Transport Company Computerization Software

Prepared By:
Nikhil Jaiswal
121CS1133
Computer Science and Engineering

Revision History

Date	Version	Description	Author
<20/01/2024>	<1.0>	SRS 1.0	Nikhil Jaiswal
<15/02/2024>	<2.0?	SRS 2.0	Nikhil Jaiswal

Table of Contents

1. Introduction

1.1 Purpose

The purpose of the TCC software is to computerize various bookkeeping activities associated with the operations of a transport company. The software aims to streamline the management of trucks, consignments, billing, and other related activities to improve efficiency and accuracy in the company's operations. It also aims to provide critical data which when analyzed will help to take important decisions for the phosphorus future of the company.

1.2 Scope

The scope of the Transport Company Computerization (TCC) software pertains to the computerization of major processes within a transport company, aimed at improving administrative efficiency, speed, and accuracy. The software will be deployed across various branches of the transport company to centralize data related to consignments, trucks, and employees, facilitating functions such as truck allocation, calculation of waiting time for consignments, and other administrative tasks.

1.3 Definitions, Acronyms and Abbreviation

TCC	Transport Company Computerization
UI	User Interface

1.4 Reference

a. https://www.geeksforgeeks.org/how-to-write-a-good-srs-for-your-project/
b. https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document

1.5 Overview

The TCC software is designed to streamline the management of a transport company's operations by providing features for managing consignments, trucks, billing, and branch information. The software will offer real-time tracking and reporting capabilities, enabling managers to make informed decisions and improve overall operational efficiency.

2. Overall Description

This section describes the functions of the software, its operating environment, some basic but valid assumptions and dependencies. It also tries to bring out the design and implementation constraints in the development of the software.

2.1 Products Functions

This software will help to make the management and administrative processes of a Transport company faster and efficient. The functionalities of the software are as follows:

- The software will be able to store the details of consignment, compute the transport charge and issue bill for the consignment.
- It will be able to automatically allot the next available truck as the consignment for a particular destination exceeds a certain limit.
- It will be able to manage the number of trucks and store its details and show the status of trucks as well as consignments at a given time.
- It should be able to store the details of the customer.
- It will also be able to compute the average waiting time for consignments and the idle time of a truck.
- It will be able to maintain the information about all the branches.
- Passwords and user ID will be used to protect the accounts of employees and managers.
- The software should be able to track all the orders and store all the details of the orders.
- The software should be able to show all the data stored in an easy to visualize manner.
- The software should be able to manage the data access level of all the employees and managers.

2.2 Operating Environment

The software is a web application made using Flutter that also makes use of a database. It must be designed to work flawlessly and without issues on a Linux (Ubuntu) and Windows computer having internet access.

2.3 Design and Implementation Constraints

The major constraints in the development of the software:

- Computers at various centers must be able to communicate in real time. For that internet connection is required.
- Good form of integration between the databases.
- Limited amount of memory can cause issues if the database is too large.
- The software will use passwords for login. The security of the software depends on password protection and on network communication.
- The algorithm following will not be an optimized one, as an optimized algorithm will be too much computationally heavy. However, it will give fairly good results in most cases.

2.4 Assumptions and Dependencies

The software will be made with the following assumptions:

- Internet connection is well available in all the branches and the computers there can communicate with each other in real time.
- Each user must remember his password and login ID, failing which, he cannot login into the system. The manager will be the only one to have the right to set and reset the password.
- Users should not tamper/experiment with the source code/executable file of the software.
- The user should have a good knowledge of the basic attributes of an object and fill in the details of the objects and employee properly.
- The centers of the Transport Company are well distributed in the map and each center performs well in terms of consignment handling (that is there is no center which only receives goods but does not send any or vice versa).

The main dependencies of the working and performance of the software are:

- The internet connection should be good enough for the computers to communicate with each other and send data to the central machine of the manager.
- All the tools on which the software is dependent must be working properly.
- The software will also depend on the database server.

