Summary:

Team Members: Abdulrahman Alaraj, Chirag Kamat, Ganesh ByrandurgaGopinath.

Title of the project: FitFoodie

Project Summary: A web based system which recommends restaurants and allows the user to order food based on nutrition preferences\requirements.

Actors: Admin, Customer and Guest.

Project Requirements:

User Requirements				
ID	**Description**	**Topic Area**	**Actor**	**Priority**
US-01	As an Admin, I need to be able to add a Restaurant.	User Account	Admin	High
US-02	As an Admin, I need to be able to add meals to a Restaurant.	User Account	Admin	High
US-03	As an Admin, I need to be able to delete a Restaurant.	User Account	Admin	Medium
US-04	As an Admin, I need to be able to delete meal from a Restaurant.	User Account	Admin	Medium
US-05	As an Admin, I need to be able to create a default profile.	User Account	Admin	Medium
US-06	As an Admin, I need to be able to view Restaurant details in the database to have a collective view.	Documentation	Admin	High
US-07	As a Customer, I need to be able to order food based on nutritional preferences.	User Account	Customer, Guest	High
US-08	As a Customer, I need to be able to order food based my profile.	User Account	Customer	High
US-09	As a Customer, I need to be able to create my own profile.	User Account	Customer	Medium
US-10	As a Customer, I need to be able to remove my own profile.	User Account	Customer	Medium
US-11	As a Customer, I need to be able to update my profile preferences.	User Account	Customer	Medium
US-12	As a Customer, I need to be able to pay for the order.	Payment	Customer, Guest	High

Business Requirements				
ID	**Description**	**Topic Area**	**Actors**	**Priority**
BR-01	Admins and Customers need to be logged in before practicing their roles.	Authentication	Admin, Customer	High
BR-02	Customers can add only one item to the shopping cart.	Shopping	Customer, Guest	High
BR-03	Admins cannot have profiles.	User Account	Admin	Medium

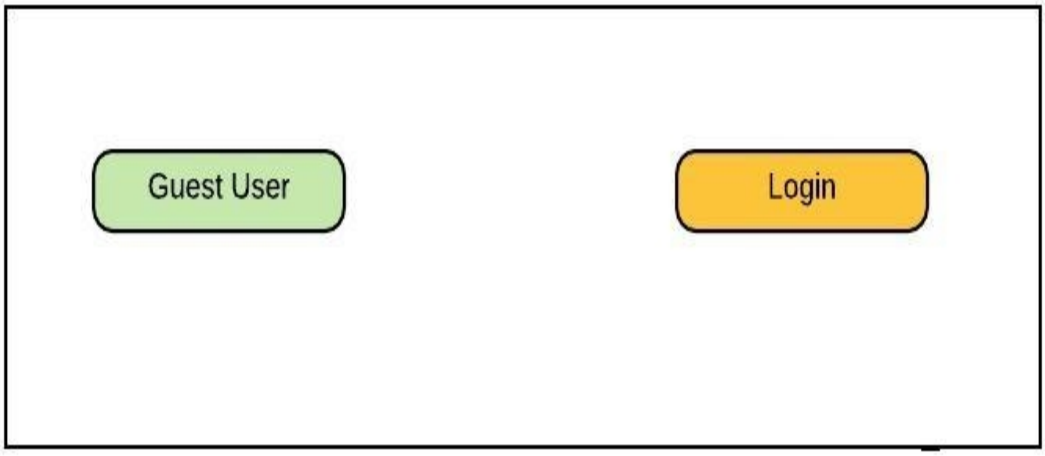
Non-Functional Requirements		
ID	**Description**	**Priority**
NFR-01	Usability: The system is easy to be used.	Medium
NFR-02	Performance: The system is able to store data for tens of users.	Medium
NFR-03	Security: The system must store users' credentials securely.	High
NFR-04	Security: The system must maintain integrity of users' data.	High

Data Storage:

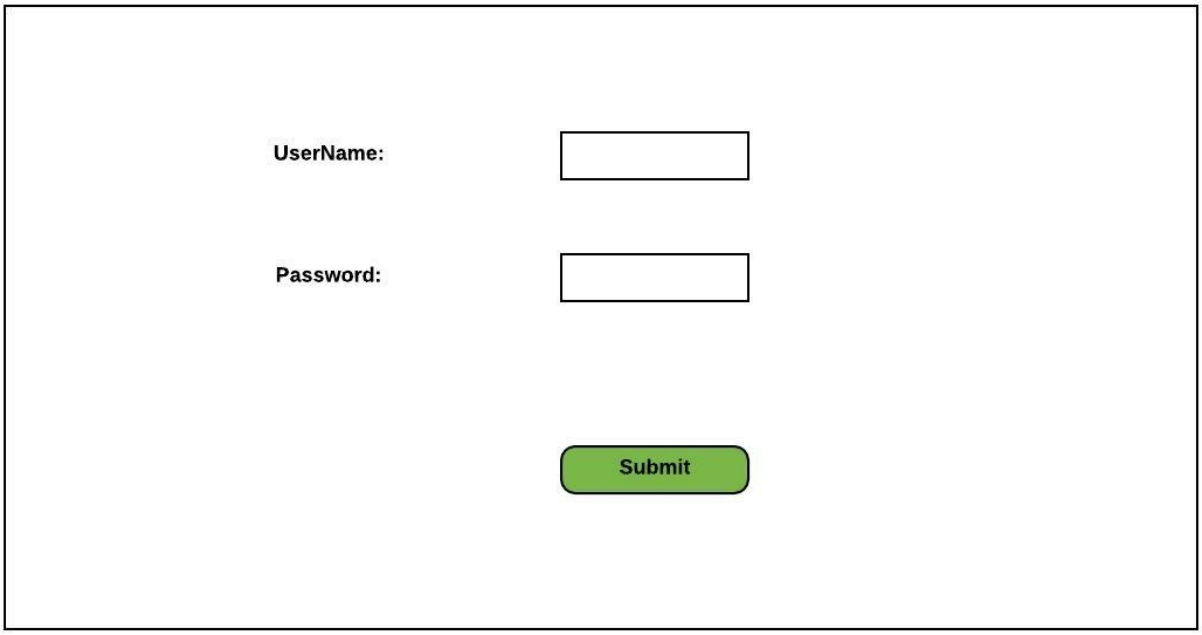
In this project, we will leverage MySQL Database in order to store and retrieve data related to customers, admins and restaurants.

