COMP2000
Software Engineering 2

20 CREDIT MODULE

ASSESSMENT: 100% Coursework

W1: 30% Set Exercises
W2: 70% Assignment

MODULE LEADER: Dr Alaa Alkhafaji

MODULE AIMS

- To learn about topics that instil best practice into the students' software development activity.
- To explore a range of commonly used programming paradigms.
- To understand the benefit of using standardised design patterns.

ASSESSED LEARNING OUTCOMES (ALO):

1. Apply HCI and parallelism to construct more efficient and usable software.
2. Illustrate the relative merits of various programming paradigms
3. Use appropriate programming paradigms to implement software that solves a given problem.
4. Compare standard design patterns and explain why they are an important aspect of software engineering

Overview

This document contains all the necessary information pertaining to the assessment of *COMP2000 software engineering 2*. The module is assessed via **100% coursework**, across two elements:

1- *30%* Set Exercises
2- 70% Assignment Submission.

The sections that follow will detail the assessment tasks that are to be undertaken. The submission and expected feedback dates are presented in Table 1. All assessments are to be submitted electronically via the respective DLE module pages before the stated deadlines.

	Submission Deadline	Feedback
Set Exercises (30%)	**21/11/2023**	**18/12/2023**
Assignment Submission (70%)	**15/12/2023**	**16/1/2024**

Table 1: Assessment Deadlines

All assessments will be introduced in class to provide further clarity over what is expected and how you can access support and formative feedback prior to submission. Whilst the assessment information is provided at the start of the module, it is not necessarily expected you will start this immediately – as you will often not have sufficient understanding of the topic. The module leader will provide guidance in this respect.

Set Exercises

There are four set exercises for you to complete during the module. The exercises should be submitted via the DLE by the deadline shown in the table above. The answers for all six exercises should be submitted in **one single PDF file**, no other format is accepted. **All exercises should be clearly labelled, and the answers should be categorised under each corresponding set exercise number.**

Assessment 1: Set Exercises

CW1 Overview

There are four set exercises outlined below that will allow you to demonstrate your knowledge and understanding around user graphical design, mobile applications, usability evaluation. Submit a single PDF file that contains the outputs from each of the exercises in the same order as they appear below. All exercises should be clearly labelled and the outputs for each exercise should be clearly described under each corresponding label.

You are to use the scenario in appendix A and base your four exercises.

Set Exercise 1: 15 marks

- Analyse the scenario to Identify the context of use and the user(s) of the system.

Set Exercise 2: 35 marks

- Produce a UIs design for a client-side mobile application. The design should follow the HCI principles that you learnt throughout the module. You might consider different screens' size for different mobile devices.
- Produce a storyboard with a suitable narrative.

Set Exercise 3: 35 marks

- produce a high-fidelity prototype for the system (you are encouraged to use Figma for this design). **NOTE: coding is not required in this stage and will not attract more marks if you do so.** Please refer to the link on the DLE for a basic Figma training course in LinkedIn
  - The interfaces should be interactive, and all links should be working.
  - Active menus and icons should be interactive.
  - Notifications should be shown to notify users where appropriate and based on the given scenario.

Set Exercise 4: 15 marks

- Error, feedback, and process messages should appear where appropriate to give feedback to users.

Deliverable

Submit **<u>a single PDF file</u>** to the DLE submission point for this module.

- The PDF document reporting your work ***<u>should show evidence for each exercise and all exercises should be clearly labelled</u>***:
  - o The users of the system.
  - o The initial design of the UIs and the storyboard (i.e. screenshots and/ or sketches) with the appropriate narratives.
  - o The storyboard of the final design with the appropriate narratives.
  - o The high-fidelity prototype (i.e. screenshots and the link to the to the prototype) (**<u>NOTE: it's student's responsibility to make sure the link is accessible</u>**).

Assessment 2: Assignment Submission

You are required to produce an android application to work with The RESTful API to continue the work you carried out in CW1. The API details is provided in appendix A below with the user stories. The API is provided at this link: (https://web.socem.plymouth.ac.uk/COMP2000/ReservationApi/api/Reservations).

