

**FIT2002 - IT PROJECT MANAGEMENT**

**PROJECT TEAM 605**

**ASSIGNMENT 1**

Team Members:

Muthu Wijesekera (34088822)

Amash Rathnayake (33904464)

Ashwin Patil (35003804)

Manumi Ekanayake (35232617)

# Team Charter

<b><i>FIT2002 – Working Agreement (Team Charter)</i></b>	
<b><i>Team number</i></b>	605
<b><i>Team members</i></b>	<i>Amasha Rathnayake</i> <i>Ashwin Patil</i> <i>Manumi Ekanayake</i> <i>Muthu Wijesekera</i>
<b><i>Team objectives</i></b>	We are trying to achieve a well planned and documented IT project that aligns with the client's business needs as part of this assignment.
<b><i>Team characteristics</i></b>	Our team is composed of members that have a diverse range of skillsets. Some members are strong in analytical and technical areas (including statistics), while others excel at communication, organisation and documentation. As a group, we balance organisation, creativity, problem-solving and attention to detail. We will leverage one member's strengths while supporting each other in improving other areas.
<b><i>Core values</i></b>	We care the most about delivering high-quality, client-focused work. Common values we share include respect for each member's opinions and input, accountability for individual tasks and meeting deadlines. We also believe that integrity and transparency in communication is vital for a successful project.
<b><i>Group norms and code of conduct</i></b>	We will work together by attending scheduled meetings (in-class and online) and come prepared to them. We will complete our assigned tasks on time and to the best possible standard. We also expect team members to communicate any issues or delays early and be respectful and constructive when giving feedback. We will also make decisions collectively so that all our opinions are heard and considered.
<b><i>Participation and collaboration approach</i></b>	We will have weekly check-ins during our applied classes and also hold additional meetings to assign tasks and discuss the progression of our tasks. We will update the Task Tracker regularly to track tasks, deadlines and progress. Active contribution is also expected from every member and the workload will be divided equally.

<b>Communications</b>	We will primarily use an Instagram group chat for communication. Formal updates and documentation will be shared via the Task Tracker, Workbook and Google Drive. We will update each other 2-3 times per week, but could be more frequent closer to deadlines.
<b>Problem solving</b>	Problems will be approached collaboratively and we will discuss potential solutions together. For project-related risks, we will apply PMBOK risk management principles. For team-related issues, we will communicate as soon as possible and redistribute tasks fairly depending on the situation. We will also update the Task Tracker to track issues and solutions.
<b>Conflict management</b>	Conflict will be addressed with respectful group discussions with a focus on the issue at hand. We will listen to all the perspectives of the members in the group before making decisions. A majority vote will also be used if a consensus cannot be reached through discussion. If the conflict cannot be resolved within the group, the matter will be escalated to the tutor.
<b>Signatures</b>	Ashwin Patil Amasha Rathnayake Manumi Ekanayake Muthu Wijesekera

## Project Charter

### Project Title and Description

*Project title:* LoopCare: A Digital Wellness Companion

*Description:* Design and deliver a mobile and web application helps users track emotional well-being, sleep, hydration and screen-time. The application must have an accessibility first UX and include daily community challenges for users. The project aims to promote wellness in users through unique, tailored insights and experiences, increasing engagement as a result.

## Project Scope and Objectives

The team will deliver an app and website. There will be an app that can be downloaded onto a smart watch that will help to track sleep, wellbeing, mood and other health indicators. The data will sync and be accessible through the app and website. Personal insights will be available for users and gamified daily challenges also included to promote well-being. Analytics will also be displayed on the dashboard. The outcomes expected are for users to gain clear visibility of their habits and trends and are encouraged towards healthier daily routines. The deliverables will be secure and accessible for users. Smart objectives aimed to be achieved include delivering a beta release by May 2026 with the health trackers, insights and challenges available on the mobile, watch and web through TestFlight and Samsung Developer. Achieving  $\geq 80\%$  UAT satisfaction for usability, performance and functionality by the end of UAT. Meet non-functional targets including 99% crash-free sessions and meeting WCAG 2.1 AA guidelines at release. Finally, reaching  $\geq 60\%$  weekly active rate with pilot users a month after the launch of the pilot.

## Project Start and Finish Dates

The intended duration of the project is 12.5 months. This duration starts from September 2025 to September 2026 as seen in the Gantt chart.

## High-Level Budget Estimate

The total estimated cost of the project is \$285,000 AUD. This is within LifeLoop's \$300,000 allocation for the project. The budget allows for a \$15,000 buffer. The buffer allows for some flexibility. However, changes must be approved and are dependent on the situation.

## Project Development Approach

We will be using a Hybrid approach for the development approach. This approach allows us to maintain the structure of Waterfall while also providing leeway for teams to adjust to evolving project needs and changing requirements. The method of delivery will be prioritising outcomes over outputs. Stakeholder demos will be performed fortnightly to maintain engagement and risk responses will be implemented including budget buffers and change control. Adaptive delivery will also be prioritised with the ability to pivot, such as changing scope for low-value features.

## Key Stakeholders

Stakeholder	Role	Interest/Influence
LifeLoop	Project Sponsor	Budget approvals, decide whether to release or not
Product Owner	Scope authority/ team contact, product backlog	Accepts increments, prioritises features and requirements
Project Manager	Project manager	Cost, scope, risk, delivery and communication
End Users	Users	Vested interest in a product meeting their needs/ requirements

## Project Success Criteria

The success of the project will be dependent on the time (released the initial version within 15 months) and within the \$285,000 total spend. Quality and performance of the app will be measured through conducting comprehensive testing for bugs and crashes (crash-free sessions  $\geq$  99%, severe defects resolved within 48 hours during

Beta testing). For user adoption,  $\geq$  3,000 monthly active users by the 3 month mark as well as app store rating above  $\geq$  4.2/5 in the app store. Stakeholder satisfaction should be  $\geq$  85% in the post-launch survey.

## Assumptions and Exclusions

### *Assumptions:*

Stakeholders will be available for feedback and approval within the previously agreed upon timelines. Users will have access to smartphones and/or web browsers capable of running the application. Data storage, analytics and user insights will rely on existing cloud services. Community challenges will operate on a daily cycle with automated resets and leaderboard updates to increase engagement.

### *Exclusions:*

Development of a dedicated wearable device is not included in the project. No third-party integrations, with other brands such as FitBit and Garmin, in the project. Offline functionality, while included in the wearable, is not available for the use of the app to sync data. The app is for wellness tracking purposes only, not medical treatment and will not include clinical or diagnostic functionality. Community challenges will be managed centrally and cannot be customised by the users. No real time messaging or chat platform. Community challenges will be interactive but not social networking.

## The Pitch

### 1. Opening Hook (10 Seconds)

We obsess over steps and calories. But sleep? Hydration? Mood? And most importantly screen time? The stuff that actually shapes our days, slips by. What if wellness were as trackable and actionable as fitness?

## **2. Problem/Opportunity (20 Seconds)**

What if wellness were as trackable and actionable as fitness?

## **3. Your Proposed Solution (30 Seconds)**

That's where LoopCare comes in. LoopCare is your new digital wellness companion. Designed to make healthy living simple, accessible, and achievable for anyone. By combining simplicity with gamified engagement, LoopCare transforms wellness into an enjoyable daily habit.

## **4. Strategic Fit (20 Seconds)**

For partners and stakeholders, LoopCare offers valuable data-driven insights into user needs, positioning us to lead the digital healthcare market with an engaging and impactful platform.

## **5. Key Features/Deliverables (20 Seconds)**

With LoopCare, users can reduce stress, build resilience, and gain personalised insights to improve their lifestyle. They can join community challenges to stay motivated, while tracking emotional well-being, sleep, hydration, and screen time all in one place. And because prevention is key, LoopCare helps detect negative health trends early, before they escalate. The project duration would take between 12 to 14 months approximately. In terms of budget, we are looking at capping finances at \$285,000.

