Software Requirements Specification

for

Delivery Company

**Version 1.0 approved**

**Prepared by Strimbeanu Mihai**

**Gavrila Sebastian**

**Sbaroi Ionut-Alexandru**

**SMA SRL**

**12/3/2024**

**Table of Contents**

[1. Introduction 1](#_Toc162266449)

[1.1 Purpose 1](#_Toc162266450)

[1.2 Document Conventions 1](#_Toc162266451)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc162266452)

[1.4 Product Scope 1](#_Toc162266453)

[1.5 References 2](#_Toc162266454)

[2. Overall Description 2](#_Toc162266455)

[2.1 Product Perspective 2](#_Toc162266456)

[2.2 Product Functions 3](#_Toc162266457)

[2.3 User Classes and Characteristics 3](#_Toc162266458)

[2.4 Operating Environment 4](#_Toc162266459)

[2.5 Design and Implementation Constraints 5](#_Toc162266460)

[2.6 User Documentation 5](#_Toc162266461)

[2.7 Assumptions and Dependencies 5](#_Toc162266462)

[3. External Interface Requirements 6](#_Toc162266463)

[3.1 User Interfaces 6](#_Toc162266464)

[3.2 Hardware Interfaces 6](#_Toc162266465)

[3.3 Software Interfaces 7](#_Toc162266466)

[3.4 Communications Interfaces 7](#_Toc162266467)

[4. System Features 8](#_Toc162266468)

[4.1 User Registration 8](#_Toc162266469)

[4.2 User Login and Navigation 9](#_Toc162266470)

[4.3 User Navbar display 9](#_Toc162266471)

[4.4 User Logout 10](#_Toc162266472)

[4.5 Home Page Redirection and Information Display 11](#_Toc162266473)

[4.6 Orders Page Access and Viewing 11](#_Toc162266474)

[4.7 Manage Users Page 12](#_Toc162266475)

[4.8 Home Page Description Editing 12](#_Toc162266476)

[4.9 Orders Page Access and Control for Administrators 13](#_Toc162266477)

[4.10 Visual Loading Animation 13](#_Toc162266478)

[4.11 User Navbar Display and Logout 14](#_Toc162266479)

[4.12 Deliveries Page Access and Viewing 15](#_Toc162266480)

[4.13 Delivery Day Start/End Marking 15](#_Toc162266481)

[4.14 Mark Items as Delivered on Deliveries Page 16](#_Toc162266482)

[4.15 Access All Deliveries for Administrators 16](#_Toc162266483)

[4.16 City Management on Home Page for Administrators 17](#_Toc162266484)

[4.17 Order Cancellation on Orders Page 18](#_Toc162266485)

[4.18 Order Creation on Orders Page 18](#_Toc162266486)

[4.19 Access Delivery Cars Page for Administrators 19](#_Toc162266487)

[4.20 Create New Delivery Car for Administrators 20](#_Toc162266488)

[4.21 Delete Delivery Car for Administrators 21](#_Toc162266489)

[4.22 Assign Employee to Delivery Car for Administrators 21](#_Toc162266490)

[5. Other Nonfunctional Requirements 22](#_Toc162266491)

[5.1 Performance Requirements 22](#_Toc162266492)

[5.2 Safety Requirements 23](#_Toc162266493)

[5.3 Security Requirements 23](#_Toc162266494)

[5.4 Software Quality Attributes 23](#_Toc162266495)

[5.5 Business Rules 24](#_Toc162266496)

[6. Other Requirements 25](#_Toc162266497)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Sbaroi Ionut-Alexandru | 12/03/2024 | - added content to Intended Audience and Reading Suggestions - added content to Product functions  - added team members, creation date, and name of the project on the cover | 1.1 |
| Sbaroi Ionut-Alexandru | 15/03/2024 | -added content to Product Scope | 1.2 |
| Sbaroi Ionut-Alexandru | 15/03/2024 | -added content Operating Environment | 1.3 |
| Sbaroi Ionut-Alexandru | 15/03/2024 | -added content Hardware Interfaces | 1.4 |
| Strimbeanu Mihai-Alexandru | 18/03/2024 | - added 5 new system features | 1.5 |
| Sbaroi Ionut-Alexandru | 23/03/2024 | - added 5 new system features  - added content for Security Requirements - added content for Software Interfaces | 1.6 |
| Gavrila Mihai Sebastian | 24/03/2024 | - added 5 new features | 1.7 |
| Strimbeanu Mihai-Alexandru | 25/03/2024 | - added 4 new features | 1.8 |
| Sbaroi Ionut-Alexandru | 25/03/2024 | - added 3 new features | 1.9 |
| Sbaroi Ionut-Alexandru | 25/03/2024 | - added content for User classes and Characteristics  - added content for Design and Implementation Constraints  - added content for Performance Requirements  - added content for Business Rules | 1.10 |
| Strimbeanu Mihai-Alexandru | 25/03/2024 | -completed content :  1.1  1.2  1.5  2.7  3.1  2.1  3.4  5.4  -updated content:  1.3 | 1.11 |

