizer and BoS Academic Secretary for comment in good time before the BoS meeting. Use this section to indicate what feedback has been solicited and received.]*

### **Colour coding and item-by-item guidance:**

*Guidance is provided in italics for each item. Please also refer to the guidance for new course proposals at <http://www.inf.ed.ac.uk/student-services/committees/board-of-studies/course-proposal-guidelines>. Examples of previous course proposal submissions are available on the past meetings page <http://web.inf.ed.ac.uk/infweb/admin/committees/bos/meetings-directory> but note that the proposal form was updated in Jan 2019.*

<b>Sections in gold</b> are for student view and are required before a course can be entered into DRPS. You must complete these sections even if your course has already been approved based on other documentation.
<b>Sections in orange</b> are for School use but are still required for all courses (even those that have already been approved based on other documentation).
<b>Section in gray</b> are for consideration by the Board of Studies. They are normally required for all new course proposals but may be omitted in some circumstances (e.g., for invited course proposals) if you obtain permission in advance.

## 1. Case for support

This section is for consideration by the Board of Studies. The final two boxes (Learning Outcomes, Graduate Attributes) will also go into the student-facing course description.

### **Overall contribution to teaching portfolio and relation to existing curriculum**

*Please explain (a) what motivates the course proposal ( e.g. a previous course having become outdated/inappropriate, an emergent or maturing research area or new research activity in the School, offerings of our competitors) and (b) how it relates to existing courses and degree programmes (including any prerequisite courses). Every new course should make an important contribution to the delivery of our [Degree Programmes](#).*

This course proposal is a modified version of the Inf2-SEPP proposal that was already approved in December as part of the changes to pre-honours curriculum. Due to staff resignations, some of the planned resource for developing and delivering the original plan for Inf2-SEPP is not available. Following discussion with the Director of Teaching (Stuart Anderson), Deputy Director for Curriculum (Sharon Goldwater), and former Inf2C-SE course organizer (Paul Jackson), it was decided that a more feasible alternative is to transition to the previously approved version of Inf2-SEPP over two years.

Therefore, this proposal presents an updated version of the 20-credit Inf2-SEPP that is an intermediate step towards the original vision of Inf2-SEPP. It requires less drastic change from the current Inf2C-SE but still considers the original motivations for the change to Inf2-SEPP and experiments with some of the plans for the final course.

#### **The main similarities/differences to Inf2C-SE and the previously approved Inf2-SEPP are:**

- Coursework structure will be similar to Inf2C-SE (3 phased hand-ins of a project, working in pairs). This structure worked well in 2019/20 except students felt it was too intensive for a 10 credit course.
- Learning outcomes are the same as in the previously approved version, except that 'Work effectively as part of a team' has been removed. As in Inf2C-SE, most students will work in a team of 2, but we will still permit students to work solo. This learning outcome will be reintroduced in the following year and teams will be larger.
- Assessment weighting is 60% cw, 40% exam (rather than 100% cw as in the previously approved version). The exam will cover similar topics as in Inf2-SE, but be slightly scaled up and become a 2-hour exam (previously it was 1-hour, which was very tight and stressful for the students, while speed is not something that should be assessed as part of this course). The coursework will also be scaled up slightly and include some reflective components (though less than in the original proposal).
- There will be 30 lectures instead of the 19 used in Inf2C-SE. The extra lectures will mainly be used for guest lectures from industry and to facilitate more integration of professional issues.
- We will keep the tutorial pattern from Inf2C-SE, which supports the exam. Like Inf2C-SE, there will be drop in labs (rather than scheduled labs as in the original proposal).

### **Target audience and expected demand**

*Describe the type of student the course would appeal to in terms of background, level of ability, and interests, and the expected class size for the course based on anticipated demand. A good justification would include some evidence, e.g. by referring to projects in an area, class sizes in similar courses, employer demand for the skills taught in the course, etc*

Same as for the approved Inf2-SEPP (i.e., UG2 students – expecting around 320).

## Anticipated Resource Requirements

*Estimate how much lecturing, tutoring, exam preparation and marking effort will be needed in steady state, and any additional resources needed to set the course up initially. Provide estimates relative to class size where applicable and discuss how support staff will be recruited and supervised, if the class is likely to be very large. Please mention any scaling limits due to equipment or space. If equipment is required, say how it will be procured and maintained.]*

We propose the following:

- 3 lectures a week instead of 2 which is currently the number for Inf2C-SE; this will facilitate the integration of professional issues, as well as inviting guest lectures from industry.
- Like in Inf2C-SE, 4 tutorials in even weeks starting in week 4, to fit with the Ins2C-CS course which is holding them in odd weeks starting with week 3; Given the likely increase in number of students, 25 tutor posts would be required.
- Like in this year's Inf2C-SE, and considering student number increases, roughly 22 drop-in 1-hour lab sessions manned by experienced lab demonstrators giving students support for working on their coursework; These lab sessions would usually be scheduled in the evenings at 5-6 pm unless there is a clash with the lecture, and their dates and frequency will be scheduled to match the three coursework deadlines.
- Like in this year's Inf2C-SE, one optional bookable 30-minute meeting between a 2-student group and one of the lab demonstrators, as support for coursework 3; According to experience in Inf2C-SE in previous years, not all groups book such a meeting.
- Overall, given last year's staffing for Inf2C-SE and estimated student number increases, 17 lab demonstrators hired at 11 hours each would be required. The lab demonstrators would also be asked to offer support with Piazza enquiries.
- One or two TAs at minimum 120 hours of work (60 hours each) will be required for preparing the assignment instructions, marking criteria, tests or auto marker and offer support with Piazza enquiries.
- Based on this year's estimates and the proposed modifications to the assignment, the marking is estimated to require around 600 hours, split into 10 posts of 60 hours each
- Exam marking would require approximately 180 hours, split into 6 posts at 30 hours each

## Quotas, special arrangements or unusual characteristics

*Please specify if this course requires any special arrangements such as quotas or other registration arrangements; is a collaboration with another school or institution, or has other atypical characteristics that may affect finances or student registration. Further justification/information may be requested for such courses.*

No.

## Narrative description of the course aims and structure

*Please describe the main goals of the course and how the course design will allow students to achieve those goals. This section should be consistent with the student-facing information provided below, but should provide additional information to help colleagues at BoS understand the vision and structure of the course. This description may refer to the learning outcomes and graduate attributes (next two boxes) and should explain how activities such as tutorials, labs, or in-lecture activities will support them, and how the proposed assessments will assess them.*

*For courses that are important pre-requisites for other courses, this section may also provide content/syllabus information which is too detailed for the student-facing description, such as a lecture-by-lecture syllabus.*

We plan for assessment to be 60% coursework and 40% exam. This would allow for more hours to be spent by students on hands-on experience, address feedback about the current course requiring too much work, and move towards the plan for 100% coursework in the 2021-2022 iteration of Inf2-SEPP. Moreover, we propose for the exam for this coming year to be changed from a 1-hour to a 2-hour one. Working against the clock is not a skill that this course is intended to develop, and the exam in Inf2C-SE required a good number of tasks in the very short timeframe of 1 hour, leading to some students being unable to cover it fully and causing a lot of stress. We propose keeping the general current structure of the exam (question 1 compulsory, a choice between questions 2 and 3), while slightly increasing the number of subtasks and/or their difficulty.

We plan to keep to the current general structure of lecture topics for 2 of the lectures every week, while putting more emphasis on modern software engineering methodologies such as agile, as backed up by a new course

textbook ("Engineering Software Products" by Ian Sommerville) and updated reading lists. These 2 lectures would also include short introductions to the coursework and its expectations and group feedback on coursework, depending on the week. The third lecture is intended to be used either for guest lectures from industry representatives on topics related to that of other lectures or the coursework, or for introducing topics and case studies around Professional Issues, as backed up by the ACM Code of Ethics and the BCS Code of Conduct.

The coursework would involve the development of a small-scale software system by groups of 2 students (although one could voluntarily take on the full work as well) and would be split into 3 parts, following the model from Inf2C-SE: requirements, design, construction/testing/documentation. It would have 3 deadlines and involve iteration, following an iterative waterfall model. The first deadline would be formative for coursework 1, with provisional marks and feedback provided. The second would require students to (re)submit coursework 1 with feedback considered, but also to submit coursework 2. It would be formative for coursework 2, with provisional marks and feedback provided. However, for coursework 1 it would be summative. Finally, the third deadline would require students to (re)submit coursework 2 and submit coursework 3, and this time it would be summative for both. While all 3 pieces of coursework would be supported by drop-in labs, the times and frequency of which would be given by approaching coursework deadlines, only courseworks 1 and 2 would thus be covered by written formative feedback. To enhance formative feedback for coursework 3, students would have the opportunity to book one 30-minute private meeting with a lab demonstrator for personalised feedback. Additionally, support would be provided on Piazza, via office hours and email.

As new to this course, the coursework would have as an important component the students' reflection on the approaches and tools used and their evaluation, as well as questions around professional issues (with the last being required on a separate topic of each member of the team as part of an individual report). Following the theme of iterative waterfall, they would need to maintain consistency in the phases of development (i.e. the different pieces of coursework) and reflect on how they have done this and the difficulties encountered. Moreover, students would be required to self-assess themselves given a high-level marking scheme, and be reflective in this process. When re-submitting a piece of coursework after formative feedback, they would need to justify how they have addressed the feedback. We believe that all of this contributes to developing their skills and graduate attributes, while also clarifying expectations. While we attempted reflective self-assessment in Inf2C-SE this year, we found that students require more guidance for being reflective, and so we plan to incorporate examples in lectures.

The coursework would also have as an important component peer assessment, which would be introduced for coursework 3 by requiring students to peer review each others' code and tests for a small module. While this component was used as optional for students this year in Inf2C-SE, we intend to make it compulsory and more formalised, by providing clear guidance and examples in lectures as to what constitutes a high quality peer review.