UI Mockup:

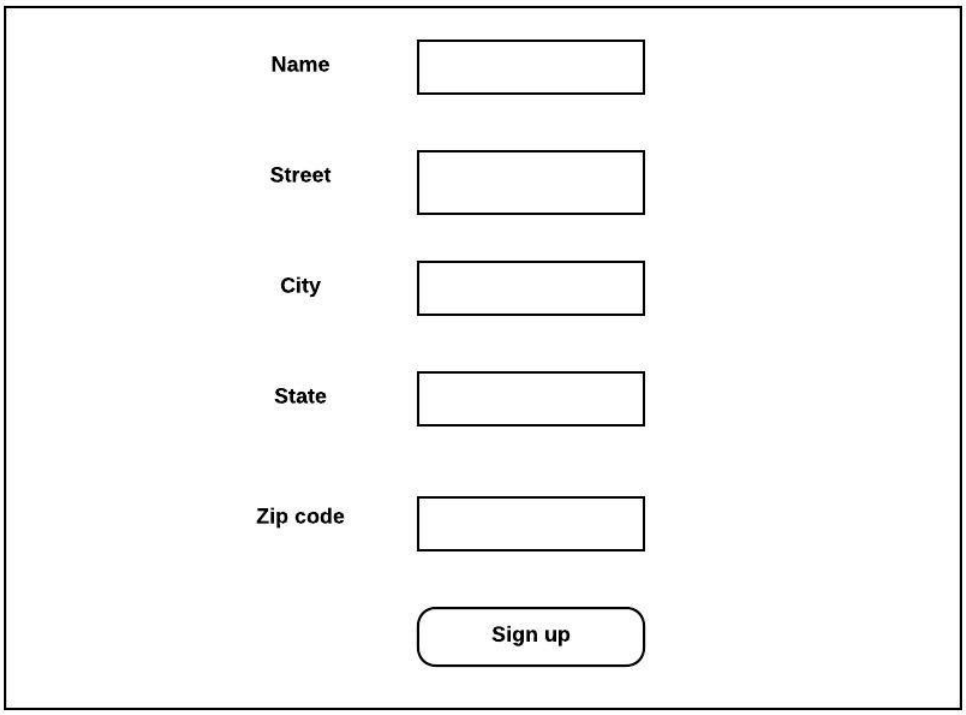
1. Main page



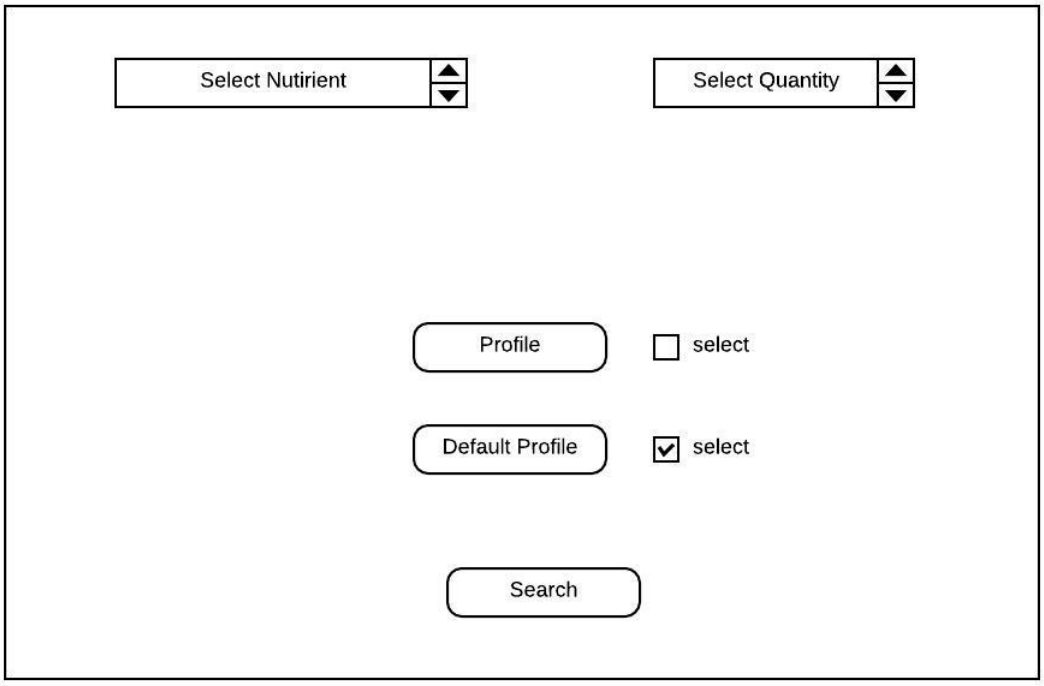
2. Login page



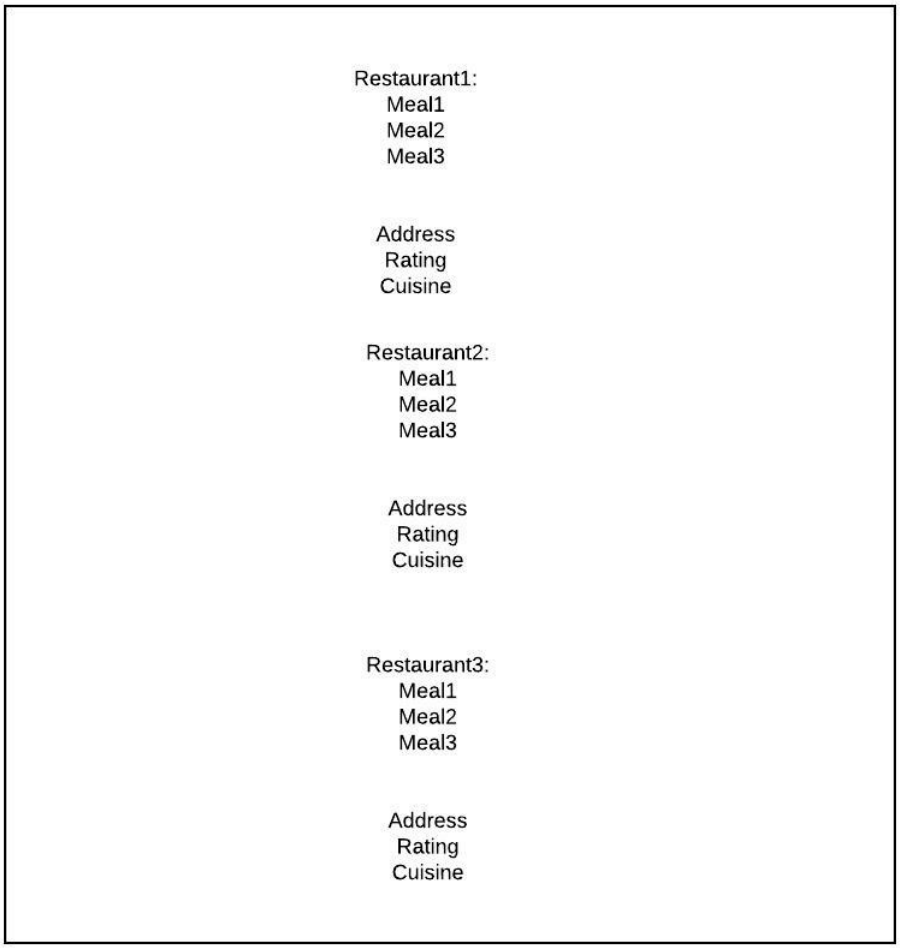
3. Signup



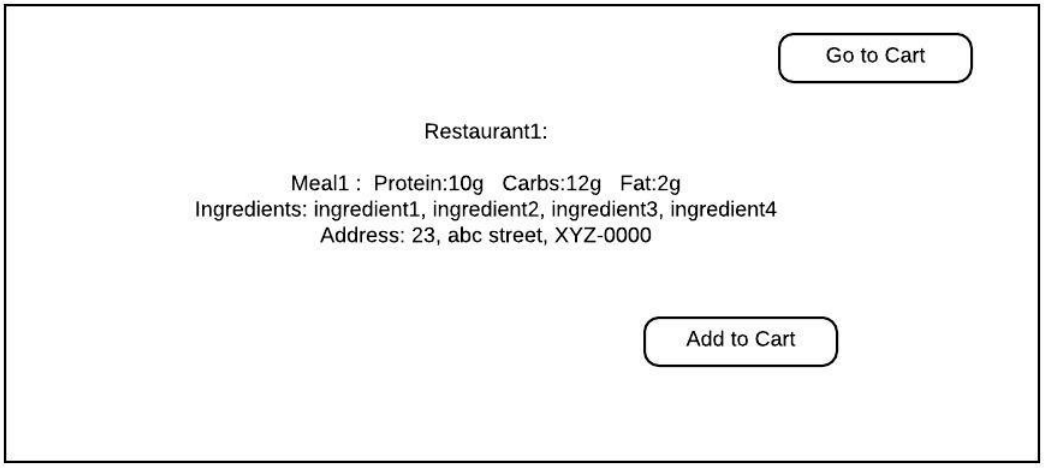
4. Search meals



5. Search results

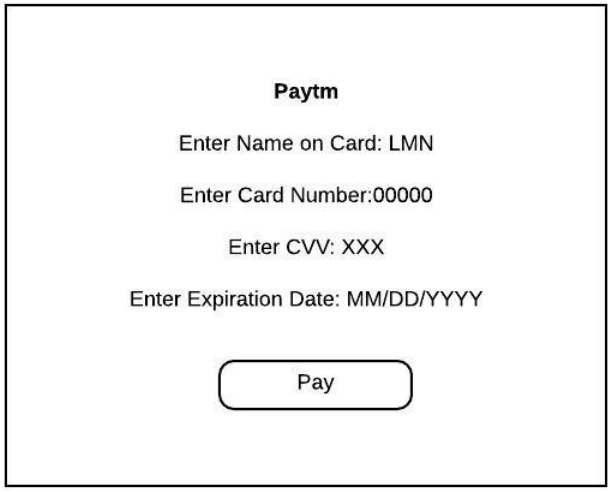


6. Meal selection



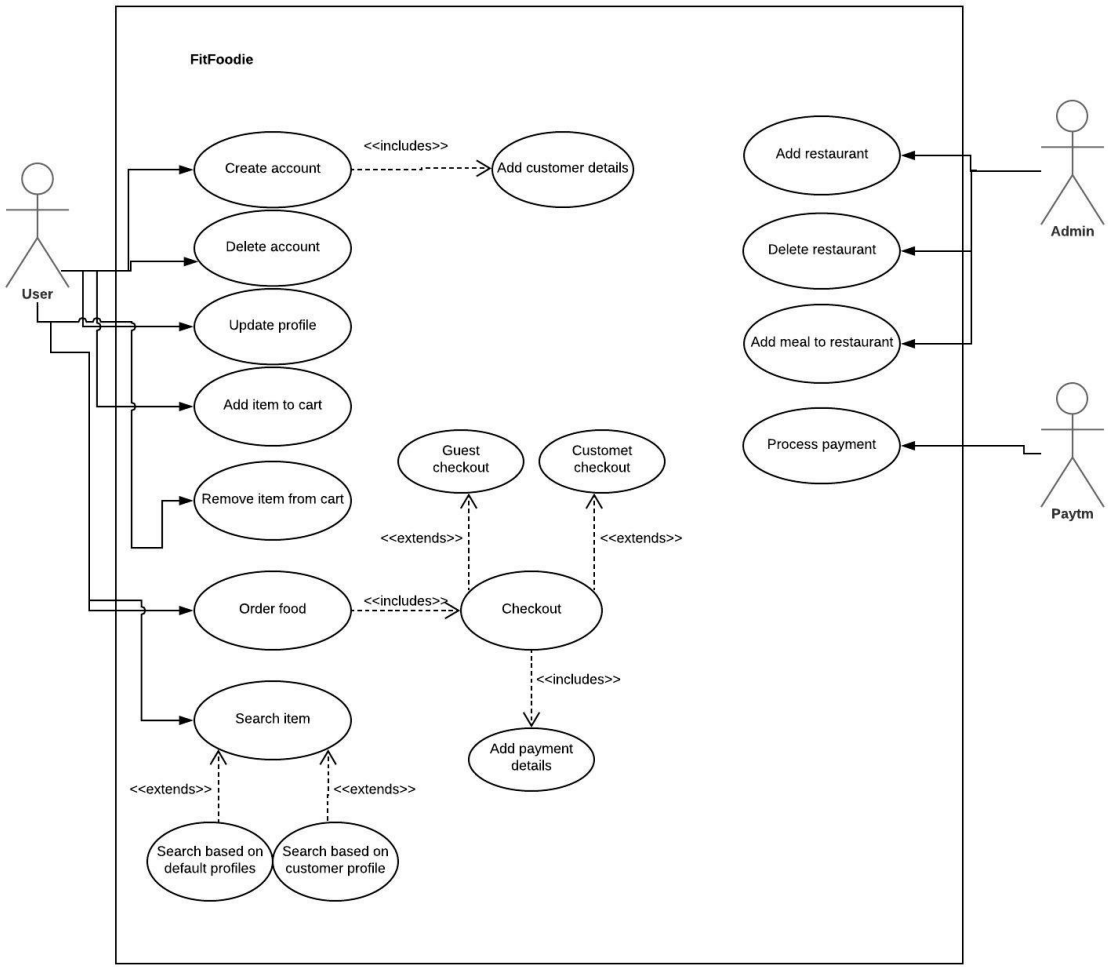
7. Shopping cart

8. Paytm interaction



9. Successful transaction

Use Case Diagram:



Use Case Documents:

Use Case ID:	UC - 01		
Use Case Name:	Order Food		
Description:	Customer should be able to search for a meal based on nutritional preference and order food		
Actors:	Customer		
Pre-conditions:	Customer should be logged in Customer should have updated his profile		
