

AMMAN ARAB UNIVERSITY

جامعة عمان العربـيــــــة

Faculty of IT

**كلية تكنولوجيا المعلومات**

HealNest

Students:

|  |
| --- |
|  |
| Sara Abuodeh | | 202210418 |

*Supervisor:*

Dr. Mohamad bani baker

*Amman – Jordan*

*2024/2025*

Table of Contents

[Chapter One](#CHAPTER_ONE)

[INTRODUCTION](#INTRODUCTION) 1

[1.1 Problem Statement](#_1.1_Problem_Statement) 1

[1.2 Business case](#_1.2_Business_Case) 1

[1.3 Aim and Objectives](#_1.3_Aim_and) 1  
[1.4 Project Scope](#_1.4_Project_Scope) 1

[Chapter Two](#CHAPTER_TWO)

[Project Planning and Requirements](#PROJECT_PLANNING_AND_REQUIREMENTS) 2

[2.1 Project Plan and Schedule](#_2.1_Project_Plan) 2

[2.1.1 Gantt Chart](#_2.1_Project_Plan) 2

[2.2 Functional Requirements](#_2.2_Functional_Requirements)  2

[2.3 Non-Functional Requirements](#_2.3_Non-Functional_Requirements)  2

[2.4 Software Development Methodology (SDLC Model)](#_2.4_Software_Development) 2

[Chapter Three](#CHAPTER_THREE)

[SYSTEM ANALYSIS AND DESIGN (Initial Phase)](#SYSTEM_ANALYSIS_AND_DESIGN) 3

[3.1 System Use Case Diagram](#_3.1_System_Use) 3

[3.2 UI Mockups](#_3.2_User_Interface) 3

Chapter One

Introduction

### 1.1 Problem Statement

In veterinary clinic ,daily tasks include recording animal data, following up on vaccination schedule ,tracking diet and vitamins ,and managing inventory of medical and nutrition supplies, in the absence of an organized digital system, the traditional method such as paper or Excel spreadsheet are relied upon , , leading to : difficult in quick accessing to animal data, forgotten important information (such as vaccination or examination schedules),wasted staff time and effort on manual task documentation and difficult on tracking of inventory consumption leading to sudden shortages of essential materials.

In a fast-paced, multi-tasking environment like a veterinary clink, these problems not only impact work efficiency, but can also lead to unintended failure to provide proper care to animals and weaken the clinic’s internal organization.

### 1.2 Business Case

With the increasing number of pets and rising awareness of animal healthcare, veterinary clinics need an efficient digital system to manage animal records and services. This project aims to develop a comprehensive Veterinary Clinic Management System to register animals, track their medical history, and manage operations such as check-in, check-out, hospitalization, vaccinations, and treatments.

### 1.3 Aim and Objectives

The main aim of this project is to develop an integrated digital system for veterinary clinics that enables efficient registration, tracking, and management of animal records, including medical history, check-in/check-out status, hospitalization, medications, and vaccinations — all within a secure and user-friendly interface.

**The key objectives of this project are:**

* Design a secure login interface:  
  To ensure that only authorized users can access and manage the system.
* Develop a comprehensive animal database:  
  To store detailed information for each animal, including name, species, age, gender, and health condition.
* Implement full CRUD functionality (Create, Read, Update, Delete):  
  To allow staff to efficiently manage animal records as needed.
* Track the status of each animal:  
  Including whether the animal is currently in the clinic, hospitalized, or discharged.
* Record and monitor medications and vaccinations:  
  To maintain a complete treatment history for each animal and optionally include alerts for future schedules.
* Build an interactive dashboard:  
  To display all animal records in an organized layout with filtering and search capabilities.
* Improve clinic workflow efficiency:  
  By replacing manual record-keeping with a digital system, making information access faster and more accurate.

### 1.4 Project Scope

To design and implement a digital management system for a veterinary clinic that allows staff to manage animal records, track health status, medications, vaccinations, and streamline daily operations like check-in, discharge, and hospitalization.

* User Authentication: Login system for clinic staff (e.g., admin, receptionist, vet).
* Animal Management Module: Add new animal records with details (name, species, age, gender, owner info, etc.), Edit animal details including health condition, medications, and vaccination status and Delete animal records if necessary.
* Vaccination and Medication Records: Input and update vaccination history, Track medication details and treatment plans and Optional alerts/reminders for upcoming vaccinations.
* Animal Status Tracking: Track whether the animal is currently: Inside the clinic., Hospitalized (overnight stay) and Discharge.
* Dashboard / Data View Page: Display list of all animals with filters/search functionality and show key information in a structured and readable format.
* Basic UI Interface: User-friendly interface for staff to navigate between pages (Add/Edit/Delete/View).

