

## Group Project # 1

**Group Project Proposal – This must be approved by the Instructor's Team  
Each Member must upload a copy to the designated dropbox in i-college**

**Due Date Proposal – 10/12/2025 @11:59 pm.**

**Due Date Project 10/26/2025 @ 11:59 pm.**

**\*\*Please read this document carefully, paying close attention to each and every point. Following all instructions thoroughly is crucial for proper understanding, execution, and evaluation.**

A signed individual statement stating "I understand that you will be graded individually on group assignments and may not receive the same grade as all members". You MUST place this statement at the end of the document then create a signature field as in (\_\_\_\_\_). All parties must play an important role as it relates to the implementation from start to finish. FYI- All Members will present an "Important phase" of the work.

In this project assignment, you will submit a proposal *in which you will create* for your group project.

This proposal must include:

- **Proposed Group Members**
- **Project summary**
- **GitHub Repository link for the project**
- **Download the statement pdf file sign it and attach to your proposal submission**
- **Proposed the wireframe** - a visual representation, a detailed outline of the user interface (UI) and layout of an app.
  - **Layouts** - Number and name of unique screens in Project
  - **Functionality** - Proposed Use case from transitioning from screen to screen.
  - **User Experience** - Proposed Use cases for each Screen
  - **Iterative design** – Test cases and approach to refine the improvements.

A statement stating you understand that you will be graded individually on group assignments and may not receive the same grade as all members. All parties must play an important role as it relates to the implementation from start to finish. FYI- All Members will present an "Important phase" of the work.

When selecting a project idea, consider the scope and complexity of the app, the target audience, and the technologies you'll need to use. Ensure that your proposal outlines the app's purpose, features, and potential benefits to users.

### **Important Additional Content – Please review**

- [18-Day Mobile App Development Milestone Schedule](#)
- [Additional Requirements](#)
- [Special Project Instructions](#)
- [Flutter GitHub Setup Guide](#)
- [About Kanban](#)

**Project ideas: You MUST select a topic from the list below:**

**(You are not allowed to change and select your own)**

### 1. Local Events Finder

- **Event Listings:** Display a list of events based on the user's location.
- **Filters:** Allow users to filter events by type (concerts, sports, workshops), date, and venue.
- **Event Details:** Show details such as time, location, ticket info, and a brief description.
- **Favourites:** Enable users to save events they're interested in.
- **Local Storage:** Store bookmarked events locally for offline access.
- **Extended Functionality:** Add dark mode and search bar.
- **Bonus :** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort (Eg.,map integration using Google Maps API.)

#### Milestone 1:

- Create UI for home screen and event list.
- Implement event cards and filtering options (static data).
- Show navigation between event list and event details screen.

#### Milestone 2:

- Add local storage for favourites (using shared\_preferences or SQLite).
- Implement real-time data or location-based event display.
- Final testing and design refinement.

---

### 2. Fitness Tracker App

- **Workout Log:** Let users record exercises (e.g., running, weightlifting) with duration and reps.
- **Calorie Tracker:** Provide a simple way to log daily calorie intake.
- **Progress Tracking:** Display a summary of workouts over time using charts.
- **Preset Routines:** Offer basic workout plans (Beginner, Intermediate, Advanced).
- **Local Storage:** Save workout history locally for review.
- **Extended Functionality:** Add notifications or reminders for workouts.
- **Bonus :** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort.

#### Milestone 1:

- Build UI screens for workout log and calorie tracker.
- Implement add/remove workout functionality with temporary data.
- Display progress screen with static data chart.

#### Milestone 2:

- Integrate local database (SQLite).
- Enable saving and viewing progress history dynamically.

- Add reminder/notification feature.

### 3. Recipe and Meal Planning App

- **Recipe Database:** Display a list of simple recipes with ingredients and steps.
- **Dietary Filters:** Allow users to filter by vegetarian, vegan, gluten-free, etc.
- **Meal Planner:** Users can select meals for the week and generate a grocery list.
- **Favourites:** Save favourite recipes for quick access.
- **Local Storage:** Store meal plans and saved recipes for offline use.
- **Extended Functionality:** Add “Share Recipe” Option.
- **Bonus:** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort.(Eg., AI Suggest Recipe” option.)

#### Milestone 1:

- Create main UI: recipe list, filter section, and detail screen.
- Implement navigation between recipe list and details.

#### Milestone 2:

- Add meal planner feature , grocery list generation and share recipe option.
- Integrate local storage for saved recipes and plans.
- Final polish and optional recipe sharing.

---

### 4. Pet Care App (Bunny, Hamster, Dog, or Cat)

- **Feeding Schedule:** Let users log and track food and water schedules.
- **Care Tips:** Provide essential pet care guides based on selected pet type.
- **Illness Symptoms:** A simple guide to identify common pet illnesses.
- **Reminders:** Set reminders for feeding, vet visits, or grooming.
- **Local Storage:** Save pet details and schedules locally.
- **Extended Functionality:** Add pet profile picture and vaccination tracker.
- **Bonus:** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort.

#### Milestone 1:

- Build UI for pet dashboard and feeding schedule.
- Implement static care tips and symptom pages.

