National University of Computer & Emerging Sciences
Karachi Campus

SOFTWARE ENGINEERING

PROJECT REPORT

MUHAMMAD AALYAN 21k-3432
Maarij Waseem 21K-4547
Huzaifa 21K-4948
Sec: J

INDEX

1) Project Requirements.....

1.1) Functional Requirements.....

1.2) Non-Functional Requirements.....

2) List of IDEs/Tools/Language.....

3) Task Distribution.....

4) Project Gantt Chart.....

Project Requirements:

Functional Requirements

The food ordering and delivery management system has following components that has their respective functional requirements.

The Ordering System

Users of the web-application must be able to do the following tasks:

1. Create an account.
2. Login to their account.
3. Can view the restaurant's menu.
4. Select an item from the menu.
5. Add an item to their cart.
6. View their current order's cart.
7. Enter delivery and payment details.
8. Track their order.
9. Able to place order for future date.
10. Buy premium for discounts.

Restaurant Management System

1. New restaurant must be able to register themselves with the application.
2. Add their menu to the restaurant's menu.
3. Can offer discounts.
4. Deliver order.

Order Management System

1. Display order in the screen.
2. Can retrieve order for users from database.
3. Generate confirmation for the order.
4. Keep track of customer order.
5. Assign a rider to deliver a particular order to customer.

Non-functional Requirements

Performance Requirements

1. Performance wise, our project is built to be very responsive and fast. The transitions between the interfaces take no time.
2. Capacity wise, our system is very storage friendly, hardly requires some Mbs of data.
3. Safety wise, the data of users, interacting with our system stays safe and can only be accessed by the system owner.

4. The software is reliable in a sense that it fulfills all the needs that it is promised to fulfill. It was tested for any sort of bugs/ issues and was fixed by the developers eventually.

Safety Requirements

1. We took extra care that our system must not cause any damage on the machine onwhich the user is running our system.
2. The only thing that the user should take care of is the entry of dummy (fake) resultsin the database. Dummy data must be deleted from the database.

Security Requirements

1. External users such as someone out of the organization must not be given access to thesystem's Admin panel. Login ensures this.
2. Only the Stakeholders should have access to the system.
3. The data of the user stays safe and untouchable. So, privacy is maintained.

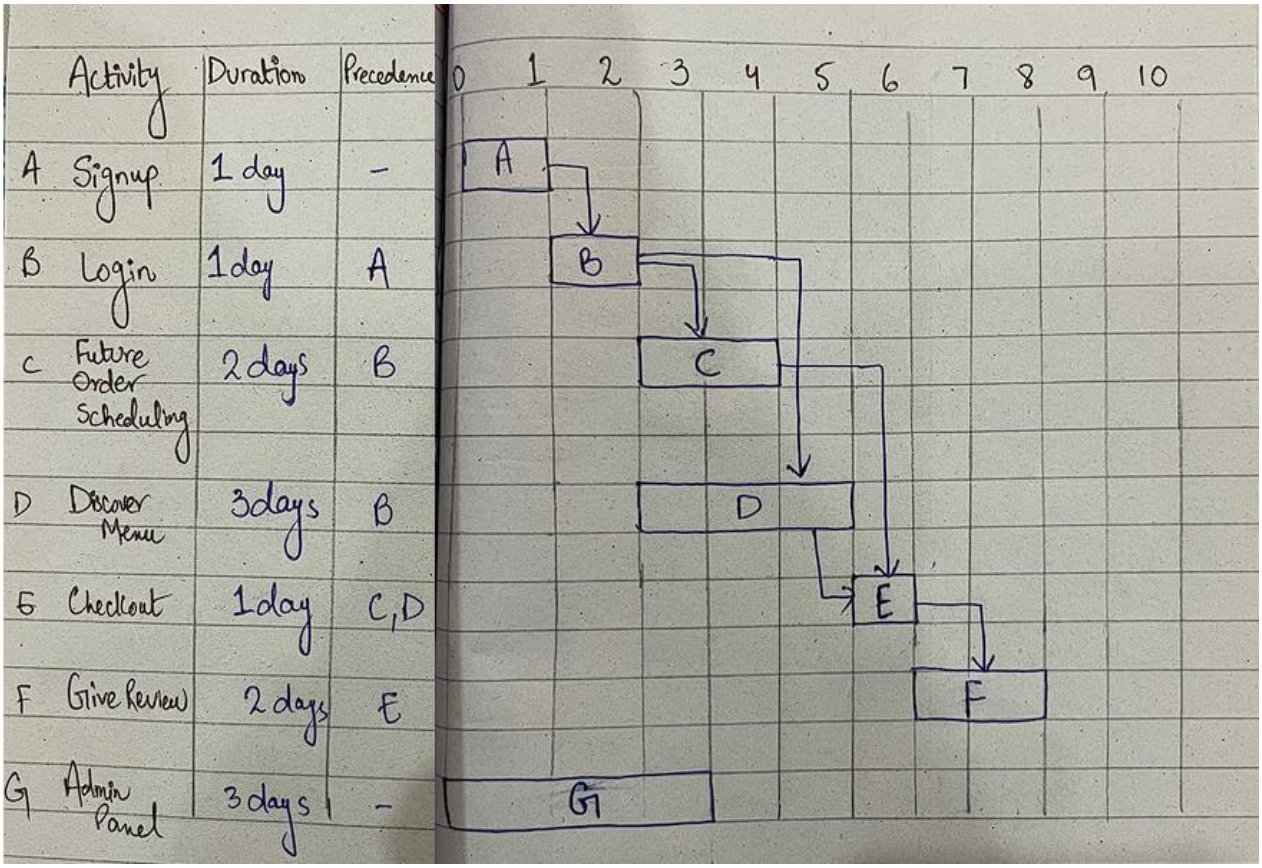
Tools and Technologies:

Operating System: Windows 10
Database: MySQL
Webserver: XAMPP
Web Technologies: HTMLL/ CSS/ JS/ FLASK
Ide and tools: Visual Studio Code

Task Distribution:

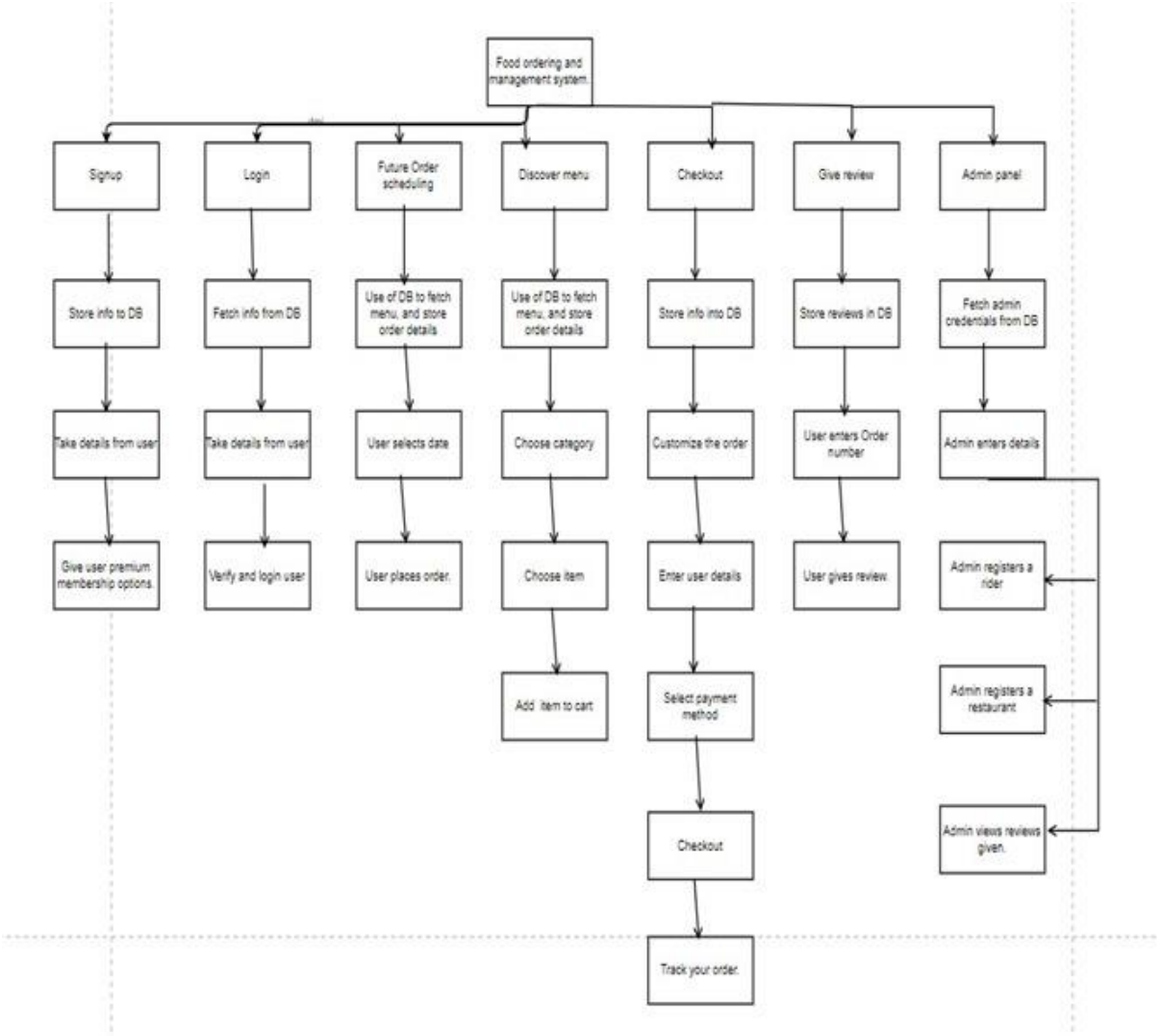
- Frontend using HTML, CSS and Bootstrap is developed by Maarij waseem (21k-4547) , Aalyan(21K-3432)..
- Backend using Flask and MySQL database is developed by Aalyan (21K-3432) and huzaifa (21K-4948).

Project Gantt Chart:

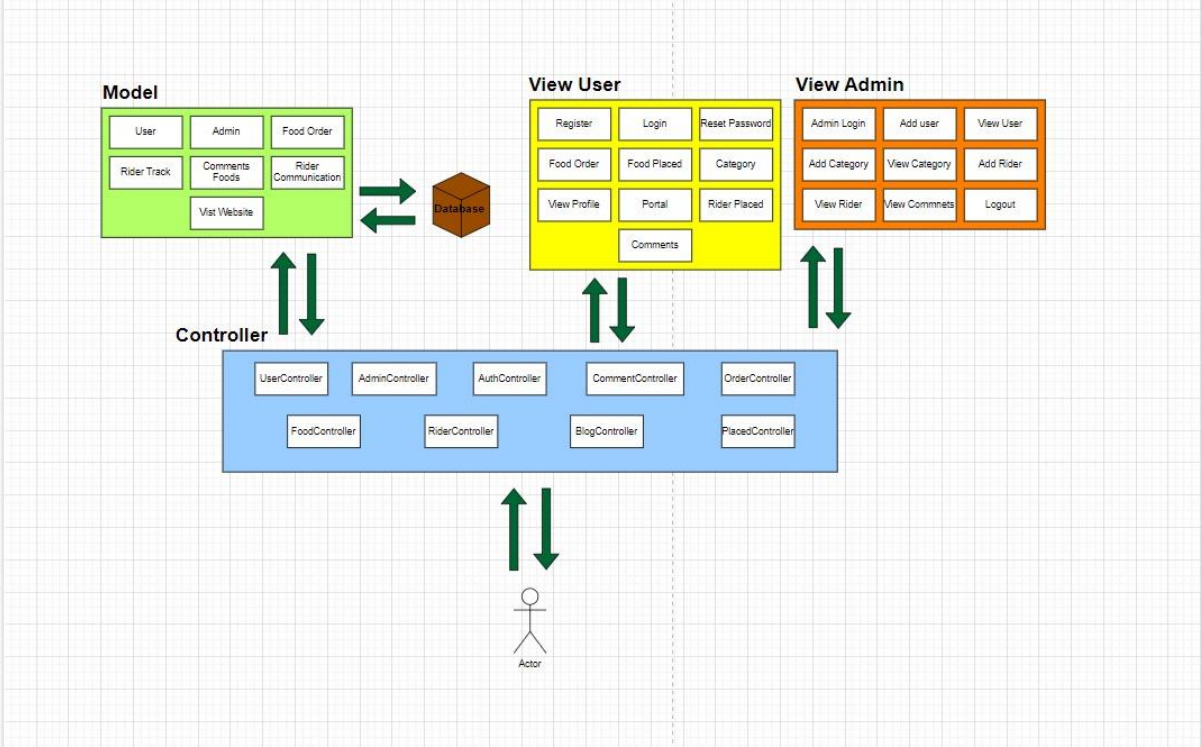


	Activity	Duration	Precedence
A	Signup	1 day	-
B	Login	1day	A
C	Future Order Scheduling	2days	B
D	Discover Menu	3days	B
E	Checkout	1day	C,D
F	Give Review	2days	E
G	Admin Panel	3days	-

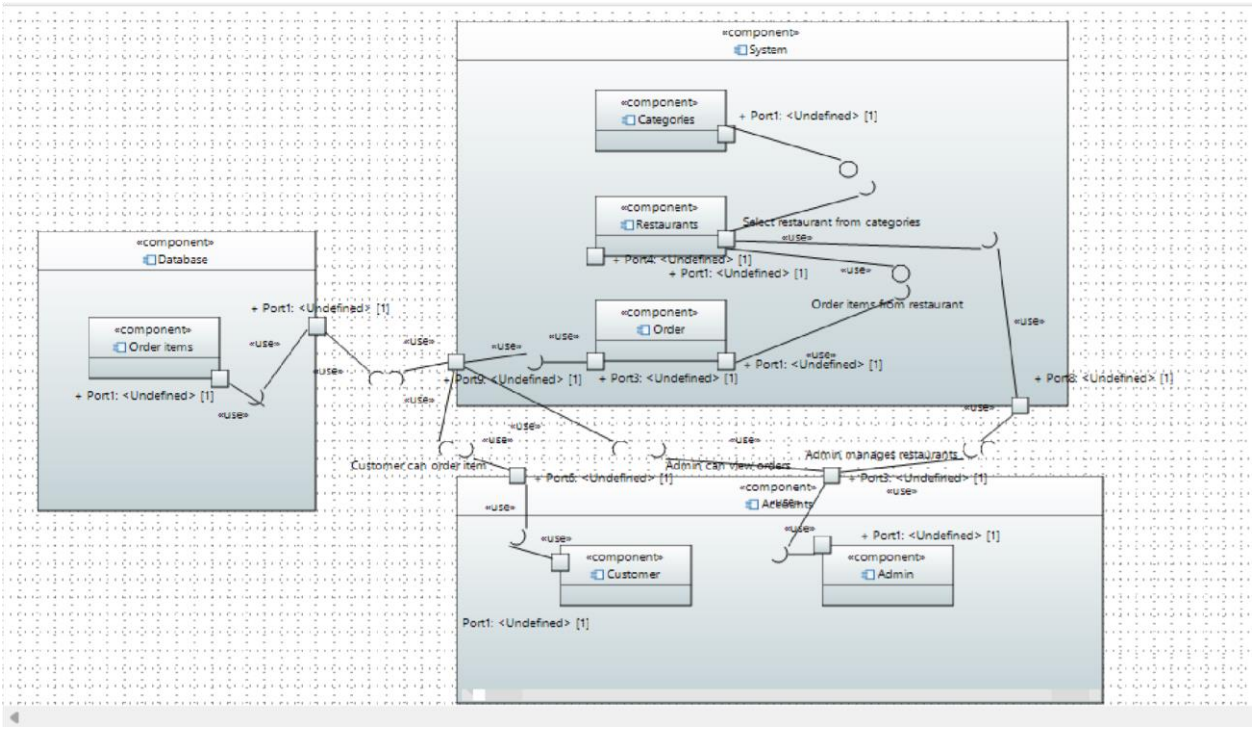
Work breakdown structure :



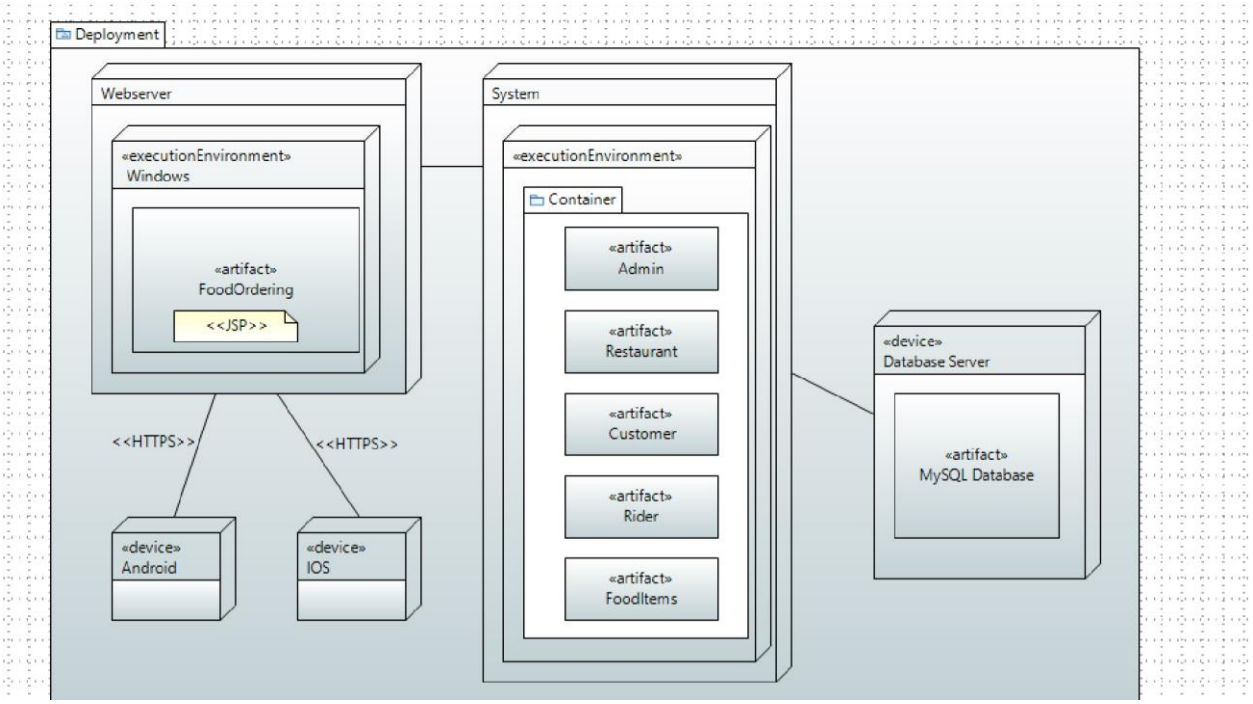
Architecture diagram :



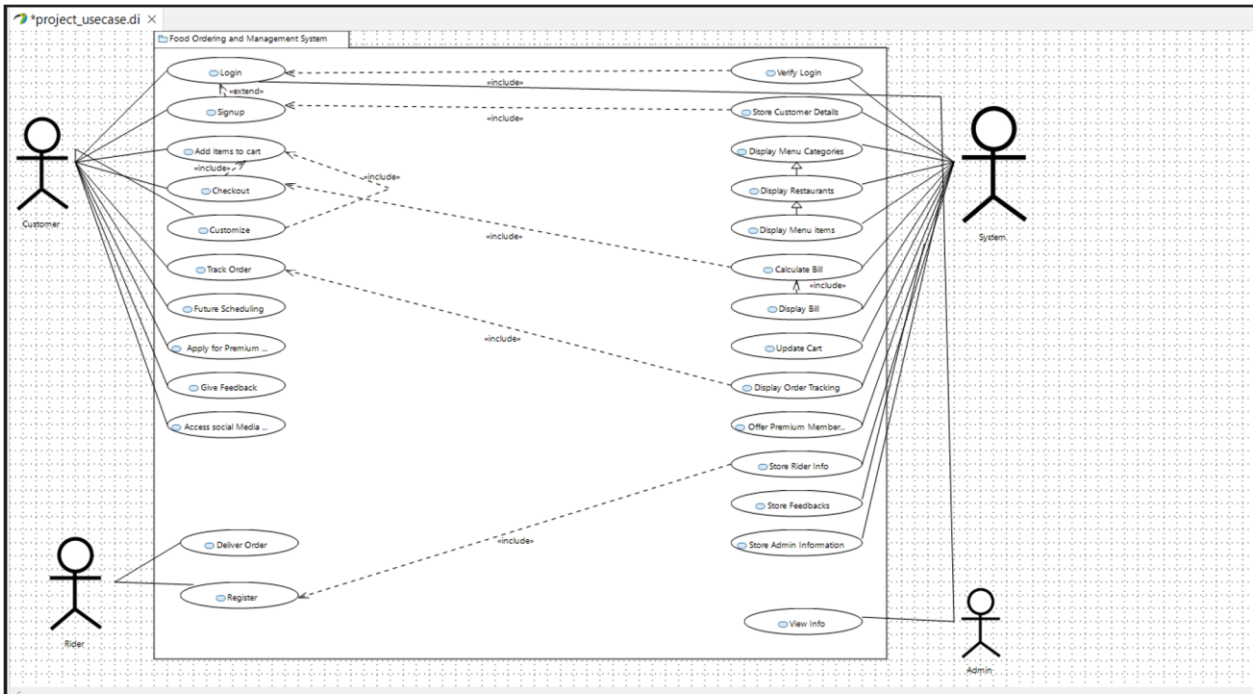
➔COMPONENT DIAGRAM



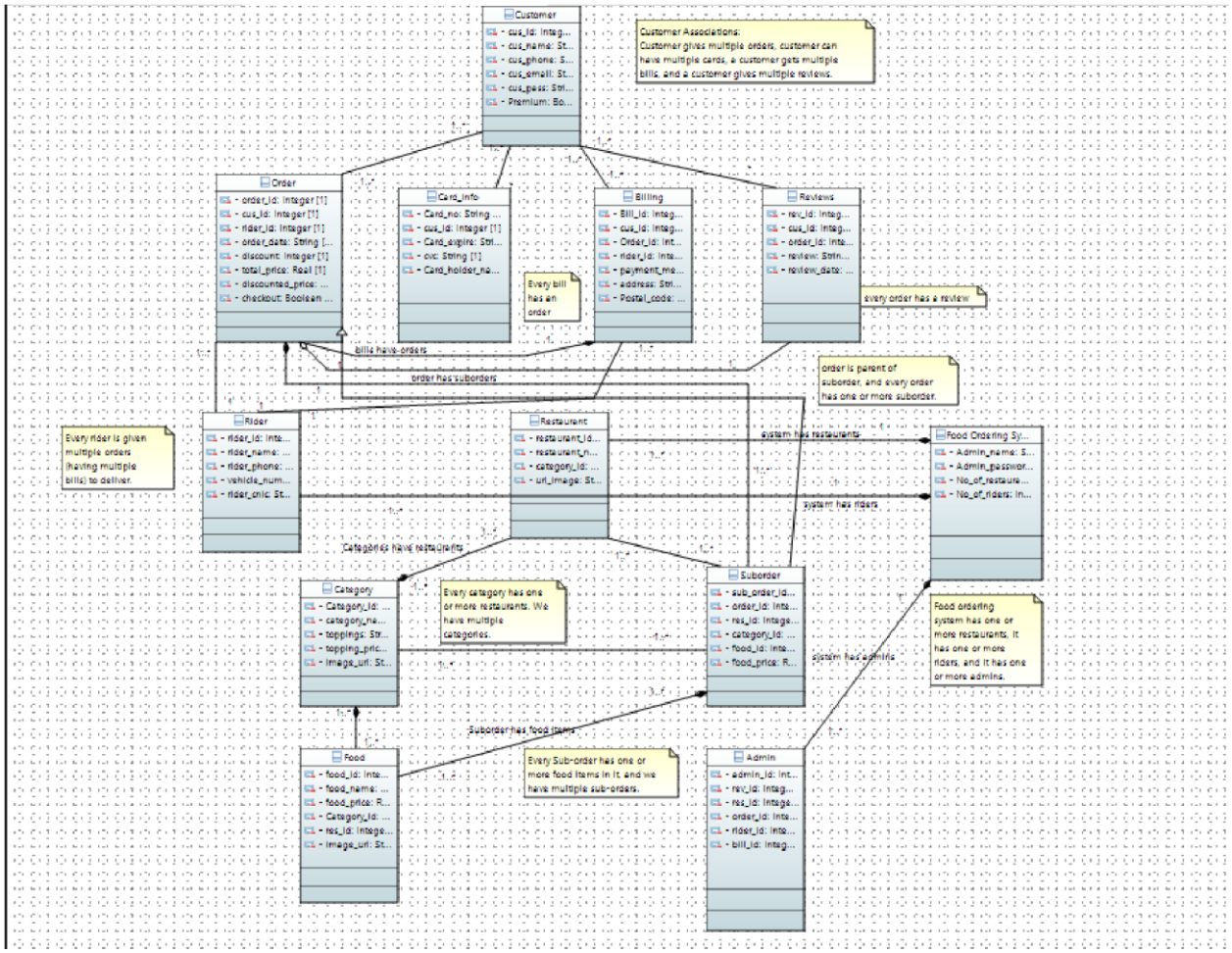
➔DEPLOYMENT DIAGRAM



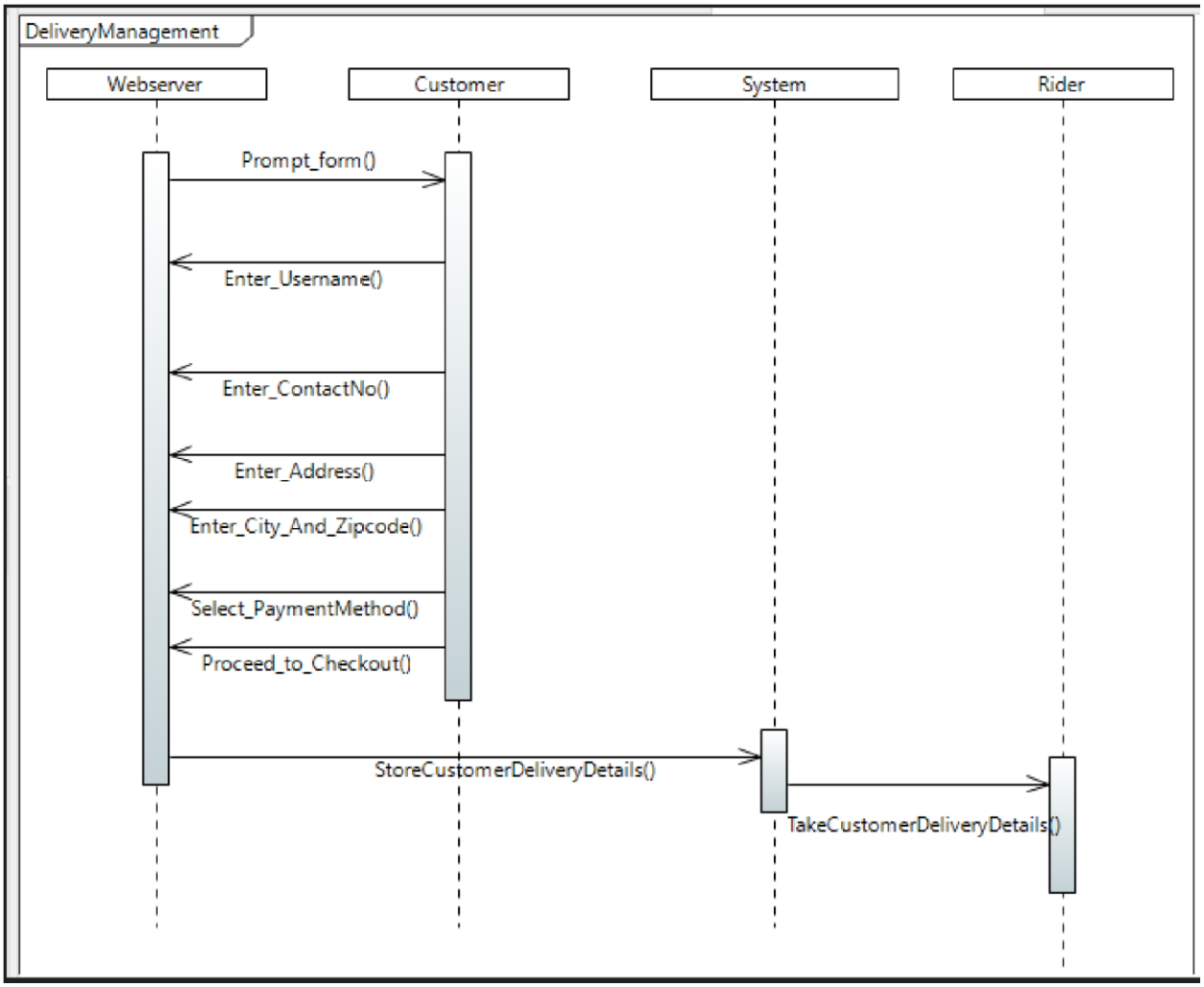
➔USE CASE DIAGRAM

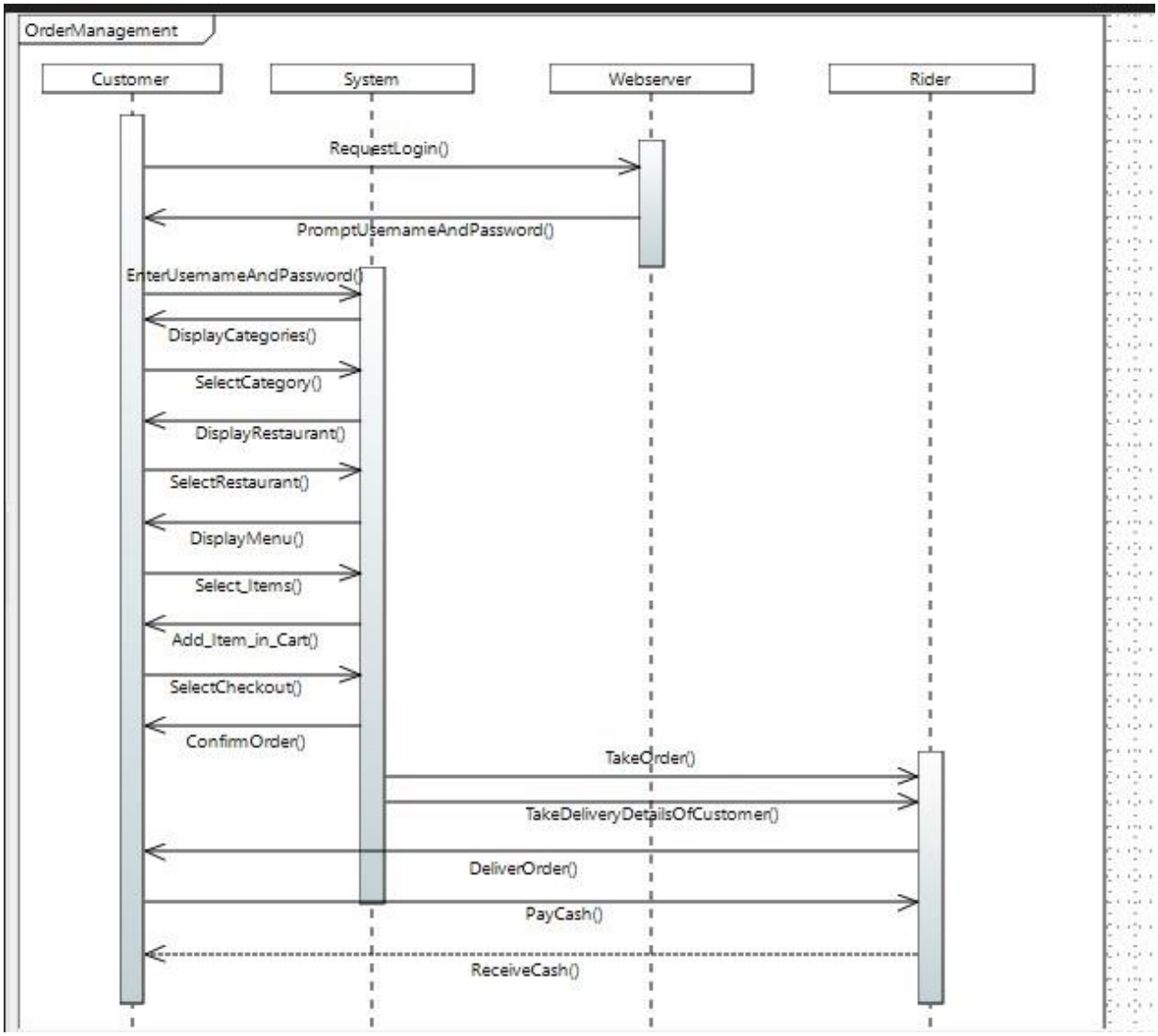


➔DOMAIN MODEL DIAGRAM

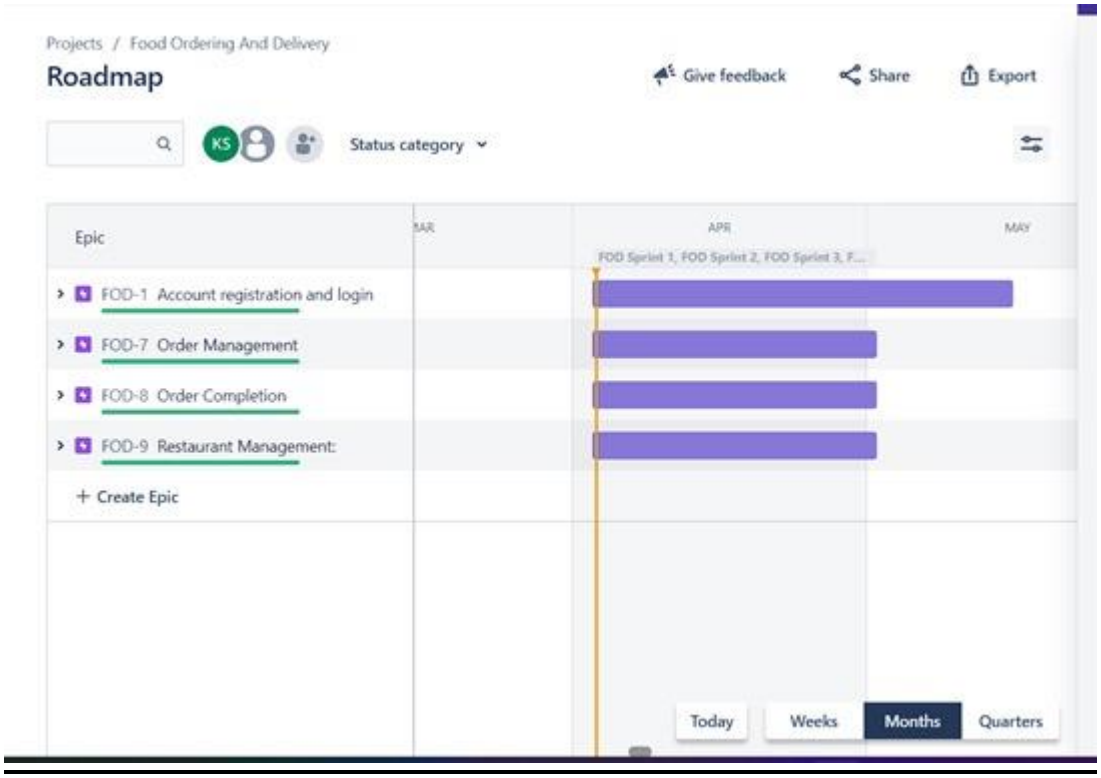

Customer
- cus_id: Integ...
- cus_name: St...
- cus_phone: S...
- cus_email: St...
- cus_pass: Stri...
- Premium: Bo...
Customer Associations:
Customer gives multiple orders, customer can have multiple cards, a customer gets multiple bills, and a customer gives multiple reviews.
Order
- order_id: Integer [1]
- cus_id: Integer [1]
- rider_id: Integer [1]
- order_date: String [...
- discount: Integer [1]
- total_price: Real [1]
- discounted_price: ...
- checkout: Boolean ...
Card_Info
- Card_no: String ...
- cus_id: Integer [1]
