A Project Proposal On

Recovery Ally

Submitted in the Partial Fulfillment of the
Requirements for the Degree of Bachelor of Software Engineering
Awarded by Pokhara University

Submitted By:

Dipak Kumar Biswakarma (21185154)

Sanchit Poudel (21185174)

Saurabh Adhikari (21185179)

School of Engineering

Faculty of Science and Technology

POKHARA UNIVERSITY

Nov 2024

ABSTRACT

"Recovery Ally" is an innovative digital platform dedicated to enhancing post-surgery rehabilitation by providing personalized exercise plans tailored to individual recovery needs. This app empowers physiotherapists to design customized exercise programs for their patients based on injury type and recovery goals. Each plan features detailed instructions, video demonstrations, and essential tips to ensure proper execution and reduce the risk of complications. The app categorizes exercises by conditions such as orthopedic surgery, joint replacement, and soft tissue recovery, enabling targeted rehabilitation. With an emphasis on evidence-based practices and a user-friendly design, "Recovery Ally" is a vital resource for therapists and patients seeking effective and structured recovery pathways.

Keywords: *User Authentication, Orthopedic Surgery, Rehabilitation*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

API	Application Program Interface
DB	Database
HTTP	Hypertext Transfer Protocol
REST	Representational State Transfer
SSR	Server-Side Rendering
UI/UX	User Interface/ User Experience

CHAPTER 1:
INTRODUCTION

1.1 Background

In the dynamic and ever-evolving field of healthcare and physiotherapy, "Recovery Ally" is conceived as a transformative platform aimed at addressing the unique needs of post-surgery recovery. With advancements in medical care leading to a higher number of successful surgeries, the demand for structured and personalized rehabilitation has grown significantly. However, traditional recovery resources often lack customization, accessibility, and modern technological integration, creating challenges for both patients and physiotherapists. "Recovery Ally" seeks to bridge this gap by providing a dedicated platform where therapists can design and manage personalized rehabilitation plans tailored to each patient's injury type and recovery stage. Unlike generic resources, this project focuses on creating a centralized digital space equipped with evidence-based exercise routines, video demonstrations, and guided instructions to support safe and effective recovery. The project is grounded in the understanding that post-surgery recovery is not a one-size-fits-all process. By leveraging technology to deliver tailored solutions, "Recovery Ally" aims to redefine the rehabilitation experience, empowering patients with accessible tools and fostering a stronger connection between patients and therapists. With its emphasis on user-centered design and innovative features, "Recovery Ally" aspires to set a new standard in post-surgery care, addressing the critical need for effective and personalized recovery solutions.

1.2 Problem Statement

In developing countries like Nepal, access to effective post-surgery rehabilitation services remains a significant challenge. Limited healthcare resources, understaffed physiotherapy centers, and a lack of affordable and personalized rehabilitation solutions often hinder patients' recovery. Many individuals, especially in rural and underserved areas, face difficulties in accessing physiotherapists or structured rehabilitation programs, leading to prolonged recovery periods and increased risk of complications. Additionally, the reliance on generalized recovery plans fails to address the unique needs of patients recovering from specific surgeries or injuries. This lack of tailored support further exacerbates disparities in healthcare outcomes. For physiotherapists, the challenge lies in managing a high patient load with limited tools and infrastructure, making it difficult to deliver personalized care

effectively. "Recovery Ally" seeks to address these issues by providing an accessible, digital platform tailored to the needs of patients and therapists in developing countries. By offering affordable, evidence-based rehabilitation plans that can be customized to individual requirements, along with easy-to-follow multimedia guidance, the platform aims to bridge the gap in rehabilitation services and ensure equitable access to quality post-surgery care.

1.3 Objectives

Overall objectives of this project include:

- To Enhance Physiotherapist Capabilities
- To Improve Recovery Outcomes after surgery.
- To Leverage Technology for Innovation

1.4 Applications

"Recovery Ally" is a versatile platform with a wide range of applications aimed at improving post-surgery rehabilitation. Its innovative features make it beneficial to patients, physiotherapists, healthcare institutions, and even remote communities. The key applications include:

- Post-Surgery and Injury Recovery.
- Personal Rehabilitation and Self-Care
- Supplementary Resource for Physiotherapists.
- Chronic Pain Management.
- Health Education and Awareness
- Data Analytics for Therapists
- Remote Monitoring and Support

1.5 Project Features

Project Features of Recovery Ally:

- Real-Time Progress Tracking.
- Personalized Rehabilitation Plans.
- Cost-Effective Solution.
- Exercise Reminders and Notifications.
- Educational Resources.

1.6 Feasibility Analysis

We will assess the economic, technical, and operational feasibility of Recovery Ally.

1.6.1 Economic Feasibility

“Recovery Ally” offers a cost-effective solution to traditional physiotherapy, reducing the need for frequent in-person visits while still providing high-quality care. Initial development costs will primarily involve building the platform, creating content, and integrating telehealth features, but these expenses can be offset by scalable digital infrastructure and cloud services. Ongoing costs will include maintaining the website, updating content, and compensating healthcare professionals for consultations, which can be managed through subscription-based models or pay-per-consultation fees. Additionally, the platform can generate revenue through partnerships with clinics, insurers, or wellness programs. With a large, growing market of individuals seeking remote rehabilitation solutions, “Recovery Ally” has strong potential for profitability while offering affordable, accessible care.

1.6.2 Technical Feasibility

The "Recovery Ally" platform leverages a robust technology stack to ensure a seamless and secure user experience. NextAuth.js is used for authentication, enabling secure login with various providers like Google or email-based authentication, while JWT (JSON Web Tokens) manages token-based access for secure, role-based authorization. Real-time communication between therapists and patients is facilitated using Socket.io, enabling instant updates, notifications, and progress tracking. Cloudinary is integrated for efficient media storage, allowing for fast and reliable handling of images and videos related to exercises, rehabilitation plans, and patient profiles. The platform's UI is built using Shadcn UI, offering a modern, responsive, and user-friendly design. MongoDB serves as the database, providing flexible and scalable storage for user data, rehabilitation plans, progress tracking, and other dynamic content. Together, these technologies ensure that "Recovery Ally" is not only functional but also secure, scalable, and equipped with advanced features for a comprehensive rehabilitation experience.