Post-conditions:	Customer orders food of his choice after making a legitimate payment		
Frequency of Use:	Every day		
Flow of Events:		**Actor Action**	**System Response**
	1	Customer selects nutrient and enters quantity and location	Searches Meal by the entered nutrient, quantity and location specified
	2		Execute a query against the database to fetch list of restaurants
	3		Display list of restaurants and the meals that are served by the corresponding restaurant.
	4	Customer selects a	Displays meal details

		meal from restaurant	
	5	Customer adds meal to cart	Cart is appropriately updated.
	6	Customer clicks on checkout	Displays cart with Total price.
	7	Customer clicks pay	Asks for card details.
	8	Customer enters Card details.	Processes Payment and displays appropriate message.
Variations:	As a Guest User, anyone can order food.		
Exceptions:	4. Error message: Payment processing failed		
Developer Notes:			

Use Case ID:	UC - 02		
Use Case Name:	Add customer details		
Description:	Customers should be able to create a custom profile based on nutritional preference		
Actors:	Customer		
Pre-conditions:	Customer should be a registered user Customer should have logged in		
Post-conditions:	Customer views a success message after updating their custom profile		
Frequency of Use:	Whenever a registered customer wants to create a custom profile		
Flow of Events:		**Actor Action**	**System Response**
	1	Customer enters	Sets Customer address