To prepare students for the exam, tutorials would include exercises from past exam papers. Moreover, as already started this year in Inf2C-SE, we intend to use TopHat for quizzes in lectures (more than this year). Finally, support would be provided on Piazza, via office hours and email.

The assessment model proposed maps with the Learning Outcomes from the next section as follows:

1. For the coursework, students will be required to reflect on their experience with applying modern techniques in the design and development of large-scale software systems - introduced in lectures and covered in detail in associated reading. This will involve their explanation of how these techniques work. Whilst they will be usually required to use certain such techniques, we plan to leave students a small amount of freedom in choosing their own for small parts of the assignment, so that they can also

introduce them as part of their reflection. These would constitute part of their assignment mark. Moreover, students will be required to explain software engineering techniques as part of tutorials and their 'bookwork'-style questions in the exam.

2. All the parts of their coursework involve the application and evaluation of the modern techniques as part of a small-scale realistic case study. As part of their reflection, students will need to explain how the chosen techniques have worked for them (i.e. evaluation). How well they have applied the techniques will be apparent both from the produced software solution, as well as from their reflective accounts. Their evaluation will be assessed from their reflective accounts.
3. Students will be asked to individually reflect on professional issues (different for each team member) as part of their coursework. The material on professional issues presented in the lecture, as well as associated reading, will offer them the background knowledge to be able to tackle this task. The quality of their reflection will constitute part of their marks on this course.
4. Reading technical documents will be an important part of the overall reading for this course, in order to understand, apply and thus be able to reflect on the techniques presented, as required by the coursework. Moreover, students will be required to write documentation presenting their solution, which will constitute part of their mark.

### Summary of Intended Learning Outcomes (MAXIMUM OF 5)

List the learning outcomes of the course. These must be assessable (i.e., observable), so must specify what the student should be able to do concretely, not simply what they should "understand". Use concrete verbs that indicate (a) what type of assessment would be appropriate, and (b) what level of knowledge/thinking is expected (from recall to analysis to novel creation). **Example verbs:** define, explain, implement, compare, justify. Assessments (described later) should be tied to the learning outcomes.

Outcomes should typically focus more on the types of thinking/skills developed than on the detailed course content, and the level of thinking should be appropriate to the level of the course: outcomes for a Level 11 course should include more higher-level thinking skills than for a Level 8 course. Further guidance on writing learning outcomes can be found at

<https://www.ncl.ac.uk/ltads/assets/documents/res-writinglearningoutcomes.pdf>

On completion of this course, the student will be able to

- 1) Explain the modern techniques used in the design and development of large-scale software systems
- 2) Apply, evaluate and reflect on these techniques in a small-scale, but realistic scenario
- 3) Analyse the professional and ethical implications of software engineering decisions and propose solutions
- 4) Comfortably read and write technical documentation.

### Graduate Attributes, Personal & Professional Skills

List the personal attributes and generic transferrable skills this course will help develop. Examples include

**Cognitive skills:** problem-solving, critical/analytical thinking, handling ambiguity

**Responsibility, autonomy, effectiveness:** independent learning, self-awareness and reflection, creativity, decision-making, leadership, organization and time management, flexibility and change management, ethical/social/professional awareness and responsibility, entrepreneurship

**Communication:** interpersonal/teamwork skills, verbal and/or written communication, cross-cultural or cross-disciplinary communication

This course develops a wide range of graduate attributes and skills across several areas:

- Cognitive skills: problem-solving, critical/analytical thinking, handling ambiguity.

- Responsibility, autonomy, effectiveness: independent learning, self-awareness and reflection, creativity, decision-making, organization and time management, flexibility and change management, ethical/social/professional awareness and responsibility.
- Communication: interpersonal/teamwork skills, verbal and written communication.

## 1. Student-facing course description and additional feedback and assessment information

Except where noted, all fields are required and will go into the DRPS entry for the course (for use by students). **Important:** any text in DRPS is effectively a contract with students, so should not include details that are likely to change from year to year.