3. Specific Requirements

3.1 External Interfaces

3.1.1 User Interface

The user interface of the software will be easy to use and interactive. Each person will have to login using his own login id and password. Only after that he will be able to make any changes to the database or have his/her queries answered.

3.1.2 Hardware Interface

The storage of the data on the physical drive will depend on the tools used for the development of software. The software will run properly on a computer having support for the database to be used. The computer should have a minimum of 2GB RAM (preferably 4GB or more) and 20GB free space (preferably 50GB or more). More memory may be required if the database is too large.

3.1.3 Software Interface

The system shall integrate with financial systems for payment processing, ensuring accurate billing and invoicing. Integration with address databases for route generation, enabling efficient truck allocation and consignment management.

3.1.4 Communications Interfaces

All information regarding the trucks and consignments are sent through networks. So, the computers at different times and the central machine must be able to communicate securely and quickly over the network. The software must take care of the communication protocol to be used or the encryption to be followed to ensure secure communication among different branches.

3.2 Functionality

3.2.1 Consignment Management

The software should be able to store details of consignments, including sender, receiver, destination, weight, and other relevant information.

It should compute transport charges based on consignment details and issue bills accordingly.

It should automatically allot the next available truck when a consignment for a particular destination exceeds a certain limit.

It should compute the average waiting time for consignments.

3.2.2 Truck Management

The software should manage the number of trucks and store details such as truck number, capacity, status, and location.

It should show the status of trucks and consignments at any given time.

It should compute the idle time of a truck.

3.2.3 Customer Management

The software should be able to store customer details, including name, contact information, and billing information.

3.2.4 Branch Management

It should maintain information about all branches, including location, contact details, and manager details.

3.2.5 Security

Passwords and user IDs should be used to protect the accounts of employees and managers.

The software should manage data access levels for employees and managers.

3.2.6 Order Tracking

The software should track all orders and store details such as order number, consignment details, and status.

3.2.7 Data Visualization

It should show all data stored in an easy-to-visualize manner, such as graphs or charts.

3.2.8 Integration and Communication

Computers at various centers must be able to communicate in real-time, requiring a good internet connection.

Integration between databases should be seamless.

The software should handle communication protocols and encryption for secure communication among different branches.

3.3 Supportability

The system shall be designed for easy maintenance and updates, ensuring minimal downtime and disruption to operations.

3.4 On-line User Documentation and Help System Requirements

The system shall provide online help documentation for users, including tutorials and FAQs, accessible from within the software interface.

3.5 Legal, Copyright, and Other Notices

The system shall display disclaimers, copyright information, and relevant notices, ensuring compliance with legal requirements and regulations.

3.6 Applicable Standards

The system shall adhere to industry standards for data security and software development, ensuring the software meets best practices and guidelines.

4. Requirements

4.1 Functional Requirements

S.No.	Requirements	Description
1.	Consignment Management	The system must store and manage details of consignments, compute transport charges, and issue bills for consignments.
2.	Truck Allocation	The system shall automatically allocate the next available truck when a consignment exceeds a certain limit for a particular destination.
3.	Truck Management	The system shall manage the number of trucks, store truck details, and show the status of trucks and consignments at any given time.
4.	Customer Management	The system shall store customer details, including name, contact information, and billing information.
5.	Reporting	The system shall provide reporting capabilities, including generating reports on consignment status, truck utilization, and branch performance.
6.	User Access Management	The system shall manage data access levels for employees and managers, ensuring data security and privacy.
7.	Online Help Documentation	The system shall provide online help documentation for users, accessible from within the software interface.
8.	Integration with Financial Systems	The system shall integrate with financial systems for payment processes, ensuring accurate billing and invoicing.
9.	Integration with Address Database	The system shall integrate with address databases for route generation, enabling efficient truck allocation and consignment management.

