,

COMP1004
Computing Practice

40 CREDIT MODULE

ASSESSMENT: 80% Coursework
20% Practice

MODULE LEADER: Dr Vasilios Kelefouras
CO-TEACHERS Dr Lauren Ansell
Mr Laurence Bowes
Mr Maxwell Gisborne
Mr Arun Darsan
Mr Sebastian Outram

MARKING PANEL: to be confirmed

MODULE AIMS

- To familiarise students with the software development lifecycle
- To develop techniques to elicit requirements from abstract problems
- To expose students to both technical and non-technical skills required to complete a software engineering project

ASSESSED LEARNING OUTCOMES (ALO):

1. Articulate the phases of the software development lifecycle.
2. Define suitable functional and non-functional requirements that when implemented in software will solve a given problem.
3. Identify and describe operational considerations of software, including the physical systems upon which they reside, cyber security and networking.
4. Implement a system that fulfils a design specification to solve a given problem.

Overview

This document contains all the necessary information pertaining to the assessment of *COMP1004 Computing Practice* module. The module is assessed using two elements of assessment. A coursework element worth 80% of the module and a practice element worth 20% of the module. The Practice element has two submission points, one at the end of Semester one and another at the end of Semester two. It may be easier to think in terms of the module marks available; see the final column of Table 1 below.

Element	Deliverable	Weighting	Marks
Practice (20%)	Interim Video	20% (30% of it)	6 module marks
	Final Presentation	20% (70% of it)	14 module marks
Coursework (80%)	Final Portfolio	80% (100% of it)	80 module marks

Table 1: Assessment Weightings/Marks

The sections that follow will detail the assessment tasks that are to be undertaken. The submission and expected feedback dates are presented in Table 2. All documents are to be submitted electronically via the module DLE pages before the stated deadlines and all presentations will be held in the School Computing labs (Smeaton 101, 107, 108, 109, 200 or 201).

	Submission Deadline	Feedback
Interim Video (see Section 4)	Monday 15th January 2024 at 15.00	within 20 working days
Final Portfolio (see Section 6)	Thursday 18th of April 2024 at 15.00	within 20 working days
Final Presentation (see Section 5)	Tuesday 7th of May 2024	within 20 working days

Table 2: Assessment Deadlines

This project module is an all year, 40 credit, module that provides you with an opportunity to learn how to apply problem-solving techniques to understand how to create technological solutions to solve a problem. This gives you the chance to really understand the core principles of Computer Science, the software development lifecycle and how that applies to the degree you are studying.

1 Project Choices

Students will create a single page web application (known as SPA) of their choice. Check out the description of a SPA here.

It is recommended that you work on a project suitable for your degree where you will focus on the early stages of the software development lifecycle whilst exploring the operational and technical trade-offs that are inherent in that cycle. Indicative scenarios can be found in a separate document (see '**comp1004_project_*scenarios.pdf***' file under the assessment tab on the module's DLE page). Each scenario has an indicated base functionality, and an initial task is to research the background to the project and to develop that functionality further. By the end of the module, you will have analysed, designed and developed a prototype of moderate complexity, the culmination is the creation of a simple, yet complete software solution.

You should **only** use technologies that you learn on this module!

2 Project Steps

All projects need to carry out the following steps:

- **Research** the background for your project. Identify why your project might be of importance (not just that you are going to be assessed on it!).
- Create a **project vision** that will encapsulate the essence of the project. Obtain feedback on this project vision during the initial meetings with the module staff.
- Begin **planning** how you will create the project.
    - You should plan your time in 2-week sprints.
    - Two of these sprints should happen prior to Christmas with the rest taking place in Semester 2.
- Research what **issues** and **challenges** you are likely to face and note these down.
- **Prototype** the look and feel for your web application ensuring you have created the home page in HTML in the early weeks of the module.
- Create a **design document** using the template provided under the Assessment tab (see '*Template.pdf'* file).
- Make sure you have created a **GitHub repository.** You are expected to use GitHub early on and not just upload all your code at the end of the project. Ensure you edit the readme.md file in GitHub so that your project title, and project vision are clearly noted.
- **Attend** each of the stand-up meetings in the sprints. At each meeting with the staff you must show what you have worked on so far and ensure you use the contact hours to good effect. This is where you will obtain feedback on your work.
- Create the following artifacts:
    - UML Diagrams containing Use Cases, Classes, Interaction diagrams.
    - Product Backlog
    - Sprint plans
- Implement your tasks identified in your product backlog as per your sprint plans.
- Ensure you obtain feedback during the stand-up meetings.

