Technical Story Card

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version No. | Date | Prepared by / Modified by | Significant Changes |
| 0.1 | 29-Oct-2021 | Sakthi | Draft version |
| 1.0 | 17-Nov-2021 | Sakthi | Reworked based on Review comments |
| 1.1 | 24-Nov-2021 | Sakthi | Reworked based on Review comments |

Glossary

|  |  |
| --- | --- |
| Abbreviation | Description |
| UI | User Interface |
| DB | Database |
| UML | Unified Modeling Language |
| DD | Detailed Design |

Table of Contents

[1 Introduction 2](#_Toc83671932)

[2 Scope 2](#_Toc83671933)

[3 Non Scope of Change 2](#_Toc83671934)

[4 Design and Detailed technical updates 3](#_Toc83671935)

[4.1 Process model 3](#_Toc83671936)

[4.1.1 Use case Model 3](#_Toc83671937)

[5 Technical Architecture Diagram 4](#_Toc83671938)

[6 Class Diagram 4](#_Toc83671939)

[7 Sequence diagram 4](#_Toc83671940)

[8 ER Diagram for database Design 4](#_Toc83671941)

[9 Other Technical changes 4](#_Toc83671942)

[9.1 CI / Build relates tasks 4](#_Toc83671943)

[9.2 Non-functional Requirements and Design 4](#_Toc83671944)

[10 Additional details 5](#_Toc83671945)

[10.1 Open Questions / clarifications / Assumptions 5](#_Toc83671946)

[10.2 Additional notes to technical team 5](#_Toc83671947)

[11 References 5](#_Toc83671948)

# Introduction

* This Application is built using ASP.NET as front-end and SQL Server as back-end.
* The objective of this project is to build a Vehicle Online food ordering system that enables the food ordering management.
* This system will allow the user to book Food by taking inputs which includes:
  + Credit card details
  + Address
* User can also see the price list of the food rates on the menu.
* This application also allows the user to see the Food available in restaurants near to him so that the user don’t get any difficulties in finding the food near by restaurants.
* It will Show the price in cart is payable.

# 

# Scope

* In the modern Technology people are becoming more advanced. So People orders food.
* Every person o\is using smart phone. There are many restaurants and hotels etc.
* There are many People who works full day so they didn’t have time to go outside. There are many restaurants to order food online.
* With the help of computerized system, we can deliver a good food to customer who wants to Order food at any Nearby restaurants.
* Food ordering system is an automatic system which delivers Food in very high speed in systematic manner.
* By using our system, they can maintain records very easily. Our system covers every area of Restaurants. In coming future there will be excessive need of Online food ordering system.

# Non Scope of Change

a) User Profile details (like Name, Mobile number, Password, etc.).

b) Encryption and Decryption of Password stored in Database.

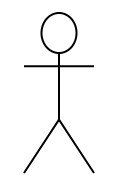
c) upload of data using files.

# Design and Detailed technical updates

## Process model

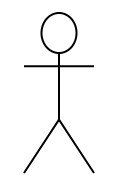
### Use case Model

***UI Screen Flow:-***



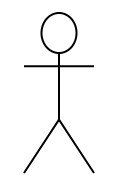
|  |  |
| --- | --- |
| Brief Description | **User Login** |
| Basic Flow | * Click on Login * Give users Credentials * Click on Login |
| Alternate Flow | If the credentials doesn’t match he/she gets an error message as “Invalid credentials” |
| Validation | The entered details will be validated against the credentials stored in the database. |
| Pre-Conditions | The user must supply the correct credentials in order to access the system. |
| Post-Conditions | The user can Order his food. |

***UI Screen Flow:-***



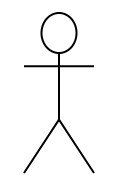
|  |  |
| --- | --- |
| Brief Description | **Register** |
| Basic Flow | * Click on Registration Tab * Enter details like Email, Name, Address, password and confirm password * After entering all required details click on Register button |
| Alternate Flow | 1. If the Email doesn’t contains “@” gets an error message as “Please include an @ in Email address” 2. If mobile number doesnot contains 10 digits gets an error message as “Mobile no must have 10 digits” |
| Validation | The entered details will be validated and stored in the database. |
| Pre-Conditions | The user must give 10 digit mobile number and email address must contain @ symbol. |
| Post-Conditions | The user can Login now in the page. |

