Chhatrapati Shahu Ji Maharaj University

Project Report

on

MEALSTATION

Submitted in Partial Fulfillment of the Requirements for the Degree of

Bachelors of Computer Application

By:

Brajendra Singh

[0302531]

Under the Supervision of

Mr. Nitin Dwivedi

PSIT College of Higher Education

Kanpur-Agra-Delhi National Highway – 2, Bhauti, Kanpur

(2022-2023)

CERTIFICATE

This is to certify that project entitled **"MEALSTATION"** submitted for partial fulfillment of the degree of **BCA** under the Department of **Bachelor of Computer Application** to through **PSIT College of Higher Education,** Kanpur, done by **Mr. Brajendra Singh**, **Roll No. 0302531** is an authentic work carried out by me under the guidance of **Mr. Nitin Dwivedi** . The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

Internal Examiner/Guide External Examiner

Head of Department

Declaration

I hereby declare that the project entitled **"MEALSTATION"** submitted for the Bachelor of Computer Application degree is my original work and the project has not formed the basis for the award of any other degree of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature of the student

(Brajendra Singh)
(0302531)
BCA
PSIT College of Higher Education, Kanpur

ACKNOWLEDGEMENT

Presentation inspiration and motivation have always played a key role in the success of any venture.

I express my sincere thanks to **Dr. APS Bhaudaria, Dean, PSIT College of Higher Education, Kanpur.**

I pay my deep sense of gratitude to **Ms. Pragati Upadhyay (HOD)** of BCA Department, **PSIT College of Higher Education** to encourage me to the highest peak and to provide me the opportunity to prepare the project. I am immensely obliged to **my friends** for their elevating inspiration, encouraging guidance and kind supervision in the completion of my project.

I feel to acknowledge my indebtedness and deep sense of gratitude to my guide **Mr. Nitin Dwivedi** whose valuable guidance and kind supervision given to me throughout the course which shaped the present work as its show.

Last, but not the least, **my parents** are also an important inspiration for me. So with due regards, I express my gratitude to them.

ABSTRACT

The purpose of Meal Station is to automate the existing manual system by the help of computerized equipment's and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Meal Station, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

The aim is to automate its existing manual system by the help of computerized equipment's and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically the project describes how to manage for good performance and better services for the clients.

INDEX

- Chapter-I:
  - Introduction 1
  - Existing System 2
  - Problems in Existing System 3
  - Proposed System 4
  - Objective of the Project 5
- Chapter-II:
  - Modules & Description of the Modules 6
  - System Requirements 8
  - Technology used in project 9
- Chapter-III:
  - Feasibility Study 34
  - Software Development Life Cycle 37
  - Data Flow Diagram 41
- Chapter-IV:
  - Data Base Schema Design 44
  - Screen/Snap-Shots of the project 50
- Conclusion 66
- References 67

Introduction

The “Meal Station” has been developed to override the problems prevailing in the practicing manual system. This website is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The website is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly. Meal Station, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the information of Employee, Canteen, Stock, Customer, Sales. Every Canteen Management System has different Canteen needs, therefore we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

Canteen management system is to provide fast services to their college students, Staffs etc. Usually, People have to go to canteen and order the foods and they have to wait in queue for a long time to get the orders, But with the help of this you just have to follow a very simple process to order your stuffs. And you need not to wait in the long queue. A canteen facility is a supplementary system that is provided by organizations for their employees/students. Organizations with large numbers of employees cannot handle a canteen with manual processes. Our canteen management system provides a friendly User Interface for numerous food outlets, menu design, billing features and lots more. Implementation of such a system makes the operation of the kitchen and the whole of the canteen as effective and quick as possible.

Meal Station, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

Existing System

The existing system is a cash and paper-based system. The payment and process take a lot of time as the customer has to pay the exact amount and wait for the change. If the change is not available at the time, a coupon is provided which should be shown at the counter at the next purchase.

In the existing system only we have to go to canteen and or order the foods and we have to wait in queue for a long time to get the orders. This is the waste of time and energy.

The existing system is a cash and paper-based system. The payment and process take a lot of time as the customer has to pay the exact amount and wait for the change. If the change is not available at the time, a coupon is provided which should be shown at the counter at the next purchase.

Problems in Existing System

This system is generally advantageous for avoiding spending time waiting in the queue by posting orders directly to the kitchen without delay and also by scheduling orders ahead of time. It saves time and also the technique dealing with is easy. The proposed Canteen Management System is an adept solution for chaos at college canteens. Highlights of cloud for example auto-scaling, load adjusting and pay as you go improve the working of the system and to some extent unravel the motivation behind the proposed system.

Drawbacks of Existing System :

- Time Consuming
- Consumes large volume of pare work
- Needs manual calculations
- Don’t have exact food court record of each student
- Due to manual system there are chances of loss of some money
- To avoid all these limitations and make the working more accuracy the system needs to be computerized

Proposed System

The main aim of this project MEALSTATION system is to provide fast services to their college students, Staff etc. Usually People have to go to canteen and order the foods and they have to wait in queue for along time to get the orders. But with the help of this you just have to follow a very simple process to order your stuffs. And you need not to wait in the long queue. This website will provide the list of different menu list with different categories. User can select any item from canteen and can order for it by using Pay on Delivery or can pay using QR Code embedded in the webpage

Expected Advantages of Proposed System.

- The system is very simple in design and to implement. The system requires very low system resources and the system will making almost all configurations. It has got following features
- Reduces wastage of food items
- Easy for calculating the exact of canteen vendor
- College authority can see the canteen's income and total orders
- Ensure data accuracy
- Minimize manual data entry
- Greater efficiency

Objective of the Project

The main objective of the Meal Station is to manage the details of Canteen, Employee, Customer, Sales, Item, and Category. It manages all the information about Canteen, Stock, Item Category, and Canteen. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an website to reduce the manual work for managing the Canteen, Employee, Stock, Customer. It tracks all the details about the Customer, Sales, Item, and Category.

The main objective of the project on canteen management system is to manage the details of Canteen, Students, Item, Stock.

The purpose of the project is to build an application program to reduce the manual work for managing the Canteen, Students, Customer, Item. It tracks all the details about the Item, Stock, Sales.

Functionalities provided by Canteen Management System are as follows:

- Provides the searching facilities based on various factors Such as Canteen, Item, Stock, Sales.
- Manage the information of Canteen, Stocks & also Students.
- Editing, adding and updating of Records is improved which results in proper resource management of Canteen data.
- Integration of all records of Sales.

Advantages of project

- Completely automated online ordering of food in a canteen.
- Order can be placed using personal android phones.
- Food ordering pages that look and feel exactly the same as the existing restaurant website.
- Food ordering pages hosted on secure and special server so no risk of customers getting redirected to servers where competitors' websites are listed
- Developed using the latest website programming protocols for minimum server loads and ultra-fast loading and processing.
- Simple user-interface, Admin interface, Admin Panel for creation and configuration of menu groups, menu items, etc.
- Built-in facility to set modifiers on different menu items
- Facility to create modifier groups, individual modifier items and assign modifier item to different groups.
- Single and individual Admin Panel and login for each Canteen.
- Detailed summary of orders placed with option to search order, update order status, see orders, etc.
- Various reports to view total sale, details of registered members with facility to see report.

Modules & Description of the Modules

- **CafeTeria Module:** Used to managing website setting, database and url mapping. This module simple tell the types url or clicked button to redirected url are valid or invalid. It manages all the url or user end as well as admin end. It is responsible to redirecting to the use to the created url. The setting is manages by this modules also how much installed apps in your project, the template directory and the database connection engine, name, username, password, host, port etc. It also contains the validation setting of the user as well as admin. This is also manages the roots of the static files and the duplicate path of the static filed and the email smtp configuration how to sent reset password email to the admin and user in request.

- **Carousel Module:** Used to managing website banner which can be modified by the admin according to the need. The admin can modify the top images displaying in the website at the user end, he can simply put the latest thing available in the store or wishing the festival to the user of displaying any type of advertisement and giving the information to the user.

- **Category Module:** Used to managing the categories of the canteen which are available and admin can add, edit and delete the category. The manipulation of the category of the available item is totally at the admin hand, he can easy add the category if any new category item is introduced in the canteen.

- **Customer Module:** Used for managing the details of Customer. How the customer can register. What are the validation are required and password encryption. The validation of the not submitting the form empty or try to register with some empty field in the form and take care of the password entered by the user for maintaining the security and the transparency of the system. Use cannot register with same email id if he is already registered with before. If he forget the password he can simply put a request in the forget password section and an email will be sent to the registered email address is exist with the password reset instruction.

- **Admin Module:** Used for managing the information and details of admins. One admin can add another admin and can modify the privileges to access the data. This module is the backbone of the canteen management system because it has whole authority to create, view, update and delete the category as well as items. Admin can simply manage the order placed by the user. Admin can update the status of the order if it is ready or delivered or pending according to the need. Admin can also make the item available or unavailable if the item is out of stock. If the item is out of stock then user cannot place the order and if the item is available to the some user cart then he also cannot place the order.

- **Service Time Module:** Used for managing the canteen open and close time details. Admin can add, update and remove the opening and closing time of the canteen. This module is basically if the admin does not want to user to place the order at some point of time then he can simply put the opening and closing time. The user can only place the order in the given time by the admin.

- **Cart Module:** This module is used to manage the card of the user who is currently logged in the system. If the user log out of the system then he cannot access the cart with out again logging in the system. This module is also take care of the total amount and listing the item in a tabular like format.

- **Order Module:** This module display the order of the user currency in the session or logged in the system. This module also displays the ordered item in the tabular like format so user can easily understand his orders

System Requirements

Hardware Requirements:

- Hard Disk : 50 GB or above
- Processor : Dual Core or above
- Processor Speed : 2.2 GHz
- Ram : 2GB
- Monitor : Display Panel (1024*76)

Software Requirements

- Operating System : Windows, Linux, Mac etc.
- Web Browser : Microsoft Edge, Mozilla Firefox, Chrome
- Fronted : HTML, CSS, Bootstrap
- Technology :- PYTHON 3, DJANGO 4.0.4
- Database Server :- My SQL [8.0.30]

Technology used in project

PYTHON :

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected.

It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0.Python 2.0 was released in 2000 and introduced new features such as list comprehensions, cycle-detecting garbage collection, reference counting, and Unicode support. Python 3.0, released in 2008, was a major revision that is not completely backward-compatible with earlier versions. Python 2 was discontinued with version 2.7.18 in 2020.

python is a general-purpose, object-oriented programming language that has several implications across the software, web development, data science and automation environments. The language's dynamic semantics, high-level built in data structures, dynamic typing and dynamic binding make it one of the most useful languages for rapid application development.

Python is a high-level programming language that is known for its simplicity and readability. It was created by Guido van Rossum and first released in 1991. Python is widely used in various domains such as web development, scientific computing, data analysis, artificial intelligence, and more.

Why

Python is omnipresent, and people use numerous Python-powered devices on a daily basis, whether they realize it or not. There are billions of lines of code written in Python, which means almost unlimited opportunities for code reuse and learning from well-crafted examples. What's more, there is a large and very active Python community, always happy to help.

There are also a couple of factors that make Python great for learning:

- It is easy to learn – the time needed to learn Python is shorter than for many other languages; this means that it's possible to start the actual programming faster;

- It is easy to use for writing new software – it's often possible to write code faster when using Python;
- It is easy to obtain, install and deploy – Python is free, open and multiplatform; not all languages can boast that.

Programming skills prepare you for careers in almost any industry, and are required if you want to continue to more advanced and higher-paying software development and engineering roles. Python is the programming language that opens more doors than any other. With a solid knowledge of Python, you can work in a multitude of jobs and a multitude of industries. And the more you understand Python, the more you can do in the 21st Century. Even if you don't need it for work, you will find it useful to know

Python is used for server-side web development, software development, mathematics, and system scripting, and is popular for Rapid Application Development and as a scripting or glue language to tie existing components because of its high-level, built-in data structures, dynamic typing, and dynamic binding. Program maintenance costs are reduced with Python due to the easily learned syntax and emphasis on readability. Additionally, Python's support of modules and packages facilitates modular programs and reuse of code. Python is an open source community language, so numerous independent programmers are continually building libraries and functionality for it.

Python Use Cases

- Creating web applications on a server
- Building workflows that can be used in conjunction with software
- Connecting to database systems
- Reading and modifying files
- Performing complex mathematics
- Processing big data
- Fast prototyping
- Developing production-ready software

Professionally, Python is great for back end web development, data analysis, artificial intelligence, and scientific computing. Developers also use Python to build productivity tools, games, and desktop apps.

Features and Benefits of Python

- Compatible with a variety of platforms including Windows, Mac, Linux, Raspberry Pi, and others
- Uses a simple syntax comparable to the English language that lets developers use fewer lines than other programming languages

- Operates on an interpreter system that allows code to be executed immediately, fast-tracking prototyping
- Can be handled in a procedural, object-orientated, or functional way

Here are some key features of Python:

Easy to Read and Write: Python emphasizes code readability with its clean and straightforward syntax, making it easier to understand and write code.

Interpreted Language: Python is an interpreted language, which means that code is executed line by line, without the need for a separate compilation step. This makes development and debugging faster.

Cross-platform: Python is available on multiple platforms, including Windows, macOS, and Linux, allowing developers to write code once and run it on different operating systems.

Large Standard Library: Python comes with a vast standard library that provides a wide range of modules and functions, making it easy to accomplish common tasks without needing to write additional code.

Third-Party Libraries: Python has a rich ecosystem of third-party libraries and frameworks, such as NumPy, pandas, TensorFlow, Django, and Flask, which extend its capabilities for specialized purposes like scientific computing, data analysis, machine learning, and web development.

Object-Oriented Programming (OOP): Python supports object-oriented programming, allowing developers to create reusable and modular code using classes, objects, and inheritance.

Dynamic Typing: Python is dynamically typed, meaning that you don't need to explicitly declare variable types. The type of a variable is inferred at runtime, providing flexibility but also requiring attention to potential type-related errors.

Python's versatility and extensive libraries have contributed to its popularity among developers. It has a large and supportive community that continuously contributes to its growth and offers resources, tutorials, and packages to help developers in their projects.

Python Syntax

- Somewhat similar to the English language, with a mathematical influence, Python is built for readability
- Unlike other languages that use semicolons and/or parentheses to complete a command, Python uses new lines for the same function

- Defines scope (i.e., loops, functions, classes) by relying indentation, using whitespace, rather than braces (aka curly brackets)

Python Flexibility

Python, a dynamically typed language, is especially flexible, eliminating hard rules for building features and offering more problem-solving flexibility with a variety of methods. It also allows uses to compile and run programs right up to a problematic area because it uses run-time type checking rather than compile time checking.

The Less Great Parts of Python On the down side, Python isn't easy to maintain. One command can have multiple meanings depending on context because Python is a dynamically typed language. And, maintaining a Python app as it grows in size and complexity can be increasingly difficult, especially finding and fixing errors. Users will need experience to design code or write unit tests that make maintenance easier.

Speed is another weakness in Python. Its flexibility, because it is dynamically typed, requires a significant amount of referencing to land on a correct definition, slowing performance. This can be mitigated by using alternative implementation of Python (e.g. PyPy).

Python and AI AI researchers are fans of Python. Google TensorFlow, as well as other libraries (scikit-learn, Keras), establish a foundation for AI development because of the usability and flexibility it offers Python users. These libraries, and their availability, are critical because they enable developers to focus on growth and building.

DJANGO :

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source.

Django is a Python-based web application framework that is free and open source. A framework is simply a collection of modules that facilitate development. They're grouped together and allow you to build apps or websites from scratch rather than starting from scratch.

"Rapid development and clean, pragmatic design" are key benefits of Django. When installed on a web server, the Django web framework can assist developers in quickly creating a feature-rich, secure, and scalable web front end.

For example, developers should not create their login screens or login processing. There are far too many possibilities for things to go wrong. Frameworks take care of this for you and handle all the tricky cases.

Django is a high-level Python web framework that follows the Model-View-Controller (MVC) architectural pattern. It provides a set of tools and libraries that simplify the development of web applications by emphasizing reusability, modularity, and the "don't repeat yourself" (DRY) principle.

Built-in admin

Django has an in-built administration interface which lets you handle your models, user/ group permissions and to manage users. With model interface in place, there is no need for a separate database administration program for all but advanced database functions.

Features of Django

- Rapid Development
- Secure
- Scalable
- Fully loaded
- Versatile
- Open Source
- Vast and Supported Community

Here are some more key features of Django:

Model-View-Controller (MVC): Django follows a slightly modified version of the MVC pattern called Model-View-Template (MVT). The model defines the data structure and interacts with the database, the view handles the logic and presentation of data, and the template defines the layout and structure of the final HTML output.

ORM (Object-Relational Mapping): Django provides an ORM that allows developers to interact with the database using Python objects instead of writing raw SQL queries. This simplifies database operations and promotes code reusability.

URL Routing: Django has a powerful URL routing system that maps URLs to corresponding view functions, allowing for clean and flexible URL configurations.

Template System: Django's template system enables developers to separate the design and presentation logic from the Python code. Templates are written in HTML with additional syntax to insert dynamic data and perform logic operations.

Form Handling: Django provides a comprehensive form handling system that simplifies form creation, validation, and processing. It includes built-in form validation, CSRF protection, and error handling.

Authentication and Authorization: Django offers a robust authentication and authorization system, making it easy to manage user authentication, permissions, and access control.

Admin Interface: Django provides a customizable admin interface out of the box, allowing developers to quickly create CRUD (Create, Read, Update, Delete) functionality for their models without writing additional code.

Security: Django includes various security features, such as protection against common web vulnerabilities like SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).

Internationalization and Localization: Django supports internationalization and localization features, making it easier to create applications that can be translated into multiple languages.

Rapid Development

Django was designed with the intention to make a framework which takes less time to build web application. The project implementation phase is a very time taken but Django creates it rapidly.

Secure

Django takes security seriously and helps developers to avoid many common security mistakes, such as SQL injection, cross-site scripting, cross-site request forgery etc. Its user authentication system provides a secure way to manage user accounts and passwords.

Scalable
Django is scalable in nature and has ability to quickly and flexibly switch from small to large scale application project.

Fully loaded
Django includes various helping task modules and libraries which can be used to handle common Web development tasks. Django takes care of user authentication, content administration, site maps, RSS feeds etc.

Versatile
Django is versatile in nature which allows it to build applications for different-different domains. Now a days, Companies are using Django to build various types of applications like: content management systems, social networks sites or scientific computing platforms etc.

Open Source
Django is an open source web application framework. It is publicly available without cost. It can be downloaded with source code from the public repository. Open source reduces the total cost of the application development.

Vast and Supported Community
Django is an one of the most popular web framework. It has widely supportive community and channels to share and connect.

Database Support: Django supports multiple databases including popular ones like PostgreSQL, MySQL, SQLite, and Oracle. It provides a consistent database API, allowing developers to work with different database backends seamlessly.

URL Routing and Views: Django's URL routing system maps URLs to views, which are Python functions or class-based views. Views receive requests, process them, and return responses. URL patterns can be defined using regular expressions or a simpler syntax for common cases.

Middleware: Django middleware is a component that sits between the web server and the view, allowing you to process requests and responses globally. Middleware can perform operations such as authentication, request/response modification, or handling exceptions.

Template Language: Django's template language allows developers to dynamically generate HTML, XML, or other output formats. It supports template inheritance, template tags for logic and rendering, template filters for data manipulation, and template contexts for passing variables to templates.

Forms: Django provides a powerful form handling system that simplifies form creation, validation, and rendering. It includes built-in form field types, form validation, error handling, and support for handling file uploads.

Security: Django has built-in security features to help developers protect their applications. This includes protection against common web vulnerabilities such as SQL injection, XSS, and CSRF attacks. Django also encourages best practices for secure password handling and user authentication.

Caching: Django provides a caching framework that allows developers to cache dynamic content and improve the performance of their applications. It supports various cache backends such as in-memory caching, database caching, and caching using external services like Redis.

Testing: Django has a comprehensive testing framework that makes it easy to write unit tests and integration tests for your applications. It provides tools for testing views, forms, models, and other components of your Django project.

Deployment: Django applications can be deployed on various platforms and web servers. It provides built-in support for running applications with popular web servers like Apache and Nginx. Additionally, there are deployment tools and platforms specifically designed for Django, such as Heroku, AWS Elastic Beanstalk, and Google App Engine.

Community and Documentation: Django has a vibrant and supportive community. It offers extensive documentation, tutorials, and examples to help developers learn and use the framework effectively. The Django community also contributes to a wide range of third-party packages and extensions that further enhance Django's functionality.

HTML :

HTML is an acronym which stands for Hyper Text Markup Language which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

Hyper Text: Hyper Text simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. Hyper Text is a way to link two or more web pages (HTML documents) with each other.

Markup language: A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page: A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. With the help of HTML only, we can create static web pages

How HTML works

HTML is a text file containing specific syntax, file and naming conventions that show the computer and the web server that it is in HTML and should be read as such. By applying these HTML conventions to a text file in virtually any text editor, a user can write and design a basic webpage, and then upload it to the internet.

The most basic of HTML conventions is the inclusion of a document type declaration at the beginning of the text file. This always comes first in the document, because it is the piece that affirmatively informs a computer that this is an HTML file. The document header typically looks like this: <!DOCTYPE html>. It should always be written that way, without any content inside it or breaking it up. Any content that comes before this declaration will not be recognized as HTML by a computer.

Doctypes are not just used for HTML, they can apply to the creation of any document that uses SGML (Standard Generalized Markup Language). SGML is a standard for specifying a specific markup language being used. HTML is one of several markup languages that SGML and doctype declarations apply to.

HTML (Hypertext Markup Language) is the standard markup language used for creating the structure and content of web pages. It is the backbone of the World Wide Web and is understood by web browsers to render web pages.

Here are some key points about HTML:

Markup Language: HTML is a markup language, which means it uses a set of markup tags or elements to describe the structure and content of a web page. These tags are placed within angle brackets (< >) and define various elements such as headings, paragraphs, images, links, tables, forms, and more.

Structure and Semantics: HTML is responsible for defining the structure of a web page. It allows developers to organize content into logical sections, such as headers, footers, navigation menus, main content areas, and sidebars. Additionally, HTML provides semantic elements that give meaning and context to different parts of the page, making it more accessible to users and search engines.

Text and Multimedia Content: HTML allows you to include various types of content on a web page. You can add text, headings, lists, and paragraphs using appropriate HTML tags. Furthermore, HTML supports the embedding of multimedia content like images, audio, video, and interactive elements.

Hyperlinks: HTML enables the creation of hyperlinks, also known as anchors, which allow users to navigate between different web pages or sections within a page. Links are defined using the <a> tag and can be styled, target different windows or tabs, and link to internal or external URLs.

Forms: HTML provides form elements that enable the collection and submission of user data. You can create input fields, checkboxes, radio buttons, dropdown lists, and more. When a user submits a form, the data can be sent to a server for processing using various methods like GET or POST.

Style and Presentation: While HTML defines the structure and content of a web page, it does not handle visual styling. CSS (Cascading Style Sheets) is typically used alongside HTML to control the presentation and layout of web pages. CSS allows developers to specify colors, fonts, spacing, backgrounds, and other visual aspects of the page.

Browser Interpretation: Web browsers interpret HTML and render it as a visual web page. Browsers read the HTML code and apply the necessary formatting and styles to display the content to users. Different browsers may handle HTML and CSS slightly differently, so developers strive to write HTML that is cross-browser compatible.

Version History: HTML has evolved over time with different versions. The current widely used version is HTML5, which was released in 2014. HTML5 introduced several new features, including semantic elements, multimedia support, form enhancements, canvas for drawing graphics, and improved APIs for web applications.

Semantic Elements: HTML5 introduced semantic elements that provide meaning to the structure of a web page. These elements, such as <header>, <nav>, <main>, <article>, <section>, <footer>, and others, help define the purpose and role of different sections of the page. Semantic elements improve accessibility and allow search engines to better understand and index the content.

Attributes: HTML elements can have attributes that provide additional information or modify their behavior. Attributes are defined within the opening tag of an element and provide instructions to the browser or specify element properties. Examples of attributes include id, class, src, href, alt, disabled, required, and many more.

Document Structure: An HTML document has a specific structure. It begins with a <!DOCTYPE> declaration, followed by the <html> element as the root of the document. The <head> section contains meta-information, the document title, and links to external resources like stylesheets and scripts. The actual content of the page is placed within the <body> element.

Headings and Text: HTML provides six levels of headings, from <h1> (the highest) to <h6> (the lowest). These tags define the hierarchical structure of the document. Other text-related tags include <p> for paragraphs, and for emphasizing text, <ul> and <ol> for unordered and ordered lists, <blockquote> for block quotes, and more.

Links and Anchors: HTML allows you to create hyperlinks using the <a> (anchor) tag. The href attribute specifies the target URL of the link. Additionally, you can use the target attribute to control how the linked page opens (in the same window, a new window, or a specific frame). You can also use anchors (<a> tags with a name attribute) to link to specific sections within a page.

Images and Multimedia: HTML provides the <img> tag for embedding images in a web page. The src attribute specifies the image source URL, and the alt attribute provides alternative text for accessibility purposes. HTML5 also introduced support for multimedia elements like <audio> and <video>, allowing you to embed audio and video content directly in the page.

Tables: HTML allows you to create tables using the <table> element. Tables consist of rows (<tr>) and cells (<td>) organized in a grid-like structure. You can use additional elements like <th> for table headers, <caption> for table captions, and <thead>, <tbody>, and <tfoot> to group table sections.

Forms and Input Fields: HTML provides a range of form elements for collecting user input. You can create text fields (<input type="text">), checkboxes (<input type="checkbox">), radio buttons (<input type="radio">), dropdown lists (<select>), text areas (<textarea>), buttons (<button>), and more. Form elements can be grouped within the <form> tag and submitted to a server for processing.

Validation and Accessibility: HTML5 introduced native form validation, allowing you to specify required fields, validate email addresses, enforce numeric input, and more

HTML is a fundamental skill for web developers. It provides the structure and content of web pages, allowing developers to create interactive and accessible websites. HTML is constantly evolving, with new features and versions being introduced over time to enhance its capabilities and support modern web development practices.

CSS :

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs,variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

CSS (Cascading Style Sheets) is a style sheet language used to describe the presentation and visual formatting of HTML (and XML) documents. It allows developers to control the appearance and layout of web pages, including elements such as colors, fonts, spacing, borders, and more.

Here are some key points about CSS:

Separation of Concerns: CSS separates the presentation layer from the content layer of a web page. HTML defines the structure and content, while CSS is responsible for the styling and visual aspects. This separation makes it easier to maintain and update the design of a website.

Selector-Based Styling: CSS uses selectors to target HTML elements and apply specific styles to them. Selectors can be based on element types (e.g., <h1>), classes (e.g., .my-class), IDs (e.g., #my-id), attributes, and more. This allows developers to apply styles to specific elements or groups of elements.

Style Declarations: CSS styles are defined as declarations within curly braces {}. Each declaration consists of a property and a value, separated by a colon (:). For example, color: red; sets the text color to red. Multiple declarations are separated by semicolons (;).

Inheritance and Cascading: CSS styles can be inherited from parent elements to their children. This means that styles applied to a higher-level element can be automatically applied to nested elements unless overridden. CSS also follows a cascading order, where styles from different sources (e.g., external stylesheets, inline styles) can interact and override each other based on specificity and order.

Box Model: CSS uses the box model to describe how elements are structured and sized. Each element is represented as a rectangular box, consisting of content, padding, border, and margin. The box model provides control over spacing and positioning of elements on the page.

Selectors and Specificity: CSS selectors have different levels of specificity, which determine which styles will be applied to an element if there are conflicting styles. Specificity is calculated based on the type of selector, IDs, classes, and inline styles. Understanding specificity is important to ensure that desired styles are applied correctly.

Media Queries: CSS supports media queries that allow developers to apply different styles based on various factors like screen size, device orientation, and resolution. Media queries enable responsive web design by adjusting the layout and appearance of a page based on the device or viewport.

Vendor Prefixes: CSS vendor prefixes are used to apply experimental or non-standard CSS features in different web browsers. By adding a vendor prefix (e.g., -webkit-, -moz-, -ms-, -o-), developers can target specific browser implementations of CSS properties.

CSS Preprocessors: CSS preprocessors like Sass, Less, and Stylus extend the functionality of CSS by introducing variables, nesting, functions, mixins, and more. Preprocessors help streamline CSS development and allow for more maintainable and reusable stylesheets.

CSS Frameworks: CSS frameworks, such as Bootstrap, Foundation, and Bulma, provide pre-designed CSS styles and components that can be used to quickly build and style web pages. These frameworks offer a collection of reusable CSS classes and JavaScript components, reducing the need for custom styling.

CSS Selectors: CSS provides a wide range of selectors that allow you to target specific elements or groups of elements for styling. Some common selectors include element selectors (e.g., h1, p, div), class selectors (e.g., .my-class), ID selectors (e.g., #my-id), attribute selectors (e.g., [type="text"]), pseudo-classes (e.g., :hover, :nth-child()), and pseudo-elements (e.g., ::before, ::after). Selectors can also be combined and nested to create more specific targeting.

Box Model: The CSS box model describes how elements are structured and sized. Each element is represented as a rectangular box that consists of content, padding, border, and margin. The content area holds the actual content of the element, while padding provides space between the content and the border. The border surrounds the padding and content, and the margin creates space outside the border. Understanding the box model is important for controlling spacing and sizing of elements.

Layout and Positioning: CSS offers various techniques for controlling the layout and positioning of elements on a web page. This includes techniques like using float for element alignment, display property for controlling the layout mode (e.g., block, inline, flex, grid), position property for absolute or relative positioning, box-sizing property for controlling how the sizing is calculated, and more. CSS also provides properties like width, height, margin, padding, and position to fine-tune the positioning and dimensions of elements.

Typography: CSS provides extensive control over typography, allowing you to specify fonts, sizes, styles, and other text-related properties. You can use properties like font-family, font-size, font-weight, line-height, text-align, text-decoration, text-transform, and more to customize the appearance of text. CSS also supports web fonts, which allow you to use custom fonts that may not be available on all devices.

Colors and Backgrounds: CSS enables you to define colors and backgrounds for elements. You can use color names, hexadecimal values, RGB or HSL values, and even CSS color functions to specify colors. CSS also provides properties like background-color, background-image, background-size, background-position, and background-repeat for customizing element backgrounds. Additionally, CSS supports gradients and other advanced background effects.

Transitions and Animations: CSS allows you to create smooth transitions and animations to enhance user interactions and visual effects. Using properties like transition, transform, and animation, you can define transitions between states and animate various properties like position, size, rotation, and opacity. CSS animations can be triggered by different events, such as hovering over an element or applying a CSS class.

Media Queries: CSS media queries enable responsive web design by applying different styles based on the characteristics of the device or viewport. Media queries allow you to specify different CSS rules for different screen sizes, resolutions, device orientations, and more. This helps ensure that web pages adapt and display optimally across various devices, from mobile phones to large desktop screens.

Flexbox and Grid: CSS provides powerful layout systems, such as Flexbox and Grid, that simplify the creation of complex and responsive layouts. Flexbox allows for flexible and dynamic layout of elements along a single axis (either horizontally or vertically) and enables features like alignment, spacing, and wrapping. Grid provides a two-dimensional layout system, allowing you to create grid-based layouts with precise control over row and column

CSS plays a crucial role in web development, allowing developers to transform the plain structure of HTML into visually appealing and responsive web pages. With CSS, the appearance and layout of a website can be customized to create a consistent and engaging user experience across different devices and browsers.

Why use CSS

These are the three major benefits of CSS:

1) Solves a big problem

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and color information are added on every single page, it will be become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

2) Saves a lot of time

CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

Advantages of CSS

CSS saves time
You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

Pages load faster
If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.

Easy maintenance
To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

Superior styles to HTML
CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

Multiple Device Compatibility
Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

Global web standards
Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

Who Creates and Maintains CSS?

CSS is created and maintained through a group of people within the W3C called the CSS Working Group. The CSS Working Group creates documents called specifications. When a specification has been discussed and officially ratified by the W3C members, it becomes a recommendation.

These ratified specifications are called recommendations because the W3C has no control over the actual implementation of the language. Independent companies and organizations create that software.

CSS Versions

Cascading Style Sheets level 1 (CSS1) came out of W3C as a recommendation in December 1996. This version describes the CSS language as well as a simple visual formatting model for all the HTML tags.

CSS2 became a W3C recommendation in May 1998 and builds on CSS1. This version adds support for media-specific style sheets e.g. printers and aural devices, downloadable fonts, element positioning and tables

CSS has gone through several versions since its inception. Here are the major versions of CSS:

CSS1: CSS1 was the first official version of CSS and was published as a recommendation by the World Wide Web Consortium (W3C) in 1996. It introduced basic styling capabilities such as font styles, colors, margins, and text alignment.

CSS2: CSS2 was published as a recommendation in 1998. It added new features like support for absolute positioning, z-index for layering elements, table layout control, generated content using pseudo-elements, and more advanced selectors.

CSS2.1: CSS2.1 was an updated version of CSS2, published as a W3C recommendation in 2011. It clarified and fixed various issues and inconsistencies in CSS2, improving browser compatibility. CSS2.1 is widely supported by modern browsers and forms the basis for most CSS implementations.

CSS3: CSS3 is not a single version but rather a collection of individual modules that introduce new features and enhancements to CSS. CSS3 modules were developed separately and have different levels of implementation and browser support. Some of the notable CSS3 modules include Selectors, Backgrounds and Borders, Box Model, Text Effects, Transitions, Animations, Flexbox, Grid Layout, and more.

CSS4: Unlike the previous versions, there is no official CSS4 specification. Instead, the new features and enhancements that were planned for CSS4 have been incorporated into the ongoing development of CSS3 modules. This approach allows for more modular and incremental updates to CSS.

CSS Level 4 and Beyond: The future of CSS is expected to continue with the development of new modules and enhancements under the CSS Level 4 umbrella. These modules will introduce additional features and improvements to CSS, addressing new design challenges and evolving web standards.

It's important to note that while CSS3 and its various modules are in use and widely supported, the term "CSS3" is often used to refer collectively to the collection of modules that make up modern CSS. Each module may have its own level or version within the CSS3 specification.

It's always recommended to check the browser compatibility and implementation status of specific CSS features when using them in your projects, as support may vary across different browsers and versions.

BOOTSTRAP :

Bootstrap is a free, open source front-end development framework for the creation of websites and web apps. Designed to enable responsive development of mobile-first websites, Bootstrap provides a collection of syntax for template designs.

As a framework, Bootstrap includes the basics for responsive web development, so developers only need to insert the code into a pre-defined grid system. The Bootstrap framework is built on Hypertext Markup Language (HTML), cascading style sheets (CSS) and JavaScript. Web developers using Bootstrap can build websites much faster without spending time worrying about basic commands and functions.

Bootstrap is the most popular HTML, CSS and JavaScript framework for developing a responsive and mobile friendly website.

- It is absolutely free to download and use.
- It is a front-end framework used for easier and faster web development.
- It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many others.
- It can also use JavaScript plug-ins.
- It facilitates you to create responsive designs.

Bootstrap is a popular front-end CSS framework that provides a collection of pre-designed and responsive CSS styles, components, and JavaScript plugins. It was originally created by Twitter and is now an open-source project maintained by a community of developers.

Here are some key points about Bootstrap:

Responsive Design: Bootstrap is built with a mobile-first approach, ensuring that websites and web applications look and function well on different devices and screen sizes. It provides a responsive grid system that allows you to create flexible and adaptive layouts.

CSS Components: Bootstrap includes a wide range of reusable CSS components, such as buttons, forms, navigation menus, cards, modals, dropdowns, alerts, badges, and more. These components are designed to be visually appealing, consistent, and easily customizable.

Layout and Grid System: Bootstrap's grid system is based on a 12-column layout, providing a flexible way to create responsive designs. It allows you to divide the page into rows and columns, enabling you to arrange and align content effectively.

CSS Utilities: Bootstrap offers a set of utility classes that provide quick and handy CSS styling options. These classes help with common tasks like adjusting margins and padding, controlling text alignment, displaying and hiding elements based on screen sizes, and more.

Customization: Bootstrap provides customization options to tailor its appearance and functionality to suit your project's needs. You can customize the color scheme, typography, spacing, and other aspects by modifying variables or using predefined themes.

Browser Compatibility: Bootstrap is designed to be compatible with modern web browsers, including Chrome, Firefox, Safari, Edge, and Internet Explorer (IE11 and above). However, note that certain advanced features may have limited support in older browsers.

Integration with JavaScript: Bootstrap includes a set of JavaScript plugins that enhance the functionality of components and add interactive features. These plugins handle tasks like dropdown menus, modals, carousels, tooltips, form validation, and more. You can use these plugins by including the Bootstrap JavaScript files or by using the BootstrapCDN (Content Delivery Network).

Documentation and Community: Bootstrap provides comprehensive documentation that guides developers on how to use its features and components effectively. The documentation includes code examples, explanations, and guidelines for customization. Additionally, Bootstrap has a large community of developers who contribute to its development, provide support, and share resources.

Grid System: Bootstrap's grid system is based on a 12-column layout, which allows for flexible and responsive page layouts. You can easily create responsive designs by dividing the page into rows and columns. The grid system supports responsive breakpoints, enabling you to adjust the layout based on different screen sizes.

CSS Components: Bootstrap provides a rich collection of pre-designed CSS components that you can easily incorporate into your projects. These components include buttons, forms, navigation bars, dropdowns, alerts, progress bars, badges, pagination, and more. These components are designed to be visually appealing, consistent, and easy to use.

Responsive Utilities: Bootstrap offers a set of utility classes that allow you to control the visibility and behavior of elements based on different screen sizes. With these utilities, you can show or hide elements, change their positioning, and adjust their spacing on different devices.

CSS Flexbox and Grid Layout: Bootstrap leverages CSS Flexbox and Grid Layout to provide more advanced layout options. Flexbox allows for flexible and dynamic layouts along a single axis, making it easier to align and distribute elements. Grid Layout provides a powerful two-dimensional grid system for creating complex layouts with precise control over rows and columns.

Typography and Styling: Bootstrap provides default styling for typography, including headings, paragraphs, lists, and inline text elements. It also offers a set of CSS classes for text alignment, text color, text formatting, and other typographic elements. Additionally, Bootstrap includes a variety of

contextual classes that allow you to apply different styles based on contextual states, such as success, warning, info, and danger.

Customization: Bootstrap provides customization options to tailor the framework to your project's needs. You can use Bootstrap's customization tool or modify the SASS variables to change aspects like colors, typography, spacing, breakpoints, and more. This allows you to create a unique look and feel for your website or application.

JavaScript Plugins: Bootstrap includes a set of JavaScript plugins that enhance the functionality and interactivity of components. These plugins handle tasks like modals, dropdowns, carousels, tooltips, scrollspy, form validation, and more. You can easily add these features to your project by including the Bootstrap JavaScript files or using the BootstrapCDN.

Browser Compatibility: Bootstrap is designed to be compatible with modern web browsers, including Chrome, Firefox, Safari, Edge, and Internet Explorer 11 (and above). However, some advanced features and layout options may have limited support in older browsers.

Community and Resources: Bootstrap has a large and active community of developers who contribute to its development, provide support, and share resources. There are numerous tutorials, documentation, templates, and third-party extensions available that can help you learn and extend the capabilities of Bootstrap.

Bootstrap is widely used in web development for its ease of use, responsiveness, and extensive set of components and utilities. It provides a solid foundation for building visually appealing and mobile-friendly websites or web applications, allowing developers to save time and effort by leveraging its pre-designed elements and responsive features.

Bootstrap is widely adopted in web development due to its ease of use, responsive design capabilities, extensive component library, and strong community support. It allows developers to quickly build professional-looking and mobile-friendly websites or web applications without starting from scratch.

Bootstrap has gone through several major versions since its initial release. Here are the key versions of Bootstrap:

Bootstrap 2: Released in January 2012, Bootstrap 2 introduced significant improvements over its predecessor. It included a responsive grid system, CSS components, JavaScript plugins, and a variety of pre-styled elements. Bootstrap 2 was a game-changer for front-end development and gained widespread popularity.

Bootstrap 3: Released in August 2013, Bootstrap 3 brought major updates and improvements. It emphasized mobile-first design, introduced a flat design aesthetic, and introduced a significant

rewrite of the framework's CSS and JavaScript code. Bootstrap 3 included a redesigned grid system, revamped components, improved documentation, and numerous new features.

Bootstrap 4: Released in January 2018 after a long development period, Bootstrap 4 was a major rewrite and refinement of the framework. It embraced flexbox as the primary layout system and introduced a more modular and scalable structure. Bootstrap 4 also included a new grid system, updated components, enhanced customization options, and dropped support for older versions of Internet Explorer.

Bootstrap 5: Released in May 2021, Bootstrap 5 introduced several notable changes and improvements. One of the most significant changes was the removal of jQuery as a dependency, making Bootstrap more lightweight. It also introduced a new CSS utility API, improved grid system with flexbox enhancements, updated components, and streamlined styles. Bootstrap 5 aimed to simplify the framework and provide more flexibility for customization.

It's important to note that each major version of Bootstrap introduced changes to the CSS structure, JavaScript components, and overall features. Upgrading from one major version to another may require adjustments in code and may introduce compatibility issues with older projects.

Bootstrap has a strong community and is continuously evolving. New releases and updates may bring new features, improvements, bug fixes, and performance optimizations. It's advisable to refer to the official Bootstrap documentation and release notes for specific details and changes in each version.

Why use Bootstrap

Following are the main advantage of Bootstrap:

- It is very easy to use. Anybody having basic knowledge of HTML and CSS can use Bootstrap.
- It facilitates users to develop a responsive website.
- It is compatible on most of browsers like Chrome, Firefox, Internet Explorer, Safari and Opera etc.

What is a responsive website

A website is called responsive website which can automatically adjust itself to look good on all devices, from smart phones to desktops etc.

Scaffolding:
Bootstrap provides a basic structure with Grid System, link styles, and background.

CSS: Bootstrap comes with the feature of global CSS settings, fundamental HTML elements style and an advanced grid system.

Components: Bootstrap contains a lot of reusable components built to provide iconography, dropdowns, navigation, alerts, pop-overs, and much more.

JavaScript Plugins: Bootstrap also contains a lot of custom jQuery plugins. You can easily include them all, or one by one.

Customize: components are customizable and you can customize Bootstrap's components, LESS variables, and jQuery plugins to get your own style.

What is Bootstrap 4

Bootstrap 4 is the newest version of Bootstrap. Bootstrap can be defined as a free and open-source framework that can be used to create responsive, mobile-first, front-end web pages.

There are various key usages of Bootstrap 4:

Supported by various Browsers:
It can be supported by every browser.

Simple to start and implement:
It is very easy to start and implement when the user has a fair amount of knowledge about HTML and CSS. In addition to that, the documentation is readily available on the official site.

Responsive design and looks:
The web pages that are created by using the Bootstrap framework are responsive and it can adapt to any screen size like mobile, desktop, etc.

Easily Customized:
It also provides some built-in components and functionalities that can be used for the purpose of easily customizing the web pages.

What are the disadvantages of bootstrap 4:

- In many cases, Bootstrap cannot be considered very practical for businesses that need a big amount of investment.
- Moreover, Bootstrap 4 can take a lot of time to create a website. Therefore, it is not a very bright idea to use bootstrap 4 when there is no investment.

- A person using Bootstrap 4 is not likely to earn any money even after investment.
- And this is one of the reasons why a user can very easily end up in a lot of debt.

Some important changes in Bootstrap 5:

1. Updated official documentation: The docs homepage of v4.5.0 has an updated look and there are some improvements in the rest of the official docs too. Bootstrap docs have some nice padding around the corners to make it more readable and highlight its content.
2. jQuery is removed completely: Earlier, bootstrap components like modals, tooltips, popovers etc, were dependent on jQuery and popper.js. After the release of the alpha 5 version, Bootstrap no longer depends on jQuery. While jQuery has been used in Bootstrap for more than 8 years, many developers had this issue that jQuery brought uneven access to complex JavaScript behaviors to them. The whole process made possible by their JavaScript maintainer https://github.com/johann-s

3. Switch to Vanilla JavaScript: JavaScript is the programming language of the web. Most of the modern websites are powered by JavaScript and all modern web browsers on phones, tablets, consoles, desktops include JavaScript Interpreters. One of the major reasons of dropping jQuery was to redefine the framework completely on modern JavaScript standards.

4. Dropped support of Internet Explorer 10 and 11: Internet Explorer was released in 1995 by Microsoft. As of today, Internet Explorer is no longer relevant with popular browsers like Chrome and Microsoft Edge which are built on chromium. When bootstrap decided to refine their framework purely on Vanilla JavaScript, they dropped the support of IE which is not compatible with modern JavaScript standards.

5. Responsive Fonts: Bootstrap has introduced responsive font sizes which will automatically resize the typography element according to the viewport size through the RFS Engine. RFS engine was developed to resize font sizes. It provides the facility to resize the CSS properties like “margin”, “padding”, “border-radius”, “box-shadow” by units.

6. Removed Cards: The new grid system of Bootstrap provides more responsive control of the layout. Hence, they removed unnecessary extra classes which can be designed by the new grid layout. The older versions of Bootstrap still support card system.

7. Updated Forms: Bootstrap 5 has introduced a section completely dedicated to Bootstrap forms with improved documentation and components. It now has a single set of form controls and redesign existing elements instead of generating new elements through pseudo-elements. Bootstrap forms give a better look now.

MYSQL :

MySQL is an open-source relational database management system (RDBMS) that is widely used for managing and organizing large amounts of data. It is a popular choice for web applications and is commonly used in conjunction with scripting languages like PHP.

MySQL is designed to store, retrieve, and manage structured data, typically in a tabular format. It follows the SQL (Structured Query Language) standard, allowing users to define, manipulate, and query the data stored in the database.

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons −

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments

Some key features of MySQL include:

Scalability: MySQL can handle large amounts of data and can be scaled to support high-traffic websites and applications.

Performance: It is known for its speed and efficiency in handling database operations.

Security: MySQL provides various security features such as user account management, encryption, and access control to protect data from unauthorized access.

Replication and Clustering: MySQL supports replication, allowing data to be replicated across multiple database servers for improved performance and data redundancy. It also supports clustering to provide high availability and fault tolerance.

Cross-platform compatibility: MySQL is available for various operating systems, including Windows, Linux, macOS, and UNIX.

Community and support: MySQL has a large and active community of developers, and there are plenty of resources, documentation, and forums available for assistance.

Data Types: MySQL supports a wide range of data types to accommodate different types of data. These include numeric types (e.g., INT, DECIMAL), string types (e.g., VARCHAR, TEXT), date and time types (e.g., DATE, DATETIME), and more. You can choose appropriate data types based on the nature of the data you're storing.

Storage Engines: MySQL offers multiple storage engines, which are responsible for how data is stored and accessed. The most commonly used storage engine is InnoDB, known for its transactional support, reliability, and concurrency control. Other engines include MyISAM (known for its simplicity and speed), MEMORY (for storing data in memory), and more.

Indexing: MySQL allows you to create indexes on tables to improve query performance. Indexes provide a way to quickly locate and retrieve data based on specific columns. By defining indexes on frequently queried columns, you can significantly speed up data retrieval operations.

Transactions and ACID: MySQL supports transactions, which ensure the integrity and consistency of data. Transactions allow you to group multiple database operations into a single logical unit, ensuring that either all the operations succeed or none of them take effect. MySQL follows the ACID (Atomicity, Consistency, Isolation, Durability) principles to maintain data integrity.

Stored Procedures and Triggers: MySQL enables you to define stored procedures, which are pre-compiled database routines that can be called from within SQL statements. Stored procedures help encapsulate business logic and improve code reusability. Triggers, on the other hand, are actions that are automatically executed when a specified event occurs (e.g., when a row is inserted, updated, or deleted).

User Management: MySQL allows you to create multiple user accounts with different privileges and access levels. You can grant or revoke privileges such as SELECT, INSERT, UPDATE, DELETE, and more on specific databases or tables, ensuring proper security and access control.

High Availability: MySQL offers features like replication and clustering to ensure high availability of data. Replication allows you to create multiple copies of a database on different servers, keeping them synchronized to provide redundancy and load balancing. Clustering allows you to distribute database servers across multiple nodes for fault tolerance and improved performance.

Tools and Utilities: MySQL provides a range of command-line tools, graphical interfaces, and APIs (Application Programming Interfaces) to manage and interact with databases. Some popular tools include MySQL Workbench (a visual database design and administration tool), MySQL Shell (a command-line interface), and various language-specific connectors and drivers.

MySQL is widely used in various applications, ranging from small personal projects to large-scale enterprise systems. It is known for its reliability, performance, and ease of use, making it a popular choice among developers and organizations worldwide.

Feasibility Study

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development.

A feasibility study is a comprehensive evaluation of a proposed project that evaluates all factors critical to its success in order to assess its likelihood of success. Business success can be defined primarily in terms of ROI, which is the amount of profits that will be generated by the project.
In a feasibility study, a proposed plan or project is evaluated for its practicality. As part of a feasibility study, a project or venture is evaluated for its viability in order to determine whether it will be successful.

As the name implies, a feasibility analysis is used to determine the viability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells us whether a project is worth the investment—in some cases, a project may not be doable. There can be many reasons for this, including requiring too many resources, which not only prevents those resources from performing other tasks but also may cost more than an organization would earn back by taking on a project that isn't profitable.

A well-designed study should offer a historical background of the business or project, such as a description of the product or service, accounting statements, details of operations and management, marketing research and policies, financial data, legal requirements, and tax obligations. Generally, such studies precede technical development and project implementation.

To conduct a feasibility study for a canteen management system project, you would typically consider the following aspects:

1. Technical Feasibility:
    - Assess the technical requirements of the project, such as hardware, software, and infrastructure.
    - Determine if the necessary technology and resources are available or can be acquired within the project's constraints.
    - Evaluate the compatibility and integration with existing systems, if any.
    - Consider scalability and performance requirements to ensure the system can handle future growth.

2. Economic Feasibility:
    - Evaluate the project's cost, including development, implementation, and maintenance expenses.
    - Assess the potential return on investment (ROI) and determine if the benefits outweigh the costs.
    - Consider alternative solutions and compare their costs to determine the most cost-effective option.
    - Evaluate the long-term financial sustainability of the canteen management system.

3. Operational Feasibility:
    - Analyze the current canteen operations and identify pain points or areas for improvement.
    - Determine if the proposed system aligns with the goals and requirements of the canteen.
    - Assess the impact of the system on daily operations, including staff training and workflow changes.
    - Evaluate the system's usability and ease of adoption by end-users.

4. Legal and Compliance Feasibility:
    - Identify and understand the legal and regulatory requirements relevant to the canteen management system.
    - Ensure compliance with data protection and privacy laws.
    - Consider any specific industry regulations or standards that need to be met.
    - Evaluate the potential risks and liabilities associated with the system.

5. Schedule Feasibility:
    - Define the project timeline and evaluate if it can be completed within the desired timeframe.
    - Identify any potential bottlenecks or dependencies that may impact the project schedule.
    - Assess the availability of resources, both human and technical, needed to complete the project within the defined timeline.

6. Organizational Feasibility:
    - Assess the organization's readiness and willingness to adopt and support the new system.
    - Evaluate the impact of the system on the organization's structure, roles, and responsibilities.
    - Consider any cultural or change management challenges that may arise during implementation.
    - Determine the level of support from key stakeholders and their commitment to the project's success.

7. Security Feasibility:
    - Identify the security requirements of the canteen management system, such as user authentication, access control, and data protection.
    - Assess the system's vulnerability to potential threats and risks, including unauthorized access and data breaches.
    - Evaluate the measures and controls that need to be implemented to ensure the system's security.
    - Consider compliance with industry standards and best practices for securing sensitive data.

8. Environmental Feasibility:
    - Evaluate the environmental impact of the canteen management system, such as energy consumption and waste generation.
    - Consider the potential for implementing eco-friendly practices, such as paperless transactions or sustainable sourcing of ingredients.
    - Assess the system's compatibility with environmental regulations and initiatives.

9. Social Feasibility:
    - Analyze the social implications of the canteen management system on various stakeholders, including employees, customers, and the local community.
    - Consider the system's impact on employee satisfaction, customer experience, and overall social well-being.
    - Evaluate the system's accessibility and inclusivity to ensure it caters to diverse user groups.
    - Assess the potential for positive social outcomes, such as promoting healthy eating habits or supporting local food suppliers.

10. Risk Analysis:
    - Identify and assess potential risks and uncertainties associated with the project, such as technical challenges, resource constraints, or external factors.
    - Develop risk mitigation strategies to minimize the impact of identified risks.
    - Consider the potential consequences of project failure and the measures to mitigate those risks.
    - Evaluate the contingency plans and fallback options in case the project encounters unforeseen difficulties.

Software Development Life Cycle

The systems development life cycle (SDLC). Or software development process in systems engineering, information systems and software engineering, is a process of creating or altering information system, and the models and methodologies that people use to develop these systems. In software engineering, the SDLC concept underpins many kinds of software development methodologies. These methodologies from the framework for planning and controlling the creation of and information system: the software development process.

A System Development Life Cycle (SDLC) adheres to important phases that are essential for developers, such as planning, analysis, design, and implementation, and are explained in the section below. It include evaluation of present system, information gathering, feasibility study and request approval. A number of system development life cycle (SDLC) models have been created: waterfall, fountain, spiral, build and fix, rapid prototyping, incremental, and synchronize and stabilize. The oldest of these, and the best known, is the waterfall model: a sequence of stages in which the output of each stage becomes the input for the next. These stages can be characterized and divided up in different ways, including the following:

The software development life cycle (SDLC) outlines the stages and processes involved in developing a software project. Let's go through the typical phases of the SDLC for a cafeteria management system built with Python and Django:

Requirements Gathering: In this phase, you gather and document the requirements for the cafeteria management system. You interact with stakeholders, such as cafeteria managers and staff, to understand their needs and expectations. The requirements will include features like menu management, order processing, inventory management, user roles, etc.

Analysis and Design: In this phase, you analyze the requirements and create a design for the system. You identify the major components and modules of the system, define their relationships and interfaces, and create a data model. With Django, you would design the database schema using Django's Object-Relational Mapping (ORM) capabilities.

Development: In this phase, you start implementing the system. You write code in Python using the Django framework. Django provides a structured approach to building web applications, including features like URL routing, views, models, templates, and forms. You develop the core functionalities of the cafeteria management system, including user authentication, menu management, order processing, and inventory management.

Testing: In this phase, you thoroughly test the system to ensure it works as expected. You create test cases and perform different types of testing, such as unit testing, integration testing, and system testing. You check for functional correctness, handle edge cases, and verify that the system is robust

and secure. Django provides tools and frameworks like the Django testing framework and third-party libraries for writing and executing tests.

Deployment: Once the system passes all the tests, you deploy it to a production environment. You set up the necessary infrastructure, including servers, databases, and web servers, to host the Django application. You configure the deployment settings, handle static files, and ensure the system is accessible to users. You may use tools like Docker or cloud platforms like AWS, GCP, or Azure for deployment.

Maintenance: After deployment, you enter the maintenance phase. Here, you monitor the system's performance, handle any issues or bugs that arise, and make necessary updates and enhancements based on user feedback or changing requirements. You may also perform periodic updates to keep up with the latest versions of Python, Django, and other dependencies.

Throughout the SDLC, it's important to follow best practices, adhere to coding standards, and utilize version control systems like Git to track changes and collaborate with a development team effectively.

Note that the SDLC phases can overlap or be iterative, depending on the project's complexity and development approach (e.g., Agile methodologies).

DESIGN:

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. The term “design” is defined as “the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization”. It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient details to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used. The system design develops the architectual detail required to build a system or product. As in the case of any systematic approach, this software too has undergone the best possible design phase fine turing all efficiency, perofrmance and accuracy levels. The design phase is a transition from a user oridented document to a document to the programmers or database personnel. System design goes through two phase of development: Logical and Physical Design

The design of a cafeteria management system built with Python and Django will involve various components and modules that work together to achieve the desired functionalities. Here's a high-level overview of the design:

User Authentication and Authorization:
Django's built-in authentication system can be used to handle user registration, login, and logout.
User roles and permissions can be defined to control access levels for different functionalities (e.g., admin, cafeteria staff, customers).

Menu Management:
The menu management module allows cafeteria staff to create, update, and delete menu items.
A database table can be created to store menu item details, such as name, description, price, and availability status.

Order Processing:
Customers can place orders through a user-friendly interface.
The system captures the selected items, quantities, and customer details.
Orders can be stored in a database table and associated with the respective customer.

Inventory Management:
The inventory module keeps track of available ingredients or items in the cafeteria.
It deducts the quantities of ingredients/items used in orders and updates the inventory accordingly.
Notifications or alerts can be implemented when inventory items are running low.

Reporting and Analytics:
This module generates reports and analytics to provide insights into cafeteria performance.
Reports can include sales summaries, popular menu items, revenue analysis, etc.
Django's template system or third-party libraries can be used to generate visually appealing reports.

Integration with Payment Gateway:
To facilitate online payments, integration with a payment gateway API can be implemented.

This allows customers to make secure payments for their orders.

User Interface (UI) Design:

The UI design should be intuitive, responsive, and user-friendly. Django's template system, along with HTML, CSS, and JavaScript, can be used to create the frontend views and interfaces.

Database Design:

Django's ORM enables the creation of database models that represent the various entities in the system (e.g., users, menu items, orders, inventory).

Relationships between entities can be defined, such as one-to-many (e.g., an order can have multiple menu items).

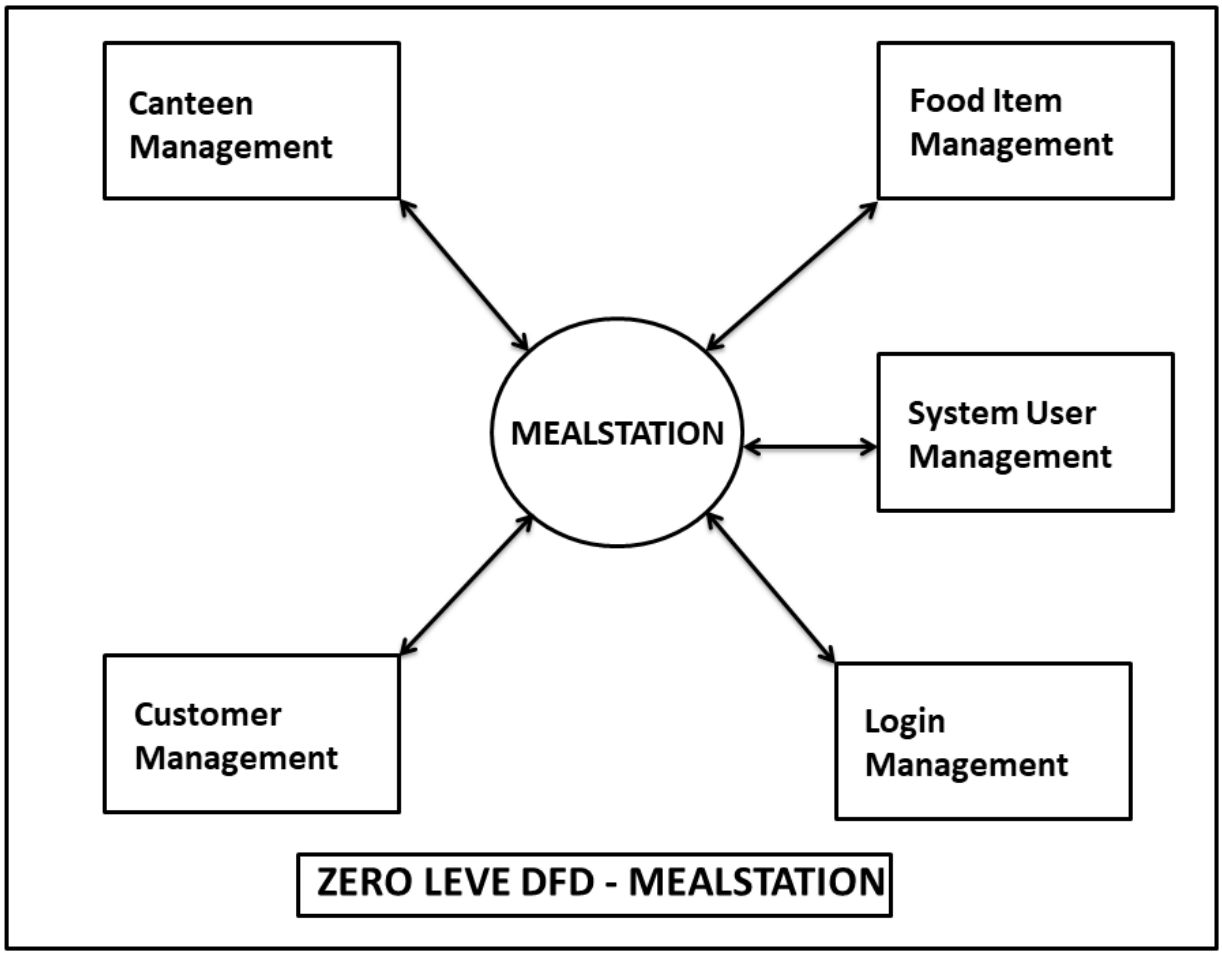
Security and Validation:

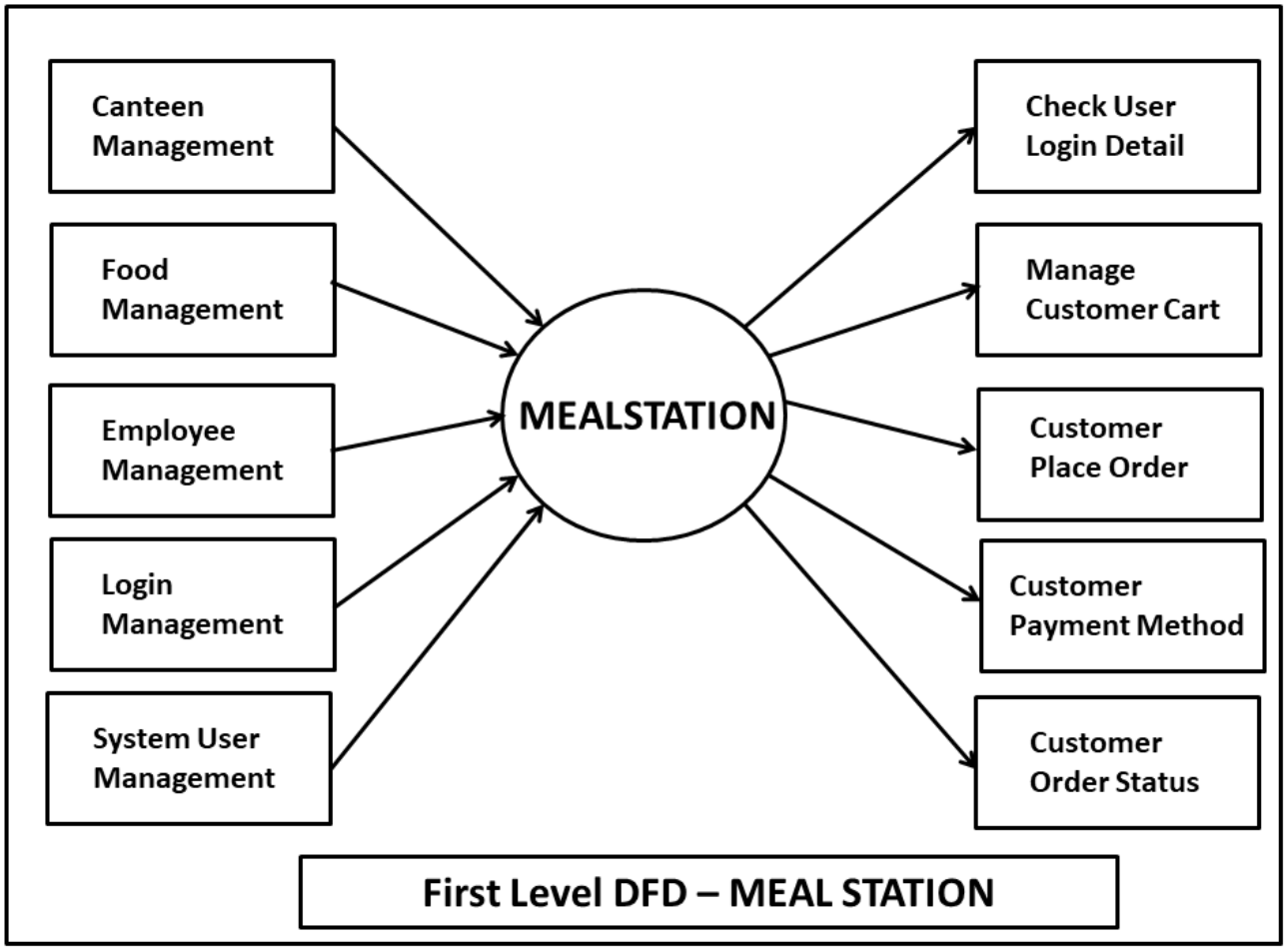
Proper security measures should be implemented, such as protecting against common web vulnerabilities like SQL injection and cross-site scripting (XSS).

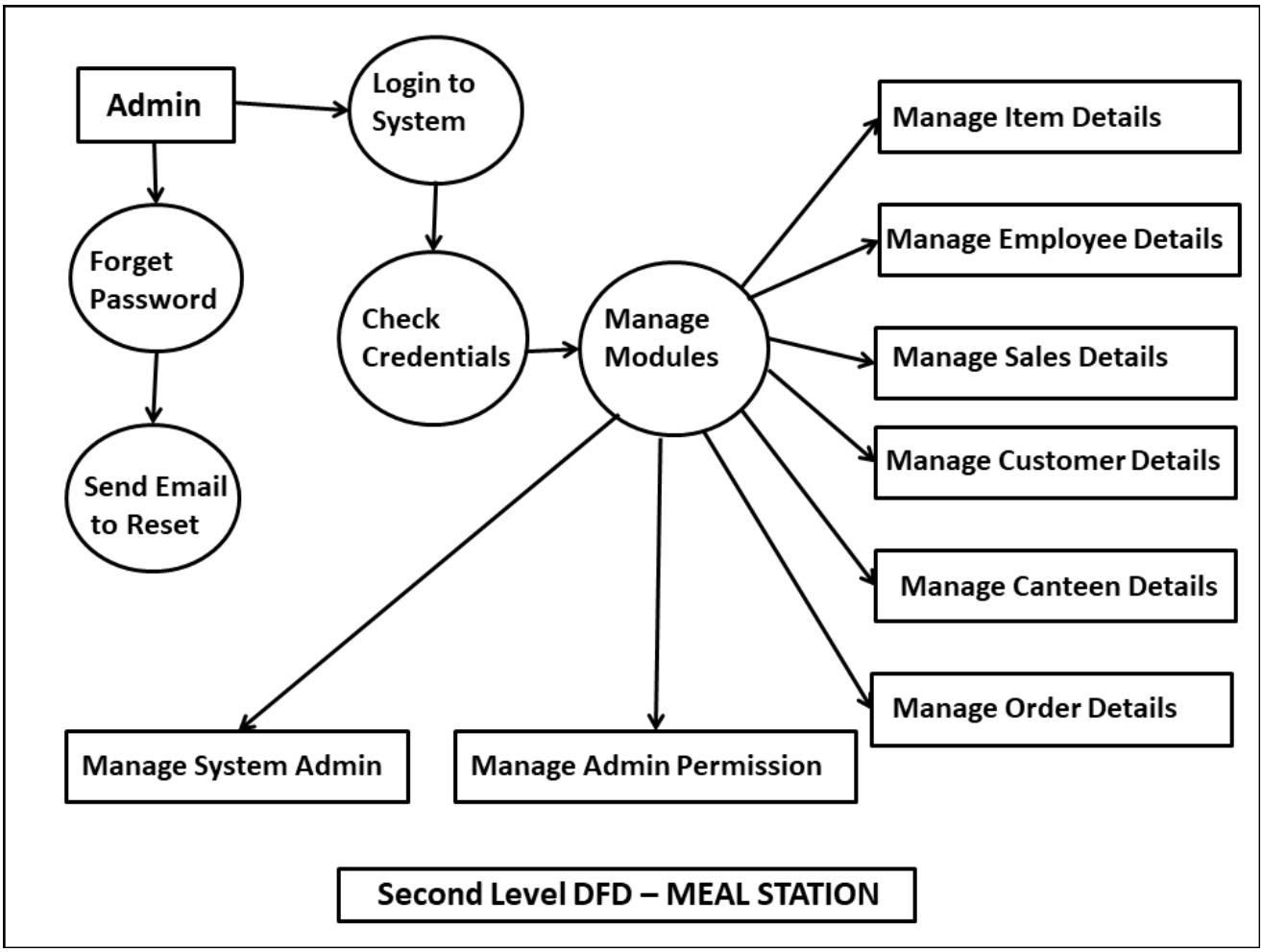
Input validation should be performed to ensure data integrity and prevent malicious inputs.

Remember, this is a high-level overview, and the actual design will require more detailed planning and consideration of specific requirements. The design can be further enhanced by leveraging Django's features, such as middleware, caching, and Django REST framework for building APIs if required.

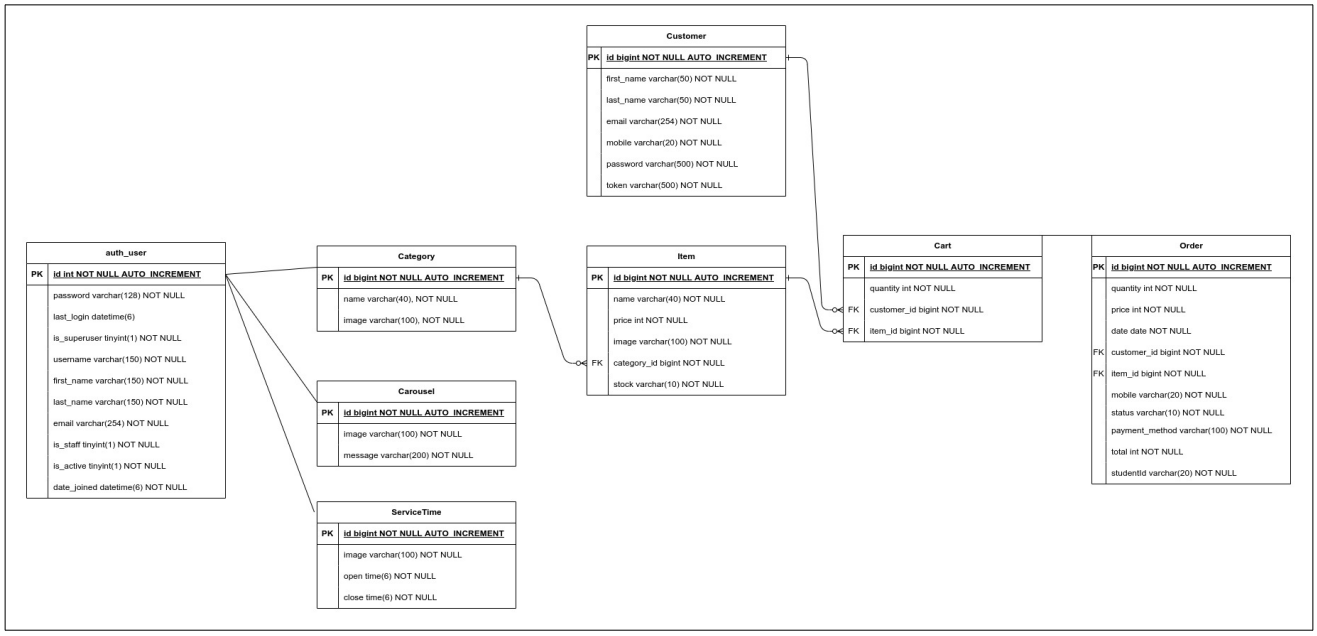
Data Flow Diagram

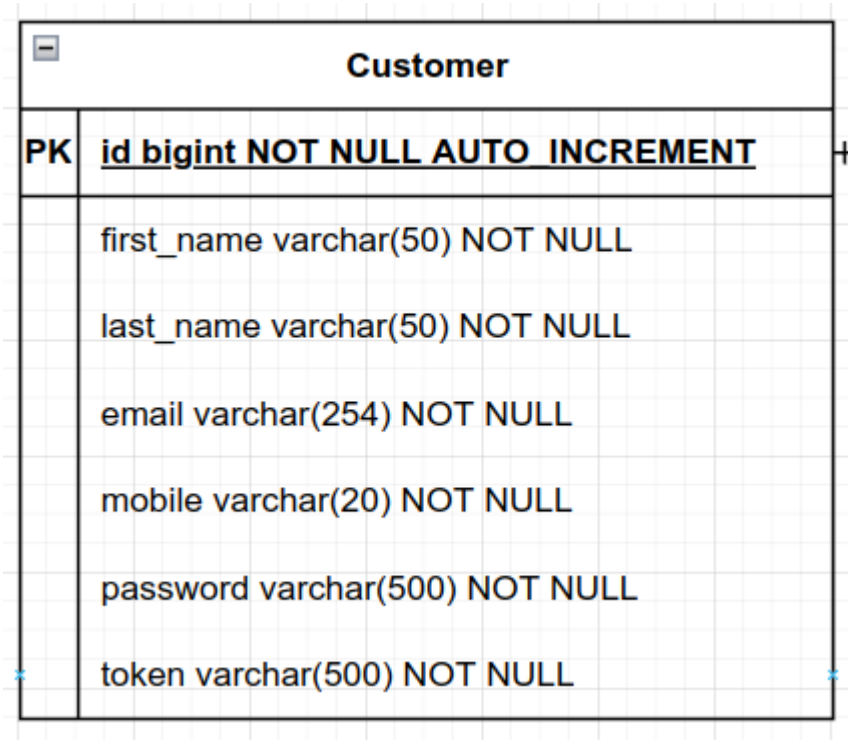


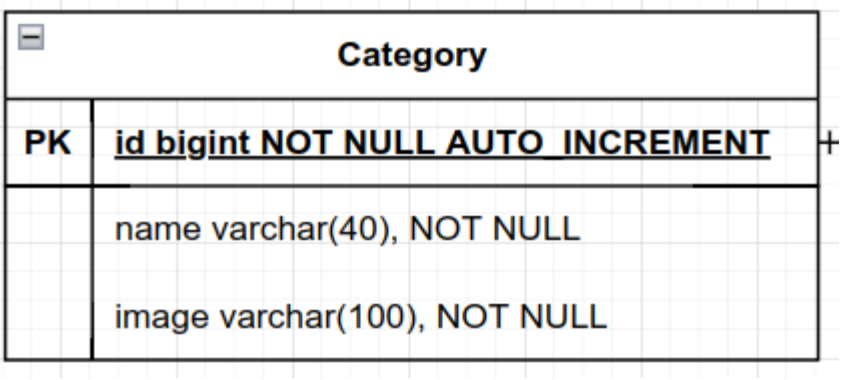

Canteen Management
Food Management
Employee Management
Login Management
System User Management
MEALSTATION
Check User Login Detail
Manage Customer Cart
Customer Place Order
Customer Payment Method
Customer Order Status
First Level DFD – MEAL STATION

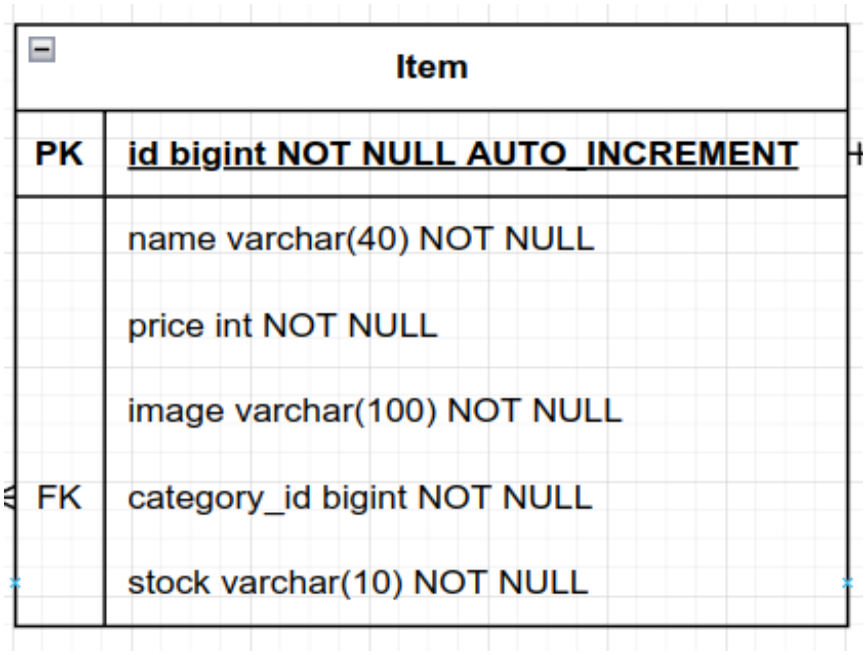

Admin
Login to System
Forget Password
Check Credentials
Manage Modules
Send Email to Reset
Manage Item Details
Manage Employee Details
Manage Sales Details
Manage Customer Details
Manage Canteen Details
Manage Order Details
Manage System Admin
Manage Admin Permission
Second Level DFD – MEAL STATION

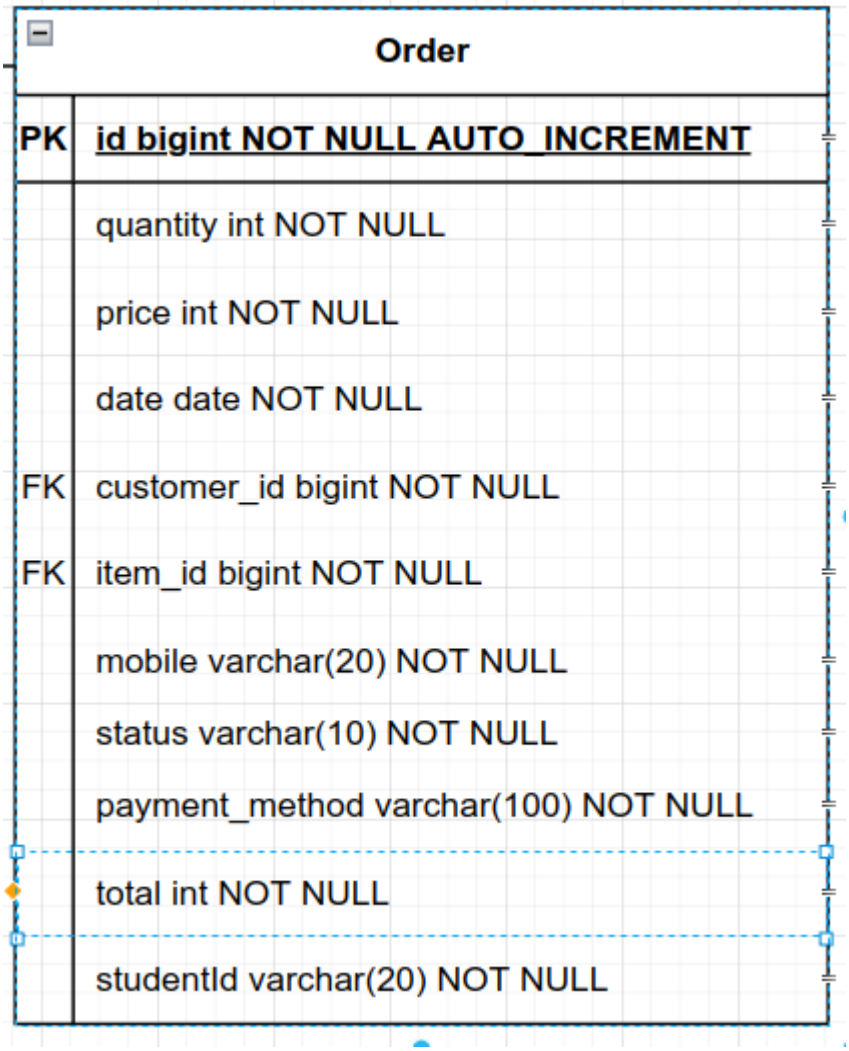
Data Base Schema Design

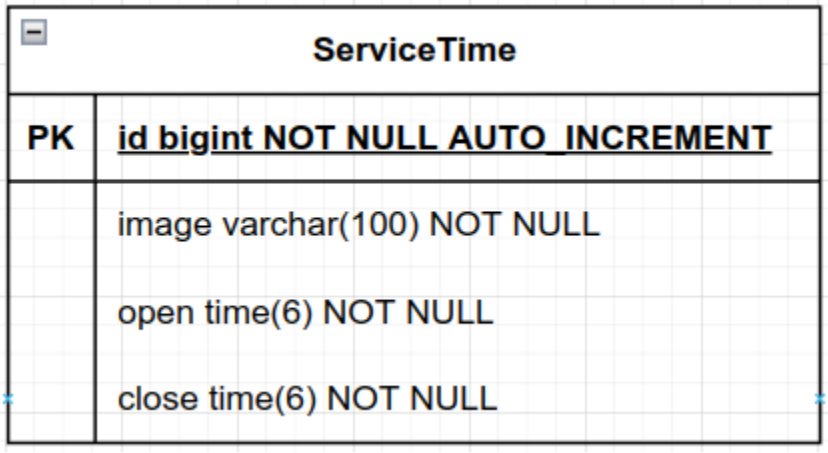


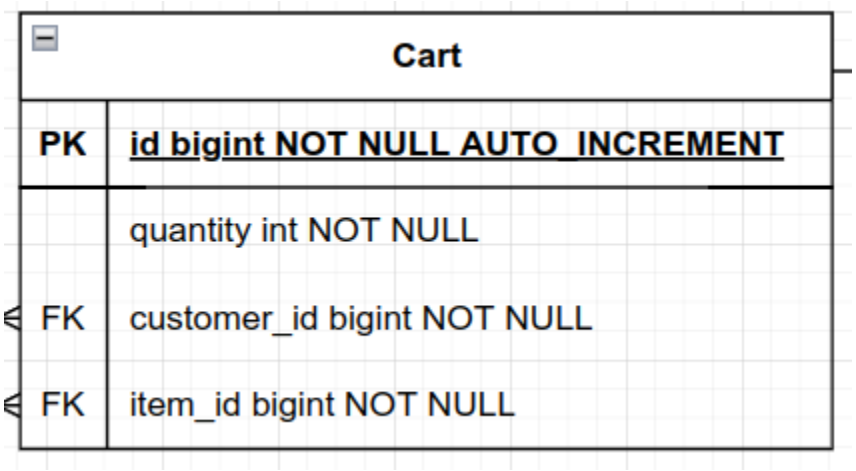


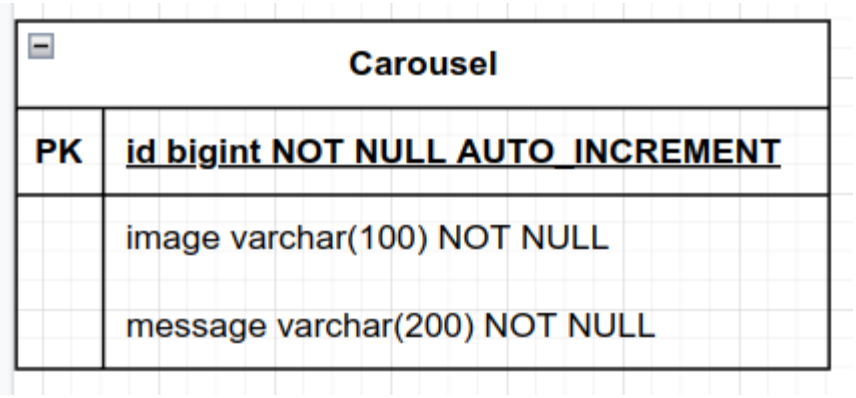

Category
PK
id bigint NOT NULL AUTO_INCREMENT
name varchar(40), NOT NULL
image varchar(100), NOT NULL


Item
PK
id bigint NOT NULL AUTO_INCREMENT
name varchar(40) NOT NULL
price int NOT NULL
image varchar(100) NOT NULL
FK
category_id bigint NOT NULL
stock varchar(10) NOT NULL


Order
PK
id bigint NOT NULL AUTO_INCREMENT
quantity int NOT NULL
price int NOT NULL
date date NOT NULL
FK
customer_id bigint NOT NULL
FK
item_id bigint NOT NULL
mobile varchar(20) NOT NULL
status varchar(10) NOT NULL
payment_method varchar(100) NOT NULL
total int NOT NULL
studentId varchar(20) NOT NULL

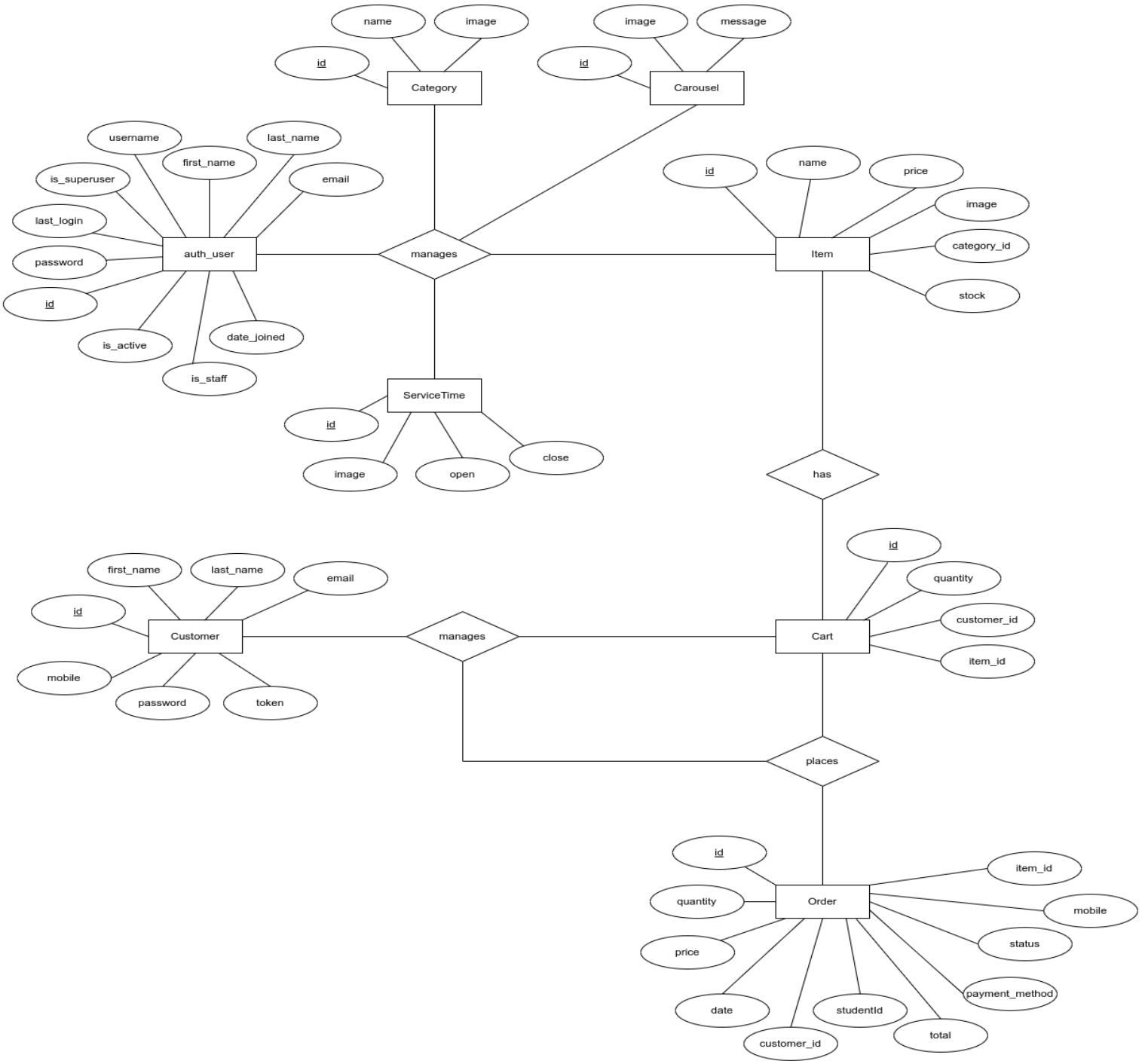

ServiceTime
PK
id bigint NOT NULL AUTO_INCREMENT
image varchar(100) NOT NULL
open time(6) NOT NULL
close time(6) NOT NULL


Cart
PK
id bigint NOT NULL AUTO_INCREMENT
quantity int NOT NULL
FK
customer_id bigint NOT NULL
FK
item_id bigint NOT NULL


Carousel
PK
id bigint NOT NULL AUTO_INCREMENT
image varchar(100) NOT NULL
message varchar(200) NOT NULL

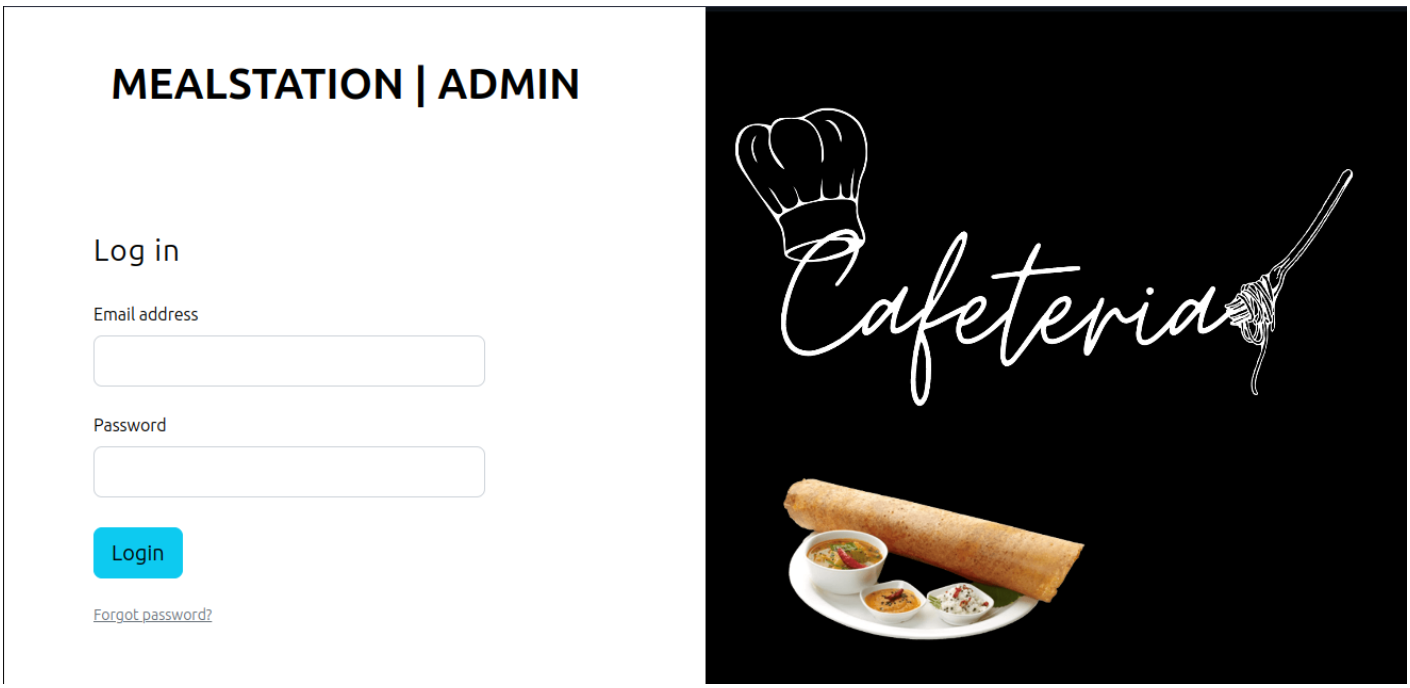
auth_user	
PK	**id int NOT NULL AUTO_INCREMENT**
	password varchar(128) NOT NULL
	last_login datetime(6)
	is_superuser tinyint(1) NOT NULL
	username varchar(150) NOT NULL
	first_name varchar(150) NOT NULL
	last_name varchar(150) NOT NULL
	email varchar(254) NOT NULL
	is_staff tinyint(1) NOT NULL
	is_active tinyint(1) NOT NULL
	date_joined datetime(6) NOT NULL

ER Diagram:

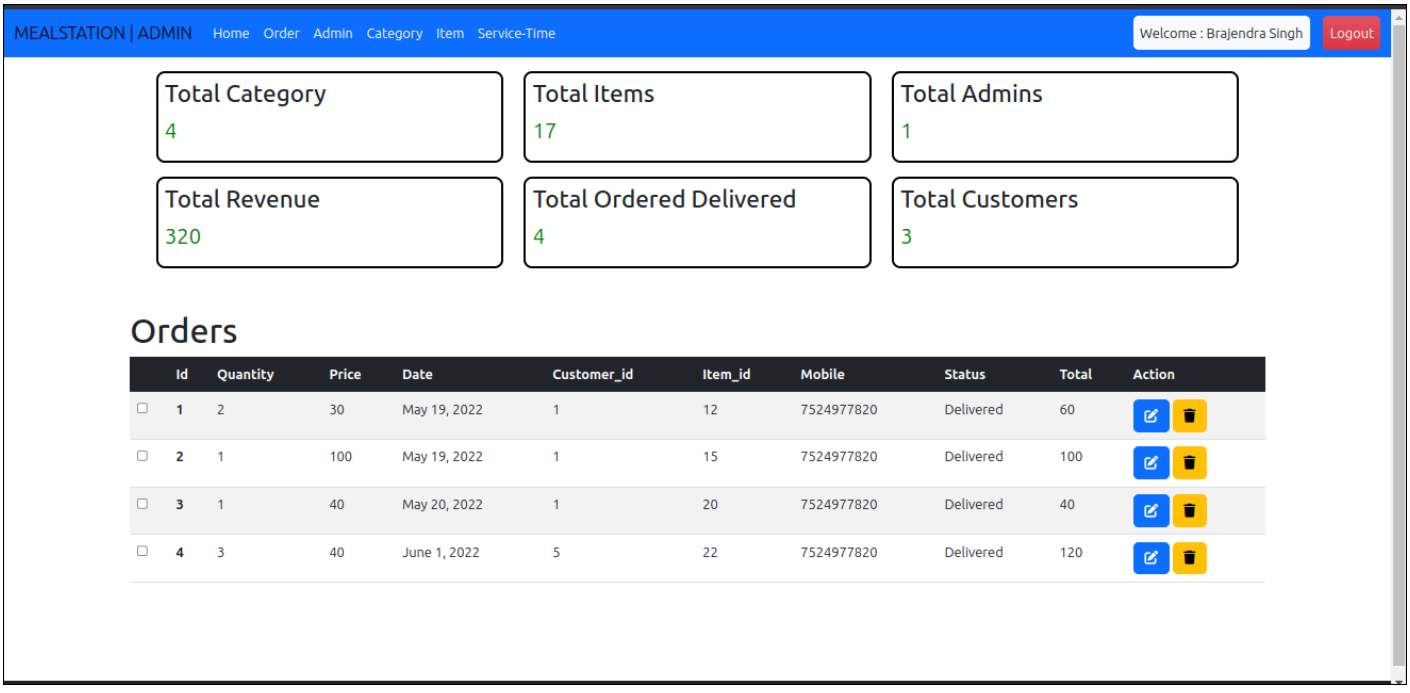

name
image
id
Category
image
message
id
Carousel
username
last_name
first_name
is_superuser
email
last_login
password
id
is_active
date_joined
is_staff
auth_user
manages
id
name
price
image
category_id
stock
Item
ServiceTime
id
image
open
close
has
id
quantity
customer_id
item_id
Cart
first_name
last_name
email
id
mobile
password
token
Customer
manages
places
id
item_id
quantity
Order
mobile
status
price
payment_method
date
studentId
total
customer_id

Screen/Snap-Shots of the project

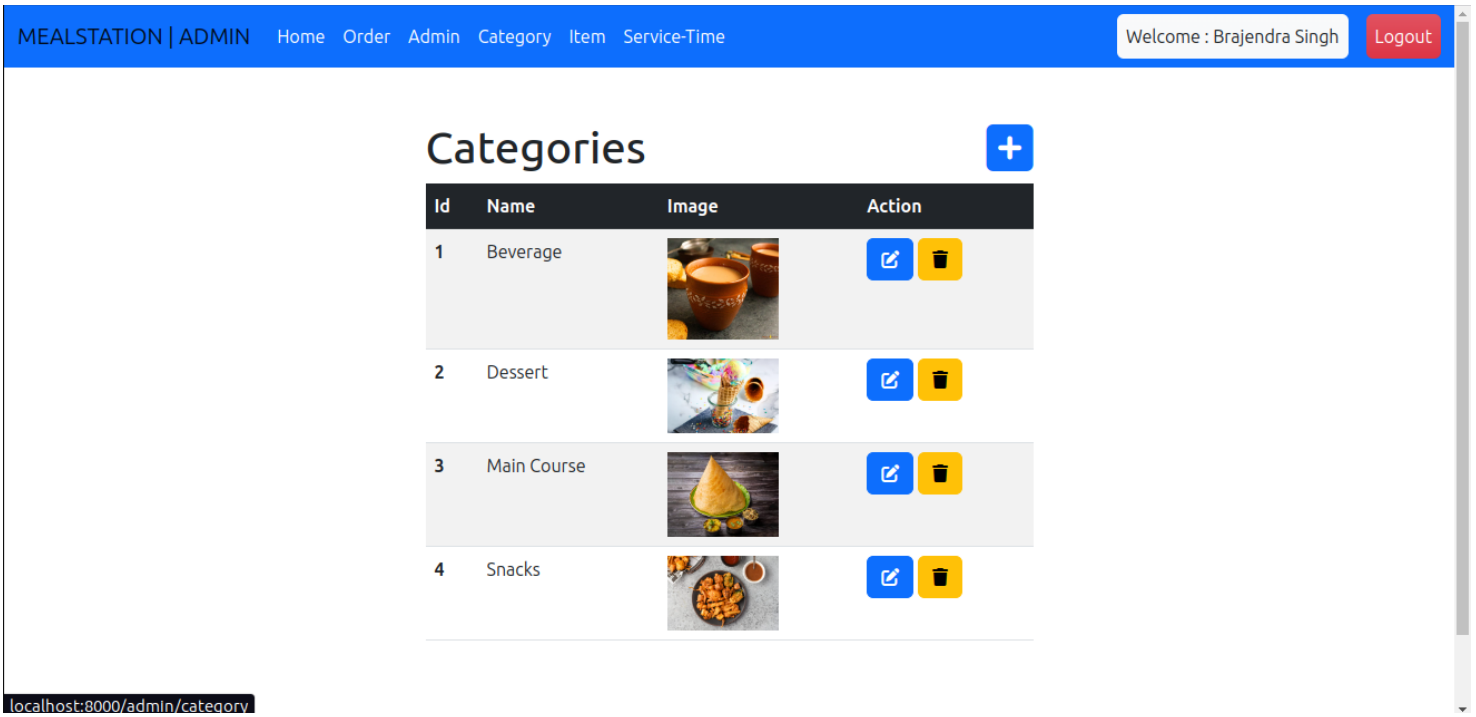
Admin Area:

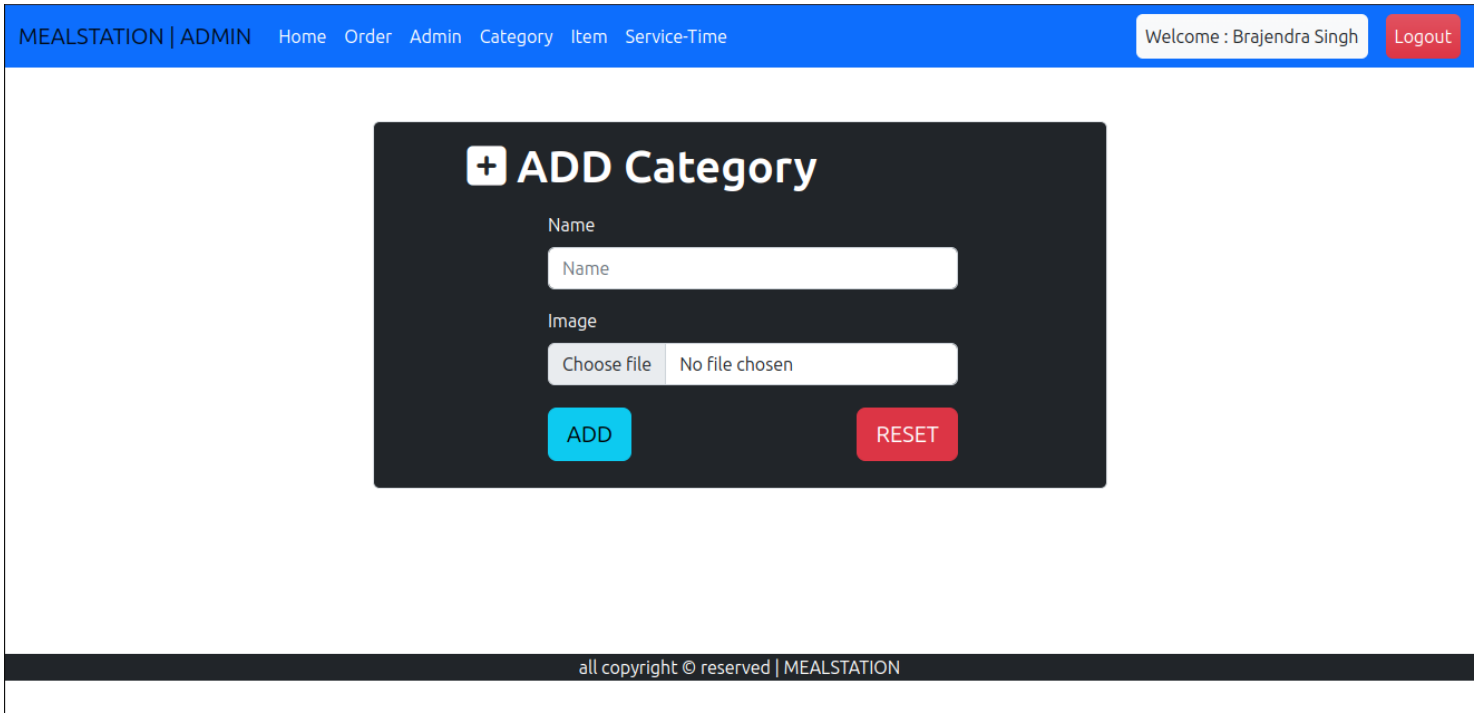


Dashboard:

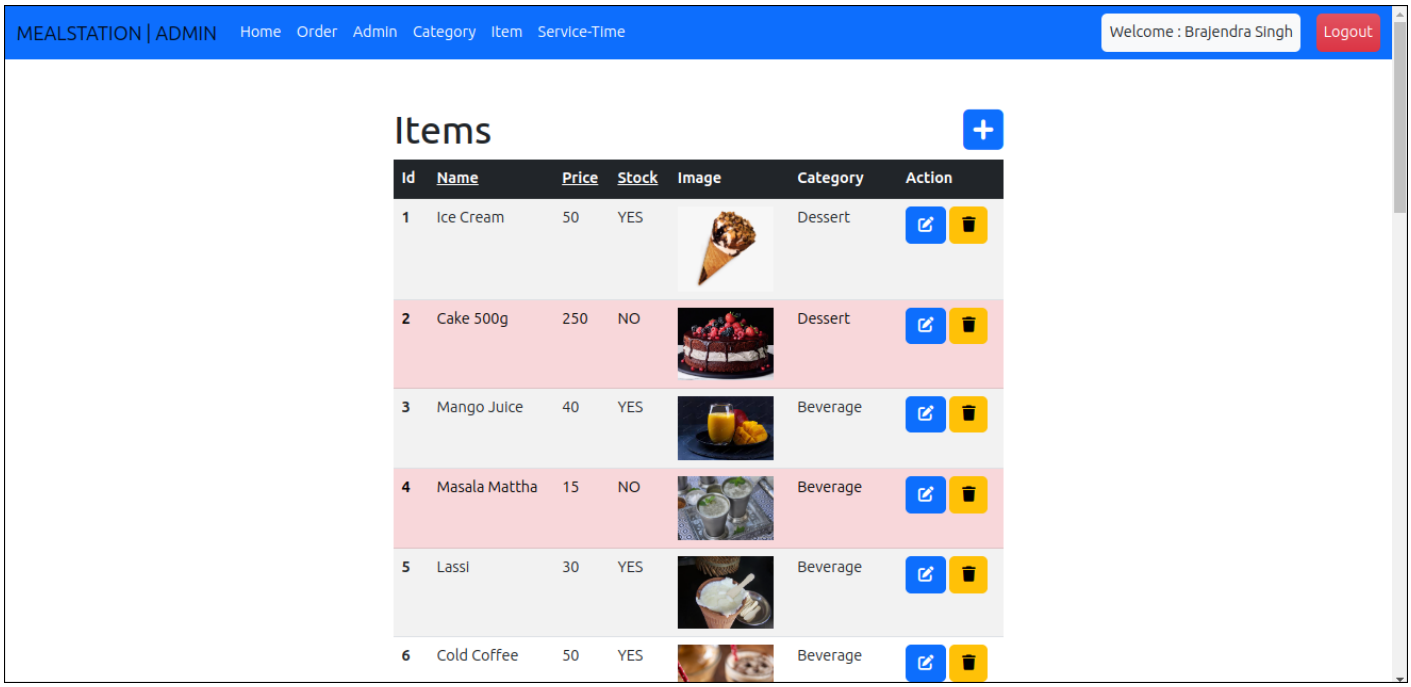


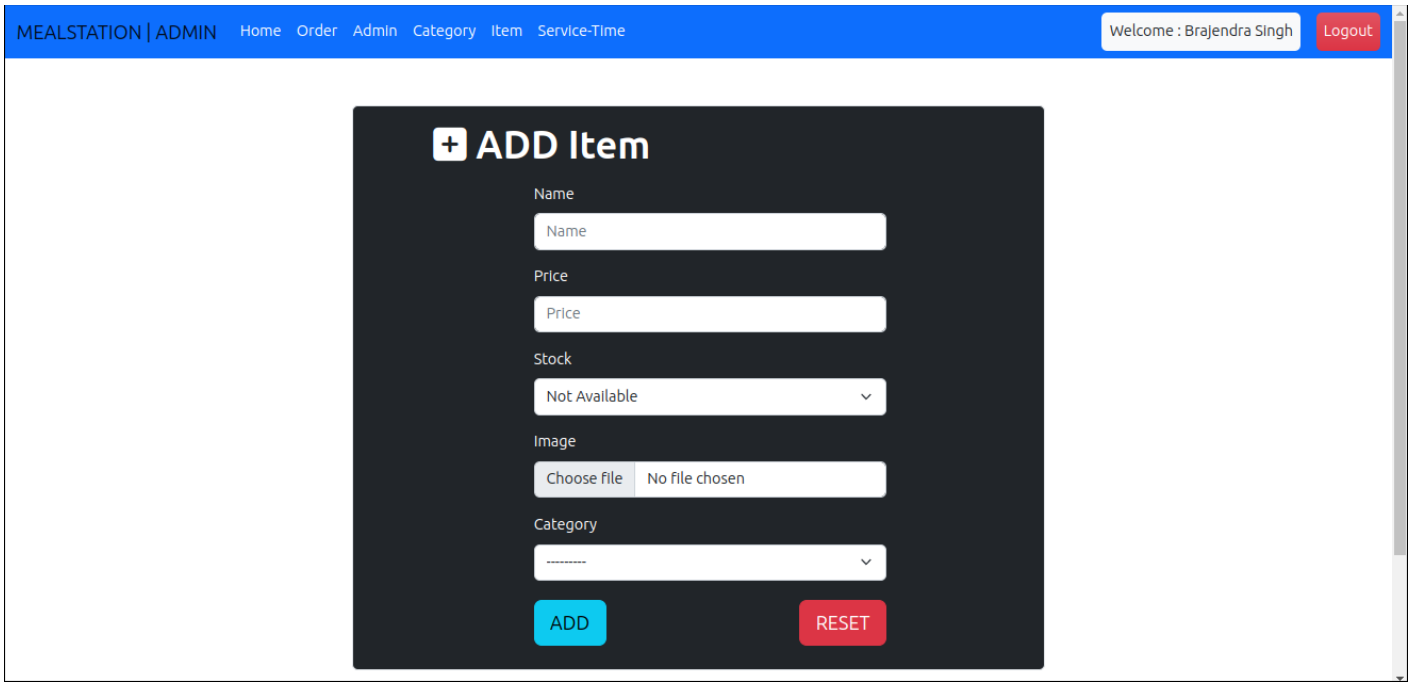
Category:


MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
Categories
Id
Name
Image
Action
1
Beverage
2
Dessert
3
Main Course
4
Snacks
localhost:8000/admin/category

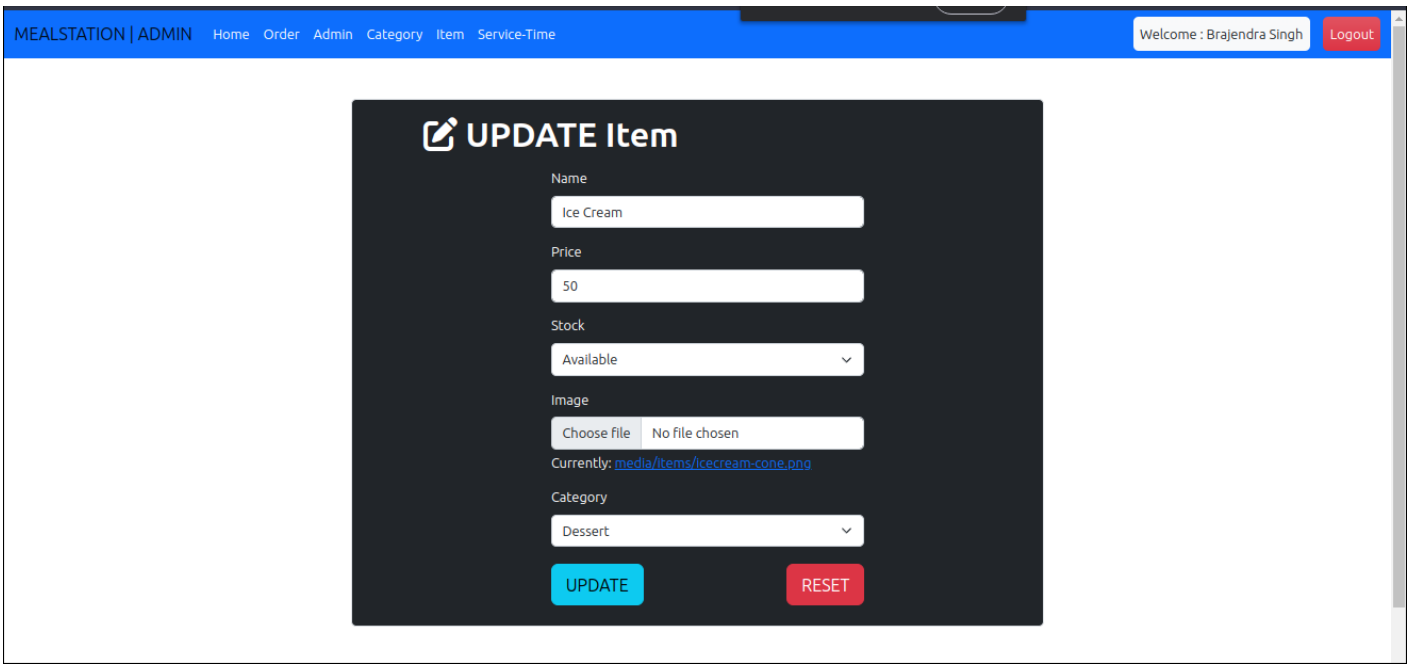

MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
ADD Category
Name
Name
Image
Choose file
No file chosen
ADD
RESET
all copyright © reserved | MEALSTATION

Items:

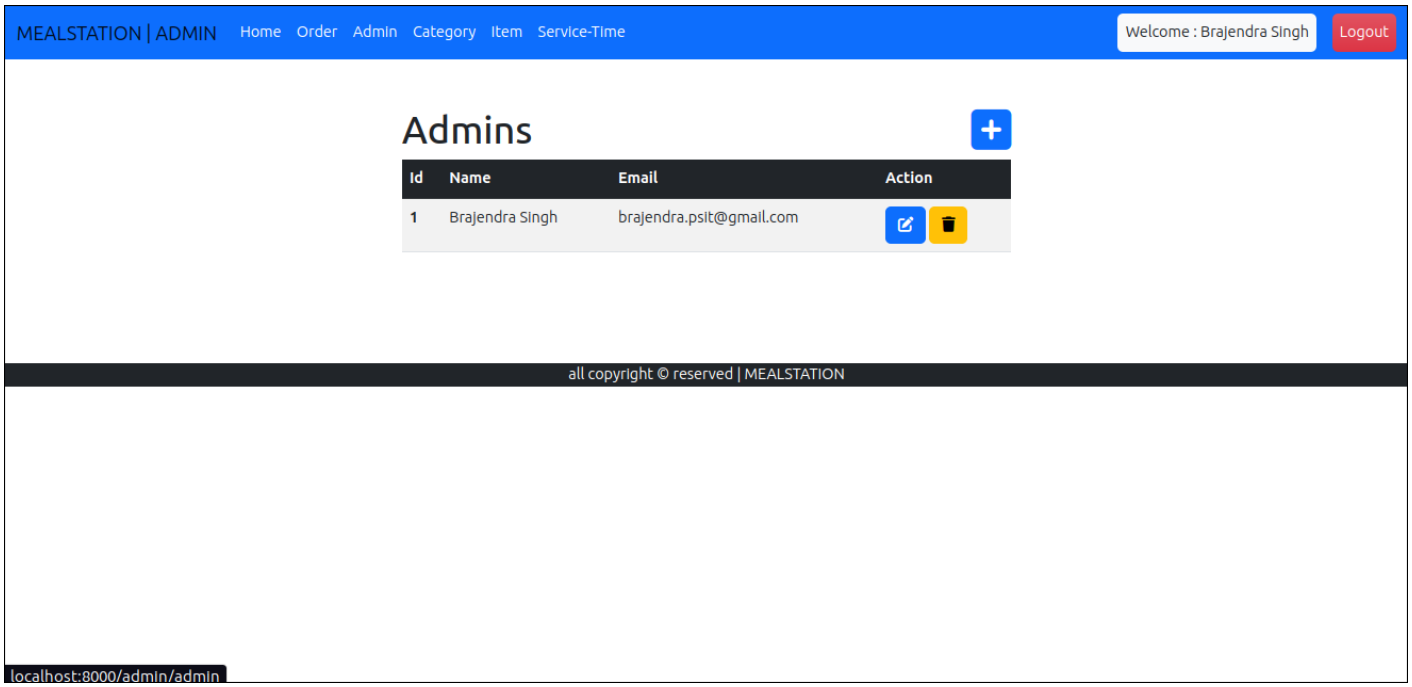

MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
Items
Id
Name
Price
Stock
Image
Category
Action
1
Ice Cream
50
YES
Dessert
2
Cake 500g
250
NO
Dessert
3
Mango Juice
40
YES
Beverage
4
Masala Mattha
15
NO
Beverage
5
Lassi
30
YES
Beverage
6
Cold Coffee
50
YES
Beverage

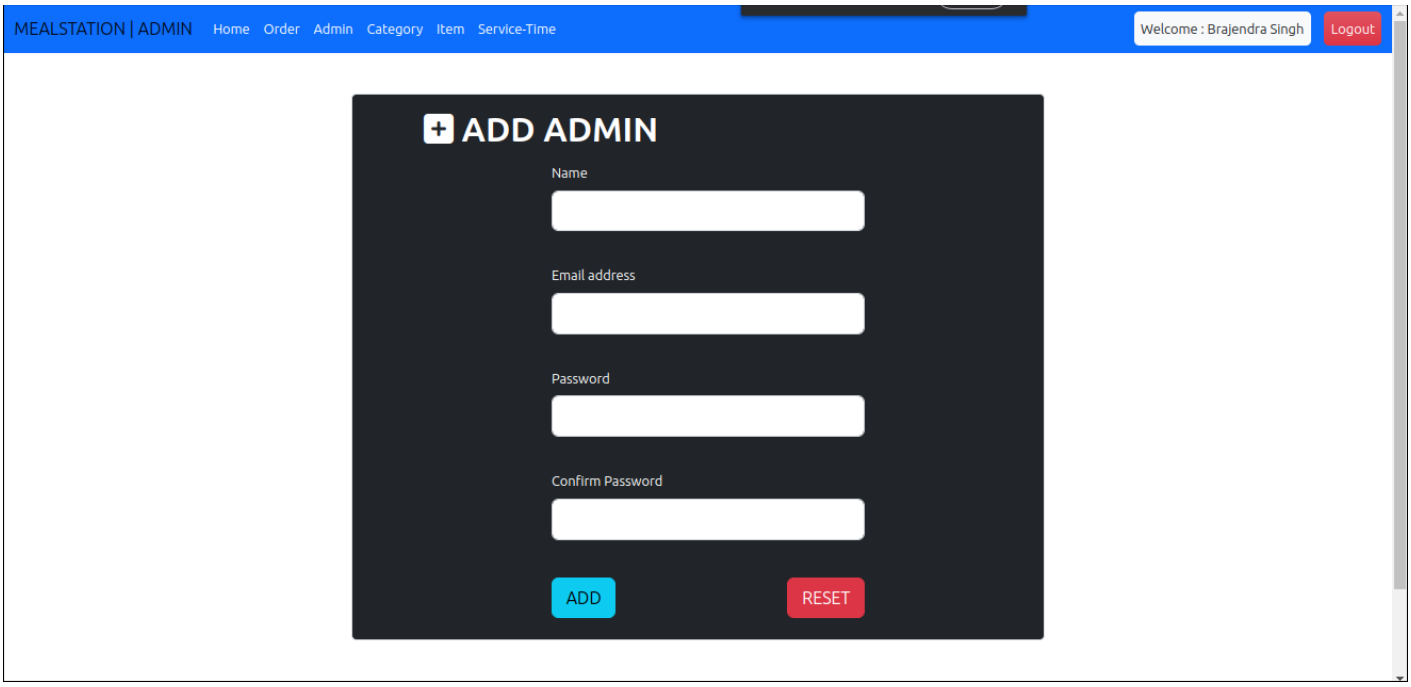

MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
ADD Item
Name
Name
Price
Price
Stock
Not Available
Image
Choose file
No file chosen
Category

ADD
RESET

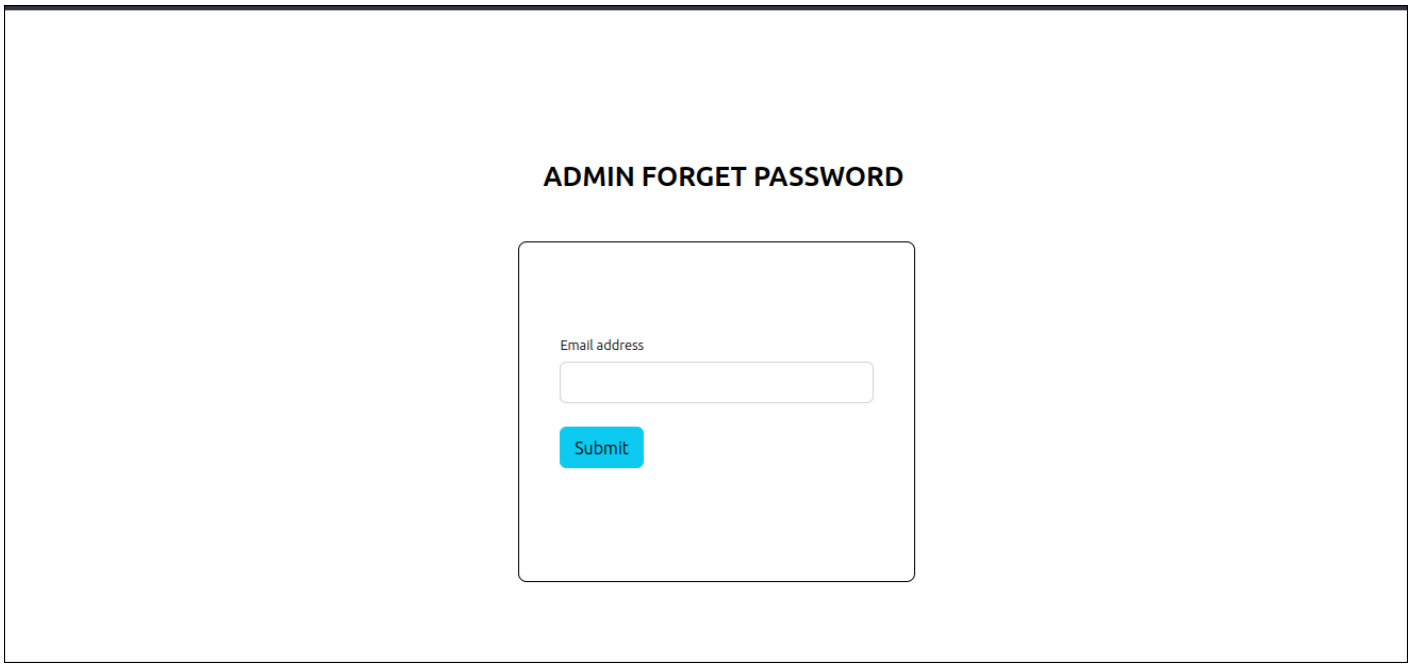

MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
UPDATE Item
Name
Ice Cream
Price
50
Stock
Available
Image
Choose file
No file chosen
Currently: media/items/icecream-cone.png
Category
Dessert
UPDATE
RESET

Admins:

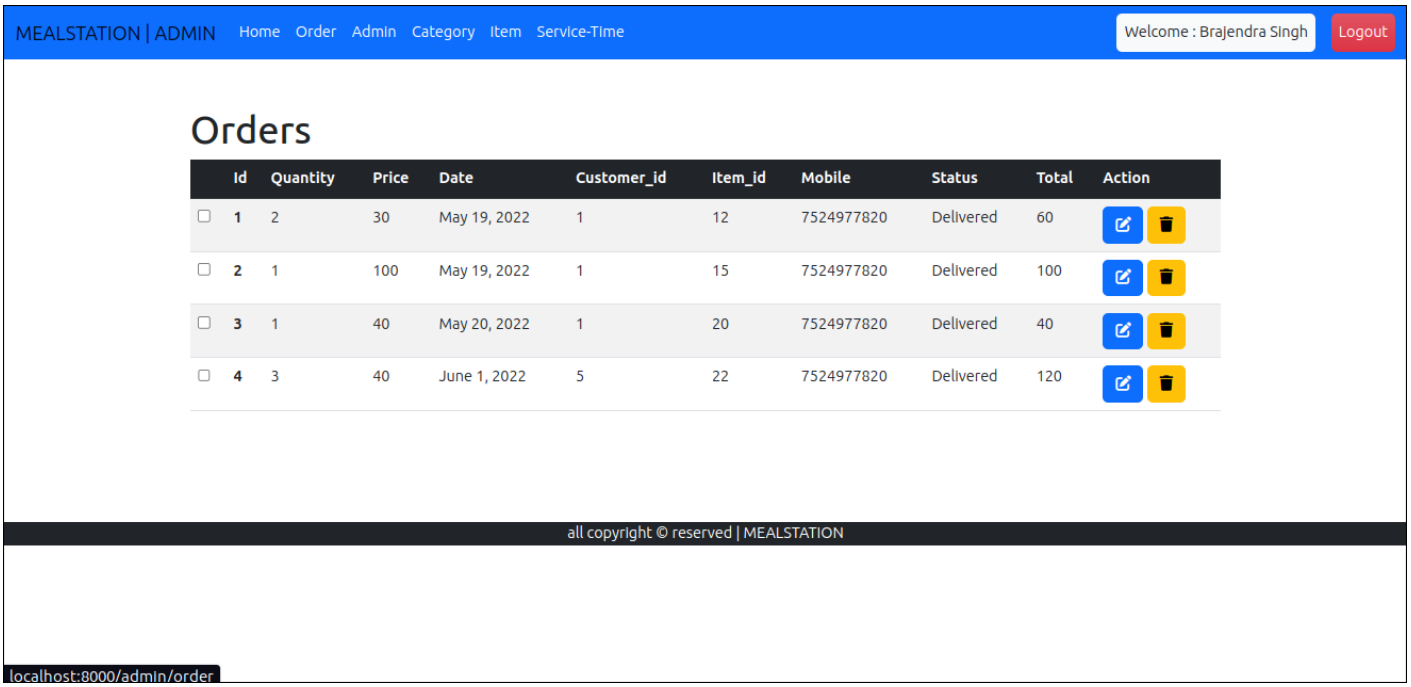

MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
Admins
Id
Name
Email
Action
1
Brajendra Singh
brajendra.psit@gmail.com
all copyright © reserved | MEALSTATION
localhost:8000/admin/admin



Admin password reset:

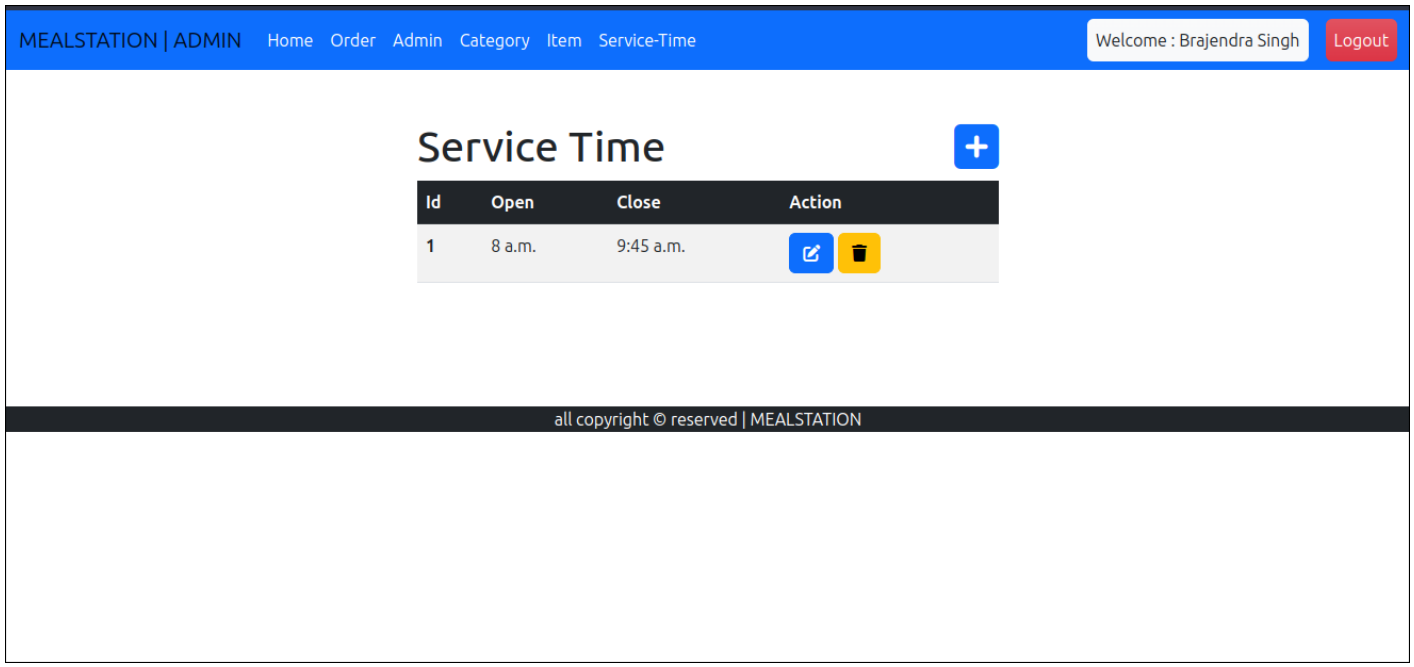


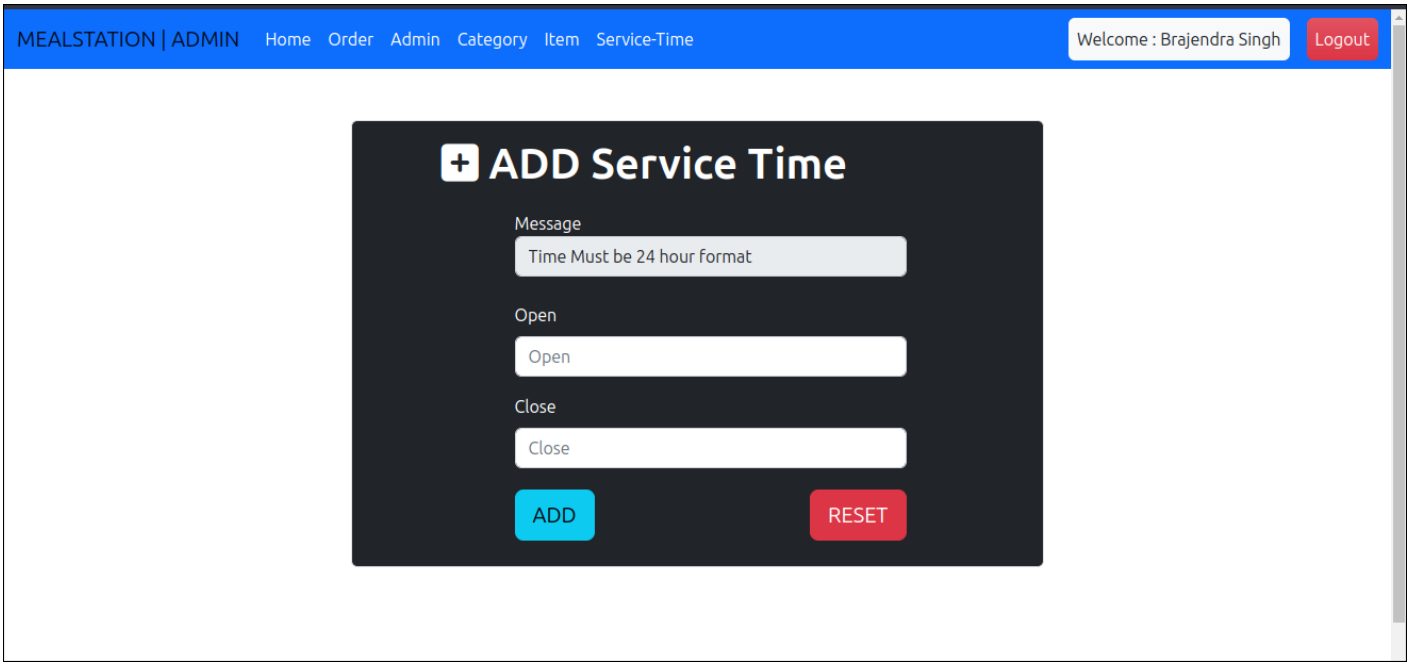
Orders:


MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
Orders
Id
Quantity
Price
Date
Customer_id
Item_id
Mobile
Status
Total
Action
1 2 30 May 19, 2022 1 12 7524977820 Delivered 60
2 1 100 May 19, 2022 1 15 7524977820 Delivered 100
3 1 40 May 20, 2022 1 20 7524977820 Delivered 40
4 3 40 June 1, 2022 5 22 7524977820 Delivered 120
all copyright © reserved | MEALSTATION
localhost:8000/admin/order

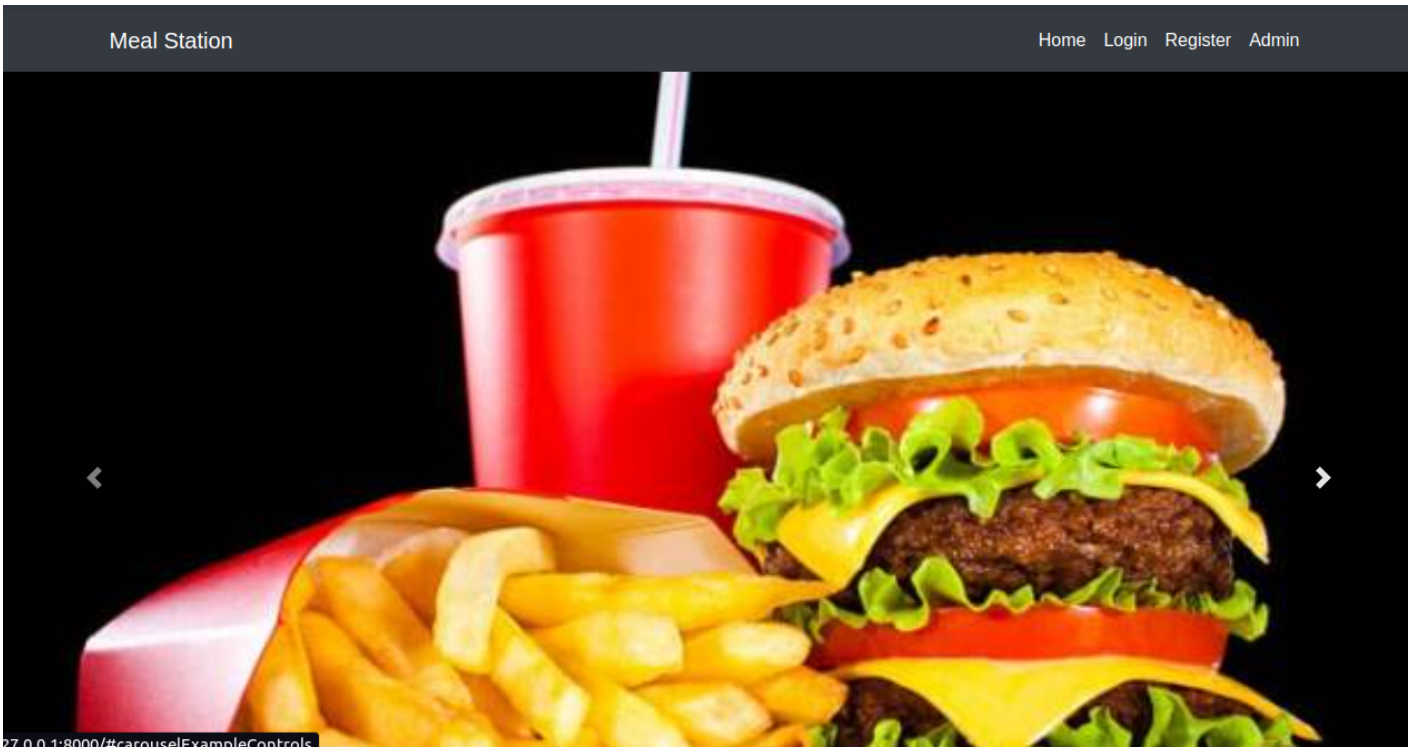

MEALSTATION | ADMIN
Home
Order
Admin
Category
Item
Service-Time
Welcome : Brajendra Singh
Logout
UPDATE Order
Item
Aloo Paratha
Customer
Brajendra Singh
Quantity
2
Price
30
Total
60
Payment method
pod
Studentid
29534
Mobile
7524977820
Date
2022-05-19
Status
Delivered
UPDATE
RESET

Service Time:

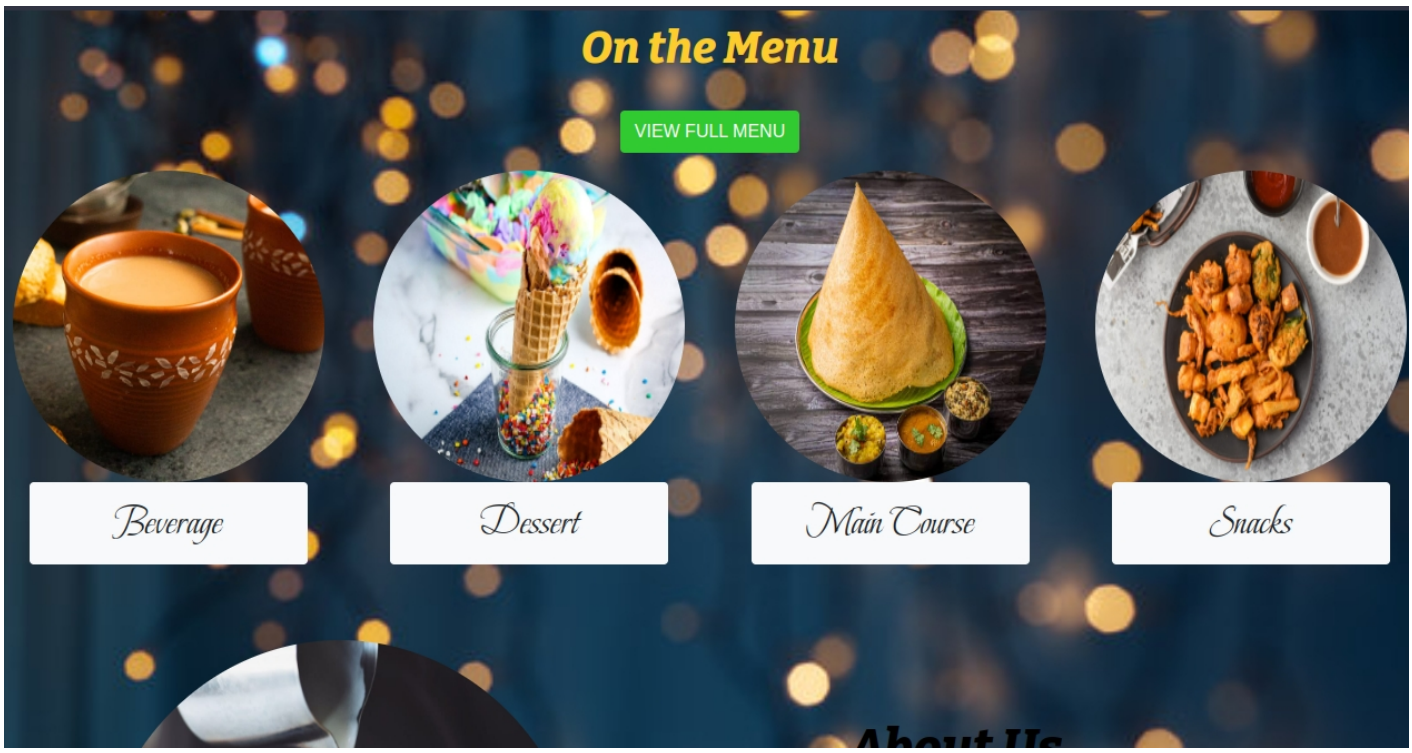


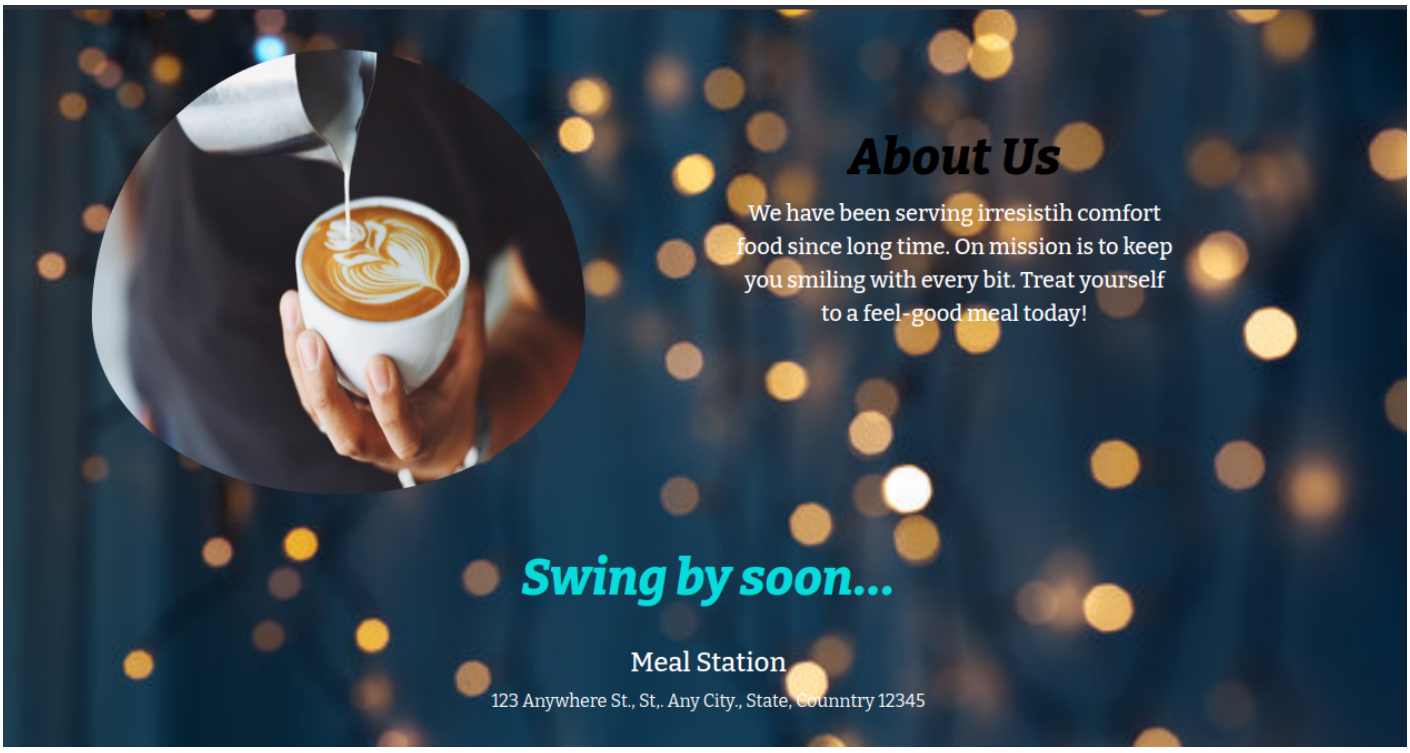


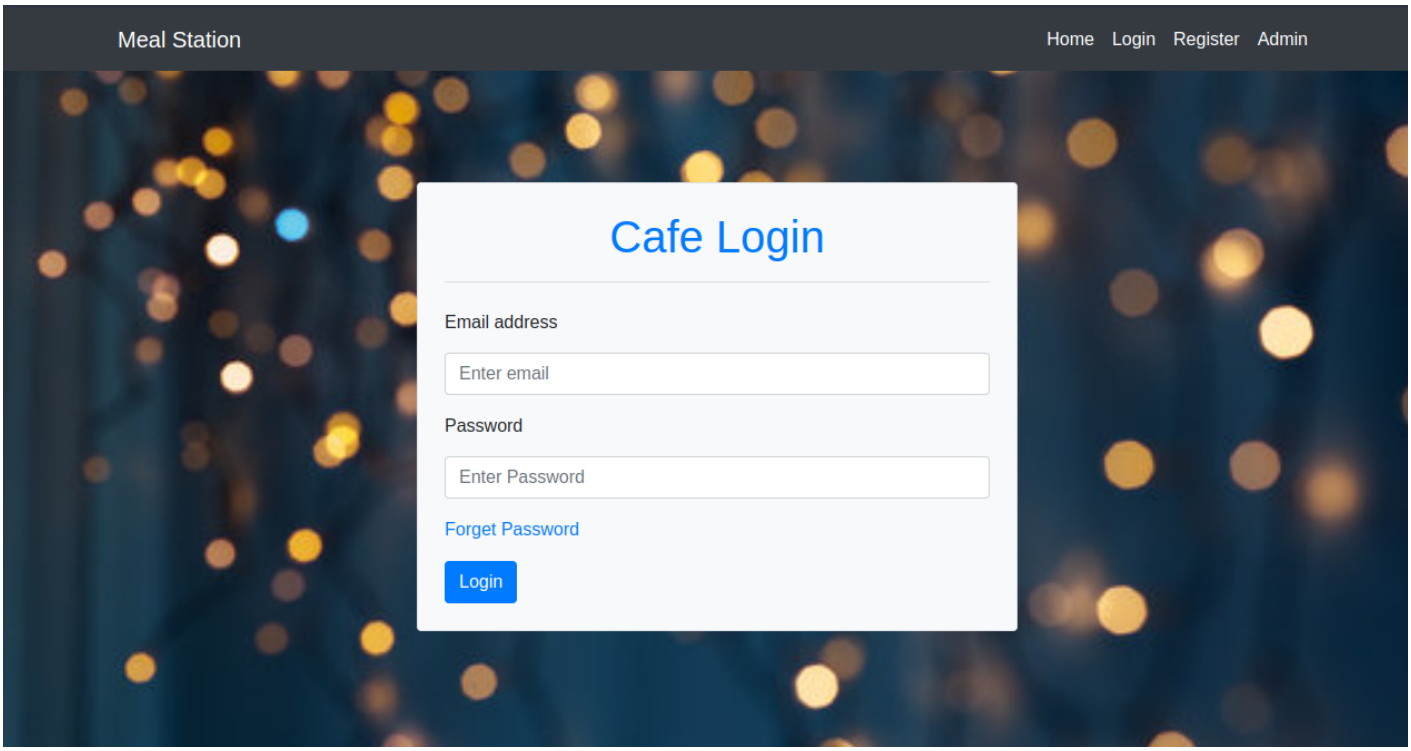
User Area :

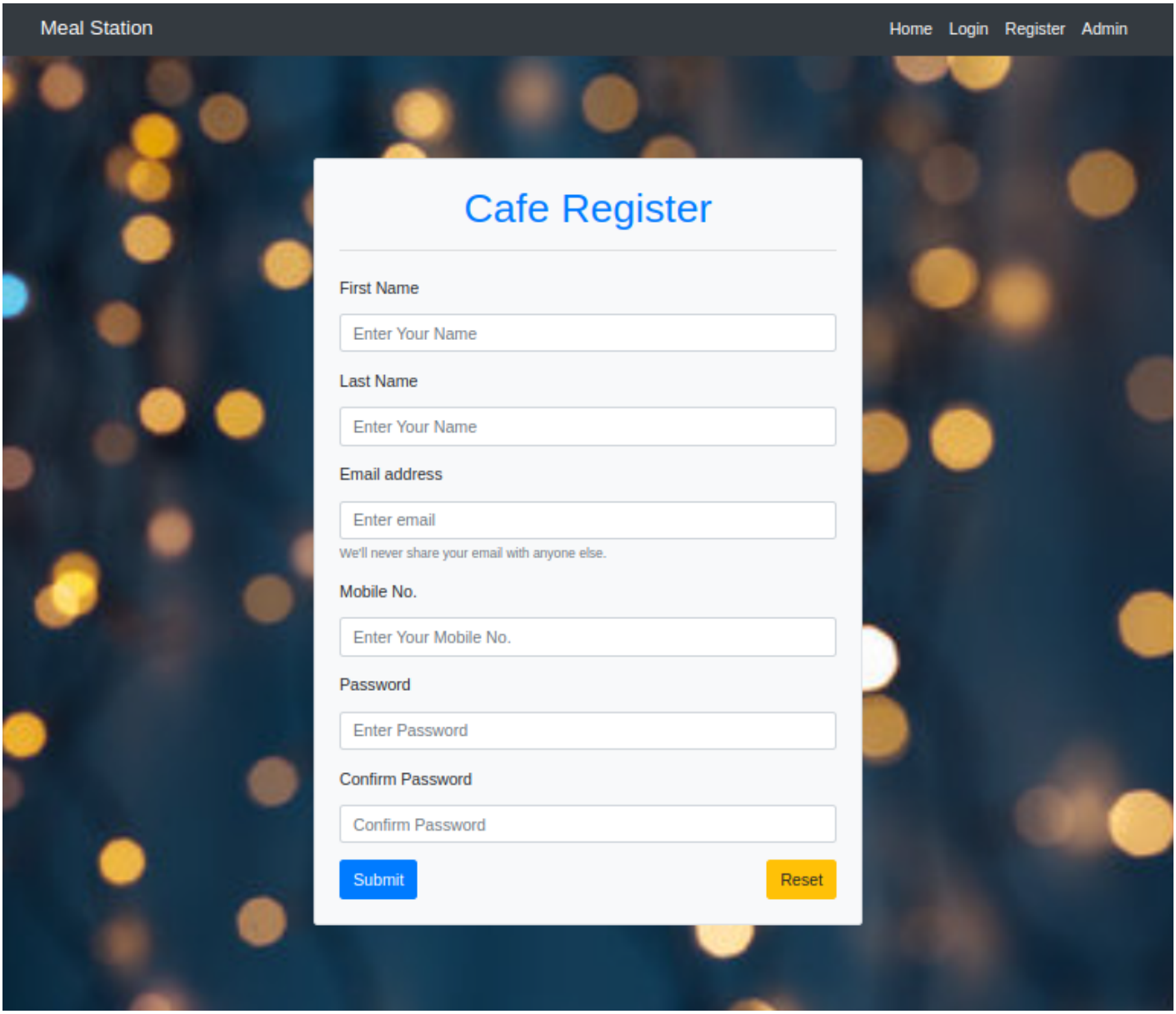

Meal Station
Home
Login
Register
Admin

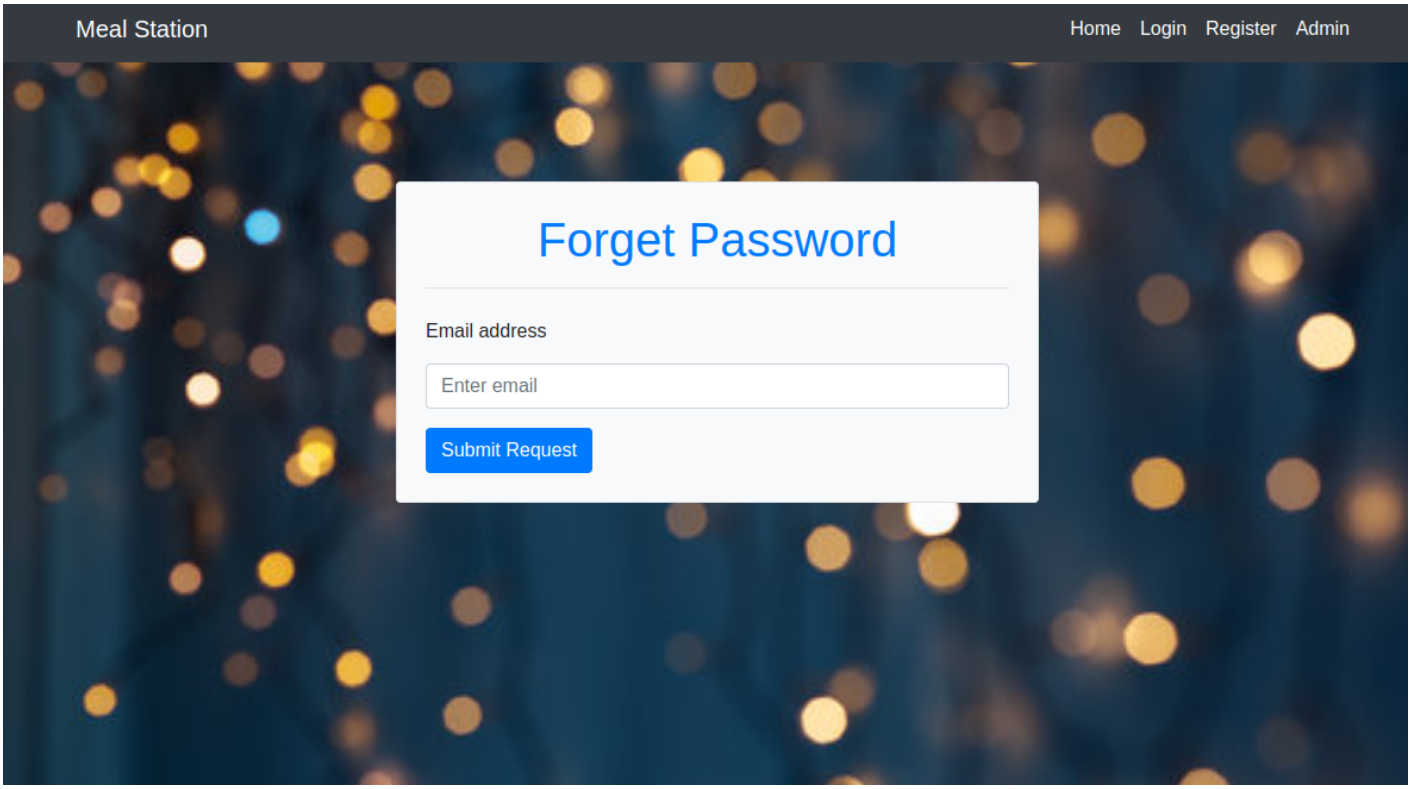

A Matter of Taste
There's a reason our regulars have been coming back for more. Word of mouth has been our only advertisement - but happy mouths are the best ambassadors!
1.
Fresh, Seasonal ingredients
Our menu follows the seasons, featuring the best local produce available
2.
Comfort Food, Elevated
We have simple crowd-favorites on the menu - but we do them well. Our flavors stand out guaranteed.
3.
Better Beverages
We're your café and barista in one, serving your favorite cuppa by sun-up, and your cocktail-of-choice by sundown.

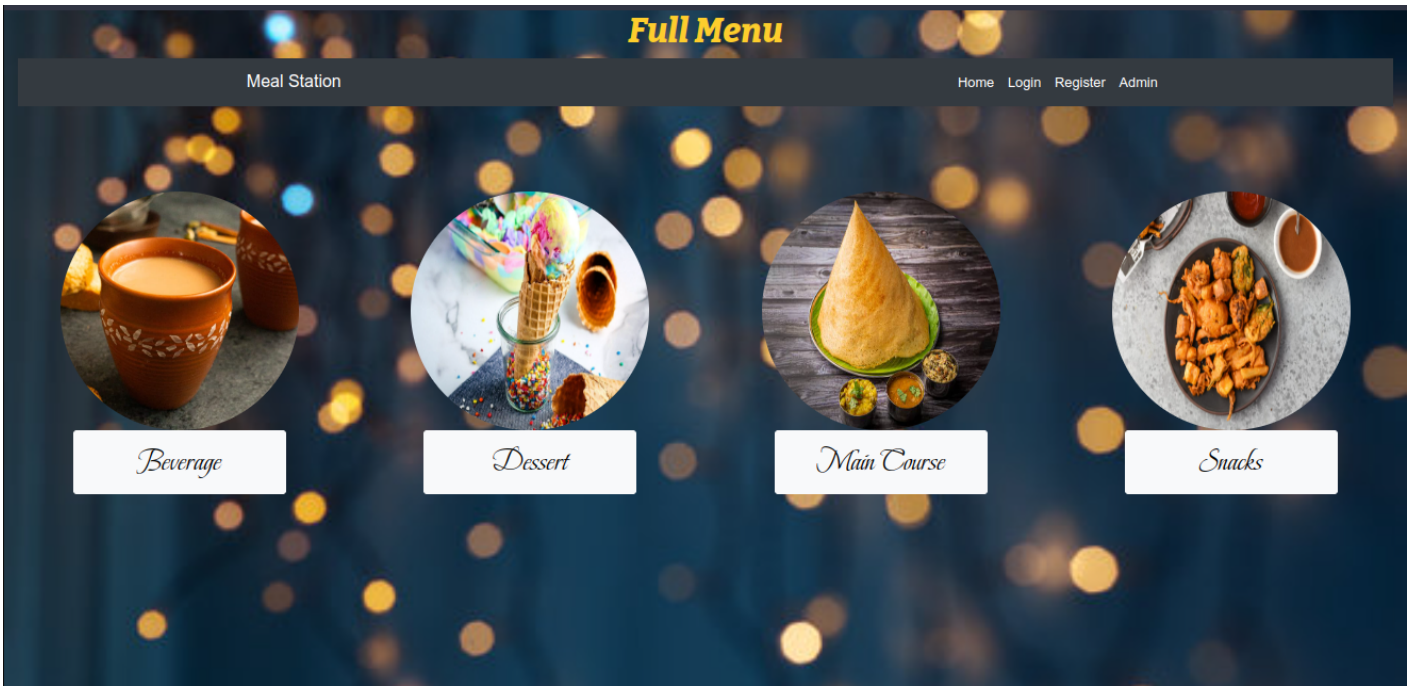

On the Menu
VIEW FULL MENU
Beverage
Dessert
Main Course
Snacks


About Us
We have been serving irresistih comfort food since long time. On mission is to keep you smiling with every bit. Treat yourself to a feel-good meal today!
Swing by soon...
Meal Station
123 Anywhere St., St,. Any City., State, Counntry 12345

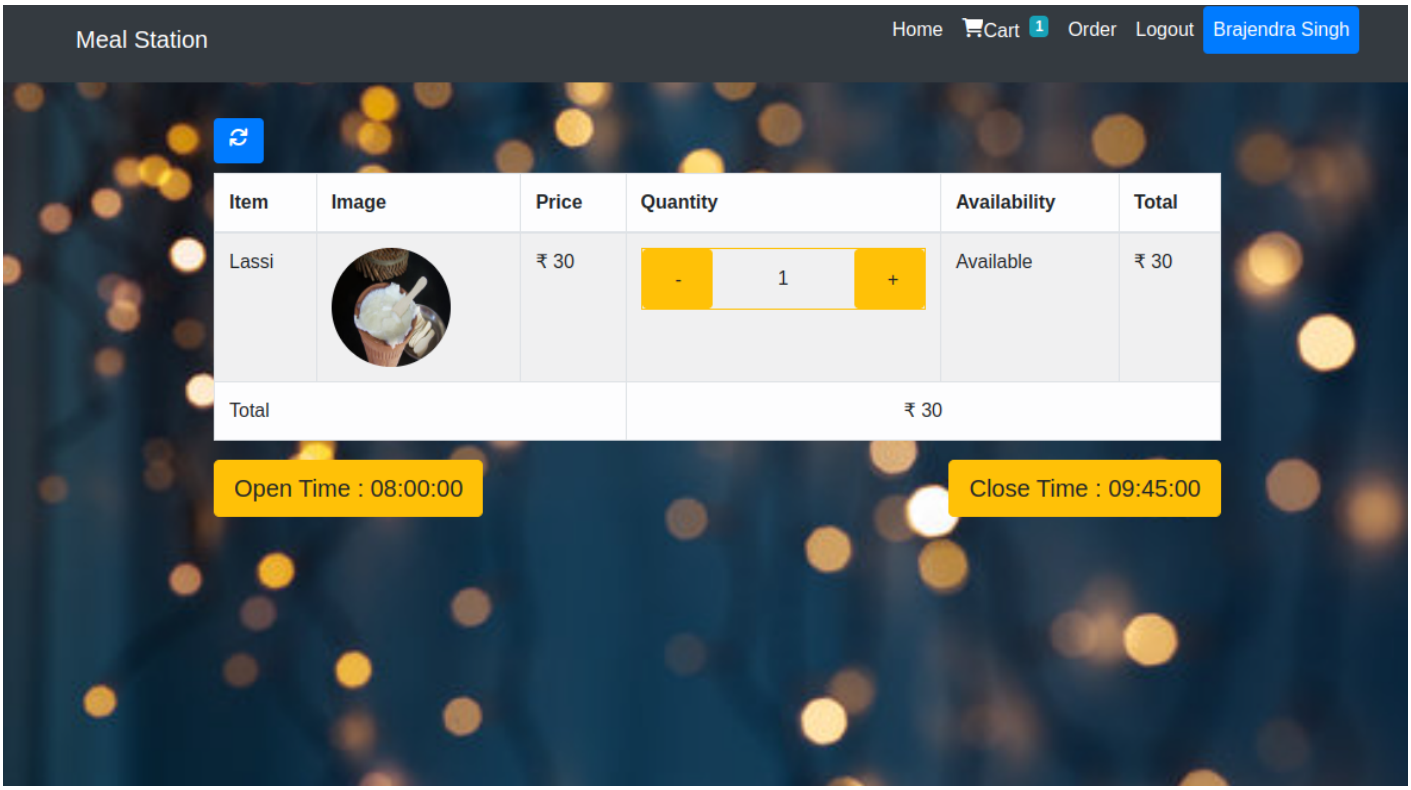

Meal Station
Home Login Register Admin
Cafe Login
Email address
Enter email
Password
Enter Password
Forget Password
Login


Meal Station
Home Login Register Admin
Cafe Register
First Name
Enter Your Name
Last Name
Enter Your Name
Email address
Enter email
We'll never share your email with anyone else.
Mobile No.
Enter Your Mobile No.
Password
Enter Password
Confirm Password
Confirm Password
Submit
Reset


Meal Station
Home Login Register Admin
Forget Password
Email address
Enter email
Submit Request


Full Menu
Meal Station
Home Login Register Admin
Beverage
Dessert
Main Course
Snacks

After User Login



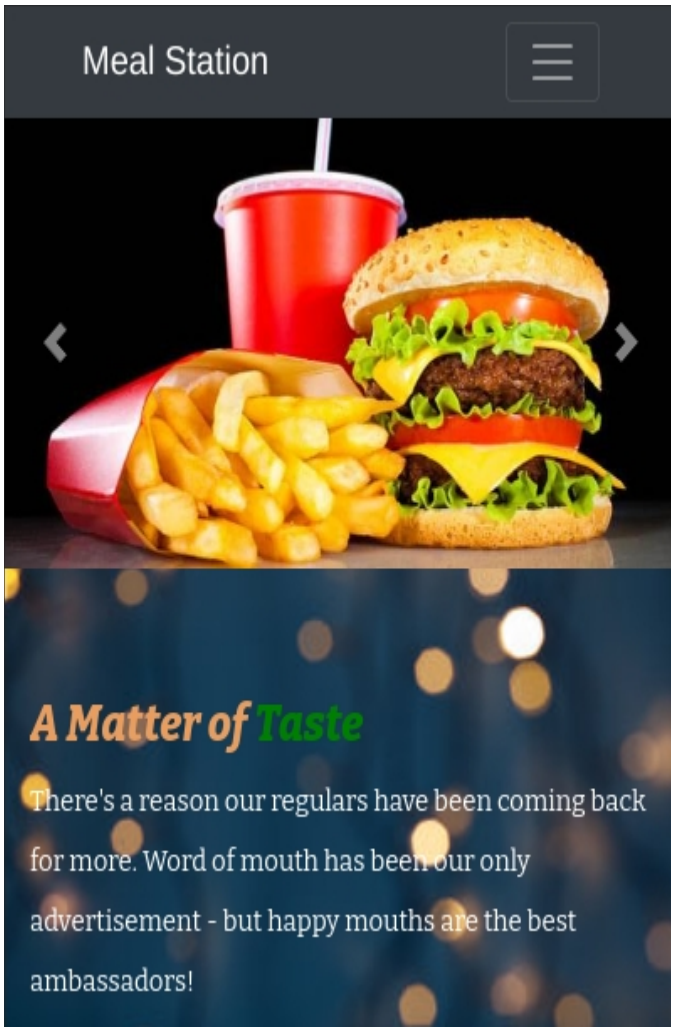
Meal Station

Home Cart 1 Order Logout Brajendra Singh

Your Orders

ID	Item	Image	Price	Quantity	Payment Method	Student ID	Total	Date	Status
16	Mango Juice		₹ 40	1	pod	29534	₹ 40	May 20, 2022	Pending
15	Paneer Sabji		₹ 100	1	pod	29534	₹ 100	May 19, 2022	Ready
14	Aloo Paratha		₹ 30	2	pod	29534	₹ 60	May 19, 2022	Delivered

Mobile View:


Meal Station
A Matter of Taste
There's a reason our regulars have been coming back for more. Word of mouth has been our only advertisement - but happy mouths are the best ambassadors!


1. Fresh, Seasonal ingredients
Our menu follows the seasons, featuring the best local produce available
2. Comfort Food, Elevated
We have simple crowd-favorites on the menu - but we do them well. Our flavors stand out guaranteed.
3. Better Beverages
We're your café and barista in one, serving your favorite cuppa by sun-up, and your cocktail-of-choice by sundown.

Conclusion

- The development of Canteen Management System involved many phases. The approach used is a top-down one concentrating on what first then how and moving to successive levels of details.
- The first phase started with a detailed study of the problems and prospects of ordering in Foods.
- This Software is efficient in maintaining customer's details and can easily perform operations on platform.

References

- Google and Stack Overflow for problem solving
- https://html5.org/
- https://www.w3.org/Style/CSS/Overview.en.html
- https://getbootstrap.com/docs/4.0/getting-started/introduction/
- https://www.python.org/
- https://www.djangoproject.com/
- https://www.javatpoint.com/
- https://en.wikipedia.org/
- https://www.youtube.com/
- https://www.tutorialspoint.com/