<p><b>Summary Description</b>  <i>Provide a brief official description of the course, around 100 words. This should be worded in a student-friendly way, it is the part of the descriptor a student is most likely to read. If this course replaces another course, please say so in this summary.</i></p>	<p>Software Engineering and Professional Practice teaches the practice of small team software development in modern society, equipping students to participate in a startup, modern tech company or a software-dependent research team.</p> <p>Students will gain experience developing a software system from scratch, using some of the key tools of the trade: analysing requirements, designing and implementing new features, testing, version control.</p> <p>Professional aspects of Software Engineering — its legal, ethical and social environment, including issues of privacy, security, equality, democracy and intellectual property — will be approached through guest lectures and some practical work.</p>
<p><b>Keywords</b>  <i>Give a list of searchable keywords.</i></p>	<p>software engineering, professional practice, ethics</p>
<p><b>Course Description</b>  <i>A more detailed student-facing description of the course, which should normally include (a) a more in-depth academic description of the learning aims, nature and context of the course, (b) a rough outline of the content or syllabus, often as bullet points, and (c) a description of how the course will be taught, how students are expected to engage with their learning and how they will be expected to evidence and demonstrate their achievement of the intended learning outcomes.]</i></p>	<p>As students enter this course they team up in groups of two (volunteers wanting to go ‘solo’ also accepted, but for full assignment load) to develop a small-scale software system from scratch using an iterative waterfall process. Over the course of the semester, they consider an incomplete specification to derive and analyse requirements, design their solution from a static and dynamic perspective using UML diagrams, construct and test their solution in Java. There is room for interpretation, creativity, and some of the requirements change along the way. Moreover, there are professional issues surrounding the problem at hand.</p> <p>Included in the experience will be use of industry standard tools for software development (integrated development environments, version control, issue tracking), and key elements of modern development practice, such as code review, peer review, and pair programming.</p> <p>As students engage in this practical work, the course will contextualise it against the broader themes, both of large-scale Software Engineering and its academic literature, and of today's urgent professional issues: the legal, ethical and social context in which software and its authors exist. Guest lecturers will speak on technical topics, but also on topics such as privacy, security, equality, democracy and intellectual property — some of which will have a direct impact on students' practical work.</p> <p>The assignment is assessed through a mixture of group and individual work, mainly on software development but also on</p>

	<p>written reflective practice. The assignment will consist of 3 parts, with the first two being iterated over after receiving formative feedback, and the requirement to maintain the whole solution consistent.</p> <p>Moreover, there is a written exam at the end of the semester.</p>
<p><b>Assessment Weightings:</b>  <i>These should correspond approximately to the proportion of learning outcomes that each component assesses. More than 30% coursework requires specific justification.</i>  <i>The expectation for a 10pt course is 20% coursework with the equivalent of one 15-20hr assessed assignment (but possibly split into smaller pieces). See 'components of assessment' below.</i></p>	<p>Written Exam __40__%  Practical Exam __0__% (for courses with programming exams)  Coursework __60__%</p>
<p><b>Further Assessment Information</b>  <i>Provide any further information that should go on DRPS for students. E.g., if the assessment includes required group work or if students must pass some individual component of assessment as well as the course overall.</i></p>	<p>The assignment is based on the use of Software Engineering techniques (e.g. UML diagrams) and tools (e.g. version control), programming, reflective writing. Work is assessed both individually and collectively as part of a 2-person team. Students deciding to work 'solo' will take on the whole load of the assignment.</p> <p>The written examination lasts 2 hours and consists of a combination of application of knowledge, problem solving and bookwork.</p> <p>Students must achieve at least 40% in the exam, as well as 40% overall, to pass the course.</p>
<p><b>Components of assessment and time spent on assignments (for BoS only)</b>  <i>If not already included in the course narrative description, please describe the type of assessments (oral presentation, report, programming, etc) and <b>how each component of assessment will assess the intended learning outcomes</b>. Where coursework involves group work, it is important to remember that every student has to be assessed individually for their contribution to any jointly produced piece of work.</i></p> <p><i>Also estimate <b>how many hours</b> students will spend on assignments. Please see the <a href="#">School policy on Workload and Assessment</a>, which states that students should not be expected to spend more than 6-7 hrs/wk per 10 credits, including contact hours.</i></p> <p><i>Note that it is often desirable to include formative assignments which are not formally assessed but submitted for feedback, often in combination with peer assessment.</i></p>	<p>We propose 3 pieces of coursework and 3 deadlines. The first deadline would be completely formative, with students submitting coursework 1 (requirements) for provisional marks and feedback. The coursework would require deriving and analysing requirements using use case diagrams. The second would allow students to (re-)submit coursework 1, but also submit coursework 2 (design). This coursework would involve designing their solution following the analysed requirements, by using UML notation. At this second deadline, coursework 1 would be assessed summatively (provisional marks considered as final if no re-submission), while coursework 2 would be assessed formatively for provisional marks and feedback. Part of the marks for coursework 2 would be awarded based on the fit with coursework 1 (i.e. consistency design-requirements). Finally, the third deadline would allow students to (re-)submit coursework 2, but also submit coursework 3 (construction, testing in Java, documentation) for summative assessment. Part of the marks for coursework 3 would be awarded based on the fit with coursework 2 (i.e. consistency construction-design).</p> <p>All courseworks would include a group reflective component on techniques used, group work, self-assessment, how formative feedback was addressed, as well as an individual component on professional issues involved in that stage (different for each member of the group). Moreover, coursework 3 would include</p>