3 Sprints

Short scrum group meetings will be held with a member of staff from the module. These will take place every two weeks during term time (starting in November). Your scrum meetings will be held in a group with other students on the module during timetabled COMP1004 sessions.

The scrum meetings require each student to describe their work to date, highlighting progress from the previous scrum meeting. During these meetings, students should present plans for the following fortnight describing any barriers they face. As with all scrum meetings, these will be short, highly focused sessions. Prior to the scrum meeting you should review your sprint backlog ensuring that completed items are removed and new items added.

4 Interim Video

This deliverable accounts for 30**%** of your **practice** element and it is worth 6 module marks.

4.1 The aim of this video is to present and demonstrate what you have done so far and hopefully demonstrate a prototype. It is recommended that students review each other's videos to obtain constructive feedback prior to submission. Your video should last a maximum of 4 minutes. You will get no marks for anything you present after 4:00.

You should present the following topics, in this exact, order to maximise your marks.

- Project Vision and background.
- Your project plan and sprints carried out to date.
- UML models to date.
- An initial prototype running (hopefully).
- A succinct description of issues/challenges faced.

4.2 What/How to submit

First, you should make PowerPoint presentation slides by using Microsoft PowerPoint Slide Presentation Software (like the slides you see in the Lectures). Then, you need to use another software tool (e.g., obsproject (https://obsproject.com/) software) to record your voice while going through the slides. Your video must be made available via YouTube and should meet the following quality criteria:

- The video length should **not** be more than 4 minutes long. Marking will stop at exactly 4:00. You must not speed up your video to gain more time. Note that your video will be paused for thorough reading, so always show each slide for at least 10 seconds.
- The resolution of the video should be 720p to 1080p.
- Show only the slides and your application running. You should not record yourself on your webcam or any other background. Please remember to narrate what you are showing, it is not supposed to be a silent movie. Also, please do not read your slides to me. Just explain what it is you are showing me on the current screen.

Please submit the link of your YouTube video by copying the link into a pdf document and submitting the file on the DLE.

- Please ensure you submit the link to the hosted video, **NOT** the video itself.
- The video must be hosted on YouTube and unlisted; please check access permissions. If you mark your video **as private it will not be viewable and therefore unable to be marked;**

you will get a zero mark in this case. My suggestion is that you ask a friend to test whether they can access your video before you submit it.

WARNING: This is a serious mistake to make. Please be careful!

4.3 Marking Criteria

This assessment is graded according to the criteria provided below.

Assessment Criteria: Interim Practice (30% of total Practice grade; the practice grade accounts for 20% of the overall mark)

Level descriptors. Note that these definitions are indicative of expected standards at each level, and may not be precise descriptors of the project submitted.

Category and marks weighting:	<30%	30-39%	40-49%	50-59%	60-69%	70-79%	80-100%
Planning (15%) Planning; define the application and what the application will do. Sprint plans appear appropriate to meet aims of project.	Little/no evidence of having prepared or followed a structured project plan.	Inadequate project planning; tasks vague &/or illogically planned.	Some project planning and logic but needs much more thought, detail &/or clarity.	Project mostly well planned & managed. Some questionable logic &/or may lack detail. Prototyping sprints in place	Project generally well planned and managed. Clear evidence of prototyping sprints being implemented	Clear and logical project planning; dynamic response to setbacks/new discoveries. 2 sprints carefully planned. Clear evidence of sprint plans and reviews.	Strong project management throughout all stages of the project clearly demonstrated.
Project vision and background (15%)	Little or no discussion regarding project vision and background	Project vision and background superficial and lacking in structure	Project vision and background communicated ok, but format could be improved	Project vision and background described well overall but with some errors or omissions	Project vision and background good, clearly communicated and well-articulated	Project vision and background as previous but with excellent justification with clear evidence of understanding context	Project vision and background clear, precise, meaningful with excellent understanding of background demonstrated

