

Revised Project Plan

SipSip – Plant Care Mobile App

Primary Instructor	Anjana Shah
Team Member	Jayden Lewis
Team Member	Uzma Shaikh
Team Member	Aidan Repchik

Document Revision History

Revision #	Date
0.1	29/09/2025
0.2	01/10/2025
0.3	04/10/2025
1.0	07/10/2025
2.0	08/02/2026

Table of Contents

1. Executive Summary.....	3
2. Project Approvers, Reviews and Distribution List.....	4
3. Scope.....	4
4. Deliverables.....	5
5. Assumptions.....	6
6. Dependencies.....	7
7. Risk Management.....	8
8. Communication.....	9
8.1 Reporting.....	9
8.2 Meetings.....	11
9. Task Listing (WBS—Work Breakdown Structure).....	11
10. Gantt Chart.....	11
11. Milestones.....	12
12. RAM – Responsibility Assignment Matrix.....	12
13. Approval.....	13

1. Executive Summary

Objective	The SipSip project aims to design and develop an Android mobile application for individuals who take care of plants. The app's main features include plant growth tracking, personalized watering schedules, and automatic reminders. With the help of SipSip, users can significantly improve the health of their plants through tailored care tips and offline functionality. The objective is to deliver a functional minimum viable product (MVP) by the end of March.
Corporate Goals Addressed	• Encourage sustainable living and plant ownership. • Promote digital innovation and mobile app development skills within the team. • Demonstrate agile collaboration, user-centered design, and technical excellence. • Provide a scalable foundation for future integration with weather APIs and retail partnerships.
Planned Start Date	September 15, 2025
Planned End Date	March 27, 2026

2. Project Approvers, Reviews and Distribution List

Project Role	Name	E-mail	Date
Project Manager	Jayden Lewis	jayden.lewis@georgebrown.ca	07/10/2025
Project Engineer	Uzma Shaikh	uzma.Shaikh2@georgebrown.ca	07/10/2025
Project Developer	Aidan Repchik	aidan.repchik@georgebrown.ca	07/10/2025

3. Scope

In Scope	Out of Scope
Development of an Android application	Development of an IOS, web, or desktop version of SipSip
Core features: plant management, reminders, plant-specific tips, and a user-friendly interface	Advanced AI diagnostics that understand plant health patterns and can offer personalised advice through a chatbot
Offline functionality for reminders and core app features	Paid subscription model for advanced features
Local data storage and optional user sign-up	Integration with potential business partner products
Implementation of UI enhancements including the top action bar search feature, redesigned dashboard filter bar, reorganized Plant Collections layout, and updated notification filtering system.	Development of custom machine-learning-based plant identification or automated plant-health analytics.

4. Deliverables

Deliverable	Description
Minimum Viable Product (MVP)	A fully functional Android application that includes plant management, reminders, plant-specific care tips, a user-friendly interface, and the new Sprint 5 UI enhancements such as a top action bar with an expandable search input, dashboard filter bar, reorganized Plant Collections layout, and improved notification filtering with timestamps.
Project Vision Document	A comprehensive document outlining the project scope, objectives, stakeholder requirements, and system features
User Manual	A user guide explaining app navigation, plant setup, reminders, and all new UI features such as the action bar search, dashboard filters, collapsible room sections, contextual menus, and notification filter categories.
Test Report	A document summarizing all testing activities, test cases, results, defect logs, and verification of the new UI elements introduced in Sprint 5.

5. Assumptions

Assumption	Implication if Invalid
Team Member Availability	Loss of team members will place an unsustainable burden on remaining members, leading to missed deadlines and scope reduction.
Development Tool Stability	If Android Studio, Kotlin SDK, or Room Database becomes unstable or requires major version updates, the team will face significant unplanned time spent on environment troubleshooting.
Fixed Platform Scope (Android API)	Expanding the target API level or including other operating systems (like iOS) will immediately invalidate the current schedule and resource allocation.
Timely Instructor Feedback	Delayed feedback will stall the project's progress, especially during approval stages for Design and Requirements, potentially causing a missed April 2nd, 2026, end date.
Testing Environment Access	Lack of reliable test devices or emulators will prevent thorough quality assurance, leading to critical bugs in the final MVP release.
API Accessibility and Stability	If required, free APIs are decommissioned or significantly change their usage terms, the team will need to rapidly integrate a replacement, jeopardizing the schedule.
Agile/Scrum Practices	Deviating from the sprint cycle will disrupt the overall project rhythm and make accurate progress tracking and risk mitigation difficult.