1.6.3 Operational Feasibility

The "Recovery Ally" platform relies on its ability to integrate advanced technology, ensure seamless user experience, and manage resources efficiently. The platform will provide a

user-friendly interface for both patients and therapists, with features such as personalized rehabilitation plans, progress tracking, and real-time feedback. By leveraging cloud-based infrastructure, the platform will be scalable and accessible, even in areas with limited internet connectivity. Moreover, continuous monitoring and updates will be implemented to ensure the platform remains secure, efficient, and aligned with healthcare standards. With a well-defined operational structure and clear user engagement strategies, the platform aims to deliver consistent value to both healthcare providers and patients.

1.7 System Requirement

1.7.1 Hardware Requirement

- **Development Machines:** Development machines will need at least 8GB of RAM and multi-core processors for efficient development.
- **Hosting Infrastructure:** Cloud-based servers with a minimum of 4GB RAM, 2 CPU cores, and 20GB of storage will be used for hosting the application.
- **End-User Devices:** Users can access the platform via desktops, laptops, or mobile devices with internet connectivity, with sufficient cloud storage for data and backups.

1.7.2 Software Requirement

- **Design Environment:** In the design environment, Wireframe.cc is used for creating simple and intuitive initial wireframes, providing a clear blueprint of the layout and functionality. Once the wireframing stage is complete, Figma takes center stage for designing the UI/UX, enabling detailed, interactive, and collaborative designs that bring the project to life with a focus on user-centric functionality and aesthetics.
- **Development Environment:** For the development environment, Visual Studio Code is utilized as the primary code editor, offering a robust and customizable platform for efficient coding. Git is employed for version control, enabling effective management of source code changes, collaboration, and maintaining a reliable development workflow. Additionally, Postman is used for API development and testing, providing a powerful tool for creating, debugging, and documenting APIs to ensure seamless backend integration.
- **Frontend Development:** For the front-end development environment, Next.js serves as the foundation, offering a powerful framework for building fast, scalable, and server-rendered applications. The design is enhanced with Shadcn UI, providing reusable, accessible components that streamline the development of elegant interfaces. Tailwind

CSS is used for styling, ensuring a utility-first approach that accelerates design consistency and responsiveness. State management is handled with Redux, enabling efficient data flow and component interaction.

- **Backend Analysis:** For the backend development environment, MongoDB is used as the database, leveraging Mongoose for managing data models and ensuring smooth interactions with the database. Next.js API routes handle server-side logic, allowing seamless integration between the frontend and backend. For authentication, NextAuth.js with Google Provider enables secure, token-based user login. JWT is used for managing session authentication, ensuring a secure and scalable way to handle user sessions. Socket.io is implemented for real-time communication, providing live updates and interactions between clients and the server. Additionally, Cloudinary is utilized for storing media assets like images and videos, ensuring fast and reliable media delivery. Zod is incorporated for server-side data validation, enhancing security and integrity by ensuring all data meets required schemas before processing.

CHAPTER 2:
LITERATURE REVIEW

The "Recovery Ally" project application focuses on post-surgical recovery through tailored exercises, resources, and support for patients and therapists. Its development is informed by studies on digital healthcare solutions, rehabilitation technologies, and cloud-based systems for personalized care delivery. This review explores the relevant literature supporting the project's core functionalities, including remote patient monitoring, therapist-patient collaboration, and scalable architecture.

- **Digital Health and Rehabilitation Technologies**: Telehealth platforms have shown to improve patient engagement and adherence through real-time communication tools, enabling continuous monitoring and support. Additionally, incorporating gamification and virtual feedback into rehabilitation exercises has been proven to enhance patient motivation and provide engaging therapy experiences, improving recovery outcomes.
- **Role of Tailored Exercise Programs:** Personalized exercise plans have been highlighted as a key factor in improving recovery efficiency compared to generic programs. Tailoring these plans to an individual’s specific condition and injury type leads to better outcomes and reduced recovery times. Real-time updates and adjustments by therapists allow for continued progression, ensuring that exercises match the patient's evolving needs.
- **Real-Time Communication and Monitoring:** The use of real-time communication tools, such as Socket.io, enhances the interaction between therapists and patients by allowing instant feedback and updates. This fosters a stronger relationship between the two, enhancing the patient's trust and accountability in their recovery process. Real-time updates also enable more effective monitoring, which contributes to better adherence to personalized treatment plans.
- **Security and Privacy Considerations:** Security and privacy are crucial for healthcare applications, especially when dealing with sensitive patient information. The use of JWT for authentication ensures secure user access, while encryption and role-based access control help maintain confidentiality. Adhering to privacy regulations, such as the GDPR, is vital in protecting data and ensuring compliance with industry standards for patient confidentiality.

The Recovery Ally project incorporates proven concepts from digital health and rehabilitation technologies to deliver an effective and scalable solution for post-surgical recovery. By combining tailored exercise programs, real-time communication, and cloud-based scalability, the application aims to bridge the gap between therapists and patients, ensuring better outcomes.

CHAPTER 3:
METHODOLOGY

3.1 Software Development Life Cycle

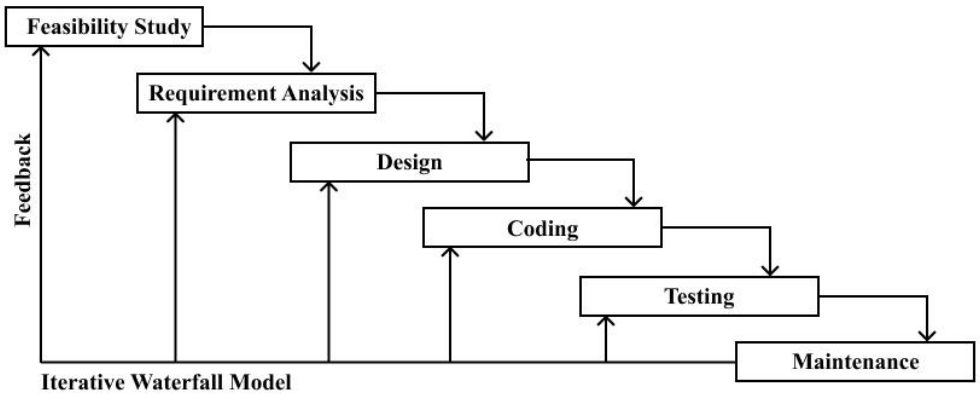

Figure 3.1.1: Iterative Waterfall Model

The Software Development Life Cycle (SDLC) for the Recovery Ally project follows the Iterative Waterfall Model, which ensures a linear flow of phases with feedback loops between adjacent phases and structured approach to development. Each phase must be completed before moving to the next, minimizing overlaps and promoting clarity. Here's what we will do in each phase:

3.1.1 Feasibility Study

A feasibility study in the Iterative Waterfall Model evaluates a project's viability by analyzing its requirements, scope, and constraints. It ensures technical feasibility by assessing the technology stack, risks, and infrastructure needs, while economic feasibility examines costs, benefits, and potential returns. Operational feasibility checks alignment with workflows, user acceptance, and training needs, and legal feasibility ensures compliance with regulations. Lastly, schedule feasibility evaluates timelines and dependencies to confirm the project's achievability. This study determines whether the project is practical and worth pursuing, guiding its progression to subsequent phases.

3.1.2 Requirements Analysis

For Recovery Ally, the requirements analysis phase will focus on understanding the needs of patients, therapists, and healthcare providers. This phase will involve gathering insights on essential features such as personalized recovery plans, exercise tracking, real-time communication between patients and therapists, progress monitoring, and secure data management. The goal is to document these requirements in detail to ensure that the platform meets the specific needs of post-surgical recovery, offering tailored exercises, seamless communication, and effective tracking of recovery progress.

3.1.3 System Design

We will design the architecture of the platform, including database schema, user interface layouts, and API endpoints. This phase includes creating wireframes for the frontend and defining backend logic.

3.1.4 Implementation (Frontend & Backend)

Development of the platform begins here:

- Frontend: Build the user interface using Next.js for responsiveness and simplicity.
- Backend: Develop APIs and database management using Node.js and MongoDB to ensure seamless data handling.

3.1.5 Testing

Conduct thorough testing of the platform, including unit tests, integration tests, and user acceptance testing. The focus will be on ensuring data integrity, responsiveness, and error handling.

3.1.6 Deployment

Deploy the platform to a hosting service like Vercel or AWS. This phase includes configuring the domain, setting up SSL certificates, and ensuring the application runs smoothly in a live environment.

3.1.7 Maintenance

After deployment, we will address user feedback, resolve bugs, and roll out updates for new features or system improvements.

3.2 System Design

The system design for the **Recovery Ally** platform focuses on building a robust, scalable, and user-friendly web application to support post-surgical recovery. This section outlines the overall architecture of the system, including how the frontend, backend, and database interact to deliver personalized recovery plans, real-time therapist-patient communication, exercise tracking, and progress monitoring. The design ensures seamless data handling, security, and scalability to meet the needs of both patients and therapists while maintaining an intuitive and responsive user experience.

3.2.1 System Components and Their Roles

- Frontend (Next.js): The frontend of Recovery Ally will serve as the user interface, allowing patients and therapists to easily manage recovery plans, track exercises, and monitor progress. It will be built using Next.js, a powerful framework that ensures fast rendering and a seamless user experience across various devices. The frontend will interact with the backend through API calls to retrieve and display personalized recovery data, real-time communication updates, and exercise progress in an intuitive and efficient manner.
- Backend (Node.js): The backend of Recovery Ally will be responsible for handling requests from the frontend, processing business logic, and interacting with the

MongoDB database to store and retrieve patient and therapist data. Developed in Node.js, the backend will efficiently manage multiple concurrent requests, ensuring smooth operation even as the user base grows. The backend will also include API endpoints to manage personalized recovery plans, exercise tracking, real-time communication, and user authentication, ensuring secure and seamless interactions between patients and therapists.

- Database (MongoDB): MongoDB will be used as the database solution for Recovery Ally due to its flexibility and ability to handle diverse and evolving data structures. It will store collections such as Patient Data, Recovery Plans, Exercise Logs, and User Profiles, allowing for easy access and updates. The database will ensure that all critical recovery-related information is securely stored and can be efficiently retrieved by both the frontend and backend as needed, supporting real-time updates and seamless communication between patients and therapists.
- Data Flow and Communication: The frontend of Recovery Ally will send HTTP requests (such as GET, POST, PUT, DELETE) to the backend through RESTful APIs. The backend, after processing these requests, will interact with the MongoDB database to fetch or update data related to recovery plans, exercise logs, patient progress, and therapist communications. The frontend will then display the updated data to the user, providing real-time insights into recovery progress, personalized exercise recommendations, and ongoing interactions between patients and therapists, ensuring a seamless and dynamic user experience.

3.2.2 System Architecture Diagram

The system architecture for "Recovery Ally" revolves around a clear interaction between its core components: the Frontend, Backend, and External Services. On the Frontend, patients, therapists, and admins interact through a web app built using Next.js, with responsive UI handled by Tailwind CSS and Shadcn UI. Patients can view and track recovery plans, while therapists create and manage those plans, and admins oversee system operations. The Backend, powered by Node.js and Express, handles API requests, user authentication (via NextAuth.js and JWT), and real-time communication through Socket.io for chat and notifications. Data storage and retrieval are managed using MongoDB, with Mongoose for schema validation. Additionally, external services like Cloudinary are integrated for storing media files (exercise videos, images), and Vercel handles cloud

deployment for scalability. This setup ensures a smooth, secure, and scalable experience for all users while allowing therapists to provide personalized care.