Requirement Analysis and Design (35%). Define the requirements, user stories and UML diagrams.	Images/graphs/figs sparse, illegible &/or irrelevant.	Images/graphs/figs do not convey required information.	Most images/graphs/figs convey required information but may lack clarity &/or contain errors.	Mainly appropriate images/ graphs/figs - aesthetics &/or labelling could improve.	Most images/graphs/figs of high standard; occasional minor errors/issues.	Images/graphs/figs of high standard, clearly conveying all required information. Architecture has a good complexity	Excellent images/graphs/figs.
Communication of information and Issues/challenges faced (20%) The presentation should be pitched at an audience that is scientifically literate, but non-expert in this particular subject specialism.	Little/no explanation of what was done over the course of the project.	Explanation provides little insight into what was done over the course of the project.	Some explanation of what was done - rather vague/confusing.	A useful explanation of what was done - clarity could improve.	A clear explanation of what was done over the course of the project.	A clear, concise and informative explanation of what was done.	Explanation clearly conveys technically demanding project work to non-expert audience.
Demonstration (15%) Verbal presentation of the project (what you did, why, and what you discovered).	Little/no ability to verbally communicate technical information.	Inadequate verbal communication of technical information.	Verbal communication of technical information is very difficult to follow.	Mostly effective verbal communication of relevant concepts/outcomes.	Good verbal communication of relevant concepts/outcomes.	Clear and eloquent verbal presentation of relevant concepts/outcomes.	Expert verbal communication; concepts/outcomes pitched at the right audience level.

5 Final Presentation

This deliverable accounts for 70**%** of your **practice** element and it is worth 14 module marks.

Within 6 minutes you must present the following, in this exact order:

1. The application executing.
2. Your final project plan and sprints.
3. One of the more complex set of UML diagrams.
4. High level summary of Issues and challenges faced.
5. Show the poster.

You can run your project either on your laptop or on a lab PC. It is strongly recommended that you arrive earlier and make sure that your application runs appropriately. If you exceed the 6 minutes you will be stopped and marked based on what you presented in the 6 minutes slot. Also note that to better present the items (2)-(5) above, you need to create PowerPoint slides.

The presentations are run as a conference style event with 4 tracks. The 4 tracks keep similar themed projects together. For more information about the time and place of your presentation see the '***presentations.pdf'*** file under the assessments tab.

Assessment Criteria: Final Practice (70% of total Practice grade; the practice grade accounts for 20% of the overall mark)

	Level descriptors. Note that these definitions are indicative of expected standards at each level, and may not be precise descriptors of the project submitted.						
Category and marks weighting:	<30%	30-39%	40-49%	50-59%	60-69%	70-79%	80-100%
Communication of information (50%) The presentation and poster should be pitched at an audience that is scientifically literate, but non-expert in this particular subject specialism. You should communicate: • What has been done so for the whole course of the project. • A summary of project results/discussion	Task totally unclear, undefined &/or irrelevant.	Inadequate task definition; very poorly defined &/or explained.	Task definition under-developed; vague, illogical &/or poorly explained.	Task clarity &/or explanation could improve.	Clear and sensibly defined task - well explained.	Task very well defined and explained.	Perfect task definition; concise, compelling and very clearly explained.
	Little/no explanation of what was done over the course of the project.	Explanation provides little insight into what was done over the course of the project.	Some explanation of what was done - rather vague/confusing.	A useful explanation of what was done - clarity could improve.	A clear explanation of what was done over the course of the project.	A clear, concise and informative explanation of what was done.	Explanation clearly conveys technically demanding project work to non-expert audience.
	Little/no presentation of project results.	Minimal insight into key project results.	Some presentation of project results - rather vague/confusing.	Useful presentation of project results - clarity could improve.	Clear presentation of key project results.	Key project results are efficiently, creatively and clearly presented.	Innovative presentation of results - appropriate to non-expert audience.
	Absent/irrelevant conclusions.	Inadequate/unjustified conclusions.	Conclusions vague and/or largely unjustified.	Relevant conclusions. Accuracy, evidence &/or clarity could improve.	Logical conclusions predominantly evidence-based and clearly presented.	Appropriate, well presented and well justified conclusions.	Clear, concise and fully quantitively justified conclusions.
	Text &/or mathematical notation incomprehensible.	Inappropriate text &/or mathematical notation.	Poor literacy &/or mathematical notation.	Mainly appropriate text and mathematical presentation - could improve.	Clear, concise text and mathematical presentation.	Efficient, appropriate use of text. Clear, unambiguous mathematical presentation.	Text/mathematical presentation of professional poster standard.