Chapter Two

Project Planning and Requirements

### 2.1 Project Plan and Schedule

The PackTrack project will be developed over several stages, following a structured timeline to ensure efficient progress and timely delivery. The project team has divided the tasks across planning, analysis, design, implementation, testing, and deployment phases.

Table 2.1 PackTrack Gantt Chart



### 2.2 Functional Requirements

FR1 Log in: the system must allow users to log in using a valid username and password.

FR2 Animal Management: user can add a new animal with some details (name, owners’ type, age, gender).

FR3 Medical Recorders and Vaccination Tracking: user can add vaccination records for each animal like vaccine name, next due date.

FR4 Inventory Management: users can manage and track stock for food, vaccines, vitamins, medical supplies.

FR5 Search and filtering: users can search animal, inventory items and medical records using keywords.

FR6 Reports: users can export animal data

FR7 Owner Management: Manage pet owners' information including name, contact details, and ownership of animals.

FR8 Appointment Scheduling: Schedule, view, and manage appointments for animal check-ups, vaccinations and other services.

FR9 Billing Systems: Generate and manage invoices for services provided.

2.3 Non-Functional Requirements

NF1 Usability: User interface should be intuitive and user-friendly.

NF2 Security: Passwords should be securely hashed and stored, all sensitive data should be securely stored and protected

NF3 Reliability: System should be available 99% of the working time

NF4 Performance: System should respond to user actions within 2 seconds.

NF5 Maintainability: Codebase should be modular and well-documented.

NF6 Scalability: Application should be designed to handle growing data and users.

NF7 Portability: The system should work on desktops.

NF8 Language support: System should support Right-to-Left (RTL) languages like Arabic.

### 2.4 Software Development Methodology (SDLC Model)

For the development of PackTrack, we will follow the Incremental SDLC Model, which enables the system to be built and delivered in structured increments. This approach is well-suited for the project as it allows:

* Step-by-step feature development to ensure functionality is built gradually.
* Flexibility to update requirements based on feedback in each phase.
* Ongoing testing to maintain system reliability after every increment.

Each increment will follow a structured cycle comprising planning, implementation, testing, and integration of specific feature group (e.g., user authentication, dashboard, API integration, admin panel).

This approach minimizes complexity while ensuring a scalable and maintainable solution.



Figure 2.2 Incremental SDLC Model

Chapter Three

System Analysis and Design

### **3.1**System Use Case Diagram

The HealNest system supports admin roles to ensure efficient workflow and accurate data handling.

