  
Parcel Delivery Tracking

Requirement Specification Document

*Table of Contents*

[Overall Description 2](#_Toc450082834)

[Purpose 3](#_Toc733044274)

[Intended Audience 3](#_Toc478864872)

[Intended Use 3](#_Toc1158196149)

[Scope 3](#_Toc542258430)

[Project Backlog Item 4](#_Toc4908573)

[Entity Relationship Diagram 5](#_Toc2014481846)

[Table Descriptions 6](#_Toc476091898)

[System Features 8](#_Toc272663660)

[Functional Requirements 8](#_Toc1725398720)

[Non-Functional Requirements 8](#_Toc479060832)

[API Endpoint List 8](#_Toc521770857)

[Parcel Information Endpoints 8](#_Toc1649592591)

[Personnel Infromation Endpoints 9](#_Toc232231723)

[Delivery Information Endpoints 9](#_Toc701284568)

[Lesson Learned 9](#_Toc142552676)

# Overall Description

Parcel Delivery Tracking System is a comprehensive software application designed to streamline and enhance the logistics and delivery processes for a delivery company. This system is built to cater to a diverse clientele, including businesses, shops, warehouses, and individual customers, who require efficient parcel tracking and delivery services.

# Purpose

The purpose of the Parcel Delivery Tracking System is to provide a comprehensive and efficient software solution for managing parcel tracking and delivery logistics for a delivery company. This system aims to streamline the entire parcel lifecycle, from creation to delivery, while ensuring ease of use.

# Intended Audience

The intended audience for the Parcel Delivery Tracking System include:

* **Delivery Company**: The system is designed for the delivery company, enabling it to optimise its parcel management, delivery operations and client interaction.
* **Clients**: Businesses, Shops, Warehouses, and individual customers who require parcel delivery services are the primary end-users of the system. Clients can request parcel deliveries and track the status of their parcels.
* **Delivery Personnel**: Delivery drivers, managers, other personnel responsible for handling deliveries will use the system to access delivery details, update statuses.
* **Administrators**: System administrators will have access to administrative features for user managements.

# Intended Use

The intended use of the Parcel Delivery Tracking System is to:

* Facilitate clients in requesting parcel deliveries providing parcel details, and tracking parcel statuses.
* Allow delivery personnel to access delivery assignments, update delivery statuses, and manage their availability.

