Software Design Description for Restaurant Reservation Web Application

Youssef Ahmed,Mohamed Hossam,Mark Saleh,Ahmed Tarek,Youssef Hany
Team 18
Supervised by: Dr. Essam Eliwa, Eng. Omar Magdy

June 3, 2023

Table 1: Document version history

Version	Date	Reason for Change
1.0	15-May-2023	SDD first version's description are defined.
2.0	19-May-2023	UML Diagrams Updated.
3.0	2-June-2023	All diagram colours changed after Dr.Essam Feedback on colours
3.1	3-June-2023	Checked Grammar and sentence structure and added a diagram
3.2	3-June-2023	Changed GitHub Diagram to be lightmode and show all insights

GitHub: https://github.com/Mark-S2004/Maestro-food-truck-website

Contents

Abstract

The design for a full-stack website for a meal ordering system is described in this software design description paper. The website intends to offer users a simple interface for browsing meal selections, placing orders, and making payments online. The system would make it easier for restaurant operators to manage their menus, keep track of orders, and handle payments.

1 Introduction

1.1 Document Purpose

This Software Design Description (SDD) document's main goal is to give readers a thorough overview of the design for a fully functional food ordering website. It describes the requirements matrix, data design, data design viewpoints, and system architecture.

1.2 Document Scope

This document focuses on the design aspects of the food ordering website project. It provides detailed information on the system architecture, design patterns, algorithms, user interface, and data design. The document does not cover implementation details or specific coding techniques.

1.3 Intended audience

The development team, project stakeholders, software architects, and designers make up the document's intended audience. It presupposes a fundamental comprehension of design patterns and software engineering principles. To safeguard user data and guarantee secure transactions, security and privacy factors should be taken into account during the construction and implementation of the website.

1.4 Reference Material

For this SDD, the following documents served as main information sources:

- Software Requirements Specification (SRS)
- Testing Plan

1.5 Definitions and Acronyms

Information on software design is communicated mostly through these Terms and Definition.

Term	Definition
Software Design Document (SDD)	used as the main channel for providing information about software design
Design Entity	An element of a design that is structurally and functionally distinct from other elements.
Design rationale	Information that documents the designer's thinking process that resulted in the system's design, including design possibilities, compromises taken into consideration, conclusions made, and the arguments for those decisions. decisions.

2 System Overview

2.1 System Scope

A platform for users to browse restaurants, see menus, place orders, and make payments online is intended to be provided through the meal ordering website. Additionally, it has features that allow restaurant operators to control their menus, keep track of orders, and handle payments. A full-stack web application, the system will include both frontend and backend elements.

2.2 System objectives

The following are the project's primary goals:

- Create a simple and easy interface so that clients can browse and place orders.
- Integrate online payments securely and easily.
- Give restaurant owners a management interface so they can control menus and orders.
- Ensure performance and scalability to support several concurrent users.
- Support integration with third-party services, such as payment gateways and geolocation.

2.3 Project Timeline

- 10 March 2023 : Proposal Document
- 31 March 2023 : Front end
- 18 April 2023 : SRS Document
- 19 May 2023 : SDD Document
- 19 May 2023 : Full project

3 Design viewpoints

3.1 Context viewpoint

The system is seen from a "black box" perspective in the context viewpoint. It specifies the services offered by the system and the users and stakeholders who engage with it. This perspective includes the following components in 3.1.1 and 3.1.2 .

3.1.1 Offered Services and Actors

The following services are provided by the food ordering website:

- **User Registration and Authentication:** Users can register accounts and log in to get personal options.
- **Browsing Restaurants and Menus:** Searching for restaurants, viewing their menus browsing the food options are all options open to users.
- **Placing Orders:** Users can choose from a variety of foods, personalise their orders, and place them for delivery or pickup.
- **Online Payment:** Using secure payment options, users can pay for their orders online.
- **Order Tracking:** Users are able to follow the progress of their orders and get updates.
- **Restaurant Management:**Owners can adjust item availability, modify their menus, and view order specifics.

3.1.2 UML context diagram

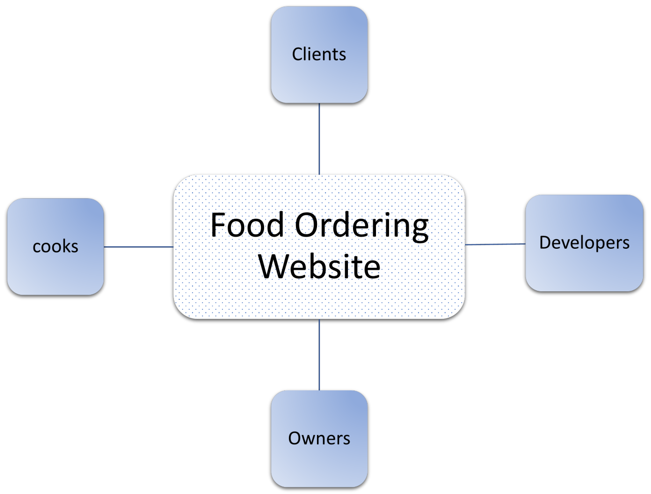


Figure 1: Context Diagram for the Food Ordering System

3.1.3 UML use case diagram

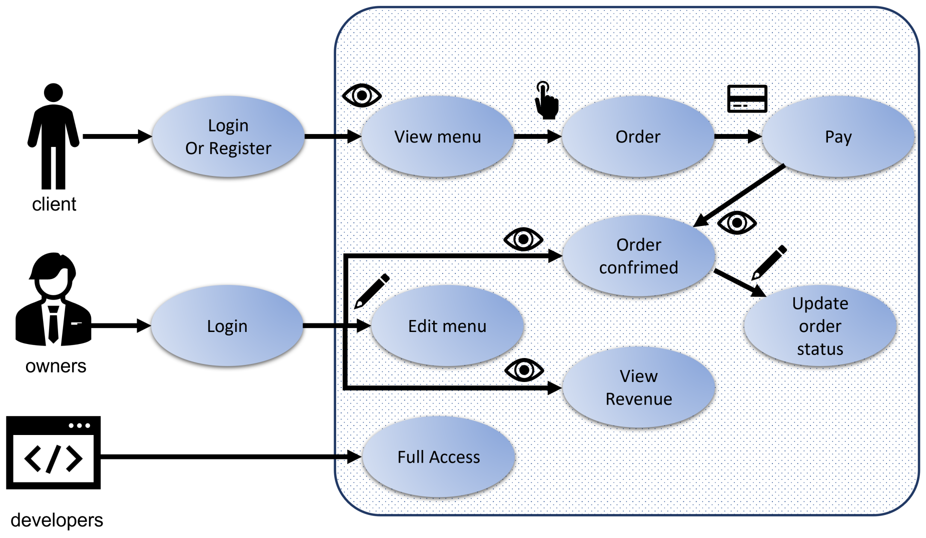


Figure 2: Use Case Diagram for the Food Ordering System

3.2 Composition viewpoint

The Composition viewpoint describes the subsystems, components, and modules that make up the structure of the design subject. Each subsystem's duties are described, along with how they cooperate to produce the intended functionality. A UML package diagram can serve as a representation of this point of view.

3.2.1 Architectural Design Description

The food ordering website can gain from a structured architectural design that supports code organisation, re-usability, and maintainability by implementing the **MVT** pattern.

- **Model:**
  - Create data models for any relevant data entities, including orders, menu items, users, and others.
  - Implement validation rules and business logic into the models.
  - Control the storing, retrieving, and manipulating of data.
- **View:**
  - Take care of HTTP requests and responses.
  - Data from the Model is retrieved and passed to the Template.
  - Implement any necessary logic for receiving requests, authenticating users, and authorising access.
- **Template:**
  - Make HTML templates with dynamic data placeholders.
  - Insert data that has been pulled from the View using template tags or expressions.
  - Define the design, presentation, and aesthetics of the user interface.

3.2.2 Design Rationale

A separation of concerns is encouraged by the **MVT** pattern in web frameworks like Django, where the Model manages data-related operations, the View manages request processing, and the Template manages user interface presentation. This separation keeps the business logic and data administration independent of the presentation layer, allowing for modular development, simpler maintenance, and flexibility in user interface customization.

3.3 Logical viewpoint

The MVT design makes it easy to depict the system's static structure using a UML class diagram. The diagram will display the classes, interfaces, and connections between them, giving a clear picture of the logical organisation of the system. Each class will also be described in detail in a table, ensuring that all of their characteristics and behaviours are fully understood.

Table 2: Description of each class

Food Ordering System:	Represents the key system component in charge of ordering, restaurants, and users.
User	Represents a customer of the online ordering system for food. includes characteristics like userId, name, email, and password. contains techniques for user actions such as sign-in, sign-out, ordering, and checking order history.
Restaurant	Represents a restaurant in the system. Contains attributes like restaurantId, name, address, and menuItems. Provides methods for managing menu items like adding, removing, updating, and viewing the menu.
Order	Represents a purchase order a user has made. includes information such as orderId, user, restaurant, items, and totalAmount. outlines ways to manage order items, figure out the total, and place the order.
MenuItem	Represents a menu item that a restaurant offers. includes details such as itemId, name, description, and price. outlines techniques for obtaining item details including name and price.
OrderItem	Represents a product inside an order. includes characteristics such as itemId, quantity, and subtotal. offers techniques for obtaining item data such as itemId, number, and subtotal.

3.4 Patterns use viewpoint

The design patterns used in the project will be listed in this part along with an explanation of why they were chosen. Design patterns are tried-and-true approaches to recurrent design issues that offer a useful way to organise and arrange code. They support the reuse, modularity, and maintainability of software. Let's talk about a typical design pattern in the framework of the MVT (Model-View-Template) architecture even though the specific design patterns utilised in the project may differ.

Observer Pattern:
The Observer pattern is used to create a one-to-many dependency between objects. The subject (observable) keeps track of a list of observers and automatically alerts them when a state change occurs. The Model governs data-related operations in the MVT architecture, the View manages request processing, and the Template manages user interface display. In situations when the Model must inform the View or Template of any changes in data or state, the Observer pattern is especially helpful.

3.5 Algorithm viewpoint

Here are some of the flowcharts simply describing parts of the system.

3.5.1 Login

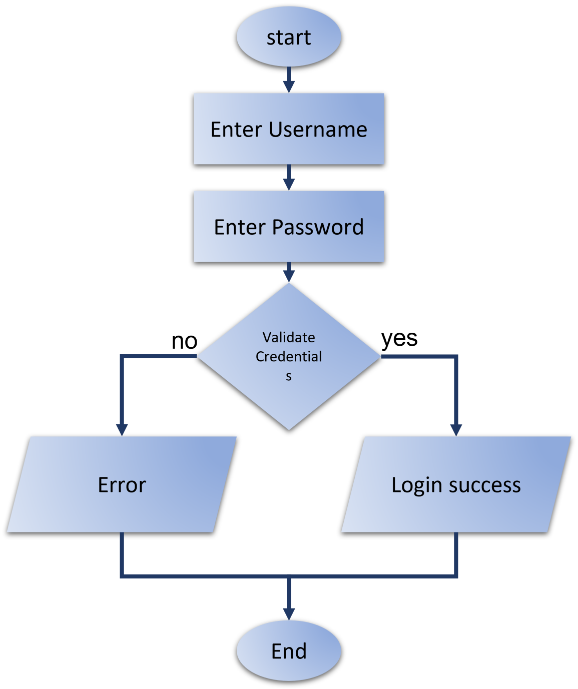


Figure 3: Flowchart Diagram for the Login Process

3.5.2 Ordering

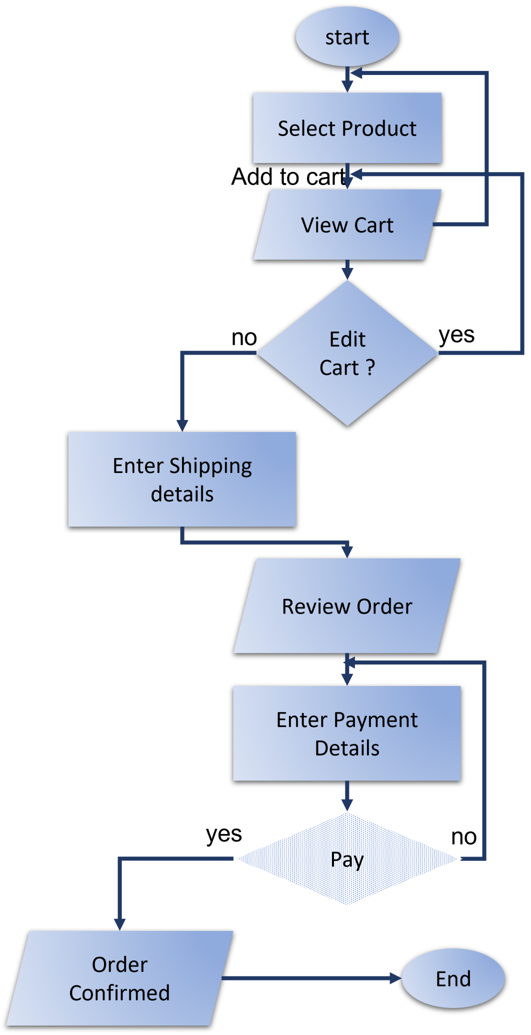


Figure 4: Flowchart Diagram for the Food Ordering Process

3.6 Interaction viewpoint

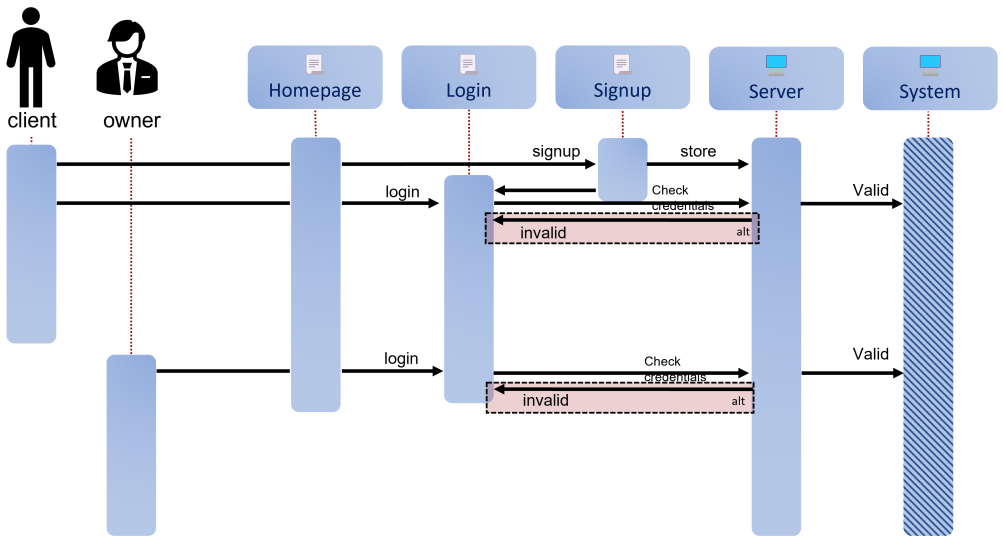


Figure 5: User Registration Sequence Diagram

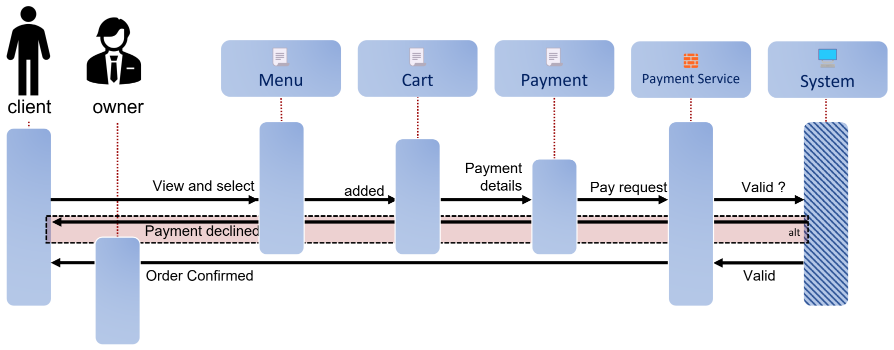


Figure 6: Ordering Sequence Diagram:

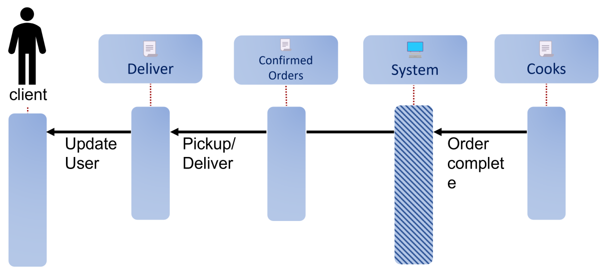


Figure 7: Order Cooked Sequence Diagram

4 Data Design

4.1 Data Description

Multiple data entities and their relationships would be included in the system. The main data entities would be menu items, orders, payments, and customer information. The relationships between these entities would be as follows:

- Customers would order food from the menu.
- Orders would contain one or more menu items
- Payments would be linked to certain orders placed by customers.

5 Human Interface Design

5.1 User Interface

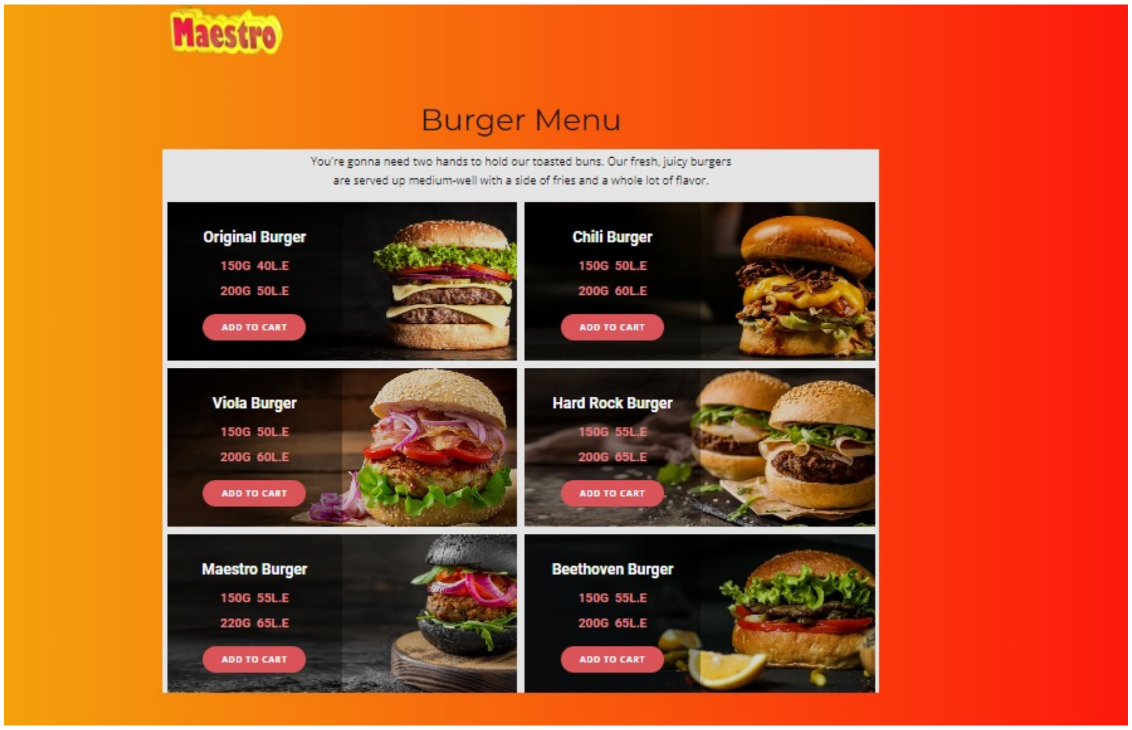
The user interface consists of multiple pages that the user will use as he/she navigates the website, For example : the home page, the registration and sign in ,sign up page, the menu page that contain list of all available burgers with their names and prices, the cart page which collect all items that have been added to the cart and it provide the customer price and quantity of each item and the total cost of the order and the checkout page which provide the customer with the total price of the order and he/she can choose whether they want their order takeaway or delivery and they can add any comments in it .The user interface will allow users to quickly find what they are looking for.

- A clear and concise menu that includes all available items.
- The ability to add items to a cart and view the total cost of the order.
- The ability to choose between delivery or pickup options.

User experience design, which concentrates on the wider picture—that is, the entire experience, not just the interface—is a subset of interface design, which focuses on the layout and operation of interfaces.These are some remarkable things to consider while designing an interface:

- **Usability**: With an organised layout and clear labelling, the interface should be simple to use and navigate.
- **Accessibility**: Ensure that all users can utilise the interface. Make use of bold fonts, contrasting colours, and labels on your button.
- **Clarity**: Make sure the UI elements are understandable and clear. To express information, use clear language and understandable icons.

5.2 Screen Images



Burger Menu

You're gonna need two hands to hold our toasted buns. Our fresh, juicy burgers are served up medium-well with a side of fries and a whole lot of flavor.

Original Burger
150G 40L.E
200G 50L.E
ADD TO CART

Chili Burger
150G 50L.E
200G 60L.E
ADD TO CART

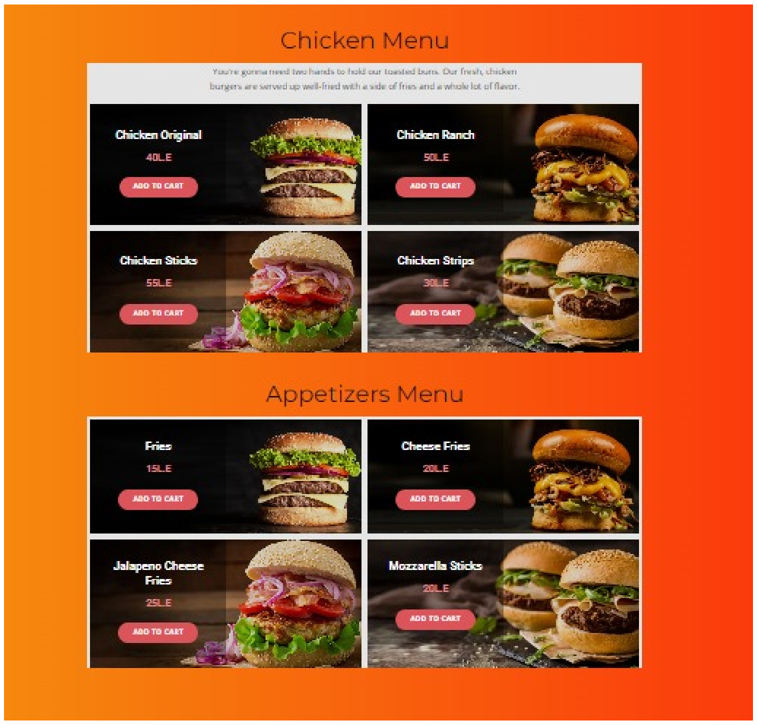
Viola Burger
150G 50L.E
200G 60L.E
ADD TO CART

Hard Rock Burger
150G 55L.E
200G 65L.E
ADD TO CART

Maestro Burger
150G 55L.E
220G 65L.E
ADD TO CART

Beethoven Burger
150G 55L.E
200G 65L.E
ADD TO CART

Figure 8: Menu page



Chicken Menu

You're gonna need two hands to hold our toasted buns. Our fresh, chicken burgers are served up well-fried with a side of fries and a whole lot of flavor.

Chicken Original
40L.E
ADD TO CART

Chicken Ranch
50L.E
ADD TO CART

Chicken Sticks
55L.E
ADD TO CART

Chicken Strips
30L.E
ADD TO CART

Appetizers Menu

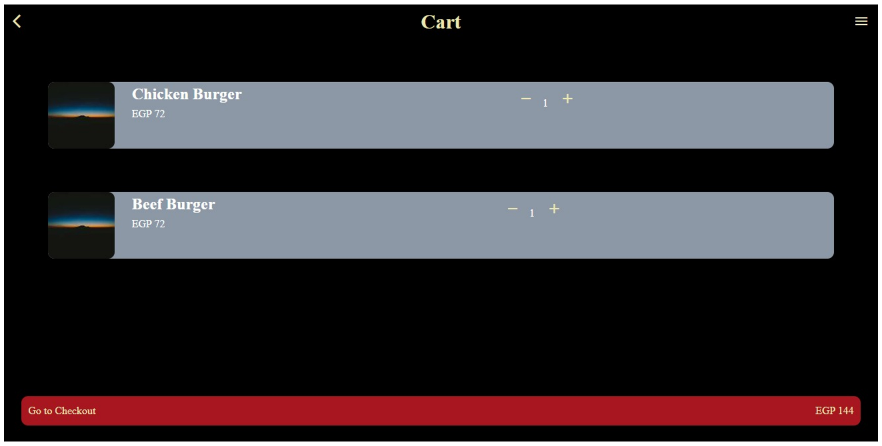
Fries
15L.E
ADD TO CART

Cheese Fries
20L.E
ADD TO CART

Jalapeno Cheese Fries
25L.E
ADD TO CART

Mozzarella Sticks
20L.E
ADD TO CART

Figure 9: Menu page



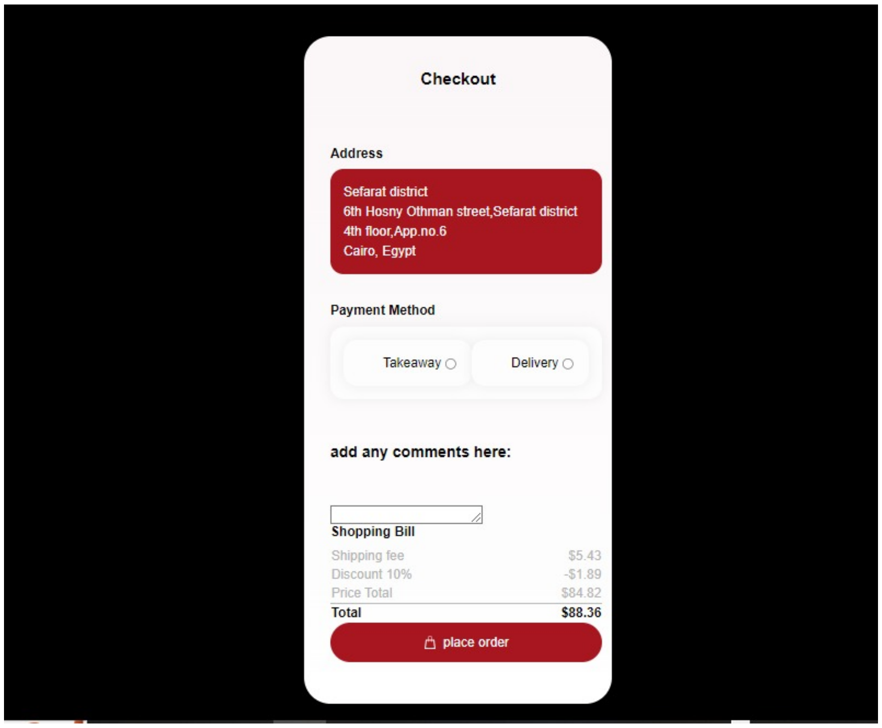
Cart

Chicken Burger
EGP 72
− 1 +

Beef Burger
EGP 72
− 1 +

Go to Checkout
EGP 144

Figure 10: Cart page


Checkout

Address

Sefarat district
6th Hosny Othman street,Sefarat district
4th floor,App.no.6
Cairo, Egypt

Payment Method

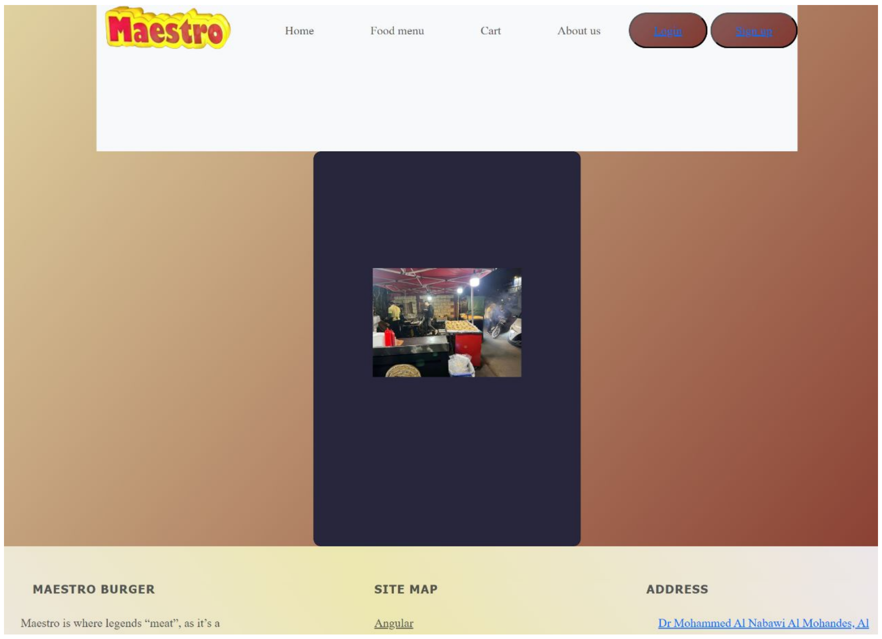
Takeaway ○ Delivery ○

add any comments here:

Shopping Bill

Shipping fee	$5.43
Discount 10%	-$1.89
Price Total	$84.82
Total	$88.36

Figure 11: Checkout page



MAESTRO BURGER

Maestro is where legends “meat”, as it’s a

SITE MAP

Angular

ADDRESS

Dr Mohammed Al Nabawi Al Mohandes, Al

Figure 12: Home page

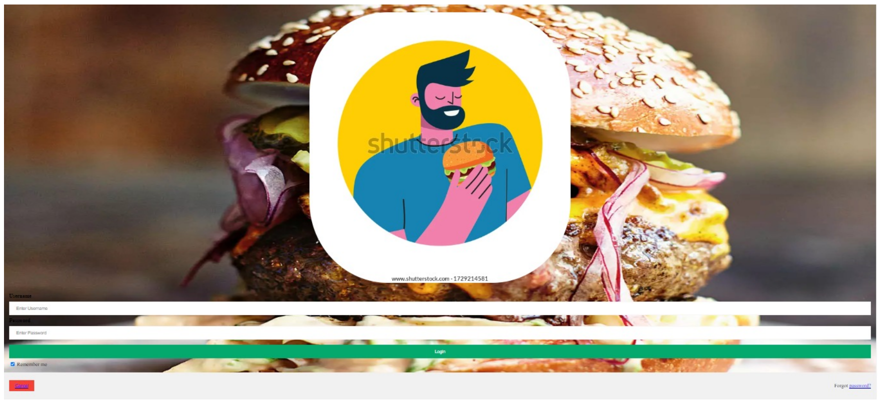
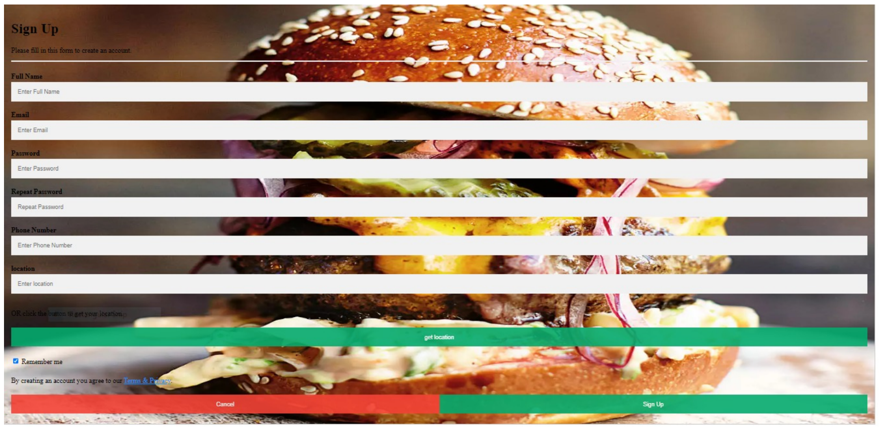


Figure 13: Sign in page



Sign Up

Please fill in this form to create an account.

Full Name

Enter Full Name

Email

Enter Email

Password

Enter Password

Repeat Password

Repeat Password

Phone Number

Enter Phone Number

location

Enter location

OR click the button to get your location

get location

☑ Remember me

By creating an account you agree to our Terms & Privacy.

Cancel Sign Up

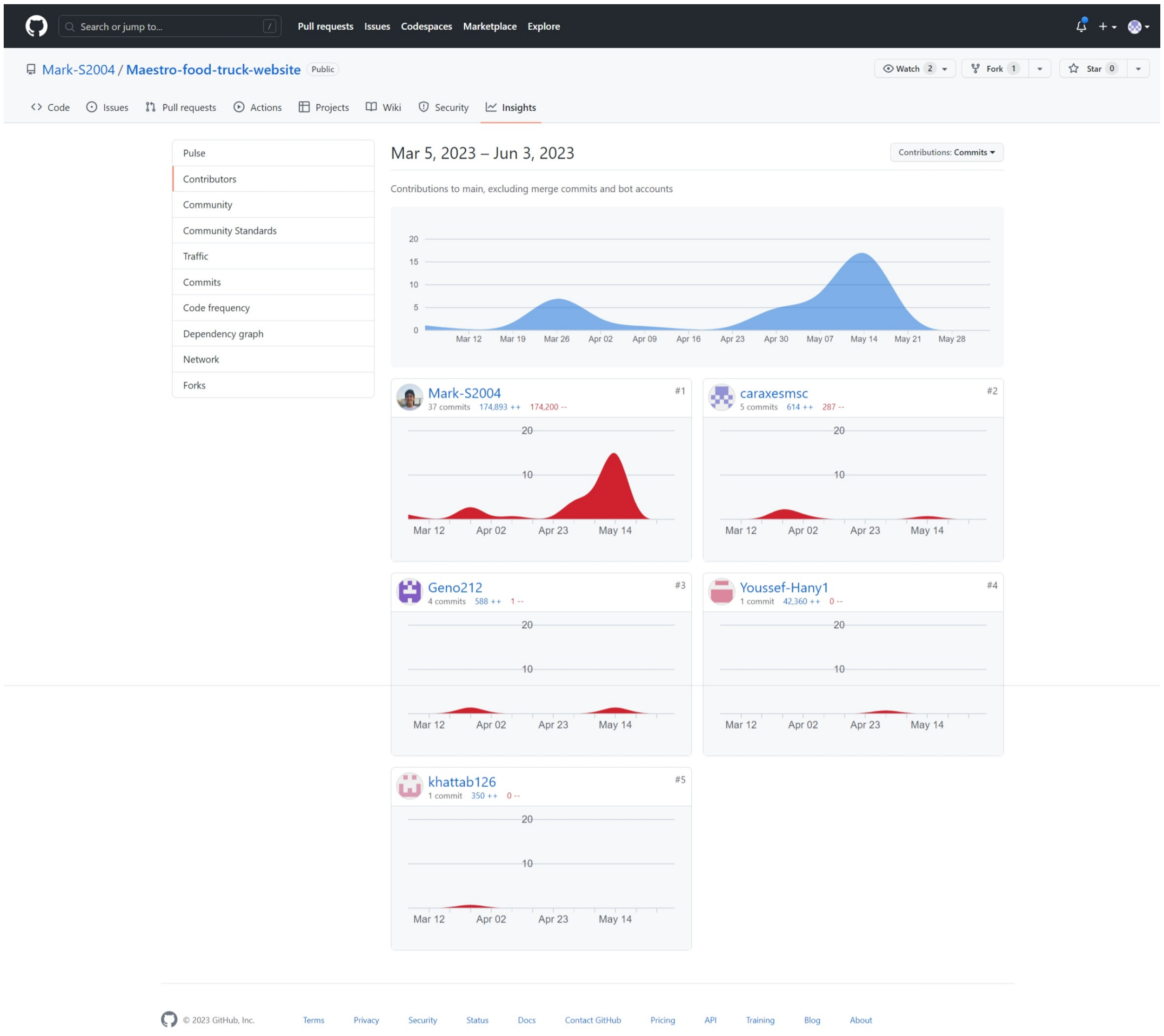
Figure 14: Sign up page

6 Requirements Matrix

Table 3: Requirements Ratrix

Req. ID	Req. desc	Class	Test cases id	Status
FR06	Add to cart	customer	TC9, TC11	Developed
FR06	Remove from cart	customer	TC10	Developed
FR01	Order	customer	TC01, TC02	Developed
FR07	Create/delete Customer	admin	TC14, TC12	Developed
FR08	Restock	admin	TC13	Developed
FR08	Setprice	admin	TC13	Developed
FR02	Clear Cart	cart	TC03	Developed
FR02	Calculate Total	cart	TC04	Developed
FR03	Login	customer	TC05, TC06	Developed

6.1 Github



Mar 5, 2023 – Jun 3, 2023

Contributions to main, excluding merge commits and bot accounts

Mark-S2004 #1
37 commits 174,893 ++ 174,200 --

caraxesmsc #2
5 commits 614 ++ 287 --

Geno212 #3
4 commits 588 ++ 1 --

Youssef-Hany1 #4
1 commit 42,360 ++ 0 --

khattab126 #5
1 commit 350 ++ 0 --

Figure 15: Github