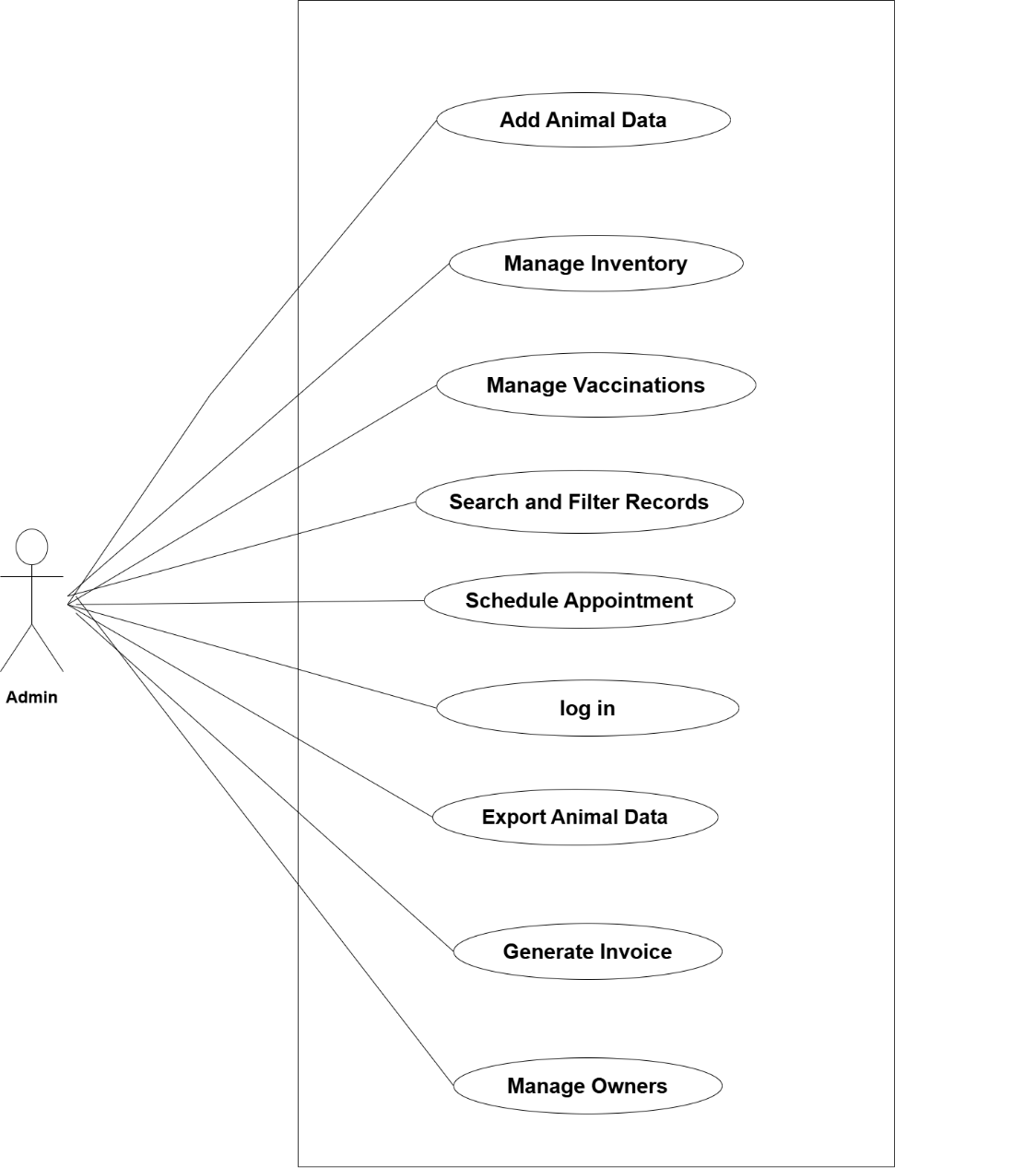


Figure 3.1 HealNest Use Case Diagram

### **3.2**User Interface Mockups

We designed basic wireframes for the most essential screens of PackTrack. The UI aims to be simple, clean, and user-friendly with a dashboard-style layout.

Figure 3.2.1 shows the login form used by users and admins to access the system. The login process is secured using a JWT-based authentication mechanism, ensuring that access to the dashboards is restricted to logged-in users only.

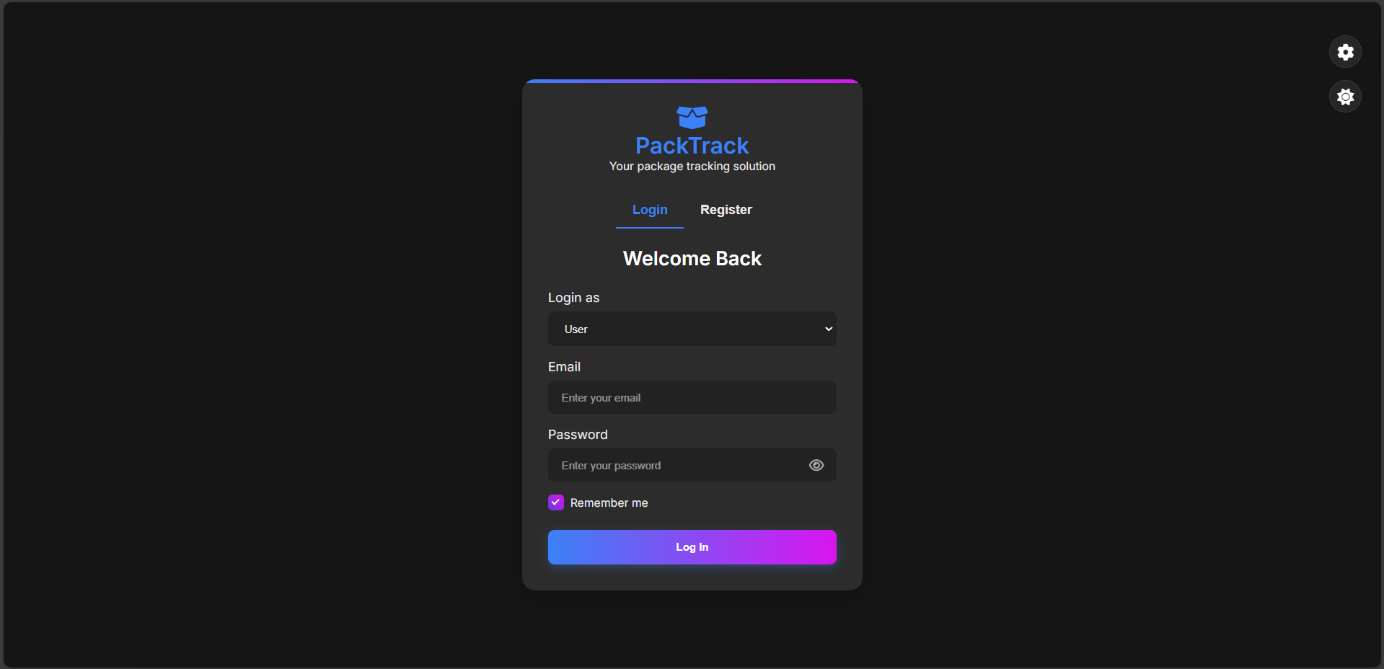


Figure 3.2.1: Login Form (JWT-Based Authentication)

Figure 3.2.2 displays the registration form. Users can sign up as either a User or an Admin. When selecting Admin, an additional field for Admin Secret Key appears. Only users with this key (provided by system owners or managers) can register as admins.