# Scope

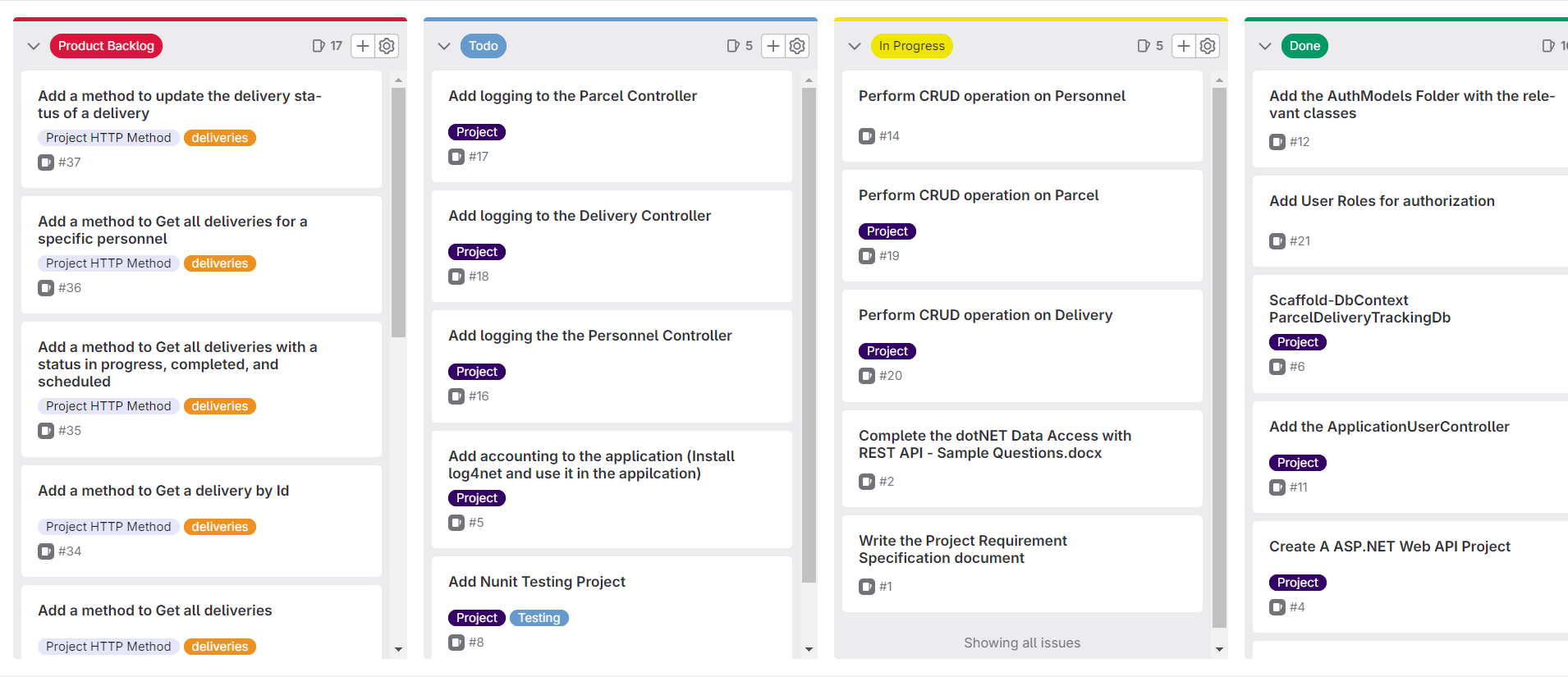
The scope of the Parcel Delivery Tracking System includes the following key functionalities:

* **Parcel Management**: Creation, update, and tracking of parcels, including sender and receiver information, parcel weights, statuses, and delivery dates.
* **Client Management**: Registration, profile management.
