Delivery management system

Functional Requirements

The delivery management component is developed to deliver stock to the customer, manage delivery and order information, and solve issues. The deliveryman will deliver the goods to the customer, and he can log into the system with a unique account type.

This component is developed based on Deliveryman. Also, the customer can see the bundles they have ordered in the delivery list.

Delivery persons access the system.

After the deliveryman logs into the system, he will see the orders to be delivered today. All the information will be displayed in this list. (Delivery time, Date, Address). He can also view his salary vehicle details and payment information through the profile.

Track the location.(Google map API)

This task is done by the deliveryman. Through this, the distance to the location to be delivered and the speed at that the vehicle can go to that location is tracked using satellite technology and given to the delivery person.

Update the information about the delivery.

The information should be updated after the delivery person has delivered the relevant consignment to the customer. It includes the time, date, and place where the goods were delivered, the name of the person who received them, and the national identity card number.

Display the delivery details.

This information is displayed on the delivery manager’s dashboard. He can see the orders delivered today and the orders to be delivered. This information is regularly updated by the delivery persons. And the customer can also see the delivery list through his profile. He can see it by using the search box and filtering it.

Generate report using delivery details.

The data displayed above should be downloadable separately as a report(Daily and Monthly). These can be analyzed and problems that arise can be resolved.

Notify customers using email.

Once the relevant inventory is delivered to the designated location, an email message will be sent to the customer through the system. And if the delivery is canceled, they will also be informed to the customer through an email.

Non-functional Requirements

Reliability- The delivery management component rests entirely on data. Therefore, customers’ IDs, names, and addresses should be secured. Also, delays or downtime can impact delivery schedules and customer satisfaction.

Scalability-As the delivery management system grows and expands, it must be able to handle increasing volumes of orders, drivers, and customers. The system should be designed to scale up or down as needed to meet changing demands, without impacting its performance or functionality.

Security- A delivery management system must be secure to protect sensitive data such as customer information, order details, and driver records. The system should be designed to prevent unauthorized access, ensure data privacy, and protect against potential cyber threats such as hacking or data breaches.

Usability-Most of the drivers in society do not have significant technical knowledge. So, the delivery management system should be designed to be easy to use, with a user-friendly interface and intuitive navigation. The system should be accessible to all users, regardless of their technical knowledge or experience, and should support multiple

Performance-The delivery management system must be designed to perform efficiently and quickly, with fast response times and minimal delays. The system should be able to handle high volumes of requests and transactions without slowing down or crashing and should provide real-time updates on the delivery status and driver locations

Ex:-Notify the customer . if this happens a day late. It will be useless.

Technical Requirements

GPS tracking-The delivery management system must be able to use GPS technology to track the location of delivery vehicles and provide real-time updates on delivery status. The system should be able to display driver locations on a map, calculate estimated arrival times, and adjust delivery routes based on traffic conditions and other factors.

Mobile optimization-The delivery management system should be optimized for use on mobile devices such as smartphones and tablets, as many delivery drivers use these devices to manage their deliveries. The system should be able to support multiple platforms and screen sizes, with a responsive design that adapts to different devices and orientations.

API integration-The delivery management system should be able to integrate with external APIs (Application Programming Interfaces) to allow for data exchange and communication with third-party systems.

Data security and privacy-The delivery management system should be designed with robust data security and privacy measures, such as encryption, authentication, and access controls, to protect sensitive data from unauthorized access or disclosure.

QR Code Scanners-QR codes can be printed on packages to uniquely identify them and link them to specific delivery orders. Delivery drivers can use a QR code scanner to quickly and accurately identify the correct package to deliver, reducing the risk of errors or mis deliveries.

Users of features and their operations

Delivery manager-

1.Delivery manager login to the system.

2.he can edit his profile.

3.he assign orders to deliverymen.

4.generate reports using delivery details

5.Finally, login-out

Deliveryman

1.Deliveryman login to the system.

2.get order list and find the first order.

3.Using the map and tracking it.

4.delivered stocks successfully and update the delivery details to the system.

5.System send email to the customer.

6. Login-out the system

Customer

1.Customer login to the system.

2.he can also edit his profile

3.View delivery list.

4.search his order and check (Successful or unsuccessful)

5.Login-out the system

.