- Your implementation MUST be in Java programming language using Android Studio (**any other languages will receive zero marks**).
- You must create a worker thread to handle the connection to the API.
- Use design patterns that you learnt throughout the module.
- You must use the design you produced in the previous assessment (CW1).
- Carry out a usability heuristic evaluation (discuss the design of your project in terms of the 10 usability heuristic) (**participants are not involved in this stage and it's out of the scope of this assessment**).
- You MUST use Git version control with GitHub in your development process.
- You are to use the GitHub Classroom set up for this module – the link for signing up is here (https://classroom.github.com/a/xE6Gl8Y5). You will be assessed on the appropriate use of version control.
-

The Java project will provide a graphical user interface (GUI) for usage. The GUI should provide the functionality listed in the Scenario in appendix A. You will be assessed on appropriate use of version control.

Assessment Criteria:

Individual Coursework comprising one assignment with a GIT repository of incremental development leading to a final build or release. Repository includes itemised deliverables for final module submission. All code to be version controlled and commented, all 3rd party assets and resources to be formally credited in the README file on the repository.

Deliverables

Submit a single PDF document reporting your work to the DLE submission point for this module.:

- The design of the UIs (sketches and/or screenshots)
- System architecture (diagrams with suitable narratives)
- Usability heuristic (discuss your UIs design in terms of the usability heuristic principles)
- A link to the GitHub repository for your project. **It's students' responsibility to make sure the link is accessible, which otherwise will receive zero marks.**
- Screenshots of the interfaces of the final working system with a suitable narrative.
- A link to the YouTube video rundown of your application and highlighting the main requirements. **It's students' responsibility to make sure the video is accessible, which otherwise will receive zero marks.**

Deliverables in detail

GIT Repository

An online repository of your project using the GIT service. You are Graded on the following criteria:

- Repository readme file includes all additional resources (art, sound FX etc.) fully credited
- No previous versions of the project are present in the repository in a .zip or other compressed format
- Commits to the repository are appropriately commented
- Commits are in a consistent timely manner, at least once every week

A PDF document reporting your work:

The full details of the design, implementation and evaluation studies that were carried out including:

- Interface design (screenshots/ sketches, and diagrams).
- Details of the implementation phase highlighting the design choices including design patterns, worker thread and push notifications.
- Testing and evaluation. Test your implementation using a testing method of your choice. Discuss your UIs design in terms of Heuristic usability. Summarise the outcome and, highlight the strength and weaknesses, and how you would improve your work.

Video rundown of your application

- Demonstrate the functionality of your project in a short video showing the program running with actual data and user interaction.
- The video should be between 3 minutes long and **in the normal pace**, which otherwise will not be marked; **<u>everything after 3 minutes will not be marked</u>**
- Please:
- You may be invited to an interview (either online or in-person) for further discussion at a later date.

Video settings

File-Type:	MP4
Resolution:	720p or 1080p
Framerate:	30
Video Bitrate:	16 MBS
Audio Bitrate:	Mono – 128 kpbs, Stereo – 384 kpbs
Compression:	H.264

You must present the work carried out in a report submitted in **<u>a single PDF format and no other format is acceptable.</u>** Your report must be approximately 2000 words, please use screenshots and links to code files to illustrate functionality where appropriate to illustrate the design of the interfaces.

The report must contain the following sections:

1. Introduction (approximately 2 paragraphs). Introduce the document and signpost the reader to what they will find in it. **Provide the links to the GitHub repository and the YouTube video that is demonstrating your app running**.
2. Background. Explain here the scenario, what information your application provides, who are the potential users.
3. Legal, Social, Ethical and Professional (LSEP). Highlight issues might arise in terms of privacy, integrity, security and discuss how you addressed these issues.
4. Design. Present here the design of the app with sketches/ screenshots and system architecture.
5. Implementation. Illustrate with screenshots and hyperlinks to your source code in your GitHub repo, how you implemented your design. Provide a suitable narrative to help your reader understand your screenshots and diagrams. Discuss any design pattern adopted in the implementation. Go through the code and explain your implementation briefly.
6. Testing and evaluation. Discuss your UIs design in terms of the usability heuristic principles and, highlight the strength and weaknesses in your design. Present how you tested your implementation and functionality clearly indicating areas for further work and improvement.
7. Summary. Summarise your work briefly and highlight the main points in each section.
8. References. State here all references you used.

The Rubric below will be used to assess and provide feedback on the submissions.