	Images/graphs/figs sparse, illegible &/or irrelevant.	Images/graphs/figs do not convey required information.	Most images/graphs/figs convey req'd info but may lack clarity &/or contain errors.	Mainly appropriate images/ graphs/figs - aesthetics &/or labelling could improve.	Most images/graphs/figs of high standard; occasional minor errors/issues.	Images/graphs/figs of high standard, clearly conveying all required information.	Creative images/graphs/figs; professional poster standard.
Demonstration (50%) Verbal presentation of the project (what you did, why, and what you discovered). Your response to the moderator's questions should demonstrate a deep understanding of the subject matter and the implications of your results. Software should be shown running	Little/no ability to verbally communicate technical information.	Inadequate verbal communication of technical information.	Verbal communication of technical information is very difficult to follow.	Mostly effective verbal communication of relevant concepts/outcomes.	Good verbal communication of relevant concepts/outcomes.	Clear and eloquent verbal presentation of relevant concepts/outcomes.	Expert verbal communication; concepts/outcomes pitched at the right audience level.
	Little/no coherent response to moderator's questions.	Vague, very confused &/or factually incorrect response to moderator's questions.	Response to moderator's questions shows limited grasp of necessary concepts.	Some sensible responses to moderator's questions - a bit naive &/or lacking depth.	Mostly clear, sensible responses to moderator's questions.	Clear, well-informed response to all moderator's questions.	Comprehensive, expert response to moderator's questions.

6 Final Portfolio

This deliverable accounts for 100**%** of your coursework element and it is worth 80 module marks.

Your final portfolio comprises the following items:

- A representation of your final project plan. This should illustrate your backlog and sprints.
- A complete set of UML diagrams to represent your application.
- A completed and finished GitHub repo. The readme file must outline your project.
- A poster illustrating the key features of your application and the architecture.
- A reflective account of approximately 500 words considering the issues and challenges arising from the project.

You must submit a design document, using the template provided (see the ***Template.pdf*** file under the assessments tab). To get all the marks, your report must contain all the sections shown in the template. When using the template provided, please remember to delete the light grey prompt text. Note that under the assessments tab you can also find students' design documents from previous years; these are provided to give you a better understanding of what you need to submit; they are not meant to show best practice.

Do not forget to include a **link to a public GitHub repositor**y. Although it is recommended to start working on a private GitHub repository, do not forget to make your repository public when you submit the coursework; if it is private the markers will not be able to assess your code and thus you will lose many marks.

This document must be submitted electronically via the module DLE prior to the deadline. The document must be in **PDF format**. DO NOT try and submit a different file type. The submission **cannot be anonymous** as you will be required to present the highlights of your report at the Final Presentation event.

The approximate word count is 3000 +/- 10% as per university rules. You must plan your time appropriately to allow time to draft and re-draft.

Assessment Criteria: **Final Portfolio**

Level descriptors. Note that these definitions are indicative of expected standards at each level, and may not be precise descriptors of the project submitted.

Category and marks weighting:	<30%	30-39%	40-49%	50-59%	60-69%	70-79%	80-100%
Plans (15%) Structured project plan for complete project produced. Sprint plans map to backlog and aims. Releases made every two weeks with appropriate plans and reviews.	Application to SDLC totally unclear, undefined &/or irrelevant.	Inadequate SDLC definition; very poorly defined.	SDLC definition under-developed; vague &/or illogical.	SDLC clarity &/or validity could improve.	Clear and sensibly defined SDLC.	SDLC very well defined, explained and justified. Evidence of reading beyond module materials provided.	Perfect SDLC definition showing understanding and reading beyond the taught materials; clear link to plan showing stages
	Little/no evidence of having prepared or followed a structured project plan.	Inadequate project planning; tasks vague &/or illogically planned.	Some project planning and logic but needs much more thought, detail &/or clarity.	Project mostly well planned & managed. Some questionable logic &/or may lack detail. Prototyping sprints in place	Project generally well planned and managed. Clear evidence of prototyping sprints being implemented	Clear and logical project planning; dynamic response to setbacks/new discoveries. All sprints carefully planned. Clear evidence of sprint plans and reviews.	Strong project management throughout all stages of the project clearly demonstrated.
Requirements (25%) Backlog shows how requirements finally realised. Implementation of agile artifacts match proposed plans deviation from plan discussed appropriately.	Little or no discussion regarding project vision	Project vision superficial and lacking in structure	Project vision communicated ok but format could be improved	Project vision described well overall but with some errors or omissions	Project vision good, clearly communicated and well articulated	Project vision as previous but with excellent justification with clear evidence of understanding context	Project vision clear, precise, meaningful with excellent understanding of background demonstrated