#### Milestone 2:

- Add reminder/notification functionality.
- Enable local data saving for feeding schedule.
- Add pet image upload and vaccination Tracker.

## 5. Mindfulness and Stress Reduction

- **Guided Exercises:** Provide simple breathing exercises or guided meditation audios.
- **Daily Affirmations:** Display a new motivational quote or positive message each day.
- **Mood Tracker:** Users can log how they feel daily and view trends.
- **Journal:** A simple space for users to write thoughts and reflections.
- **Local Storage:** Store journal entries and mood logs locally.
- **Extended Functionality:** Add soothing background music.
- **Bonus:** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort. (Eg., daily streak counter.)

### Milestone 1:

- UI for home, affirmation, and journal pages.
- Implement mood tracker with temporary storage.

### Milestone 2:

- Integrate local database for journal and mood tracking.
- Add quote rotation or sound integration.
- Final testing and optional animation/streak feature.

---

## 6. Personal Finance Manager

- **Income & Expense Tracker:** Allow users to log transactions.
- **Category Breakdown:** Categorize expenses (e.g., food, rent, entertainment).
- **Savings Goals:** Let users set goals and track progress.
- **Basic Reports:** Display simple charts of income vs. expenses.
- **Local Storage:** Save financial data locally for privacy.
- **Extended Functionality:** Add budget alerts.
- **Bonus:** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort (Eg., export to CSV.)

### Milestone 1:

- Create UI for transaction entry and category view.
- Show total balance and sample chart using static data.

### Milestone 2:

- Connect SQLite database for persistent storage.
  - Add reports, savings goals, and export option.
  - Refine layout and test all screens.
-

## 7. Campus Swap & Freecycle (Local-Only Marketplace)

- **Item Listings:** Allow users to post items available for swap or free pickup.
- **Filters:** Enable users to search by category, condition, and availability.
- **Swap Tokens:** Implement a token system to manage item exchanges.
- **Moderation:** Provide a way to flag inappropriate or duplicate listings.
- **Local Storage:** Save listings, filters, and user preferences locally for privacy.
- **Extended Functionality:** Add chat/messaging between users, image uploads for items.
- **Bonus:** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort.(Eg., location tagging.)

### Milestone 1:

- Design UI screens for item listing, item details, and search/filter functionality.
- Implement basic add, view, and delete item features with temporary data.
- Create navigation between main listing and details pages.

### Milestone 2:

- Integrate local storage for item listings and filters.
  - Implement basic swap token logic and moderation/flagging option.
  - Add features (like image upload or user chat).
  - Final testing and UI refinements.
- 

## 8. Vehicle Maintenance Tracker

- **Maintenance Log:** Record oil changes, tire rotations, and inspections.
- **Reminders:** Notify users of upcoming service dates.
- **Expense Tracker:** Log maintenance costs for budgeting.
- **Vehicle Info:** Save car details (make, model, mileage).
- **Local Storage:** Store logs and reminders locally.
- **Extended Functionality:** Enable multiple vehicle profiles for users managing more than one car.
- **Bonus:** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort.(Eg., AI service suggestion or export report.)

### Milestone 1:

- Build UI for maintenance log and vehicle info input.
- Implement add/edit/delete functionality with temporary data.
- Basic navigation between vehicle dashboard and log screens.

### Milestone 2:

- Integrate local storage for logs and reminders.
- Add notification system for upcoming maintenance.
- Add multiple vehicle profiles for users managing more than one car.

## 9. Smart Grocery List

- **Grocery Items:** Add, edit, and delete grocery items with quantity, notes, and categories.
- **Categories:** Organize groceries into sections (Produce, Dairy, Snacks, etc.) for easier navigation.
- **Checklist:** Mark items as purchased to track shopping progress.
- **Priority Tag:** Highlight important or urgent items (e.g., “Need Today”).
- **Search & Filter:** Quickly find or filter items by name, category, or priority.
- **Local Storage:** Save grocery lists offline using `shared_preferences` or `SQLite`.
- **Extended Functionality:** Add weekly list generator, price estimation.
- **Bonus:** If you add extra or innovative functionalities beyond the project description may receive bonus credit for creativity and effort. (Eg., barcode scanning for item entry.)

### Milestone 1:

- Design UI for adding and viewing grocery items (home screen and detail/edit screen).
- Implement add/delete/edit features using temporary (in-memory) data.
- Add category and priority selection fields.
- Include simple sorting or filtering by category or name.

### Milestone 2:

- Integrate `SQLite` or `shared_preferences` for saving grocery lists locally.
- Implement checklist functionality to mark items as purchased.
- Add features like weekly list generator or price calculator.
- Test all functionalities and ensure smooth navigation between screens.

Hint \*\*\* Application of local storage to save app data like custom flash cards, journal entries, and mood data locally on the device is required.

---

### Project submission details.

Along with the GitHub link, include the APK file and a team presentation. Add the presentation link in the README file. You can upload the presentation to YouTube and provide the link there. The presentation should explain the project and demo the app. Please see the second document about the additional submission requirements.