- Card_expire: Stri...
- cvc: String [1]
- Card_holder_na...
Every bill has an order
Billing
- Bill_id: Integ...
- cus_id: Integ...
- Order_id: Int...
- rider_id: Inte...
- payment_mo...
- address: Stri...
- Postal_code: ...
Reviews
- rev_id: Integ...
- cus_id: Integ...
- order_id: Inte...
- review: Strin...
- review_date: ...
every order has a review
bills have orders
order has suborders
order is parent of suborder, and every order has one or more suborder.
Every rider is given multiple orders (having multiple bills) to deliver.
Rider
- rider_id: Inte...
- rider_name: ...
- rider_phone: ...
- vehicle_num...
- rider_cnic: St...
Restaurant
- restaurant_id...
- restaurant_n...
- category_id: ...
- url_image: St...
system has restaurants
Food Ordering Sy...
- Admin_name: S...
- Admin_passwor...
- No_of_restaura...
- No_of_riders: In...
system has riders
Categories have restaurants
Category
- Category_id: ...
- category_na...
- toppings: Str...
- topping_pric...
- Image_url: St...
Every category has one or more restaurants. We have multiple categories.
Suborder
- sub_order_id...
- order_id: Inte...
- res_id: Intege...
- category_id: ...
- food_id: Inte...
- food_price: R...
Food ordering system has one or more restaurants. It has one or more riders, and it has one or more admins.
system has admins
Suborder has food items
Food
- food_id: Inte...
- food_name: ...
- food_price: R...
- Category_id: ...
- res_id: Intege...
- Image_url: St...
Every Sub-order has one or more food items in it, and we have multiple sub-orders.
Admin
- admin_id: Int...
- rev_id: Integ...
- res_id: Intege...
- order_id: Inte...
- rider_id: Inte...
- bill_id: Integ...

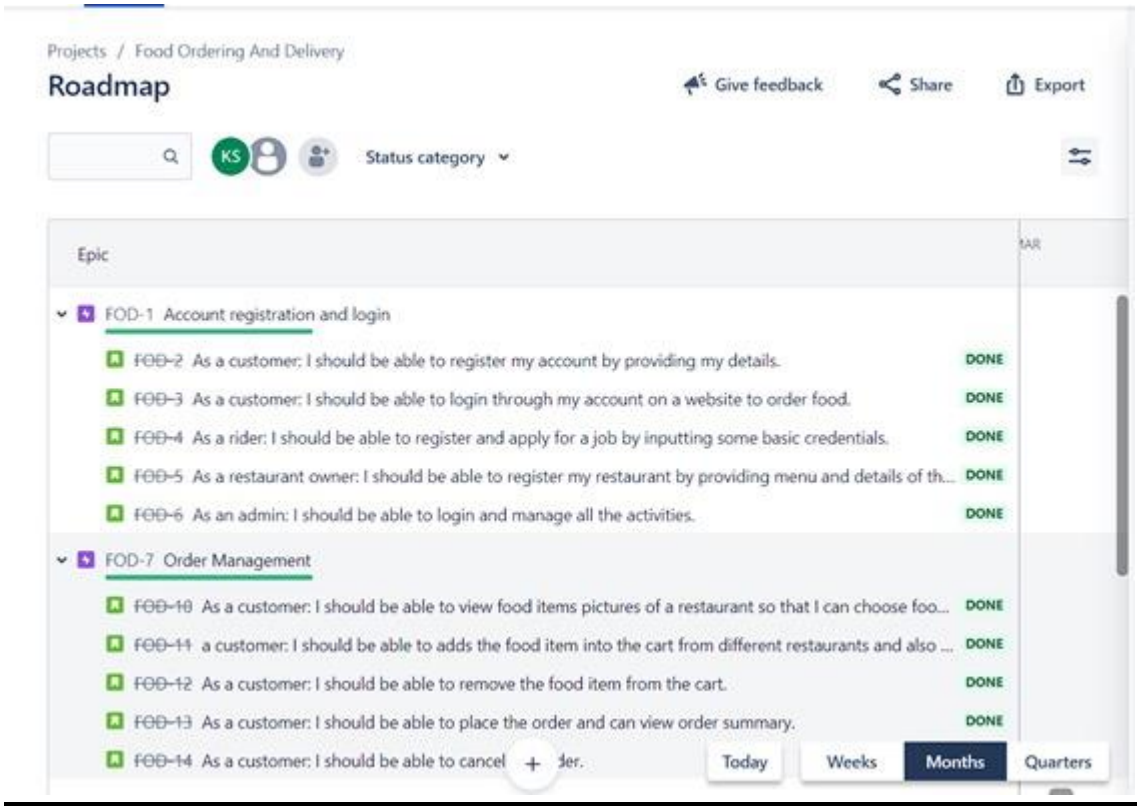
➔SEQUENCE DIAGRAMS

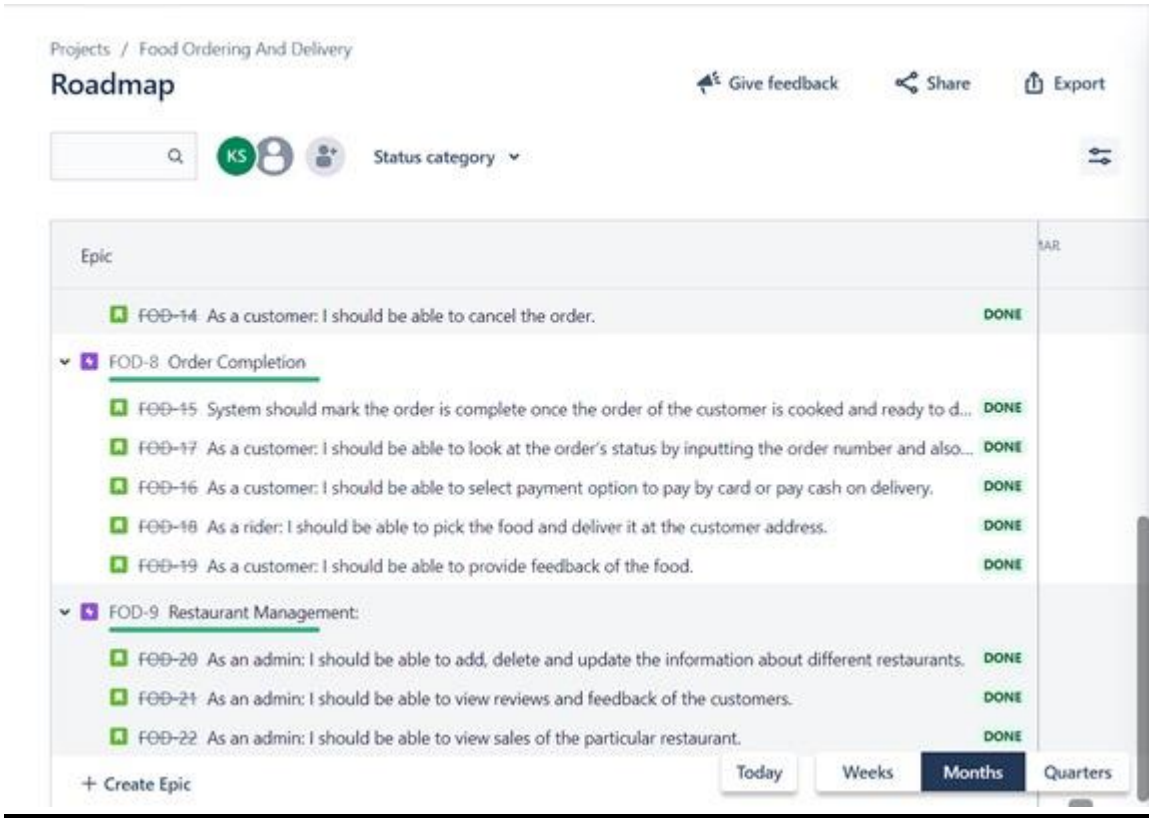

DeliveryManagement
Webserver
Customer
System
Rider
Prompt_form()
Enter_Username()
Enter_ContactNo()
Enter_Address()
Enter_City_And_Zipcode()
Select_PaymentMethod()
Proceed_to_Checkout()
StoreCustomerDeliveryDetails()
TakeCustomerDeliveryDetails()


OrderManagement
Customer
System
Webserver
Rider
RequestLogin()
PromptUsernameAndPassword()
EnterUsernameAndPassword()
DisplayCategories()
SelectCategory()
DisplayRestaurant()
SelectRestaurant()
DisplayMenu()
Select_Items()
Add_Item_in_Cart()
SelectCheckout()
ConfirmOrder()
TakeOrder()
TakeDeliveryDetailsOfCustomer()
DeliverOrder()
PayCash()
ReceiveCash()

Screen shot of Project management tool (jira):


Projects / Food Ordering And Delivery
Roadmap
Give feedback
Share
Export
KS
Status category
Epic
APR
MAY
FOD Sprint 1, FOD Sprint 2, FOD Sprint 3, F...
FOD-1 Account registration and login
FOD-7 Order Management
FOD-8 Order Completion
FOD-9 Restaurant Management:
Create Epic
Today
Weeks
Months
Quarters


Projects / Food Ordering And Delivery
Roadmap
Give feedback
Share
Export
KS
Status category
Epic
FOD-1 Account registration and login
FOD-2 As a customer: I should be able to register my account by providing my details. DONE
FOD-3 As a customer: I should be able to login through my account on a website to order food. DONE
FOD-4 As a rider: I should be able to register and apply for a job by inputting some basic credentials. DONE
FOD-5 As a restaurant owner: I should be able to register my restaurant by providing menu and details of th... DONE
FOD-6 As an admin: I should be able to login and manage all the activities. DONE
FOD-7 Order Management
FOD-10 As a customer: I should be able to view food items pictures of a restaurant so that I can choose foo... DONE
FOD-11 a customer: I should be able to adds the food item into the cart from different restaurants and also ... DONE
FOD-12 As a customer: I should be able to remove the food item from the cart. DONE
FOD-13 As a customer: I should be able to place the order and can view order summary. DONE
FOD-14 As a customer: I should be able to cancel
Today
Weeks
Months
Quarters


Projects / Food Ordering And Delivery
Roadmap
Give feedback
Share
Export
KS
Status category
Epic
FOD-14 As a customer: I should be able to cancel the order. DONE
FOD-8 Order Completion
FOD-15 System should mark the order is complete once the order of the customer is cooked and ready to d... DONE
FOD-17 As a customer: I should be able to look at the order's status by inputting the order number and also... DONE
FOD-16 As a customer: I should be able to select payment option to pay by card or pay cash on delivery. DONE
FOD-18 As a rider: I should be able to pick the food and deliver it at the customer address. DONE
FOD-19 As a customer: I should be able to provide feedback of the food. DONE
FOD-9 Restaurant Management:
FOD-20 As an admin: I should be able to add, delete and update the information about different restaurants. DONE
FOD-21 As an admin: I should be able to view reviews and feedback of the customers. DONE
FOD-22 As an admin: I should be able to view sales of the particular restaurant. DONE
+ Create Epic
Today
Weeks
Months
Quarters