College of Engineering, Construction and Living Sciences
Bachelor of Information Technology
ID721001: Mobile Application Development
Level 7, Credits 15

Project

Assessment Overview

In this **individual** assessment, you will develop

Learning Outcomes

At the successful completion of this course, learners will be able to:

1. Implement and publish complete, non-trivial, industry-standard mobile applications following sound architectural and code-quality standards.
2. Identify relevant use cases for a mobile computing scenario and incorporate them into an effective user experience design.
3. Follow industry standard software engineering practice in the design of mobile applications.

Assessments

Assessment	Weighting	Due Date	Learning Outcome
Practical	20%	13-11-2024 (Wednesday at 4.59 PM)	2, 3
Project	80%	13-11-2024 (Wednesday at 4.59 PM)	1, 2, 3

Conditions of Assessment

You will complete majority of this assessment during your learner-managed time. However, there will be time during class to discuss the requirements and your progress on this assessment. This assessment will need to be completed by **Wednesday, 13 November 2024** at **4.59 PM**.

Pass Criteria

This assessment is criterion-referenced (CRA) with a cumulative pass mark of **50%** over all assessments in **ID721001: Mobile Application Development**.

Authenticity

All parts of your submitted assessment **must** be completely your work. Do your best to complete this assessment without using an **AI generative tool**. You need to demonstrate to the course lecturer that you can meet the learning outcome for this assessment.

However, if you get stuck, you can use an **AI generative tool** to help you get unstuck, permitting you to acknowledge that you have used it. In the assessment's repository **README.md** file, please include what prompt(s) you provided to the **AI generative tool** and how you used the response(s) to help you with your work. It also applies to code snippets retrieved from **StackOverflow** and **GitHub**.

Failure to do this may result in a mark of **zero** for this assessment.

Policy on Submissions, Extensions, Resubmissions and Resits

The school's process concerning submissions, extensions, resubmissions and resits complies with **Otago Polytechnic** policies. Learners can view policies on the **Otago Polytechnic** website located at https://www.op.ac.nz/about-us/governance-and-management/policies.

Submission

You **must** submit all program files via **GitHub**. The latest program files in the **master** or **main** branch will be used to mark against the **Documentation** criterion. Please test your **master** or **main** branch application before you submit. Partial marks **will not** be given for incomplete functionality. Late submissions will incur a **10% penalty per day**, rolling over at **5:00 PM**.

Extensions

Familiarise yourself with the assessment due date. Extensions will **only** be granted if you are unable to complete the assessment by the due date because of **unforeseen circumstances outside your control**. The length of the extension granted will depend on the circumstances and **must** be negotiated with the course lecturer before the assessment due date. A medical certificate or support letter may be needed. Extensions will not be granted for poor time management or pressure of other assessments.

Resits

Resits and reassessments **are not** applicable in **ID721001: Mobile Application Development**.

Instructions

Documentation - Learning Outcomes 2, 3 (100%)

For each game, in a **Microsoft Word** document, explain the following:

- Introduction

  - **Purpose** - Define the purpose of UAT, which is to ensure that the mobile games meet the main features and mechanics.
  - **Scope** - Outline the scope of UAT, including the main features and mechanics to be tested.
  - **Objectives** - Validate that the mobile games are user-friendly and confirm that the mobile applications meet the main features and mechanics.

- Test Preparation
  - **Test Plan** - Detail the testing strategy, scope, resources, schedule and deliverables.
  - **Test Cases** - Develop test cases based on the main features and mechanics.
  - **Test Environment** - Define the devices and operating systems for testing. For example, screen sizes, iOS versions and Android versions.
- Execution - Conduct functionality, usability and compatibility testing.
- Evaluation
  - Collect feedback from users regarding functionality, usability and overall experience. Use surveys and interviews.
  - Document and prioritise any issues and bugs reported during testing. If so, provide steps to reproduce and screenshots.
  - Retest any resolved issues and bugs to confirm that they have been fixed. Verify that no new issues and bugs have been introduced.
- Reporting
  - Summarise the testing process, coverage and overall findings.
  - Detail the pass/fail status of each test case.
  - Provide a list of open issues and bugs.
  - Offer recommendations for improvements based on the test results and user feedback.