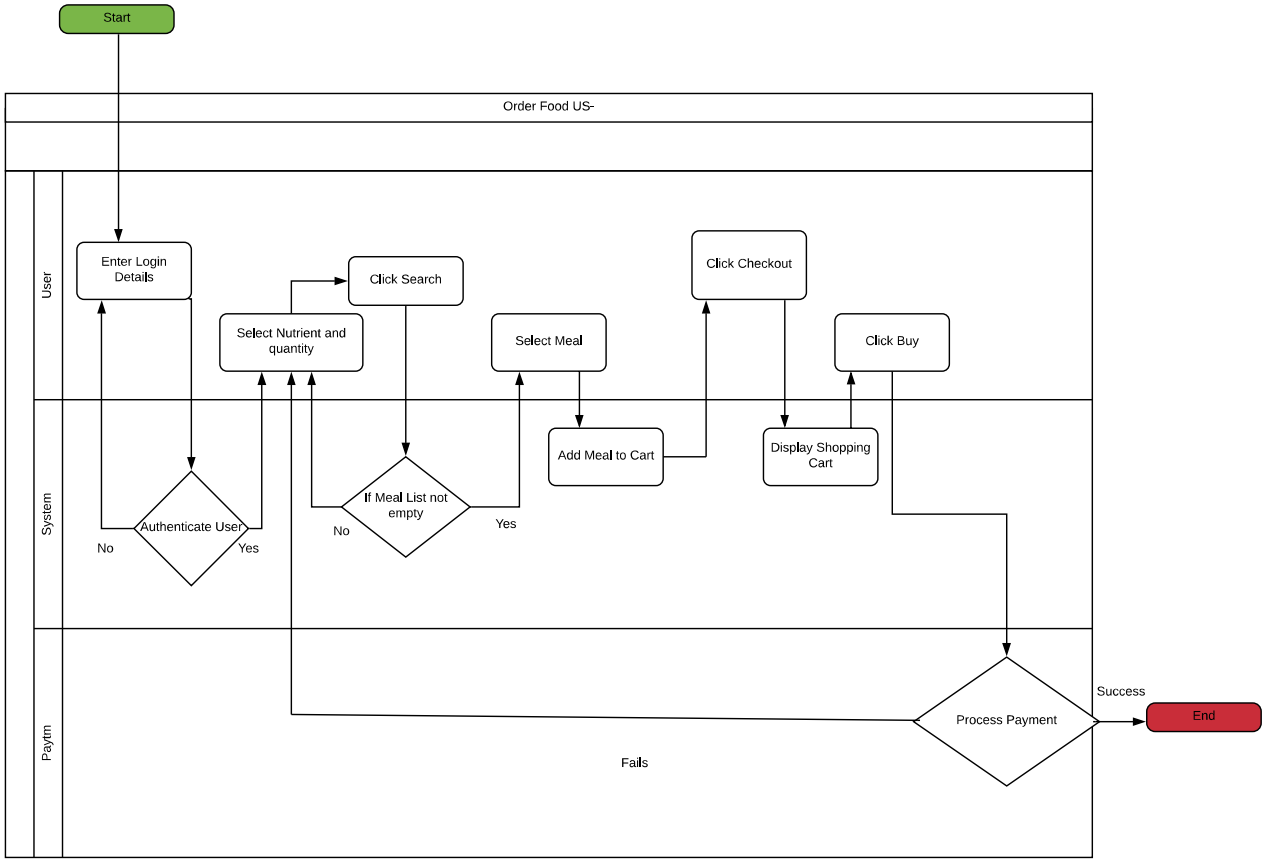
		personal details	details
	2		Adds Customer address details into DB
	3	Customer enters diet profile and nutrient preferences	Sets customer-diet profile and nutrient preferences
	4		Adds Customer diet profile and nutrient preferences to Database
	5		Displays success message
Variations:			
Exceptions:			
Developer Notes:			

Use Case ID:	UC - 03
Use Case Name:	Add restaurant
Description:	Admin should be able to add a restaurant into the database
Actors:	Admin
Pre-conditions:	Admin Should be logged in

