Of course. Based on the detailed information and analysis documents you have provided, I have completed the "Team and Project Plan" for your project.

Here is the completed document:

Team and Project Plan

PORTFOLIO TASK 1

Unit code: COS40005
Unit Name: Computing Technology Project A
Submission date: *<Date of Submission>*

Student Name	Student Id	Description of contribution in team and project planning
Daniel	102777801	Coordinated project planning, led the definition of the solution architecture, and drafted the primary sections of the project plan, including the AI strategy and risk mitigation.
Alif Harriz	102782711	Contributed to the front-end technology selection process, provided input on UI/UX considerations for the dashboard milestones, and helped define the teamwork roadmap.
Ashley Jong	102780087	Assisted in gathering project requirements from the proposal, contributed to the analysis of system components, and helped structure the product backlog.
Basil Agas	102778888	Provided key insights during the AI framework analysis, contributing to the decision on the agentic AI architecture and the definition of AI-related tasks.
Edison Ho	102779496	Led the analysis of backend and database technologies, contributed to the selection of the BaaS platform, and helped outline the document management and quality plan sections.

ACKNOWLEDGMENT OF COUNTRY

<Please complete the statement>

Each team member identifies: the Traditional Owners of the land they lived on while completing this work (if living in Australia).

PART 1 TEAM CODE OF CONDUCT

1. TEAM PROFILE

The team is a multidisciplinary group of computer science students with complementary skills in Artificial Intelligence, Software Development, and Full-Stack Engineering. Daniel and Basil provide a strong foundation in AI and systems architecture. Alif brings specialized front-end and UX design skills, ensuring a user-centric approach. Edison offers robust full-stack and backend expertise, crucial for system integration. Ashley provides a blend of software development and AI knowledge, alongside project management skills, ensuring a cohesive and well-documented development process.

Table 1: Team Profile

Student name	Technical skills	Soft skills	Communication	Teamwork
Daniel	Full-stack development, AI/ML (agentic systems, image classification), multi-platform apps (IoT, games), UI design.	High attention to detail, passion for automation, user-focused design sense.	Proactive, formal written communication, primary client contact point.	Initiative-taker, project coordinator, integrator between front-end, back-end, and AI.
Alif Harriz	Front-end (HTML, CSS, JS, Vue.js), UI/UX design, prototyping (Figma).	User empathy, focus on accessibility, intuitive design.	Prefers visual communication (prototypes), comfortable with asynchronous chat.	Focuses on translating design into functional product, open to feedback.
Ashley Jong	Software design, system analysis, AI-driven problem solving, project management, requirement gathering, testing.	Detail-oriented, methodical, adaptable, strong organisational skills.	Comfortable with formal and technical discussion across multiple channels.	*Awaiting task allocation to define specific teamwork contribution.*
Basil Agas	AI development (NLP, predictive analytics), AI architecture, infrastructure (Cisco, Windows Server).	Analytical, strategic, focus on practical and ethical AI deployment.	Articulates complex technical concepts clearly.	Focuses on developing, deploying, and optimizing the core AI solutions.
Edison Ho	Full-stack development (PHP, Python), database management, cybersecurity fundamentals.	Writes clean, maintainable code, strong UI/UX consideration.	Proactive, transparent, presents ideas clearly to ensure team alignment.	Reliable, takes ownership of tasks, ensures smooth integration with other components.

2. TEAM ROLE

Team roles have been assigned based on individual strengths, academic specializations, and interests to maximise efficiency and quality for the project's distinct components.

Table 2: Team Role

Student name	Team role	Justification

Student name	Team role	Justification
Daniel	**Project Lead & AI Engineer**	With a full-stack perspective and a major in AI, Daniel is ideally suited to lead the project, ensuring all parts (frontend, backend, AI) integrate seamlessly. He will co-lead the AI development.
Alif Harriz	**Frontend Developer (Clinician Dashboard)**	Alif's focus on front-end development and UX design makes him perfect for creating the data-rich, intuitive, and responsive clinician dashboard.
Ashley Jong	**Frontend Developer (Patient Dashboard)**	Ashley's double major in Software Development and AI, combined with her user-focused approach, makes her well-suited to build the patient-facing dashboard, ensuring it is both functional and easy to navigate.
Basil Agas	**AI Engineer**	Basil's specialisation in the architectural and operational aspects of AI systems makes him the ideal candidate to focus on developing and optimising the core recommendation engine and automation triggers.
Edison Ho	**Backend Developer**	Edison's hands-on experience in building full-stack applications and managing databases is critical for developing the server-side logic, data ingestion pipeline, and overall backend infrastructure.