# Introduction

## Purpose

*The purpose of this document is to specify the functional and non-functional requirements for the Delivery Company Web Application. This system will facilitate the operational and logistical management of a delivery company, enabling efficient order placement, tracking, and management.*

## Document Conventions

*Standard conventions for specifying software requirements are followed throughout this document:*

*- API, Aplication Programing Interface*

*-SRS, Software requirement specification*  
 *- to be done ###################################################*

## Intended Audience and Reading Suggestions

Types of Readers:

*-Developers: Developers will be interested in the detailed technical specifications, including system architecture, database design, and API documentation.*

*-Marketing Staff: Marketing staff may be interested in the high-level features and functionalities of the software to better grasp its market positioning and potential customer benefits.*

*-Users(admins): Users will want to know how to interact with the system, including user authentication, order placement, and tracking functionalities.*

*-Testers: Testers will look for detailed functional and non-functional requirements to create test cases and ensure the software meets the specified criteria.*

*The SRS provides a detailed look into the Courier Management System. It covers everything from its scope, functional and non-functional requirements, system architecture, user roles, and security considerations. Additionally, you'll find in-depth descriptions of system behaviors, interfaces, and data requirements within the SRS.*

## Product Scope

*The software we're talking about here is a web application for a delivery company. Its main purpose is to make order management easier and more efficient. With this app, users can create, delete, and track orders, as well as automatically assign them to delivery employees.*

*Here are the key objectives and benefits of using this application:*

*-Efficient Order Management: This app will make it seamless for users to create and delete orders, reducing the need for manual effort and minimizing errors that can happen with traditional methods.*

*-Order Tracking: Users will be able to easily track the status and history of their orders, giving them transparency and improving customer satisfaction.*

*- Automated Assignment: The application will automate the process of assigning orders to delivery employees based on factors like proximity, workload, and availability. This feature aims to optimize delivery routes and ensure orders are fulfilled on time.*

*- Enhanced Productivity: By automating repetitive tasks and providing user-friendly interfaces for order management, this application will boost the productivity of both users and delivery employees.*

*-Improved Customer Experience: The app's efficient order handling and timely delivery will greatly enhance the overall customer experience, leading to customer loyalty and retention.*

*- Alignment with Business Goals: This software aligns perfectly with the company's goals of improving operational efficiency, optimizing resource utilization, and providing exceptional customer service.*

*By offering a centralized platform for order management and utilizing automation for order assignment, this web application aims to revolutionize the way orders are processed and fulfilled. Ultimately, it will contribute to the company's success in the competitive delivery services market.*

## References

*C# - programing standard (Microsoft):*

[*https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions*](https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions)

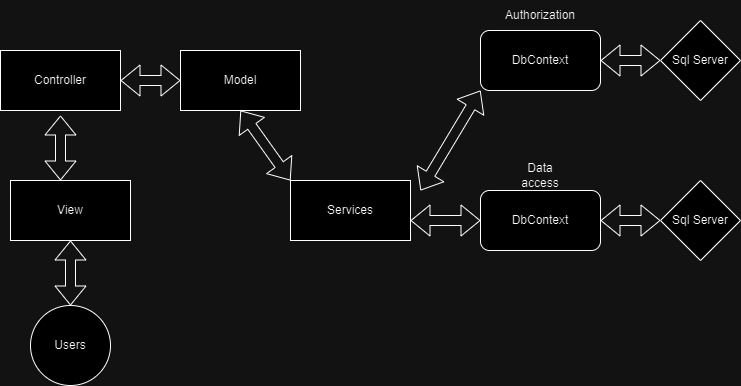
*Javascript – programing standard (WordPress):*

[*https://developer.wordpress.org/coding-standards/wordpress-coding-standards/javascript/*](https://developer.wordpress.org/coding-standards/wordpress-coding-standards/javascript/)

# Overall Description

## Product Perspective

*A simple flow diagram of the system:*



## Product Functions

*User Authentication and Access Control:*

*-Users can log in with a username and password.*

*-Different user roles (Administrator, Client, Courier) have distinct access privileges.*

*Order Management:*

*-Users can place, cancel, and confirm orders.*

*-System automatically assigns couriers based on various criteria*

*-Users can monitor order status and view order history.*

*User Management:*

*-Administrators can add, edit, and manage employees.*

*-Couriers can mark the start and end of their workday/delivery and update order status.*

*Vehicle Management:*

*-Administrators can allocate and manage vehicles for courier operations.*

*Data Viewing:*

*-Users can view real-time information about orders, deliveries, and courier assignments.*

*Security:*

*-Ensure the protection of personal and sensitive data.*

## User Classes and Characteristics

***Administrator:***

*Frequency of Use: High, as administrators are responsible for managing the system and its various components.*

*Subset of Product Functions Used: Full access to all administrative functions, including managing users, delivery cars, and other system settings.*