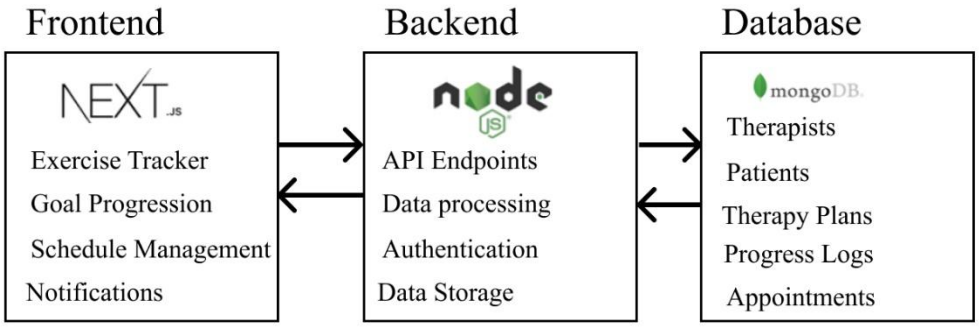

Figure 3.2.1: System architecture diagram

3.3 Algorithm

1. Start
2. Initialize System
    - Load the homepage.
    - Connect to the database (User Database, Exercise Database, Feedback Database).
3. User Registration and Login
    - If User is not registered:
        1. Prompt for details: Name, Email, Password, Role (Patient/Physiotherapist).
        2. Validate input and store in the User Database.
    - If User is registered:
        1. Authenticate using Email and Password.
        2. Redirect to Dashboard based on Role.
4. Dashboard Options
    - If User Role is "Patient":
        1. Display options:

Book Appointment.

View Exercise Plan.

Submit Session Feedback.

Track Progress.

- If User Role is "Physiotherapist":
    1. Display options:

Manage Appointments.

Create/Update Exercise Plans.

View Feedback.

Update Patient Progress.

5. Appointment Booking (Patient)
    - Input: Select a date, time, and available physiotherapist.
    - Validate availability in the database.
    - Confirm booking and notify the physiotherapist.
    - Store details in the Appointment Database.
6. Exercise Plan Generation (Physiotherapist)
    - Input: Patient details and specific requirements.
    - Select exercises from the Exercise Database.
    - Generate and save the personalized exercise plan.
    - Notify the patient of the new plan.
7. Feedback Submission (Patient)
    - Input: Session rating and comments.
    - Store feedback in the Feedback Database.
    - Notify the physiotherapist of the feedback.
8. Progress Tracking
    - Patient:
        1. View progress reports (sessions completed, exercise adherence, improvement metrics).
    - Physiotherapist:
        1. Update progress (session notes, improvements observed).
        2. Generate reports for patients.
9. Admin Functions (Optional)
    - Manage users (add/remove).
    - Oversee appointments, feedback, and database health.
    - Generate platform usage analytics.
10. End User Session
    - Allow the user to log out and secure their session.