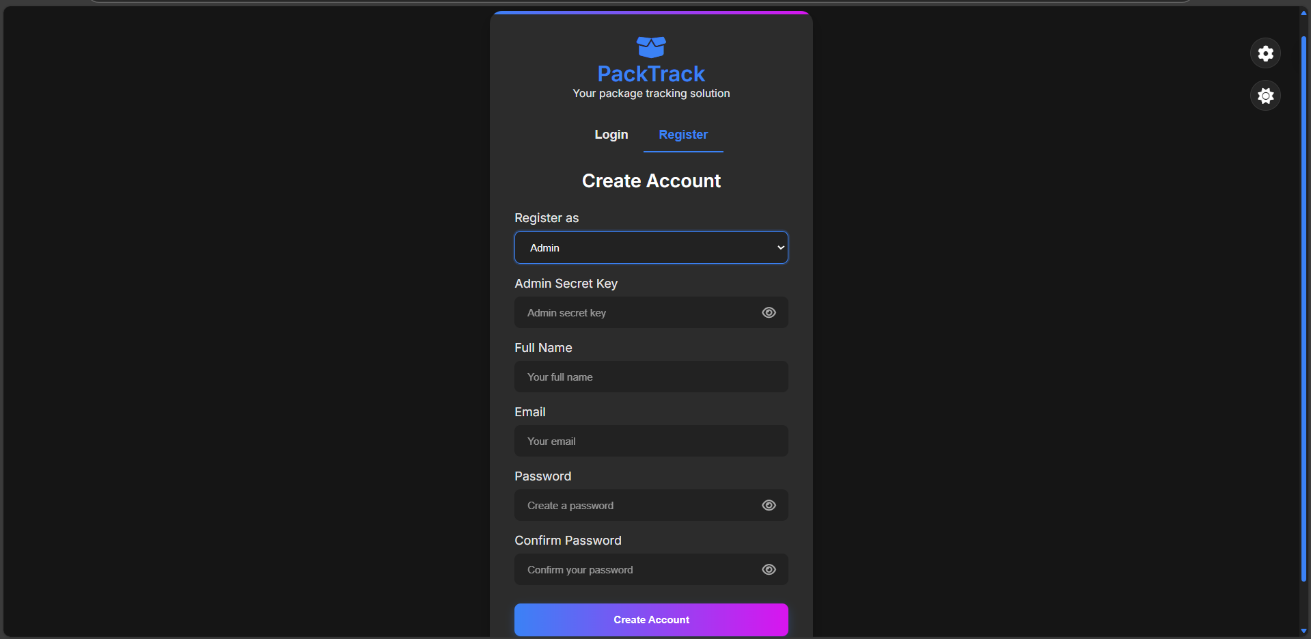


Figure 3.2.2: Registration Form with Role-Based Access

Figure 3.2.3 illustrates the User Dashboard, accessible exclusively to authenticated users, This dashboard provides an overview of package statuses for the current month, along with interactive data visualizations:

* Package Activity Chart (line chart): Tracks the number of received and delivered packages over a selected period (week, month, or year).
* Package Status Chart (donut chart): Displays the distribution of package statuses, including delivered, in transit, out for delivery, and delayed.

The dashboard also features an Add Delivery Form for submitting new package details and a Delivery Logs Table for tracking past deliveries. A navigation menu ensures seamless interaction with other sections of the application.



Figure 3.2.3: User Dashboard

Figures 3.2.4 and 3.2.5 present the Admin Panel, which is accessible exclusively to authenticated administrators. The panel provides a comprehensive dashboard for managing users, tracking package statuses, and monitoring system performance. Key functionalities include:

* User Management: View registered users, track their last login activity, and manage their accounts.
* System Statistics: Displays key metrics, like total users, total packages, active users, and delayed packages, with percentage-based trend indicators.
* Analytics & Reports: Interactive charts supporting data-driven decision-making.
* Activity Logs: Logs recent user actions, including login timestamps and packages status changes.

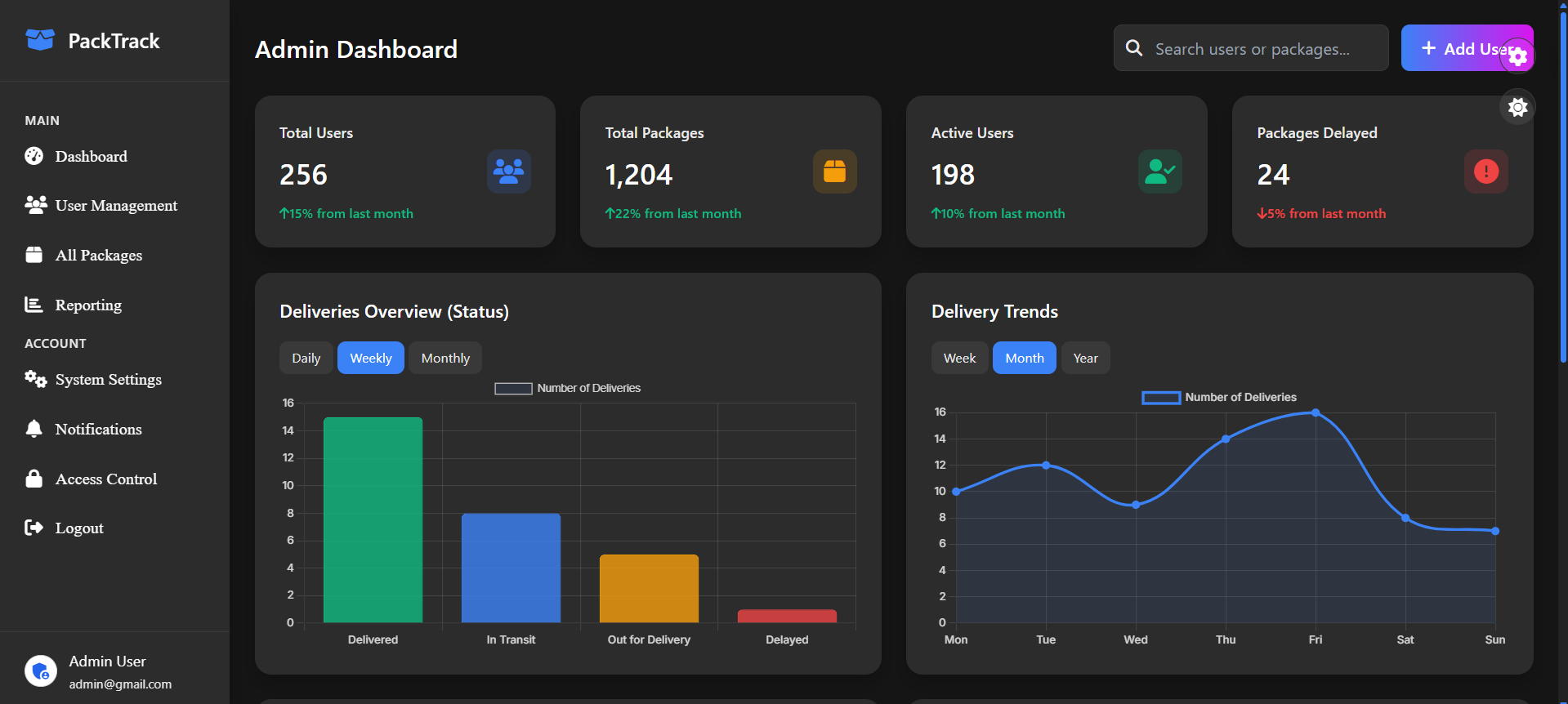


Figure 3.2.4: Admin panel Overview

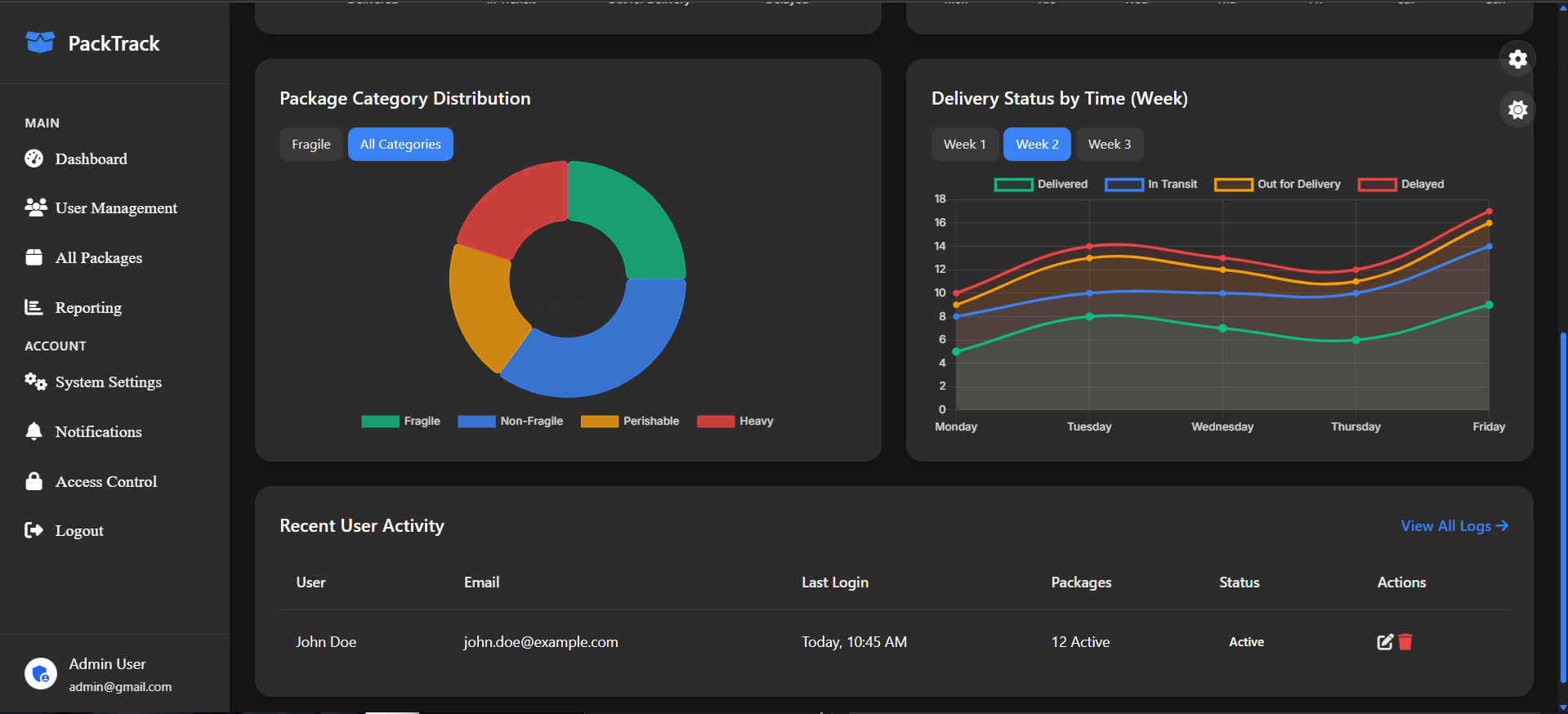


Figure 3.2.5: Admin panel User Management & Activity Logs

Figure 3.2.6 appears when a user tries to access a protected route without the required permissions (e.g., accessing the admin panel as a regular user). It enforces role-based access control and ensures only authorized users can view sensitive content.

Key Points:

* Security Enforcement: Blocks unauthorized access based on user roles validated through JWT tokens.
* Clear User Feedback: Friendly error message and provides quick navigation to login or dashboard.

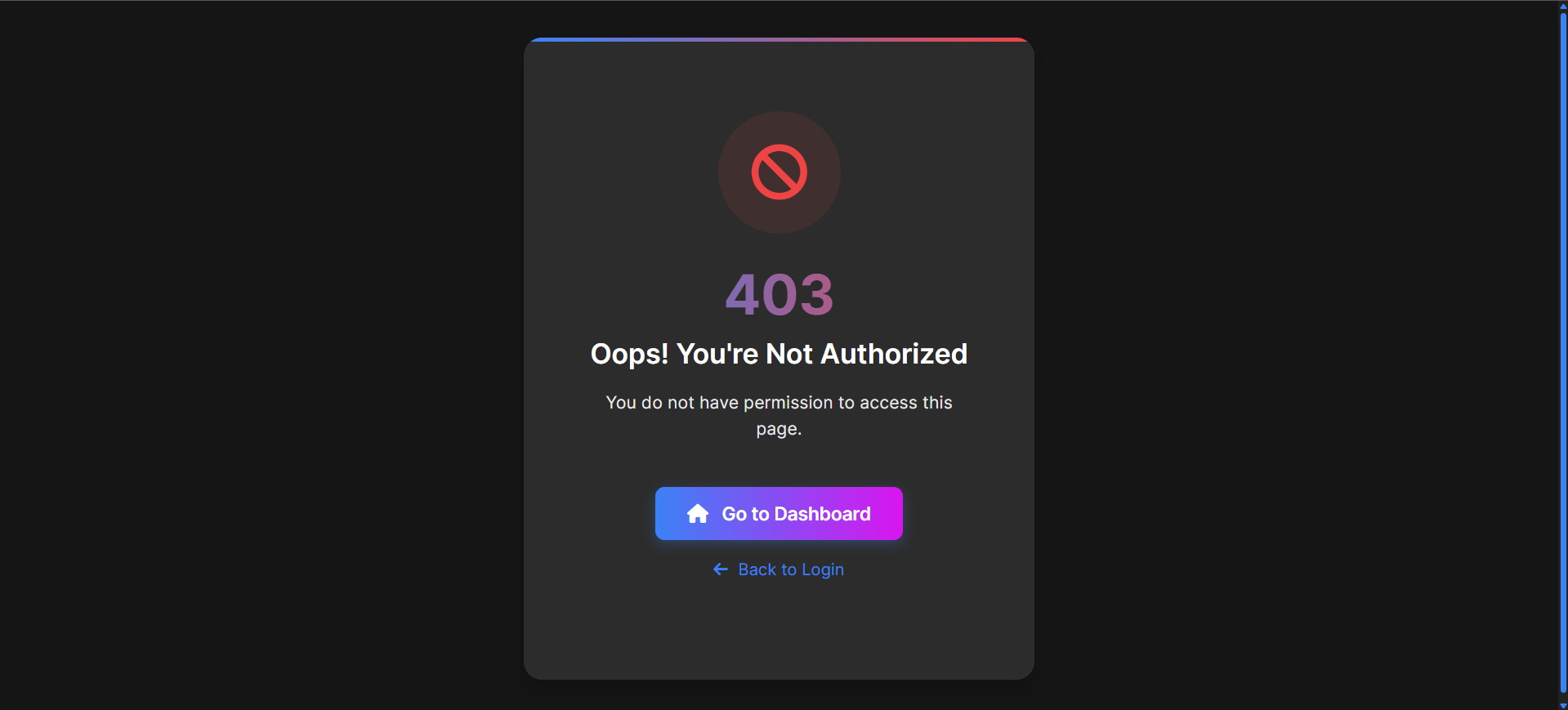


Figure 3.2.6: 403 Forbidden Access Page

Figure 3.2.7 is displayed when an unexpected issue occurs on the server side, preventing the requested operation from being completed.

Key Points:

* Error Handling: Captures server-side issues and displays an informative error message to users.
* User Experience: Offers options to navigate back to the dashboard or reload the page, ensuring users can recover from errors quickly.
* Security: Does not reveal sensitive server details, protecting the system from potential exploits.

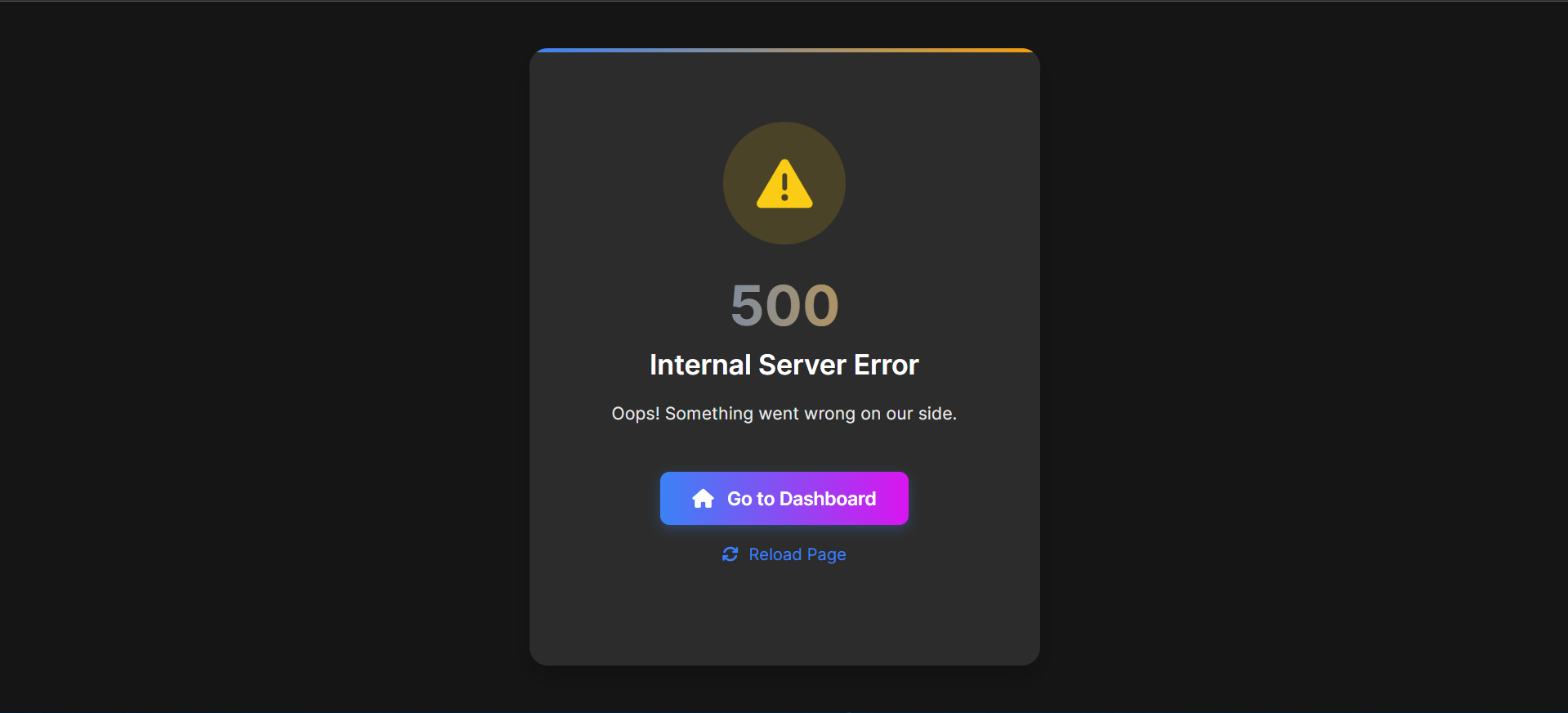


Figure 3.2.7: 500 Internal Server Error

Figure 3.2.8 enhances the user experience by displaying an animated loading screen during page transitions or while retrieving user data. With dynamic messages and helpful tips, it keeps users informed and engaged, effectively reducing perceived wait time.

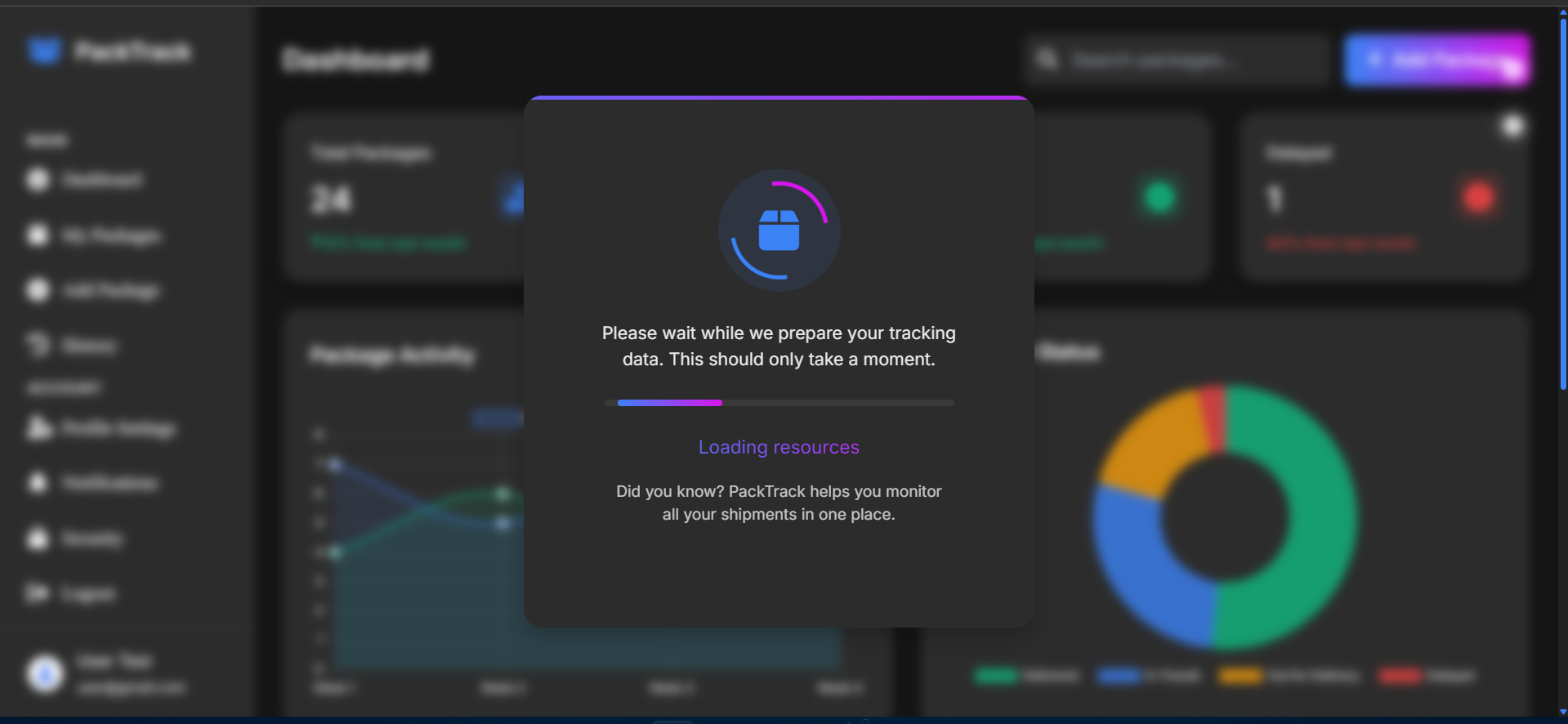


Figure 3.2.8: Loading Animation