## **6. Why Your Project Should Be Selected (20 Seconds)**

LoopCare is more than an app. It's a step toward a healthier, more balanced future. Together, let's make digital wellness simple, engaging, and accessible for everyone. Let's make it happen!

# Requirements Traceability Matrix (RTM)

<b>Project Name:</b>	LoopCare: A Digital Wellness Companion				
<b>Project Manager Name:</b>					
<b>Project Description:</b>	Design and deliver a mobile and web application helps users track emotional well-being, sleep, hydration and screen-time. The application must have an accessibility first UX and include daily community challenges for users. The project aims to promote wellness in users through unique, tailored insights and experiences, increasing engagement as a result.				
<b>ID</b>	<b>Requirements (Functional or Non-Functional)</b>	<b>Assumption(s) and/or Customer Need(s)</b>	<b>Category</b>	<b>Source</b>	<b>Status</b>
R1	Load time under 3 seconds	Tested on local server	Non-Functional	Tutor	Open
R2	The app should provide gamified challenges (e.g., 7-day hydration challenge) with badges and leaderboards	Community engagement features increase motivation	Functional	Pitch brief	Open
R3	App must load within 3 seconds on a 4G connection	Tested on standard Android/iOS devices	Non-functional	Tutor	Draft
R4	Data privacy and Australian Privacy (APPs) and GDPR	Cloud storage and analytics vendor meet compliance	Non-functional	Industry standards	Open
R5	Accessibility must meet WCAG 2.1 AA	Designed for inclusivity (colour-blind, screen readers)	Non-functional	ISO/Accessibility standard	Planned

# Project Scope Statement

## Section 1: Scope Management

### 1.1 Project Scope Statement

Purpose: The aim of the LoopCare: Digital Wellness Companion project is to develop and implement a unique mobile app which would respond directly to an escalating health crisis due to stress, lack of sleep, excessive screen time, and waning mental health. Nowadays people live in a rapid and technological world, with no time to focus on basic needs, such as hydration, sleep, emotional care. LoopCare is trying to bridge this urgent void by having smartphones become proactive contributors to well-being rather than distractors. LoopCare supports users in making healthier behaviours, maintaining such patterns, and ensuring long-term changes in wellbeing by combining personalisation, accessibility, and gamified engagement.

#### *Objectives:*

The project aims at:

1. Encourage Preventive Health - Empower early identification of unhealthy behavioural habits (e.g. sleep disorders, excessive screen time, mood swings) and give practical advice to minimise health risks in the long term.
2. Presentation of Personalised Insights - Guide users to small, measurable behaviour changes that can collectively increase emotional resilience, sleep quality, hydration and digital balance through the use of data-driven analytics and AI-generated nudges.
3. Promote Community and Motivation- develop a socially interactive setting where users have the ability to partake in challenges, progress updates and form accountability via peer support.
4. Be Accessible and Inclusive - Design the app by following the WCAG 2.1 AA guidelines to ensure that individuals with all abilities have an easy time using the application.

5. Protect User Trust - Assure that data privacy and security are upheld by following the Australian Privacy Act (1988) and using industry-established encryption protocols that make sure that sensitive health data is secure.
6. Attain Sustainable Adoption - Provide an engaging, gamified user interface to ensure at least 70 percent of active monthly retention in the first year of post-launch operation.

## 1.2 Deliverables & Acceptance Criteria

<b>Project Name:</b>	LoopCare: A Digital Wellness Companion			
<b>Project Manager Name:</b>				
<b>Project Description:</b>	Design and deliver a mobile and web application helps users track emotional well-being, sleep, hydration and screen-time. The application must have an accessibility first UX and include daily community challenges for users. The project aims to promote wellness in users through unique, tailored insights and experiences, increasing engagement as a result.			
<i>Deliverable</i>	<i>Functional Requirements</i>	<i>Acceptance Criteria</i>	<i>Non-Functional Requirements</i>	<i>Acceptance Criteria / Quality Metrics</i>
<b>D1: Mood Tracking Module</b>	(i) Users can log daily mood (ii) Historical mood graphs available	Logs save successfully; graphs display trends	Reliability (99% uptime)	Logs accessible 24/7
<b>D2: Sleep &amp; Hydration Tracker</b>	(i) Users can input hours slept (ii) Users can record daily water intake	Data recorded accurately; summaries visible	Usability ( $\leq 3$ taps to record)	User test shows $>85\%$ ease-of-use
<b>D3: Screen Time Limiter</b>	(i) Detects app usage duration (ii) Sends alerts when preset limit exceeded	Alert triggers at correct time	Performance (<2 sec response time)	Stress-tested to 10k users
<b>D4: Community Challenge Platform</b>	(i) Join challenges (ii) View leaderboard	Users join/leave without errors	Accessibility (WCAG 2.1 AA)	Pass accessibility audit
<b>D5: Personalised Nudges &amp; Insights</b>	(i) AI-based nudges (ii) Weekly insights dashboard	Nudges delivered on schedule; insights generated	Security (AES-256 encryption)	Audit verifies encryption compliance

### 1.3 Exclusions

In order to keep the project focused and deliver it on time and within the budget the following things are not included:

- Wearable Device Integration – The first launch would not support third-party wearables like Fitbit or Garmin.
- Web/Desktop Platforms Loopcare will only be created in iOS and Android mobile platforms. Desktop or browser versions are not in scope.
- High-level Clinical or Diagnostic Characteristics - LoopCare will not perform medical diagnosis, treatment, or professional counselling. It is adhered to as a lifestyle and wellness support product rather than a regulated healthcare product.
- Insurance/Corporate Partnerships - Although there is an opportunity to consider long-term partnerships, insurer or corporate wellness integrations will not be part of the MVP.
- Offline Features (Other than basic) - Unless logging hydration or mood offline (with syncing on reconnect), all offline functionality (including community challenges) will not work without an internet connection.

### 1.4 Constraints & Assumptions

Constraints:

The following characteristics are the major limitations of the LoopCare project:

- Budgetary Constraint 100 Total expenditure should not be more than AUD \$300,000 whch will cover the design, development, testing and launch. This limits the use of advanced AI/ML features in the MVP.

- Timeframe Constraint - The project should be completed in 11 months or less and that encompasses planning, development, testing and deployment. This restricts the capabilities of Phase 1.
- Regulatory Constraint – The Australian Privacy Act (1988) and principles that are similar to the GDPR require that sensitive wellness data be handled in accordance with those principles.
- Technology Constraint The application will need a cross-platform (iOS/Android) deployment, based on the use of a common technology stack (e.g., Flutter or React Native) to manage the cost and maintainability.
- Resource Constraint The project team size is hardened by the size of the allocation of LifeLoop, restricting specialised functions (e.g., data scientists) during initial stages.

### 1.5 Assumptions:

The planning of the project takes into account the following assumptions that need to be checked at the initial stages of initiation and early development:

- User Access Assumption- The target users will have access to modern smartphones ( iOS v14 / Android v11) with steady internet connectivity to use the community and cloud-based functionality of LoopCare.
- Assumption of Stakeholder Engagement - The stakeholders of LifeLoop will give immediate feedback when reviewing the sprints thus facilitating an iterative and agile development process.
- Cloud Infrastructure Assumption- There will be access to reliable cloud services (e.g., AWS, Azure, or Google Cloud) to host the product, and the scalability will be ensured to 10,000 simultaneous users at the initial launch.