	<p>a peer review of another group's code and tests for a different module than that implemented by the group (half of the groups would be implementing one module, half another so that they can be matched up).</p> <p>Students will be expected to spend on average 4 hours a week working on the assignments, not considering preparation time and including work during the labs.</p>
<p><b>Feedback Information</b></p> <p><i>Provide a high-level description of how and what type of feedback will be provided to students, for inclusion in DRPS.</i></p>	<p>Students will be provided with formative feedback and provisional marks on their first submission of the first two assignments. They will have the opportunity to consider the feedback and re-submit together with the next assignment for summative feedback and final marks. A lack of re-submission will result in the provisional marks becoming final. The only deadline for the third assignment is for summative feedback and final marks.</p> <p>Formative feedback will also be provided during drop-in lab sessions scheduled irregularly and more frequently as assignment deadlines are approaching, and during fortnightly tutorials. For the third assignment, extra support in terms of a bookable 30-minute private meeting with a demonstrator will be made available. Moreover, students will gather feedback from another group on part of their code and tests as part of a peer review activity. Finally, ongoing support will be provided via the Piazza online forum, office hours and email.</p>
<p><b>Additional Feedback Information (for BoS use only)</b></p> <p><i>If not already included in the course narrative, provide further details on planned feedback arrangements. This includes how course feedback is solicited from the class and responded to, as well as what feedback students will get (either on work that contributes to their final mark, or not).</i></p> <p><i>The University is committed to a <a href="#">baseline of principles</a> regarding feedback that we have to implement at every level, and the School encourages submission of at least one piece of written work for formative feedback.</i></p> <p><i>In general, formative feedback:</i></p> <ul style="list-style-type: none"> <li>• <i>Should say how students can improve.</i></li> <li>• <i>Need not be on individual work (e.g., consider a lecture or document summarizing common issues.)</i></li> <li>• <i>Can include oral feedback during labs/tutorials</i></li> <li>• <i>Can include feedback from peers</i></li> <li>• <i>Clickers/TopHat/equivalents can provide in-class feedback for both students and lecturer.</i></li> <li>• <i>Is returned in time for other forms of assessment to which it relates, to allow feedforward.</i></li> </ul>	<p>Course feedback will be solicited both from students and the course team. From students, we aim to conduct mid-term feedback using a combination of a short TopHat quiz to be used in a lecture and a separate online questionnaire to be used outside the lecture. Moreover, student progress in coursework and as reported by the lab demonstrators will be a good source of information. End of term feedback from students will be collected through the official Course Enhancement Questionnaire. We also aim to gather mid-term and end of term feedback from the tutors and lab demonstrators through online chats and a final face-to-face discussion. Future course components and sources of support will be adapted in line with the feedback results.</p> <p>Students will be provided with formative feedback for their first submission of courseworks 1 and 2, both in writing and as in-lecture feedback. This formative feedback will touch on the quality of the produced solution, its match with the previous stage (for coursework 2), the quality of the group reflective discussion on the techniques used and their evaluation, team work, self-assessment, the quality of the individual reflective discussion on professional issues. Constructive advice on how to improve and hints to reading will be provided. We will aim to deliver formative feedback to the students within at most 10 days of the deadline (as opposed to the typical 14 days for</p>

	<p>summative assessment) so that they have time to consider it for their resubmission of the coursework at the next deadline. Students will be provided with summative feedback for coursework 1 after deadline 2, and for courseworks 2 and 3 after deadline 3, both in writing and as in-lecture feedback (only for coursework 2). This feedback will be of a similar standard to the formative feedback. Students will be given both their group feedback on their work and reflection, as well as feedback on their individual reflection on professional issues.</p>																				
<p><b>Breakdown of Learning and Teaching Activities</b>  <i>State how many hours students spend on each part of the course. The total should be 10 x course credits, but please also see the <a href="#">School policy on Workload and Assessment</a>, which states that students should not be expected to spend more than 6-7 hrs/wk per 10 credits, including contact hours.</i></p> <p><i>Assume 10 weeks of lectures slots and 10 weeks of tutorials, but these need not all be used. As a guideline, a 10-pt course typically has 18-20 lecture hours, but should have only around 15 lectures of examinable material; the rest should be used for guest lectures, revision sessions, introductions to assignments, etc.</i></p>	<p><b>Contact hours</b></p> <table border="1"> <thead> <tr> <th>Hours</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>30</td> <td>Lecture Hours</td> </tr> <tr> <td>4</td> <td>Seminar/Tutorial Hours</td> </tr> <tr> <td>0</td> <td>Dissertation Project Supervision Hours</td> </tr> <tr> <td>22</td> <td>Supervised practical/Workshop/Studio hours</td> </tr> <tr> <td>3 (2 of them as part of lecture hours)</td> <td>Feedback/Feedforward hours</td> </tr> <tr> <td>2</td> <td>Summative assessment hours</td> </tr> <tr> <td>1 (as part of of lecture hours)</td> <td>Revision Session Hours</td> </tr> </tbody> </table> <p><b>Non-contact hours</b></p> <table border="1"> <thead> <tr> <th>Hours</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>141</td> <td>Directed Learning &amp; Independent Learning hours</td> </tr> </tbody> </table> <p><b>Total hours: 200</b></p>	Hours	Type	30	Lecture Hours	4	Seminar/Tutorial Hours	0	Dissertation Project Supervision Hours	22	Supervised practical/Workshop/Studio hours	3 (2 of them as part of lecture hours)	Feedback/Feedforward hours	2	Summative assessment hours	1 (as part of of lecture hours)	Revision Session Hours	Hours	Type	141	Directed Learning & Independent Learning hours
Hours	Type																				
30	Lecture Hours																				
4	Seminar/Tutorial Hours																				
0	Dissertation Project Supervision Hours																				
22	Supervised practical/Workshop/Studio hours																				
3 (2 of them as part of lecture hours)	Feedback/Feedforward hours																				
2	Summative assessment hours																				
1 (as part of of lecture hours)	Revision Session Hours																				
Hours	Type																				
141	Directed Learning & Independent Learning hours																				
<p><b>Reading List/Learning Resources</b>  <i>You are encouraged to create resource lists using <a href="#">LEGANTO</a></i></p>	<p>Sommerville “Engineering Software Products”  ACM code of ethics: <a href="https://www.acm.org/code-of-ethics">https://www.acm.org/code-of-ethics</a>  BCS code of conduct: <a href="https://www.bcs.org/membership/become-a-member/bcs-code-of-conduct/">https://www.bcs.org/membership/become-a-member/bcs-code-of-conduct/</a></p>																				