* **Delivery Management**: Assignment of delivery personnel to parcel, real-time status update, and tracking of deliveries.

# Definition and Acronyms

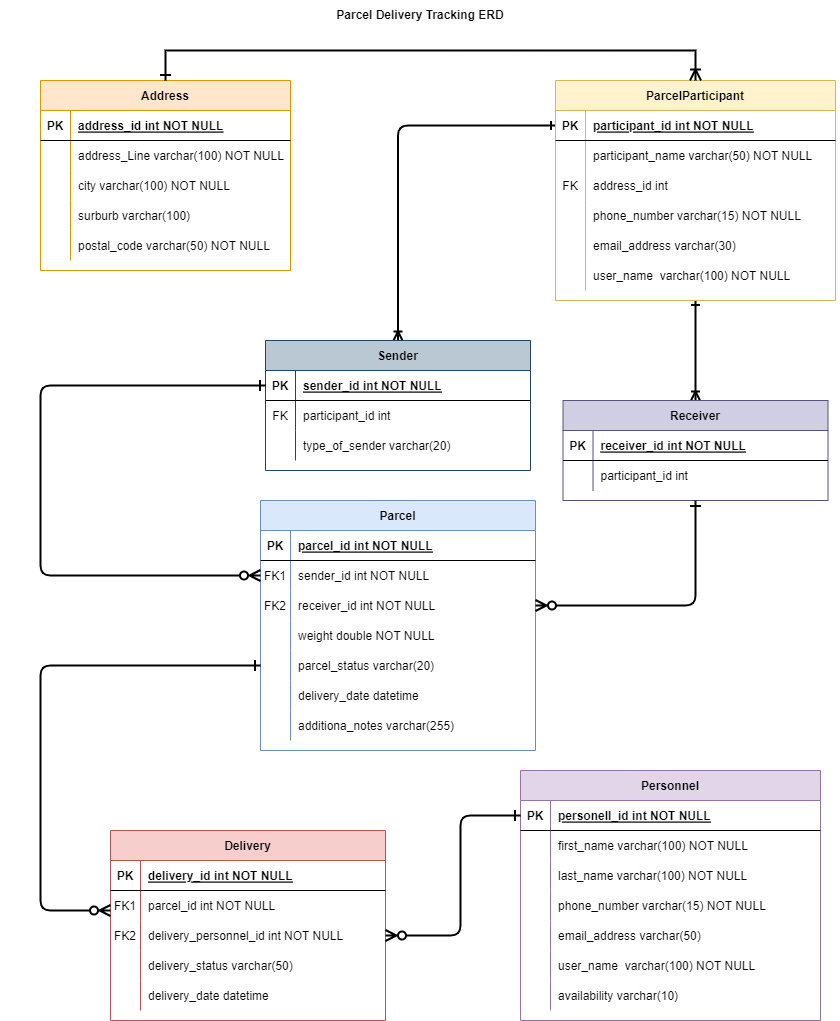
* EF – Entity Framework
* ERD – Entity Relationship Diagram
* API – Application Programming Interface
* DB - Database