*Technical Expertise: Moderate to high technical expertise required, as administrators need to navigate and manage complex system functionalities.*

*Security or Privilege Levels: Highest privilege levels with full access to all system features and settings.*

*Educational Level/Experience: May vary, but typically administrators possess a higher educational level and/or experience in system administration.*

*Characteristics: Administrators play a crucial role in maintaining and optimizing system performance, ensuring smooth operations, and making strategic decisions to enhance efficiency.*

***Employee:***

*Frequency of Use: Moderate to high, depending on their role within the system.*

*Subset of Product Functions Used: Access to specific functionalities such as managing deliveries, marking orders, and interacting with the system interface.*

*Technical Expertise: Varies based on job role; may range from basic to moderate technical skills.*

*Security or Privilege Levels: Limited access compared to administrators; permissions tailored to their specific roles and responsibilities.*

*Educational Level/Experience: Varied educational backgrounds and levels of experience; may include students, delivery personnel, or administrative staff.*

*Characteristics: Employees interact with the system as part of their job responsibilities, performing tasks related to deliveries, orders, and other functionalities.*

***User:***

*Frequency of Use: Moderate to high, depending on their needs and interactions with the system.*

*Subset of Product Functions Used: Access to functionalities such as viewing orders, placing new orders, and tracking deliveries.*

*Technical Expertise: Varies widely; may include individuals with basic to moderate technical skills.*

*Security or Privilege Levels: Limited access compared to employees; permissions focused on accessing and utilizing user-specific features.*

*Educational Level/Experience: Varied; may include students, faculty, average people or other community members with different levels of familiarity with similar systems.*

*Characteristics: Users interact with the system to perform tasks related to their own orders, deliveries, or other activities.*

***Importance of User Classes:***

*The most important user class for this product is the Administrator, as they have full control over system management and settings, ensuring the system operates effectively.*

*The Employee class is also crucial as they interact directly with the system to perform tasks related to deliveries and orders, contributing to the system's primary function.*

*The User class is very important because it is the main point of the application to servers as a hub for the average people to place and manage orders.*

## Operating Environment

*Hardware Platform:*

*The servers hosting the application should have the capability to support the .NET framework and SQL Server. It is important to ensure that the hardware meets the minimum requirements in terms of CPU, RAM, and disk space to handle the expected workload and data storage needs.*

*Operating System and Versions:*

*During development and testing, the application will primarily be focused on Windows-based operating systems. For server deployment, it is recommended to use Windows 10 or 11, every version after version 1507 of Windows 10 is suitable .For development environments, Windows 10 or 11 is recommended.*

*Software Components:*

*The application will be developed using the MVC architecture with .NET 8, which ensures compatibility with the latest features and enhancements provided by the framework. SQL Server will be used as the relational database management system to store and manage data related to orders, users, delivery employees, and other relevant entities. Compatibility with SQL Server versions compatible with .NET 8 need to be minimum SQL Server 2005. Visual Studio 2022 or later will serve as the primary IDE for designing, coding, debugging, and testing the application. When it comes to web servers, options like Internet Information Services (IIS), Apache HTTP Server, Azure Devops WebApp and AWS can be chosen, as long as they are compatible with ASP.NET MVC applications and have the necessary configurations to support .NET applications.*

*Other Considerations:*

*To provide a consistent user experience across different platforms, the web application will be designed to be compatible with modern web browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari. Security measures will also be integrated into the application, user authentication, authorization mechanisms, and data encryption, to protect sensitive information, everything built using EntityFramework.Identity Nuget package from .NET.*

## Design and Implementation Constraints

***Hardware Limitations:***

*As the application will grow, a consideration for memory and processing constraints need to be taking into consideration because database and the system this application is deployed needs to support the necessary amount of computing and storage power.*

***Maintenance and Support Responsibilities:***

*Consideration of training and documentation requirements to enable university staff to manage and support the software effectively.*

## User Documentation

*<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>*

## Assumptions and Dependencies

*Third-party or Commercial Components:*

*-We assume seamless integration of the Bootstrap Icons version 1.11.3 for everything regarding Icons, SVG, Icon fonts.*

*-We assume seamless integration of the Bootstrap version 5.3.3(min) for everything regarding layout and components*

*Development or Operating Environment Issues:*

*-We assume availability of the .NET (MVC) version 8.x.x Development Framework for building the application's core functionalities.*

*-We assume availability of the Entity Framework 8.x.x for building the application's Data access and authorization*

# External Interface Requirements

## User Interfaces

***Software Components Requiring User Interface:***

*- Authentication Module*

*- Home Screen*

*- Contact Screen*

*- Orders*

*- Deliveries*

*- User Module*

*- Delivery cars*

***Standard Buttons and Functions:***

* *Navigation elements Navigation bar.*
* *Common actions like "Save", "Cancel", "Submit", ”Edit”, and "Delete."*

## Hardware Interfaces

**Web Interface (User Interface):**

Logical Characteristics: The web interface offers a user-friendly platform that can be accessed through web browsers on different devices like desktops, laptops, tablets, and smartphones.