## Further information for BoS consideration: sample materials

A full proposal for a new course must include examples of exercises and assessment. Please provide these below, along with publicity information if the course is to be advertised outwith the School.

<p><b>Course information and publicity</b>  <i>The course web page (typically the Learn landing page) will be linked from the Sortable Course List, and information such as timetables and assignment deadlines must be made available prior to the start of the academic year. Please specify here if any additional info/publicity is needed for your course: typically only if it is aimed largely at non-Sol students.</i></p>	<p>No additional information/publicity is needed for this course.</p>
<p><b>Sample tutorial/lab sheet questions</b>  <i>Provide a list of tutorial questions and answers and/or samples of lab sheets. These need not be fully fleshed out but should indicate what sort of exercises will be provided to help students learn the material.</i></p>	<p>We intend to reuse part of the Inf2C-SE tutorials and their sample solutions, which can be found on the Learn page for Inf2C-SE under “Tutorials”.</p> <p>There will be no lab sheet questions, as the lab demonstrators will offer support for the coursework to the students.</p>
<p><b>Sample assessment materials</b>  <i>If the course is primarily assessed by <b>exam</b>, provide a sample exam question with model answers. Any non-standard exam format must be justified. The online list of past exam papers gives an idea of typical and alternative exam formats:  <a href="http://www.inf.ed.ac.uk/teaching/exam_papers/">http://www.inf.ed.ac.uk/teaching/exam_papers/</a>.</i></p> <p><i>If the course is largely or primarily assessed by <b>coursework</b>, provide a sketch of a possible assignment with an estimate of effort against each sub-task and a description of marking criteria.</i></p>	<p>An example system that students could be required to develop would be that for a booking system for festival events. In coursework 1, the students could be provided with a high-level description of the system, with some ambiguities intentionally left in and some underdefined parts. They could be required to analyse the description to extract the list of stakeholders, functional requirements in terms of use cases and a list of non-functional requirements, as well as discuss and attempt to find solutions to potential ambiguities. Moreover, they could be asked to evaluate and reflect on the advantages and disadvantages of use cases and use case diagrams, ways of capturing non-functional requirements and related difficulties, how ambiguities could be elucidated together with the stakeholders (i.e. appropriate requirements elicitation techniques). In individual reports, they could be asked to discuss either about issues related to security and privacy (e.g. in relation to the storage of personal data or card details when creating an account), or not acting in the public interest (e.g. if the system allows overbooking).</p> <p>In coursework 2, the students could be offered some clarification to the system description, together with a requirement to update the relevant use cases. Moreover, they could be asked to design the core of the system by using UML class, sequence and communication diagrams. They could be advised to use a technique such as CRC cards to simulate the design working in practice, evaluate and reflect on its use. Similarly, they could be asked to evaluate, compare and reflect on the differences between the different diagrams used, or on how their solutions produce good design. Individual reports could require a discussion about the importance of working at high standards and keeping to the original brief.</p>

	<p>In coursework 3, there could be new changes brought to the system description, together with the requirement to update the design accordingly. Furthermore, students could be asked to implement their solution in Java, develop unit and integration tests, develop documentation, use a version control system. We could suggest that they also use a tools for reporting bugs and checking test coverage. They could be required to evaluate and reflect on the tools used (required and chosen), their advantages and disadvantages for them. For the peer review, we could provide a structured pro-forma, and then ask them to reflect on their experience. For the individual reports, the students could be asked to discuss issues related to competence, robustness and risk with regards to testing and debugging the given system, or related to working with others.</p>
<p><b>Any other relevant materials</b>  <i>Include anything else that is relevant, possibly in the form of links. If you do not want to specify a set of concrete readings for the official course descriptor, please list examples here.</i></p>	

**1. Additional Course Details for DRPS**

Except where otherwise noted, these fields are required for entry into EUCLID and will be visible to students in the DRPS entry.