4.2 Non-Functional Requirements

S.No.	Requirements	Description
1.	Reliability	The system shall ensure reliable back-end processes for daily operations, minimizing downtime and disruption.
2.	Performance	The system's performance shall be efficient, ensuring timely processing of consignments, truck allocations, and billing computations.
3.	Security	The system shall handle customer and financial information securely, using encryption and access controls to protect sensitive data.
4.	Scalability	The system shall be designed to scale as the transport company grows, handling increasing data volumes and user interactions.
5.	Maintainability	The system shall be designed for easy maintenance and updates, ensuring minimal disruption to operations.
6.	Usability	The system shall be user-friendly and intuitive, requiring minimal training for employees and managers to use effectively.
7.	Compatibility	The system shall be compatible with standard computing hardware and software, ensuring optimal performance and compatibility.
8.	Legal Compliance	The system shall comply with relevant legal and regulatory requirements, including data protection and privacy laws.
9.	Documentation	The system shall be well-documented, including user manuals, technical documentation, and system architecture documentation.
10.	Training and Support	The system shall provide user training materials and support services to help users understand and use the system effectively.

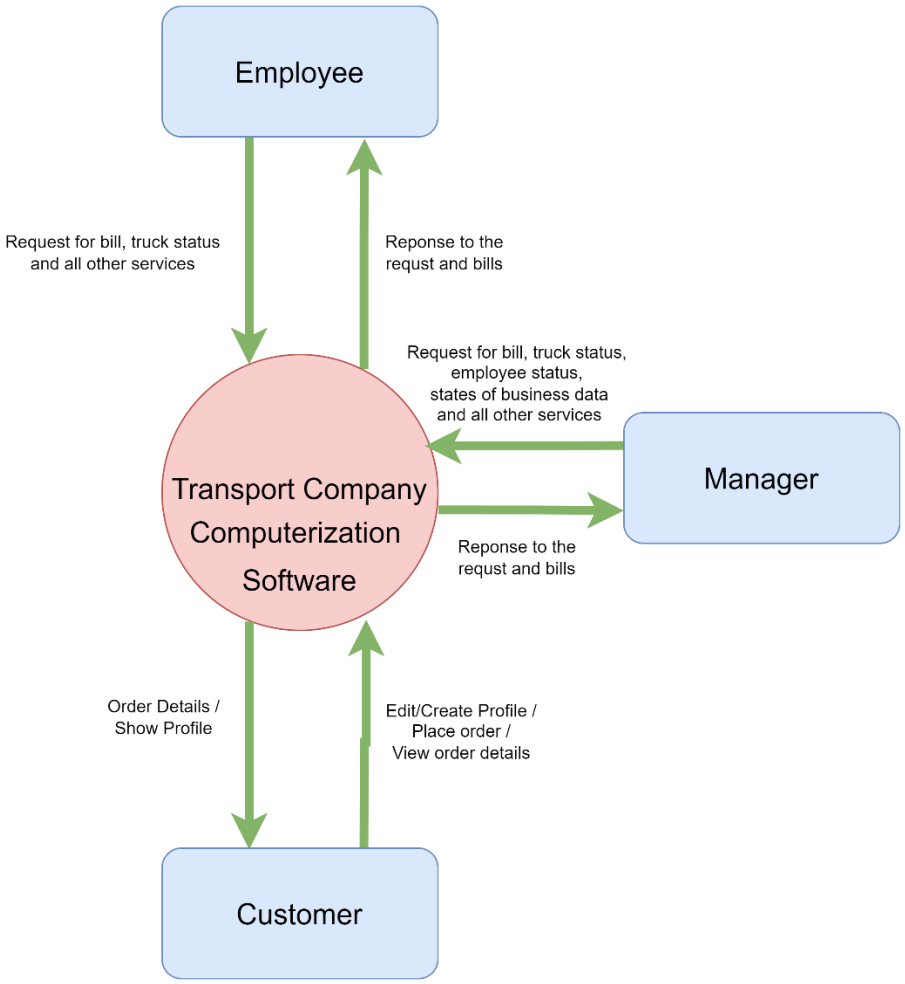
.