- Adoption Assumption- The market demand is adequate and the users would be encouraged to use digital wellness solutions, which will justify the business viability of the project.
- Data Integrity Assumption Users will enter self-reported data (e.g., mood, hydration) at least once, and accurately, to draw valuable insights, so that it will be possible to rely on third-party integrations

## 1.6 Annotated Work Breakdown Structure (WBS) & Gantt Chart

### **Work Breakdown Structure (WBS)**

The project of LoopCare was broken down into a hierarchical WBS to Level 3-4 so as to have clear visibility of deliverables, responsibilities and schedules. The high level phases (Initiation, Planning, Execution, Testing, Deployment, and Closure) are subdivided into major tasks which again, are further subdivided into sub-tasks. This makes all activities measurable, manageable and in line with the project goals.

### **Gantt Chart**

The Gantt chart gives a time frame of all the activities, dependencies, sequencing and milestones over the 11-month plan (Sep 2025 -Sep 2026). It indicates important path activities which include:

- Charter approval (M1)
- Requirements and scope sign-off (M2)
- Design completion (M3)
- Development finalisation (M4)
- Beta testing (M5)
- Launch (M6)
- Closure (M7)

Task dependencies are well mapped (e.g., requirements approval should come first before design, backend development should come first before system integration and QA should come first before deployment).

## **Annotations & Time Justifications**

Time assignments will be made on the basis of the complexity of the task, interdependencies and best practice in what the industry does in software projects.

- Initiation(14 days): time to draft, review and approve charter and prepare/deliver pitch. This keeps them in line with the stakeholders prior to planning.
- Planning (18 days): The requirements gathering, RTM, scope and WBS are assigned the time of about 3 weeks of time so that they can be complete and minimize downstream risks.
- Implementation (85 days / approximately 16 weeks): The longest stage, including UI/UX, back-end modules (authentication, wellness dashboards, APIs), front-end (iOS, Android, web, wearables) and system integration. The time indicates complexity and the necessity to perform blooms on various platforms.
- Testing (82 days / 16 weeks): Prolonged to support extensive QA, such as UAT, accessibility and security, and iterative bug fixes. This avoids serious post-release problems.
- Deployment (40 days / c. 8 weeks): Prepared in preparation and production deployment in order to address risks in environment setup, data migration and app store submissions.
- Closeout (40 days / (approximately) 8 weeks): Long to facilitate knowledge transfer, training, reporting and post-deployment support (warranty bug fixes). This makes the lifecycle a profession, and makes it sustainable once it has been delivered.

The combination of these phases enables the balance of realism and feasibility, which guarantees that the project can be implemented in 11 months (Sep 2025 -Sep 2026) and comply with the expectations of the stakeholders and requirements.

# Work Breakdown Structure (WBS)

## 1. Initiation: (14 days)

1.1 - Project charter: Drafting charter document (Milestone: Charter approved)

- 8 days

    1.1.1 Draft charter - 3 days

    1.1.2 Internal review and updates - 3 days

    1.1.3 Sponsor approval and sign-offs - 2 days

    M1: Charter approved

1.2 - The pitch: Deliver Pitch script - 6 days

    1.2.1 Prepare script - 3 days

    1.2.2 Rehearse and timing check - 2 days

    1.2.3 Deliver pitch - 1 day

## 2. Planning: 18 days

2.1 - Requirements gathering - 15 days

    2.1.1 Gather requirements - 5 days

    2.1.2 Draft functional requirements (mood, sleep, hydration, screen-time) - 5 days

    2.1.3 Draft non-functional requirements - 5 days

    2.1.4 Build RTM - 3 Days

    2.1.5 Approval of requirements - 2 Days

2.2 - Scope and WBS - 8 days

    2.2.1 Write scope - 5 days

    2.2.2 Build WBS structure - 3 Days

    M2: Requirements and scope approved

2.3 Cost and budget - 4 days

    2.3.1 Labour estimates - 2 days

    2.3.2 Materials and services estimates - 2 days

2.4 Risk and quality plan - 4 Days

    2.4.1 Draft risk plan - 2 Days

    2.4.2 Draft quality plan - 2 Days

### **3. Execution: - 85 days**

#### 3.1 UI/UX design - 25 days

3.1.1 Wireframes & prototypes - 15 days

3.1.2 Accessibility review (WCAG 2.1 AA) - 10 days

M3: Design completion

#### 3.2 Backend development - 30 days

3.2.1 Authentication module(SSO/email) - 10 days

3.2.2 Wellness tracking dashboard - 10 days

3.2.3 Mood, sleep, hydration endpoints (core tracking APIs) - 15 days

3.2.4 Screen-time input- 7 days

3.2.5 Gamified Challenge system - 8 days

#### 3.3 Frontend development - 45 days

3.3.1 Mobile app interface (iOS/Android) - 20 days

3.3.2 Web app interface - 15 days

3.3.3 WatchOS app interface (iOS/ Andriod) - 10 days

3.3.4 Watch connectivity (phone and watch sync) - 5 days

3.3.5 Wearable integration (Apple Watch + Android Wear) - 15 days

#### 3.4 System integration - 15 days

3.4.1 Merge backend and frontend - 15 days

M4: Development finalised

### **4. Testing - 82 days**

#### 4.1 Test planning - 5 days

4.1.1 Write functional test cases - 5 days

4.1.2 Write non-functional and compliance test cases - 5 days

#### 4.2 Test execution - 40 days

4.2.1 System integration testing - 10 days

4.2.2 User acceptance testing (UAT) - 10 days

4.2.3 Accessibility testing - 5 days

4.2.4 Security and privacy testing - 5 days

4.2.5 Wearable field testing (Apple + Android watches) - 10 days

#### 4.3 Bug fixing & re-testing - 37 days

4.3.1 Log issues in tracker - 5 days

4.3.2 Implementation of bug fixes - 15 days

- 4.3.3 Re-test after fixes - 10 days
  - 4.3.4 TestFlight beta (iOS and Android) - 7 days
- M5: Beta version tested

## **5. Deployment - 40 days**

- 5.1 Deployment preparation - 15 days
    - 5.1.1 Release deployment plan - 10 days
    - 5.1.2 Readiness review - 5 days
  - 5.2 Production Launch - 25 days
    - 5.2.1 Production environment setup - 5 days
    - 5.2.2 Data migration - 15 days
    - 5.2.3 Website submission - 5 days
    - 5.2.4 Wearable store submission (Apple App Store + Google Play Wear) - 5 days
- M6: Launching Website & APP

## **6. Closure - 40 days**

- 6.1 Reporting - 10 days
    - 6.1.1 Final project report - 5 days
    - 6.1.2 Deliver documentation package - 5 days
  - 6.2 Training & Knowledge Transfer - 10 days
    - 6.2.1 Training and knowledge transfer workshops - 5 days
    - 6.2.2 Internal team handover completed - 5 days
  - 6.3 Post-Deployment Support - 20 days
    - 6.3.1 Monitor bug reports - 10 days
    - 6.3.2 Provide fixes during warranty period - 10 days
- M7: Closure of the project

## **Time justifications:**

### **Initiation - 14 days**

The initiation phase has been allocated 14 days to establish a solid foundation before planning begins. The schedule also includes preparation, rehearsal, and delivery of the pitch. Two weeks allows us to confirm objectives, clarify scope and bring stakeholders into alignment to prevent any future misunderstandings.

### **Planning - 18 days**

The planning phase runs for 18 days, which allows us to capture, refine, and validate all requirements. The focus here is on turning ideas into something concrete, with enough time to review and validate decisions with stakeholders. The timeline is deliberately compact to keep momentum, but not so short that quality is compromised. By the end of planning, the team will be ready to move into execution with a clear direction.

### **Execution - 85 days**

Eighty-five days are allocated to execution, reflecting the scale and complexity of design and development tasks. This includes UI/UX design, backend and frontend coding, and integration work. Several modules such as authentication, dashboards, tracking features, and gamified challenges need to be developed and tested in parallel, which requires careful coordination. The extended timeline ensures the team can build iteratively, address technical challenges, and involve stakeholders in reviews, resulting in a working product that is ready for full-scale testing.