Design documentation illustrates functional and non-functional requirements of final project. Diagrams clearly match code implemented in all sprints Design documentation shows existing prototype implementation	Little or no indication of requirements	Requirements present but vague and poorly defined. Diagrams not provided.	Requirements present but need more thought and development. Few diagrams present but lack of understanding demonstrated.	Requirements presented ok with appropriate diagrams. Some questionable logic demonstrated. Some errors and omissions	Requirements clear and sensibly defined. Requirements appropriate for complexity of project. Coverage of application is appropriate.	Requirements have good depth of coverage for application. Diagrams are clearly derived from requirements.	Requirements excellent with little further to add. Diagrams match with requirements and could be used in professional setting.
	Little or no indication of requirements having been implemented	Requirements shown as implemented in sprints inconsequential	Requirements shown as implemented but illustration not clear and lacks clarity	Requirements matched to implementation in sprints ok, some errors and omissions	Requirements matched to implementation are appropriate for all sprints.	Requirements matched to implementation are clearly indicative of minimum viable product for sprints	Requirements matched to implementation are of professional standard.
Issues (30%) Consideration given to any changes in constraints of physical system, cyber security and network as identified earlier. Showing a clear understanding of relevant subject matter throughout the project. Any changes to Legal, Social, Ethical and Professional issues clearly and appropriately discussed along with appropriate critical comment and authoritative citations.	Absent/irrelevant issues.	Inadequate/ unjustified issues.	Issues vague and/or largely unjustified.	Relevant issues. Accuracy, evidence &/or clarity could improve.	Logical issues predominantly evidence-based and clearly presented.	Appropriate, well presented and well justified issues.	Clear, concise and fully quantitively justified issues.
	Absent/irrelevant constraints.	Inadequate/ unjustified constraints.	Constraints vague and/or largely unjustified.	Relevant constraints. Accuracy, evidence &/or clarity could improve.	Logical constraints predominantly evidence-based and clearly presented.	Appropriate, well presented and well justified constraints.	Clear, concise and fully quantitively justified constraints.

	Missing/irrelevant list of references. Absence of valid citations throughout report.	Superficial reference list &/or predominately incorrect/inadequate citations.	Reference list present. Major presentation issues &/or poor citation style.	Reference list &/or citations could be more clearly/correctly presented.	Most citations & reference list entries correctly presented; some minor issues.	Highly competent use of citations and reference list.	Extensive list of references & citation style: peer reviewed journal standard.
	Little or no evidence of understanding of appropriate LSEP issues.	Inadequate understanding of relevant LSEP issues. Vague mentions made with little demonstrable understanding	Some understanding of LSEP issues with errors &/or poor conceptual framework.	Reasonable understanding of LSEP issues; minor errors. May require more depth.	Good understanding of LSEP issues with a clear conceptual framework.	Excellent understanding of LSEP issues beyond the level of taught modules.	Expert understanding of LSEP issues with rigorous attention to relevant detail.
Implementation (30%) Evidence of appropriate implementation for a 40 credit module. Demonstration of skill in implementation Implementation of agile artifacts match proposed plans, deviation from plan discussed appropriately. Implementation of code at appropriate level with demonstration of good software engineering principles such as DRY, YAGNI and SOLID.	Little or no evidence of coding skills in project implementation.	Poor skills in implementing code - incorrect &/or very confused.	Some skill in implementing the software, but with errors &/or confusion. Application provides more functionality than log in and registration.	Skill in most areas of software implementation - some issues/errors. Implementation of moderate complexity with suitable functionality demonstrated.	Competent implementation of software with minor issues/errors. Application is of suitable complexity, has appropriate storage for any data, architecture is not monolithic but demonstrates interactions	Highly skilled implementation of software (far beyond the level of taught modules). Application has good complexity and shows good quality software engineering. Data storage if required is good, application goes way beyond a form of data storage with front end.	Expert level of skill in all relevant areas clearly evident throughout. Software is of commercial quality and could be implemented in real world situation with very little modification.