# Project Backlog Item



# Entity Relationship Diagram

The major features of the parcel delivery tracking database system as shown in below ERD.



## Table Descriptions

1. Address Table:

* Contains information about addresses including address line, city, suburb, and postal. code. The table will be used to store address details for parcel participants (sender/receiver).

1. Parcel Participations Table:

* Represents participants (either sender or receiver) involved in parcel transactions.
* References Address table to store address information.
* Include fields for participant name, phone number, and email address.

1. Sender Table:

* Represent senders of parcels.
* References the ParcelParticipant table for sender details.
* Includes a “type” field, which can specify the type of sender (e.g., individual, business).

1. Receiver Table:

* Represent receivers of parcels.
* References the ParcelParticpartion Table for receive details.

1. Parcel Table:

* Contain information about parcels, including details such as sender, receiver, parcel weight, status, delivery date, and additional notes.
* References the Sender and Receiver Tables to establish relationships.

1. Personnel Table:

* Represents personnel involved in parcel delivery.
* Incudes fields for first name, last name, phone number, email address, role, and availability.

1. Delivery Table:

* Represents delivery transactions.
* References the Parcel Table for the parcel being delivered and the Personnel table for the delivery personnel.
* Includes fields for delivery status and deliver date.

# System Features

## Functional Requirements

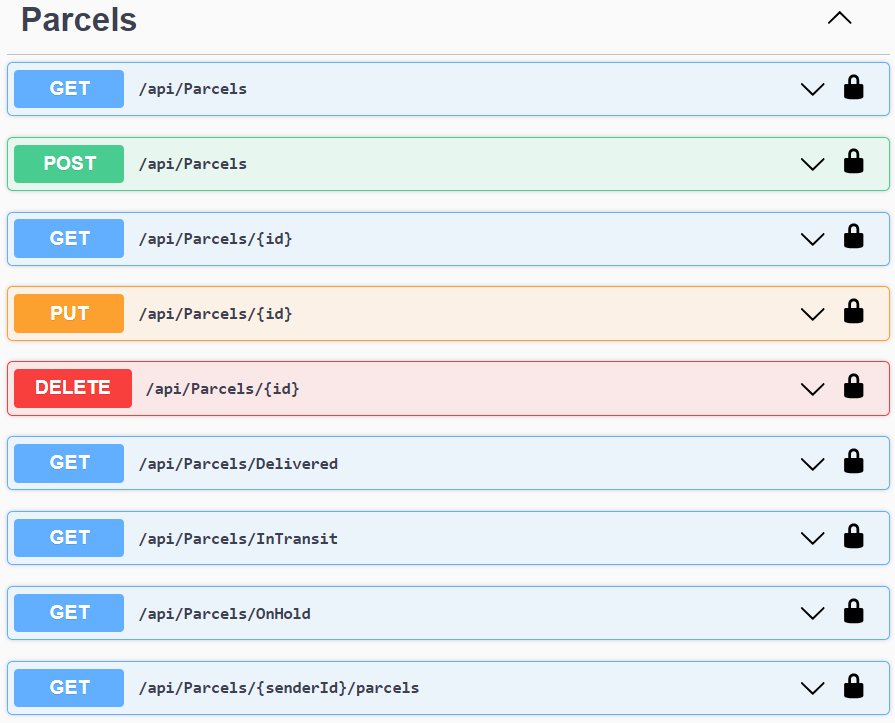
* **Parcel Management**: The system allows for the efficient management of parcels, capturing details such as sender and receiver information, parcel weight, and status (In Transit, Delivered, On Hold, Cancelled)
* **Client Management**: Client, whether businesses or individuals, can register and access the system. They can maintain their profiles, including contact information and relevant details.
* **User Authentication**: The system incorporates user authentication and role-based access control. Clients register as users with unique usernames and passwords, allowing them to log in to access their accounts and request parcel deliveries.
* **Parcel Tracking:** Clients can request parcel deliveries by providing parcel details. They can track the status of their parcel in real-time
* **Delivery Management:** The system assigns delivery personnel based on availability. Delivery personnel update the delivery status as they complete deliveries or encounter issues.

## Non-Functional Requirements

* **Security**: The system incorporates user authentication and authorization and role-based access control.
* **Monitoring and Logging**: The system log relevant events, errors, and user activities for auditing and troubleshooting purposes.
* **Data Integrity:** Data stored in the application database should maintain its accuracy, and consistency.

# API Swagger Endpoint List

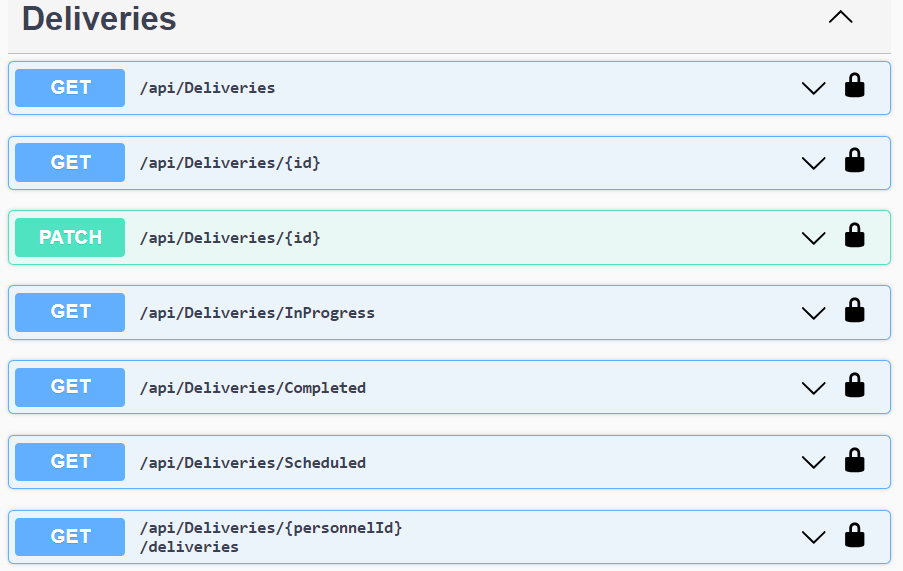
## Parcel Information Endpoints



## Personnel Infromation Endpoints



## Delivery Information Endpoints



# Lesson Learned

* Understanding of ASP.NET Core Identity – How ASP.NET Core Identity provides user identity management. The use of JWT for authentication and authorization in an ASP.NET Core application.
* How to use Repository Pattern for data access, to create a layer of abstraction between the business logic and data access.
* Unit Testing with Mocking – Learned how to write unit tests using mock objects