Physical Characteristics: Users interact with the application by using graphical elements such as buttons, forms, and menus that are displayed on their device screens.

Data and Control Interactions: Users input data (e.g., order details) and control application actions (e.g., creating or deleting orders) by sending HTTP requests to the web server. The server processes these requests and responds with the appropriate data or status updates.

**Database Interface:**

Logical Characteristics: The database interface enables the storage, retrieval, and manipulation of data within the SQL Server database management system. Physical Characteristics: Data is stored in tables within the SQL Server database, where each table represents a specific entity (e.g., orders, customers, employees).

Data and Control Interactions: The application communicates with the database using SQL queries and commands like SELECT, INSERT, UPDATE, and DELETE. These operations allow for tasks such as retrieving order information, adding new orders, updating order statuses, and deleting orders The connection and the queries are established using EntityFrameworkCore from .NET framework.

**Server Interface:**

Physical Characteristics: The web application is deployed on a server environment, which can be either physical or virtual. It runs the necessary operating system and hosting software (e.g., IIS, Azure, AWS).

Data and Control Interactions: The application server receives HTTP requests from clients (web browsers) and processes them. It executes the required methods and services’s callsand interacts with the database as needed. Finally, it generates HTTP responses that contain the requested data or HTML content for rendering on the client side, based on the page view the user is accessing.

By describing these interface characteristics, the software product's interactions with hardware components are clear, ensuring a comprehensive understanding of how data and control flow between the software and the underlying hardware infrastructure.

## Software Interfaces

**Bootstrap 5**: This front-end framework is used for designing responsive and attractive user interfaces. It provides a set of CSS and JavaScript components for creating web pages and web applications. The interface between your application and Bootstrap 5 involves utilizing its classes, components, and utilities to style and structure your user interface elements.

**Ajax (Asynchronous JavaScript and XML)**: Ajax is used for making asynchronous HTTP requests from the client-side to the server-side. It allows your web application to update parts of a web page without reloading the entire page. The interface with Ajax involves making requests to your server-side application endpoints and handling the responses asynchronously using JavaScript.

**Entity Framework**: Entity Framework is an Object-Relational Mapping (ORM) framework for .NET applications. It provides a way to interact with databases using .NET objects. The interface with Entity Framework involves defining your data model using classes (entities) and interacting with the database through Entity Framework's APIs for querying, inserting, updating, and deleting data.

**Identity Framework**: Identity Framework is used for authentication and authorization in ASP.NET applications. It provides a set of APIs for managing user authentication, user accounts, roles, and permissions. The interface with Identity Framework involves integrating it into your application to handle user registration, login, logout, password management, and access control.

## Communications Interfaces

***Network Server Communication Protocols:***

*Communication between the web server and client browsers will utilize the HTTPS protocol.*

*The application will handle HTTP requests and responses using ASP.NET and MVC architecture.*

***Database Communication (Entity Framework):***

*The application will communicate with the database using Entity Framework, an ORM (Object-Relational Mapping) framework.*

*Requirements include:*

* *Establishing connections to the database server.*
* *Executing database queries and updates.*
* *Handling database transactions.*

***Electronic Forms:***

*The application may utilize electronic forms for data input and submission.*

*Requirements include:*

* *Rendering form elements (textboxes, dropdowns, etc.) in web pages.*
* *Validating form data on the client and server-side.*

# System Features

## User Registration

* + 1. Description and Priority

Description: Allow potential users to register with the system by providing necessary information, enabling them to create an account and access the system's functionalities.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User navigates to the registration page.

Response: System prompts the user to enter registration information.

Stimulus: User presses the Register button.

Response: System validates entered information and proceeds accordingly.

* + 1. Functional Requirements

REQ-1: The system shall prompt the user to enter their email address in a text field (mandatory).

REQ-2: The system shall prompt the user to enter their desired password in a password field (mandatory).

REQ-3: The system shall check if the provided email address already exists in the system.

REQ-4: The system shall check if the provided password meets the system's criteria for complexity and length.

REQ-5: If all checks (email address availability and password criteria) are passed, the system shall create a new user account with the provided email and password.

REQ-6: Upon successful registration, the system shall progress the user to the login page.

REQ-7: If any other errors occurred during registration, the system shall notify the user that registration has been unsuccessful without specifying the reason.

REQ-8: If there are multiple problems regarding the fields validation, the system shall allow the user to retry the registration process.

## User Login and Navigation

* + 1. Description and Priority

Enable potential users to log in with their credentials, navigate through the system's functionalities, and view their name/email in the navbar for easy access to logout.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User navigates to the Login page.

Response: System prompts the user to enter login information.

Stimulus: User presses the Login button.

Response: System validates entered information and proceeds accordingly.

* + 1. Functional Requirements

REQ-1: The system shall prompt the user to enter their email address in a text field (mandatory), if the email is not provided, the system shall display an error message prompting the user to enter their email.