***UI Screen Flow:-***



|  |  |
| --- | --- |
| Brief Description | **Booking Page** |
| Basic Flow | * Click on Check out in Cart * Select Cash on Delivery or Credit Card * If User selects Credit Card, then enter Card details and Give address and confirm payment * If user select Cash on delivery, then Enter address and Place order. * Display Invoice |
| Alternate Flow | If card number does not contain 16 digits then User gets an error message as “Card number must contain of 16 digits” |
| Validation | The entered details will be validated against all the validation rules  . |
| Pre-Conditions | User need to checkout the products in the cart page |
| Post-Conditions | After the successfully validation Order will place successful and displays invoice. |

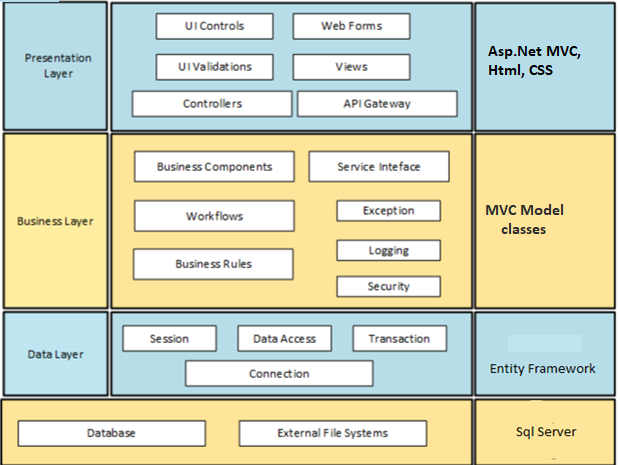
***UI Screen Flow:-***



|  |  |
| --- | --- |
| Brief Description | **Admin Login** |
| Basic Flow | * Click on Login * Give Admin Credentials * Click on Login |
| Alternate Flow | If the credentials doesn’t match he/she gets an error message as “Invalid credentials” |
| Validation | The entered details will be validated against the credentials **“User Name :- Admin” , “Password :- 123”** |
| Pre-Conditions | The user must supply the correct credentials in order to access the system. |
| Post-Conditions | The Admin Can Update any changes. |

**Technical Architecture Diagram**

|  |  |  |
| --- | --- | --- |
| Layer | Component | Example(s) |
| Presentation Layer | Views | User Registration()  Login()  Update() |
| Business Layer | Business components(Controller) | UserDetails ()  Admin() |
| Data Layer | Transaction(Model) | LoginViewModel()  RegistrationViewModel()  SignOut()  AboutUs() |

****

**Presentation Layer**

The presentation layer has two main parts:

a) Client side code and b) Server side code.

The client side code store data from server (if needed) and does validation of input parameters. It also collects meta data information of the client which will be used by the application. The server side code is responsible for the UI components displayed on the screen. It interacts with the “Business Layer” which contains the business logic

Various Components in this layer are:

C.1.1. UI Controls: Interactive components in user interface like buttons, textbox, etc.

C.1.2. Web Pages: Forms with various fields to submit data to the server.

C.1.3. UI Validations: Validations to be done at UI level. E.g. Password validation, date validation, etc.

C.1.4. Controllers: User interface (UI) controllers serve as a connection between your UI and any business logic in your application that controls, or is instructed by, that UI

C.1.5. Views: Basic building block of UI. Customized for each screen or page.

**Business Layer**

This layer has the business logic. It collects data from the Presentation Layer, receives the request from the client, processes the requests, connects to the database layer if needed and sends the response back to the presentation layer.

Various Components in this layer are:

C.1. Workflows: A Workflow is a sequence of tasks that processes a set of data.

C.2. Business Rules: A business rule defines or specify constraints some aspect of business and always resolves to either true or false.

C.3. Exception: Exception Handling is a mechanism to handle runtime errors such as ResourceNotFoundException, etc.

C.5. Security: Security is a vast topic that encompasses many areas. These are available as services, which include data encryption like hashing of password, authentication, and authorization.

**Data Layer**

This layer will access the data stored in the database and files. When the “Business Layer” requests for information, it searches the tables in the database and provides the information. It also stores the data for each session. It stores the log of all the steps followed by the user in a given session.

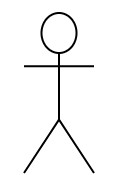
Various Components in this layer are:

C.3.1. Session: It represents the connection between an application and the relational database that stores its persistent objects.

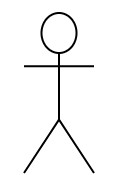
C.3.2. Data Access: It is a pattern that provides an abstract interface to some type of database or other persistence mechanism. By mapping application calls to the persistence layer, the DAO provides some specific data operations without exposing details of the database.