3. TEAMWORK ROADMAP

- **Meeting Schedule:** The team will hold mandatory meetings during scheduled class times on Tuesdays and Thursdays. Two additional working sessions will be scheduled on Fridays and Saturdays. Meetings will be in-person by default, with online options available if necessary.
- **Communication:** WhatsApp will be the primary channel for informal daily communication and quick updates, with an expected response time of 12-24 hours. Formal communication will be handled via email.
- **Task Management:** All project tasks will be managed using **GitHub Projects** and **GitHub Issues**. Tasks will be assigned to individuals, given deadlines, and tracked on a project board to ensure transparency and monitor progress.
- **Deadlines & Revisions:** For all weekly submissions and major milestones, an internal draft deadline will be set for 48 hours prior to the official due date. This allows for a 24-hour window for peer review and a final 24-hour window for revisions.

4. DOCUMENT MANAGEMENT

- **Code:** All source code will be version-controlled using **Git** and hosted on **GitHub**. The team will use a feature-branch workflow with pull requests for code reviews before merging into the main branch.
- **Documents & Collaboration:** Project documentation, reports, and collaborative files will be managed on **Microsoft OneDrive** and edited using the Microsoft Office 365 suite to leverage real-time collaboration and version history.
- **Backups:** Regular manual backups of critical documents and code will be maintained to prevent data loss.

5. RISK MITIGATION

The following risks have been identified, along with a plan to mitigate their impact.

Risk ID	Risk Description	Impact	Likelihood	Mitigation Plan
R1	**Technical Skill Gap**	Medium	Medium	Promote a culture of knowledge sharing and peer-to-peer training. Allocate time for members to learn new technologies. If a critical gap is identified, the team will consult with the project supervisor for guidance.
R2	**Team Member Unavailability**	High	Low	Implement a "buddy system" where key responsibilities are shared between two members. Ensure all code is well-documented and pushed to GitHub daily so others can take over if needed.
R3	**Scope Creep**	Medium	Medium	All change requests must be formally documented and reviewed by the Project Lead. The team will assess the impact on the project timeline and consult with Dr. Vong before agreeing to any changes to the original scope.
R4	**Integration Issues**	High	Medium	Define clear API contracts between the frontend, backend, and AI services early in the project. Conduct regular integration testing throughout the development cycle, not just at the end.

PART 2 PROJECT OVERVIEW AND PLAN

1. PROBLEM STATEMENT

Chronic disease patients face challenges due to fragmented health data and a lack of real-time, personalised feedback. Current systems often fail to leverage AI to detect early warning signs or provide actionable recommendations. The aim of this project is to develop a prototype of an AI-enabled digital health platform that integrates multi-source health data (glucose, activity, diet) to provide timely, AI-powered insights, empowering both patients and clinicians to manage chronic conditions more effectively.

2. SCOPE

- **In-Scope:**
  - Development of all features outlined in the six project milestones.
  - A functional prototype integrating patient and clinician views.
  - An AI-powered recommendation engine and automation layer.
  - A functional patient-facing chatbot.
  - Implementation of role-based access control for security.
  - Use of simulated patient data for development and testing.
- **Out-of-Scope:**

- Deployment to a live production environment with real patient data.
- Integration with real, physical health monitoring devices.
- The legal process of achieving HIPAA compliance (this is a client responsibility for a production system).
- Company-level administrative dashboards.

3. STAKEHOLDER

- **Client:** BioTective Sdn Bhd
- **Project Supervisor:** Ts. Dr. Vong Wan Tze
- **End Users:** Patients with chronic diseases and their associated healthcare providers (clinicians).
- **Development Team:** The student project team (Daniel, Alif, Ashley, Basil, Edison).

4. HIGH-LEVEL REQUIREMENTS

This project is the development of a new, complete prototype system.

- **Functional Requirements:**
  - Secure user authentication with patient and clinician roles.
  - A data ingestion pipeline for simulated health data (glucose, HbA1c, diet, activity).
  - A patient dashboard with data visualizations (trend charts, weekly summaries).
  - A clinician dashboard displaying an aggregated view of their assigned patients.
  - An AI engine that provides personalised health recommendations.
  - An automation layer (LAM Triggers) to send alerts and reminders based on health patterns.
  - A patient-facing chatbot for health data Q&A.
- **Non-Functional Requirements:**
  - **Platform:** The system must be mobile-oriented (primary) and accessible via a web browser (secondary).
  - **Security:** The system must implement role-based access control to ensure data privacy.
  - **Usability:** The interface should be intuitive and accessible to non-technical users, including older adults.