REQ-2: The system shall prompt the user to enter their password in a password field (mandatory),

if the password is not provided, the system shall display an error message prompting the user to enter their password.

REQ-3: The system shall check if the entered email/ and password fields are completed correctly, if any of the fields are incomplete, the system shall inform the user to complete them.

REQ-4: The system shall verify if there is a user with the provided email and password that matches a saved user in the database, if the email and password combination do not match any user in the database, the system shall inform the user of incorrect credentials and prompt them to try again.

REQ-5: If the login is successful, the system shall redirect the user to the Home page (About).

REQ-7: The system shall display only the pages the user has rights to based on their role.

## User Navbar display

* + 1. Description and Priority

Display the name/email of the logged-in user in the navbar.

Priority: Medium

* + 1. Stimulus/Response Sequences

Stimulus: User successfully logs into the application.

Response: Name/email of the authorized user is displayed in the right corner of the application navbar.

Stimulus: User clicks on the displayed name/email in the navbar.

Response: Dropdown menu is displayed.

* + 1. Functional Requirements

REQ-1: Upon successful login, the system shall display the name/email of the authorized user in the right corner of the application navbar.

REQ-2: When the user clicks on the displayed name/email in the navbar, the system shall display a dropdown menu.

## User Logout

* + 1. Description and Priority

Provide the option to sign out from the application.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User selects the "Sign Out" option from the dropdown menu.

Response: User is logged out from the application and redirected to the Log In page.

* + 1. Functional Requirements

REQ-1: The dropdown menu shall include options, including "Sign Out".

REQ-2: Upon selecting the "Sign Out" option from the dropdown menu, the system shall log the user out from the application.

REQ-3: After logging out, the system shall redirect the user to the Log In page.

## Home Page Redirection and Information Display

* + 1. Description and Priority

Redirect signed-in users or employees to the Home page upon logging in, where they can view a short description of the application and the cities serviced by the Delivery Company.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User successfully logs in.

Response: System automatically redirects the user to the Home page.

* + 1. Functional Requirements

REQ-1: Upon logging in, the system shall redirect the user to the Home page.

REQ-2: The Home page shall display a descriptive text about the application. (for users/employees, the description about the application shall be presented in read-only mode).

REQ-3: The Home page shall contain a table listing the cities serviced by the Delivery Company. (for users/employees, the table displaying cities shall be in read-only mode)

REQ-4: The application logo shall be prominently displayed on the Home page.

## Orders Page Access and Viewing

* + 1. Description and Priority

Allow users to access the Orders page to view the history and status of their orders.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User successfully logs in.

Response: User can select the Orders page from the navigation bar.

* + 1. Functional Requirements

REQ-1: Upon logging in, the user shall have the option to select the Orders page from the navigation bar.

REQ-2: The Orders page shall display a history of past orders along with their current status.

REQ-3: The user shall be able to view order details on the Orders page, including order ID, items, destination, status, price, and other order details.

## Manage Users Page

* + 1. Description and Priority

Allow administrators to access the Manage Users page for creating new users with various roles, managing existing users, and viewing a list of users categorized by their roles.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator can select the Manage Users page from the Navigation bar.

* + 1. Functional Requirements

REQ-1: Upon logging in, the administrator shall have the option to select the Manage Users page from the Navigation bar.

REQ-2: The Manage Users page shall provide options to create new users, manage existing users, and display a list of users categorized by their roles.

REQ-3: The administrator shall be able to create new users by providing necessary information such as username, password, email, and role selection (including Admin role).

REQ-4: The administrator shall be able to manage existing users by updating their credentials or information as needed.

REQ-5: Users shall be listed along with their basic information, such as username, email, and role.

REQ-6: Actions performed by the administrator, such as creating new users or updating existing ones, shall be saved and reflected in the system.

## Home Page Description Editing

* + 1. Description and Priority

Allow administrators with the Admin Role to access the Home page and edit/update the description of the application.

Priority: Medium

* + 1. Stimulus/Response Sequences

Stimulus: Administrator with Admin Role successfully logs in.

Response: System directs the administrator to the Home page.

* + 1. Functional Requirements

REQ-1: Upon logging in, the system shall direct the administrator with the Admin Role to the Home page.

REQ-2: The Home page shall display the current description of the application.

REQ-3: The administrator with the Admin Role shall have the ability to edit and update the description of the application directly on the Home page.

REQ-4: Changes made to the description by the administrator shall be reflected immediately upon saving.

## Orders Page Access and Control for Administrators

* + 1. Description and Priority

Allow administrators to access the Orders page with the ability to view the history and status of all orders, regardless of the user who created them. Additionally, grant administrators full control over canceling orders that are not yet in transit.

Priority: Hight

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator can select the Orders page from the navigation bar.

* + 1. Functional Requirements

REQ-1: Upon logging in, the administrator shall have the option to select the Orders page from the navigation bar.

REQ-2: The Orders page shall display the history of all orders, including details such as order ID, items, destination, user, and status.