					between levels and/or layers of software.		
	Little or no evidence of relevant skills in project implementation.	Poor skills in implementing the methodology - incorrect &/or very confused.	Some skill in implementing the methodology, but with errors &/or confusion.	Skill in most areas of methodology implementation - some issues/errors.	Competent implementation of methodology with minor issues/errors.	Highly skilled implementation of methodology (far beyond the level of taught modules).	Expert level of skill in all relevant areas clearly evident throughout.
	Little or no relevant work done to achieve project aims & objectives.	Minimal relevant work done to achieve project aims & objectives.	Some relevant work but far short of that expected in the time allocated for 40 credits.	Project engagement &/or depth of coverage of task could improve.	Good project engagement and coverage of the task.	Good depth of coverage of the task, showing a high level of project engagement.	Comprehensive coverage of a highly demanding task. Very high level of engagement.

7 General Guidance

7.1 Extenuating Circumstances

There may be a time during this module where you experience a serious situation which has a significant impact on your ability to complete the assessments. The definition of these can be found in the University Policy on Extenuating Circumstances here:
https://www.plymouth.ac.uk/uploads/production/document/path/15/15317/Extenuating_Circumstances_Policy_and_Procedures.pdf

7.2 Referral

Students who fail the module or have Extenuating Circumstances (EC's) for non-submission may be offered the chance to carry out a referral over the summer. The referral is suitable only for those who have carried out some work during the academic year. It is not suitable for those who have not been able to work on the project during the academic year. Those students should ask to repeat.

This module has two elements, and it is possible to be referred in one or both elements. Students unable to complete their practice element, (for example, due to EC's were unable to submit the Interim Video or attend the Final Presentation) should claim their EC's for non-submission and providing their referred credits do not exceed the current limit (60 credits) you will be invited to present at a Final Presentation day in the August referral period. This will take place during the University referred examinations period. The information of the exact date can be found here https://www.plymouth.ac.uk/student-life/your-studies/essential-information/regulations/important-dates

7.3 Plagiarism

All of your work must be of your own words. You must use references for your sources, however you acquire them. Where you wish to use quotations, these must be a very minor part of your overall work.

To copy another person's work is viewed as plagiarism and is not allowed. Any issues of plagiarism and any form of academic dishonesty are treated very seriously. All your work must be your own and other sources must be identified as being theirs, not yours. The copying of another persons' work could result in a penalty being invoked.

Further information on plagiarism policy can be found here:

Plagiarism: https://www.plymouth.ac.uk/student-life/your-studies/essential-information/regulations/plagiarism

Examination Offences: https://www.plymouth.ac.uk/student-life/your-studies/essential-information/exams/exam-rules-and-regulations/examination-offences

Turnitin (http://www.turnitinuk.com/) is an Internet-based 'originality checking tool' which allows documents to be compared with content on the Internet, in journals and in an archive of previously submitted works. It can help to detect unintentional or deliberate plagiarism.

It is a formative tool that makes it easy for students to review their citations and referencing as an aid to learning good academic practice. Turnitin produces an 'originality report' to help guide you. To learn more about Turnitin go to:
https://guides.turnitin.com/01_Manuals_and_Guides/Student/Student_User_Manual

7.4 Referencing

The University of Plymouth Library has produced an online support referencing guide which is available here: http://plymouth.libguides.com/referencing
Another recommended referencing resource is Cite Them Right Online; this is an online resource which provides you with specific guidance about how to reference lots of different types of materials.

The Learn Higher Network has also provided a number of documents to support students with referencing:
References and Bibliographies Booklet:
http://www.learnhigher.ac.uk/writing-for-university/referencing/references-and-bibliographies-booklet/
Checking your assignments' references:
http://www.learnhigher.ac.uk/writing-for-university/academic-writing/checking-your-assigments-references/