C.3.4. Connection: A Database connection is required to send commands and receive response usually in the form of a result set.

**Use Case Diagram**

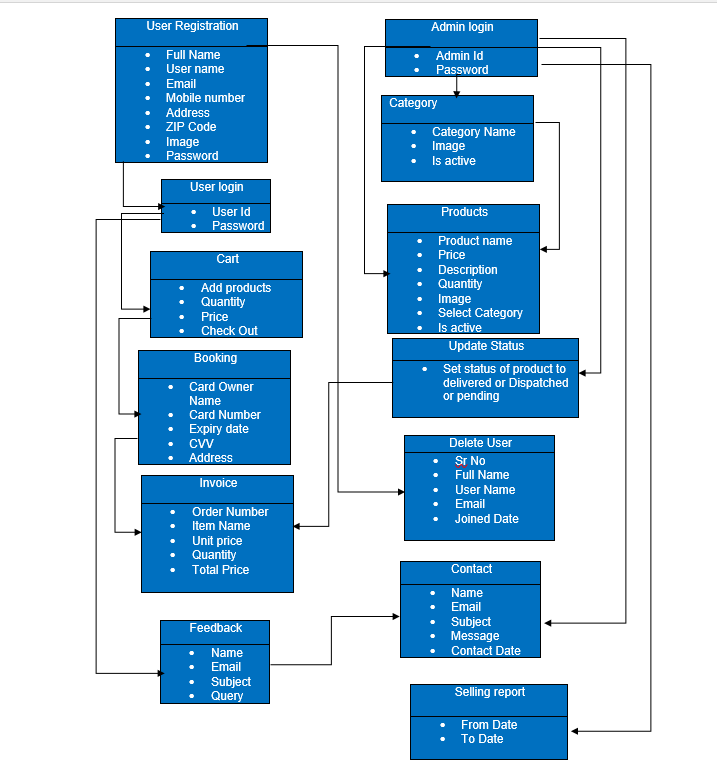


Admin

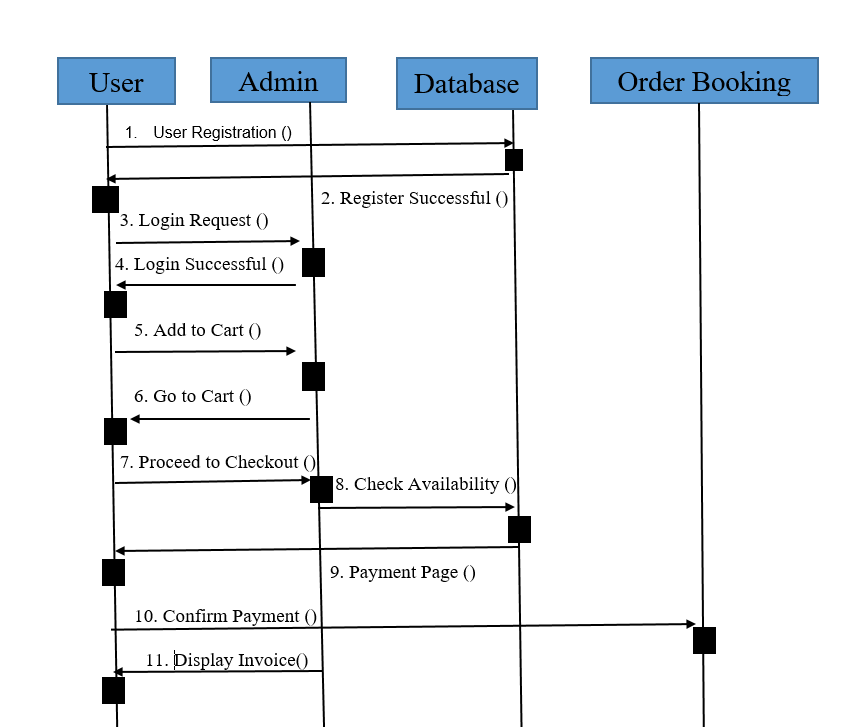


User

# Class Diagram

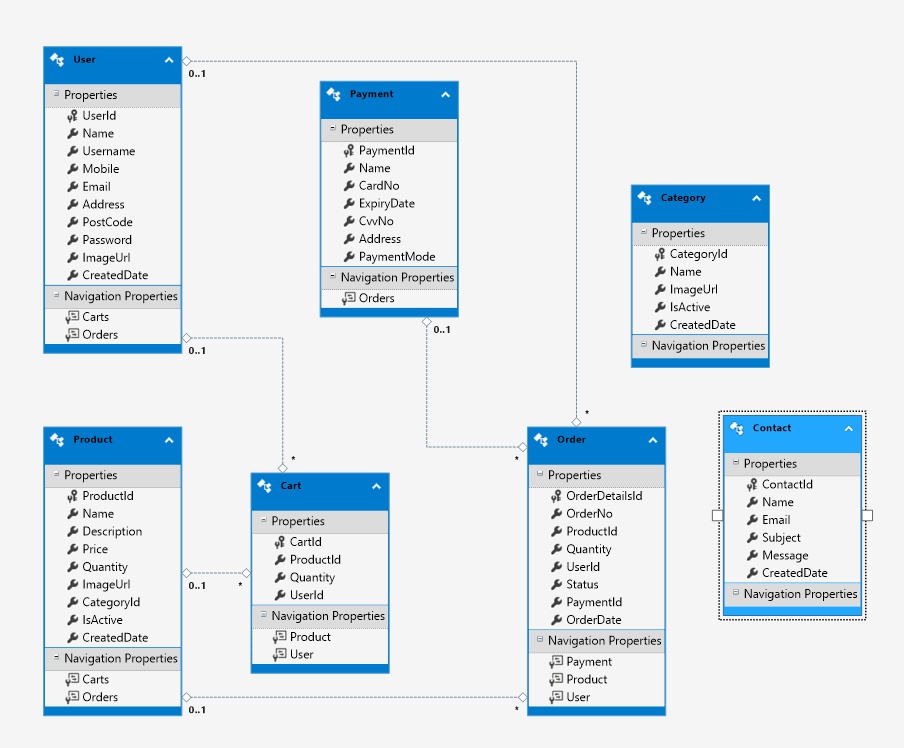


# Sequence diagram



Proceed with the Payment

# ER Diagram for database Design



# Other Technical changes

# Non-functional Requirements and Design

* The central repository should be platform independent so that it can be accessible and store application data via the web application.
* The server should be able to handle concurrent requests from different users.
* The system should provide confidentiality for user data using database encryption.
* The web application provides high availability and high accuracy in finding the locations.

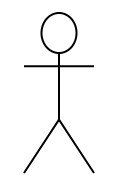
# Open Questions / clarifications / Assumptions

**Assumptions:**

User Profile page is omitted from the scope because it is assumed that most people don’t care about their profile page once they login to the system.

# Enhancement User Stories

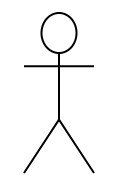
***UI Screen Flow:-***



|  |  |
| --- | --- |
| Brief Description | **Register** |
| Basic Flow | * Click on Registration Tab * Enter details like Email, Name, Address, password and confirm password * After entering all required details click on Register button |