REQ-3: The administrator shall be able to view the status of each order and identify orders that are not yet in transit.

REQ-4: The administrator shall have full control over the canceling status of orders, with the ability to cancel orders that are not yet in transit.

REQ-5: If necessary, the administrator shall be able to review order details to assist users or resolve any issues.

## Visual Loading Animation

* + 1. Description and Priority

Implement visually appealing loading animations to provide feedback to users during page transitions and various actions, enhancing the user experience.

Priority: Low

* + 1. Stimulus/Response Sequences

Stimulus: User triggers an action that requires loading or processing time (e.g., submitting a form, fetching data).

Response: Visual loading animation is displayed during the loading process.

* + 1. Functional Requirements

REQ-1: The loading animation shall provide feedback to the user about the loading progress, ensuring it is intuitive and visually appealing.

REQ-2: Whenever the user triggers an action that requires loading or processing time, such as submitting a form or fetching data, a loading animation shall be displayed.

REQ-3: During the loading process, the rest of the user interface shall be partially obscured or disabled to prevent user interaction until the operation is complete.

## User Navbar Display and Logout

* + 1. Description and Priority

Display the email of the signed-in user in the navbar and provide the option to sign out from the application.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User successfully logs in.

Response: Name/email of the authorized user is displayed in the right corner of the application navbar.

* + 1. Functional Requirements

REQ-1: Upon successful login, the system shall display the email of the authorized user in the right corner of the application navbar.

REQ-2: When the user clicks on the displayed email in the navbar, the system shall display a dropdown menu.

REQ-3: The dropdown menu shall include options, including "Sign Out".

REQ-4: Upon selecting the "Sign Out" option from the dropdown menu, the system shall log the user out from the application.

REQ-5: After logging out, the system shall redirect the user to the Log In page.

## Deliveries Page Access and Viewing

* + 1. Description and Priority

Description: Allow users (employees or administrators) to access the Deliveries page to view a comprehensive list of all deliveries, including historical and active ones.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User successfully logs in.

Response: User can select the Deliveries page from the navigation bar.

* + 1. Functional Requirements

REQ-1: Upon logging in, the user should have the option to select the Deliveries page from the navigation bar.

REQ-2: When the user clicks on the Deliveries button on the navigation bar, they should be redirected to the Deliveries page.

REQ-3: The Deliveries page will display a comprehensive list of all deliveries, including historical and active ones.

## Delivery Day Start/End Marking

* + 1. Description and Priority

Allow users (employees or administrators) to mark the start and end of a delivery day on the Deliveries page.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User marks the start of the day.

Response: The Status of the delivery is updated and seen in the UI.

Stimulus: User marks the end of the day.

Response: The Status of the delivery is updated and seen in the UI.

* + 1. Functional Requirements

REQ-1: Upon logging in, the user shall have the option to access the Deliveries page.

REQ-2: The Deliveries page shall include options to mark the start and end of a new delivery day.

REQ-3: User can initiate the start or end of the delivery day by selecting the respective option, based on the state of the delivery.

## Mark Items as Delivered on Deliveries Page

* + 1. Description and Priority

Allow users (employees or administrators) to mark items as successfully delivered on the Deliveries page, updating the delivery status accordingly.

Priority: Medium

* + 1. Stimulus/Response Sequences

Stimulus: User marks a delivery item as delivered.

Response: The status of the delivery item is updated, and the status of the order as well.

* + 1. Functional Requirements

REQ-1: Upon logging in, the user shall have the option to access the Deliveries page.

REQ-2: The Deliveries page shall include functionality to mark items as successfully delivered.

REQ-3: User can update the order status of items to "Delivered" by selecting the respective orders from that delivery which must be active.

## Access All Deliveries for Administrators

* + 1. Description and Priority

Allow administrators to access the Deliveries page to view all deliveries for all employees, including historical and active ones.

Priority: Medium

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator can access the Deliveries page with all the data from all the employees.

* + 1. Functional Requirements

REQ-1: Upon logging in, the administrator with the Admin Role should have the option to access the Deliveries page.

REQ-2: The Deliveries page should display all deliveries for all employees, including historical and active ones.

REQ-3: Administrator can view deliveries for all employees, regardless of the employee who initiated the delivery.

## City Management on Home Page for Administrators

* + 1. Description and Priority

Allow administrators to access the Home page with the ability to add or delete cities serviced by the Delivery Company.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator is directed to the Home page.

Stimulus: Administrator can select and create a new city from the Cities menu.

Response: The changes persist in the UI and in the database for all the users.

Stimulus: Administrator can delete a city from the Cities menu.

Response: The changes persist in the UI and in the database for all the users.

Stimulus: Administrator can update the description of the application.

Response: The changes persist in the UI and in the database for all the users.

* + 1. Functional Requirements

REQ-1: Upon logging in, the system will direct the administrator to the Home page.

REQ-2: The Home page shall include a table listing the cities serviced by the Delivery Company.