### **Testing and QA - 82 days**

Testing has been given 82 days to guarantee that the system is reliable, secure, and meets compliance standards. The schedule covers system integration, user acceptance, accessibility, and security testing, as well as wearable field trials. Time has also been reserved for bug fixing and re-testing, since defects are often uncovered during integration. This extended period reflects the importance of quality assurance: without thorough testing, the risk of deployment failure would be unacceptably high.

## **Deployment - 40 days**

The deployment phase runs for 40 days to support a smooth transition from testing into production. Activities include preparing the release plan, reviewing readiness, migrating data, and submitting apps to the Apple and Google stores. Time has been built in to account for approval delays and final readiness checks. Allowing just over a month for deployment avoids the risks associated with rushing this stage and ensures that the launch process is accurate, stable, and dependable.

## **Closure - 40 days**

Closure has been allocated 40 days to wrap up the project in a professional and sustainable way. This includes preparing reports, delivering documentation, providing training, and supporting the system in its initial post-launch period. The schedule ensures there is time to transfer knowledge, monitor early bug reports, and provide fixes during the warranty period. By dedicating this time, the team not only hands over a system but also ensures continuity, user confidence, and a smooth transition to business-as-usual operations.

## **Key Milestones:**

### **M1: Charter Approved – Initiation**

*Specific:* Final charter signed off, including objectives, scope, and constraints.

*Measurable:* Documented approval by LifeLoop sponsor.

*Achievable:* Based on early draft reviews and structured feedback cycles.

*Relevant:* Provides official mandate to proceed into planning.

*Time-bound:* Completed by 18 Sep 2025.

### **M2: Requirements and Scope Approved – Planning**

*Specific:* Approval of RTM, scope statement, and WBS baseline.

*Measurable:* Signed-off documentation in ProjectLibre and workbook.

*Achievable:* Derived from structured stakeholder consultations.

*Relevant:* Locks scope and dependencies before development.

*Time-bound:* By 06 Oct 2025.

### **M3: Design Completed – Execution**

*Specific:* All UX deliverables (wireframes, prototypes, accessibility checks) finalised.

*Measurable:* Stakeholder sign-off on designs as “build-ready”.

*Achievable:* Fits within an allocated 3-week sprint.

*Relevant:* Provides the blueprint for developers to implement.

*Time-bound:* By 13 Nov 2025.

### **M4: Development finalised – Execution**

*Specific:* Backend + frontend integration complete; system ready for testing.

*Measurable:* Verified completion of core features and modules.

*Achievable:* Based on structured sprint planning and buffers.

*Relevant:* Ensures system readiness for full-scale testing.

*Time-bound:* By 05 Feb 2026.

### **M5: Beta Version Tested – Testing**

*Specific:* Beta release tested for core functions (tracking, dashboard, community challenge).

*Measurable:* UAT feedback completed; ≥ 90% defects resolved.

*Achievable:* Within structured test cycles and defect management.

*Relevant:* Validates features before launch.

*Time-bound:* By 01 Jun 2026.

### **M6: Launching Website and APP – Deployment**

*Specific:* App released on App Store and Google Play after readiness review.

*Measurable:* Successful submission and stakeholder sign-off.

*Achievable:* Includes buffer for approval delays.

*Relevant:* Core deliverable of the project.

*Time-bound:* By 27 Jul 2026.

### **M7: Closure of the project – Closure**

*Specific:* Documentation, training, lessons learned, and post-deployment support completed.

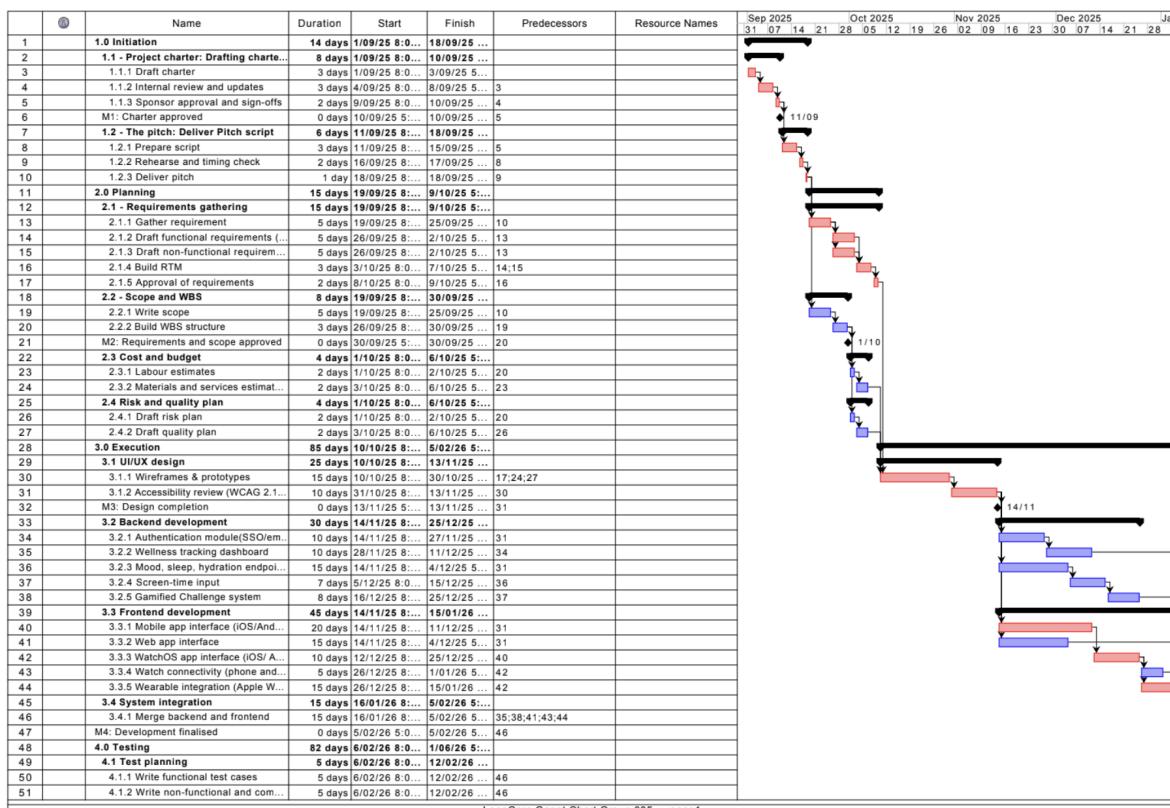
**Measurable:** Closure report approved by stakeholders.