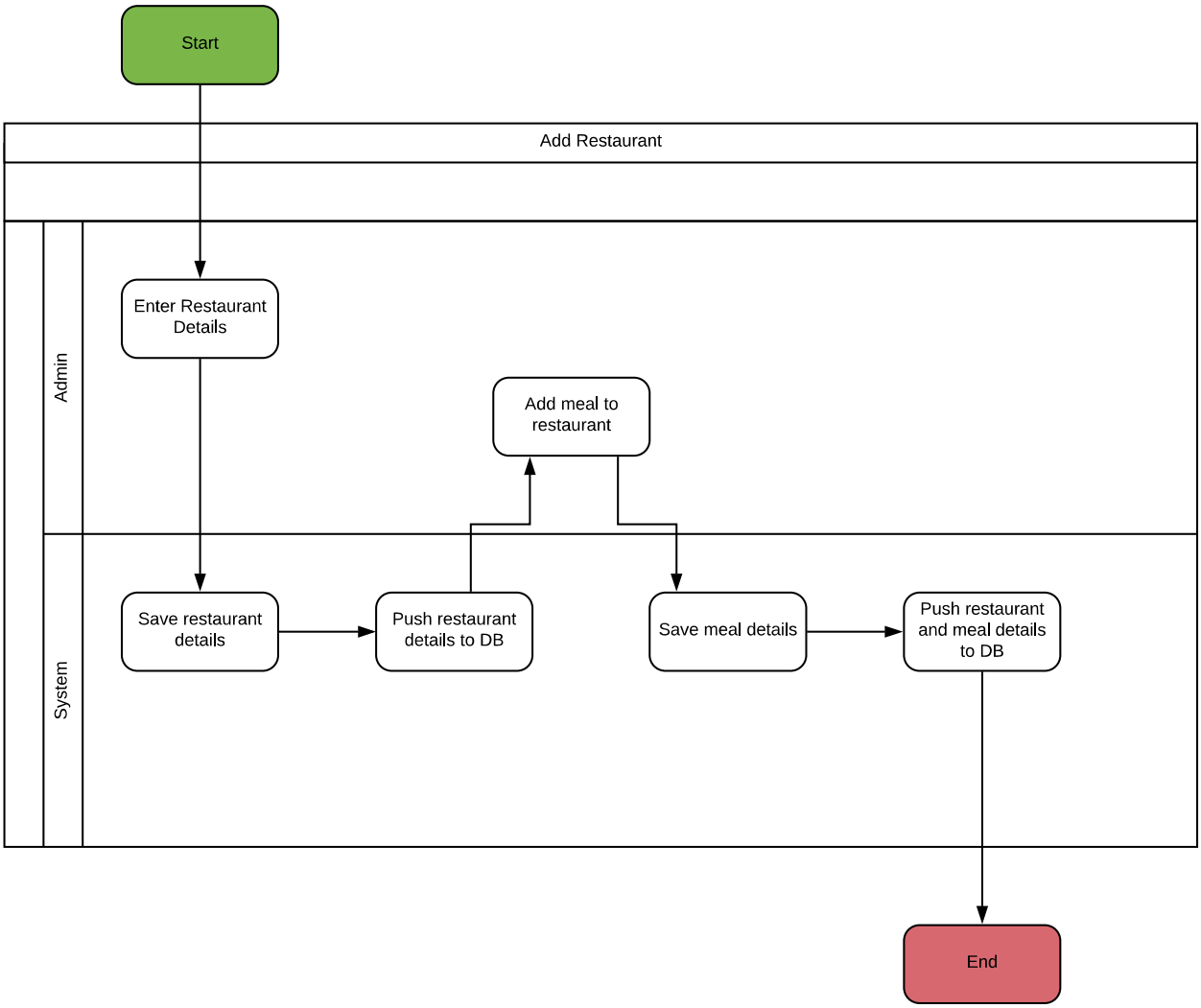
Post-conditions:	Admin can view restaurant details and meals associated with each restaurant.		
Frequency of Use:	Whenever an admin wants to add a restaurant		
Flow of Events:		**Actor Action**	**System Response**
	1	Admin enters restaurant details	Sets restaurant details
	2		Restaurant details flushed into database
	3	Admin enters meal details.	Set meal details
	4	Admin enters nutrient details.	Set nutrient details for the meal
	5		Meal details and nutrient details flushed into database
			Display success message
Variations:			
Exceptions:			
Developer Notes:			

Activity Diagrams:

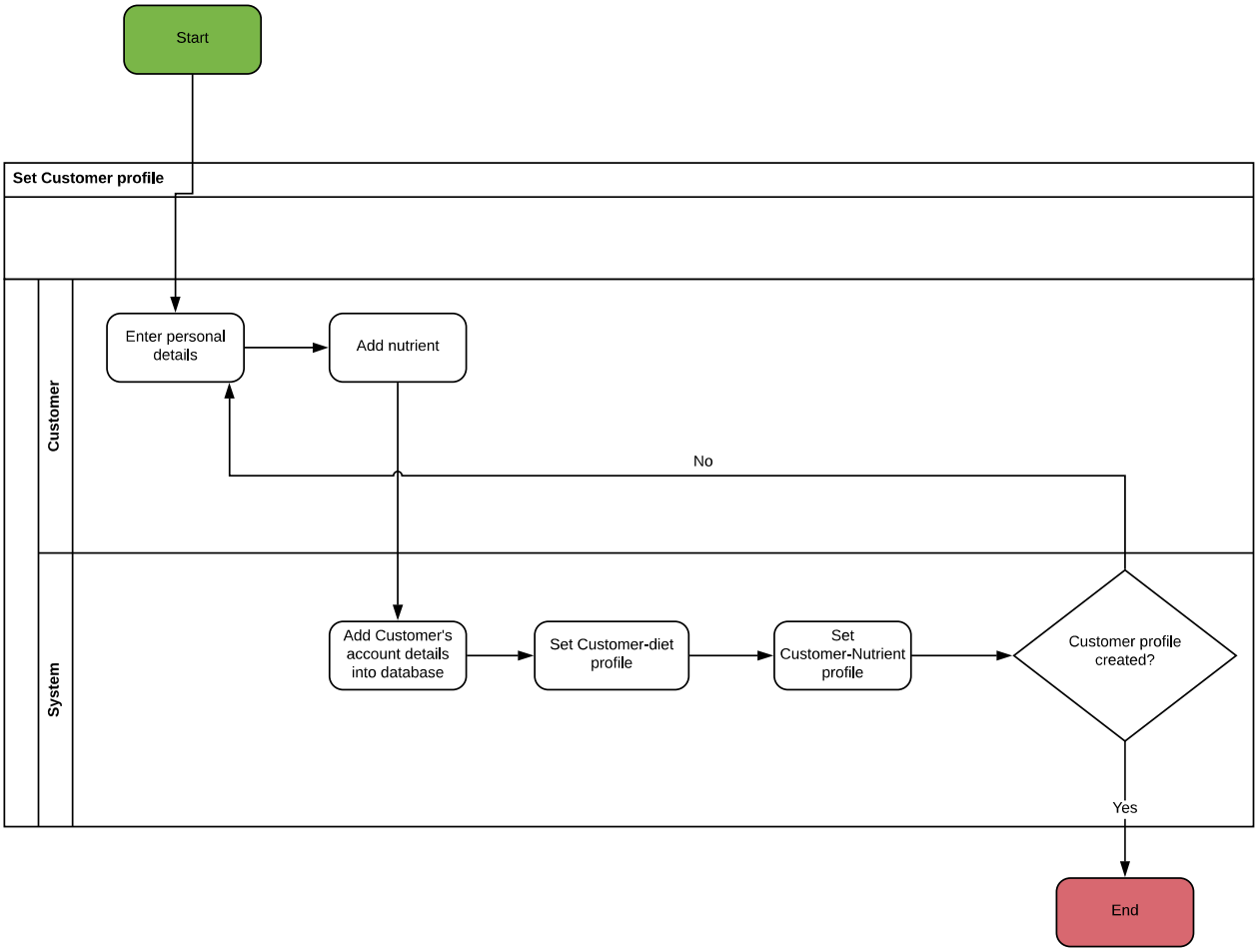
Requirement ID : US - 08
Use Case ID : UC - 01
Use Case name: Order Food