REQ-3: The administrator should have the ability to add new cities to the table by providing the city name.

REQ-4: The administrator should have the ability to delete existing cities from the table as needed.

REQ-5: All modifications made by the administrator shall be saved and reflected in the system.

## Order Cancellation on Orders Page

* + 1. Description and Priority

Allow users to access the Orders page to cancel orders that are not yet in transit.

Priority: Medium

* + 1. Stimulus/Response Sequences

Stimulus: User successfully logs in.

Response: User can select the Orders page from the navigation bar.

Stimulus: User can cancel an order that it is not already in transit.

Response: The status of the order will pe updated to Canceled.

* + 1. Functional Requirements

REQ-1: Upon logging in, the user should have the option to select the Orders page from the navigation bar.

REQ-2: The Orders page should display a list of orders with their respective statuses.

REQ-3: The user should be able to cancel orders that are not yet in transit.

REQ-4: The option to cancel orders should be grayed out for orders with statuses other than "not yet in transit", or not visible.

## Order Creation on Orders Page

* + 1. Description and Priority

Allow users to access the Orders page to create new order.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: User successfully logs in.

Response: User can select the Orders page from the navigation bar.

Stimulus: User can create a new order by providing the information in the creation pop-up.

Response: The changes persist in the UI and in the database regarding the order.

* + 1. Functional Requirements

REQ-1: Upon logging in, the user should have the option to select the Orders page from the navigation bar.

REQ-2: The Orders page should provide fields for the user to input details such as items, destination, weight, width, height, and length for creating a new order.

REQ-3: Before creating the order, the system shall check if the provided details match the criteria for transportation based on existing delivery cars in the city.

REQ-4: If no delivery car is available, the system shall set the order status to "Unassigned."

REQ-5: The system should dynamically calculate and display the price of the order based on the provided weight, width, height, and length.

REQ-6: After placing the order, the details should be added to the Orders page table, including the order ID, items, destination, status, and price.

## Access Delivery Cars Page for Administrators

* + 1. Description and Priority

Allow administrators to access the Delivery Cars page to retrieve existing cars from the database.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator can select the Delivery Cars page from the navigation bar.

Stimulus: Administrator select the Delivery Cars page from the navigation bar.

Response: A list will all available cars will be displayed on the screen.

* + 1. Functional Requirements

REQ-1: Upon logging in, the administrator shall have the option to access the Delivery Cars page from the navigation bar.

REQ-2: The Delivery Cars page shall display a list of existing delivery cars retrieved from the database.

REQ-3: Only administrators shall have access to view the Delivery Cars page.

## Create New Delivery Car for Administrators

* + 1. Description and Priority

Allow administrators to create a new delivery car on the Delivery Cars page.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator can access the Delivery Cars page from the navigation bar.

Stimulus: Administrator can select to create a new car and provide the necessary informatiosn.

Response: The car will be created and the changes will persist in the UI and database.

* + 1. Functional Requirements

REQ-1: Upon logging in, the administrator should have the option to access the Delivery Cars page from the navigation bar.

REQ-2: The Delivery Cars page should include an option to create a new delivery car.

REQ-3: The administrator should be able to fill in details such as city, and capacity, etc for the new delivery car.

REQ-4: Upon submission, the system should add the new delivery car to the database.

REQ-5: The newly created delivery car should be displayed in the list of existing cars on the Delivery Cars page.

.

## Delete Delivery Car for Administrators

* + 1. Description and Priority

Allow administrators to delete an existing delivery car on the Delivery Cars page.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator can access the Delivery Cars page from the navigation bar.

Stimulus: Administrator can delete an already existing car.

Response: The car will be created and the changes will persist in the UI and database.

* + 1. Functional Requirements

REQ-1: Upon logging in, the administrator should have the option to access the Delivery Cars page from the navigation bar.

REQ-2: The Delivery Cars page should display a list of existing delivery cars retrieved from the database.

REQ-3: The administrator should be able to select a delivery car from the list to delete.

REQ-4: The system should prompt the administrator to confirm the deletion action.

REQ-5: Upon confirmation, the system should remove the selected delivery car from the database.

REQ-6: The deleted delivery car should no longer be displayed in the list of existing cars on the Delivery Cars page.

## Assign Employee to Delivery Car for Administrators

* + 1. Description and Priority

Allow administrators to assign an employee to a specific delivery car on the Delivery Cars page.

Priority: High

* + 1. Stimulus/Response Sequences

Stimulus: Administrator successfully logs in.

Response: Administrator can access the Delivery Cars page from the navigation bar.

Stimulus: Administrator can assign an already existing employee to a car.

Response: The car will be created and the changes will persist in the UI and database.

* + 1. Functional Requirements

REQ-1: Upon logging in, the administrator should have the option to access the Delivery Cars page from the navigation bar.

REQ-2: The Delivery Cars page should display a list of existing delivery cars retrieved from the database.

REQ-3: The administrator should be able to select a delivery car from the list to assign an employee.