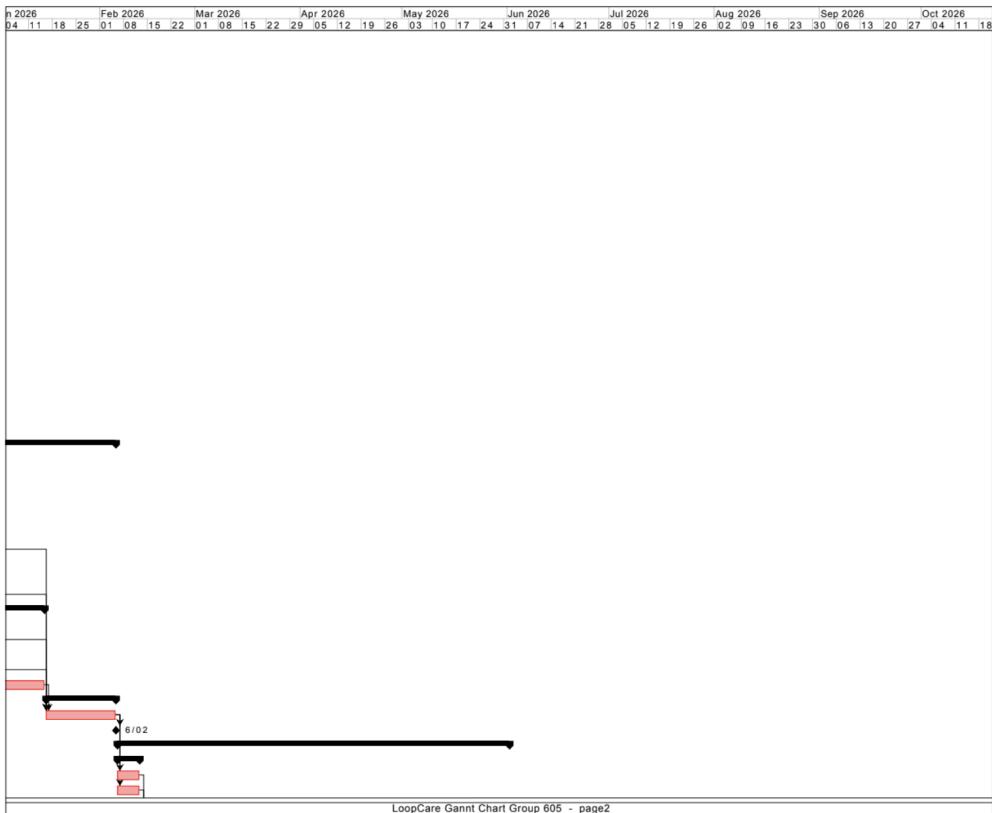
**Achievable:** Scheduled over 8 weeks with stabilisation support.

**Relevant:** Ensures sustainable adoption and knowledge transfer.

**Time-bound:** By 21 Sep 2026.

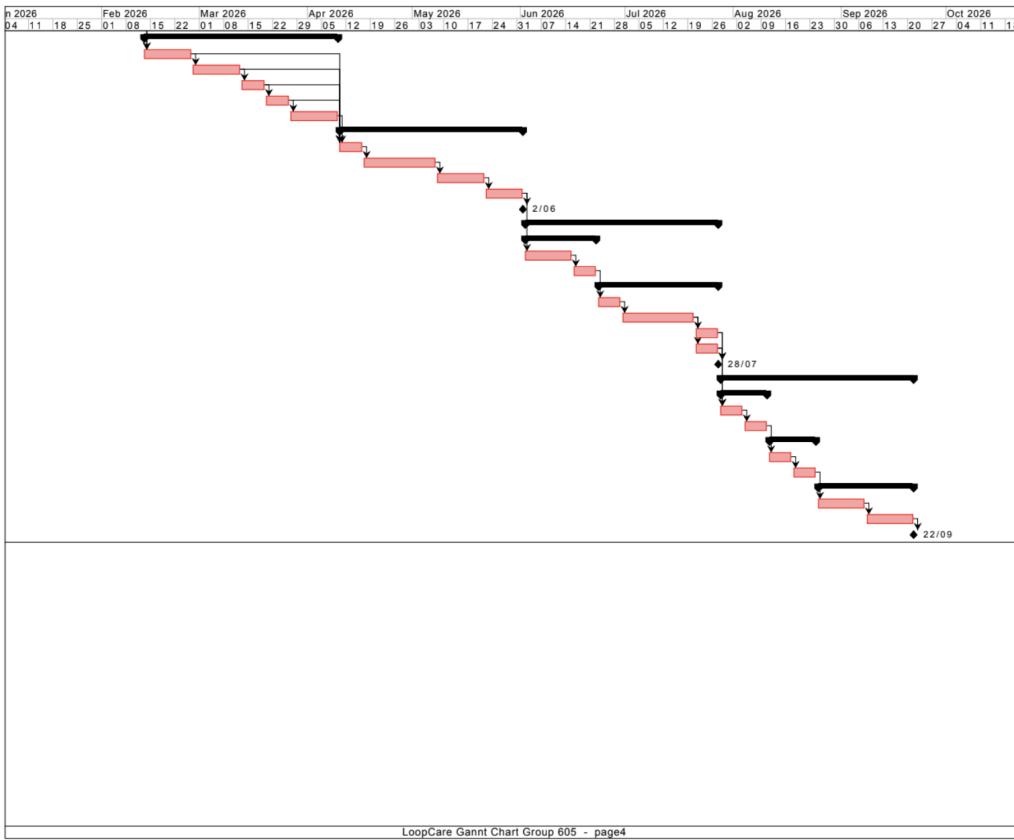
## Proposed Schedule (Gantt Chart)





	①	Name	Duration	Start	Finish	Predecessors	Resource Names	Sep 2025	Oct 2025	Nov 2025	Dec 2025	Jan 2026											
				31	07	14	21	28	05	12	19	26	02	09	16	23	30	07	14	21	28		
52		<b>4.2 Test execution</b>	<b>40 days</b>	<b>13/02/26 8...</b>	<b>9/04/26 5...</b>																		
53		4.2.1 System integration testing	10 days	13/02/26 8...	26/02/26 ...	50;51																	
54		4.2.2 User acceptance testing (UAT)	10 days	27/02/26 8...	12/03/26 ...	53																	
55		4.2.3 Accessibility testing	5 days	13/03/26 8...	19/03/26 ...	54																	
56		4.2.4 Security and privacy testing	5 days	20/03/26 8...	26/03/26 ...	55																	
57		4.2.5 Wearable field testing (Apple ...	10 days	27/03/26 8...	9/04/26 5...	56																	
58		<b>4.3 Bug fixing &amp; re-testing</b>	<b>37 days</b>	<b>10/04/26 8...</b>	<b>9/06/26 5...</b>																		
59		4.3.1 Log issues in tracker	5 days	10/04/26 8...	16/04/26 ...	53;54;55;56;57																	
60		4.3.2 Implementation of bug fixes	15 days	17/04/26 8...	7/05/26 ...	59																	
61		4.3.3 Re-test after fixes	10 days	8/05/26 8...	21/05/26 ...	60																	
62		4.3.4 TestFlight beta (iOS and Andro...	7 days	22/05/26 8...	1/06/26 5...	61																	
63		M5: Beta version tested	0 days	1/06/26 5:0...	1/06/26 5...	62																	
64		<b>5.0 Deployment</b>	<b>40 days</b>	<b>2/06/26 8:0...</b>	<b>27/07/26 ...</b>																		
65		<b>5.1 Deployment preparation</b>	<b>15 days</b>	<b>2/06/26 8:0...</b>	<b>22/06/26 ...</b>																		
66		5.1.1 Release deployment plan	10 days	2/06/26 8:0...	15/06/26 ...	62																	
67		5.1.2 Readiness review	5 days	16/06/26 8...	22/06/26 ...	66																	
68		<b>5.2 Production Launch</b>	<b>25 days</b>	<b>23/06/26 8:0...</b>	<b>27/07/26 ...</b>																		
69		5.2.1 Production environment setup	5 days	23/06/26 8...	29/06/26 ...	67																	
70		5.2.2 Data migration	15 days	30/06/26 8...	20/07/26 ...	69																	
71		5.2.3 Website submission	5 days	21/07/26 8...	27/07/26 ...	70																	
72		5.2.4 Wearable store submission (A...	5 days	21/07/26 8...	27/07/26 ...	70																	
73		M6: Launching Website & APP	0 days	27/07/26 8:0...	27/07/26 ...	71;72																	
74		<b>6.0 Closure</b>	<b>40 days</b>	<b>28/07/26 8:0...</b>	<b>21/09/26 ...</b>																		
75		<b>6.1 Reporting</b>	<b>10 days</b>	<b>28/07/26 8:0...</b>	<b>10/08/26 ...</b>																		
76		6.1.1 Final project report	5 days	28/07/26 8...	3/08/26 5...	71;72																	
77		6.1.2 Deliver documentation package	5 days	4/08/26 8:0...	10/08/26 ...	76																	
78		<b>6.2 Training &amp; Knowledge Transfer</b>	<b>10 days</b>	<b>11/08/26 8:0...</b>	<b>24/08/26 ...</b>																		
79		6.2.1 Training and knowledge transf...	5 days	11/08/26 8...	17/08/26 ...	77																	
80		6.2.2 Internal team handover compl...	5 days	18/08/26 8...	24/08/26 ...	79																	
81		<b>6.3 Post-Deployment Support (3 M...</b>	<b>20 days</b>	<b>25/08/26 8:0...</b>	<b>21/09/26 ...</b>																		
82		6.3.1 Monitor bug reports	10 days	25/08/26 8...	7/09/26 5...	80																	
83		6.3.2 Provide fixes during warranty ...	10 days	8/09/26 8:0...	21/09/26 ...	82																	
84		M7: Closure of the project	0 days	21/09/26 5:0...	21/09/26 ...	83																	