5. Data Flow Diagrams (DFD)

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one.

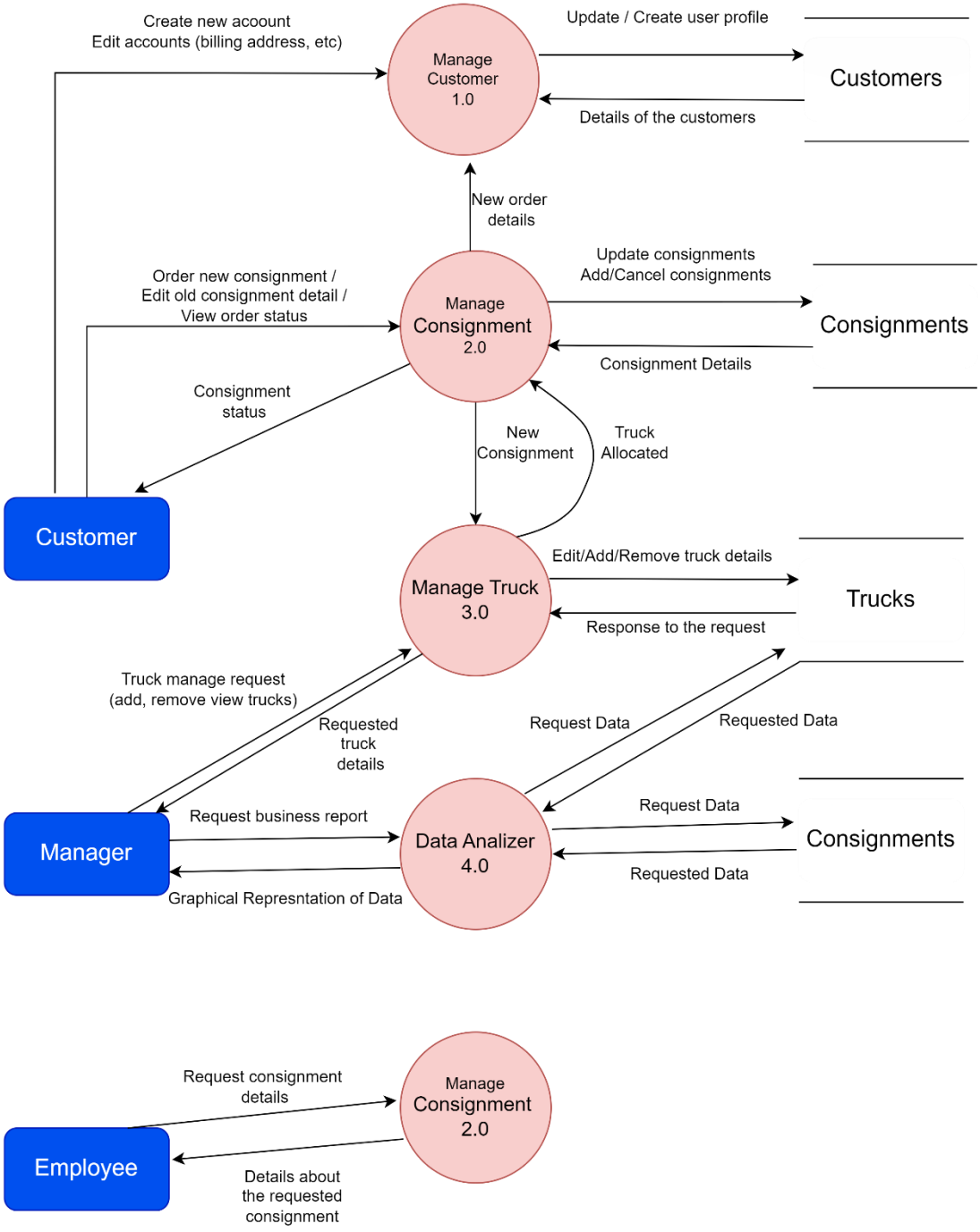
5.1 Level-0 DFD

Level 0 DFD is also known as Context Diagram. The DFD provides a high-level view of the TCC software system, showing its relationship with external entities such as users and databases. It illustrates how data flows into and out of the system at a broad level.