5. SOLUTION APPROACH

The proposed architecture is based on a comprehensive analysis of modern development frameworks, prioritising developer efficiency, scalability, and suitability for AI integration.

- **Architecture:**
  - **Frontend (Mobile & Web): Flutter**. This cross-platform framework was chosen to meet the "code once, deploy anywhere" requirement, ensuring a high-performance, native-like experience on both mobile and desktop from a single codebase. The `fl_chart` library will be used for data visualizations.
  - **Backend: Python**, using a lightweight framework like **FastAPI**. This is non-negotiable due to its extensive libraries and robust support for developing and serving AI models.
  - **Database & BaaS (Backend-as-a-Service): Supabase**. Chosen for its powerful PostgreSQL foundation, integrated authentication, and, most critically, its enterprise-grade Row-Level

  - Security (RLS) feature, which is ideal for implementing the required security and role-based access rules at the database level.
  - **AI Agent Framework: LangChain**. This framework will be used to build the "Large Action Model" (LAM) system, enabling the LLM to use "tools" (custom Python functions) to interact with the database and other services.

- **Alternatives Considered:**

  - **Frontend:** Quasar Framework was a strong contender for its exceptional development velocity, especially for web-based applications. However, Flutter was chosen for its superior performance and custom UI capabilities, which are better suited for a premium mobile experience.
  - **BaaS:** Firebase was the leading alternative, particularly for its seamless integration with the Google Cloud AI ecosystem. However, Supabase was selected for its open-source nature and the powerful, fine-grained control offered by its PostgreSQL and Row-Level Security features.

6. PRODUCT BACKLOG

The project will be developed across three major sprints, with each sprint covering two of the project's milestones.

ID	Task/User Story	Priority	Depends On	Tentative Sprint
EPIC 1: Project Foundation & Patient View (Milestones 1 & 2)				
T1.1	Define system architecture and set up GitHub repository	**High**	-	1
T1.2	Set up Supabase project with initial database schema	**High**	T1.1	1
T1.3	Implement secure login and role-based access (patients/clinicians)	**High**	T1.2	1
T1.4	Develop patient data simulator to generate realistic datasets	**High**	T1.2	1
T2.1	Build the functional patient dashboard UI using Flutter	**High**	T1.3	1
T2.2	Integrate data visualizations (glucose, HbA1c, diet, activity charts)	**High**	T2.1, T1.4	1
EPIC 2: AI Core & Automation (Milestones 3 & 4)				
T3.1	Research and select a base LLM for the recommendation engine	**High**	-	2

ID	Task/User Story	Priority	Depends On	Tentative Sprint
T3.2	Develop the personalized recommendation engine module in Python	**High**	T3.1	2
T3.3	Integrate the engine to provide suggestions on the patient dashboard	**High**	T3.2, T2.1	2
T4.1	Design and implement the automation layer (LAM Triggers)	**High**	T3.2	2
T4.2	Create triggers for glucose spike reminders and motivational prompts	**Medium**	T4.1	2
EPIC 3: Clinician View & Finalisation (Milestones 5 & 6)				
T5.1	Build the functional clinician dashboard UI using Flutter	**High**	T1.3	3
T5.2	Implement aggregated patient data view and anomaly highlighting	**High**	T5.1	3
T5.3	Develop the patient-facing chatbot interface	**Medium**	T3.1	3
T6.1	Conduct end-to-end system testing with multi-patient datasets	**High**	All	3
T6.2	Strengthen data security (encryption, anonymization review)	**High**	All	3
T6.3	Finalize project documentation and user guide	**High**	All	3

7. QUALITY PLAN

The team is committed to delivering a polished, functional, and reliable prototype. Quality will be assured through the following processes:

- **Code Reviews:** All code will be submitted through GitHub Pull Requests and must be reviewed and approved by at least one other team member before being merged. This ensures code quality, consistency, and knowledge sharing.
- **Coding Standards:** The team will adhere to established style guides: **PEP 8** for Python and **Effective Dart** for Flutter.

- **Testing:**
    - **Unit Testing:** Critical backend logic, especially within the AI and automation modules, will be covered by unit tests.
    - **Integration Testing:** Regular testing will be conducted to ensure the frontend, backend, and database are communicating correctly.
    - **User Acceptance Testing (UAT):** At the end of each sprint, team members will perform manual, end-to-end testing from the perspective of both patient and clinician users to identify bugs and usability issues.
- **Final Deliverable Expectation:** The final prototype will be a polished and functional mobile application that successfully demonstrates all core features, with a particular focus on the seamless operation of the AI-driven recommendation and action (LAM) components.