MAIN PROJECT REPORT

ON

“FOOD SQUARE”

BACHELOR OF COMPUTER APPLICATIONS

(VI SEMESTER)

DONE BY

JINS JOSEPH (Reg.No :210021091429)

Under the guidance of

Ms. JASMINE MATHEW

(Asst.Professor)

Department of Computer Science

ST. JOSEPH’S ACADEMY OF HIGHER EDUCATION AND RESEARCH, MOOLAMATTOM

(Affiliated to Mahatma Gandhi University)
2023-2024

ST. JOSEPH’S ACADEMY OF HIGHER EDUCATION AND RESEARCH, MOOLAMATTOM

(Affiliated to Mahatma Gandhi University)

Department of Computer Science

CERTIFICATE

Certified that this is a bonafide report of the Project with the topic “FOOD SQUARE” for St. Joseph’s Academy of Higher Education and Research, Moolamattom done by JINS JOSEPH (Reg No: 210021091429) during the year 2023-2024 in partial fulfilment of the requirements for the award of the degree of Bachelor of Computer Applications of Mahatma Gandhi University, Kottayam, Kerala.

Principal

Head of the Department

Project Guide

Submitted for the Viva-Voce Examination held on ……….

External Examiner

ACKNOWLEDGEMENT

I give all the honors and praise to the lord who gave me wisdom and enabled me to complete this project successfully.

I express my sincere thanks to Dr. THOMSON JOSEPH my beloved principle for providing necessary facilities for the completion of my project successfully.

I thank profusely Mr. Geo Kurian, Head of the department of Computer Science for his guidance and inspiration throughout my course of study.

I express my sincere gratitude to my project guide Ms.Jasmine Mathew,Assistant professor, for the valuable advices and guidance throughout the completion of the project.

I also express my gratitude and thanks to all my teachers and friends for their sincere and friendly cooperation in the successful completion of my project.

By

Jins Joseph

INDEX

INTRODUCTION

1.INTRODUCTION

Welcome to Food Square, where culinary delights meet convenience at your fingertips. As a premier food delivery organization, we're dedicated to bringing the flavors of the world to your doorstep with speed, reliability, and unparalleled taste.

At Food Square, we understand that food is more than just sustenance – it's an experience. That's why we've curated a diverse selection of top-rated restaurants and eateries, offering a wide range of cuisines to satisfy every palate. Whether you're craving hearty comfort food, exotic international dishes, or healthy options to fuel your day, Food Square has you covered.

Our mission is simple: to connect food enthusiasts with the best culinary experiences in their area. With our user-friendly platform and intuitive ordering process, you can explore menus, customize your meals, and place orders with just a few clicks. No more waiting in line or struggling to make reservations – with Food Square, delicious dining experiences are just a tap away.

But our commitment to excellence doesn't stop there. We prioritize quality, freshness, and customer satisfaction in everything we do. From the ingredients used in each dish to the packaging materials we employ, we ensure that every aspect of your Food Square experience exceeds expectations.

Furthermore, we're committed to sustainability and environmental responsibility. That's why we offer eco-friendly packaging options and work with our restaurant partners to minimize food waste and promote eco-conscious practices throughout our operations.

Whether you're dining solo, hosting a gathering, or simply seeking a delicious meal on the go, Food Square is here to elevate your dining experience. Join us on a culinary journey like no other and discover why Food Square is the ultimate destination for food lovers everywhere. Welcome to Food Square – where every bite tells a story, and every meal is a celebration of flavor.

ABOUT THE ORGANISATION

ABOUT THE ORGANISATION

"Food Square" is envisioned as an organization where culinary delights meet digital convenience. At the heart of Food Square is its management team, orchestrating operations, finance, and strategy to ensure the platform's success. The technology team, adept at coding and innovation, crafts the user-friendly website, mobile apps, and robust backend systems, enabling seamless ordering and delivery experiences.

Complementing the tech wizards, the operations team coordinates with partner restaurants, manages logistics, and oversees delivery drivers, ensuring hot meals reach hungry customers promptly. Customer support, a cornerstone of Food Square's ethos, promptly addresses inquiries and resolves issues, ensuring customer satisfaction.

Fueling growth and engagement, the marketing team devises creative campaigns, cultivates brand identity, and forges strategic partnerships to expand Food Square's reach and entice new users. Legal and compliance teams diligently navigate regulatory landscapes, safeguarding data privacy and food safety while ensuring adherence to industry standards.

In the financial realm, the finance and accounting department meticulously manages budgets, forecasts, and financial strategies, ensuring the organization's fiscal health and sustainability. Meanwhile, business development scouts new opportunities, fostering collaborations with restaurants and suppliers to diversify offerings and enhance user experiences.

Driving innovation and user-centric design, the product development team iteratively improves the platform based on user feedback and market trends, ensuring Food Square remains at the forefront of food delivery technology. Quality assurance specialists rigorously test the platform's functionality, reliability, and security, upholding Food Square's commitment to excellence.

Security experts vigilantly safeguard user data and transactional integrity, fortifying Food Square against cyber threats and ensuring a safe and secure ordering environment. Together, these teams synergize to realize Food Square's vision: to be the premier destination where gastronomic delights meet digital convenience, enriching lives one meal at a time.

SYSTEM ANALYSIS

EXISTING SYSTEM

The existing system refers to the system that is being followed till now, a detailed study of the existing system is carried out along with all the steps in system analysis. The existing system requires more computational time, and more manual calculations and the complexity involved in the Selection of features is high. The other disadvantages are lack of security of data, Deficiency of Data accuracy, Time consumption, etc. To avoid all these limitations and make the work more accurate the system needs to be computerized.

The existing food delivery application comprises several interconnected modules tailored to different user roles: admin, restaurant, user, and delivery personnel.The admin module serves as the backbone, managing the entire system's operations, including user authentication, restaurant onboarding, menu management, order tracking, and resolving disputes. It ensures smooth functioning by overseeing all activities and resolving any technical glitches or disputes that may arise.

The restaurant module allows registered eateries to create and customize their profiles, upload menus, manage orders, and track their performance. It provides them with insights into customer preferences and sales trends, enabling them to optimize their offerings and enhance customer satisfaction.

The user module enables customers to browse through a variety of restaurants, explore menus, place orders, make payments, and track their deliveries in real-time. It offers personalized recommendations based on past orders and preferences, enhancing the overall user experience.

The delivery personnel module facilitates seamless coordination between delivery agents and restaurants, assigning orders based on proximity and availability. It provides delivery personnel with essential information such as order details, navigation assistance, and customer contact information, ensuring timely and efficient deliveries.

Together, these modules form a comprehensive ecosystem that streamlines the food delivery process, from order placement to delivery, enhancing convenience for both users and restaurants while optimizing operational efficiency.

- The Existing system only provides a test-based interface, which is not as user-friendly.
- Since the system is implemented in the manual, the response is very slow.

- **Drawbacks of the existing system.**

- Limited Restaurant Selection.
- Not user-friendly.
- Delivery Time Variability.
- High Commission Fees.
- User Interface Complexity:.
- Time-Consuming.
- Technical Glitches .
- Not properly updating data.
- Lack of good Interface.
- Quality Control
- Customer Service Issues.

Hence, there is a need for a reformation of the system with more advantages and flexibility. To avoid all these limitations and make the work more accurate the system needs to be computerized.

PROPOSED SYSTEM

The proposed food delivery system aims to revolutionize the user experience while addressing inherent limitations in the existing platform. At its core, the system will prioritize expanding the network of partner restaurants to offer users a broader selection of culinary options, catering to diverse tastes and preferences. By onboarding a wider array of eateries, users will have access to a more comprehensive range of cuisines, enhancing their overall satisfaction and engagement with the platform.

One of the key enhancements of the proposed system lies in its implementation of advanced algorithms to provide more accurate and predictive delivery time estimates. By analyzing real-time data such as traffic patterns, order volume, and historical delivery times, the system will offer users a clearer picture of when to expect their orders, reducing variability and improving customer satisfaction. This feature will not only enhance user experience but also streamline operations for delivery personnel and partner restaurants.To ensure a fair and sustainable partnership with restaurants, the proposed system will introduce transparent commission structures. By offering fair compensation for their participation, the system aims to attract high-quality establishments while maintaining a viable business model. This approach fosters long-term relationships with partner restaurants, ensuring a diverse and reliable selection of dining options for users.Usability is paramount in the proposed system, which will feature an intuitive user interface designed to streamline the ordering process. By prioritizing simplicity and ease of navigation, the system aims to enhance user engagement and satisfaction. Moreover, a robust technical infrastructure will underpin the platform, minimizing technical glitches and ensuring reliable performance even during peak hours.

Stringent quality assurance measures will be implemented to uphold high standards of food quality and safety. Partner restaurants will be required to adhere to strict guidelines, ensuring consistency and trust among users. Additionally, responsive customer service channels will be available to address any issues or concerns promptly, further enhancing the overall user experience.

Finally, the proposed system will leverage machine learning algorithms to provide personalized recommendations to users based on their past orders, preferences, and browsing history. This personalized approach not only enhances user engagement but also facilitates discovery, encouraging users to explore new dining experiences within the platform.

In summary, the proposed food delivery system aims to create an efficient, reliable, and user-centric platform that meets the needs of both users and partner restaurants. Through its emphasis on expansion, accuracy, fairness, usability, quality, responsiveness, and personalization, the system seeks to elevate the food delivery experience for all stakeholders involved..

Advantages of the Proposed System

The system is very simple in design and to implement. The system requires very low system resources and the system will work in almost all configurations.

It has got following features

- Expanded Restaurant Selection.
- Accurate Delivery Time Estimates.
- Fair Commission Structure.
- Intuitive User Interface.
- Robust Technical Infrastructure.
- Stringent Quality Assurance:.
- Responsive Customer Service.
- Personalized Recommendations.
- Streamlined Operations
- Enhanced Trust and Satisfaction.

SOFTWARE REQUIREMENT SPECIFICATION(SRS)

SOFTWARE REQUIREMENTS SPECIFICATION

A software requirements specification (SRS) is a comprehensive description of the intended purpose and the environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform. As SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will system hardware, other programs, and human users in a wide variety of real-world situations.

1. User Requirements

There are three types of users in this system-Admin, User, Restaurant,and Delivery Boy. The Admin controls all the flow of the applications and coordinates overall system performance and delivery the items of users. The User needs to order appropriate food items from a list restaurant by following price and rating and the restrauant need to upload the food items in the organization .Admin can reject or accept registration request sent by hotel that is checked by admin.

2. Functional Requirements

Functional Requirements indicate what a software system must do and how it must function; there are food items features that focus on user needs. These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Admin

- Dashboard: Access to overall system statistics, user activity, and financial reports.
- Restaurant Management: Features for onboarding new restaurants, managing menus.
- Feedback and Support: Access to customer feedback, resolving disputes, and providing support to users.
- Analytics: Reporting tools for tracking key metrics such as orders, revenue, customer retention, etc.

Restaurant

- Menu Management: Ability to add, edit, and remove items from the menu. Set prices, descriptions, and availability.
- Order Management: Receive and manage incoming orders, update order status (accepted, preparing, dispatched).
- Analytics: Access to sales reports, customer feedback, and performance metrics. □ Account Settings: Manage profile information and notification preferences.

Users

- Registration and Authentication: Create an account, login/logout securely
- Browsing and Search: Search for restaurants, cuisines, and dishes. Filter options based on preferences.
- Order Placement: Select items from menus, customize orders, specify delivery address, and payment method.
- Order Tracking: Real-time tracking of order status from placement to delivery.
- Profile Management: Edit profile information, view order history, and manage addresses and payment methods
- Notifications: Receive notifications for order status updates, promotions, and other relevant information.
- Complaint and Feedback: Ability to complaint and feedback restaurants and delivery agents.

Delivery Boy

- Order Assignment: Receive incoming delivery requests, accept or reject based on availability.
- Order Status Updates: Notify users about the status of their delivery (picked up, en route, delivered).
- Earnings and Payments: View earnings, track payments, and manage payout preferences.

These requirements should provide a solid foundation for developing a hotel hub platform that meets the needs of both guests and restaurant staff.

3. Non-Functional Requirements

Functional requirements specify what a system is supposed to do whereas non-functional requirements define how a system is supposed to be. Non-functional requirements are often called qualities of a system.

- Performance: It helps the users access the system quickly.
- Compatibility: The system is supported in Chrome, Microsoft Edge, Firefox, etc.
- Efficiency: Users can do their tasks quickly.
- Learnability: User-friendly interface is provided which helps users to learn and use the system quickly.
- Usability: The system is easy to learn without a tutorial and allows users to accomplish their goals without errors.

4. System Requirements

To be used efficiently, all computer software needs certain hardware components or other software resources to be present on a system. These prerequisites are known as system requirements and are often used as a guideline as opposed to an absolute rule.

4.1 Hardware Specification

Hardware is the term used to describe the electrical and electronic components that make up a computer system. The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirement list is often accompanied by a hardware compatibility list (HCL), especially in the case of operating systems. An HCL lists tested compatible, and sometimes incompatible hardware devices for a particular operating system or application.

- Processor: Intel Pentium or above
- Hard Disk: up to 200 MB of available space may be required
- Display Type: PC Display
- Keyboard: PC/AT Enhanced PS/2Keyboard (110/10Key)
- Mouse: First/Pilot Mouse Serial(c48) / Touchpad
- Input Device: Mouse, Keyboard
- Output Device: Monitor

4.2 Software Specifications

The software requirements specification document enlists all requirements that are required for the project development. To derive the requirements, we need to have a clear and thorough understanding of the products to be developed. This is prepared after detailed communication with the project user.

- Operating System: Windows 10 or above for better performance Front end: HTML5, CSS3.0, Java script.
- Back end: Python Django
- Language: Python
- Database: SQLite3
- Web browser: Internet explorer/Google chrome/Firefox

4.3 WINDOWS 10

The operating system is defined as a program that manages computer hardware. An OS can be viewed as a scheduler, where it has resources for which it has changed. Resources include CPU, memory, I/O device, and disk space. In another view, the OS is a new machine. The third view is that the OS is a multiplexer that allows sharing of resources provides protection from interference and provides a level of cooperation between users. This project is developed using Windows 10 as the OS and supports its latest version. Windows 10 is a series of personal computer OS produced by Microsoft as part of its Windows NT family of OS. Windows 10 received mostly positive reviews upon its original release in July 2015.

One of Windows 10's most notable features is support for universal apps. Windows 10 also introduced the Microsoft Edge web browser, a virtual desktop system, a window, and desktop management feature called Task View, support for fingerprint and face recognition login, new security features for enterprise environments, and DirectX12. Windows 10 received mostly positive reviews upon its original release in July 2015. Critics praised Microsoft's decision to provide a desktop-oriented interfacing line with previous versions of Windows, contrasting the tablet-oriented approach of 8, although Windows 10's touch-oriented user interface mode was criticized for containing regressions upon the touch-oriented interface of Windows 8. Critics also praised the improvements to Windows 10's bundled software over

Windows 8.1. Xbox Live integration, as well as the functionality and capabilities of the Cortana personal assistant and the replacement of Internet Explorer with Microsoft Edge. However, media outlets have been critical of changes to operating system behaviours, including mandatory update installation, privacy concerns over data collection performed by the OS for Microsoft and its partners, and the adware-like tactics used to promote the operating system on its release.

1.1 Python Django

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics developed by Guido van Rossum. It was originally released in 1991. Designed to be easy as well as fun, the name "Python" is a nod to the British comedy group Monty Python. Python has a reputation as a beginner-friendly language, replacing Java as the most widely used introductory language because it handles much of the complexity for the user, allowing beginners to focus on fully grasping programming concepts rather than minute details. 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 the reuse of code. Python is an open- source community language, so numerous independent programmers are continually building libraries and functionality for it.

Django is a high-level Python web framework that enables the rapid development of secure and maintainable websites. Built by experienced developers, Django 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 is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support. Django is a Python-based web framework that allows you to quickly create efficient web applications. It is also called batteries included framework because Django provides built-in features for everything including Django Admin

Interface, default database – SQLlite3, etc. When you’re building a website, you always need a similar set of components: a way to handle user authentication (signing up, signing in, signing out), a management panel for your website, forms, a way to upload files, etc. Django gives you ready-made components to use and that too for rapid development.

Advantages of Django

Few advantages of using Django which can be listed below:

- **Object-Relational Mapping (ORM) Support** − Django provides a bridge between the data model and the database engine and supports a large set of database systems including MySQL, Oracle, Postgres, etc. Django also supports the NoSQL database through the Django-nonrel fork. For now, the only NoSQL databases supported are MongoDB and google app engine.
- **Multilingual Support** − Django supports multilingual websites through its built-in internationalization system. So you can develop your website, which would support multiple languages.
- **Framework Support** − Django has built-in support for Ajax, RSS, Caching, and various other frameworks.
- **Administration GUI** − Django provides a nice ready-to-use user interface for administrative activities.
- **Development Environment** − Django comes with a lightweight webserver to facilitate end-to-end application development and testing.

4.4 HTML5

HTML stands for Hyper Text Markup Language. It is the standard markup language for creating Web pages. HTML describes the structure of a Web page. It consists of a series of elements. HTML elements tell the browser how to display the content

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web. HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG). The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight, and Google Gears.

4.5 CSS 3.0

Cascading Style Sheets (CSS) is a simple mechanism for adding style (e.g., fonts, colors, spacing) to Web documents. **CSS3** is the latest evolution of the Cascading Style Sheets language and aims at extending CSS2.1. It brings a lot of new features and additions, like rounded corners, shadows, gradients, transitions, or animations, as well as new layouts like multi-columns, flexible boxes, or grid layouts.

4.6 JavaScript

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complementary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

Applications of JavaScript

- Client-side validation - This is important to verify any user input before submitting it to the server and Javascript plays an important role in validating those inputs at the front end itself.
- User Notifications - You can use Javascript to raise dynamic pop-ups on the webpages to give different types of notifications to your website visitors.
- Back-end Data Loading – Java script provides Ajax library which helps in loading back-end data while you are doing some other processing. This gives an amazing experience to your website visitors.
- Server Applications - Node JS is built on Chrome's Java script runtime for building fast and scalable network applications. This is an event-based library that helps in developing very sophisticated server applications including Web Servers.

4.7 SQLite3

SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. It is a database, which is zero-configured, which means like other databases you do not need to configure it in your system. SQLite engine is not a standalone process like other databases, you can link it statically or dynamically as per your requirement with your application. SQLite accesses its storage files directly. It is a popular choice as an embedded database for local/client storage in application software such as web browsers. It is also used in many other applications that need a lightweight, embedded database.

SQLite is ACID-compliant and implements most of the SQL standards, using a dynamically and weakly typed SQL syntax that does not guarantee domain integrity.

To use SQLite in a C/C++ program, you can use the sqlite3 API, which provides a lightweight, simple, self-contained, high-reliability, full-featured, and SQL database engine. The API is implemented as a library of C functions that can be called from your program. One of the main benefits of using SQLite is that it is very easy to get started with. To create a new database in SQLite, you simply need to create a new file on your filesystem and connect to it using the sqlite3 API.

SQLite was created in the year 2000 by D. Richard Hipp, who continues to lead the development of the software today. SQLite was designed to be a lightweight and simple database engine that could be easily embedded into other applications. It was created as an alternative to more complex and heavyweight database engines, such as MySQL and PostgreSQL. Over the years, SQLite has gained widespread adoption and is now one of the most widely used database engines in the world. It is used in many applications, including web browsers, mobile phones, and a wide variety of other software.

SQLite is a C library that provides a lightweight disk-based database that doesn’t require a separate server process and allows accessing the database using a nonstandard variant of the SQL query language. Some applications can use SQLite for internal data storage. It’s also possible to prototype an application using SQLite and then port the code to a larger database such as PostgreSQL or Oracle.

Python SQLite3 module is used to integrate the SQLite database with Python. It is a standardized Python DBI API 2.0 and provides a straightforward and simple-to-use interface for interacting with SQLite databases. There is no need to install this module separately as it comes along with Python after the 2.5x version.

Databases offer numerous functionalities by which one can manage large amounts of information easily over the web and high-volume data input and output over a typical file such as a text file. SQL is a query language and is very popular in databases. Many websites use MySQL. SQLite is a “light” version that works over syntax very much similar to SQL. SQLite is a self-contained, high-reliability, embedded, full-featured, public-domain, SQL database engine. It is the most used database engine on the world wide web. Python has a library to access SQLite databases, called sqlite3, intended for working with this database which has been included with the Python package since version 2.5. SQLite has the following features.

QUALITY ATTRIBUTES

Reliability:

Measure if the product is reliable enough to sustain in any condition. Should give consistently correct results. Product reliability is measured in terms of working of the project under different working environment and different conditions.

Maintainability:

Different versions of the product should be easy to maintain. For development it should be easy to add code to the existing system, should be easy to upgrade for new features and new technologies from time to time. Maintenance should be cost-effective and easy. The system is easy to maintain and correcting defects or making a change in the software.

Usability:

This can be measured in terms of ease of use. The application should be user-friendly. Should be easy to learn. Navigation should be simple. The system must be: Easy to use for input preparation, operation, and interpretation of the output. Provide consistent user interface standards or conventions with our other frequently used systems. Easy for new or infrequent users to learn to use the system.

Portability:

This can be measured in terms of Costing issues related to porting, technical issues related to porting, Behavioural issues related to porting.

Correctness:

The application should be correct in terms of its functionality, calculations used internally and the navigation should be correct. This means the application should adhere to functional requirements.

Efficiency:

Major system quality attribute measured in terms of time required to complete any task given to the system. For example, the system should utilize processor capacity, disk space and memory efficiently. If system is using all the available resources, then the user will get degraded performance failing the system for efficiency. If system is not efficient then it cannot be used in realtime applications.

Integrity or Security:

Integrity comes with security. System integrity or security should be sufficient to prevent unauthorized access to system functions, preventing information loss, ensure that the software is protected from virus infection, and protecting the privacy of data entered into the system.

Testability:

The system should be easy to test and find defects. If required should be easy to divide into different modules for testing.

Flexibility:

Should be flexible enough to modify. Adaptable to other products with which it needs interaction. Should be easy to interface with other standard 3rd party components.

Reusability:

Software reuse is a good cost-efficient and time-saving development way. Different code libraries classes should be generic enough to use easily in different application modules. Dividing application into different modules so that modules can be reused across the application. Applying above quality attributes we can determine whether the system meets the requirements

FEASIBILITY ANALYSIS

FEASIBILITY ANALYSIS

Feasibility study is carried out to select the best system that meets performance requirements. This entails identification, description, and evaluation of candidate systems and selection of the best system for the job. The main aim of the feasibility study is to evaluate alternative system and to propose the most feasible and desirable system for development. During the detailed study conducted in existing system, I have noticed several problems. It is a time consuming and a job pending one. After Initial investigation culminates in a proposal that determines whether an alternative system is feasible. There are three aspects to be considered in the feasibility study regarding the online aptitude test.

5.1 TECHNICAL FEASIBILITY

Technical feasibility is done to check whether the working for the project be done with the current equipment, existing software and available personnel. The proposed system need not required any skilled person in computer. The existing staff can be trained for the computerized system effectively. By using the software, the candidate system can be done very effectively. Since visual basic provide an excellent development environment, using this system we can do many things without programming.

5.2 OPERATIONAL FEASIBILITY

The operational feasibility can be stated as: if the developed system is implemented there will be reactions from the employee that will determine the possible operational benefits. The user of proposed system must have great interest in computerizing the system. The candidate system will eliminate the employee’s workload and facilitate effective functioning and double the works the present system can. The proposed system will not eliminate any staff, but the staff can be replaced to another productive job. Thus, we conclude that the change made in computerization is operationally feasible.

5.3 ECONOMIC FEASIBILITY

It is also called cost/benefit analysis. It evaluates the benefits and savings expected from a candidate system compares them with lost. The variable cost of the existing system is higher than the proposed system. In the proposed system all records are kept in magnetic disk. This provides quick reference and reduces variable cost. The current system requires several employees and large amount of money is needed as salary.

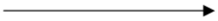
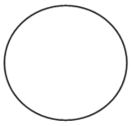
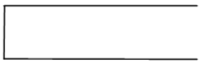
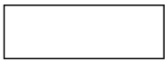
DATA FLOW DIAGRAM(DFD)

6. DATA-FLOW-DIAGRAM(DFD)

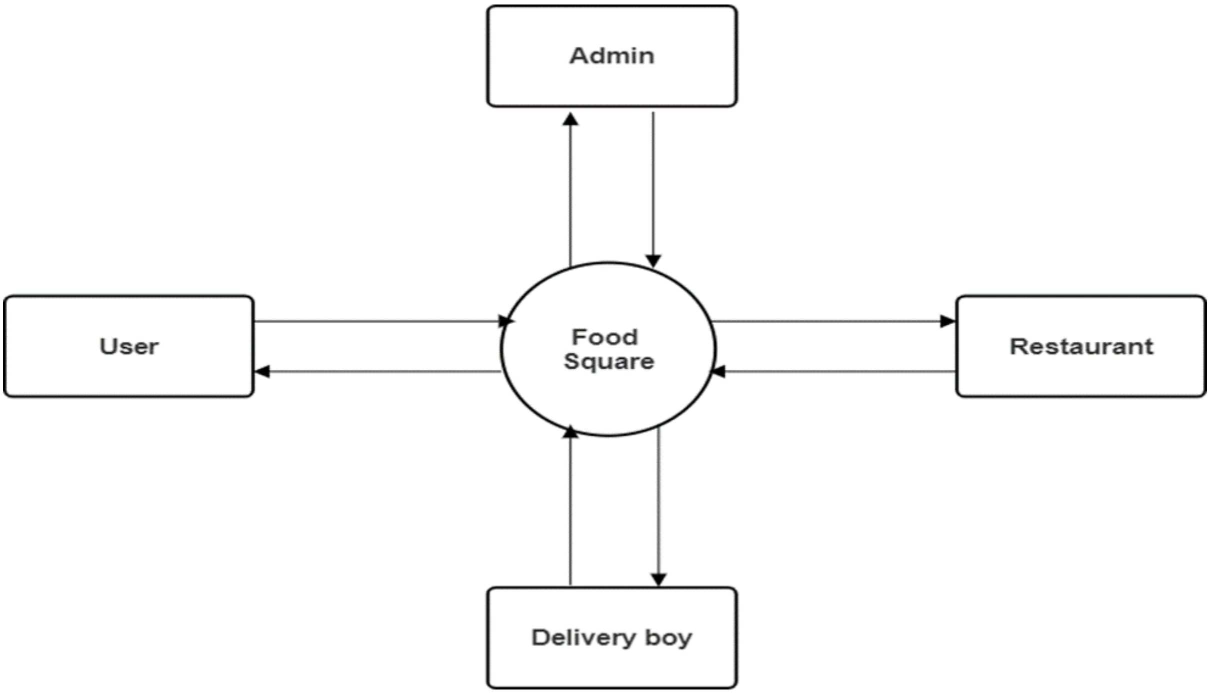
Data Flow Diagram is a network that describes the flow of data and processes that change, or transform, data throughout the system. This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs. There are various symbols used in a DFD. Bubbles represent the processes. Named arrows indicate the data flow. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. Each component in a DFD is labelled with a descriptive name. Process names are further identified with a number.

The Data Flow Diagram shows the logical flow of a system and defines the boundaries of the system. For a candidate system, it describes the input (source), outputs (destination), database (files) and procedures (data flow), all in a format that meet the user's requirements. The main merit of DFD is that it can provide an overview of system requirements, what data a system would process, what transformations of data are done, what files are used, and where the results flow. This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, outputdata-flows which go to other processes or external entities or files. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, outputdata-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

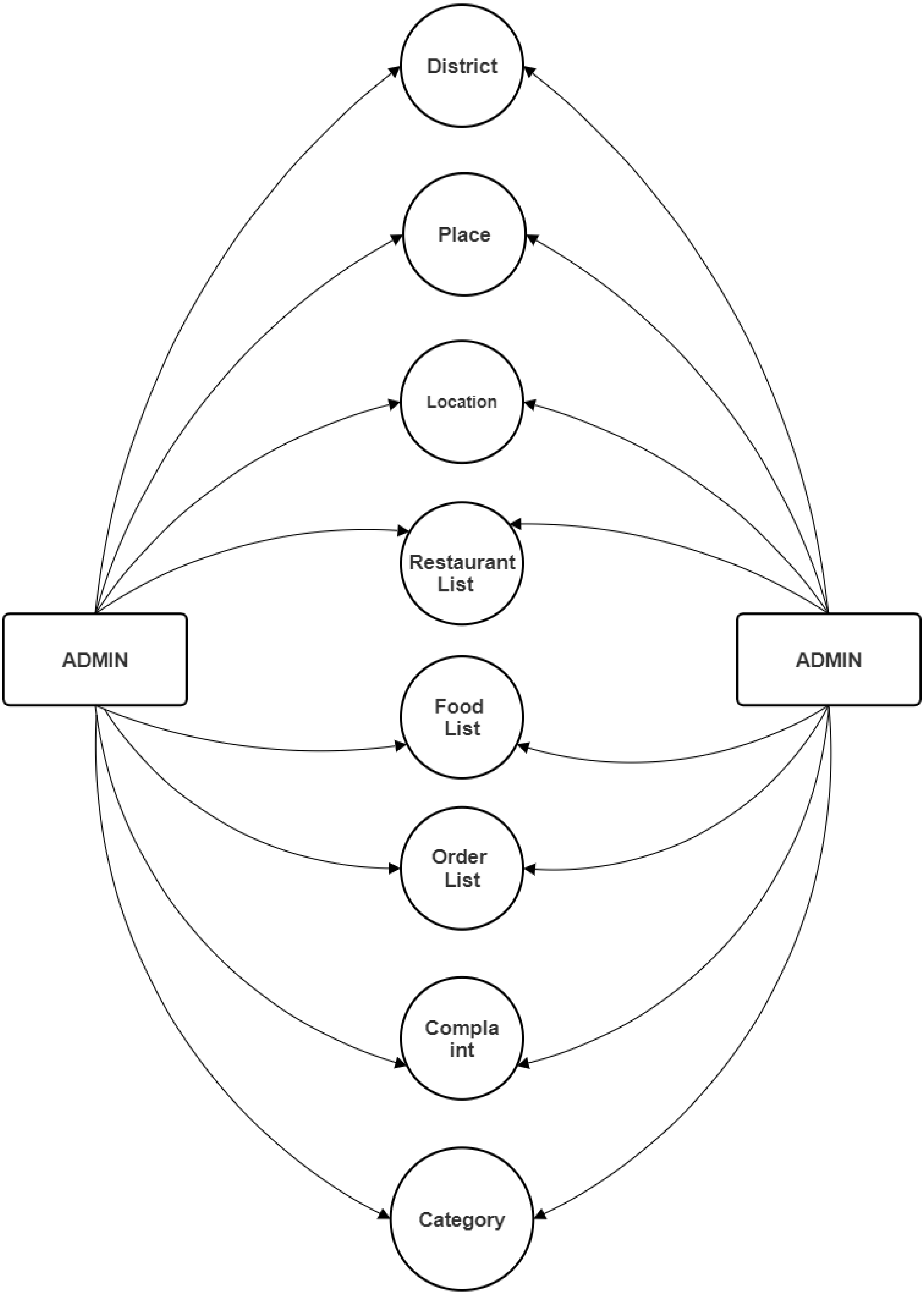
Basic Data Flow Diagram Symbol:

	A data flow is a route, which enables packets of data to travel from one point to another. Data may flow from a source to a process and from data store or process. An arrow line depicts the flow, with arrow head pointing in the direction of the flow.
	Circles stands for process that converts data in to information. A process represents transformation where incoming data flows are changed into outgoing data flows.
	A data store is a repository of data that is to be stored for use by a one or more process may be as simple as buffer or queue or sophisticated as relational database. They should have clear names. If a process merely uses the content of store and does not alter it, the arrowhead goes only from the store to the process. If a process alters the details in the store then a double-headed arrow is used.
	A source or sink is a person or part of an organization, which enters or receives information from the system, but is considered to be outside the contest of data flow model.

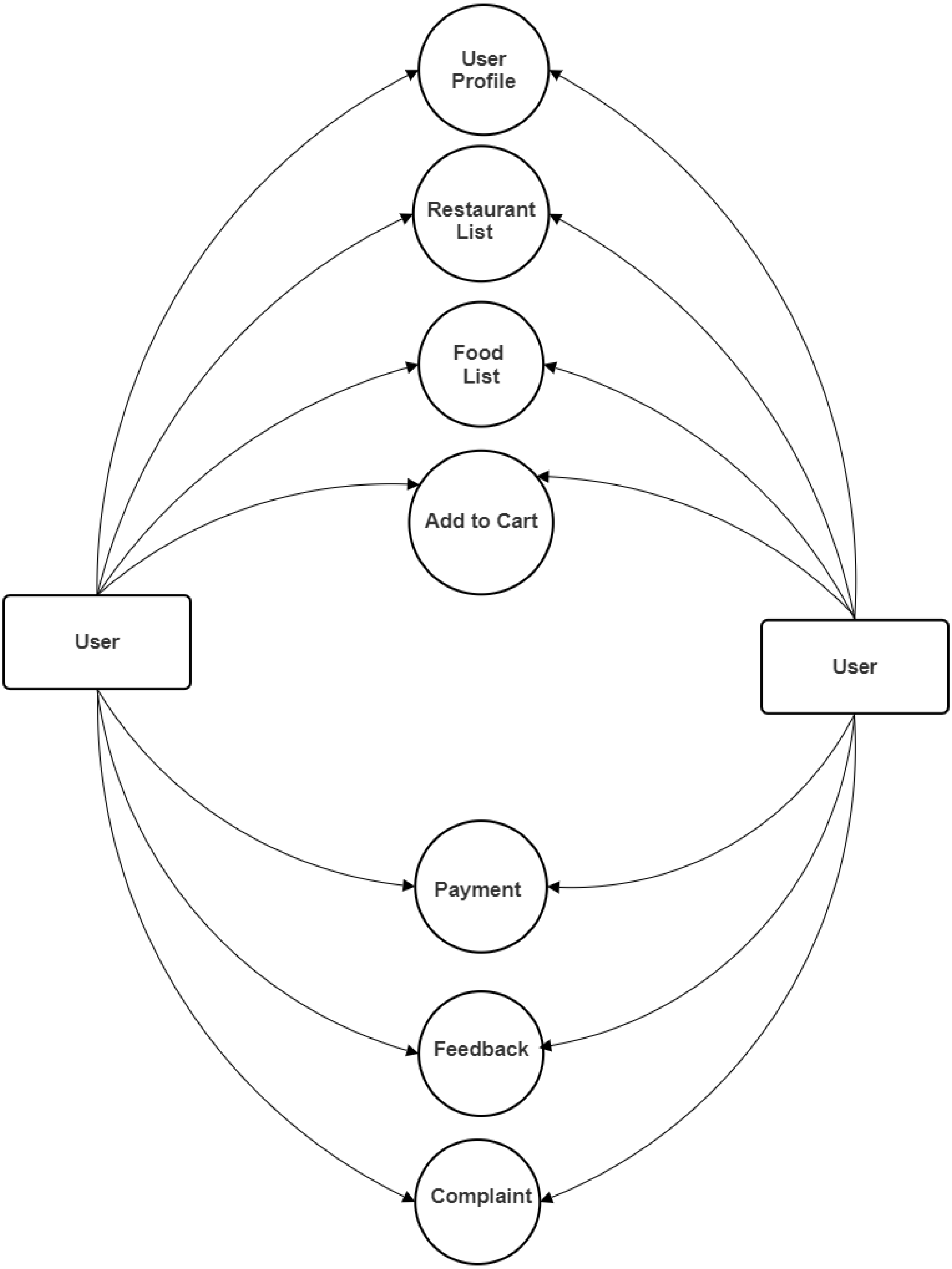
Context level



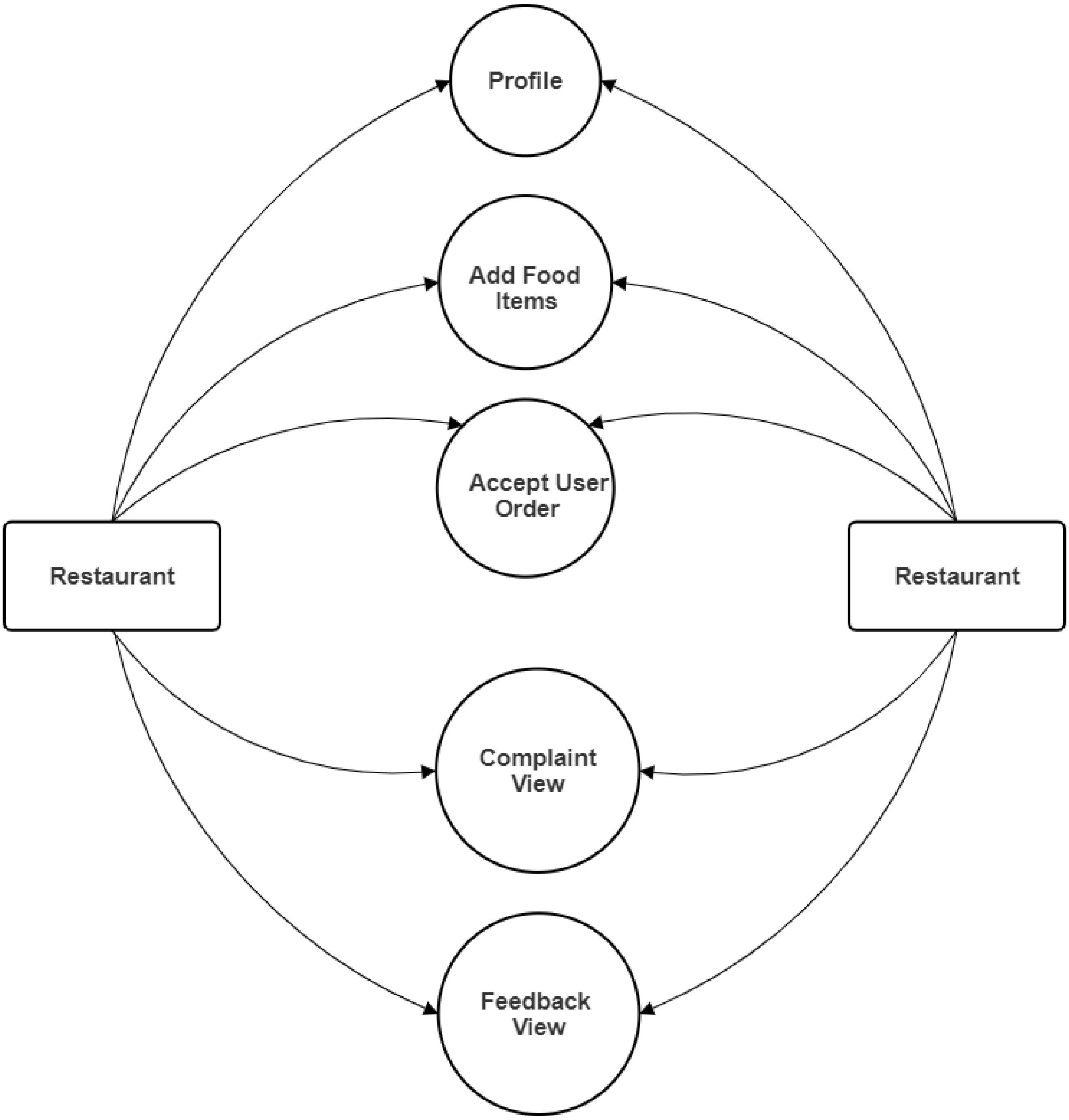
Level 0 Admin



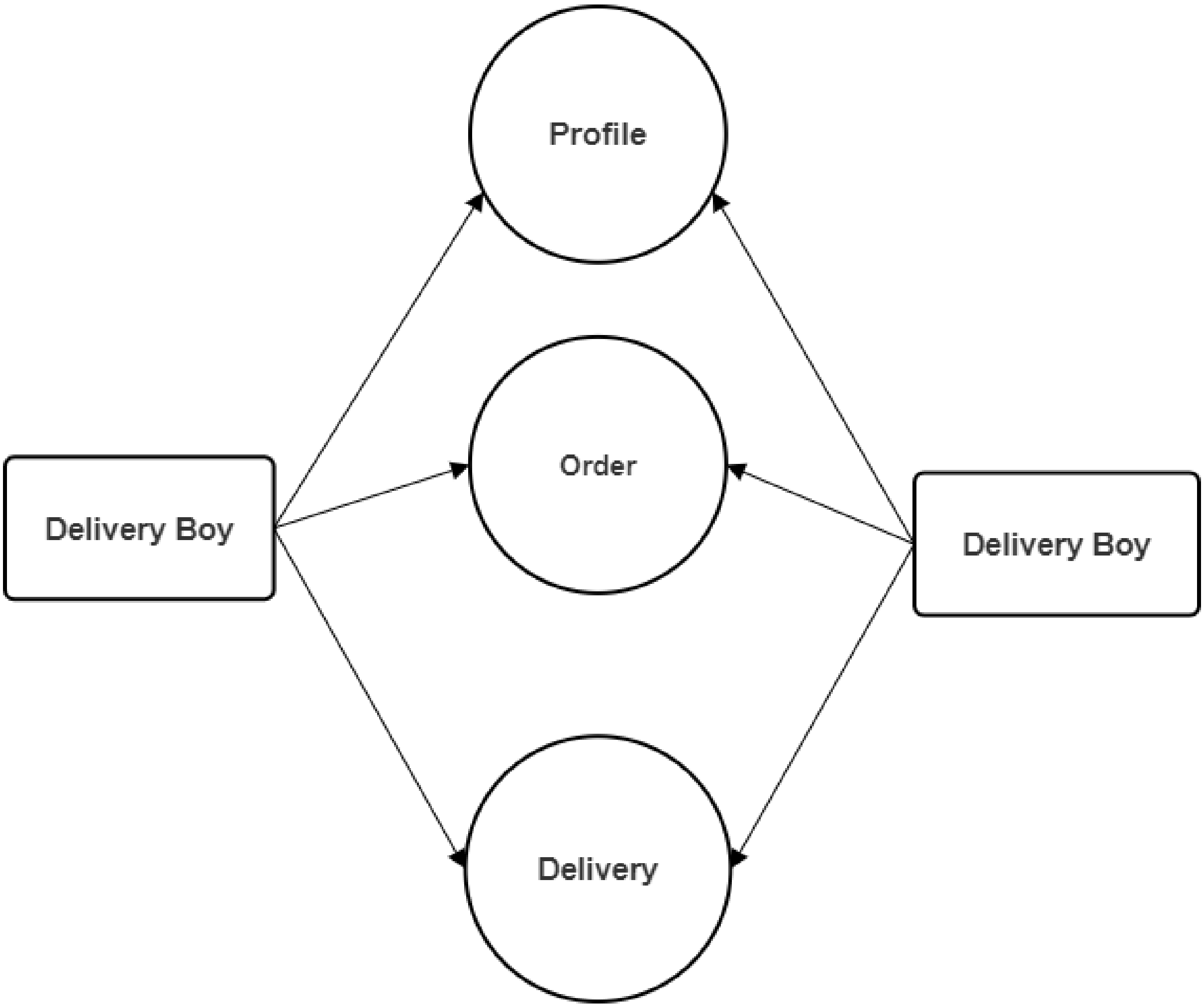
Level 1 User


User Profile
Restaurant List
Food List
Add to Cart
User
User
Payment
Feedback
Complaint

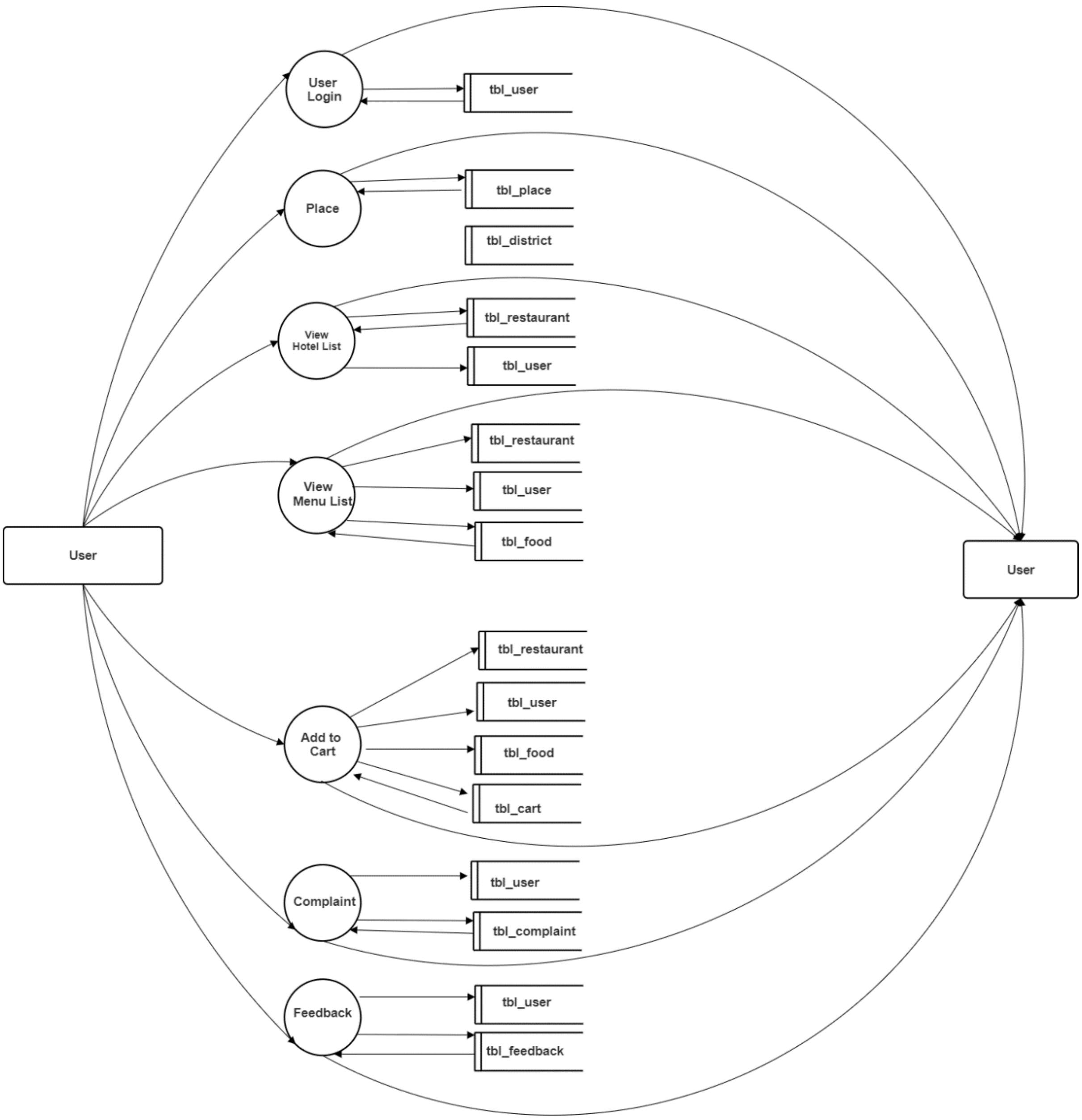
Level 0 Restaurant



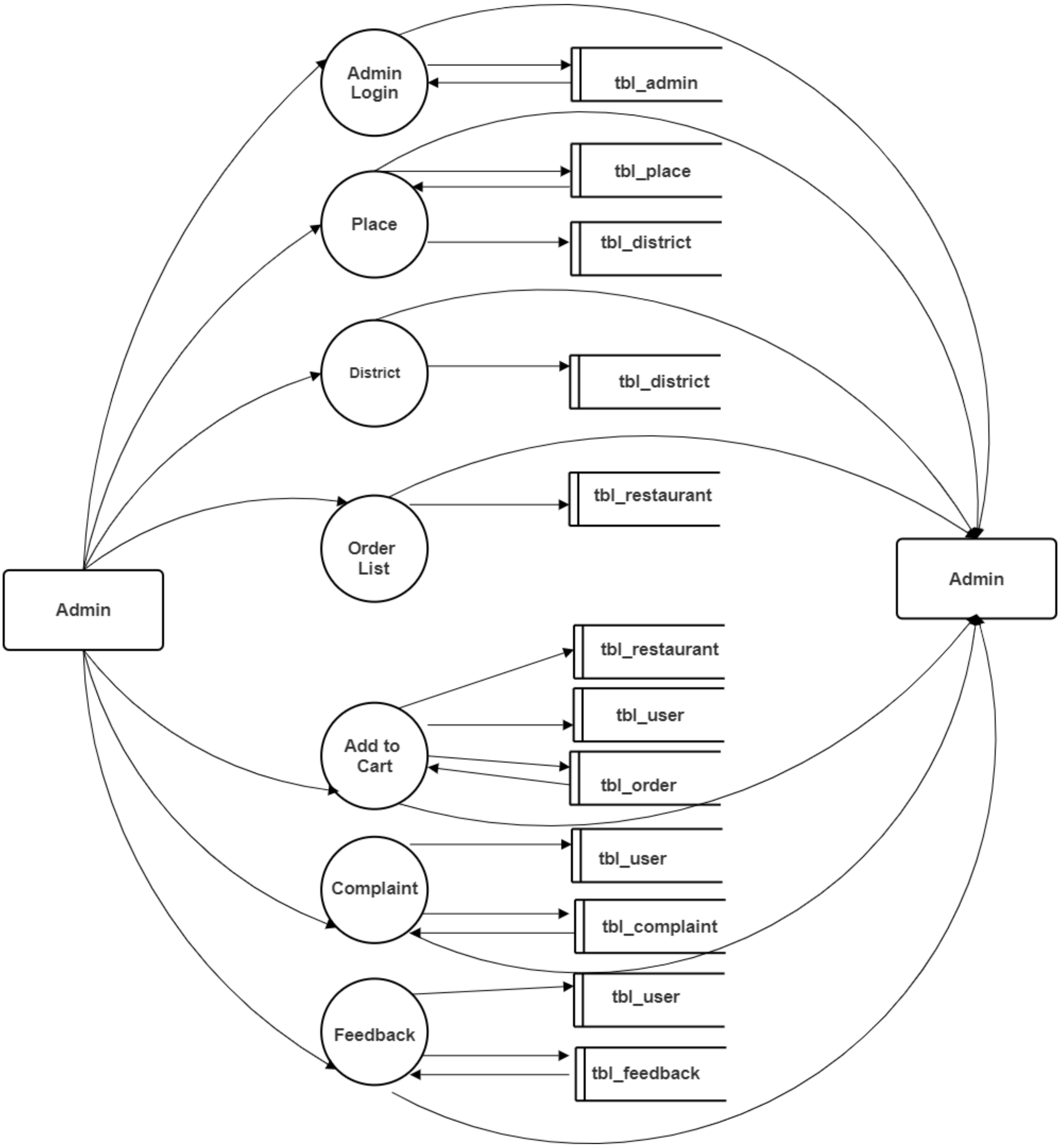
Level 0 of Delivery Boy



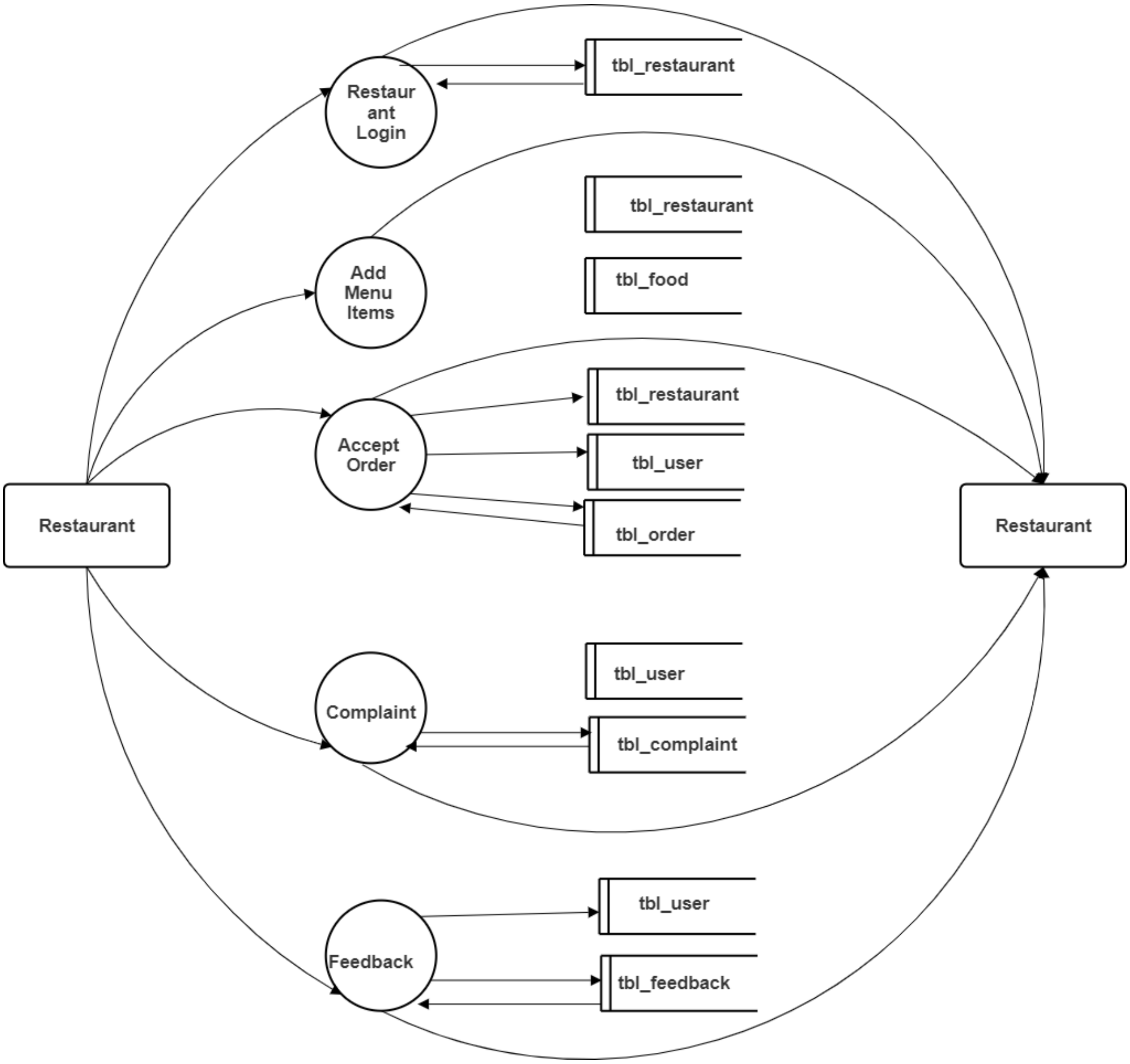
Level 1 User


User
User Login
tbl_user
Place
tbl_place
tbl_district
View Hotel List
tbl_restaurant
tbl_user
View Menu List
tbl_restaurant
tbl_user
tbl_food
Add to Cart
tbl_restaurant
tbl_user
tbl_food
tbl_cart
Complaint
tbl_user
tbl_complaint
Feedback
tbl_user
tbl_feedback
User

Level 1 Admin



Level 1 of Restaurant



Level 1 delivery boy

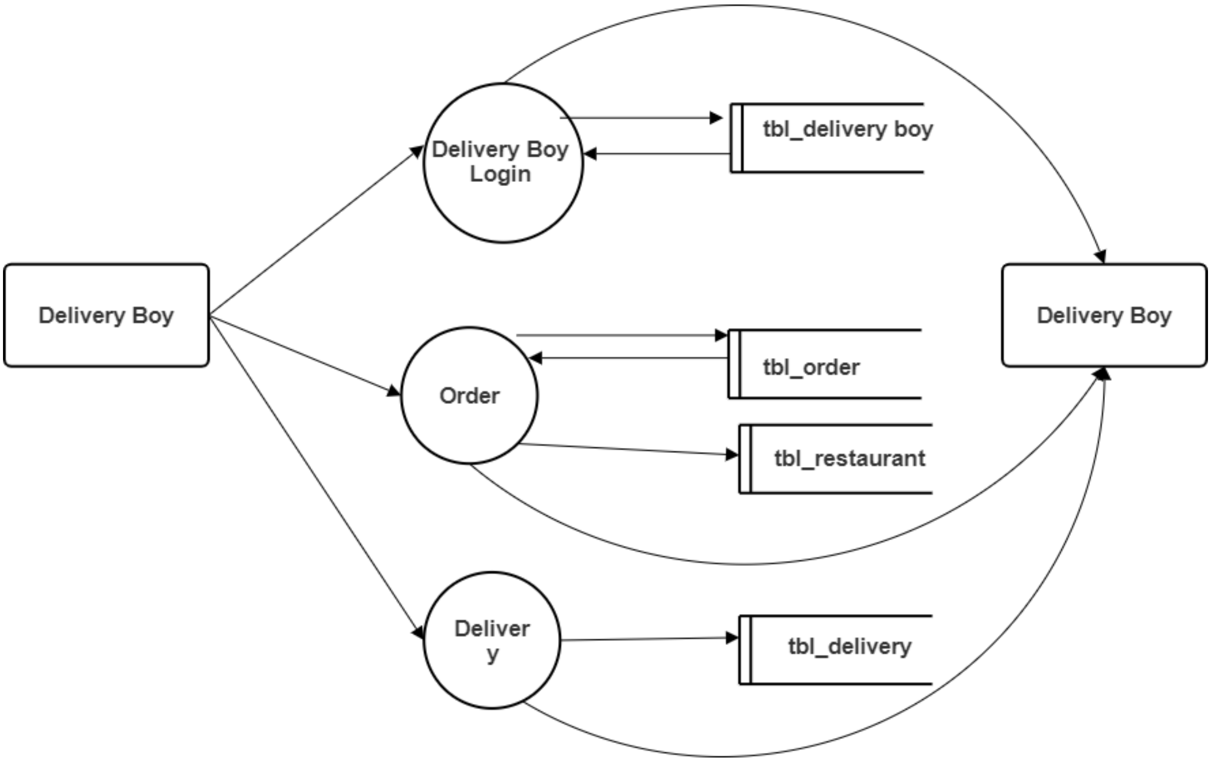


TABLE DESIGN

TABLE DESIGNING

What is Database Designing ?