11. End

3.4 Flowchart

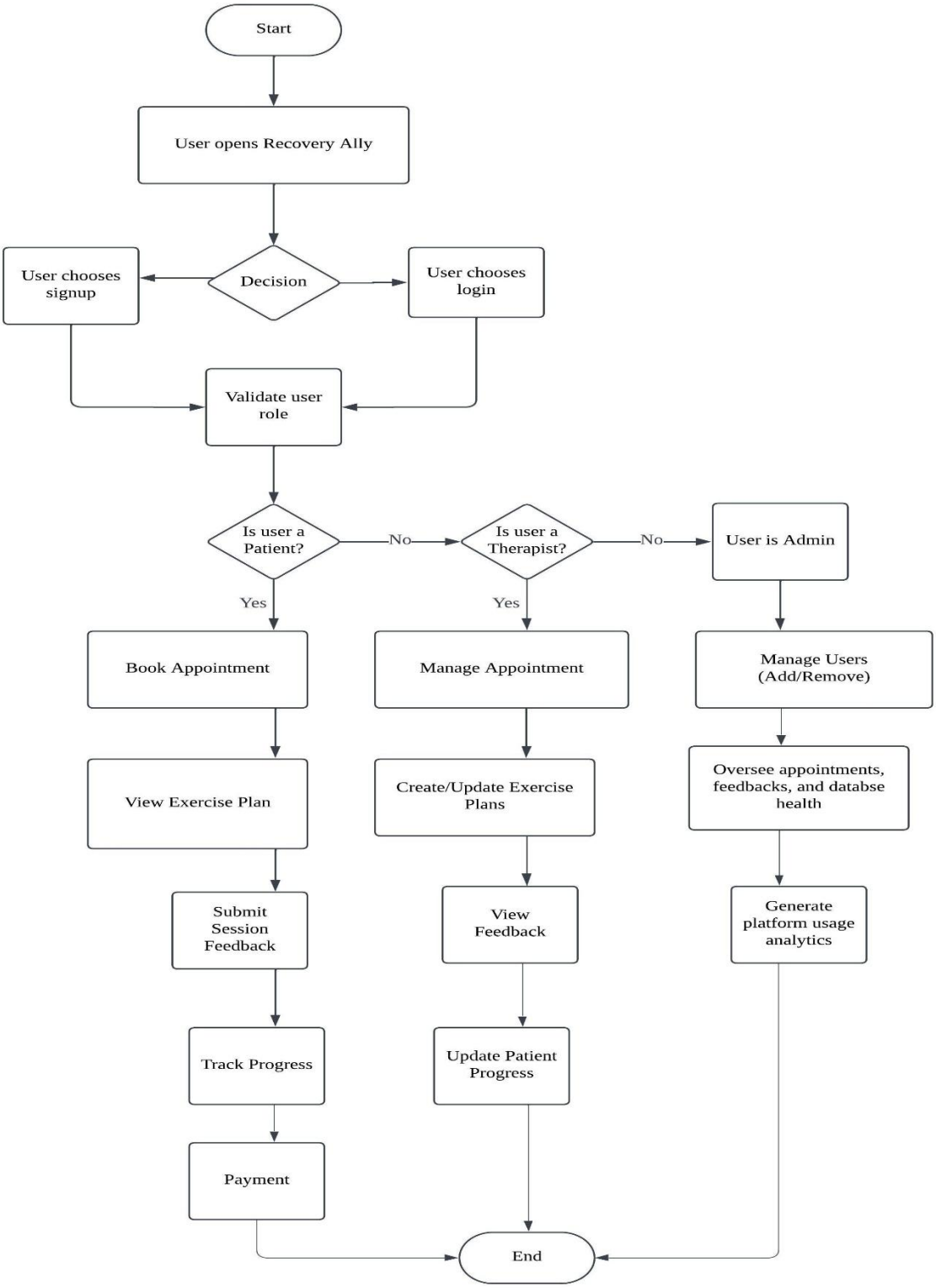

Figure 3.4.1: Flowchart

3.5 ER Diagram

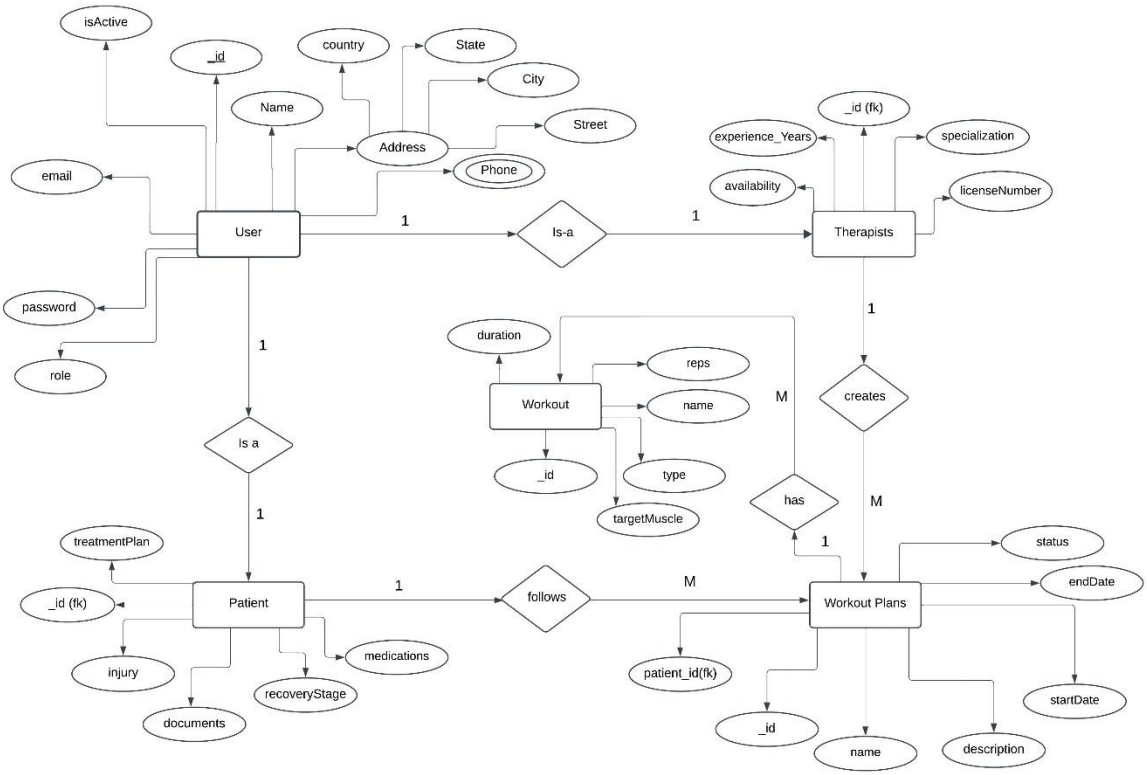

Figure 3.5.1: ER Diagram

3.6 Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) visually represents how data moves within the Recovery Ally. It shows the flow of information between the external entities (users) and the internal system (frontend, backend, and database).

