Report on

**Transport Management System**

1. **Project Description**
   1. **Link to Github repository**

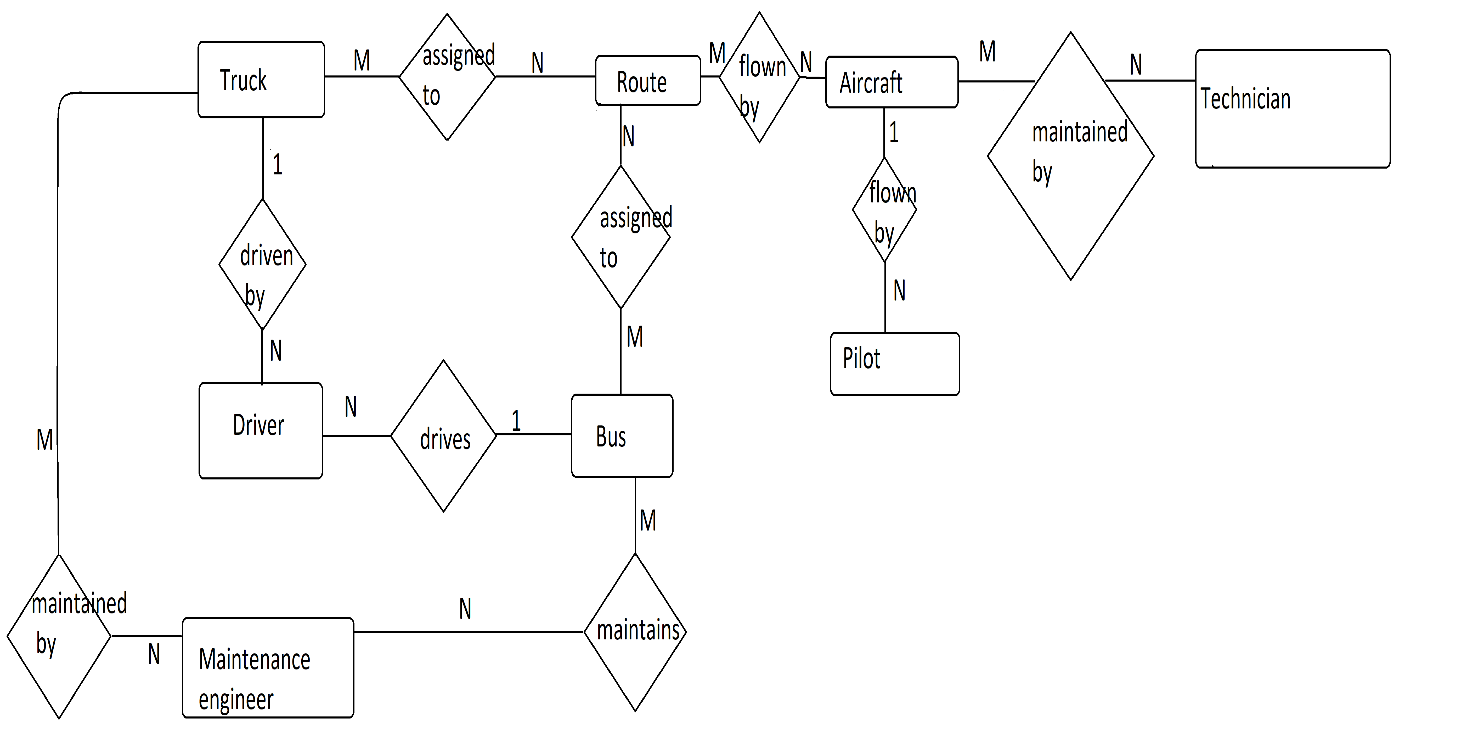
http://github.com/raj211p/logistics\_management\_app

In this modern era, where everything is heavily reliant on technology, there are many opportunities to develop unique applications which can solve real-world problems faced by customers every day and achieve the functionality that they desire using a real time system. Our project aims to satisfy those goals.

