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 two mobile applications using **React Native** and **Expo**. In addition, marks will be allocated for code quality and best practices, documentation and **Git** usage.

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**.

Submission

You **must** submit all application files via **GitHub Classroom**. Here is the URL to the repository you will use for your submission – https://classroom.github.com/a/Nh2sKWnc. If you do not have not one, create a **.gitignore** and add the ignored files in this resource - https://raw.githubusercontent.com/github/gitignore/main/VisualStudio.gitignore. Create a branch called **project**. The latest application files in the **project** branch will be used to mark against the **Functionality** criterion. Please test 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**.

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 | Te Pūkenga** policies. Learners can view policies on the **Otago Polytechnic | Te Pūkenga** website located at https://www.op.ac.nz/about-us/governance-and-management/policies.

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

You will need to submit a mobile application and documentation that meet the following requirements:

Functionality - Learning Outcomes 1, 2, 3 (50%)

- Cookbook application
  - The mobile application needs to run without code or file structure modification in **Visual Studio Code**.
  - Usable on a variety of mobile devices, i.e., devices with different screen sizes.
  - Free of bugs that significantly affect the usability.
  - Food data needs to be fetched from **food-data.json**.
  - Display **bottom tab navigation** with the following screens:
    - Milestone One: Daily specials - Due Monday, 26 August 2024 (Week 6) at 4.59 PM
      - This screen will display six random recipes from the **food-data.json** file.
      - Display the random recipes in a **FlatList**.
      - Each recipe item in the **FlatList** needs to display the recipe's name and image. Truncate the recipe's name if it is too long.
      - When a recipe item is pressed, display the recipe's name, image, cuisine, ingredients and instructions in a **ScrollView**.
    - Milestone Two: Recipes - Due Monday, 2 September 2024 (Week 7) at 4.59 PM
      - This screen will display all cuisines from the **food-data.json** file.
      - When a cuisine item is pressed, display all recipes for that cuisine in a **FlatList**.
      - Each recipe item in the **FlatList** needs to display the recipe's name, description and image. Truncate the recipe's name and description if it is too long.
      - When a recipe item is pressed, display the recipe's name, image, cuisine, ingredients and instructions in a **ScrollView**.
    - A heart icon needs to be displayed in the top right corner of the screen. When the heart icon is pressed, the recipe is added to the **Favourites** screen. Persist the favourite recipes using **AsyncStorage**.
    - A plus icon needs to be displayed next to the heart icon. When the plus icon is pressed, the recipe's ingredients are added to the **Shopping list** screen. Persist the shopping list using **AsyncStorage**.
    - Milestone Three: Favourites - Due Monday, 9 September 2024 (Week 8) at 4.59 PM
      - This screen will display all recipes that have been added to the **Favourites** screen.
      - Display the favourite recipes stored in **AsyncStorage** in a **FlatList**.
      - Ability to delete a favourite recipe from the **FlatList**.
    - Shopping list - Due Monday, 16 September 2024 (Week 9) at 4.59 PM
      - This screen will display all ingredients from the recipes that have been added to the **Shopping list** screen.
      - Display the shopping list stored in **AsyncStorage** in a **FlatList**.
      - Ensure there are no duplicate ingredients in the **FlatList**.
      - Ability to delete an ingredient from the **FlatList**.
    - Appropriate image used for the splash screen and app icon.
    - Visually attractive UI with a coherent graphical theme and style using **Tailwind CSS**.
- Milestone Four: Application of your choice - Due Wednesday, 13 November 2024 (Week 15) at 4.59 PM
  - The mobile application needs to run without code or file structure modification in **Visual Studio Code**.
  - Usable on a variety of mobile devices, i.e., devices with different screen sizes.
  - Free of bugs that significantly affect the usability.
  - Ten features **must** be implemented.
  - Store data in either **SQLite** or **Firebase**.
  - Appropriate image used for the splash screen and app icon.
  - Visually attractive UI with a coherent graphical theme and style using **Tailwind CSS**.

Documentation - Learning Outcomes 2, 3 (10%)

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

- Initial design sketches. These can be hand-drawn or created using a design tool. This is due Tuesday, 21 August 2024 (Week 5).
- Known issues and bugs.
- Future improvements.

Code Quality and Best Practices - Learning Outcomes 1, 3 (35%)

- A **Node.js .gitignore** file is used.
- Appropriate naming of files, variables, functions and components.
- Idiomatic use of control flow, data structures and in-built functions.
- Efficient algorithmic approach.
- Sufficient modularity.
- Each **component** file has a **JSDoc** comment located at the top of the file.
- Formatted code.
- No dead or unused code.

Git Usage - Learning Outcomes 2, 3 (5%)

- A **GitHub** project board or issues to help you organise and prioritise your development work. The course lecturer needs to see consistent use of the **GitHub** project board or issues for the duration of the assessment.
- Your **Git commit messages** should:
  - Reflect the context of each functional requirement change.
  - Be formatted using an appropriate naming convention style.

Additional Information

- **Do not** rewrite your **Git** history. It is important that the course lecturer can see how you worked on your assessment over time.
- You need to show the course lecturer the initial **GitHub** project board or issues before you start your development work. Following this, you need to show the course lecturer your **GitHub** project board or issues at the end of each week.