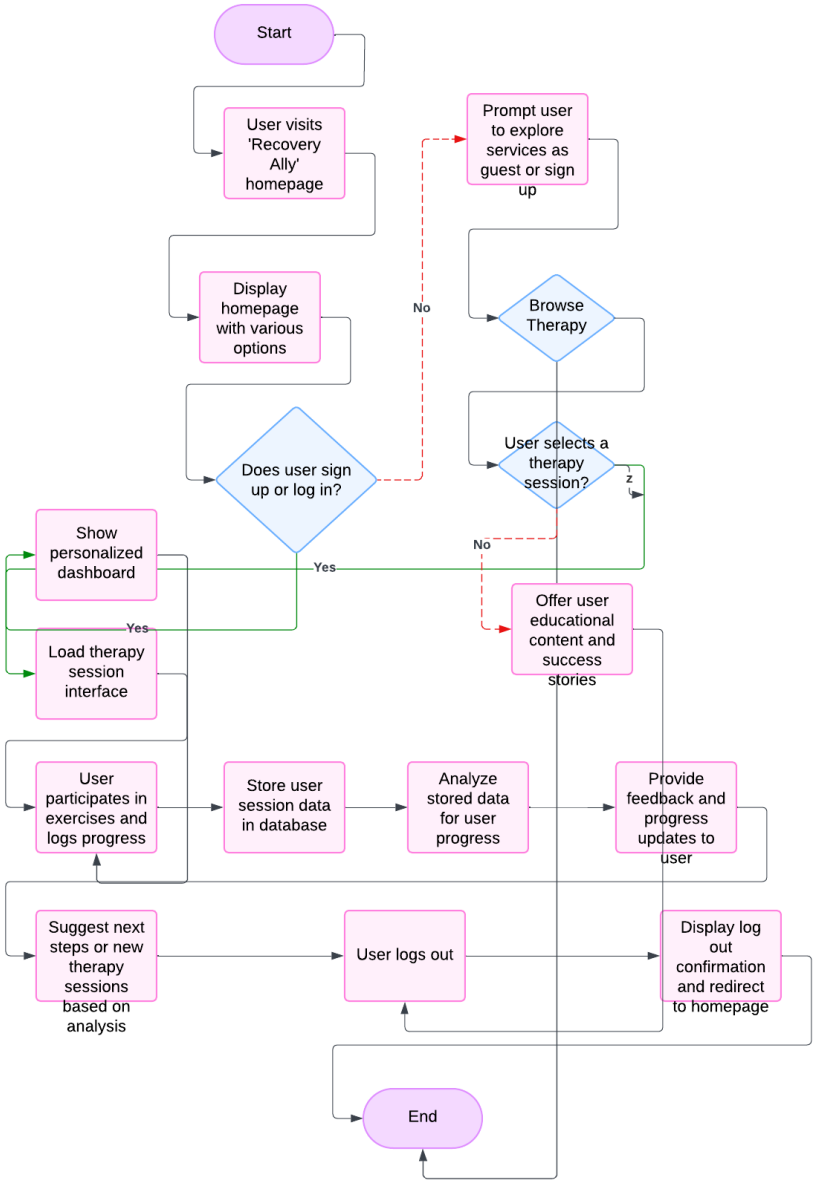

Figure 3.6.1: Data Flow Diagram

3.7 Use Case Diagram

The Use Case Diagram for Recovery Ally illustrates the different actors interacting with the system and the use cases (specific functionalities or actions) that the platform supports. These use cases represent various operations related to post-surgical recovery, such as managing recovery plans, tracking exercises, monitoring progress, and enabling real-time communication between patients and therapists. The diagram helps visualize the core functionalities of the platform, including personalized exercise recommendations, progress tracking, and secure interactions between patients and healthcare providers.

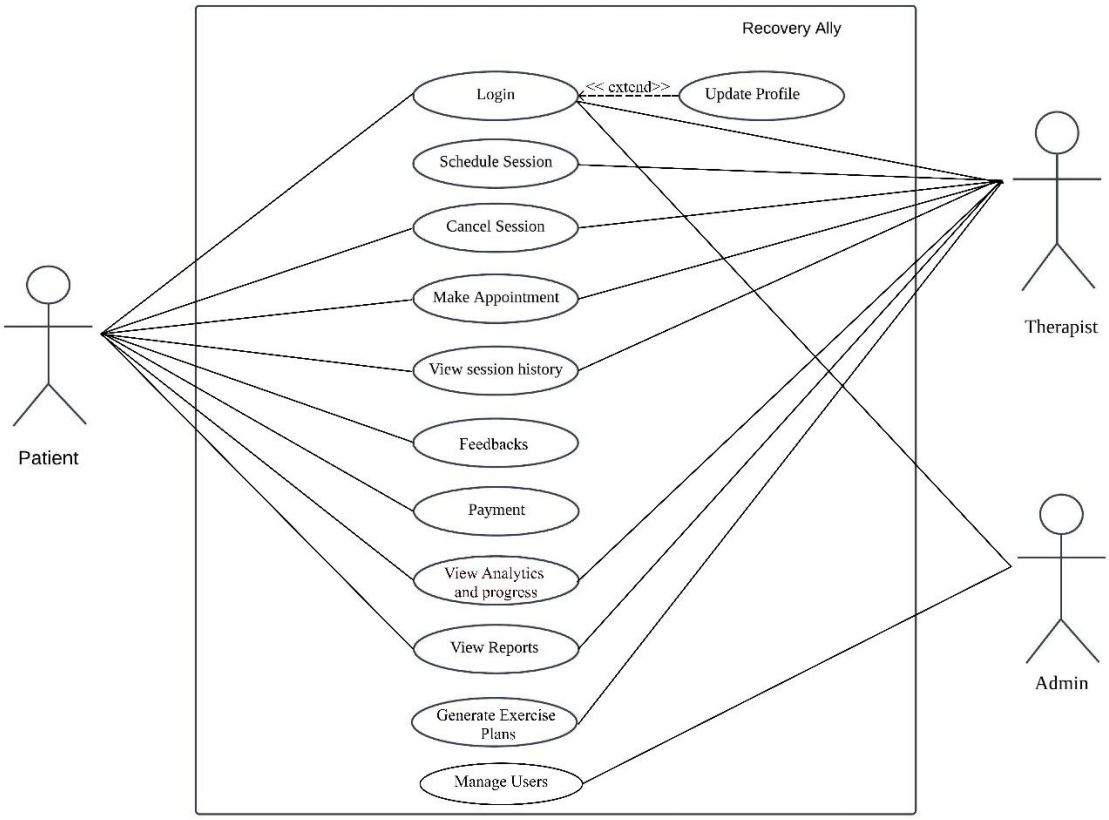

Figure 3.7.1: Use Case Diagram

3.8 Sequence Diagram

The Sequence Diagram for Recovery Ally illustrates the interactions between different actors and the system over time, focusing on the sequence of operations performed during specific functionalities. This diagram captures the dynamic behavior of the system, highlighting how various components interact to execute tasks related to post-surgical recovery.

Key scenarios visualized in the sequence diagram include:

- **Creating Recovery Plans**: Shows the interactions where therapists input patient details, design personalized exercises, and save the plan in the system.
- **Tracking Exercises**: Depicts the flow where patients log their activities, and the system updates their progress and provides feedback.

- **Monitoring Progress**: Highlights the sequence where the system generates progress reports based on tracked data, which are then reviewed by therapists and shared with patients.
- **Real-Time Communication**: Illustrates the steps involved in establishing secure messaging between patients and therapists, ensuring effective communication.