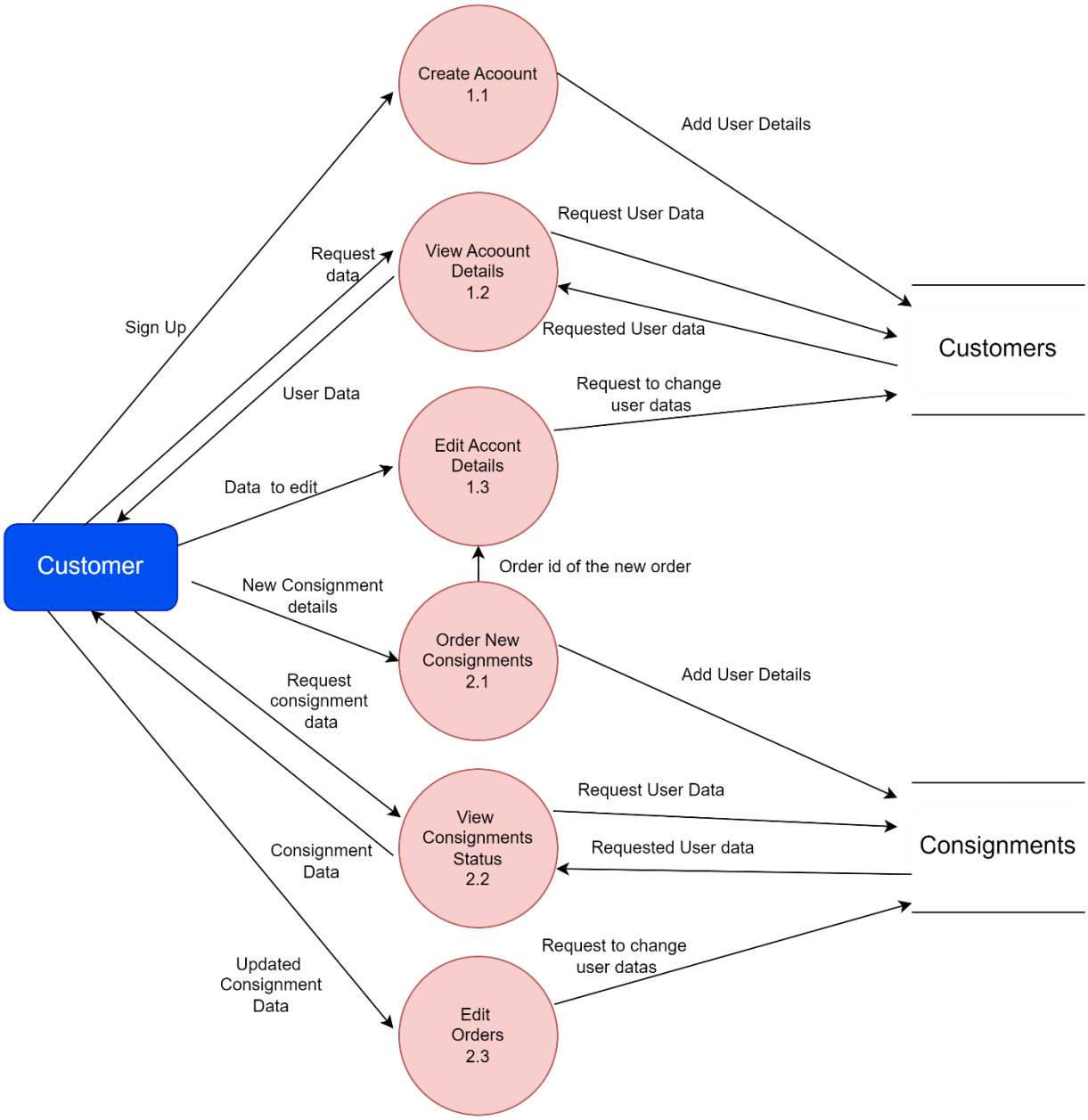
5.2 Level-1 DFD

DFD Level 1 provides a more detailed breakout of pieces of the Context Level Diagram. You will highlight the main functions carried out by the system, as you break down the high-level process of the Context Diagram into its sub processes.



