**Detailed Software Technical Design (DSTD)**

**For**

***V2V Couriers***

Table of Contents

1 Introduction.. 5

1.1 Purpose. 5

1.2 Scope. 5

1.3 Definitions, Acronyms and Abbreviations. 5

1.4 References. 5

2 Assumptions.. 6

3 PROJECT OVERVIEW AND USECASE DIAGRAMS.. 6

4 Detailed Design for Module: <<Module Name>>. 7

4.1 Detailed Design for Feature: <<Feature Name>>. 7

**4.1.1** **User Profile**. 7

**4.1.2** **Navigation Map**.. 7

**4.1.3** **UI Screen Design**.. 7

*4.1.3.1* *Screen Layout* 7

*4.1.3.2* *Validations*. 8

**4.1.4** **Object Model/Data Flow Design**.. 8

*4.1.4.1* *Interface Description*. 8

*4.1.4.2* *Class Design*. 8

4.1.4.2.1 Algorithmic/Business Logic Description (Optional) 8

4.1.4.2.2 Local Data Structure (Optional) 8

*4.1.4.3* *Data Transfer/Value objects*. 8

*4.1.4.4* *Related Database Tables*. 9

*4.1.4.5* *Dependencies with Other Sub-systems/Components*. 9

4.2 Detailed Design for Feature: <<Feature Name>>. 9

5 Detailed Design for Module: <<Module Name>>. 10

6 Database Design.. 11

6.1 Entity Relationship Diagrams. 11

7 Packaging/Folder Structure and System Artifacts.. 12

7.1 System Artifacts. 12

8 Core Technical Services Design.. 13

8.1 Persistence. 13

8.2 Inter-process Communication. 13

8.3 Authentication and Authorization. 13

8.4 Error Handling.. 13

8.5 Logging.. 13

8.6 Transaction Management. 13

8.7 Other Applicable Technical Services. 13

9 Non Functional Requirements.. 14

Appendix.. 14

1. Design Metrics to be Used.. 14

2. Supplementary Information. 14

3. DQI – Design Quality Index Checklist. 14

**Document Revisions**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** | **Reviewer** | **Approver** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Information Handling**

These documents are strictly for **Virtusa/Client** use only. They shall not be shared with an external party other than the client concerned. These documents should always be kept securely and employees shall use reasonable care protecting these documents from unauthorized use or disclosure to a third party. This category also covers client intellectual property where Virtusa has a non-disclosure agreement with the client.

# **1** **Introduction**

## **1.1** **Purpose**

The objective of the project is to deliver an efficient Courier Management System whose main functionality apart from calculating the courier Bill includes predicting the time required to reach the destination.

The system will be used for day to day activities like out return,company details,booking,non-delivery and pickup centres.

V2V Couriers is to automate the process by developing the software.The main purpose of this system is to connect all branches to the database.

## **1.2** **Scope**

Courier management is the incorporate of appropriate technology to help administrator manage information.The main aim of this project is to computerize the maintenance of courier management

## **1.3** **Definitions, Acronyms and Abbreviations**

[This sub-section must provide definitions for all the terms, acronyms, and abbreviations used in this document.]

## **1.4** **References**

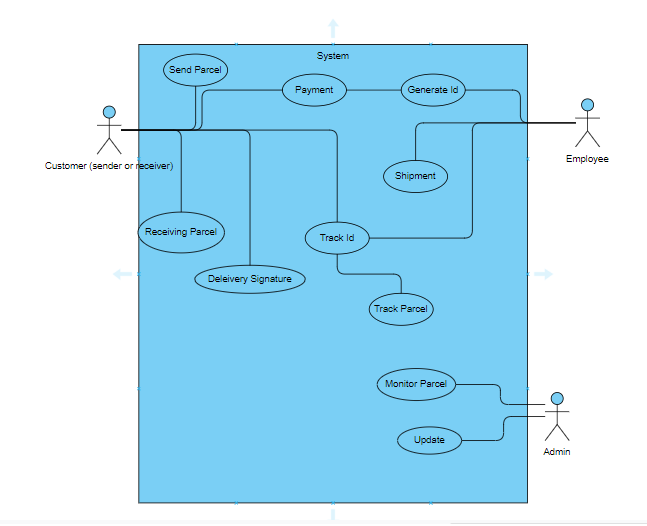
[This sub-section must provide a complete list of all references. Identify each document by title, document number and the version. Specify the sources from which the references can be obtained.

# **2** **Assumptions**

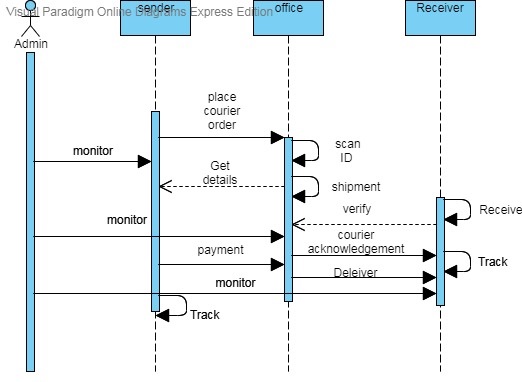
[List the assumptions, if any, here.]

# **3** **PROJECT OVERVIEW AND USECASE DIAGRAMS**

In this project, you can simply perform operations to manage the couriers and the parcels. This project intends to develop the concept of the working mechanism of the courier department. You can log in as the system administrator and also can add and delete the managers. You can also add the departments.



# **4** **Detailed Design for Module: <<V2V>>**



## **4.1** **Detailed Design for Feature: <<Feature Name>>**

### **4.1.1** **User Profile**

[Describe the different user types with the security profiles and specify the roles that can access these screens.]

### **4.1.2** **Navigation Map**

[Describe the navigation map along with use case traceability of all screens for this feature.]

### **4.1.3** **UI Screen Design**

|  |  |
| --- | --- |
| **Page Summary** | |
| **Name** | Search for Customer File |
| **Description** | The user retrieves the customer information based on the search criteria such as File ID, last name, first name, middle name, business name, SSN/TIN, address, city, state and/or zip. |
| **Use Case Number** | UC-002 |

#### *4.1.3.1* *Screen Layout*

[Include a screen shot and describe each element/button in the following table.]

#### *4.1.3.2* *Validations*

|  |  |  |
| --- | --- | --- |
| **Error Trigger** | **Action** | **Description** |
| If the user group is not selected. | Displays the message “Please select a User Group from the list”. | Prompts the user to select an entry from the User Group list. |
| User enters an invalid date | Displays the message “Enter a valid date. The date must be in <mm-dd-yyyy> format”. | Displays invalid date message. |

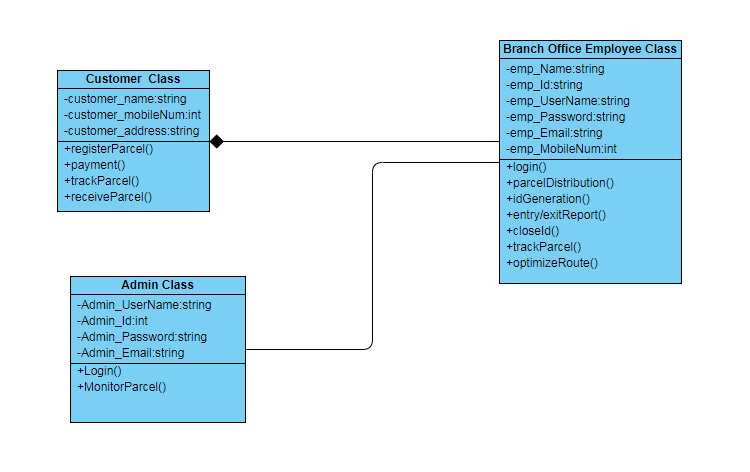
### **4.1.4** **Object Model/Data Flow Design**

[This sub-section describes the detailed design for the system pertaining to the <<Feature Name>> feature. Interaction diagrams showing the details of the component structure, behavior, or information/control flow may be included in the sub-section devoted to that particular component.]

#### *4.1.4.1* *Interface Description*

[Describe the interfaces including component interfaces and interfaces to other systems, products, or networks.]

#### *4.1.4.2* *Class Design*



##### **4.1.4.2.1** **Algorithmic/Business Logic Description (Optional)**

[Describe complex algorithms used in the class (if any).]

##### **4.1.4.2.2** **Local Data Structure (Optional)**

[Describe key data structure algorithms used (if any).]

#### *4.1.4.3* *Data Transfer/Value objects*

[Based on the design pattern being used, identify and describe the value/transfer objects.]

#### *4.1.4.4* *Related Database Tables*

[Give the list of related database tables.]

#### *4.1.4.5* *Dependencies with Other Sub-systems/Components*

[Describe the dependencies with other sub-system/components, if any.]

## **4.2** **Detailed Design for Feature: <<Feature Name>>**

Repeat all sub-sections given in 3.1 for this feature.

# **5** **Detailed Design for Module: <<Module Name>>**

[Detailed design for Module 2.]

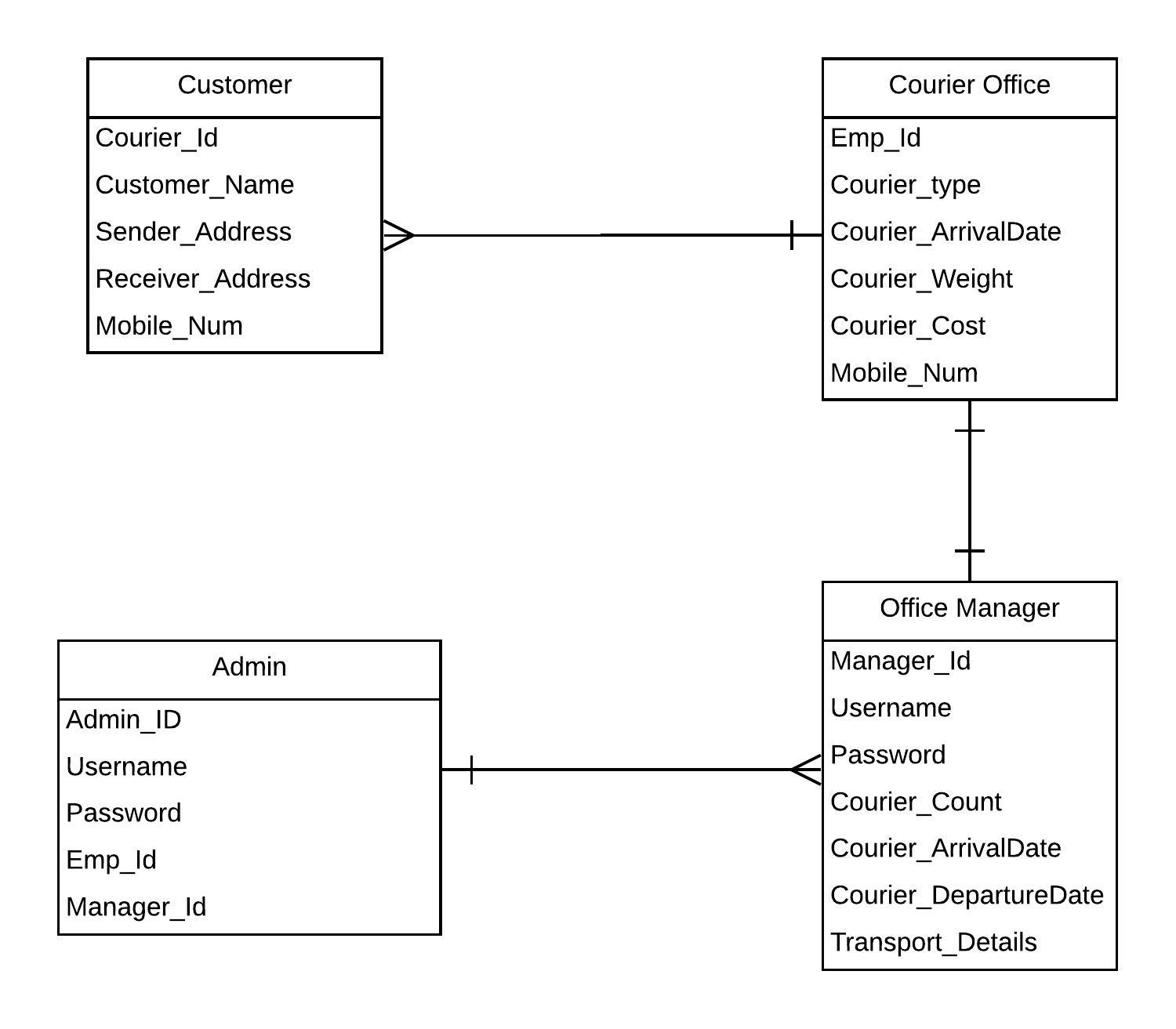
Repeat all sub-sections given in section 3 for this module.

# **6** **Database Design**

[This section describes the persistent data and metadata used and generated by the module/sub-system. This will include database schemas (if the database schema is large, use a separate document for database design), registry structures, property file structures, etc.

|  |  |
| --- | --- |
| **Models** | **Tools** |
| Database Entity Relationship Diagrams | ERWin |
| Object Models | Rational Rose |

## **6.1** **Entity Relationship Diagrams**



# **7** **Packaging/Folder Structure and System Artifacts**

[This section describes the file structure for the <Project Name> project, which will indicate where the code, unit tests, executables, libraries, etc. are to be placed during implementation. Optionally the file structure can be described in a separate GA chapter and a reference given here.]

**[Sample structure**

**End Sample Structure]**

## **7.1** **System Artifacts**

[Provide information on the Registry structure, INI files, property files, configuration files, etc.]

# **8** **Core Technical Services Design**

[This section describes the detailed design and usage pattern for the core technical services of the system.]

## **8.1** **Persistence**

[This sub-section captures the detailed design and usage patterns of the persistence service for the application.]

## **8.2** **Inter-process Communication**

[This sub-section captures the detailed design and usage patterns of the inter-process communication framework for the application.]

## **8.3** **Authentication and Authorization**

[This sub-section captures the detailed design and usage patterns of the security services for the application.]

## **8.4** **Error Handling**

This sub-section captures the detailed design and usage patterns of the error handling services for the application.]

## **8.5** **Logging**

[This sub-section captures the detailed design and usage patterns of the logging framework for the application.]

## **8.6** **Transaction Management**

[This sub-section captures the detailed design and usage patterns of the Transaction Management service for the application.]

## **8.7** **Other Applicable Technical Services**

[This sub-section captures the detailed design and usage patterns of the other core technical services to be handled by the system such as the installation mechanism, failure prevention, fault tolerance, caching design, internationalization, validation framework, client and server initialization, error handling, etc.]

# **9 Non Functional Requirements**

# **Appendix**

[Presents information that supplements the design specification.]

## **1.** **Design Metrics to be Used**

[A description of all design metrics to be used during the design activity is listed here.]

## **2.** **Supplementary Information**

[Provide as required.]

## **3.** **DQI – Design Quality Index Checklist**

This checklist shall be used to perform self-review as a reference while preparing software design.