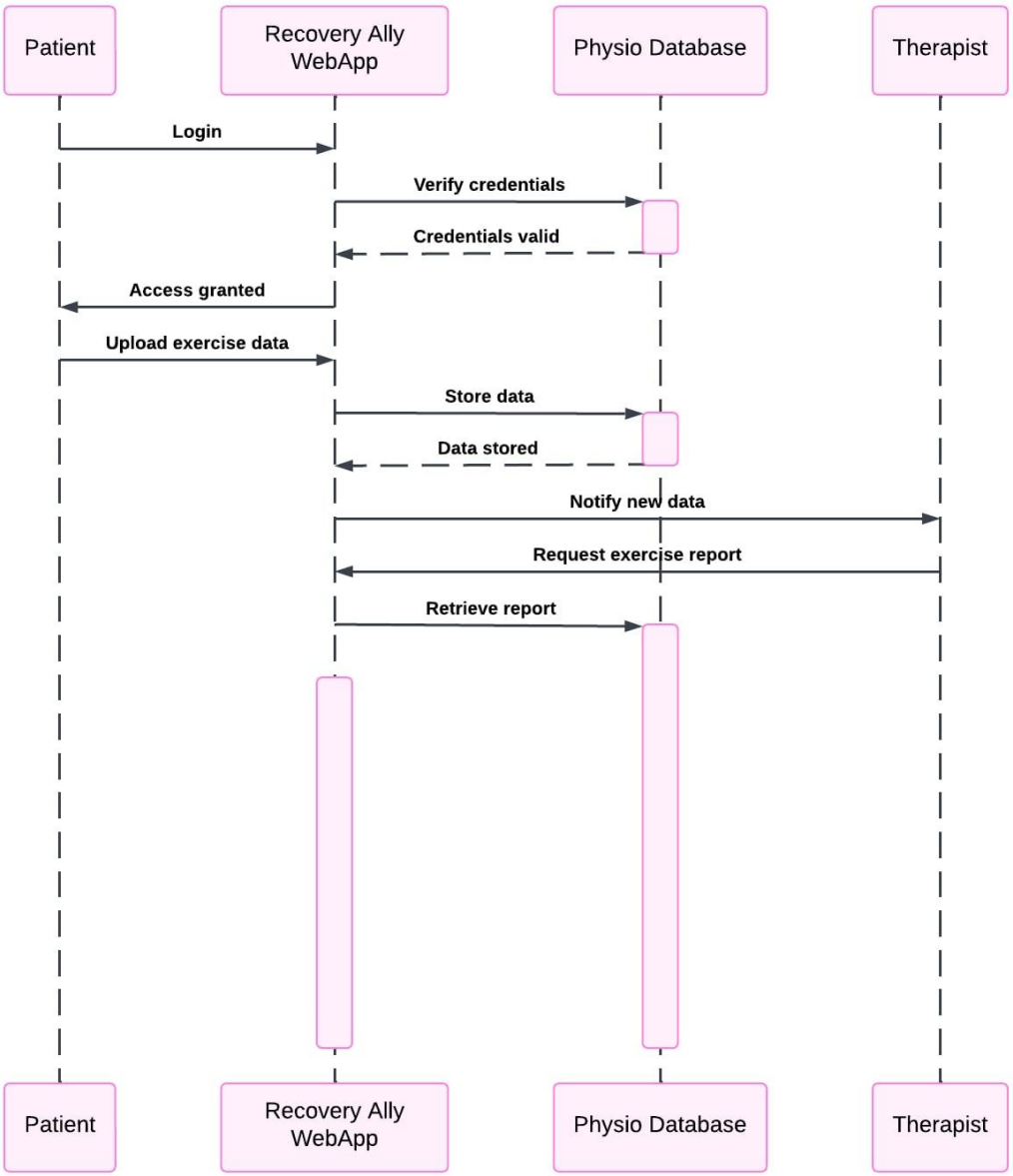

Figure 3.8.1: Sequence Diagram

3.9 Progress

3.9.1 Requirement Analysis and Planning

- Clearly defined project objectives, features, and deliverables tailored to therapists and patients.
- Conducted needs assessment through discussions with physiotherapists and patients to prioritize features such as recovery plans, workout displays, and tracking.
- Mapped out user personas and workflows to align the application with real-world scenarios.

3.9.2 UI/UX Design

- Designed interactive wireframes and prototypes using tools like Figma.
- Conducted user testing sessions with therapists and patients to gather feedback and refine the user interface for accessibility and ease of use.
- Focused on creating a responsive and mobile-friendly design for a seamless experience across devices.

3.9.3 Authentication and Role Management

- Implemented secure authentication for therapists and patients using email and password.
- Added features for role-based access:
- Integrated session management and password recovery functionality.

3.9.4 Recovery Plan Creation

- Developed a module for therapists to create personalized recovery plans for each patient.
- Included options for therapists to add structured workouts, including descriptions, multimedia (e.g., videos or images), and recommended frequency.
- Enabled customization of plans to suit individual patient recovery goals and therapy progress.

3.9.5 Workout Display and Tracking

- Built a user-friendly interface for patients to view assigned workouts, including step-by-step instructions and visual aids.
- Enabled patients to track their daily workout completion and provide feedback on exercises.
- Added progress indicators for patients to monitor their adherence to recovery plans.

3.10 Current State of the Project

- Created a basic interface for therapists to view and manage patient profiles and recovery plans.
- Enabled therapists to create and assign personalized recovery plans to patients, including work out details.
- Designed a patient interface to display assigned workouts with instructions and visual aids (e.g., images/videos).
- Secured role-based authentication for therapists and patients, ensuring access to appropriate features.

3.11 Core Features

3.11.1 Role Based Authentication

- **Therapist**: Can manage patient therapy sessions, create recovery plans, and track progress.
- **Patient**: Can view their progress, schedule appointments, track their recovery milestones, see workout videos, and communicate securely with their therapist.
- **Admin**: Has full control over user roles, platform settings, and system-wide reports.

3.11.2 Exercise Guidance

- Library of guided exercises with instructional videos.
- Assigned exercise routines with frequency and intensity tracking.
- Notifications for missed or completed exercises.

3.11.3 Therapy Session Tracking

- Real-time logging of therapy sessions.
- Session notes and progress assessments.
- Customizable therapy plans.

3.11.4 Progress Monitoring

- Visual progress charts for milestones and goals.
- Daily, weekly, and monthly insights on patient activities.
- Recovery progress percentage indicators.

3.12 Next Steps

3.12.1 Progress Monitoring

- Visual progress charts for milestones and goals.
- Daily, weekly, and monthly insights on patient activities.
- Recovery progress percentage indicators.

3.12.2 Recommendations Portal

- Use of AI/ML for suggesting services, exercises, or products based on the data.
- Allows users to update their condition and receive ongoing recommendations.
- Allows Collects user data (e.g., age, type of injury, symptoms, pain level, activity level)
- Connect patients to the most suitable therapists based on the injury type or grade of the injury.

3.12.3 Feedback portal

- Allow patients to share their experiences after using the services.
- Allow patients to rate the effectiveness of workout plans and therapists.