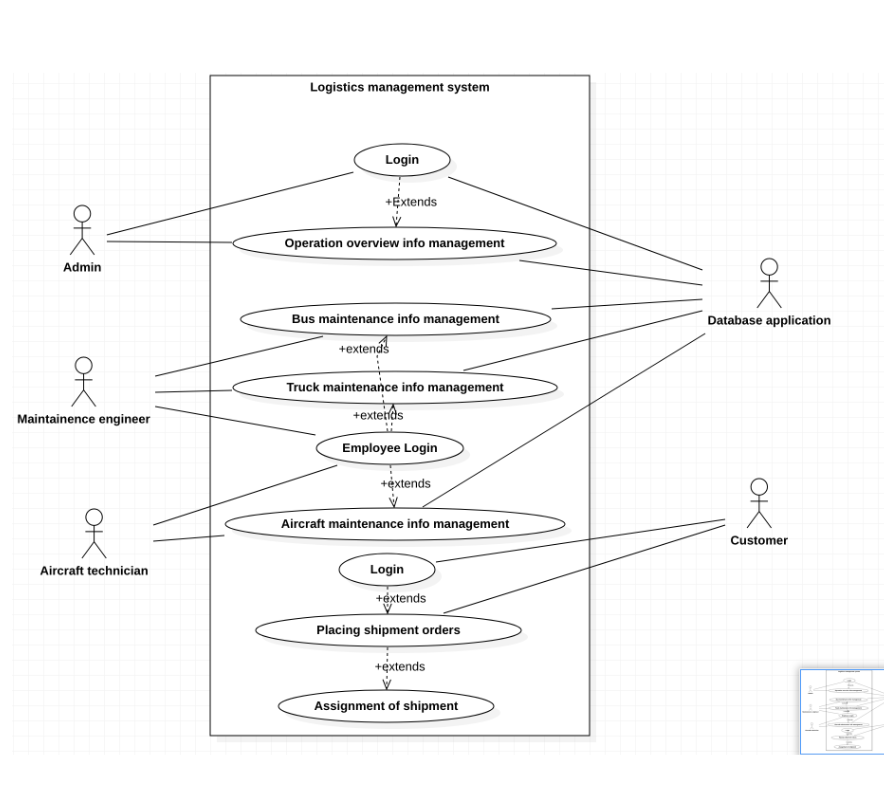
The aim of our project is to build a web application using Java Server Pages and MySQL to help manage the day-to-day operations of a logistics company. It will be capable of performing all the standard CRUD operations and giving users an overview of how well the company is performing with detailed metrics. It will also allow customers to place orders for the shipment of goods