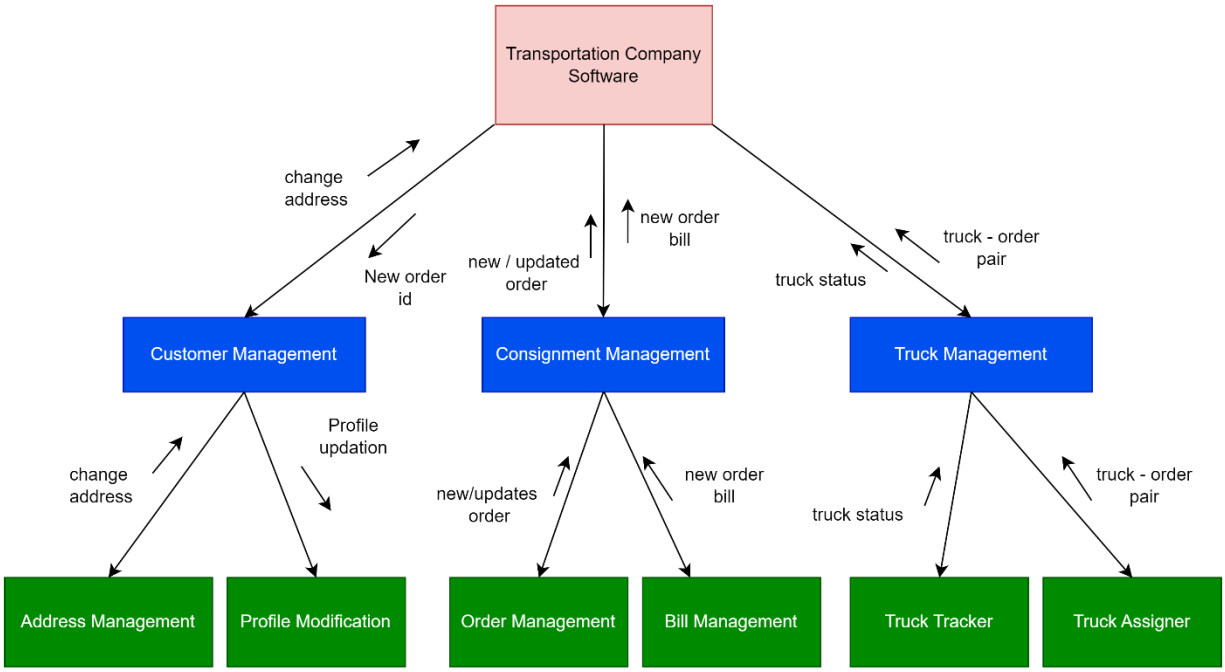
5.3 Level-2 DFD

This level provides an even more detailed view of the system by breaking down the sub-processes identified in the level 1 DFD into further sub-processes. Each sub-process is depicted as a separate process on the level 2 DFD.