3.12.4 Push Notification Integration

- Implement push notifications to send personalized reminders to patients about upcoming therapy sessions, workout plans, recovery milestones, or subscription renewals, helping them stay on track.

- Notify patients instantly about important updates, such as new available workout plans, therapist recommendations, or changes to their schedule, enhancing user engagement.

3.12.5 Payment Gateway Integration

- Integrate a secure payment gateway (e.g., Esewa or Khalti) to handle transactions for premium features, therapy sessions, and subscription models, ensuring a smooth checkout experience within the app.
- Offer multiple payment options, including credit/debit cards, and implement recurring subscription plans with automatic renewals, providing users with flexible payment models for long-term use.

CHAPTER 4:
EPILOGUE

4.1 Expected Output

The expected output of the **Recovery Ally** project is a platform that will help patients and therapists efficiently manage post-surgical recovery. The expected outputs include:

- **Personalized Recovery Plans**: Therapists create customized recovery plans for patients based on their specific injury type and recovery progress.
- **Real-Time Communication:** Therapists and patients communicate via real-time messaging and video calls for continuous support and feedback.
- **Progress Tracking:** Track patients' progress on exercises, milestones, and recovery goals, with regular check-ins from therapists.
- **User-Friendly Interface:** The app is designed with an intuitive, easy-to-navigate interface to accommodate users of all ages and technical skills.
- **Integrated Educational Resources:** Patients have access to a library of recovery tips, nutritional advice, and injury prevention techniques.

4.2 Budget Analysis

The table below provides a breakdown of the estimated costs involved in the development, deployment, and maintenance of the Recovery Ally platform.

Asset/Work	Estimated Cost (NPR)
Software Development	15,000
UI/UX Design	5,000
Cloud Hosting and Storage	4,000/year
Domain Name and SSL Certificates	5,000/year
Marketing and Launch Costs	4,000
Server and Database Maintenance	10,000/year
Support and Updates	8,000/year
Contingency Costs	5,000
Total Estimated Budget	**56,000 NPR**

Table 4.2.1: Budget Analysis

4.3 Work Schedule

The Work Schedule for the Recovery Ally project outlines the timeline for key activities and milestones involved in the system's development, from requirements gathering to deployment and maintenance. Below is a structured Gantt Chart that visualizes the sequence of tasks and the timeline over which these tasks are scheduled.

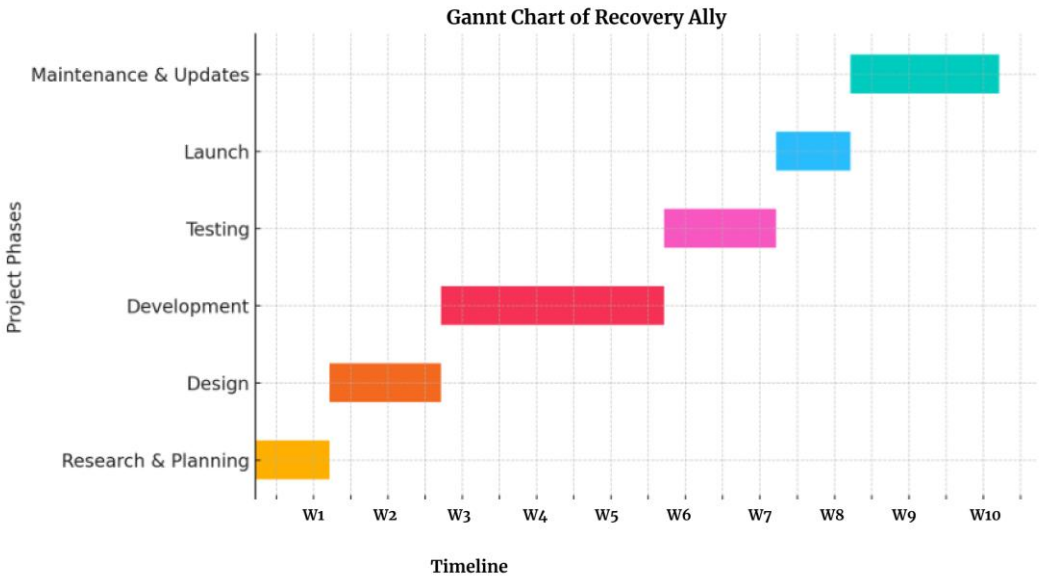

Figure 4.3.1: Gantt Chart

CHAPTER 5:
REFERENCES

[1] R. e. a. Bashshur, "The Empirical Foundations of Telemedicine Interventions in Chronic Disease Management," *Telemedicine and e-Health,* vol. 22, no. 5, pp. 396-409, 2016.

[2] R. &. L. B. Proffitt, "Considerations in the Efficacy and Effectiveness of Virtual Reality Interventions for Rehabilitation," *Frontiers in Neurology,* vol. 6, p. 147, 2015.

[3] O. M. e. a. Giggins, "Wearable Sensors in Rehabilitation and Healthcare Settings," *Smart Healthcare Systems,* vol. 1, no. 2, pp. 82-94, 2013.

[4] M. e. a. Tousignant, "Telerehabilitation for Knee Osteoarthritis and Chronic Respiratory Conditions," *Journal of Telemedicine and Telecare,* vol. 17, no. 5, pp. 256-261, 2011.

[5] A. M. H. Kuo, "Opportunities and Challenges of Cloud Computing in Healthcare," *Journal of Medical Internet Research,* vol. 13, no. 3, p. e67, 2011.

[6] R. &. L. L. Zhang, "Security Models and Requirements for Healthcare Application Clouds," *Proceedings of IEEE Cloud,* pp. 101-106, 2010.

[7] B. e. a. Martínez-Pérez, "Methodological Review of mHealth Intervention Effectiveness," *BMC Medical Informatics and Decision Making,* vol. 15, p. 28, 2015.

[8] W. &. R. V. Raghupathi, "Big Data Analytics in Healthcare: Promise and Potential," *Health Information Science and Systems,* vol. 2, no. 3, pp. 1-10, 2014.

[9] E. Commission, "General Data Protection Regulation (GDPR)," *European Union*, pp. 1-88, 2018.