1. **Analysis and Design Models**

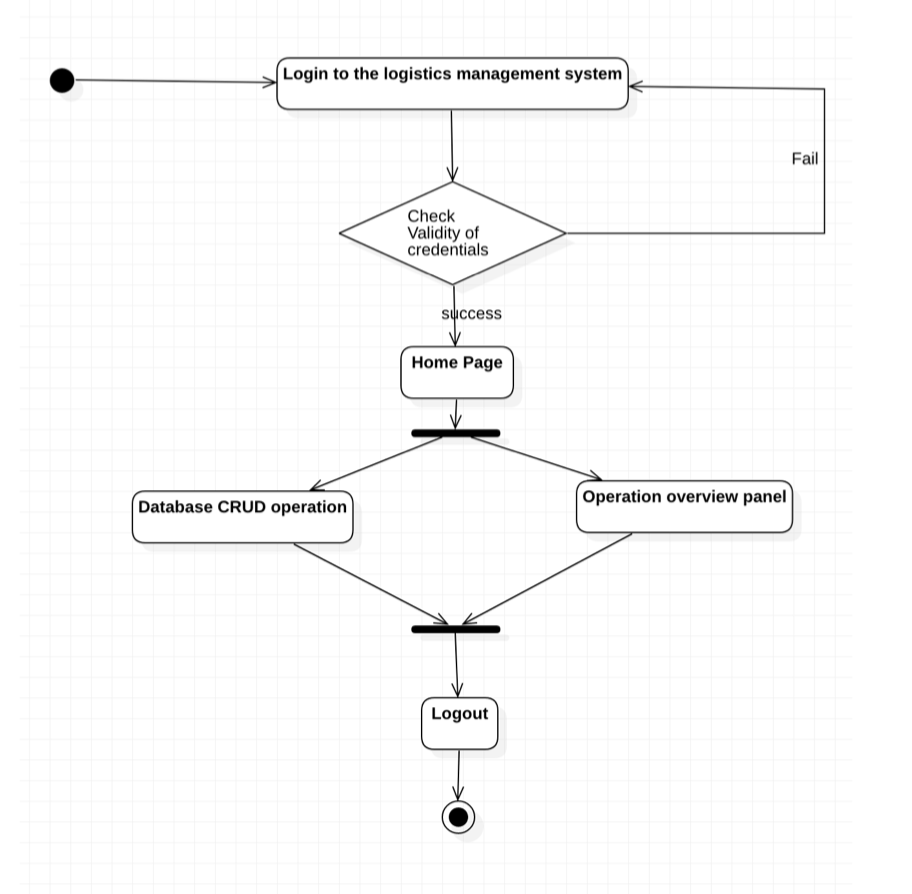
**2.1 ER diagram**



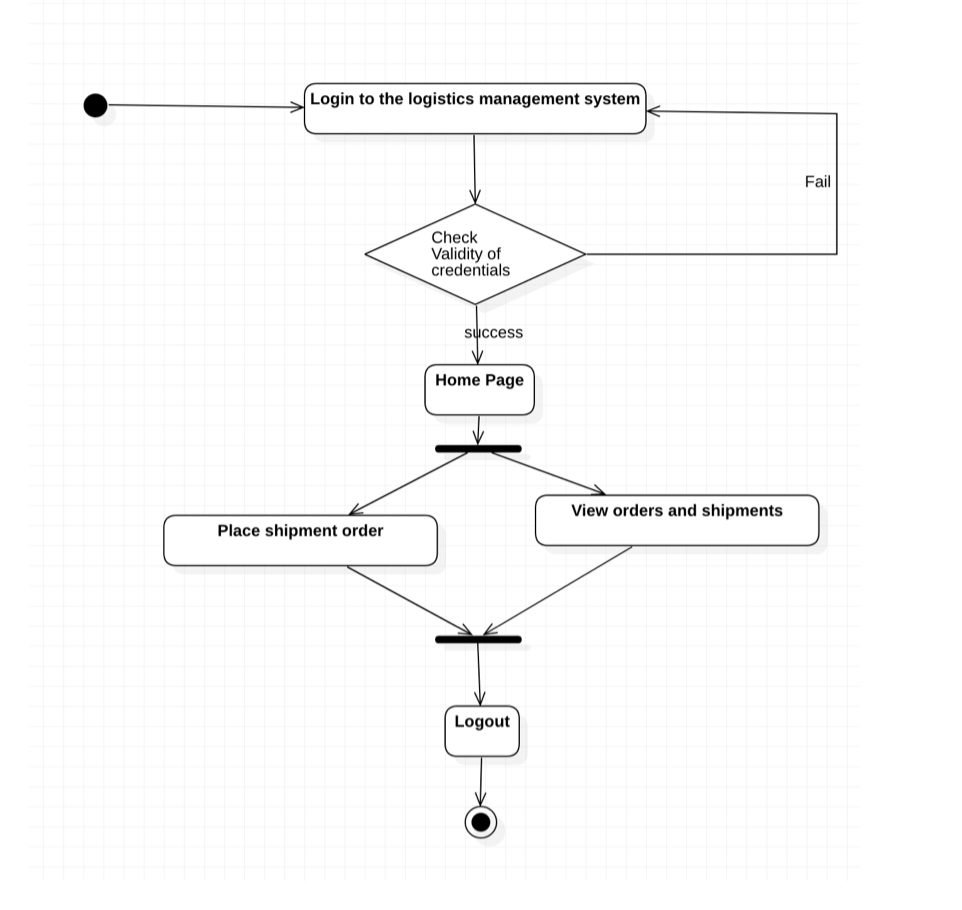
**2.2 Use case diagram**



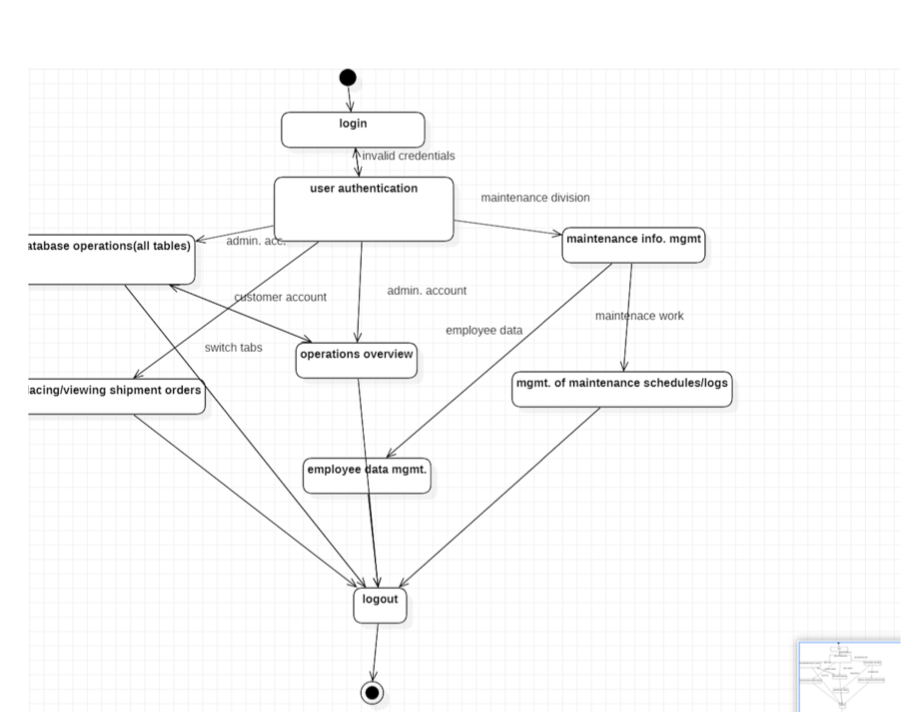
**2.3 ACTIVITY DIAGRAM FOR EMPLOYEES**



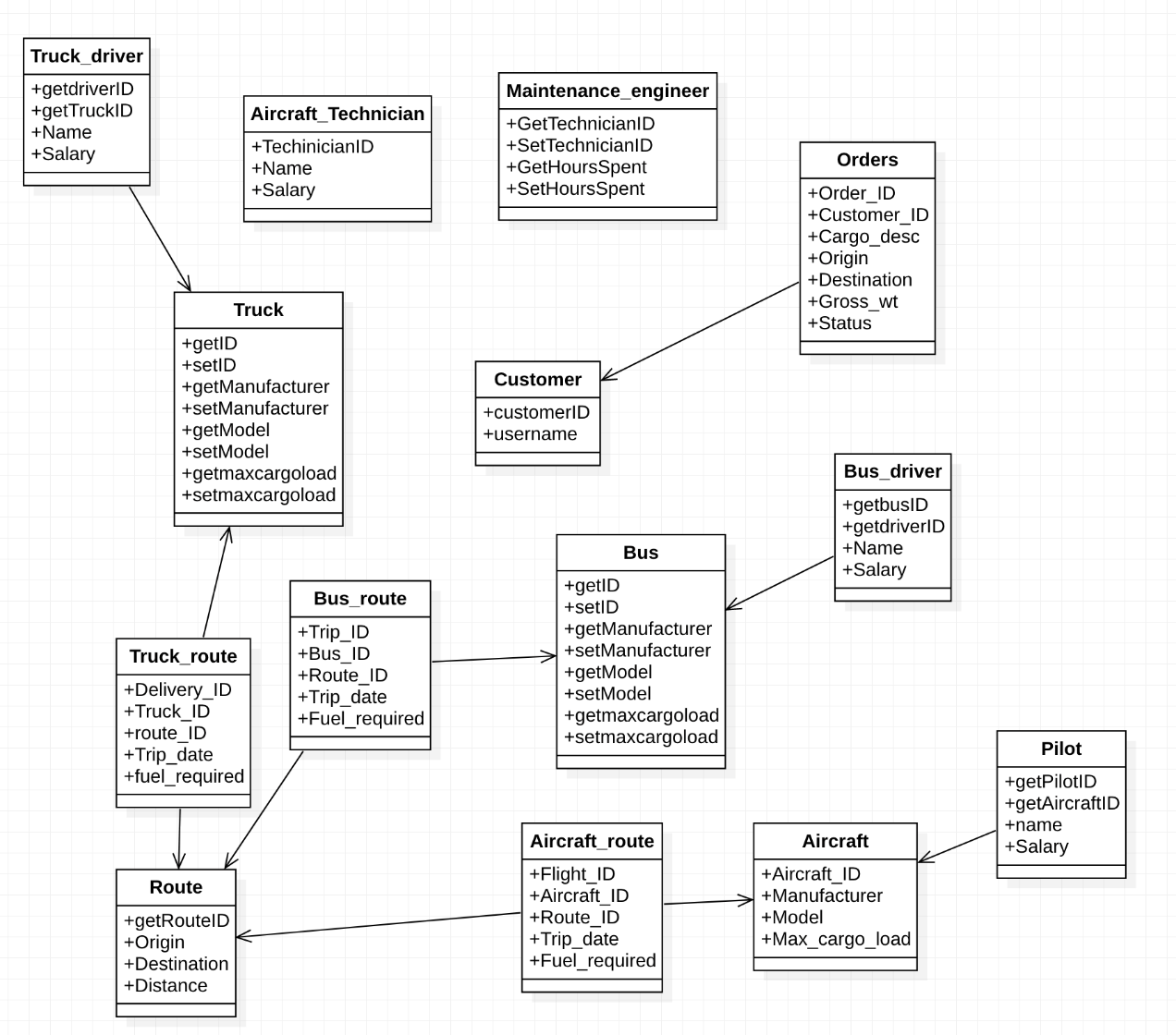
**2.4 ACTIVITY DIAGRAM FOR CUSTOMERS**



**2.5 STATE DIAGRAM FOR ADMIN**



**2.8 CLASS DIAGRAM**



**3.Tools and Frameworks Used**

1)Eclipse Java EE: Version: For the development of web - based solutions, Eclipse's enterprise edition allows developers to employ Servlet, Java Server Pages (JSP), and other related tools.

It works best with Java Enterprise Edition, which is intended specifically for building online and enterprise projects.

2)Tomcat 9.0.55:

A free Java servlet container that supports a variety of Java Enterprise Specs, including the Websites API, Java-Server Pages, and, of course, the Java Servlet.

3) MySQL : MySQL is a SQL (Structured Query Language)-based relational database management system. Data warehousing, e-commerce, and logging applications are just a few of the uses for the application. However, the most popular application of MySQL is as a web database.

4)OpenJDK 17: **It** is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source_software) implementation of the Java Platform.

**Design Principles and Design Patterns Applied**

**1)Model View Controller(MVC):** MVC is an [application](https://techterms.com/definition/application) design model comprised of three interconnected parts. They include the model ([data](https://techterms.com/definition/data)), the view ([user interface](https://techterms.com/definition/user_interface)), and the controller ([processes](https://techterms.com/definition/process) that handle input).

Model: model is data used by a program. This may be a [database](https://techterms.com/definition/database), [file](https://techterms.com/definition/file), or a simple object

View: A view is the means of displaying objects within an application

Controller: A controller updates both models and views. It accepts [input](https://techterms.com/definition/input) and performs the corresponding update.

**2)Single class responsibility:** The Single Responsibility Principle states that a class should have one and only one reason for the change.

The benefits of SRP include:

* + Reduction in complexity of code
  + Increased readability, extensibility, and maintenance
  + Reusability and reduced errors
  + Better testability

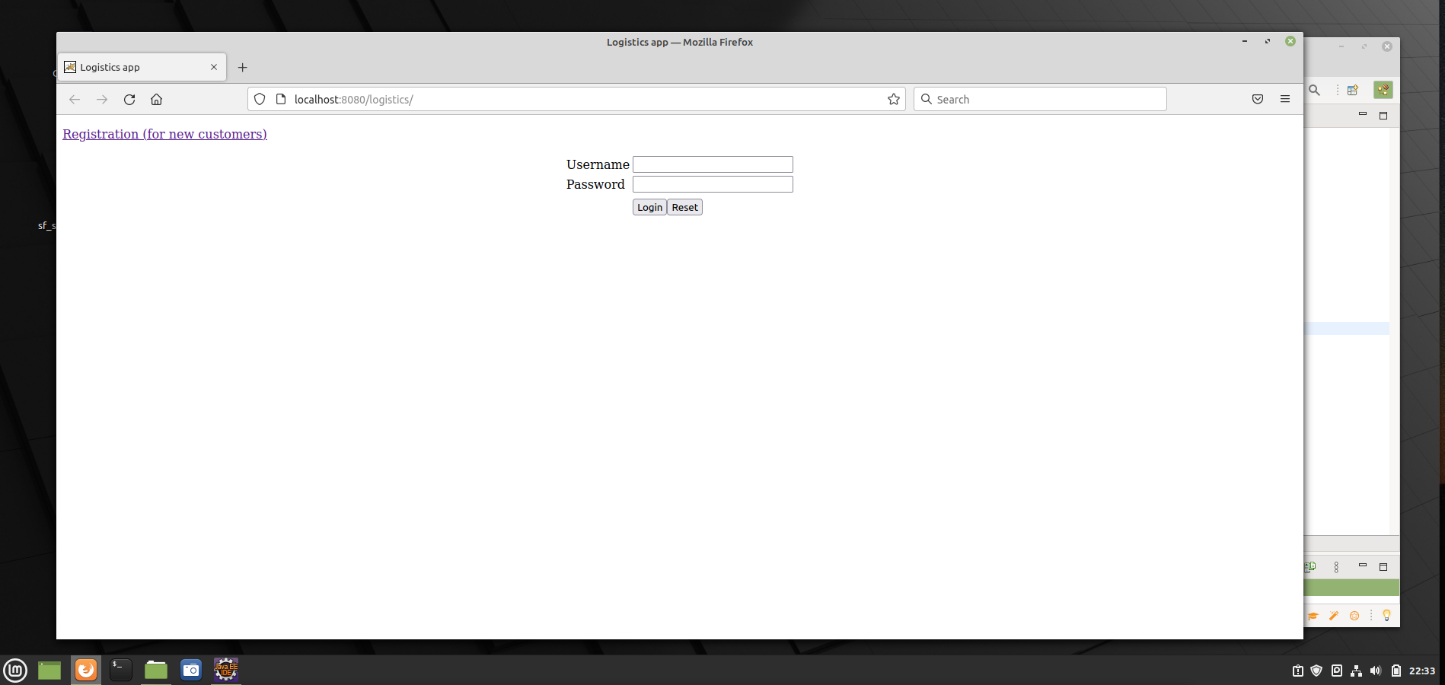
**3)Separation of roles of user based on privileges**

**4)Client server pattern:** This pattern aids in the creation of distributed systems that include a client, a server, and a connected network.

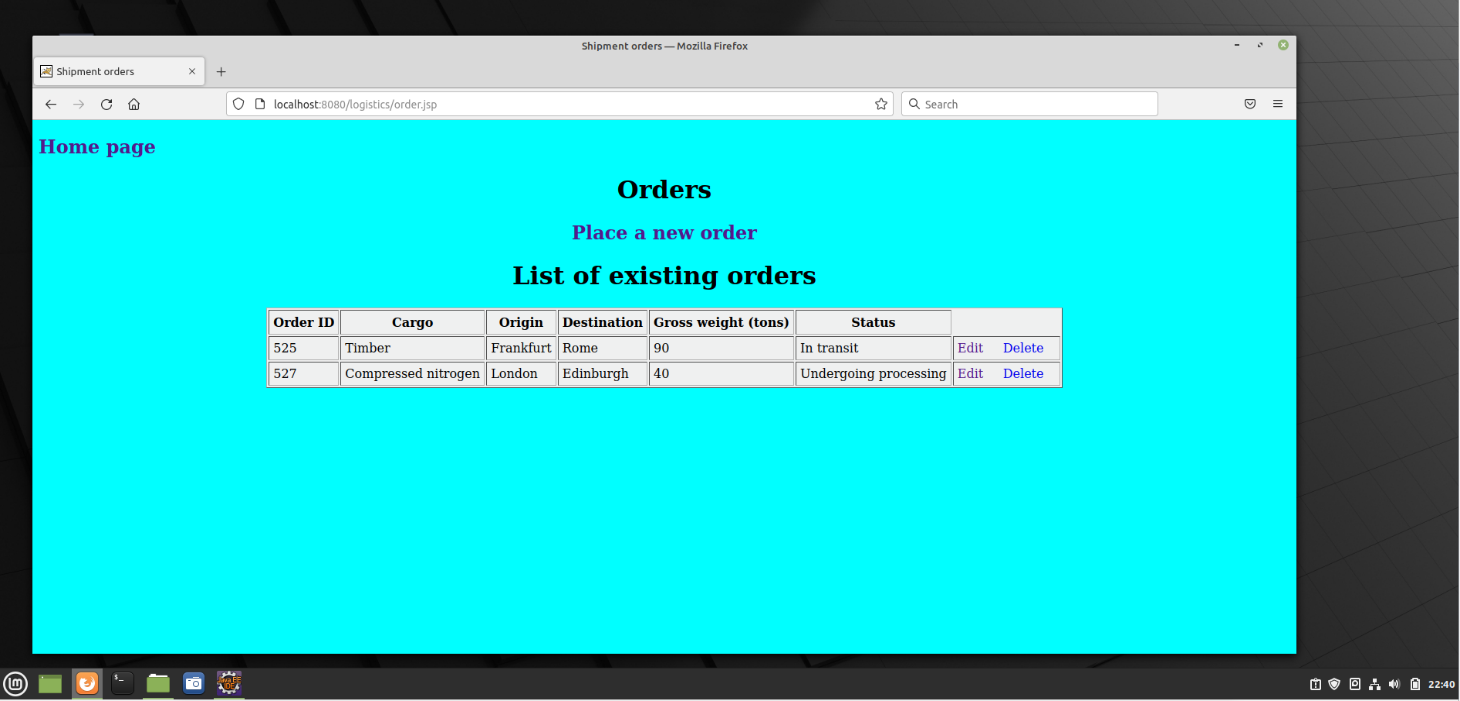
A server application is accessed directly by numerous clients in the simplest type of client/server architecture.