| Alternate Flow | 1. If the Email doesn’t contains “@” gets an error message as “Please include an @ in Email address” 2. If mobile number doesnot contains 10 digits gets an error message as “Mobile no must have 10 digits” |
| Validation | The entered details will be validated and stored in the database. |
| Pre-Conditions | The user must give 10 digit mobile number and email address must contain @ symbol. |
| Post-Conditions | The user can Login now in the page. |

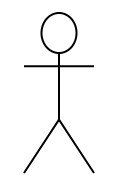
**Users Profile:-**

***UI Screen Flow:-***



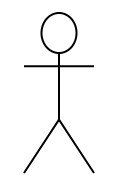
|  |  |
| --- | --- |
| Brief Description | **User Login** |
| Basic Flow | * Click on Login * Give users Credentials * Click on Login |
| Alternate Flow | If the credentials doesn’t match he/she gets an error message as “Invalid credentials” |
| Validation | The entered details will be validated against the credentials stored in the database. |
| Pre-Conditions | The user must supply the correct credentials in order to access the system. |
| Post-Conditions | The user can Order his food. |

***UI Screen Flow:-***



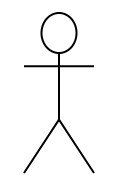
|  |  |
| --- | --- |
| Brief Description | **Cart** |
| Basic Flow | * Add product from menu. * Click on check out. |
| Alternate Flow | If the product is not available then User gets an error message as “The product is out of stock” |
| Validation | The Ordering Products will be validated against the products stored in the database. |
| Pre-Conditions | The user need to add the products into cart from the menu. |
| Post-Conditions | The Order will be placed successfully. |

