**Project Title: Cab Management System**

**Introduction:**

The Cab Management System is designed to simplify and automate the process of managing cab services. The system provides a platform for managing cabs, drivers, and bookings, allowing users to book cabs, and manage their bookings.

The system enables users to select their pickup and drop-off locations, choose the type of cab, view fare estimates. The system helps streamline the cab management process.

**Technologies Used:**

* Core Java
* Spring Boot + Rest Controller + JPA with Hibernate
* Angular for UI design (Front End)
* Postgres (or any other preferred relational database management system)
* Maven (for project management)

**System Features:**

* User Registration and Authentication:
* The system allows users to register and create accounts with secure authentication mechanisms.
* Cab Management:
* The system provides features to manage cab details.
* Customers can easily book cabs using the system, specifying pickup and drop-off locations, desired cab type, and other preferences. The system facilitates a smooth and efficient booking process.
* Driver Management:
* The system provides features to manage driver details, including their availability, assigned cab and other associated attributes.
* Fare Calculation:
* The system calculates the estimated fare based on the distance between pickup and drop-off locations.
* Booking Management:
* Users can view and manage their bookings.
* Users can view their booking history.
* Ride Management:
* Drivers starts the ride as soon as passenger boards the cab. Driver ends the ride once the passenger is dropped at the target location.
* Reporting Management:
* The system provides various reports related to cab and driver availability, bookings, ride, and other indicators.
* Admin Panel:
* Administrators have access to an admin panel for managing cabs, drivers, users, booking, rides.

**Testing:**

* Unit testing is performed to ensure the correctness of business logic and functionality.

**Project Directory structure:**

* The src folder contains the source code for the project.
* The main folder contains the main codebase of the application.
* The java folder is where you place your Java source code.
* The <com>.<projectname>.<packagename> is a suggested package structure.
* The controllers package contains the controllers that handle the HTTP requests.
* The entity package contains the entity classes that represent the database tables, such as Cab, Driver, and TripBooking etc.
* The repositories package contains the repository interfaces for database operations, such as CabRepository, DriverRepository, and TripBookingRepository etc.
* The services package contains the service classes that implement the business logic, such as CabService, DriverService, and TripBookingService etc.
* The security package contains the security configuration for authentication and authorization, such as SecurityConfig.
* The resources folder contains non-Java resources such as properties files.
* The application.properties file holds the configuration for the main application.
* The test folder contains the test code for unit testing.
* The application-test.properties file holds the configuration for testing.

Remember, this is just a suggested directory structure, and you can customize it based on your project requirements. It's important to follow proper package and class naming conventions and organize your code in a way that is maintainable and scalable.