1. **Application Screenshots**

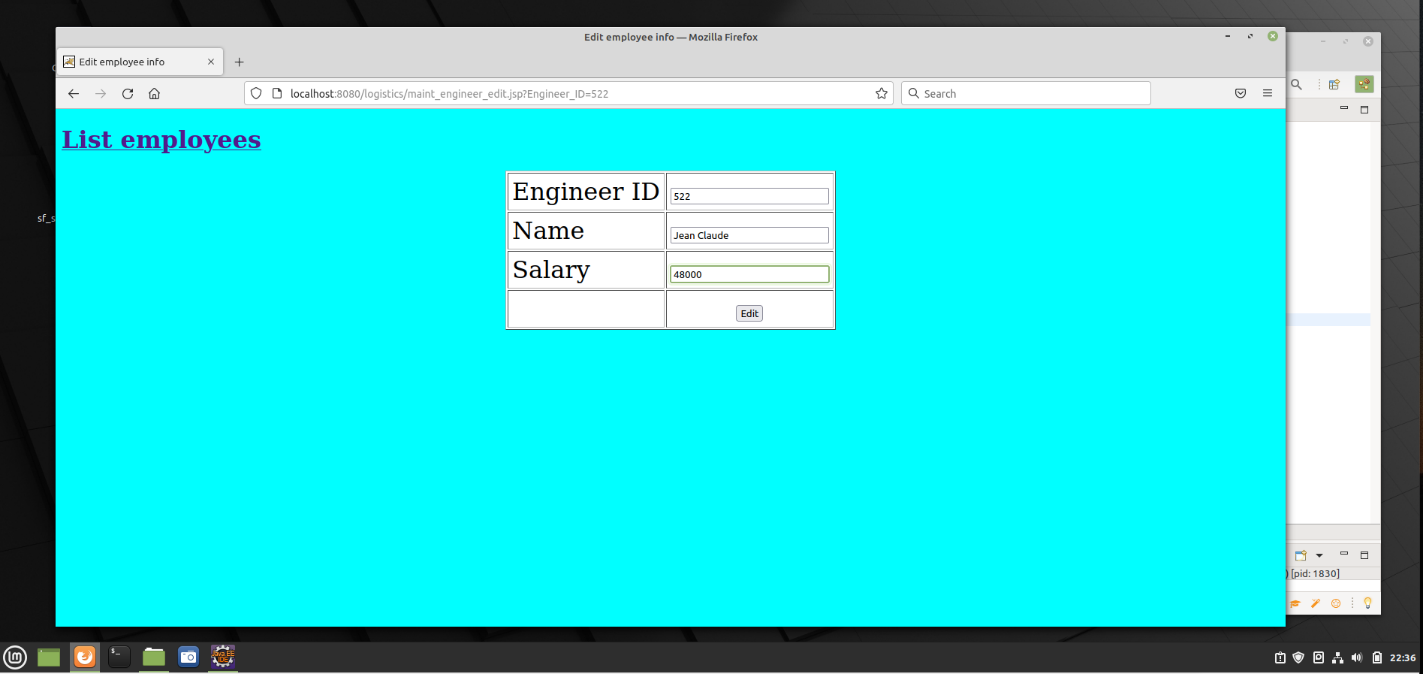
**4.1 Login page**



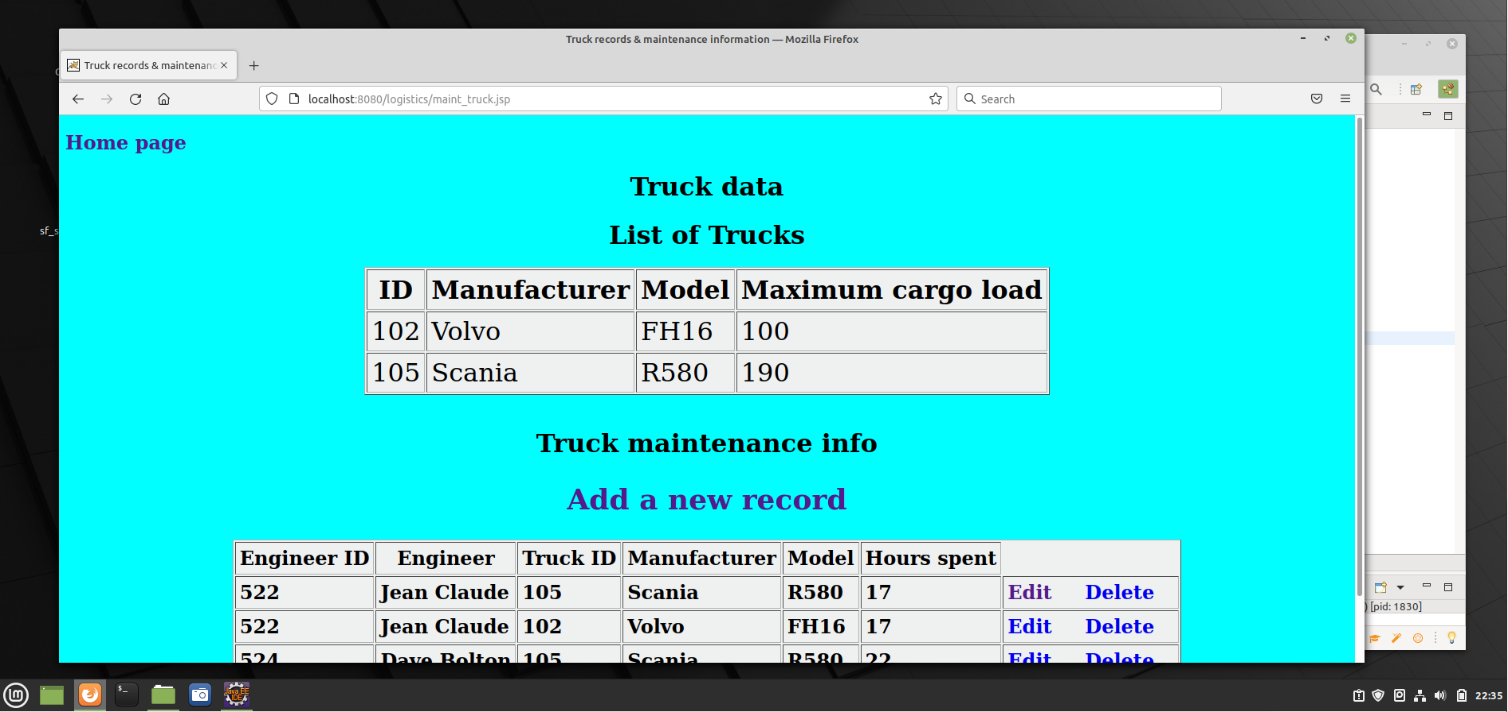
**4.2 View orders**



**4.3 Edit salary**



**4.4 Truck info**



**4.4 Operations Overview**