***UI Screen Flow:-***



|  |  |
| --- | --- |
| Brief Description | **Booking Page** |
| Basic Flow | * Click on Check out in Cart * Select Cash on Delivery or Credit Card * If User selects Credit Card, then enter Card details and Give address and confirm payment * If user select Cash on delivery, then Enter address and Place order. * Display Invoice |
| Alternate Flow | If card number does not contain 16 digits then User gets an error message as “Card number must contain of 16 digits” |
| Validation | The entered details will be validated against all the validation rules  . |
| Pre-Conditions | User need to checkout the products in the cart page |
| Post-Conditions | After the successfully validation Order will place successful and displays invoice. |

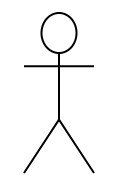
***UI Screen Flow:-***



|  |  |
| --- | --- |
| Brief Description | **Feedback Page** |
| Basic Flow | * Enter Your name * Enter your mail Id * Enter Subject * Enter Query * Click on Submit |
| Alternate Flow | If user doesnot enter any information in query page then User gets an error message as “This field is required” |
| Validation | The entered details will be validated against all the validation rules. |
| Pre-Conditions | The user need to type a query what does he have. |
| Post-Conditions | After sending Query Admin Will watch the query and solves it. |

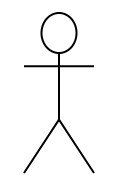
***Admin Profile:-***

***UI Screen Flow:-***



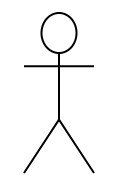
|  |  |
| --- | --- |
| Brief Description | **Admin Login** |
| Basic Flow | * Click on Login * Give Admin Credentials * Click on Login |
| Alternate Flow | If the credentials doesn’t match he/she gets an error message as “Invalid credentials” |
| Validation | The entered details will be validated against the credentials **“User Name :- Admin” , “Password :- 123”** |
| Pre-Conditions | The user must supply the correct credentials in order to access the system. |
| Post-Conditions | The Admin Can Update any changes. |

***UI Screen Flow:-***



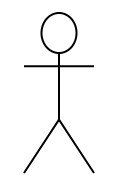
|  |  |
| --- | --- |
| Brief Description | **Categories Page** |
| Basic Flow | * Click on Category. * Add image * Add the Product or Update Product or Delete the product. * Click on save |
| Alternate Flow | If The image is not in the format of JPG, PNG, JEPG then User gets an error message as “The image must be in the format of JPG, PNG, JEPG ”. |
| Validation | The entered details will be validated against all the validation rules. |
| Pre-Conditions | The admin should make any changes or need to add the Category. |
| Post-Conditions | After Adding or updating that changes will be made in Users page |

***UI Screen Flow:-***



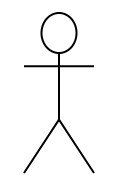
|  |  |
| --- | --- |
| Brief Description | **Products Page** |
| Basic Flow | * Click on Products. * Select the Category * Add image * Add the Product or Update Product or Delete the product. * Click on save |
| Alternate Flow | If The image is not in the format of JPG, PNG, JEPG then User gets an error message as “The image must be in the format of JPG, PNG, JEPG ”. |
| Validation | The entered details will be validated against all the validation rules. |
| Pre-Conditions | The admin should make any changes or need to add the product. |
| Post-Conditions | After Adding or updating that changes will be made in Users page |

***UI Screen Flow:-***



|  |  |
| --- | --- |
| Brief Description | **Update Status Page** |
| Basic Flow | * Click on Update Status. * Update status to delivered or dispatched. * Click on update. |
| Alternate Flow | If Admin Didn’t update the status the By default the status will be set as pending |
| Validation | The Updated details will be validated against all the validation rules. |
| Pre-Conditions | The admin should Update the status of the product. |
| Post-Conditions | After updating that changes will be made in Users page |

***UI Screen Flow:-***



|  |  |
| --- | --- |
| Brief Description | **Selling report Page** |
| Basic Flow | * Select Selling Report. * Select From date and to date of the report. * Click on search. |
| Alternate Flow | If Admin Selects to date after present days date then Alert message will pop up as “ ToDate cannot be greater than current date!” |
| Validation | The To date will be validated against all the Present days date. |
| Pre-Conditions | The admin should select from date and to date. |
| Post-Conditions | After selecting dates the report will be viewed. |

# References

* <https://dotnet.microsoft.com/learn/csharp>
* <https://www.w3schools.com/css/default.asp>