REQ-4: The system should provide a list of available employees in the same city as the selected delivery car.

REQ-5: The administrator should be able to choose an employee from the list to assign to the selected delivery car.

REQ-6: Upon assignment, the selected employee should be associated with the chosen delivery car.

REQ-7: The system should update the database to reflect the assignment of the employee to the delivery car.

# Other Nonfunctional Requirements

## Performance Requirements

***Page Load Time:***

*Requirement: All pages should load within 3 seconds under normal server load conditions.*

*Rationale: To ensure a smooth and responsive user experience, especially for users accessing the system from different devices and network conditions. Faster page load times improve user satisfaction and encourage continued usage of the system.*

***Database Query Response Time:***

*Requirement: Database queries should return results within 1 second for commonly accessed data sets.*

*Rationale: Quick retrieval of data is essential for efficient system operation, particularly when users are browsing through large datasets or conducting searches. Minimizing database query response time enhances overall system performance and user productivity.*

***Concurrency Handling:***

*Requirement: The system should support at least 100 concurrent user sessions without noticeable degradation in performance.*

*Rationale: As the number of users accessing the system simultaneously increases, the system must maintain responsiveness and handle requests efficiently to prevent slowdowns or timeouts. Ensuring robust concurrency handling supports scalability and accommodates peak usage periods.*

***Error Handling Response Time****:*

*Requirement: Error messages should be displayed to the user within 2 seconds of encountering a system error.*

*Rationale: Prompt notification of errors enables users to take corrective action or seek assistance efficiently. Delayed error handling may frustrate users and disrupt their workflow, impacting overall usability and satisfaction.*

## Safety Requirements

*<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>*

## Security Requirements

**User Identity Protection:**

-Implement measures to protect user identities and sensitive information stored within the system, such as encryption of stored passwords and sensitive user data.

**Access Control:**

-Regularly review and update access control policies to ensure that users are assigned appropriate roles and permissions based on their responsibilities within the application.

**Data Protection:**

-Maintain data protection measures, such as encryption of sensitive data at rest and in transit, to safeguard user data against unauthorized access or disclosure.

-Implement data retention and disposal policies to securely manage and dispose of user data.

## Software Quality Attributes

*Correctness: The app should accurately perform all functions, including order processing and tracking. Ensure that orders are correctly assigned to delivery personnel.*

*Flexibility: The app should offer flexibility in customization for different delivery needs, such as varying delivery fees based on package size and weight. Users should be able to customize their delivery preferences easily.*

*Maintainability: The codebase should be well-structured and documented, facilitating easy maintenance and future updates. Aim for a low technical debt ratio and adhere to best practices in coding standards.*

*Portability: The app should be portable across different devices and operating systems, such as desktop computers. Ensure compatibility with major browsers.*

*Reliability: The app should consistently perform as expected under various conditions, handling high volumes of concurrent users and traffic spikes.*

*Reusability: Design components and modules in the app to be reusable across different parts of the system, promoting efficiency in development and reducing redundancy. Measure reusability by the percentage of reusable components in the codebase.*

*Robustness: The app should gracefully handle errors, exceptions, and invalid inputs without crashing or compromising data integrity. Test robustness by simulating various failure scenarios and ensuring the app's resilience.*

*Testability: The app should be designed for easy testing, with well-defined test cases and automated testing frameworks in place. Aim for a high test coverage percentage, ensuring thorough validation of all functionalities.*

*Usability: The app should be intuitive and user-friendly, with clear navigation, minimalistic design, and responsive interfaces.*

*Adaptability: The app should be able to adapt to changes in delivery details. For example, it should allow for easy canceling of the order, or automatically asign to a delivery car when one is available without causing disruptions.*

## Business Rules

***Administrator:***

*Role Description: Administrators have the highest level of access and control over the system. They are responsible for managing users, delivery cars, and system settings.*

*Actions they can perform:*

*Administrators can access all areas of the system, including user management, delivery car management, and system configuration settings.*

*They have the authority to create, update, or delete user accounts, assign roles, and reset passwords.*

*Administrators can add new delivery cars, delete existing ones, and assign employees to specific delivery cars.*

***Employee:***

*Role Description: Employees are individuals responsible for handling deliveries, managing orders, and performing day-to-day operations within the system.*

*Actions they can perform:*

*Employees have access to functionalities related to managing deliveries, marking orders, and interacting with customers.*

*They can view and update delivery statuses, mark items as delivered, and initiate the start or end of a delivery day.*

*Employees may be assigned to specific delivery cars by administrators, allowing them to fulfill delivery tasks assigned to them.*

***User:***

*Role Description: Users are individuals who interact with the system primarily to place orders and track deliveries.*

*Actions they can perform:*

*Users have access to functionalities such as viewing order history, placing new orders, and canceling orders that are not yet in transit.*

# Other Requirements

*<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

**Appendix A: Glossary**

*<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>*

**Appendix B: Analysis Models**

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams*.>

**Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*