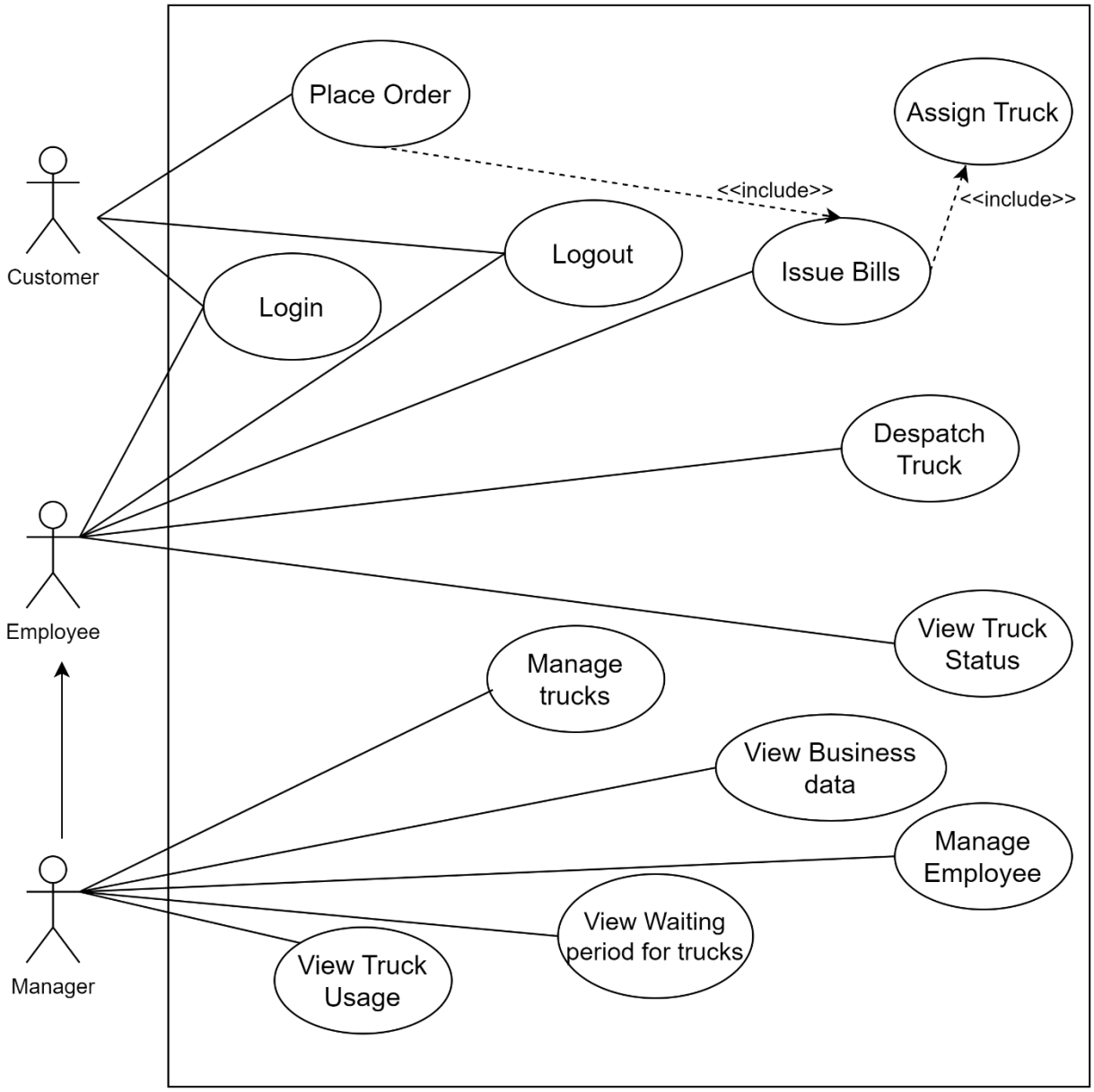
6. Structure Chart

Structure Chart represents the hierarchical structure of modules. It breaks down the entire system into the lowest functional modules and describes the functions and sub-functions of each module of a system in greater detail.



7. Use Case Diagram

A use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behavior , and not the exact method of making it happen. Use cases once specified can be denoted both textual and visual representation.



8. Class Diagram

A class diagram in the Unified Modeling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects.