e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:02/February-2024 Impact Factor- 7.868 www.irjmets.com

ONLINE FOOD MART SYSTEM

Pranjal Shejal[*1], Rinky Yadav[*2], Prasad Khade[*3], Radhika Potdar[*4]

[*1,2,3,4]Department Of Electronics And Telecommunication Engineering, BRACT's Vishwakarma Institute Of Information Technology, Pune, India.

DOI : https://www.doi.org/10.56726/IRJMETS49102

ABSTRACT

Foodmart is an order management software for food delivery companies. Foodmart helps you order food from lots of different places all at once or one at a time. It's like having a super convenient system for getting your favorite meals delivered to you. Users can order through a website, while administrators handle restaurant, food, and order data. Delivery personnel use a mobile app. All these apps get their information from a main server. The article explains how the software system was built, detailing the technology, tools, and methods used in the development.

Keywords: Spring MVC, Spring boot, Angular 13, MySql, User Interface.

I. INTRODUCTION

The primary goal behind creating the Food Mart project is to modernize the conventional method of taking orders by introducing a computerized system. A crucial aspect driving the development of this project is its ability to swiftly generate accurate order summary reports whenever necessary. The scope of this project is extensive. It can be effectively utilized by various types of restaurants or fast-food establishments to maintain comprehensive records of their customers' orders. This project offers simplicity, speed, and precision in its operations. Additionally, it consumes minimal disk space, thereby optimizing storage efficiency. The utilization of MYSQL Server as the backend for Online Food Mart ensures the utmost security and eliminates the risk of data loss. Lots of places have apps for ordering food from local restaurants. But usually, these apps only let you order from one place at a time. This works fine when each restaurant handles its own deliveries. But in some smaller towns, a bunch of restaurants team up with the same delivery service. For example, most local restaurants work with one delivery company. Right now, if you want to order food, you have to call that company, which isn't very convenient. A better way would be to have a system that focuses on the delivery. FoodMart does this. It's a website where you can order food and see all the restaurants and their menus. Unlike other apps, FoodMart lets you order from different restaurants all in one go. You don't need separate orders for each place. There's also a part of the website just for the delivery company folks. They can see and manage all the orders, add new restaurants or menus, and change the existing ones. The delivery people have an app too. It helps them see the orders, accept them, and get all the details they need to make sure the food gets to you on time.

II. LITERATURE SURVEY

1. Paper Name: Netfood: A Software System for Food Ordering and Delivery

Author: Cristina-Edina Domokos, Barna Séra, Károly Simon, Lajos Kovács, Tas-Béla Szakács

Abstract: Netfood is a food delivery software handling orders from multiple restaurants, letting users order individually or as a group via a web interface. Administrators manage restaurant, food, and order data. A central server serves data to both client applications—a web interface for users and a mobile app for delivery personnel. The article details the system's architecture, implementation, and the technologies utilized in its development.

2. Paper Name: Spring Framework Reference Documentation

Author: Rod Johnson , Juergen Hoeller , Keith Donald , Colin Sampaleanu , Rob Harrop , Thomas Risberg, Alef Arendsen , Darren Davison , Dmitriy Kopylenko.

Abstract: the Spring framework architecture encompasses a modular approach for building robust Java applications. Spring Boot simplifies configuration by providing defaults and auto- configurations, streamlining development. Spring MVC is a part of Spring framework focusing on building web applications using the Model-View-Controller pattern, facilitating the development of web APIs and web applications in Java.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:02/February-2024 Impact Factor- 7.868 www.irjmets.com

3. Paper Name: AN IMPROVED ONLINE FOOD ORDERING SYSTEM

Author: MADU UGOCHUKWU JEFF

Abstract: This literature paper explores [topic], examining [main themes/ideas]. The system successfully integrated a database with a user-friendly website, enabling customers to place, review, and modify orders while receiving confirmations. Admins could track orders, update menus, and manage food categories. Implementing this system at Mountain Top University is seen as highly beneficial, streamlining food access for students.

III. METHODOLOGY

DEVELOPMENT TOOL AND METHODOLOGIES

The food mart application was developed using a three-tier architecture, integrating Spring Boot for the backend, Angular for the frontend, and MySQL as the database. The backend was implemented by configuring RESTful APIs for user authentication, product management, and order processing. Angular components were designed following Material Design principles to ensure a responsive and intuitive user interface. MySQL was employed to create normalized tables storing user information, products, and transactional data. The Agile methodology facilitated continuous integration using Git for version control. GitLab acts as a one-stop platform for managing tasks, tracking issues, storing and versioning code, and automating the continuous integration and delivery processes.

Two environments were used during the development: STS (Spring Tool Suite) for the Spring boot server and Visual Studio Code for the Web interface. Maven was used as build automation and dependency management tool for the Spring boot server and the web client was managed with the npm (Node Package Manager) system.

IV. THE FOODMART PROJECT

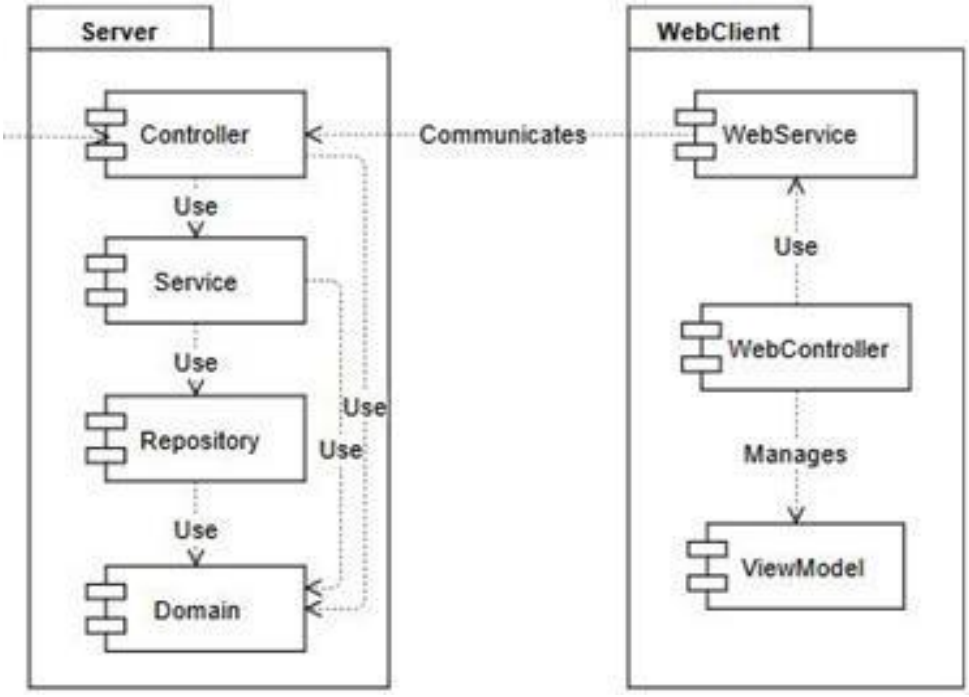

Fig. 1: The architecture of the system

A. Functionality:

The server manages two main tasks: handling user requests and communicating with the database. It connects to both a website and a mobile app through a special way of talking called a RESTful API. All the information is stored in a database that's like a big organized collection of data. People who use the system, both regular users and administrators, can sign in using a web page. They can use their Google account or make an account with their own username. When you visit the website, the first thing you see is the food available for the day. You can

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:02/February-2024 Impact Factor- 7.868 www.irjmets.com

change the date and explore different food categories too. If you like something, you can put it in a shopping cart, change the quantities, and remove items. Once you've picked what you want, you give your delivery address and phone number, and then you can place your order. If you've ordered before, some parts of the order form might already be filled out for you. You can also team up with friends to order together. Everyone can see what's in the shopping cart, and new friends can join in. You'll know when each person finishes ordering, and when everyone's done, the order can be placed. Clients have extra things they can do too, like checking their past orders, seeing the restaurants connected to the delivery company, editing their profile, and making friends with other users to order together in the future. Admins have special powers. They can see all the orders for a particular date and manage things like adding, deleting, or changing food items, restaurants, categories, and even the roles of users (like changing a regular user to a delivery person or an admin). For delivery folks, there's a special app. Only people with a delivery role can use it. After logging in, they can see the orders for the day. They can accept parts of orders (like specific items) from both individual and group orders. There's also a list of orders they've accepted already, and it includes info like the customer's phone number, which they can call directly. from the application. The system can be notified after the completion of an accepted order.

B. Architecture:-

The system consists of three main components: a Java-based Spring Boot server, an Angular 13 and webclient application (see Fig. 1).The Spring Boot server has a Four Layer architecture. repository module is manages interactions with the database and performs various tasks related to data manipulation and storage. The data is represented by model classes that can be found in the Domain module. These Entity classes are used by all the server components. The Controller module handles the incoming requests from REST APIs, while the Service layer constructs the responses by utilizing the necessary elements/components. The Controller acts as the receiver of requests, and the Service layer functions to generate the appropriate responses for those requests. the Service layer holds the rules and logic that define how the application operates. It uses the Repository layer to handle the actual data operations needed to execute the requested tasks. Essentially, the Service layer manages how things should work based on the business rules and relies on the Repository layer to handle the actual data tasks.The web client talks to the server using the WebService layer. The WebController layer utilizes services and handles the ViewModel, controlling what the user sees on the user interface

V. THE SERVER

The server is built using the Spring Java framework. It uses an Inversion of Control (IoC) container to manage components and configurations. This framework also supports Dependency Injection (DI), which helps manage how different parts of the system rely on one another.

A. Spring Boot:

Spring Boot is like an upgrade to the Spring Framework, making it simpler and faster to create and run all kinds of applications, whether they're straightforward or web-based. It's like a shortcut that speeds up the setup and configuration process. Spring Boot merges the Spring Framework with built-in Webservers: Tomcat is used within the FoodMart project.

B. Data Model:

The model classes, as shown in Figure 2, are essentially represented using Java Persistence API (JPA) entities. These entities act as blueprints that define different components of your data. Each entity contains private attributes (think of them as characteristics or properties) that are accessed through public methods called getters and setters. These attributes are made private to control how they're modified or accessed from outside the class. Moreover, these entities have a special kind of method, a constructor, that doesn't take any arguments and is accessible to anyone (public). This constructor initializes the object when it's created. To connect these entities to a database, they're marked with annotations in Java, specifically @Entity and @Table. These annotations serve as instructions for the program, indicating that each entity corresponds to a table in the database. This means that the structure and data in these Java entities align with tables and records in the database. They are stored in the com.foodmart.Entity package.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:02/February-2024 Impact Factor- 7.868 www.irjmets.com

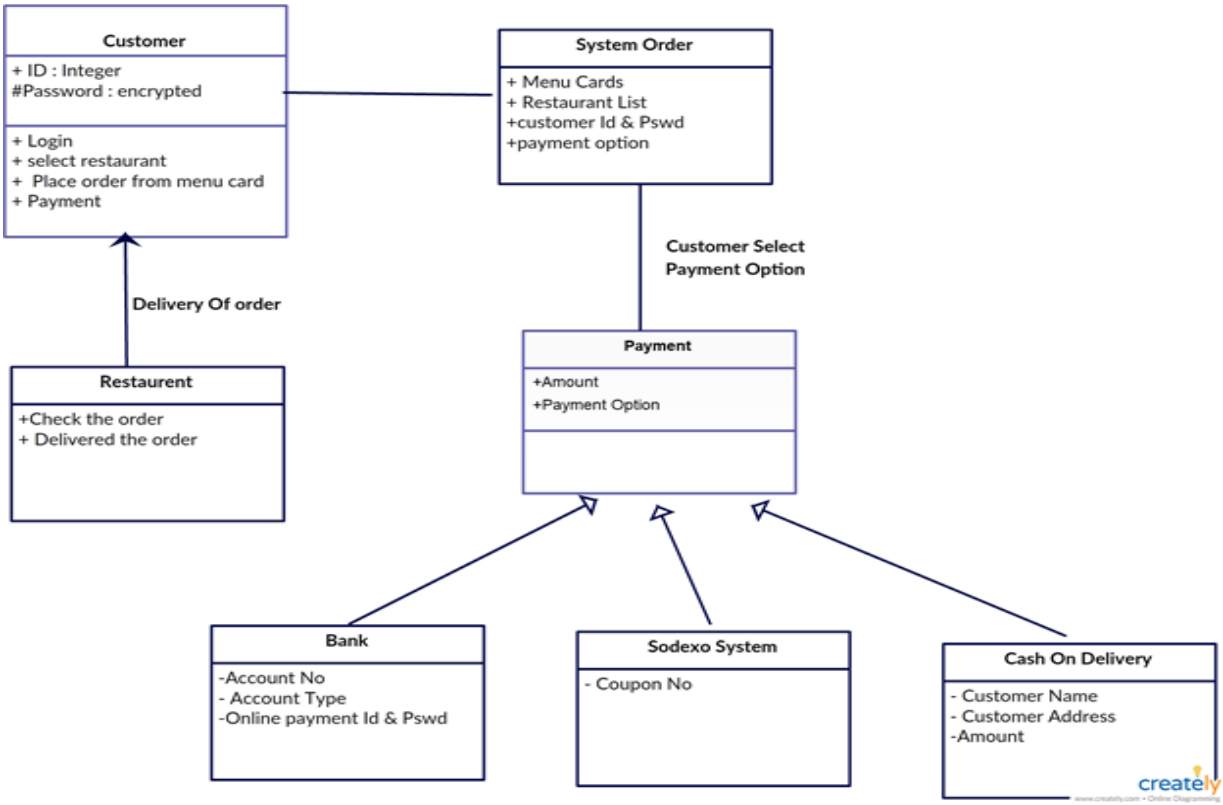

Fig. 2: Class diagram representing the model classes.

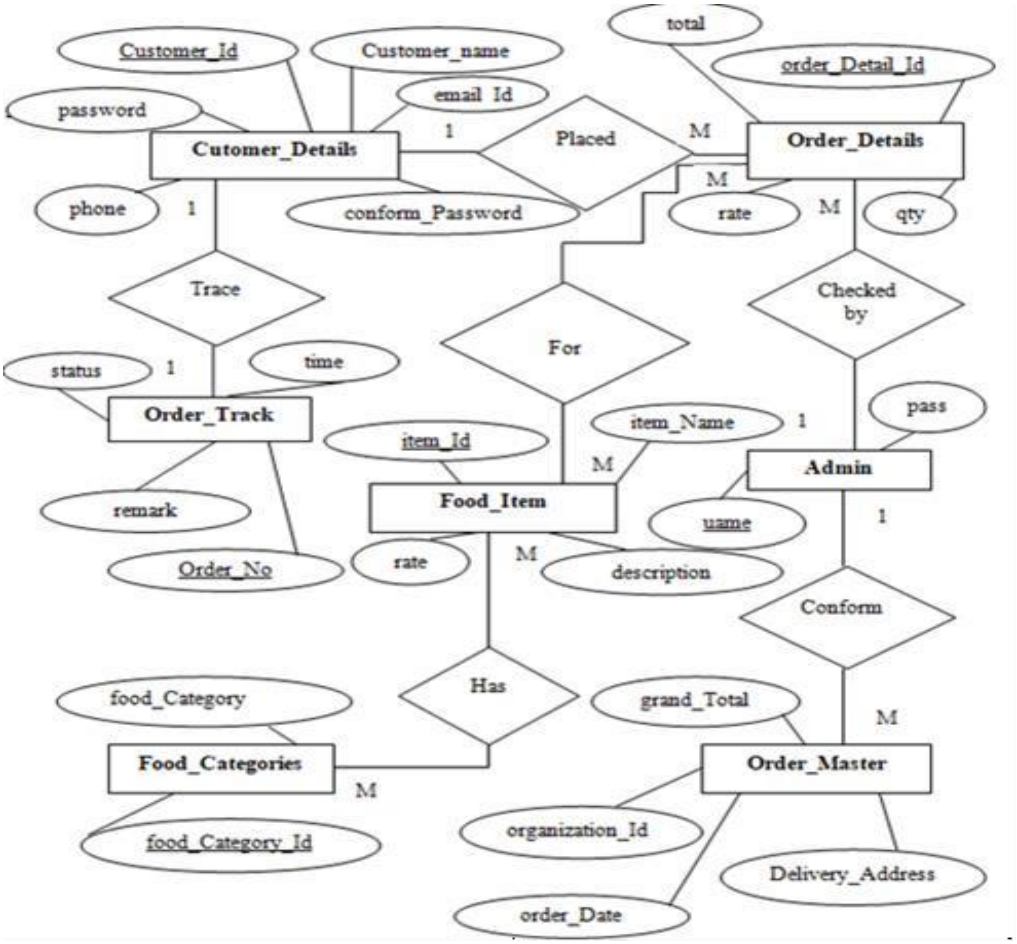

Fig. 3: Entity -Relationship Diagram of Food Mart

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:02/February-2024 Impact Factor- 7.868 www.irjmets.com

C. Data Access Layer:

The data access layer serves as a bridge between the program and the database, handling tasks related to data manipulation. In this system, a MySQL database is used to store the data. To facilitate the interaction between Java code and the database, the system employs Hibernate, which is an Object-Relational Mapping (ORM) framework. This Hibernate framework is implemented as part of the Java Persistence API (JPA) on the server. Additionally, an abstraction layer is built on top of Hibernate using the Spring Data JPA module. In this setup, the entities (representing tables in the database) are annotated with specific instructions using JPA annotations. Moreover, the Repository interfaces, which handle interactions with the database, are derived from the JpaRepository interface. These interfaces define methods for data manipulation, and they use method names that follow certain conventions. The framework automatically generates the necessary database queries based on these method names, simplifying the process of accessing and modifying data. The FoodMart repository interfaces are stored in the com.foodmart.repository package.

D. Service Layer:

The service layer acts as the brains behind the business logic in the project. When requests come in from the controllers, they're directed to the service layer. Here, the data is handled and processed. Now, within this service layer, there's interaction happening with the data access layer. Think of it this way: the service components, which manage the business logic, talk to the data access layer to handle tasks related to manipulating the data. So, when the service components need to perform actions involving the database, like retrieving information, updating records, or deleting data, they communicate with the data access layer. It's like they're asking the data access layer to do the heavy lifting when it comes to managing the actual data stored in the database. The service classes are annotated with @Service and they are stored in the com.foodmart.service package.

E. RESTful API:

REST (Representational State Transfer) is an architecture model used for client-server communication, typically allowing data exchange based on the HTTP protocol. The data and the operations are represented as resources, and these resources can be manipulated by simple operations, based on a HTTP analogy:

- POST: create resource
- DELETE: delete resource
- GET: get resource state
- PUT: change resource state

Various formats can be used for data transfer, e.g. XML,JSON, HTML, PDF, etc. The NetFood system uses the JSON format.

F. Spring Web MVC

The Web MVC framework can be used for creating MVC (Model-View-Controller) web applications and RESTful web services.The framework is built around a DispatcherServlet, this receives all of the requests and forwards them to the handlers.The handler classes are annotated with @RestController and @RequestMapping. The resources are associated with controller classes. The @RequestMapping annotation indicates the handler where the incoming request is forwarded. The requests are received by the methods from the controller classes, and after a pre-processing they are forwarded to the service layer.

G. Spring Security:

Spring Security is a powerful framework that provides authentication, authorization, and protection against common security vulnerabilities in Java applications. When combined with JSON Web Tokens (JWT), it offers a robust and stateless way to handle authentication and authorization. JWT is a compact, self-contained way to transmit information between parties as a JSON object. It consists of three parts: a header, a payload, and a signature. The header contains the type of token and the hashing algorithm used, the payload contains the claims (information), and the signature verifies that the token is valid. In Spring Security, JWTs are often used for authentication. Here's a high-level overview of how it works:

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:02/February-2024 Impact Factor- 7.868 www.irjmets.com

Authentication: When a user logs in with valid credentials, the server generates a JWT containing information about the user (like username, roles, etc.).

Token Issuance: The server signs this JWT using a secret key and sends it back to the client.

Token Storage: The client stores this token (usually in local storage or cookies) and includes it in subsequent requests in the Authorization header.

Token Verification: For protected resources, the server verifies the JWT's signature using the secret key. If valid, it extracts the user information from the token's payload. Spring Security provides mechanisms to integrate JWT authentication into your application:

Configuration: Define security configurations to validate incoming JWTs, set up authentication filters, and specify endpoints that require authentication.

Authentication Providers: Implement logic to parse and verify JWT tokens. Spring Security allows you to create custom authentication providers or use existing ones to validate tokens.

Authorization: Once the user is authenticated, Spring Security allows you to define authorization rules based on the user's roles or other claims within the JWT.

Token Refresh: Optionally, you can implement token refresh mechanisms where a user can obtain a new JWT without requiring credentials for a certain period, enhancing security and user experience. This combination of Spring Security and JWTs offers a flexible, scalable, and secure way to handle user authentication and authorization in your Java web applications, ensuring that endpoints are protected and only accessible to authorized users.

VI. THE WEB INTERFACE

Angular is front end Framework it provides the a single page application (SPA). It is developed using the Angular 13 front-end framework. The angular framework is responsible for managing the web components and it resolves the dependencies between them. It provides data binding (one-way data binding and two-way data binding) support, linking the data model to the view. Angular directives act like instructions for building dynamic HTML pages, while TypeScript enhances JavaScript by adding features for object-oriented programming, types, and classes, making code organization and maintenance more efficient.

The web client communicates with the server using the Angular HttpClient module. There are restricted resources within the FoodMart project, accessible only from specific user roles. For accessing these restricted resources interceptors are used, provided by the HttpClient. Every time a request is made through HTTP, an interceptor steps in, attaching the login-acquired token onto the request's header before sending it forward. the request to the server. In this way the user can be identified by the server and the access can be granted to the requested resource.

Similarly to resources, some services can be accessed only by users with proper permissions. The web interface supports two types of user roles: client and admin. The paths to restricted interfaces are protected by route guards.A guard functions as a gatekeeper, permitting access to specific views or services only when particular criteria are fulfilled. For instance, it might ensure the user is logged in or has a specific role before granting access.

Otherwise, if the user doesn't meet the specified conditions, they'll be directed to a default page. A route guard, which is required to implement the CanActivate interface and its canActivate method, checks conditions for access. If the conditions return true, the user gains entry to the feature. Assigning these classes to component paths limits access based on user roles. Paths without assigned guards are accessible to all. In simpler terms, without the necessary permissions, users get rerouted to a default page. These guard classes act as bouncers, deciding who gets access to specific parts of the website based on certain criteria. If a part of the site doesn't have a guard, it's open to everyone.

VII. USING THE FOODMART SYSTEM

A. User Interface:

On the home page of the web interface the available foods list are displayed. Users have the option to log in with a registered user or via goggle account or phone number by accessing the Login menu. They can also request

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:02/February-2024 Impact Factor- 7.868 www.irjmets.com

forgot password, or they can navigate to the registration page.

1. **Registration:** Application provides a link for the Users/Client Registration.
2. **Log In**: Administrator and Client can log in by entering user name and password and manage their work on website.
3. **Save information**: Client enter all its necessary information by filling personal info form and system save that information.
4. **Change requirements**: Customer can change any of their information any time.
5. **Food Menu**: Admin can insert, update and delete the food items from the menu list.
6. **Show Food Menu:** There is a list of all types of food the company is dealing with the available themes.
7. **Record Order Details**: Customer can select food items from menu and can add the desired food items to the cart. Customer can place the order and gets the confirmation against that Order in the form of order no.
8. **Show Order Status**: Customer can check the status of his/her placed order.
9. **Trace Orders:** Admin can view the placed order and delivered orders.
10. **Log-out**: After the payment or surf the product the customer will log out.

VIII. CONCLUSION

Food Mart is a part of e-commerce. E-commerce or business through net means distributing, buying, selling, marketing, and servicing of products or services over electronic systems such as the Internet and other computer networks. Thus if we owner of the restaurant we need to upload menu online to connect with potential customers. The online food ordering system gives restaurants the ability to increase sales and expand their business by giving customers the facility to order food online. Through an online restaurant menu ordering system, customers have the flexibility to place orders at any time, around the clock, offering continuous accessibility for ordering food. Thus it is a simple, fast and convenient food ordering system giving an edge over the competition at an affordable price. The internet's massive growth makes having an online presence crucial. Using an Online Ordering System, we can put our restaurant menu online, allowing easy orders with a click. This system helps track orders, manage customer details, and enhance food delivery. Orders can come through emails, fax, or directly on the internet, streamlining our process.

IX. REFERENCES

[1] Domokos, Cristina-Edina, et al. "Net food: A Software System for Food Ordering and Delivery." 2018 IEEE 16th International Symposium on Intelligent System and Informatics (SISY). IEEE, 2018.

[2] Johnson, Rod, Juergen Hoeller, Keith Donald, Colin Sampaleanu, Rob Harrop, Thomas Risberg, Alef Arendsen et al. "The spring. framework-reference documentation." interface 21 (2004): 27.

[3] Moiseev, Anton, and Yakov Fain. Angular Development with TypeScript. Simon and Schuster, 2018.

[4] Http Client, [Online], Available: https://angular.io/guide/http.

[5] Alex, B., Taylor, L., Winch, R., Hillert, G., Grandja, J., & Bryant, J. (2004). Spring security. reference. URL https://docs.spring.io/springsecurity/site/docs/current/reference/htmlsingle/.[utoljáramegtekintve: 2017. 04. 21.], 12.

[6] Ladd, Seth, Darren Davison, Steven Devijver, Colin Yates, Rob Harrop, and Keith Donald. Expert spring MVC and web flow. Vol. 1. Berkeley, CA: Apress, 2006.