LoopCare Gantt Chart Group 605 - page3



## Cost Model

The cost estimate below showcases all the relevant costs required to carry out the project.

### [LoopCare Cost Model](#)

## Cost Estimation Methodology and Justification

- **Bottom-Up Estimation:** Used For Detailed Line-By-Line Breakdown.

The cost model was built from the ground up by linking each WBS activity to the resources required. Hours and costs for roles such as the Project Manager, developers, and QA testers were derived directly from the work packages in execution and testing phases. For example, the 70 days allocated to Full Stack Developers were tied to backend, frontend, and integration tasks in WBS 3.2–3.4. This role was outsourced to an offshore vendor, which allowed us to apply reduced

daily rates compared to local contractors. By working task by task, every cost could be traced back to a specific part of the plan, giving confidence that labour effort matched the complexity of what was scheduled rather than being a rough estimate.

- **Analogous Estimation:** Referencing Past Project Costs Or Market Comparisons.

To keep the model realistic, labour rates were benchmarked against trusted industry sources. Roles based in Australia, such as the Project Manager, Business Analyst, and QA staff, were costed using the Hays IT Contractor Rates Guide (2024/25). The outsourced Full Stack Developer role was benchmarked against Witarist (2025), which outlines average rates for software developers in India, and this supported the decision to use a \$150 daily rate. Hardware and software costs were referenced from vendors such as Apple, Lenovo, Samsung, Atlassian, and Figma, ensuring that the estimates reflected current market conditions.

- **Parametric Estimation:** Based On Unit Rates (E.G., Cost Per Hour, Cost Per Server).

Some costs were worked out using simple unit calculations multiplied over time. For example, Figma Pro was costed at \$135 per month and extended across the 13-month project to support UI/UX design tasks in WBS 3.1. Similarly, user testing costs were based on a flat \$50 incentive per participant, multiplied by the number of people recruited for UAT in WBS 4.2.2. Using unit rates in this way made recurring and scalable costs consistent and transparent.

#### *Justifications:*

The assumptions built into this model were grounded in both the WBS and external benchmarks. Labour rates for Australian-based roles came from Hays, while the outsourced Full Stack Developer rate was based on Witarist, reflecting a cost-effective option without compromising skill level. Hours were mapped directly to tasks in the WBS. For instance, QA Testers' allocations were tied to integration, accessibility, and security testing in WBS 4.2, while the Systems Engineer was linked to initiation and planning deliverables. Hardware and software prices came from

vendor sites, which provided reliable, up-to-date figures. Finally, vendor and tool choices were made for strategic reasons as much as cost: Apple and Samsung devices were selected to guarantee cross-platform compatibility, Figma and Jira were chosen because they are established industry tools, and outsourcing was introduced as a deliberate strategy to reduce labour spend while maintaining delivery capacity.

By combining bottom-up, analogous, and parametric estimation, the cost model is both transparent and defensible. It reflects the real scope of work in the WBS, aligns with industry standards, and balances efficiency with practicality to deliver a financial plan that supports successful delivery.

## Cost Baseline

The cost baseline was developed by phasing the project's total budget of \$325,744 across the timeline set out in the WBS. Each phase carries its share of the costs depending on when activities are scheduled. Early spending is relatively modest, covering initiation, planning and initial hardware/software purchases. Expenditure rises sharply during execution and testing, where the bulk of labour costs are incurred. This includes the outsourced full stack developer as well as mobile developer and QA tester whose hours are tied to WBS 3.2-3.4 and 4.2, which includes backend, frontend and integration deliverables. Costs then taper off through deployment and closure, where smaller but important allocations are set aside for training and documentation.

This baseline matches the totals from the cost model to make it easy to see where money is concentrated over time. Most of the budget is consumed through execution and testing, which is consistent with the complexity of design and coding. This also means that mid-project months show the heaviest spending, while the beginning and end of the project are lighter by comparison. This distribution provides a clear picture of how financial commitments align with delivery milestones, highlighting that the greatest outflows coincide with periods of intensive technical work.

To ensure the project remains within budget, costs will be tracked and controlled as work progresses. Major milestones will serve as natural checkpoints, where planned spending can be compared against actual outlays to confirm the project is tracking as expected. On top of this, regular monthly reviews will provide an extra layer of oversight which helps catch any emerging issues early in the project. If variances do occur, they will be investigated quickly and managed using the contingency reserve, ensuring the project stays financially viable without putting delivery at risk.

This approach keeps financial oversight practical and focused. The baseline gives a time-phased view of planned expenditure, while milestone and monthly reviews ensure those plans hold true as the project unfolds. By linking costs back to the WBS and aligning monitoring with deliverables, the project team is able to maintain a clear visibility over spending and respond to any issues that may arise.

## Project Risk Register

RISK ID	RANK	RISK DESCRIPTION	IMPACT DESCRIPTION	IMPACT LEVEL	PROBABILITY LEVEL	PRIORITY LEVEL	RISK RESPONSE	OWNER
A unique identifier	Based on Priority Level	Give a brief summary of the risk.	What will happen if the risk is not mitigated or eliminated?	Rate 1 (LOW) to 5 (HIGH)	Rate 1 (LOW) to 5 (HIGH)	(IMPACT X PROBABILITY) Address the highest first.	What can be done to lower or eliminate the impact or probability?	Who's responsible?

R01		Team capacity drops	Increased workload for remaining staff, going off schedule	4	4	<b>16</b>	Mitigate - Leave blockouts close to deadlines, cross-skill pairing	Project Manager
R02		iOS and WatchOS app rejected	Delays submission and deployment schedule	5	3	<b>15</b>	Mitigate - Pre-checks against Apple's standards and requirements, TestFlight pilot and implementing a re-submission buffer	Software Developer
R03		Delays in backend API development	Due to technical complexity	3	5	<b>15</b>	Mitigate - Breakdown modules into smaller sprints, assign extra developers, and conduct frequent reviews.	Software Developer
R04		Increase in cost estimates	Due to unexpected licensing fees, third	3	4	<b>12</b>	Transfer - Negotiate contracts early. Accept -	Project Manager

			<i>party APIs, or hosting services</i>				<b>Allocate 15% contingency cost</b>	
R05		Key client unavailable	Delays in approval (charter, scope, design, UAT)	4	3	<b>12</b>	Mitigate - Set a backup approver, book reviews 1-2 weeks in advance	Project Manager
R06		Poor user-interface design	Reduce usability and may lead to negative adoption reviews	3	4	<b>12</b>	Mitigate - Follow industry level design standards and conduct usability testing	UX/UI Designer
R07		Sponsor disengagement	Reduce project support or funding	5	2	<b>10</b>	Mitigate - Regular sponsor updates, and align deliverables with business goals.	Project Manager
R08		Fail WCAG 2.2 AA	Re-do UI/UX and front-end development	3	3	<b>9</b>	Mitigate - Conduct thorough usability testing	UX/UI Designer

# Risk Prioritisation

## Prioritisation justification:

R01 (Team capacity drops) was given a high priority as this would affect every work stream and directly threatens the schedule making it top priority. R02 (iOS/ watchOS app rejected) was also given a high priority as you cannot go-live without app submission. Many issues are also only caught at submission so the possibility of the risk is also high. R03 (delays with backend API delays) was given a priority level of 15 as it is a critical path. Without this going ahead, other paths cannot move forward without core APIs, dashboard and integrations. R04 (increase in cost estimates) has a medium priority as there has been an allocated \$15,000 contingency but the probability of increase in cost expenditure is high. R05 (key client unavailable for approval) was also rated medium as key pathways cannot proceed without approval including the Charter, requirements and design. The probability of it remains medium as it can be mitigated with prior planning. R06 (poor user interface design) was rated medium as a poor design can lead to low adoption but is preventable. R07 (sponsor disengagement) given a priority level of 10 as slow feedback and changing targets may cause sponsor withdrawal, increasing impact and probability. R08 (fail WCAG 2.2 AA) was also given a medium priority level as they are highly controllable but have a very high chance of occurring during development.

## Response planning:

### R01:

Preventative controls include block-outs for leave near deadlines and also cross skill pairing to avoid tasks being delayed due to staff shortages (mitigation). Availability should also be monitored through the week and more closely during execution. Non-critical work can be resequenced as well. An opportunity could be using interns for more simpler tasks such as testing automation.

**R02:**

Mitigate is the primary strategy with app store pre-checks for privacy and HealthKit purposes. TestFlight and Samsung Developer pilot to check during testing. Rapid resubmission within 48 hours should also be implemented along with a release buffer to account for rejection.

**R03:**

Mitigation by breaking bigger API features into smaller tasks, agreeing on a simpler API contract early and building mock endpoints. Another developer could be added as a contingency.

**R04:**

The strategy is transferring for fixed-price deals, capping cloud usage and also setting spend alerts. If the cost does go over, the \$15,000 reserve can be used and moving to cheaper tiers for subscriptions and cloud could be implemented.

**R05:**

A backup approver could be agreed upon and review meetings could be booked 1-2 weeks in advance to mitigate risks. If it does occur, changes could be made and recorded, and escalation could be needed.

**R06:**

Following Apple/ Android guidelines and testing the prototype thoroughly with users could be used to mitigate the risks associated with poor user-interface design. Fixing the top issues before UAT and pushing cosmetic adjustments to later should be a strategy that is implemented while fixing bugs.

**R07:**

Meetings with clients every 2-weeks to update them on recent developments and showing a quick demo could keep the sponsors engaged in the project and mitigate the risk. The meetings could also include the decisions needed from their end to keep on schedule.

**R08:**

Mitigation of the risk through the use of contrast tokens and conducting frequent and thorough usability testing. If it does occur, the release can be blocked, then the issue fixed and retested within a 48 hour period. Minor issues can be fixed in a later version.

**Time and cost analysis:**

**R01:** The preventative cost for block outs and skill pairing is \$400 (4 hours with Project manager). If the risk happens, 10 days could be added to the current schedule which would result in developers working across 10 days (\$1,500, 80 hours).

**R02:** Preventative costs include prechecks and Beta testing involving the developers (16 hours, \$300) and project manager (4 hours, \$400) which makes it \$700 in total. If the risk happens, it would result in a 7 day slip in schedule for resubmission (8 hours and \$800 for PM and developers working 24 hours costing \$450).

**R03:** This can be prevented through more tests which will be conducted by developers (32 hours, \$600). If the risk happens, then the developers will work over 5 days (\$750).

**R04:** Preventative measures will be taken by the project manager (6 hours, \$600) and Developers (4 hours, \$75). If the risk occurs, this will not result in any changes to schedule.

**R05:** This is prevented through booking reviews and setting a backup reviewer, which is done by the project manager (3 hours, \$300). If the risk were to occur, it could push the schedule back by up to 10 days and the project manager will have to work an additional 40 hours (\$4000).

**R06:** Preventative measures can be taken through prototype testing and analysis of usability and efficiency (UX designers (12 hours, \$1,125) and developers (8 hours,

\$150)). If the risk happens, the 2 day slip will result in developers working for 2 days (\$300) and UX designer for 1 day (\$750) to fix critical issues.

**R07:** Sponsor disengagement includes fortnightly demos as preventative measures which is already accounted for. If the risk were to occur, the schedule would be affected by 2 days (8 hours from Project manager, \$800).

**R08:** Preventative costs will include UX designer (8 hours, \$750) and QA (8 hours, \$500). If the risk were to happen, it would result in a 2 day slip, with developers working across the 2 days (\$300) and the QA working for 1 day (\$500) to fix and re-test.

## Group Reflection

Working on Loopcare has allowed our team to collaborate effectively under time and scope constraints. We anchored teamwork in our charter by clarifying roles, expectations and feedback norms (PMI, 2021). In our team leadership wasn't confined to one person as everyone took charge for different aspects enabling us to work efficiently. For example, one member in our team was able to take control of the WBS and Gantt chart, while someone else helped lead the risk register. This approach to taking leadership on different tasks meant we could play to each other's strength, which echoes Tuckman's idea which talks about teams being more effective once they reach a performing stage (Tuckman, 1965).

Support was one of our strongest points. This was due to the fact that we checked up on each other's work regularly allowing us to resolve any confusions we had on what needs to be done. When one of us felt unsure, others stepped in with advice or shared templates to guide the process. A good example was when the requirements traceability matrix seemed unclear. Rather than leaving it one person, the group worked together to refine it. This kind of support created a sense of physiological safety where people felt comfortable asking for help (Edmondson, 1999).

Feedback also played a big role in our collaboration. We made a habit of changing our work regularly depending on everyone's ideas so that everyone had a chance to give input. The feedback was constructive and linked to the rubric, which meant it was focused on improving quality rather than just pointing out mistakes. This open exchange helped us polish documents like the charter and scope statement to be professional. It also made the process feel less stressful because we knew we could rely on each other's suggestions.

Conflicts did arise throughout the assignment, which was mainly around how long tasks should take or what should be considered in scope. However, instead of letting disagreements escalate, we looked back at the evidence from our WBS and acceptance criteria to figure out what we needed. This kept the discussions objective and allowed us to make decisions without creating tension. By doing this, we were able to stay on track and maintained good teamwork within the group.

Looking back, we can see areas that we could have improved. One challenge was that we sometimes underestimated how long certain tasks would take. To fix this we could have built a more realistic timeline by leaving extra buffer for reviewing each section of the project which would have allowed us to complete each section with more time. Overall, though, our teamwork on LoopCare was constructive, supportive and adaptive which will be useful for future projects.

## Individual Reflections

Amasha (33904464) Individual Reflection:

**Description:** In the LoopCare project assignment for FIT2002, I had joined as a planner. I was involved in coordinating the charter, team working agreement, WBS, scheduling meetings, assigning tasks, the Gantt chart and risk management for the project.

**Feelings:** I had begun the project quite confident in planning but anxious about balancing the evolving requirements and deliverables with fixed dates. Mid-pressure rose when we strayed from our previously agreed upon timelines.

**Evaluation:** I created structured templates and scheduled meetings with clear agendas to keep us aligned with the rubric. I set up the ProjectLibre file, identified tasks and ensured group members had updated materials. However, I sometimes accepted overly optimistic task durations and delayed starting my own work, which limited the time I had to support team members and refine drafts.

**Analysis:** When I overpolished the charter and WBS, specifically the Gantt chart, progress slowed and timeline issues appeared later. Prioritising certain tasks over others, sticking to stricter deadlines and collaborating with team members with different perspectives helped strengthen deliverables.

**Conclusion:** My organisational strengths helped the team, however, I grew the most by learning to challenge timelines, start earlier and seek help when needed. I leave the project more confident about balancing quality with pace, and in inviting peer feedback and drafting earlier.

**Action Plan:** Going forward, I want to ensure I begin tasks within a day of allocation, review drafts of tasks with my team members and do a weekly reflection of my approach.

**Grade Evaluation:** I would give myself a Distinction for strong planning leadership. I would give my team a Distinction, as we collaborated effectively and maintained a high standard overall.

Ashwin (35003804) Individual reflection:

**Description:** In the LoopCare project, my main contributions were drafting the scope statement, assisting with the requirements traceability matrix, and supporting the cost model. I also took part in group discussions, especially when aligning deliverables with the rubric.

**Feelings:** At first, I felt overwhelmed by the number of tasks, but I grew more confident once I began working on the timeline. I enjoyed collaborating because it gave me the chance to test my ideas against my teammates' perspectives, which made me feel more secure in my contributions.

**Evaluation:** I believe I was effective in producing clear drafts that aligned with assignment requirements. My attention to detail helped identify gaps early, though I could have spoken up more in planning meetings rather than waiting until later to contribute, which would have saved time.

**Analysis:** This experience highlighted the need to balance detail with efficiency. Spending too long perfecting one section risked slowing progress, so I learned to prioritise and ask for help earlier. It also reinforced how valuable constructive peer review is for improving quality.

**Conclusion and Action Plan:** Next time, I want to be more proactive early in planning and contribute more during live discussions. I will also seek feedback more often to broaden my ideas.

**Evaluation:** I would give myself a distinction, as I delivered quality work and collaborated effectively. For the team, I would assign a high distinction, as our teamwork was strong and we supported each other consistently.

Manumi Ekanayake (35232617) Individual reflection:

**Description:** I worked in a team of four people to design LifeLoop, a digital wellness app called LifeLoop. I was in charge of developing the Requirements Traceability Matrix (RTM) and writing the Project Scope Statement to establish requirements, assumptions, and limits. I also worked on the Work Breakdown Structure (WBS) and Gantt chart, ordered the tasks and matched milestones.

**Feelings:** In the initial days I was nervous about the size and the tight deadlines of the project and the fact that some of the major deliverables were to be done within

the first six weeks. Division of tasks into bits and working closely with fellow teammates enhanced my confidence and motivation. I understood the power of working as a team and doing a lot collectively more than individually.

**Evaluation:** Documentation felt pressured by time, and the good communication via Instagram and Google Meets helped us to maintain the pace. We were able to proceed in a linear manner and divide the responsibilities equally and welcome feedback.

**Analysis:** I also got to know that teamwork and flexibility are equally significant as technical skills. We used to meet the deadlines with the help of time management and communication. Other factors that contributed to success were leadership and accountability.

**Conclusion:** I believe that I contributed greatly with regard to deadline keeping and standing behind my teammates. The work on the Requirements Traceability Matrix (RTM) and Scope Statement made sure that the requirements and boundaries were clearly understood, and my input contribution to the WBS and Gantt Chart was useful to organize the project schedule. However, I could have done more to identify the possible risks in these planning tasks earlier and this would have enhanced the project outcomes further.

**Action Plan:** Going forward, I will assume more authorship roles, enhance risk awareness and keep offering constructive feedback to enhance team performance.

**Evaluation:** Self-evaluation: Distinction - I performed my tasks reliably and contributed to team success, though I could have done better with my parts.

Team: High Distinction - The team worked effectively, with strong communication, responsiveness to feedback, and a supportive environment that allowed us to complete the project successfully.

Muthu reflection:

**Description:** My contributions to the loopcare project centred on the cost model, cost baseline and working heavily on the gantt chart. I was also reviewing

teammates' sections to ensure consistency across deliverables and alignment with the rubric

**Feelings:** At the start, I felt uncertain about whether I could manage the cost sections, since I had not worked much with estimation frameworks before. However, as I linked costs back to the WBS and researched benchmarks, I grew more confident. I enjoyed learning from my teammates' perspectives and felt motivated when our ideas came together in a unified report

**Evaluation:** I believe I delivered solid work on the cost model and gantt chart by carefully using industry references and tying them to the WBS. I also provided helpful feedback when going over teammates' work. However at a stage I underestimated how long it would take to refine the baseline, which created last-minute pressure. I could have started that task a little bit early or asked for more help to get it completed.

**Analysis:** This project showed me the importance of combining technical accuracy with team communication. Producing a cost model wasn't only calculations as it also involved explaining my reasons clearly to others. I realised that contributing feedback on different sections helped strengthen the whole report and not just my part.

**Self-evaluation (Distinction):** I carried out my responsibilities consistently and supported the team's success, though there is room for improvement in my individual contributions.

**Team evaluation (High Distinction):** The team collaborated effectively, maintained clear communication, responded well to feedback, and fostered a supportive environment that enabled us to deliver the project successfully.

## References

# References

- Edmondson, A. (1999). Psychological safety and learning behavior in work teams. *Administrative Science Quarterly*, 44(2), 350–383. <https://doi.org/10.2307/2666999>
- Project Management Institute. (2021). *A guide to the project management body of knowledge (PMBOK® Guide)* (7th ed.). PMI.
- Tuckman, B. W. (1965). Developmental sequence in small groups. *Psychological Bulletin*, 63(6), 384–399. <https://doi.org/10.1037/h0022100>
- Atlassian. (2025). *Jira software pricing*. Atlassian.  
<https://www.atlassian.com/software/jira/pricing>
- BrowserStack. (2025). *Pricing plans*. BrowserStack. <https://www.browserstack.com/pricing>
- Figma. (2025). *Professional plan pricing*. Figma. <https://www.figma.com/pricing>
- Hays. (2024). *Hays salary guide FY24/25*. Hays Recruitment.  
<https://www.hays.com.au/salary-guide>
- Lenovo. (2025). *ThinkPad laptop pricing*. Lenovo.  
<https://www.lenovo.com/au/en/laptops/thinkpad>
- Apple. (2025). *MacBook Pro M4 – Technical specifications*. Apple.  
<https://www.apple.com/au/macbook-pro>
- Apple. (2025). *iPhone 17 – Technical specifications*. Apple. <https://www.apple.com/au/iphone>
- Samsung Australia. (2025). *Galaxy S25/S25+ pricing*. Samsung.  
<https://www.samsung.com/au/smartphones/all-smartphones>
- Samsung Australia. (2025). *Galaxy Watch pricing*. Samsung.  
<https://www.samsung.com/au/watches>
- Witarist. (2025). *Cost to hire a software developer in India*. Witarist.  
<https://witarist.com/blogs/cost-to-hire-a-software-developer-in-india/>

Zwikael, O., & Globerson, S. (2006). From critical success factors to critical success processes. *International Journal of Production Research*, 44(17), 3433–3449.  
<https://doi.org/10.1080/00207540600575714>

Vanhoucke, M. (2014). An overview of project data for integrated project management and control. *Journal of Modern Project Management*, 1(2), 34–55.  
<https://doi.org/10.19255/JMPM01103>

Fleming, Q. W., & Koppelman, J. M. (2010). Earned value project management—an introduction. *Project Management Journal*, 41(1), 17–24.  
<https://doi.org/10.1002/pmj.20158>

Kerzner, H., & Saladis, F. P. (2022). *Project management: A systems approach to planning, scheduling, and controlling* (13th ed.). Wiley.