6. Dependencies

Dependency	Type	Preceding Task/Resource	Impact on Success
System Architecture and Data Models	Internal (Technical)	System Requirements & Design Portfolio (from Sprint 3)	Core Features Development (Coding) cannot begin without the approved database structure and system design.
Approved Wireframes / Prototype	Internal (Design)	Wireframes / Prototype (from Sprint 4)	Core Features Development cannot finalize the user interface (UI) and user experience (UX) without approved design mock-ups.
Sprint 5 UI Enhancement Designs	Internal (Design)	Approved UI wireframes and updated component layouts	Incorrect or unapproved UI designs will delay development, user manual updates, and UI-based testing.
Plant Care Data API/Databases	External (Resource)	Availability of chosen API/Data Source	Core Features Development and Testing & Bug Fixing depend entirely on having reliable data to power the plant tips and scheduling logic.

Dependency	Type	Preceding Task/Resource	Impact on Success
Google Play Developer Console Access	External (Resource)	Successful registration and payment for the Google Developer Account.	The Deployment to Google Play task is impossible without this access.
Completed Requirement Analysis	Internal (Project)	Completion of Requirement Analysis (Oct 13, 2025)	All subsequent phases (UX/UI Design, Core Features Dev) depend on having a finalized and approved list of requirements.

7. Risk Management

Potential Risk	Severity (H/M/L)	Likelihood (H/M/L)	Management Strategy
Delay in development timeline	H	M	Regular progress reviews and agile sprints
Incompatible Android versions	M	M	Test on multiple Android versions
Weather API failure	M	L	Implement fallback to manual input
Insufficient user adoption	M	M	Marketing and user feedback integration
UI inconsistency across different	M	M	Conduct responsive design testing on

screen sizes and resolutions			multiple Android screens and emulators.
Search & filtering performance degradation	M	L	Optimize queries, test for performance under typical user load.
Increased UI complexity confusing users	L	L	Follow usability best practices and gather early feedback.
Team member unavailability	H	L	Cross-train team members

8. Communication

8.1 Reporting

Report	Audience	Purpose/Description	Due Date
Project Vision Document	Instructor (Anjana Shah), Team Members	Defines the project's overall vision, goals, and value proposition. Establishes a shared understanding of the app's purpose.	Sprint 1
High-Level Business Requirements Document	Instructor, Team	Captures the business and functional requirements forming the basis for design and development.	Sprint 1

Report	Audience	Purpose/Description	Due Date
User Personas Document	Instructor, Team	Represents different user types and needs to guide user-centered design.	Sprint 1
User Stories Document	Instructor, Team	Details user-centered stories to define app functionality and acceptance criteria.	Sprint 1
Team Charter	Instructor, Team	Defines team structure, roles, responsibilities, and communication rules.	Sprint 2
Project Plan	Instructor, Team	Outlines project scope, schedule, resources, risks, and communication plan.	Sprint 2
Product Backlog	Instructor, Team	Lists and prioritizes all app features and tasks aligned with user stories.	Sprint 2
Sprint Backlog	Instructor, Team	Defines specific sprint goals and tasks to be completed during each sprint.	Sprint 2
System Requirements & Design Portfolio	Instructor, Team	Includes analysis and design deliverables such as system architecture and data models.	Sprint 3
Wireframes/Prototype	Instructor, Team	Visual representation of the user interface to validate user flow and design decisions.	Sprint 4
Technical Requirements Document	Instructor, Team	Describes hardware, software, and API dependencies for implementation.	Sprint 4

Report	Audience	Purpose/Description	Due Date
Revised Project Plan	Instructor, Team	Project roadmap including updated scope, deliverables, risk management strategies, and Completed UI enhancements.	Sprint 5
Project Status Report I	Instructor, Team	Summary documenting completed UI improvements, upcoming objectives, task assignments, timeline forecasts, identified risks, and mitigation strategies.	Sprint 5
System Implementation I	Instructor, Team	Documentation covering architecture deployment, feature integration processes, database configuration, testing procedures, and system setup specifications.	Sprint 6
Presentation I	Instructor, Class	Summarizes Sprint 1 & 2 deliverables, team collaboration, and project direction.	Midterm (Semester 5)
Presentation II	Instructor, Class	Demonstrates Sprint 3 & 4 outcomes, including prototype, testing, and design integration.	End of Semester 5
Presentation III	Instructor, Class	Demonstrates Sprint 5 & 6 outcomes, including prototype, testing, design integration and System Implementation of code.	Midterm (Semester 6)