Requirement ID : US - 01
Use Case ID : UC - 03
Use Case name: Add restaurant

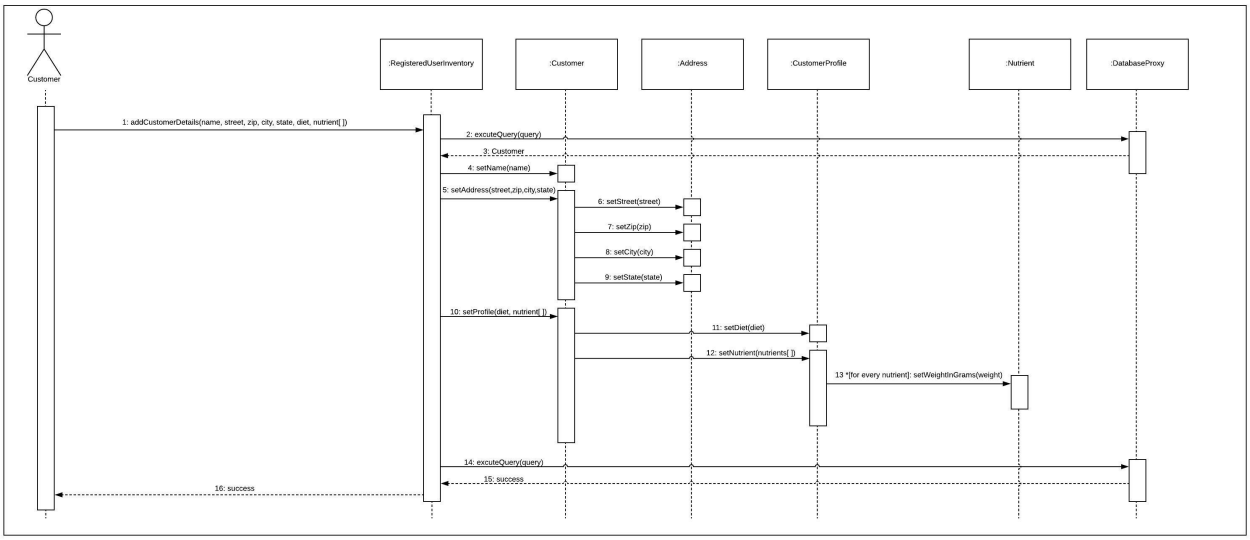


Requirement ID : US **-** 09
Use Case ID : UC **-** 02
Use Case name: Add customer details

