DIP2005/DIP225 Project

Release Date:	2nd October 2025
Due Date:	21st November 2025 (Week 14 Friday) 1100pm Malaysia time
Value:	60% of total assessment
Marks:	Marked out of 100
Assessment Mode:	Group (Max. 2) and Individual Assessment

RATIONALE

This assignment is designed to give students practical experience in building a modern Android application using Kotlin and industry-standard tools. Students will learn how to design accessible and responsive UI with Jetpack Compose, implement multi-screen navigation, integrate with online APIs, and persist data locally with Room and DataStore. They will also practice debugging, testing, and applying clean architecture, equipping them with essential skills to build scalable, user-friendly, and reliable mobile apps suitable for real-world use.

- CLO3: Organize in a group to work on a mobile application

CASE STUDY

You are tasked to develop a Kotlin Android mobile application named SmartFit. The app helps users track their daily activities, such as steps, workouts, and calorie intake. The application should demonstrate modern Android development practices, using Jetpack Compose, Material Design, network integration, local data persistence, navigation, debugging, and testing.

In general, the app should allow users to:

a. Add, view, edit, and delete their activity logs.
b. View daily/weekly summaries of their activities.
c. Receive suggestions or tips from an online resource via REST API.

The detailed functional requirements are as below:

1. User Interface – Material Design

a. Implement UI using Jetpack Compose following Material Design principles.
b. Include at least one animation to enhance user experience.
c. Apply themes (light/dark) consistently throughout the app.
d. Ensure accessibility features such as proper contrast, content descriptions for images, and readable text.

2. Navigation and Adaptive Layouts

a. Create multi-screen navigation (e.g., Home, Activity Log, Profile).
b. Pass data between screens (e.g., selected activity details).
c. Implement dynamic navigation (e.g., redirect to a welcome or dashboard screen after login).
d. Ensure adaptive layout for both phone and tablet orientations

3. Connect to Network

a. Fetch workout suggestions or nutrition data from the Internet using HTTP/REST API.
b. Use coroutines for network calls to avoid blocking the UI.

c. Load and display images (e.g., exercise thumbnails) from the internet.
d. Implement a data layer architecture with repository pattern and manual dependency injection.

4. Data Persistence

a. Store user activity logs locally using Room database.
b. Use DataStore to save user preferences like theme mode or daily step goals.
c. Demonstrate reading, writing, and updating persisted data in the app.

5. Debugging & Testing (Every group member must do at least two for each requirement below)

a. Add logging statements to monitor key operations such as network requests and database updates.
b. Write a unit test to verify a core feature (e.g., calorie calculation or step goal progress).
c. Create an automated UI test to ensure smooth navigation or correct display of fetched data.

TASKS

1. Create a report to guide users to use your mobile app in a user manual style with screen shots provided.
2. Debugging and testing evidence solved by each group member. Make sure to clearly mention in your report which issues were solved by each group member.
3. Create a video to demo your mobile application. Make sure every group member is involved.

SUBMISSION CHECK LIST

i. Soft copy submitted through the LMS **MUST** include the following items:
   a. Report as described in TASKS
   b. Video file in mp4 format.
   c. Android project folder

REMARKS

i. Assignments should be prepared using word processing software with 12 points Times New Roman font, 1.5 spacing, and page numbering.
ii. If you are not clear about the assignment requirements, please contact the module leader.
iii. You may refer to materials from textbooks or any other references, but you must acknowledge the quotation, no matter how brief. Failure to do so will result in zero marks being awarded for the assignment.
iv. A penalty of 5 marks per working day (including Saturday) deduction from the marks obtained for this assignment will be imposed for late submission.
v. If students not able to submit any item in the checklist, students will be awarded 0 mark for this assignment for all group members.

Use of AI generative tools	1. Utilizing AI generative tools should be seen to augment creativity, problem-solving, and productivity while maintaining the integrity of the subject matter and upholding academic and professional standards. 2. Students must demonstrate their own knowledge, skills, and understanding of the subject matter in their work. 3. Where an assignment requires ChatGPT to be cited, you must reference all the content from Generative AI tools that you include. Failure to reference externally sourced, non-original work can result in Academic misconduct. 4. Students are to apply **APA Style** for any AI content generated that is utilized and integrated into the final work. This acknowledgment is to be added to the **footnote** of the respective pages. • **To cite AI-generated work that you did not edit or revise:** Name of AI Tool. (Year, Month Day you generated the content). Exact text of question or prompt you entered [AI-generated text/image/video, etc.]. Name of Company/Developer if different than name of AI tool. URL. Example: ChatGPT. (2023, June 3). "Steps of creating a running website using XAMPP?" [AIgenerated text]. OpenAI. https://chat.openai.com. • **To cite AI-generated work that you edited or revised:** Name of AI Tool & Your Last Name, First Initial. (Year, Month Day you generated the content). Exact text of question or prompt you entered [AI-generated text/image/video, etc.]. Name of Company/Developer if different than name of AI tool. URL. Example: ChatGPT & Chong, L.Y. (2023, July 19). How to setup cloud based CI/CD for a website deployment? [AI-generated text]. OpenAI. https://chat.openai.com/c/9bb6771b-209b-4c8c-ac79-6a8a9f39604a • **General Format for In-Text Citations** Examples: To make fluffy basmati rice, "rinse the rice twice in cold water" (ChatGPT, 2023) Cloud-based CI/CD requires the following steps to be applied (ChatGPT & Chong, L.Y., 2023) 5. Any violation of this policy will result in appropriate academic sanctions, which may include penalizing your grades, failing the course, and other disciplinary actions.