Report	Audience	Purpose/Description	Due Date
Project Progress Reports / Minutes of Meeting	Instructor, Team	Track progress, highlight blockers, and document weekly discussions.	Weekly
Peer Evaluation	Instructor	Evaluates each team member's contribution and collaboration.	Completed (Semester 5)
Peer Evaluation	Instructor	Evaluates each team member's contribution and collaboration.	Ongoing

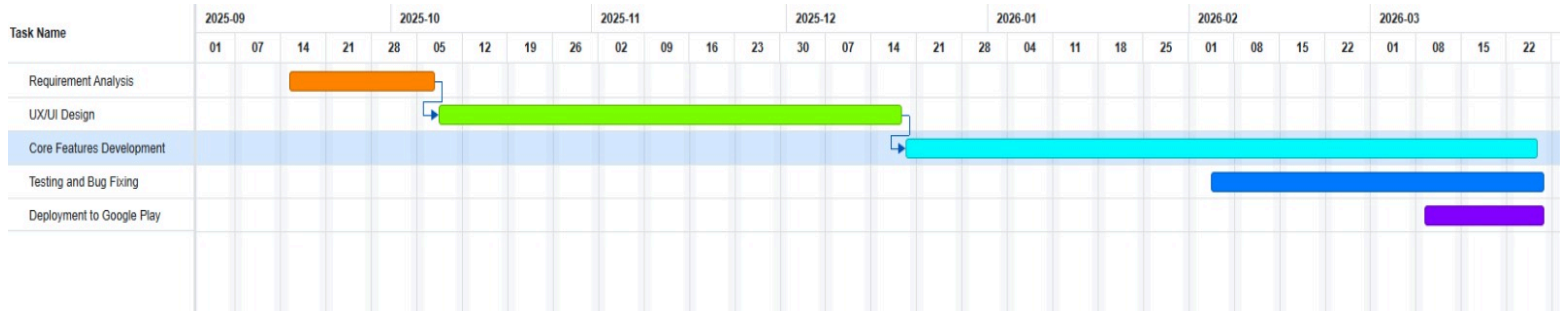
8.2 Meetings

Meeting	Purpose	Attendees	Frequency
Sprint	Define tasks and goals for each sprint	Project Team + Instructor	Bi-weekly
Weekly Class (Team Sync)	Track weekly progress and assign next tasks	Entire Project Team	Weekly
Instructor Check-In	Seek feedback and approval	Project Team + instructor	Weekly

9. Task Listing (WBS—Work Breakdown Structure)

Reference	Tasks	Duration	Dependency
A	Requirement Analysis	4 Week	None
B	UX/UI Design	7 Week	A
C	Core Features Development	14 Week	B
D	Testing and Bug Fixing	4 Week	C
E	Deployment to Google Play	2 Week	D

10. Gantt Chart



11. Milestones

Major Activity or Milestone	Estimated Milestone Target date	Owner/Reviewer Team Members
Requirement Analysis	October 17, 2025	Jayden Lewis
UX/UI Design	November 9, 2025	Uzma Shaikh
Core Features Development	February 20, 2026	Aidan Repchik
Testing and Bug Fixing	Mid-March 2026 (Approx. Mar 14)	Uzma Shaikh
Deployment to Google Play	March 27, 2026	Jayden Lewis

12. RAM – Responsibility Assignment Matrix

Task	Requirement Analysis	UX/UI Design	Core Feature Development	Testing and Bug Fixing	Deployment to Google Play
Jayden Lewis	P	S	S	S	P

Uzma Shaikh	S	P	S	S	S
Aidan Repchik	S	S	P	P	S

Legend:
P = Primary
S = Secondary

13. Approval

Project Role	Name	Signature	Date
Project Manager	Jayden Lewis	Jayden Lewis	08/02/2026
Project Engineer	Uzma Shaikh	Uzma Shaikh	08/02/2026
Project Developer	Aidan Repchik	Aidan Repchik	08/02/2026