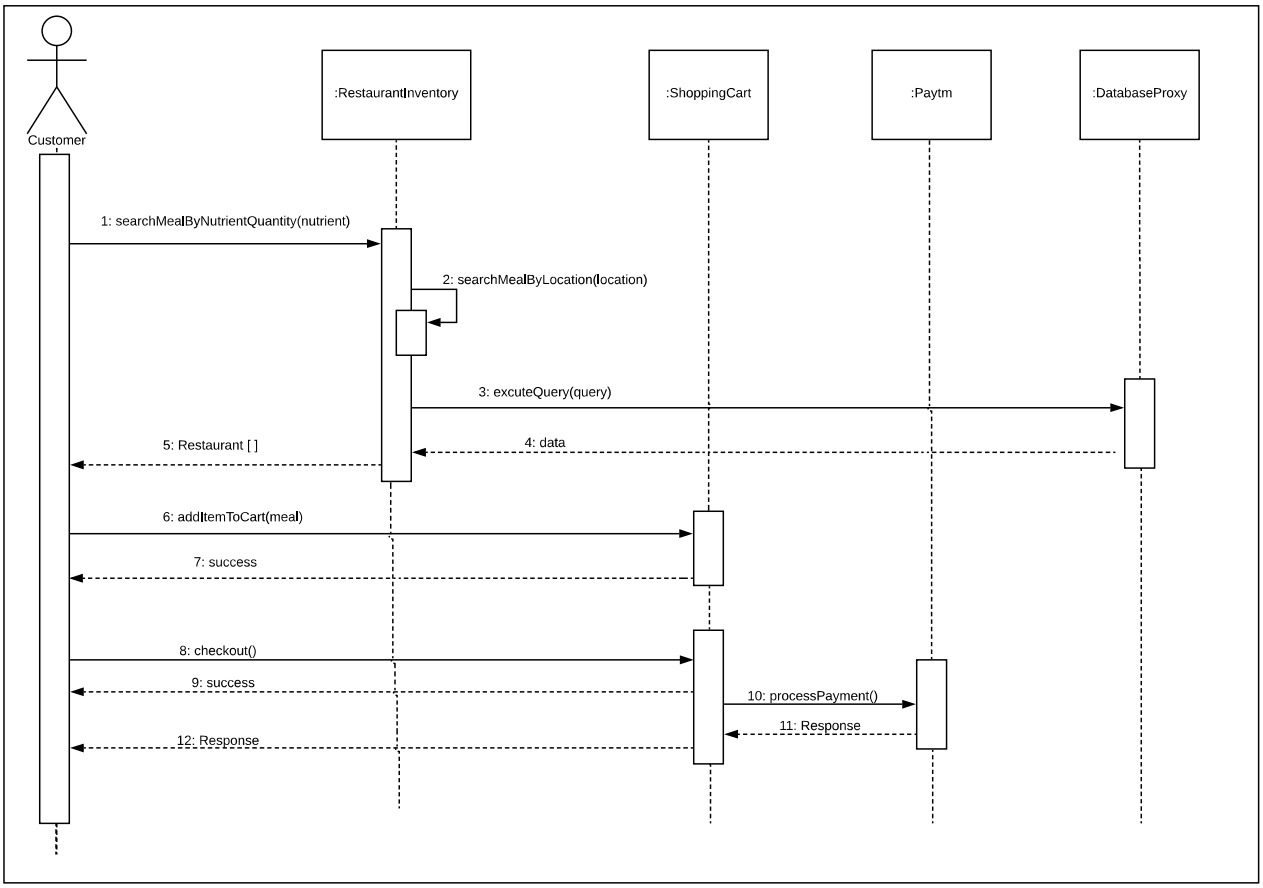


Sequence Diagrams:

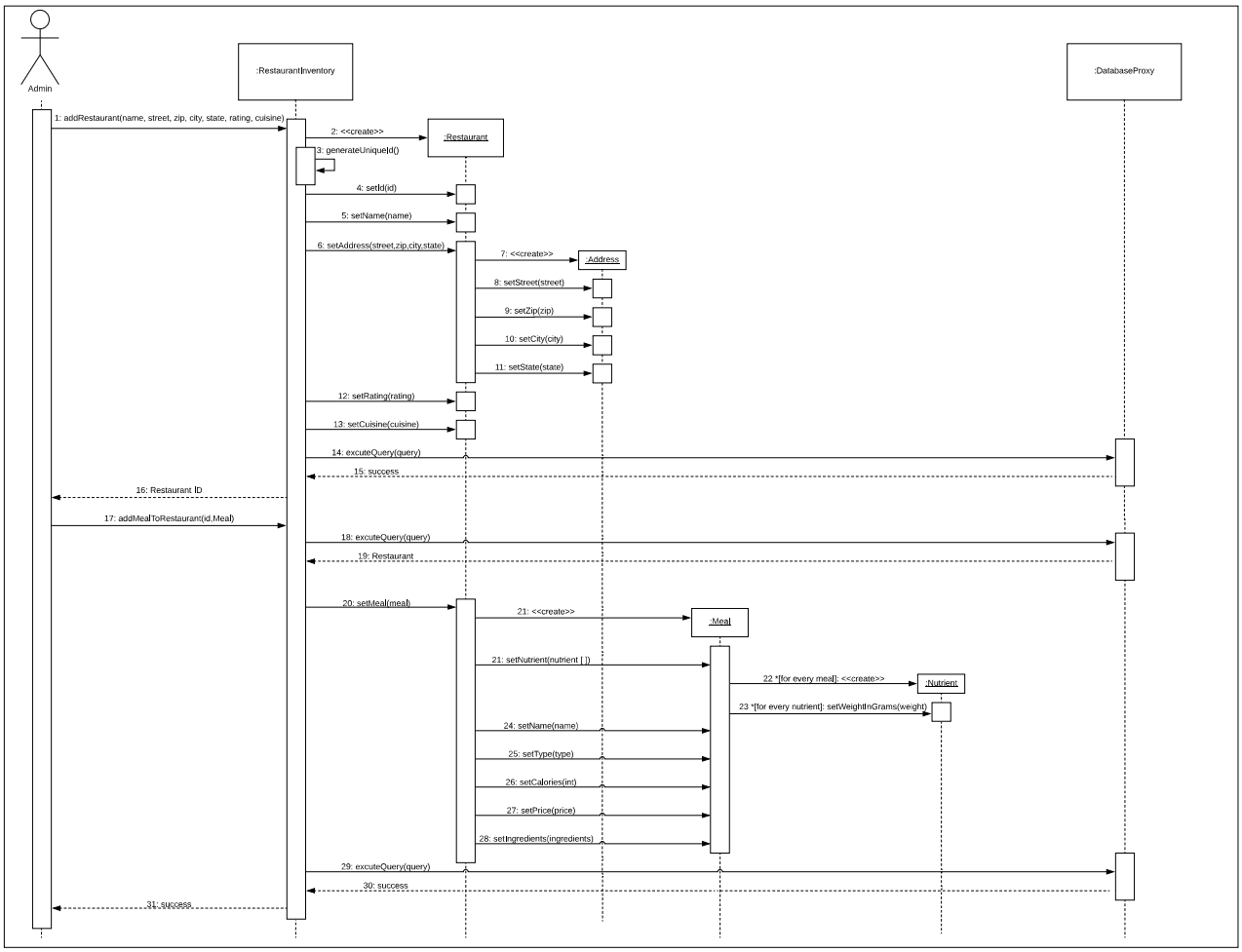
Requirement ID : US - 09
Use Case ID : UC - 02
Use Case name: Add customer details



Requirement ID : US **-** 08
Use Case ID : UC **-** 01
Use Case name: Order Food



Requirement ID : US - 01
Use Case ID : UC - 03
Use Case name: Add restaurant



Class Diagram: