

ONLINE FOOD DELIVERY SYSTEM

AN INTERNSHIP REPORT

Submitted by

PATEL DEVEN PRAVINKUMAR

190320116036

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

Information Technology Department

L. J. Institute of Engineering & Technology

Ahmedabad



Gujarat Technological University, Ahmedabad

[May - 2023]



L. J. Institute of Engineering & Technology
Ahmedabad

CERTIFICATE

This is to certify that the project report submitted along with the project entitled **ONLINE FOOD DELIVERY SYSTEM** has been carried out by **PATEL DEVEN PRAVINKUMAR** under my guidance in partial fulfillment for the degree of Bachelor of Engineering in **Information Technology**, 8th Semester of Gujarat Technological University, Ahmadabad during the academic year 2022-23.

Prof. Foram Gohel

Internal Guide

Prof. Prayag Patel

Head of the Department



Enterprise Analytic
BUSINESS SOLUTIONS

211, Maruti Titanium, S.P. Ring Road,
Nikol, Ahmedabad - 382350, Gujarat
+91 9714844045
www.enterpriseanalytic.com
info@enterpriseanalytic.com

Date : 21 April, 2023

TO WHOM IT MAY CONCERN

This is to certify that PATEL DEVEN PRAVINKUMAR, a student of L. J. Institute of Engineering & Technology, Ahmedabad has successfully completed his internship in the field of Web Development from 30/01/2023 to 21/04/2023 (Total number of Weeks:12) under the guidance of Mr. Sandip Patel.

The intern is involved in developing Web applications as part of his internship activities.

During the period of her internship program with us, he had been exposed to different processes and was found diligent, hardworking and inquisitive.

We wish him every success in his life and career.

Sandip Patel

Co- Founder

Enterprise Analytic LLP

Email: sandip@enterpriseanalytic.com

Signing Authority

FOR ENTERPRISE ANALYTIC
FOR, ENTERPRISE ANALYTIC

PARTNER



L. J. Institute of Engineering & Technology

Ahmedabad

DECLARATION

We hereby declare that the Internship report submitted along with the Internship entitled **ONLINE FOOD DELIVERY SYSTEM** submitted in partial fulfillment for the degree of Bachelor of Engineering in **Information Technology** to Gujarat Technological University, Ahmedabad, is a bonafide record of original project work carried out by me at Enterprise analytic LLP under the supervision of Sandip Patel and that no part of this report has been directly copied from any students' reports or taken from any other source, without providing due reference.

Name of the Student

Patel Deven Pravinkumar

Sign of Student

ACKNOWLEDGEMENT

I have taken efforts in this Internship. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am very grateful and would like to thank my supervisor and external guide **Mr. Sandip Patel** who has been mentoring me through the whole journey of this project and internship. I would also like to thank **Prof. Prayag Patel**, Head of Department and **Prof. Foram Gohel**, my Internal guide for their continued support.

Without their substantial support and valuable expertise, this internship and project could not have been accomplished. They consistently provided assistance whenever I required it. Despite my utmost care and profound interest in preparing this report, I acknowledge that it may have flaws and imperfections. This opportunity has been beneficial in terms of improving my coding abilities, increasing my comfort with code, and taking on responsibilities. I am grateful for this experience.

Patel Deven Pravinkumar

(190320116036)

DATE:

ABSTRACT

The utilization of information technology holds great potential to revolutionize industries and transform how businesses operate. Within the realm of business, the internet is utilized for a wide range of purposes, including the exchange of information, media promotion, electronic mail, mailing lists, dialogue and discussions, and online consultations with consumers. Two primary forms of electronic commerce applications exist, namely business-to-consumer and business-to-business commerce.

Despite the numerous benefits that information technology can provide, one significant obstacle is the substantial cost involved in establishing and maintaining an online network, as well as supplying the necessary devices. However, the implementation of information technology can play a pivotal role in promoting trade and contributing to national economic growth, ultimately benefiting the public welfare.

In today's rapidly evolving business landscape, the adoption of information technology has become increasingly crucial for companies seeking to remain competitive and thrive in their respective markets. By leveraging the power of technology, businesses can streamline their operations, improve communication and collaboration, and enhance their overall efficiency and productivity. As such, investing in information technology infrastructure and applications can be a valuable long-term strategy for driving growth and success.

LIST OF FIGURES

Fig 1.1	Turnover Center Product	1
Fig 1.2	Enterprise Recruit Product	1
Fig 1.3	School Pandit Product	1
Fig 1.4	E-commerce web chart	2
Fig 2.1	SDLC	4
Fig 2.2	Layout of process	5
Fig 3.1	Iterative Model	9
Fig 3.2	Gantt chart	15
Fig 4.1	Features of new system	20
Fig 4.2	List of main Modules	21
Fig 5.1	Main display of website	22
Fig 5.2	Login page	23
Fig 5.3	Signup page	23
Fig 5.4	Contact us page	23
Fig 5.5	Order details	24
Fig 5.6	Tracker	24
Fig 5.7	After Id password	25
Fig 5.8	Sequence Diagram	25
Fig 5.9	Backend of our system	26
Fig 5.10	Contact us backend	27
Fig 5.11	Order update	27
Fig 5.12	User's orders	28
Fig 5.13	Admin's products	28
Fig 5.14	DFD for admin	29
Fig 5.15	UML sequence diagram	30

Fig 5.16	Cart form	31
Fig 6.1	Navbar page	33
Fig 6.2	CSS page	33
Fig 6.3	JavaScript file for cart	34
Fig 6.4	jQuery for update cart	34
Fig 6.5	Models for backend	35
Fig 6.6	Admin side fields	35
Fig 6.7	Views file for redirection pages	36
Fig 6.8	URL path for all pages	36
Fig 6.9	Main page	37
Fig 6.10	End page	37
Fig 6.11	Categories page	38
Fig 6.12	Quick view	38
Fig 6.13	Cart option	38

LIST OF TABLES

List of figures.....	iii
List of tables.....	v
List of abbreviation.....	vi
Table of contents.....	vii
Table 3.1 Project plan.....	11
Table 3.2 Estimated lines of code.....	13
Table 4.1 Hardware requirement.....	17
Table 7.1 Test cases.....	41

LIST OF ABBREVIATION

<u>Abbreviation</u>	<u>Full Form</u>
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheet
JS	Java Script
DB SQLite	Database Structured Query Language
ERP	Enterprise Resource Planning
SDLC	System Development Life Cycle
SRS	Software Requirement Specification

TABLE OF CONTENTS

Acknowledgement	i
Abstract	ii
List of Figures	iii
List of Tables	v
List of Abbreviations	vi
Table of Contents	vii
Chapter 1 Overview of the Company	1
1.1 History	1
1.2 Different Products	1
1.3 Organization Chart	2
Chapter 2 Overview of different process in company	3
2.1 Details about the works in each section	3
2.2 Technical details of major equipment in each department	4
2.3 Sequence of steps for manufacturing product	4
2.4 Details about each stage of production	5
Chapter 3 Introduction	6
3.1 Introduction of project	6
3.2 Purpose	6
3.3 Objectives	7
3.4 Tools and technologies	7

3.5 Project Scope	7
3.6 Description of technology	8
3.7 Functional Requirements	8
3.8 Non-Functional Requirements	10
3.8.1 Roles and Responsibilities	11
3.8.2 Time, Effort and cost analysis	12
3.8.3 Group Dependencies	14
3.9 Internship Scheduling	15
Chapter 4 System Analysis	16
4.1 Study Of Current System	16
4.2 Problem and Weakness of Current System	16
4.3 Requirements of New System	16
4.4 Hardware and Software	17
4.5 System Feasibility	18
4.5.1 System's contribution to the overall objectives of the organization	19
4.5.2 System's implementation using the current system	24
4.5.3 System's integration with other systems	20
4.6 Features of new system	20
4.7 List of main modules	21
4.8 Selection approaches and justification	21
Chapter 5 System Design	22
5.1 System Design and Methodology	22
5.2 Database Design	26
5.3 Input/ Output and Interface Design	30

5.3.1 UML sequence diagram	30
5.3.2 Samples of forms	31
Chapter 6 Implementation	32
6.1 Implementation Platform/ Environment	32
6.2 Program/Technology	32
6.3 Findings/Results/Outcomes	37
6.4 Result Analysis	39
Chapter 7 Testing	40
7.1 Testing Plan/ Strategy	40
7.2 Test Results and Analysis	41
Chapter 8 Conclusions and Discussion	42
8.1 Summary of Internship/ Project Work	42
8.2 Limitation and Future Enhancement	42
References	43

1. OVERVIEW OF THE COMPANY

1.1 HISTORY

Enterprise Analytic is a Technology solution provider specializing in the following solutions: Our aim is to analyse the specific needs of our customers and provide quality solutions on time and within budget. We leverage Agile and Lean principles for quick go-to-market and Return on Investment.

We provide business value with a clear focus on quality and long-term relationships. Our solutions are focused on meeting your strategic business objectives. Our delivery process encompasses industry best practices and proven quality frameworks.

1.2 DIFFERENT PRODUCTS



Figure 1.1 Turnover Center Product



Figure 1.2 Enterprise Recruit Product



Figure 1.3 School Pandit Product

1.3 Organization Chart



Figure 1.4 E-commerce web chart

Company Site: <https://enterpriseanalytic.com/>

2. OVERVIEW OF DIFFERENT PROCESS IN COMPANY

2.1 DETAILS ABOUT THE WORK IN EACH SECTIONS

- **Project Manager:**

Project managers have the responsibility of planning, organizing, and motivating employees. They ensure customer satisfaction, push for documentation, and optimize the workflow for efficient project delivery.

- **Business Analyst:**

Business analysts assist organizations by gathering, analyzing, developing, and documenting business requirements. They act as a liaison between the business and IT to improve overall efficiency.

- **Quality Assurance engineers:**

Quality assurance engineers are responsible for advocating the quality of the product by monitoring each stage of software development, debugging, and defining corrective measures. They do more than just testing features or products.

- **Designer:**

UI/UX designers focus on creating user-friendly and intuitive interactions by building a route that logically flows from one stage to the next. They are also responsible for designing the visual experience of the user interface.

- **Developer:**

Software developers are the creative, brainstorming eminence grise that power computer programs of all kinds. Among their daily duties are software development and testing, upgrading, quality monitoring, and documenting all processes for future reference.

2.2 A LIST OF THE TECHNICAL DETAILS AND SPECIFICATIONS FOR THE PRIMARY EQUIPMENT USED IN EACH DEPARTMENT.

List of major equipment used in each department individually is not provided to us by the company.

Hence, I have specified the equipment I used as a software developer intern:

Back end and Frontend coding: Visual Studio 2019

Framework and library: Django framework from Visual Studio Libraries

Database: DB SQLite, AdminLTTE3

2.3 THE SEQUENTIAL STEPS INVOLVED IN THE MANUFACTURING PROCESS OF THE FINAL PRODUCT.

The company utilizes the standard Software Development Life Cycle (SDLC) approach to software development as it is proven to be the most effective. Therefore, software development is conducted following the basic SDLC rule.

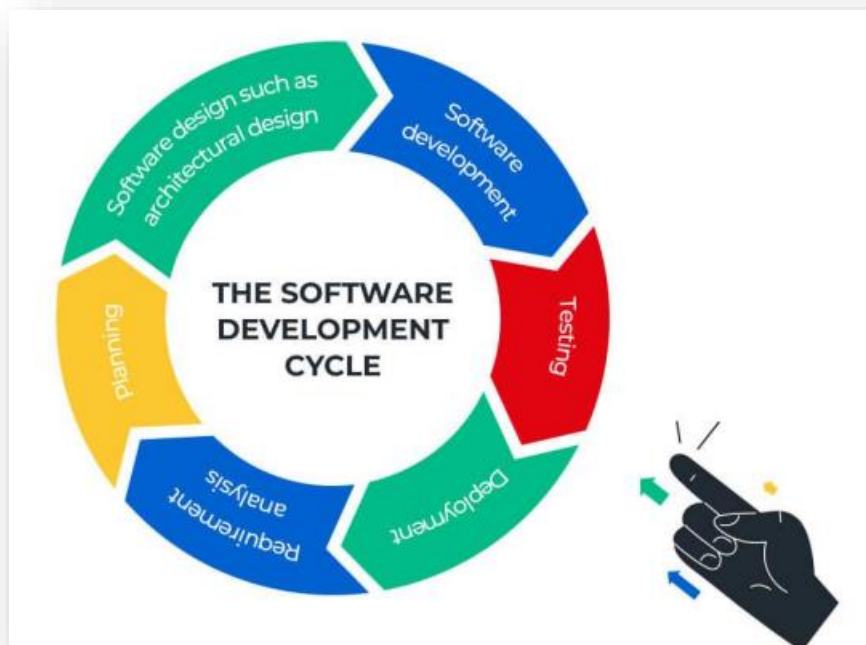


Figure 2.1 SDLC

Schematic or sequence of operation to produce a software:

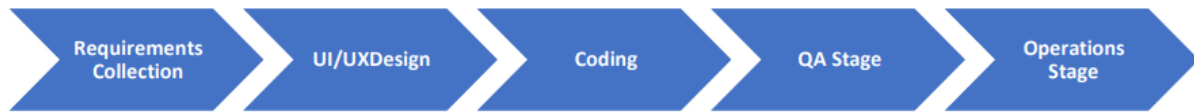


Figure 2.2 Layout of Process

The above given process is a schematic representation of the sequence of stages that the software development team undertakes to produce a software. It is basically a SDLC with a use of differently chosen words.

<https://www.uptech.team/blog/product-development-life-cycle>

2.4 DETAILS ABOUT EACH STAGE OF PRODUCTION

1. Requirement Gathering:

At the start of any software development project, it is crucial to gather clear and comprehensive functional and non-functional requirements. The project manager is responsible for communicating these requirements effectively to the development and design team.

2. Front-end Design:

The UI/UX designers of the company are responsible for developing the visual solution of the product once the design and user interface requirements are understood. This step is called user interface design, which focuses on how users perceive the app.

3. Coding:

In this step, the development team implements all the models, business logic, and service integrations specified in the prior stages. The coding stage is the most extended stage and serves as the foundation of the entire process. The development team follows the software requirements to build a product that meets the stakeholder's expectations.

4. Quality Assurance Review:

The testing specialists review the product during the Quality Assurance (QA) stage and identify any issues that arise while using the system. The testing frameworks may vary depending on the project, including automation and/or manual testing.

5. Deployment Stage:

Finally, the application is deployed in a live environment. This stage involves deployment, support, and maintenance necessary to keep the system functional and up-to-date.

3. INTRODUCTION

3.1 INTRODUCTION OF PROJECT

Smart Restaurant Management System is a new generation of restaurant management software. When users/customer will enter in the website, he/she should have an account. If user does not have an account, user has to create a new account to order food. To create a new account user should enter unique username, email and new mobile no. with password. User fill his/her address for food delivery. Once user enters in the website, you can see different types of food available in restaurant. First select category of food from soups, starters, the main course dishes and desserts. After that search food as your interest. select food you want to order. After selecting all your meal place your order and confirm your address. Then website will saw you various type of payment methods and your total bill amount. You can pay cash on delivery or there are many more options for online payment to get benefits. online payment methods:

- Credit/Debit card payments
- Bank transfers
- E-Wallets
- UPI payments

You can choose best deal for your meal.

3.2 PURPOSE

The reason for choosing to work with Mr. Sandip Patel was his extensive experience of more than 10 years in the field and his outstanding track record of managing over 30 clients presently. The primary objective of working on this project was to apply the knowledge gained during the internship period to practical use. I aimed to persuade the client to adopt a user-friendly system for managing inventory and company records, especially during the current pandemic situation where online options have become the preferred choice for almost everything. It has become increasingly challenging to find skilled employees to manage organizational operations. This website offers an effective solution by not only saving time but also providing multiple options to simplify work. The system is capable of storing large amounts of data, making it easy to store and verify data later, and can be easily understood and operated by non-skilled employees.

3.3 OBJECTIVES

- To provide immediate help with ease to the Clients
- To save with easy-to-understand system.
- To provide user with various choice features like saving data, updating data, saving details of all employees etc.
- To let user, know, the current balance while recording transaction details.
- To store delivery and reception of stocks.
- To allow only the admin to log in.

3.4 TOOLS AND TECHNOLOGIES

Technologies:

- Django
- Python
- Bootstrap
- HTML
- CSS

Tools:

- Git
- Visual Studio Code
- PyCharm

3.5 PROJECT SCOPE

The project scope for a food delivery website would involve designing and developing a platform that allows customers to place food orders online and have them delivered to their location. The website would need to have an intuitive user interface, robust database for menu items and customer information, secure payment gateway, and efficient order tracking system. Additionally, the scope would include creating a mobile-responsive design to cater to users on different devices. The website would also require integration with third-party services such as map APIs for delivery tracking, and SMS or email notifications for order updates. Overall, the project aims to provide a seamless and convenient food ordering experience for customers.

3.6 DESCRIPTION OF TECHNOLOGY

I. HTML and CSS:

They are essential for creating web pages. HTML provides the basic structure for web pages, allowing you to add text, images, and videos. On the other hand, CSS is used for styling and formatting HTML pages. To create the web application's structure, I combined HTML with ASP Components and CSS.

II. JavaScript:

It is a lightweight, cross-platform, and interpreted scripting language used for both client-side and server-side development. Meanwhile, jQuery is an open-source JavaScript library that simplifies interactions between an HTML/CSS document and JavaScript, specifically the Document Object Model (DOM). I incorporated JavaScript and jQuery, specifically on some components, for client-side scripting in the project.

III. DB SQLite:

SQLite is a free and open-source relational database management system that uses a file-based approach. It is self-contained, serverless, and supports SQL syntax. It is a popular choice for applications that need to store and retrieve data on a device or in a small-scale system, such as mobile apps or embedded systems.

3.7 FUNCTIONAL REQUIREMENTS:

Customers:

1. Sign Up (only for new customer)

Input: "Signup" option selected.

Output: customer prompted to enter the details.

2. Login

Input: "Login" option selected.

Output: customer prompted to enter the username and password.

3. Forgot password

Input: "forgot password" option selected.

Output: customer prompted to enter the email and new password.

4. Select food items

State: The customer has logged in and the main menu has been displayed.

Input: Items are selected customer feel free to order.

Output: System will display selected items.

5. Changes to order

Input: “go to cart” option selected.

Output: customer can delete or add food item in order.

6. Review the order before submitting

Input: “Order Place” option selected.

State: Customer name, phone number, location (address) display or enter the all of information.

Output: customer prompted to pay the bill.

7. Payment

State: The different types of payment method are display.

Input: choose any payment method.

Output: customer prompted to enter the verification code if choose online payment.

State: Display order no., payment details and confirmation of delivery.

8. Logout

Input: “Logout” option selected.

Output: you are successfully logout.

State: System display login page.

Administrators:

1. Login (admin login page)

Input: “Login” option selected.

Output: admin prompted to enter the username and password.

2. Logout

Input: “Logout” option selected.

Output: you are successfully logout.

State: System display login page.

3.8 NON-FUNCTIONAL REQUIREMENTS:

1. Portability:

System running on one platform can easily be converted to run on another platform.

2. Reliability:

The ability of the system to behave consistently in a user-acceptable manner when operating within the environment for which the system was intended.

3. Availability:

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs.

4. Maintainability:

A commercial database is used for maintaining the database and the application server takes care of the site.

5. Security:

Secure access of confidential data (customer information).

6. User friendly:

System should be easily used by the customer.

7. Performance:

Performance should be fast.

8. Efficient:

System should be efficient that it won't get hang if heavy traffic of order is placed.

9. Safety:

Data in the database of system should not loss or damage.

10. Privacy:

Personal data of the system should not disclose to anyone.

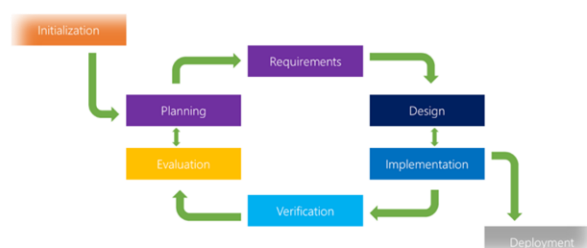


Figure 3.1 Iterative Model

Table3.1 Project Plan

TASKS	Start Date	End Date
Introduction to the company system	30/01/2023	03/02/2023
Introduction to frontend languages	06/02/2023	10/02/2023
Web page design and JavaScript	13/02/2023	17/02/2023
Structural designing of User Interface and CSS application	20/02/2023	24/02/2023
Understanding the Python basics	27/02/2023	03/03/2023
Understanding Django Framework components like pip, pillow	06/03/2023	10/03/2023
Starting off with Backend programming using DB SQLite	13/03/2023	17/03/2023
Learnt how to Database Integration and SMTP authentications	20/03/2023	24/03/2023
Connect navigation bar to backend	27/03/2023	31/03/2023
Understanding Requirements and programming for Transaction	03/04/2023	07/04/2023
Product editing from admin side	10/04/2023	14/04/2023
AdminLTE3 introductions and implementation	17/04/2023	21/04/2023

3.8.1 Roles and Responsibilities

During my internship, I had various duties and obligations which included:

- Communicating
- Generating reports
- Analyzing requirements
- Establishing connections to databases
- Writing code
- Conducting testing.

3.8.2 Time, Effort and Cost analysis

COCOMO Model

- Like all estimation models for software, the COCOMO models require sizing information.
- Three different sizing options are available as part of the model hierarchy: object points, function points, and lines of source code.
- Like function points, the object point is in direct software that is computed using counts of the number of

1. Screens (at the user interface),
2. Reports, Components likely to be required to build the application.
3. There are three types of software project:
 - Organic project
 - Semi-detached project
 - Embedded project

Cost required to develop project= effort*RS/month

- **Effort Estimation(E):**
 - In Organic= $2.4 (KLOC)^{1.05}PM$
 - In semidetached= $3.0(KLOC)^{1.12}PM$
 - In Embedded= $3.6(KLOC)^{1.20}PM$
- **Duration Estimation(D):**
 - In Organic= $2.5(effort)^{0.38}months$
 - In semidetached= $2.5(effort)^{0.35}months$
 - In Embedded= $2.5((effort)^{0.32}months$
- **Person Estimation:**
 1. $P=E/D$ **KLOC=Kilo Line of Code**

Table 3.2 Estimated Lines of code

Apps	Page	Estimated lines of code
Shop	-	1989
	About	308
	Basic	234
	Checkout	183
	Contact	300
	Index	178
	Order view	234
	Password reset	129
	Payment Status	22
	Product view	63
	Search	185
	Tracker	153
Admin		962
	Index	453
	Settings.py	150
	Views.py	259
	Models.py	60
	Urls.py	40
CSS		10241
	Admin style	100
	Style	1533
	Animate	4072
	Bootstrap-grid	3871
	Bootstrap-reboot	325
	Bootstrap-reboot-min	340

JS	-	4904
	Main	198
	Bootstrap	4432
	jQuery-sticky	274

- Total line of code=18,096 (actual value is more than calculated figure)
- KLOC=18.09

Effort Estimation(E):

$$=2.4(\text{KLOC})^{1.05}\text{PM}$$

$$=50.18\text{PM}$$

Duration Estimation(D):

$$=2.5(\text{effort})^{0.38}\text{month}$$

$$=2.5(50.18)^{0.38}\text{months}$$

$$\sim 11 \text{ months}$$

Project Cost:

$$=\text{effort}*\text{RS/month}$$

$$=50.18*\text{RS/month}$$

3.8.3 Group Dependencies

Two individuals, including myself and my colleague Jaykishansinh, collaborated on a joint project during our internship. Jaykishansinh was responsible for communication as the system connectivity was set up on her desktop, while we both worked collaboratively on the design, coding, and testing aspects of the project.

3.9 INTERNSHIP SCHEDULING (GANTT CHART)

The weekly work done is shown in the GANTT chat shown below:

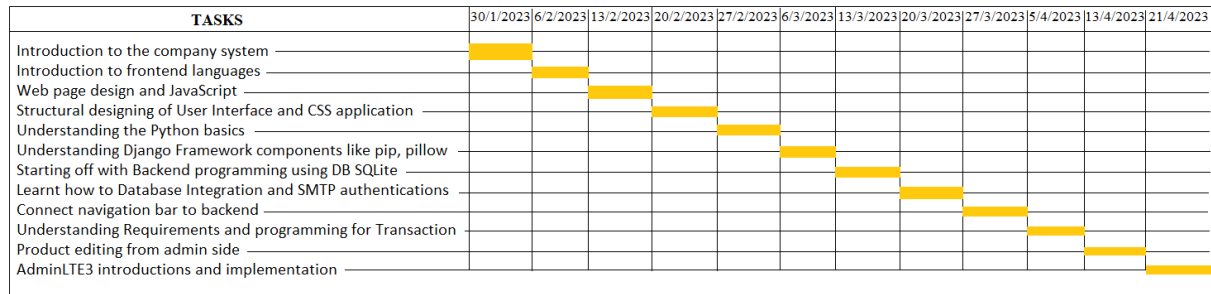


Figure 3.2 Gantt Chart

4. SYSTEM ANALYSIS

4.1 STUDY OF CURRENT SYSTEM

The study of the current system of a food website involves a thorough analysis of its existing structure, features, and functionality. This process is aimed at identifying the strengths and weaknesses of the website's current system, as well as any areas that may require improvement. It also entails conducting user research and gathering feedback to understand user behaviour and preferences. By studying the current system of a food website, one can identify opportunities for optimization and innovation, which can lead to an improved user experience, increased traffic and engagement, and ultimately, higher revenue. The insights gained from this study can also inform the development of a roadmap for future enhancements to the website.

4.2 PROBLEM AND WEAKNESSES OF CURRENT SYSTEM

The current system of a food website may have several problems and weaknesses that need to be addressed to improve its overall performance. These issues can range from poor user experience to technical problems, content-related challenges, and more. For instance, the website may have slow loading times, inadequate navigation, unappealing design, outdated or irrelevant content, or limited functionality. Other problems may include security vulnerabilities, poor search engine optimization (SEO), or difficulties in managing user accounts and interactions. By identifying and addressing these problems and weaknesses of the current system, the website can be improved to meet user expectations, enhance engagement, and achieve business goals.

4.3 REQUIREMENTS OF NEW SYSTEM

Developing a new system for a food website requires a comprehensive understanding of the business goals and user needs. The new system should address the problems and weaknesses of the current system while providing a better user experience and meeting the website's objectives. Key requirements for the new system may include improved website performance, faster loading times, mobile responsiveness, and a user-friendly interface. It should also include enhanced features for easy navigation, effective search functionality, engaging multimedia content, and social media integration. Furthermore, the new system should be secure, easy to manage, and have the ability to integrate with other systems such as payment gateways or CRM tools.

User Requirement:

- It describes the type of user which deals with the applications. Basically, this application has one type of user as given below:
 1. Administrator

1. Administrator:

- Responsibility of administrator is to manage the application database and update the data in database regularly .For e.g. manage the data of booked tours and manage cancellation.
- User who uses this portal should know how to operate the windows. Because the software has the same look and features like whole software is menu driven.
- Just click navigation bar and corresponding page will open up.

4.4 Hardware and Software:

Hardware requirements

Component	Minimum	Recommended
Processor	1.9 gigahertz (GHz) x86- or x64-bit dual core processor with SSE2 instruction set	3.3 gigahertz (GHz) or faster 64-bit dual core processor with SSE2 instruction set
Memory	2-GB RAM	4-GB RAM or more
Display	Super VGA with a resolution of 1024 x 768	Super VGA with a resolution of 1024 x 768

Table 4.1 Hardware requirement

Software Requirements:

- Editor Required: Visual Studio 2019
- Database Server: DB SQLite server 2008 R2
- Install Django Framework in the system through visual studio installer
- Install Libraries from VS installer

4.5 SYSTEM FEASIBILITY:

The term "feasibility study" refers to an analysis of the software product's practical value to the organization. The purpose of the feasibility study is to determine whether the product development and implementation will be beneficial to the organization. The feasibility study is conducted for various reasons, including assessing the suitability of the software product for development, implementation, and determining the project's contribution to the organization. The study involves evaluating the product's feasibility in terms of technical, economic, legal, operational, and scheduling aspects. By analyzing these factors, the feasibility study determines whether the software product is viable and worthwhile for the organization.

Types of Feasibility Analysis:

- Technical Feasibility
- Operational Feasibility
- Economic Feasibility
- Scheduling Feasibility

Technical Feasibility:

In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyzes technical skills and capabilities of technical team, existing technology can be used or not, maintenance and up-gradation is easy or not for chosen technology etc. Hence, the system should be maintained from time to time.

Operational Feasibility:

The System is to be developed for any user who wants to use it. We want our system user friendly and easy to use. The administrator also may be non-technical, so the user interface will be designed in such a way that it gets comfortable for non-technical person to operate easily.

Economic Feasibility:

In Economic Feasibility study cost and benefit of the project is analyzed. Means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analyzed whether project will be beneficial in terms of finance for organization or not. For the system to be economically feasible, the profits from the project must be greater than or equal the cost of development.

Scheduling Feasibility:

The project was quite feasible when it comes to scheduling. Routine reporting as done for the progress of the project.

4.5.1 System's contribution to the overall objectives of the organization

Upon starting my internship, I was tasked with developing a web application for a system that had already been created as a desktop application. This system was developed to meet the requirements of the organization and aimed to create personalized systems for clients. The web application version was designed to fulfill the same objectives as the desktop application, which was developed by the company itself. The system was expected to be fully functional and meet all the organizational objectives by providing customized solutions for clients.

4.5.2 System's implementation using the current system

The new system was developed by using the current system as a reference point. While the new system was built from scratch, the development process was informed by analysing the current system. The company provided the database, but the flow of data from one page to another table was personally studied. The current system was a valuable resource while creating the new system, as it helped identify areas where the current system was falling short

and where improvements could be made to meet the organization's requirements. By studying the current system, the team was able to create a new system that addressed the shortcomings of the previous system and provided improved functionality.

4.5.3 System's integration with other systems

System integration refers to the incorporation of the provided desktop application into the new system. In the future, the new system can be integrated with other systems to allow for multiple departmental inventory management at once. Users can choose from a range of options from different companies. For instance, if a client manages data for multiple companies at once and wants to access the details of a specific company, they will have the option to select and log in to that company's data. The system's infrastructure for all companies will be the same, but the database ID will be different for each company to enable the storage and retrieval of data specific to that company.

4.6 FEATURES OF NEW SYSTEM

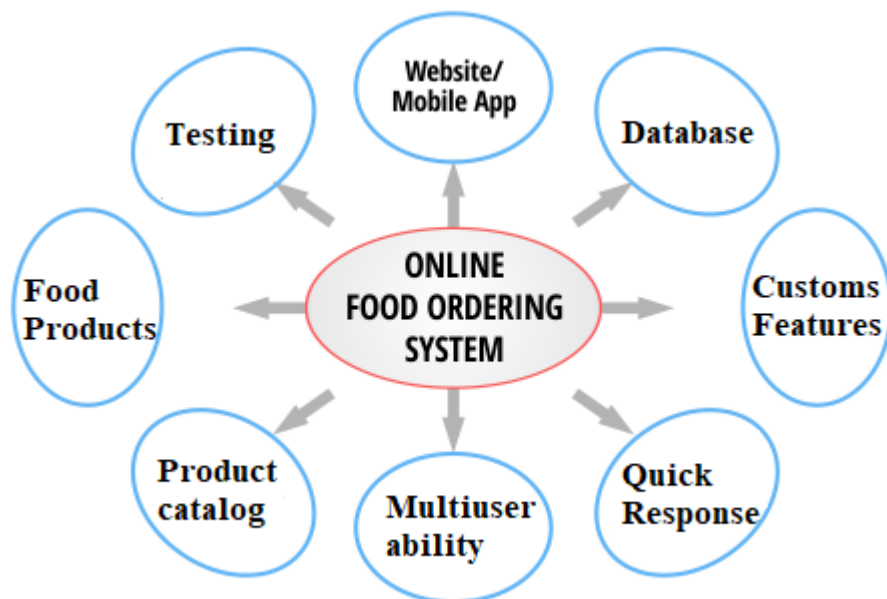


Figure 4.1 Features of new system

4.7 LIST OF MAIN MODULES

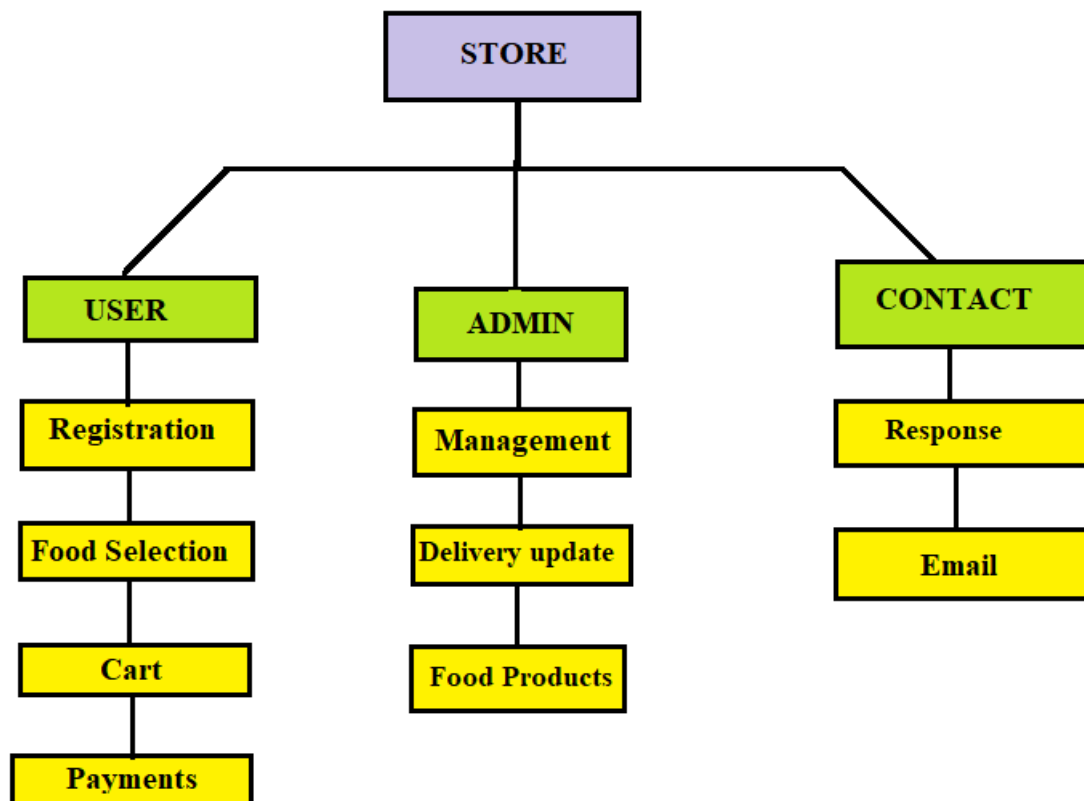


Figure 4.2 List of Main Modules

4.8 SELECTION APPROACHES AND JUSTIFICATION

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. It includes many built-in features for easy website development and maintenance.

Python is a popular and versatile programming language used for a wide range of tasks, from web development to scientific computing. It has a simple syntax and is known for its readability and ease of use.

DB SQLite is a lightweight and fast relational database management system that is widely used in embedded systems and small-scale applications. It is a self-contained database that requires no configuration or maintenance and is easy to use.

5.0 SYSTEM DESIGN

5.1 SYSTEM DESIGN AND METHODOLOGY

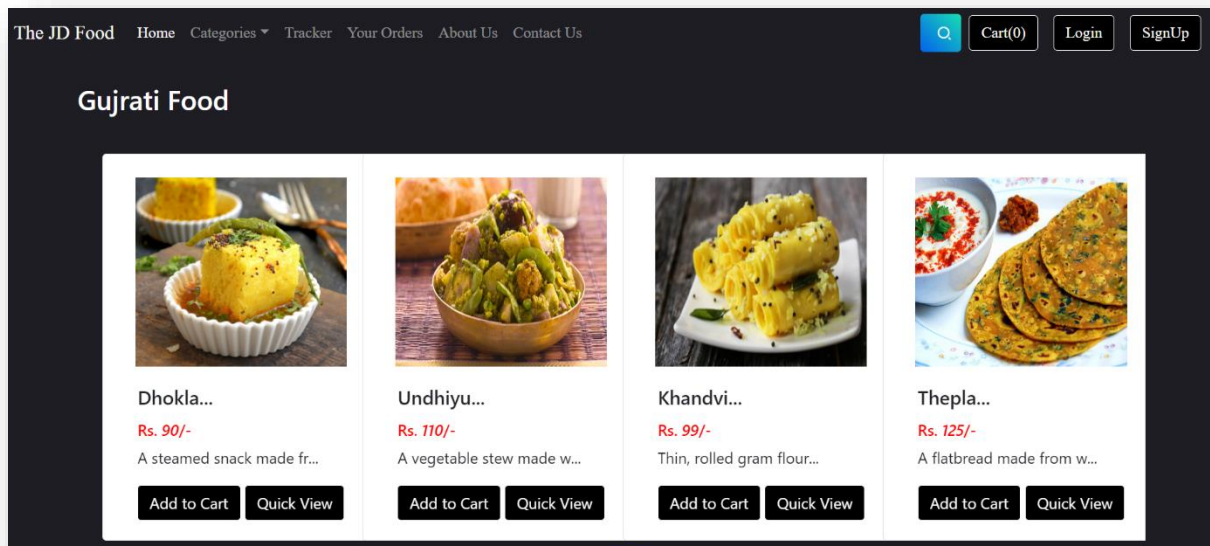


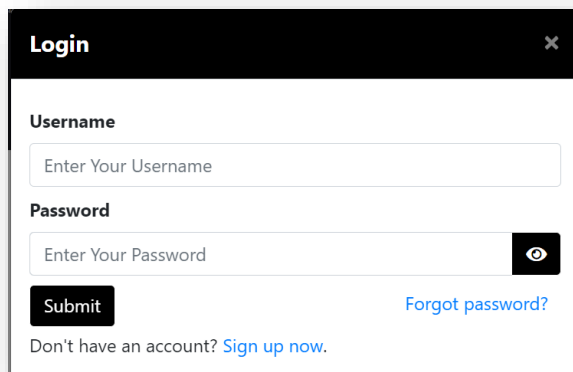
Figure 5.1 Main Display of website

This UI gives a various options in main display where user can register themselves and can add a food product in cart after that they can check their food status in tracker option which is displayed in above photo and check their food history in your order option.

Contact us option is for those user who have any query as well as want to give feedback. Search option is working on particular food as well as an particular category.


Moreover, it can be seen that the option of quick view will give a description of particular product, their price and brief knowledge of that item.

There are some other snapshots which describes more about our website



Login

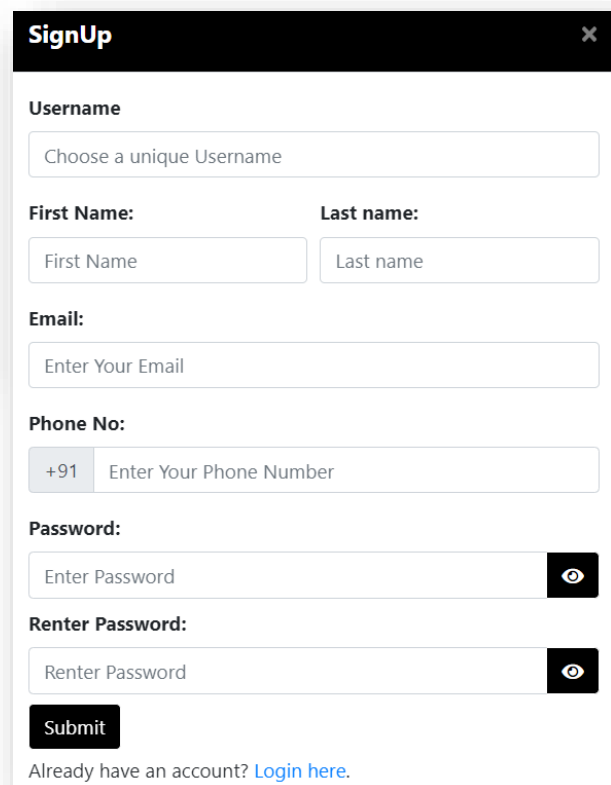
Username

Password
 

Submit [Forgot password?](#)

Don't have an account? [Sign up now.](#)

Figure 5.2 Login page




SignUp


Username

First Name: **Last name:**

Email:

Phone No:

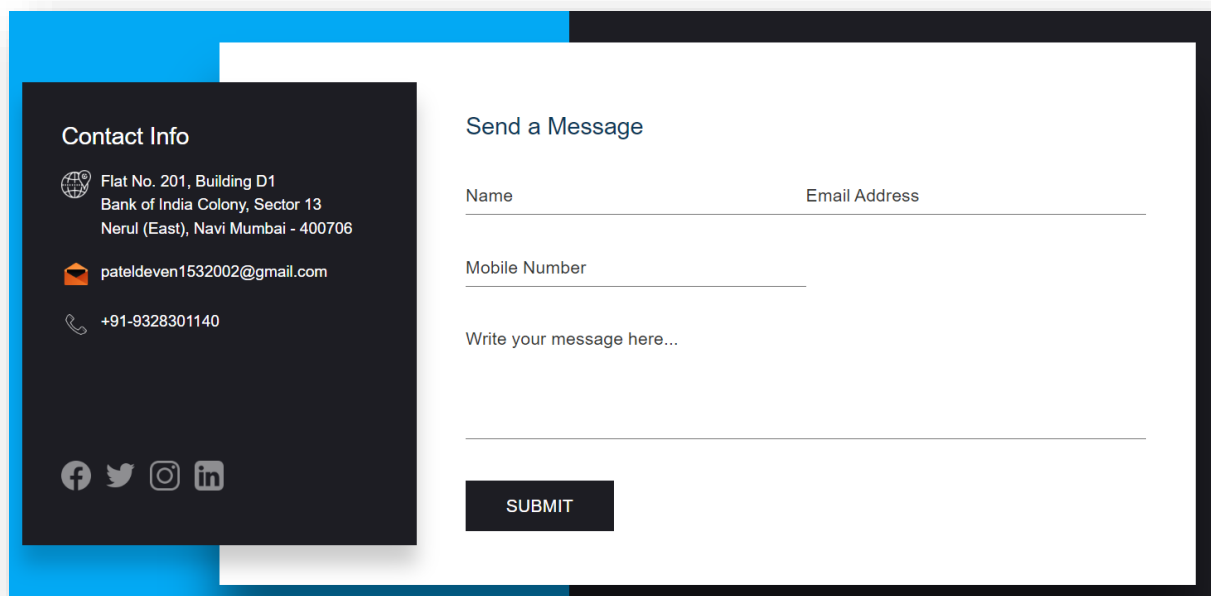
Password:
 

Renter Password:
 


Submit


Already have an account? [Login here.](#)


Figure 5.3 Signup page







Contact Info

 Flat No. 201, Building D1
 Bank of India Colony, Sector 13
 Nerul (East), Navi Mumbai - 400706

 pateldeven1532002@gmail.com

 +91-9328301140

Send a Message

Name **Email Address**

Mobile Number

Write your message here...

SUBMIT

Figure 5.4 Contact us page









Order Details								
						 Print	 Refresh List	
No.	Order Id	Name	Address	Email	Phone	Order Date	Amount	Items
1	55	Deven Patel	26,Arpan Bungalows ,Near ..	pateldeven1532002@gmail.com	9328301140	March 26, 2023, 12:04 p.m.	180	
2	56	Deven Patel	26,Arpan Bungalows ,Near ..	pateldeven1532002@gmail.com	9328301140	March 26, 2023, 2:59 p.m.	180	
3	57	Deven Patel	26,Arpan Bungalows ,Near ..	pateldeven1532002@gmail.com	9328301140	March 26, 2023, 3:01 p.m.	180	
4	58	Deven Patel	26,Arpan Bungalows ,Near ..	pateldeven1532002@gmail.com	9328301140	March 26, 2023, 3:24 p.m.	180	
5	59	Deven Patel	26,Arpan Bungalows ,Near ..	pateldeven1532002@gmail.com	9328301140	March 26, 2023, 4:24 p.m.	180	
6	60	Deven Patel	26,Arpan Bungalows ,Near ..	pateldeven1532002@gmail.com	9328301140	April 9, 2023, 2:52 p.m.	540	

Figure 5.5 Order details

This webpage displays a comprehensive record of your purchase history, providing a complete overview of the details from the date of purchase to the total cost. In addition, it also includes information about the total number of items purchased. You can also generate a receipt for your order, which allows you to obtain a printed copy of the bill. This receipt can serve as proof of purchase for future reference. By providing this detailed information, the webpage ensures transparency and facilitates tracking of your order history.

Enter your Order Id and Email address to track your order.

Order Id:

Email:

Password:

Order Id

Email

Enter Password

Track Order

Figure 5.6 Tracker

The Option where user can track his/her order is Tracker. Given photo shows before position of track. In this field user give their order id which is given after payment and email further more user have to fill password field with the original registered password.

Enter your Order Id and Email address to track your order.

Order Id: Email: Password:

Your Order Status

Delivered 4/16/2023, 3:41:26 PM

Your Order Details

Cheese N Corn Pizza... 3

Figure 5.7 After Id password

Upon filling out all the necessary fields accurately, the user will receive a comprehensive overview that includes information on the time of the order and the quantity of food ordered. Additionally, the user will be able to check the status of their order, whether it has been delivered or not. This feature adds an extra layer of convenience for users, enabling them to track their food delivery in real-time. By providing this level of detail, the system ensures a seamless user experience, improving customer satisfaction and increasing the likelihood of repeat orders. Overall, this feature enhances the user's sense of control and transparency over the ordering process, allowing them to have a hassle-free experience.

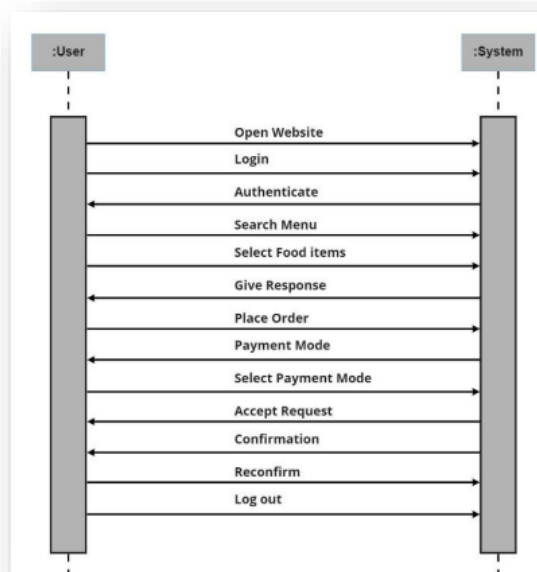


Figure 5.8 Sequence Diagram

5.2 DATABASE DESIGN

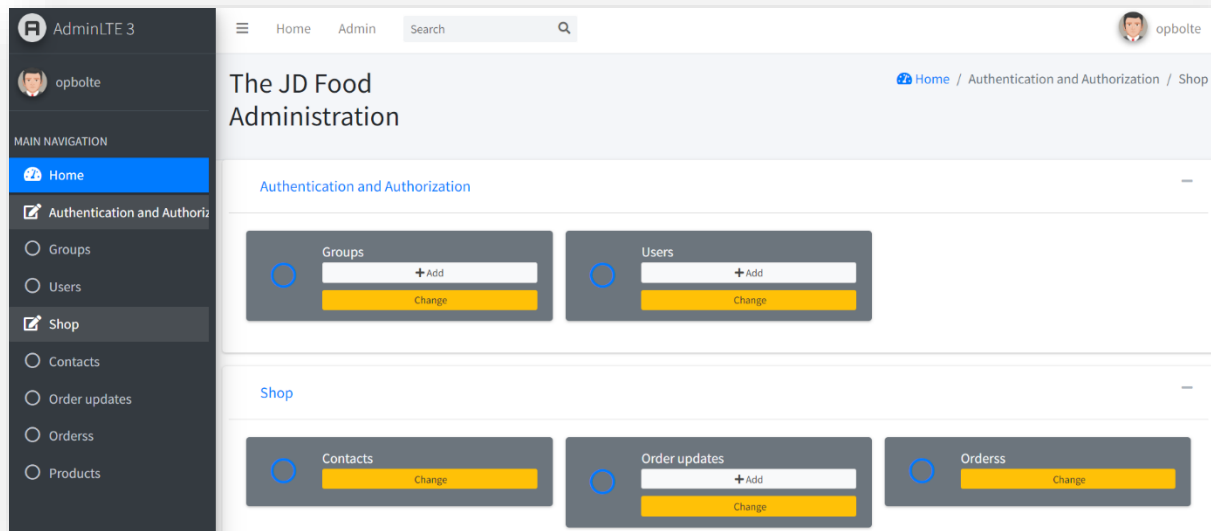


Figure 5.9 Backend of our system

Prior to using AdminLTE3, I used Django administration for managing my website. However, the design was basic and the user interface lacked interactivity. After switching to AdminLTE3, managing the system for admin became incredibly easy. The admin interface provided by AdminLTE3 is highly user-friendly, enabling the admin to handle products and user orders with ease. The intuitive design of the interface streamlines the management process, reducing the time and effort required for administration tasks. With the improved functionality and usability provided by AdminLTE3, managing the website has become a much smoother and more efficient process. Overall, the switch to AdminLTE3 has vastly improved the website management experience, enabling the admin to focus on other important aspects of running the website.

Select contact to view Home / Shop / Contacts

[Filter](#)

Search:

Name	Desc	Email	Timestamp
	sadasd	pateldeven1532002@gmail.com	April 16, 2023, 3:11 p.m.
Deven Patel	xdvSDVSDcSDCSDV	pateldeven1532002@gmail.com	April 16, 2023, 4:03 p.m.
Deven Patel	rtgrtg	pateldeven1532002@gmail.com	April 16, 2023, 3:12 p.m.
Deven Patel	Tremendous Food	pateldeven1532002@gmail.com	April 11, 2023, 5:47 p.m.
DevenPatel	dfhrtsgrdtg	pateldeven1532002@gmail.com	April 16, 2023, 3:15 p.m.
sasasas	sdsdsdsd	pateldeven1532002@gmail.com	April 16, 2023, 3:14 p.m.

Figure 5.10 Contact us backend

With the help of an efficient email management system, the admin can easily handle all user queries and concerns. This system enables the admin to receive and respond to emails from users, addressing their issues and providing solutions to their problems. By promptly responding to user queries, the admin can improve customer satisfaction and build trust with the user base. Moreover, the email management system can help streamline the communication process, reducing the time and effort required to manage user queries. With the ability to quickly respond to user issues, the admin can enhance the website's reputation and build a loyal user base. Overall, an effective email management system is crucial for successful website administration, enabling the admin to handle user queries and concerns in a timely and efficient manner.

Select order update to change Home / Shop / Order updates

[+ Add order update](#) [Filter](#)

Search:

Order id	Update desc	Timestamp
72	Delivered	April 16, 2023, 3:41 p.m.
71	-	April 16, 2023, 3:41 p.m.
70	Out for Delivery	April 9, 2023, 9:44 p.m.
69	Order Confirmed	April 9, 2023, 6:28 p.m.
68	Pending	April 9, 2023, 5:44 p.m.

Figure 5.11 Order update

Select orders to view [Home](#) / [Shop](#) / [Orders](#)

[Filter](#)

Action: 0 of 18 selected Search:

<input type="checkbox"/>	Order id	Userid	Name	Email	Timestamp
<input type="checkbox"/>	72	18	Deven Patel	pateldeven1532002@gmail.com	April 16, 2023, 3:41 p.m.
<input type="checkbox"/>	71	18	Deven Patel	pateldeven1532002@gmail.com	April 16, 2023, 3:41 p.m.
<input type="checkbox"/>	70	18	Deven Patel	pateldeven1532002@gmail.com	April 9, 2023, 9:44 p.m.
<input type="checkbox"/>	69	18	Deven Patel	pateldeven1532002@gmail.com	April 9, 2023, 6:28 p.m.
<input type="checkbox"/>	68	18	Deven Patel	pateldeven15032002@gmail.com	April 9, 2023, 5:44 p.m.
<input type="checkbox"/>	67	18	Deven Patel	pateldeven15032002@gmail.com	April 9, 2023, 4:18 p.m.
<input type="checkbox"/>	66	18	Deven Patel	pateldeven15032002@gmail.com	April 9, 2023, 4:17 p.m.

Figure 5.12 User's Orders

<input type="checkbox"/>	Fafda	Gujrati Food	240
<input type="checkbox"/>	Thepla	Gujrati Food	125
<input type="checkbox"/>	Khandvi	Gujrati Food	99
<input type="checkbox"/>	Undhiyu	Gujrati Food	110
<input type="checkbox"/>	Dhokla	Gujrati Food	90
<input type="checkbox"/>	Adai Dosa	Dosa	220
<input type="checkbox"/>	Pesarattu	Dosa	330
<input type="checkbox"/>	Set Dosa	Dosa	230
<input type="checkbox"/>	Rava Dosa	Dosa	199

Figure 5.13 Admin's products

The first image provides a comprehensive overview of each order, displaying all the necessary details such as the order number, date and time, customer details, and items ordered. This enables the admin to easily track each order and ensure that they are fulfilled accurately and promptly. The second image showcases the process of adding and removing products from the website, which can only be done by the super user. This feature ensures that only authorized individuals can make changes to the website's products, reducing the risk of errors or unauthorized modifications.

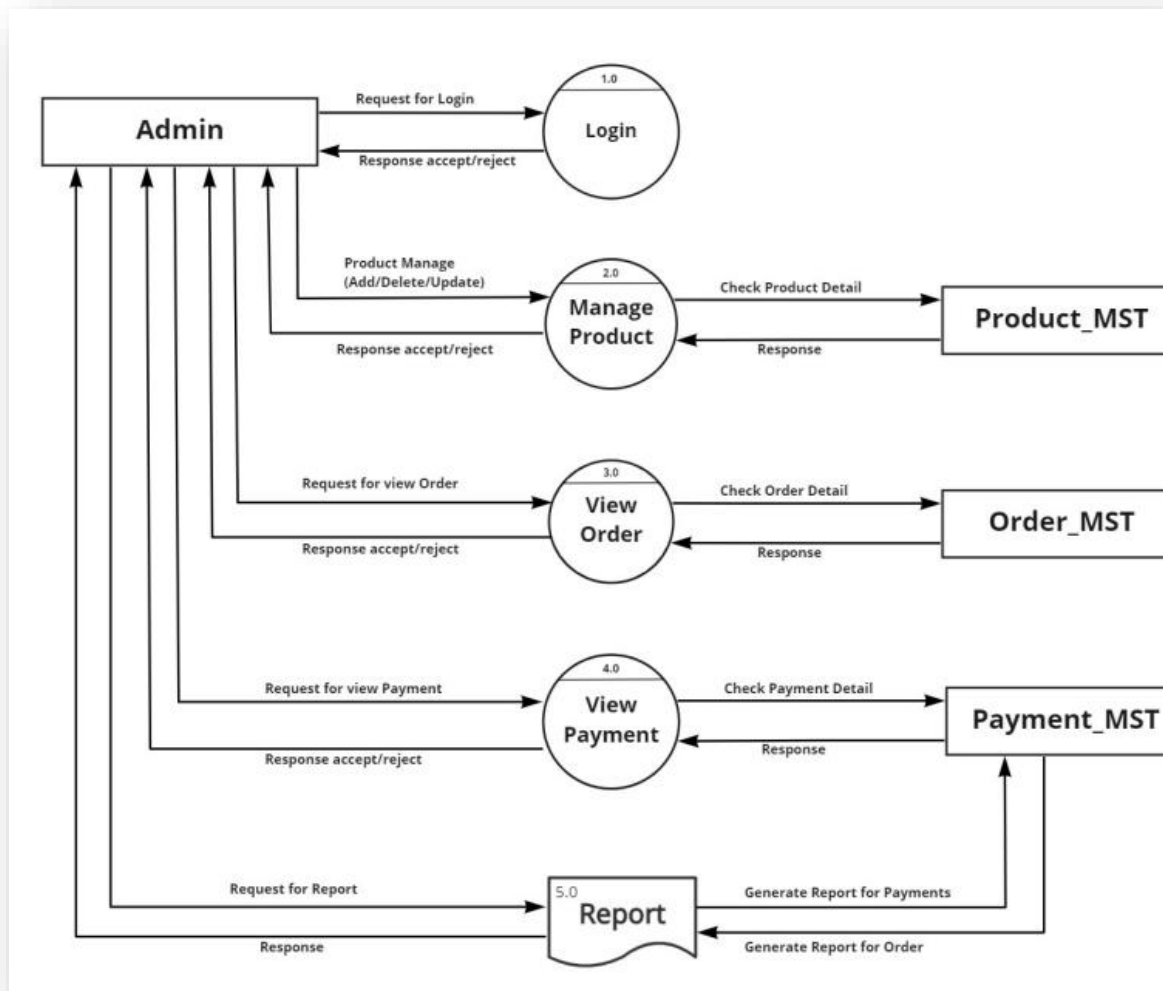


Figure 5.14 DFD for Admin

5.3 INPUT/ OUTPUT AND INTERFACE DESIGN

5.3.1 UML Sequence diagram

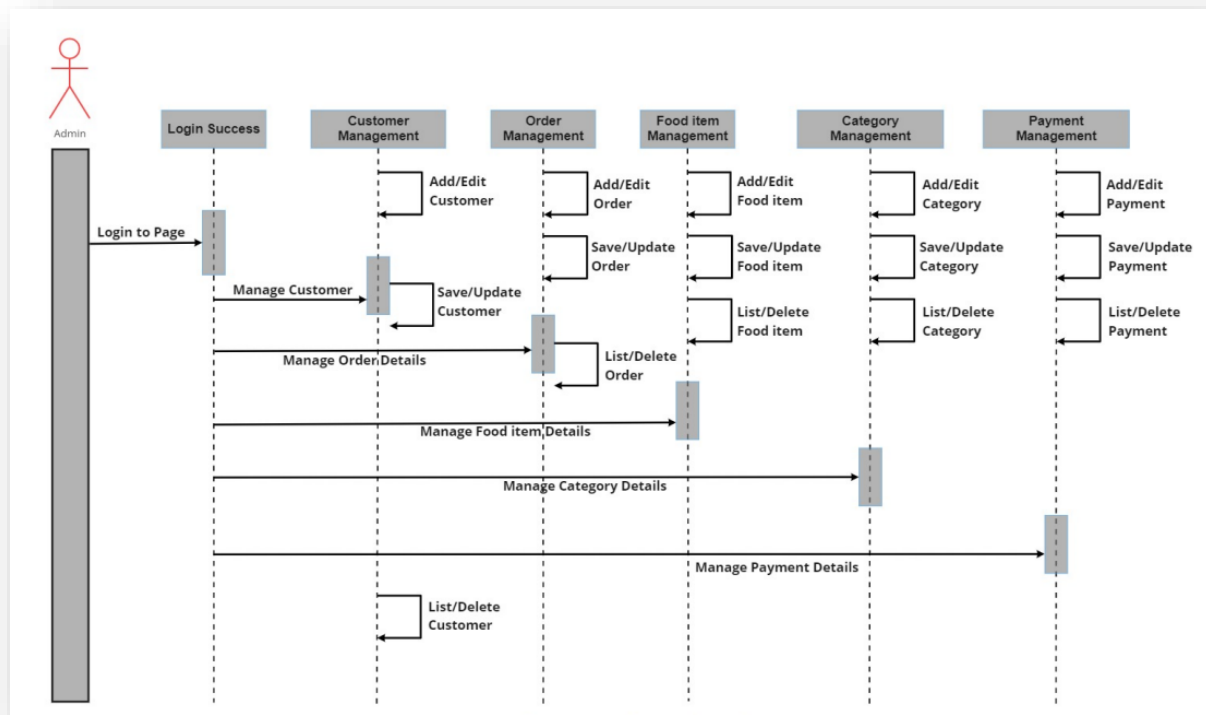


Figure 5.15 UML Sequence diagram

A UML sequence diagram is a powerful tool for illustrating the flow of events and interactions between different components or actors in a system. For a food delivery system, a sequence diagram can be used to show the interactions between the customers, the restaurant, and the delivery person.

At the start of the sequence, the customer places an order through the system, which is then forwarded to the restaurant for processing. Once the order is ready, the restaurant confirms the order and notifies the delivery person. The delivery person then picks up the order and delivers it to the customer. During this process, there may be additional interactions, such as the customer requesting updates on the status of their order or the delivery person confirming delivery.

5.3.2 Samples of Forms

Check your Food items and quantity.

1 : Khandvi... 1 x Rs. 99 = Rs. 99

Total Price: 99

Enter Address & Other Details

Name Email

Address

Address Line 2

City State Zip

Phone Number

Online Pay Cash On Delivery

Figure 5.16 Cart form

The users of the food delivery system are required to provide their delivery address on a form. Following this, they must choose between two payment options: cash on delivery or online payment.

The form provided to the user requests their delivery address, which is necessary to ensure the timely and accurate delivery of the food. The system offers two payment options, cash on delivery and online payment, to provide users with flexibility and convenience. Choosing cash on delivery allows users to pay in person when the food is delivered, while online payment options may include credit card or mobile payment methods.

By offering users these two payment options, the food delivery system aims to cater to the diverse needs and preferences of its users. The system's user-friendly interface and convenient payment options ensure a smooth and seamless food delivery experience for all users.

6. IMPLEMENTATION

6.1 IMPLEMENTATION PLATFORM/ ENVIRONMENT

After completing the system design, the next step was to implement the ideas, layout, features, and functionalities of the system based on the requirements. To achieve this, we used two main software tools: Microsoft Visual Studio 2019 and DB SQLite Server.

Microsoft Visual Studio - Visual Studio is an integrated development environment (IDE) that supports the development of various computer programs, websites, and web applications, among others. It has a code editor that supports code completion and refactoring.

DB SQLite Serve - DB SQLite Server is a software platform that provides comprehensive data management and business intelligence capabilities. It serves as the primary tool for storing and retrieving data as requested by other software applications. Hence, we used Visual Studio for coding and DB SQLite Server for managing the database of the system.

6.2 PROGRAM/ TECHNOLOGY

There were certain different technologies/ languages that were used in the development of the web application. We worked with languages like HTML, CSS, JAVASCRIPT, PYTHON, JQUERY.

Process Specifications:

At the start of our internship, we were introduced to programming languages such as Python and the Django framework, and were also trained on the software tools we would be using in the development process. Once we had received this basic training, we began applying our knowledge to practical applications. As the design, features, and functionalities of the web application had already been discussed and finalized during the earlier stages of the software development life cycle (SDLC), we started by creating a basic layout and framework for the application. This involved adding a menu bar, tool bar, and different menus to the system. We then proceeded to create all the webpages and forms that would appear in the various menus using HTML. We started with the Navbar menu and then moved on to other menus such as Transactions, Tracker, Order Status, Contact Us, and so on.

```

80     <div class="collapse navbar-collapse" id="navbarSupportedContent">
81         <ul class="navbar-nav mr-auto">
82             <li class="nav-item active">
83                 <a class="nav-link" href="/shop">Home <span class="sr-only">(current)</span></a>
84             </li>
85             <li class="nav-item dropdown" id="categoryList">
86
87             </li>
88             <li class="nav-item">
89                 <a class="nav-link" href="/shop/tracker">Tracker</a>
90             </li>
91             <li class="nav-item">
92                 <a class="nav-link" href="/shop/orderView/">Your Orders</a>
93             </li>
94             <li class="nav-item">
95                 <a class="nav-link" href="/shop/about">About Us</a>
96             </li>
97             <li class="nav-item">
98                 <a class="nav-link" href="/shop/contact">Contact Us</a>
99             </li>
100        </ul>

```

Figure 6.1 Navbar page

This HTML page describes the Navbar which contains the various kind of options such as home, tracker, Your order, About us, Contact us.

Once designing done we moves toward the CSS and make it good designing

```

11     <style>
12         .popover-header {
13             background-color: #dfd7d7;
14         }
15         .btn{
16             background-color:black;
17             border-color:#fff;
18             color:#fff
19         }
20         .btn:hover{
21             background-color:#fcaf2d;
22             color:black;
23             border-color:black;
24         }
25     }
26     body{
27         background-color:#1d1d23;
28     }
29     .search {
30         display: inline-flex;
31         align-items: center;
32         background-image: linear-gradient(45deg, #0561ee, #18e0b5);
33         color: #fff;
34         padding: 7px;
35         border-radius: 4px;
36     }

```

Figure 6.2 CSS page

Once we had completed the system design and styling, we proceeded to implement necessary field validation based on the system requirements. This involved applying various types of validation, such as making certain fields required, providing error messages for missing or incorrect inputs, restricting input length or data type, and so on. To achieve this, we used programming languages like JavaScript and jQuery. Below are some code snippets that illustrate the use of JavaScript and jQuery for validation purposes.

```

110 function updatePopover(cart) {
111     var popStr = "";
112     popStr = popStr + "<div class='mx-2 my-2'>";
113     var i = 1;
114     var j = 0;
115     for(var item in cart) {
116         popStr = popStr + "<b>" + i + "</b>. ";
117         popStr = popStr + document.getElementById('name' + item).innerHTML.slice(0, 15) + "... (Qty: " + cart[item][0] + ')<br>';
118         i = i + 1;
119         j = j + 1;
120     }
121     if(j == 0) {
122         popStr = popStr + "<p><b> No item available in your cart </b></p><div class='mx-2 my-2'>";
123         popStr = popStr + "</div> <a href='/shop'><button class='btn btn-primary'>Add items</button></a> ";
124     }
125     else{
126         popStr = popStr + "</div> <a href='/shop/checkout'><button class='btn btn-primary' id='checkout'>Checkout</button></a> <a href="
127     }
128     document.getElementById("popcart").setAttribute('data-content', popStr);

```

Figure 6.3 JavaScript file for cart

```

93  $('.divpr').on('click', 'button.cart', function() {
94      var idstr = this.id.toString();
95      if (cart[idstr] != undefined) {
96          qty = cart[idstr][0] + 1;
97      } else {
98          qty = 1;
99          name = document.getElementById('name'+idstr).innerHTML;
100         price = document.getElementById('price'+idstr).innerHTML;
101         cart[idstr] = [qty, name, parseInt(price)];
102     }
103     updateCart(cart);
104 });

```

Figure 6.4 jQuery for update cart

After Designing we started to write code for backend where all data of user's can be stored below photo shows the product and contact models which generate the fields in AdminLTE3 where admin can add a product and give response to the user from contact fields

```

5  class Product(models.Model):
6      product_id = models.AutoField
7      product_name = models.CharField(max_length=50)
8      category = models.CharField(max_length=50, default="")
9      subcategory = models.CharField(max_length=50, default="")
10     price = models.IntegerField(default=0)
11     desc = models.CharField(max_length=500)
12     pub_date = models.DateField()
13     image = models.ImageField(upload_to="shop/images", default="")
14
15     def __str__(self):
16         return self.product_name
17
18
19  class Contact(models.Model):
20     msg_id = models.AutoField(primary_key=True)
21     name = models.CharField(max_length=50)
22     email = models.CharField(max_length=70, default="")
23     phone = models.CharField(max_length=70, default="")
24     desc = models.CharField(max_length=500, default="")
25     timestamp = models.DateTimeField(default=timezone.now)
26
27     def __str__(self):
28         return self.name
29

```

Figure 6.5 Models for backend

```

6  class OrderUpdateAdmin(admin.ModelAdmin):
7      list_display = ('order_id', 'update_desc', 'timestamp')
8      list_filter = ['timestamp']
9
10     def has_delete_permission(self, request, obj=None):
11         return False
12
13
14  class OrdersAdmin(admin.ModelAdmin):
15      list_display = ('order_id', 'userId', 'name', 'email', 'timestamp')
16      list_filter = ['timestamp']
17
18     def has_add_permission(self, request):
19         return False
20
21     def has_change_permission(self, request, obj=None):
22         return False
23
24
25  class ProductAdmin(admin.ModelAdmin):
26      list_display = ('product_name', 'category', 'price')
27      list_filter = ['category']
28      search_fields = ['product_name']

```

Figure 6.6 Admin side fields


```

151 def productView(request, myid):
152     product = Product.objects.filter(id=myid)
153     # print(product)
154     return render(request, 'shop/prodView.html', {'product': product[0]})
155
156
157 def handleLogin(request):
158     if request.method == "POST":
159         # Get the post parameters
160         loginusername = request.POST['loginusername']
161         loginpassword = request.POST['loginpassword']
162
163         user = authenticate(username=loginusername, password=loginpassword)
164         if user is not None:
165             login(request, user)
166             messages.success(request, "Successfully Logged In")
167             return HttpResponseRedirect(request.META.get('HTTP_REFERER'))
168         else:
169             messages.warning(request, "Invalid credentials! Please try again")
170             return HttpResponseRedirect(request.META.get('HTTP_REFERER'))
171
172     return HttpResponse("404- Not found")

```

Figure 6.7 Views file for redirection pages

```

5  urlpatterns = [
6      path('', views.index, name='index'),
7      path('signup/', views.handleSignUp, name='handleSignUp'),
8      path('login/', views.handleLogin, name="handleLogin"),
9      path('logout/', views.handleLogout, name="handleLogout"),
10     path('about/', views.about, name="AboutUs"),
11     path('contact/', views.contact, name="ContactUs"),
12     path('tracker/', views.tracker, name="TrackingStatus"),
13     path('search/', views.search, name="Search"),
14     path('checkout/', views.checkout, name="Checkout"),
15     path('productView/<int:myid>', views.productView, name="productView"),
16     path('orderView/', views.orderView, name="orderView"),
17     path("handlerequest/", views.handlerequest, name="HandleRequest"),
18     path('send_order_details/', views.send_order_details, name='send_order_details'),
19     # path('send-email/', views.send_email, name='send_email'),
20
21
22     path('reset_password/',
23         auth_views.PasswordResetView.as_view(template_name="shop/password_reset.html"),
24         name="reset_password"),
25
26     path('reset_password_sent/',
27         auth_views.PasswordResetDoneView.as_view(template_name="shop/password_reset_sent.html"),
28         name="password_reset_done"),

```

Figure 6.8 URL path for all pages

6.3 FINDINGS / RESULTS / OUTCOMES

After completing both front-end and back-end programming, we have achieved our goal of creating a desktop application that precisely matches the intended design and meets all required specifications. Our team is proud of the successful outcome and excited to present the final product.

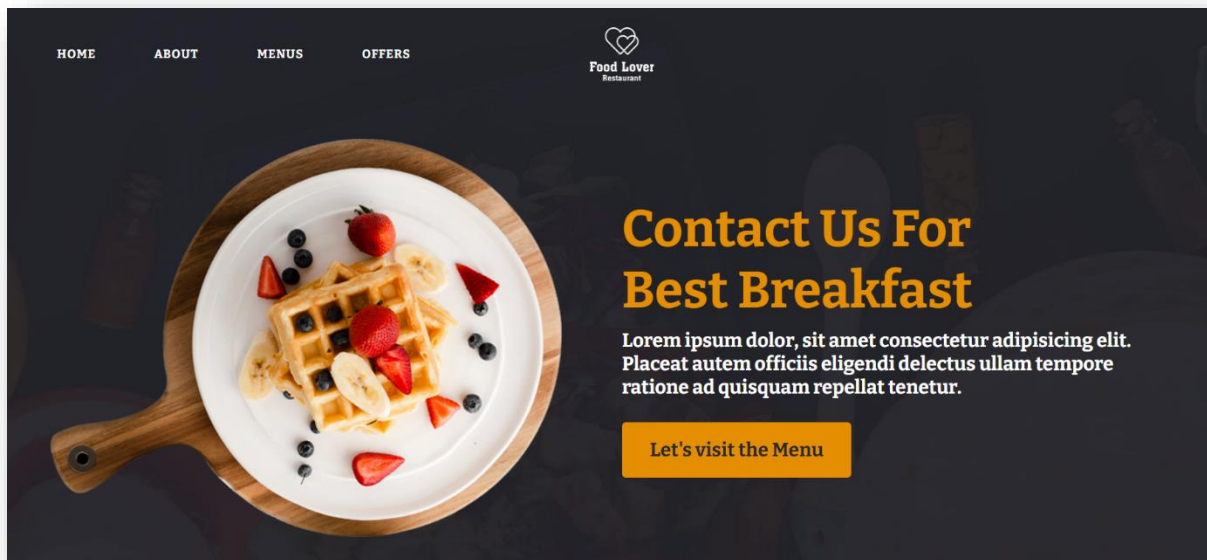


Figure 6.9 Main page

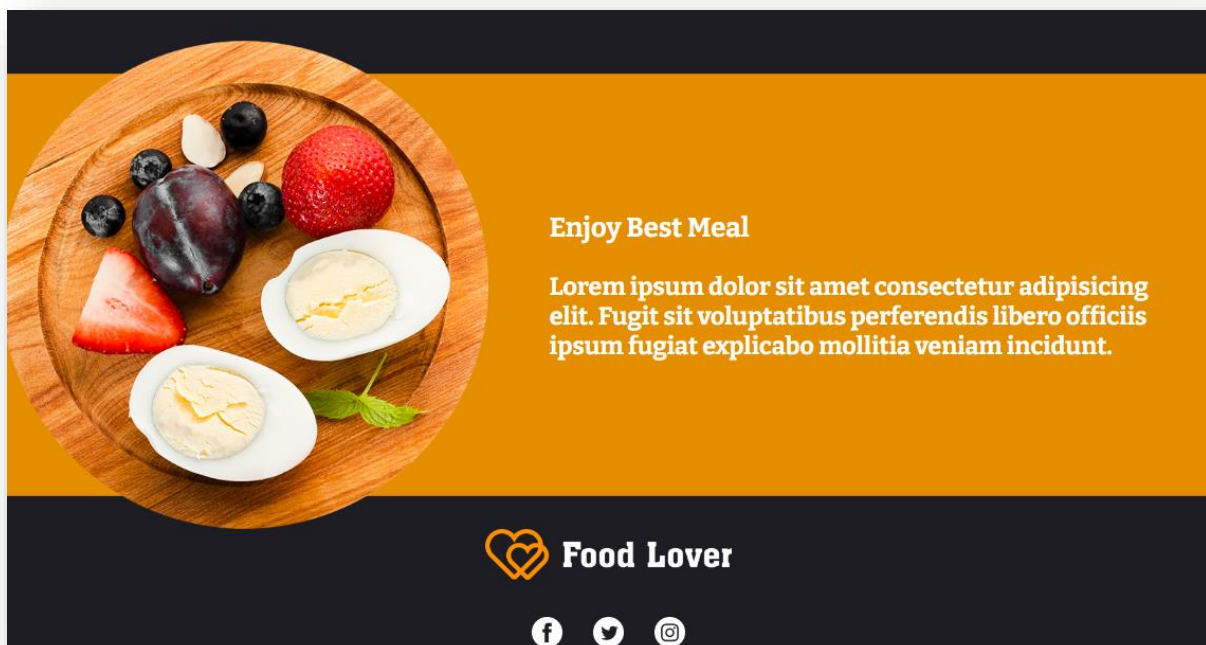


Figure 6.10 end page

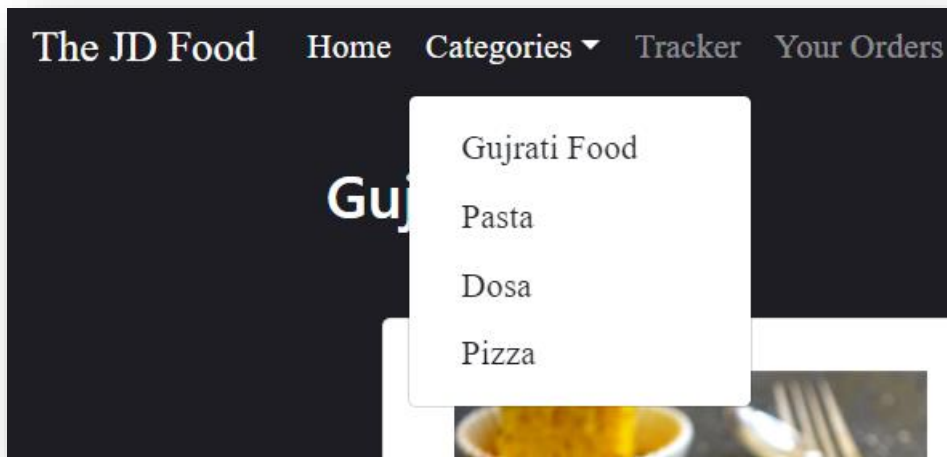


Figure 6.11 categories page



Figure 6.12 Quick view

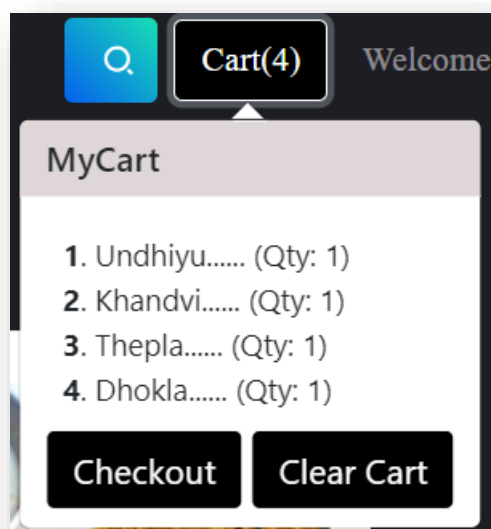


Figure 6.13 Cart options

6.4 RESULT ANALYSIS/ COMPARISONS/DELIBERATIONS

1. The user interface of the web application is visually appealing and runs seamlessly.
2. The web application is designed to be user-friendly and easily navigable.
3. The web application offers the same functionality and features as the desktop application.
4. All elements, such as Tracker and Order update, work effectively as required.
5. The web application allows users to conveniently insert and retrieve data from the database.
6. All buttons offer users easy access to relevant pages.
7. The user identification and authentication process in the web application prioritizes user privacy.

7. TESTING

7.1 TESTING PLAN / STRATEGY

System testing is a comprehensive testing process that evaluates a fully integrated system's compliance with its specified requirements. This testing technique aims to assess both the system's design and behaviour and the customer's expectations. It goes beyond the scope of the Software Requirements Specification (SRS) and tests the system's overall performance. The company's testing strategy was unique and aggressive, comprising various testing methodologies to ensure that the system is efficient and meets all user expectations.

Different testing strategies used are as follow:

Functionality Testing: It is a type of testing that seeks to establish whether each application feature works as per the software requirements. Each function is compared to the corresponding requirement to ascertain whether its output is consistent with the end user's expectations.

Usability Testing: It is a type of testing method for measuring how easy and User-friendly a software is, carried out by a small focus group similar to the users of the system. Its also known as User Experience (UX) Testing. It tests how easily can a user navigate through the system.

Interface Testing: Three areas are tested here: Application, Database and Web Browser.

- **Application:** Test requests are sent correctly to the Database and output at the client side is displayed correctly.
- **Web Server :** Test Web server is handling all application requests without any service denial.
- **Database Server:** Make sure queries sent to the database give expected results.

Database Testing: is a type of software testing that checks the schema, tables, triggers, etc. It also checks data integrity and consistency. It checks whether the data entered in the web application is stored correctly in the database and vice versa.

Compatibility Testing: Compatibility testing is a part of non-functional testing conducted on application software to ensure the application's compatibility with different computing environment. Different web browsers like Google Chrome, Opera, etc were used to test the system.

7.2 TEST RESULTS AND ANALYSIS

The testing strategies implemented by the company resulted in the development of several test cases to ensure the efficient and smooth functioning of the system. Here is a brief overview of the different test cases and their results:

TEST ID	TEST CONDITION	EXPECTED OUTPUT	ACTUAL OUTPUT	REMARKS
T001	Launch application	Index page	Index page	Success
T002	Open categories	Category box with results	Category box with results	Success
T003	Sign up & Login page	Display Sign up & Login page	Display Sign up & Login page	Success
T004	Add items in cart	Items added	Items added	Success
T005	Quick view	Items description displayed	Items description displayed	Success
T006	Contact the admin	Message send	Message send	Success
T007	Shipping address	Data received	Data received	Success
T008	Payment	Cash on delivery	Cash on delivery	Success
T009	Forgot password	Received mail	Received mail	Success

Table 7.1 Test cases

8. CONCLUSIONS AND DISCUSSIONS

8.1 SUMMARY OF INTERNSHIP / PROJECT WORK

A food delivery website is an online platform that enables customers to order food from a wide range of restaurants and food outlets. The website allows users to browse menus, place orders, and pay online for delivery or pickup. Many food delivery websites offer features like personalized recommendations, ratings and reviews, and tracking of the order status. The website typically works with a network of restaurants and delivery partners to ensure quick and reliable delivery to customers. These websites have become increasingly popular due to their convenience, ease of use, and the variety of food options they offer. They are also an excellent platform for small food businesses to expand their reach and attract new customers.

8.2 LIMITATION AND FUTURE ENHANCEMENT

Limitations of food delivery websites include issues such as food quality control during the delivery process, timely delivery, and logistical challenges like weather and traffic conditions. There may also be limitations in terms of the range of food options and the delivery area coverage. Another challenge is that the customer experience can be affected by the reliability and efficiency of the delivery partners.

To address these limitations, food delivery websites could consider implementing enhancements like real-time tracking and delivery updates for customers, improving their logistics and delivery processes, and expanding the range of food options and delivery areas. There could also be opportunities for partnerships with local food producers and suppliers, creating more personalized experiences for customers.

REFERENCES

During the course of project, we took help from certain external sources. Some of them are as follow:

- 1) <https://www.w3schools.com/>
- 2) <https://www.youtube.com/kudvenkat>
- 3) <https://www.youtube.com/@CodeWithHarry>
- 4) <https://www.youtube.com/@TheCodersCubes>
- 5) <https://www.djangoproject.com/>
- 6) <https://www.youtube.com/@wscubetech>