<p><b>Planned Academic Year of Delivery</b> <i>(The first year you anticipate the course running, e.g. AY 2019-20)</i></p>	<p>2020-2021</p>
<p><b>Course Organiser</b> <i>(By default, the course proposer)</i></p>	<p>Cristina Adriana Alexandru</p>
<p><b>Intended Delivery Period</b></p>	<p><input type="checkbox"/> Semester 1  <input checked="" type="checkbox"/> Semester 2  <input type="checkbox"/> Full Year  <input type="checkbox"/> Summer  <input type="checkbox"/> Other (please specify):</p>
<p><b>Timetable considerations/conflicts</b> <i>For School use. Please specify any constraints to be considered (e.g. overlap of popular combinations, other specialism courses, external courses etc). Include whether the semester delivery is constrained or could be flexible.</i></p>	<p>Should not conflict with other UG2 or UG1 courses.</p>
<p><b>Is this course available to visiting students?</b></p>	<p><input type="checkbox"/> Yes (default)  <input checked="" type="checkbox"/> No  <b>If no, please provide a justification here:</b></p>
<p><b>Required pre-requisite courses</b> <i>Use sparingly: these are enforced in PATH and can only be waived by approval from the School's Curriculum Approval Officer. Note that cross-year required pre-requisites may prevent MSc students from registering; consider using recommended pre-requisites or "other requirements" instead.</i></p>	<p><input type="checkbox"/> No  <input checked="" type="checkbox"/> Yes (please specify full course name(s) and code(s)):          Inf1- Introduction to Computation (INFR08025)          Inf11-Object Oriented Programming (INFR08029)</p>
<p><b>Recommended pre-requisite courses</b></p>	<p><input checked="" type="checkbox"/> No  <input type="checkbox"/> Yes (please specify full course name(s) and code(s)):</p>
<p><b>Required co-requisite courses</b> <i>Specify any courses that must be taken in parallel with the existing course. Note that this leads to a timetabling constraint that should be mentioned elsewhere in the proposal.</i></p>	<p><input checked="" type="checkbox"/> No  <input type="checkbox"/> Yes (please specify full course name(s) and code(s)):</p>

<p><b>Prohibited Combinations</b>  <i>Specify any courses that may not be taken in combination with the proposed course].</i></p>	<p><input type="checkbox"/> No  <input checked="" type="checkbox"/> Yes (please specify full course name(s) and code(s)):  <u>Informatics 2C-Introduction to Software Engineering (INFR08019)</u></p>
<p><b>Other Requirements/Additional Information</b>  <i>This information is often used by MSc students and students from other Schools to see if they have appropriate background without having done our School's courses. So please avoid course titles, instead list specific knowledge and skills (such as mathematical concepts, programming ability or specific languages, etc).</i></p> <p><i>Also list any other constraints on registration, for example: "Only available to 4th Year Informatics students including those on joint degrees." or "This course is open to all Informatics students including those on joint degrees, and to students in the School of Mathematics. Other external students whose DPT does not list this course should seek permission from the course organiser."</i></p>	<p><input type="checkbox"/> No  <input checked="" type="checkbox"/> Yes (please specify):</p> <p>Only open to 2nd year Informatics students, including those on joint degrees.</p> <p>Prerequisite knowledge of object oriented programming required.</p>
<p><b>Visiting Student Pre-requisites</b></p>	<p><input type="checkbox"/> Same as "other requirements"  <input type="checkbox"/> Different than "other requirements" (please specify):</p>

## 2. Placement in degree programme tables: for level 9-11 courses only

This section is for consideration by the Board of Studies and will be used later by ITO to determine where the course will be added to existing degree programme tables.

<p><b>Is this course restricted to students on a specific degree?</b>  <i>E.g., some courses are only available to students on a specific CDT or MSc.</i></p>	<p><input type="checkbox"/> No  <input type="checkbox"/> Yes (please specify and provide justification):</p>
<p><b>Is this course compulsory for students on any degree(s)?</b></p>	<p><input type="checkbox"/> No  <input type="checkbox"/> Yes (please specify and provide justification):</p>
<p><b>Any issues for part-time students?</b>  <i>Normally, part-time students have access to the same courses as full-time students on the equivalent degree. If you anticipate any problems with this, please specify here.</i></p>	

### For optional courses:

If this course is available but non-compulsory for students on various degrees (most courses), please fill in this section. The choices here determine where the course appears in degree programme tables (DPTs) and the 2-3 character tags are displayed in the Informatics sortable course list.

<p><b>Should this course be tagged as 'ML' (machine learning foundations and methods)?</b>  <i>Courses with the ML tag are typically very high-demand and most degrees limit the number of ML credits. If your course might appeal to a similar audience but draw off students from these large courses, please select 'no' and choose one of the tags below.</i></p>	<p><input type="checkbox"/> No  <input type="checkbox"/> Yes</p>
<p><b>If you chose 'no', please choose at least one of the following tags...</b>  <i>Ideally, select exactly one, unless there is a good argument for more than one. These three are used in various combinations for many of our degrees.</i></p>	<p><input type="checkbox"/> <b>FSS</b> (CS foundations, systems, and software)  <input type="checkbox"/> <b>AIA</b> (artificial intelligence applications and paradigms)  <input type="checkbox"/> <b>COG</b> (cognitive science: including HCI and NLP courses, but not most other AI courses. Please restrict to courses most relevant to natural cognition.)</p>
<p><b>...and also tick if any of the following tags or categories apply.</b>  <i>Do not tick any of these if you selected 'ML' already.</i></p>	<p><input type="checkbox"/> <b>NS</b> (natural systems: e.g., computation by or about biological or social systems. Many COG courses are also NS. This tag is mainly relevant for MSc in Informatics.)  <input type="checkbox"/> <b>SE</b> (software engineering: including courses that are highly relevant to SE degrees. All SE courses should also be FSS. This tag is mainly relevant for UG SE degrees.)  <input type="checkbox"/> <b>Databases and data management systems</b> (used for Data Science MSc and MSc(R))</p>

	<ul style="list-style-type: none"><li>__ Unstructured data and applications (used for Data Science MSc and MSc(R))</li><li>__ Level 11 Security courses (used for Security MSc)</li></ul>
<b>If you are not sure which tags are most appropriate or have other questions about this section, please note any comments/issues here.</b>	

### 3. Comments from colleagues

All course proposal should be sent to relevant colleagues in the area as well as to the appropriate year organizer and BoS Academic Secretary for comment in good time before the BoS meeting. Please indicate here what feedback has been solicited and received.

<p><b>Additional Comments</b>  <i>Summarise any comments received from relevant individuals prior to proposing the course. If you have not discussed this proposal with others please note this.</i></p>	<p>This proposal was informed by discussions with the Informatics Director of Teaching and current lecturer of the 3<sup>rd</sup> year Professional Issues course (Stuart Anderson), the Deputee Director of Teaching (Sharon Goldwater) and the former course organiser of the Inf2C-SE course (Paul Jackson).</p> <p>Moreover, we discussed plans for the full roll out of the course that this increment is working towards for 2021-2022 with:</p> <ul style="list-style-type: none"> <li>● Lecturers who have taught Inf2C-SE: Paul Jackson, Perdita Stevens, Nigel Goddard, Ajitha Rajan</li> <li>● Lecturers who are teaching related courses:             <ul style="list-style-type: none"> <li>○ Inf2B: Paul Anderson, Volker Seeker</li> <li>○ SDM: Perdita Stevens</li> <li>○ ST: Ajitha Rajan</li> <li>○ Professional Issues: Stuart Anderson</li> <li>○ SDP: Barbara Webb</li> </ul> </li> <li>● Judy Robertson, who is specialised in Pedagogy</li> <li>● The Director of Teaching (Stuart Anderson) and Deputee Directors of Teaching (Sharon Goldwater, Paul Patras)</li> <li>● Administrative (Gillian Bell), learning technology (Alex Burford) and library staff (Angela Nicholson) members</li> <li>● Current tutors and demonstrators on Inf2C-SE, most of whom have also been students on a previous iteration of Inf2C-SE</li> </ul> <p>We have also considered past and current student feedback on Inf2C-SE from last year’s Course Enhancement Questionnaire and this year’s mid term feedback collected through TopHat and a Jisc survey</p> <p>Considering initial input from Stuart Anderson, Sharon Goldwater, Paul Jackson, Volker Seeker, Paul Anderson, Perdita Stevens and Judy Robertson, we first decided to organise an ELDeR workshop, to have the opportunity to brainstorm ideas with colleagues. This workshop took place on the 17th and 18th of September. Its participants were: Paul Jackson, Paul Anderson, Paul Patras, Gillian Bell, Alex Burford and Angela Nicholson. Vidminas Mikucionis (who is a tutor on the current iteration of Inf2C-SE and a former student on this course) joined us as a ‘critical friend’ at the end of the workshop to provide feedback on our plan. The decisions reached as part of ELDeR were materialised in the first draft of this proposal.</p> <p>Going further, we organised discussions with more lecturers as recommended at the first BoS: Ajitha Rajan, Nigel Goddard, Barbara Webb. Moreover, we have considered past and current</p>
--	--

	<p>student feedback on Inf2C-SE, and held a one-hour feedback discussion with the current tutors and demonstrators from Inf2C-SE, most of whom have also been students on a previous iteration of Inf2C-SE.</p>
<p><b>Year Organiser Comments</b>  <i>Year Organisers are responsible for maintaining the official Year Guides for every year of study, which, among other things, provide guidance on available course choices and specialist areas. The Year Organisers of all years for which the course will be offered should be consulted on the appropriateness and relevance on the course. Issues to consider here include balance of course offerings across semesters, subject areas, and credit levels, timetabling implications, fit into the administrative structures used in delivering that year.]</i></p>	
<p><b>BoS Academic Secretary Comments</b>  <i>Proposals must be checked by the Secretary of the Board of Studies prior to discussion at the actual Board meeting. This is a placeholder for their comments, mainly on the formal quality of the content provided above.</i></p>	