Table 1: Marking scheme for Assessment 2: Assignment Submission

	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%
Application (40%) Implementation of interface design is appropriate. Implementation of code demonstrates good quality coding approaches. Application is shown running in the video. Implementation of code uses appropriate design patterns. Clear indication of innovative and proactive thought going beyond materials provided. GitHub commits are appropriate and not left to just before deadline.	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 architecture which is not monolithic but demonstrates interactions between levels and/or layers of software.	Highly skilled implementation of software (far beyond the level of taught modules). Application has good complexity and shows good quality software engineering.	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.
	Little or no indication of user requirements having been implement ed	User Requirements shown as implemented inconsequential	User Requirements shown as implemented but illustration not clear and lacks clarity	Requirements matched to implementation ok, some errors and omissions	Requirements matched to implementation are appropriate. Meaningful choice of requirements implemented.	Requirements matched to implementation are clearly indicative of minimum viable product	Requirements matched to implementation are of professional standard.
	Little to no indication of any attempt to go beyond the coursework teaching/brief.	Inadequate and poorly defined plan	Plan for innovation vague and/or largely unjustified	Relevant features considered. Accuracy, evidence &/or clarity could improve	Logical consideration given to innovative features, predominantly evidence-based and clearly articulated	Appropriate, well presented and well justified innovations	Clear, concise and fully justified innovation plan. Shows original thinking and proactive development

Design and Testing (40%) Appropriate use of UI components to provide a suitable HCI experience Interface design illustrates clear application of robust usability testing and HCI principles. Application architecture demonstrates suitable use of parallelism Tests have been designed to evaluate application appropriately	Project is devoid of appropriate testing plan	Poor skills in applying testing, incorrect &/or very confused	Some relevant testing applied. V&V superficial, sparse &/or often flawed	Appropriate testing in place but with some omissions, issues &/or errors	Competent testing plan in place. Appropriate Validation and Verification approach in place.	Highly competent testing regime in place both in plan and implementation. Shows a deep understanding of testing above and beyond taught modules.	Expert testing plans and implementations in place, could be appropriate for commercial application with very little modification.
	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.	Requirement s 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.
	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; peer reviewed journal standard.

Approach (20%) Illustration provided for how HCI and parallelism applied Coding approach outlined How application meets needs for scenario demonstrated. Reflection at an appropriate depth and understanding demonstrated of design patterns and their application.	Writing incomprehensible.	Inappropriate written work	Poor literacy	Mainly appropriate style of writing and presentation - could improve.	Clear style of writing and presentation.	Lucid style of writing. Clear, unambiguous presentation.	Literacy presentation: peer reviewed journal standard.
	Little to no indication of any attempt to go beyond the coursework teaching/brief.	Inadequate and poorly defined plan	Plan for innovation vague and/or largely unjustified	Relevant features considered. Accuracy, evidence &/or clarity could improve	Logical consideration given to innovative features, predominantly evidence-based and clearly articulated	Appropriate, well presented and well justified innovations	Clear, concise and fully justified innovation plan. Shows original thinking and proactive development

General Guidance

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/22/22876/Extenuating_Circumstances_Policy_and_Procedures.pdf

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://www.plymouth.ac.uk/about-us/teaching-and-learning/digital-education/turnitin-for-students

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: https://www-citethemrightonline-com.plymouth.idm.oclc.org/; this 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/

Appendix A

Product Vision

A big restaurant that is near the sea wants to develop a mobile application to enable a booking facility for customers through a mobile phone. The reservation could be for breakfast, lunch or dinner. The restaurant has an inside and outside seating area, and the maximum table size is 10 people. Also, it's busy, so the reservation should be a week in advance. The app is to enable customers make and manage their booking (i.e. view and edit the booking's details). Customers could choose where to sit (inside or outside, seaside or garden side). Customers could cancel their booking 24 hours in advance. The app is supposed to push notifications to inform users, when a booking approved/declined, and also to inform of any update regarding the booking (e.g. when bookings get updated/ cancelled). Users need to login to their account in order to perform the functionalities. Users should be able to turn notifications on/off when needed. Users can manage preferences in their account (e.g. set their favourite meals or location within the restaurant).

The customer side:

Functional Requirements

- As a customer I wish to view the available tables.
- As a customer I wish to make a booking (specific time, date, mealtime, table's size and seating area (inside or outside).
- As a customer I wish to manage my booking (edit/cancel).
- As a customer I wish to add a review
- As a customer I wish to view my previous bookings (history).