Designing a proper database reduces the maintenance cost thereby improve data consistency and the cost-effective measures are greatly influenced in terms of disk storage space. Therefore there has to be a brilliant concept of designing a database. The designer should follow the constraints and decide how the elements correlate and what kind of data must be stored. The main objective behind database designing are to produce physical and logical design models of the proposed database system. To elaborate this, the logical model is primarily concentrated on the requirements of data and the consideration must be stored independent of the physical conditions. On the other hand, the physical database design model includes a translation of the logical design model of the database by keep control of physical media using hardware resources and software systems such as Database Management System(DBMS).

There are mainly 16 tables in the project.They are:

1.tbl_admin

2.tbl_district

3.tbl_category

4.tbl_complainttype

5.tbl_place

6.tbl_foodtype

7.tbl_user

8.tbl_restraurant

9.tbl_delivery

10.tbl_food

11.tbl_post

12.tbl_complaint

13.tbl_feedback

14tbl_page

15.tbl_booking

16.tbl_cart

1.tbl_admin

Primary Key:admin_id

Foreign Key:Nill

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	admin_id	INT(20)	Primary Key	Unique Id
2	admin_name	VARCHAR(20)	Not Null	Name of admin
3	admin_email	VARCHAR(30)	Not Null	Email of admin
4	admin_contact	VARCHAR(30)	Not Null	Contact of admin
5	admin_password	PASSWORD	Not Null	Password of admin

2.tbl_district

Primary Key:district_id

Foreign Key:Nill

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	district_id	INT(20)	Primary Key	Unique Id of District
2	district_name	VARCHAR(50)	Not Null	Name of District

3.tbl_category

Primary Key:category_id

Foregin_Key:Nill

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	category_id	INT(20)	Primary Key	Unique Id of Category
2	category_name	VARCHAR(50)	Not Null	Name of Category

4.tbl_complainttype

Primary_key:complaint_id

Foregin_key:Nill

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	complaint_id	INT(20)	Primary Key	Unique Id of complaint
2	complain _type	VARCHAR(100)	Not Null	Title of Complaint

5.tbl_place

Primary Key:place_id

Foreign_key:district_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	place_id	INT(20)	Primary Key	Unique Id of place
2	place_name	VARCHAR(20)	Not Null	Name of Place
3	place_pincode	INT(30)	Not Null	Pincode of Place
4	district_id	INT(20)	Foreign Key	District Id of District

6.tbl_foodtype

Primary_key:foodtype_name

Foreign_key:Nill

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	foodtype_id	INT(20)	Primary Key	Unique Id of foodtype
2	foodtype_name	VARCHAR(20)	Not Null	Name of Foodtype

7.tbl_user

Primary Key:tbl_user

Foreign Key:place_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	user_id	INT(20)	Primary Key	Unique Id of User
2	user_name	VARCHAR(50)	Not Null	Name of User
3	user_gender	VARCHAR(50)	Not Null	Gender of User
4	user_contact	VARCHAR(50)	Not Null	Contact of User
5	user_email	VARCHAR(50)	Not Null	Email of User
6	user_password	PASSWORD	Not Null	Password of User
7	place_id	INT(50)	Foreign Key	Place id of place

8.tbl_restrauant

Primary Key:rest_id

Foreign Key:place_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	rest_id	INT(20)	Primary Key	Unique Id of Restrauant
2	rest_name	VARCHAR(50)	Not Null	Restrauant Name
3	rest_email	VARCHAR(50)	Not Null	Restrauant Email
4	rest_conatct	VARCHAR(50)	Not Null	Restrauant Contact
5	rest_password	PASSWORD	Not Null	Restrauant Password
6	rest_photo	FileField()	Not Null	Restrauant Photo
7	rest_proof	FileField()	Not Null	Restrauant Proof
8	rest_status	INT(20)	Not Null	Restrauant Status
9	Place_id	INT(20)	Foreign Key	Place id of Restrauant

9.tbl_delivery

Primary Key:dboy_id

Foreign Key:place_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	dboy_id	INT(20)	Primary Key	Unique Id of delivery boy
2	dboy_name	VARCHAR(50)	Not Null	Name of Delivery Boy
3	dboy_gender	VARCHAR(50)	Not Null	Gender of Delivery Boy
4	dboy_contact	VARCHAR(50)	Not Null	Contact of Delivery Boy
5	dboy_email	VARCHAR(50)	Not Null	Email of Delivery Boy
6	dboy_password	PASSWORD	Not Null	Password of Delivery Boy
7	dboy_photo	FileField()	Not Null	Photo of Delivery Boy
8	place_id	INT(20)	Foreign Key	Place id of delivery Boy
9	dboy_proof	FileField()	Not Null	Proof of Delivery Boy
10	dboy_status	VARCHAR(50)	Not Null	Deliveryboy status

10.tbl_food

Primary_key:food_id

Foreign_Key:foodtype_id,category_id,rest_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	food_id	INT(20)	Primary Key	Unique Id of food
2	Food_name	VARCHAR(100)	Not Null	Name of Food
3	Foodtype_id	INT(20)	Foreign Key	Id of Foodtype
4	Category_id	INT(20)	Foreign Key	Id of category
5	Rest_id	INT(20)	Foreign Key	Id of Restrauant
6	Food_photo	FileField()	Not Null	Photo of Food
7	Food_price	FileField()	Not Null	Price of Food

11.tbl_post

Primary key:post_id

Foreign key:Nill

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	Post_id	INT(20)	Primary Key	Unique Id of Post
2	Post_caption	VARCHAR(100)	Not Null	Caption of Post
3	Post_description	VARCHAR(440)	Not Null	Description of Post
4	Post_photo	FileField()	Not Null	Photo of food

12.tbl_complaint

Primary Key:complaint_id

Foreign key:user_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	Complaint_id	INT(20)	Primary Key	Unique Id of Complaint
2	Complaint_details	VARCHAR(100)	Not Null	Details of Complaint
3	Complaint-title	VARCHAR(100)	Not Null	Title of Complaint
4	Complaint_postDate	DateField()	Not Null	Date of complaint posted
5	Complaint_reply	VARCHAR(100)	Not Null	Replay of Complaint
6	Complaint_replyDate	DateField()	Not Null	Reply Date of Complaint
7	Complaint_status	VARCHAR(100)	Not Null	Status of Complaint
8	User_id	INT(20)	Foreign Key	Id of User

13.tbl_feedback

Primary Key:feeback_id

Foreign Key:user_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	Feeback_id	INT(20)	Primary Key	Unique Id of feedback
2	Feedback_subject	VARCHAR(100)	Not Null	Subject of Feedback
3	Feeback_details	VARCHAR(100)	Not Null	Details of Feedback
4	Feedback_postdate	DateField()	Not Null	PostDate of Feedback
5	Feedback_status	VARCHAR(10)	Not Null	Status of Feedback
6	User_id	INT(20)	Foreign Key	Id of user

14.tbl_page

Primary Key:page_id

Foreign key:user_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	Page_id	INT(20)	Primary Key	Unique Id of page
2	Page_name	VARCHAR(100)	Not Null	Name of Page
3	Page_bio	VARCHAR(100)	Not Null	Bio of page
4	Page_doj	DateField()	Not Null	Doj of page
5	Page_photo	FileField()	Not Null	Photo of page
6	User_id	INT(20)	Foreign Key	Id of user

15.tbl_booking

Primary Key:booking_id
Foreign Key:user_id,dboy_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	Booking_id	INT(20)	Primary Key	Unique Id Booking
2	Booking_amount	VARCHAR(100)	Not Null	Booking Amount
3	Booking_datetime	DateField()	Not Null	Date and Time of Booking
4	Booking_status	INT(10)	Not Null	Status of Booking
5	User_id	INT(20)	Foreign Key	Id of user
6	Dboy_id	INT(20)	Foreign Key	Id of Delivery Boy

16.tbl_cart

Primary Key:cart_id

Foreign_key:booking_id,food_id

SL NO	NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
1	Cart_id	INT(20)	Primary Key	Unique Id of cart
2	Cart_qty	VARCHAR(100)	Not Null	Qty of cart
3	Cart_status	INT(20)	Not Null	Status of cart
4	Food_id	INT(20)	Foreign Key	Id of food
5	Booking_id	INT(20)	Foreign Key	Id of booking

SYSTEM DESIGN

8. SYSTEM DESIGN

The goal of design process is to produce a model or a representation of a moving from the problem domain to the solution domain. In top level design focus is on deciding which modules are needed for the system, the specification of these modules and how these modules can be interconnected. In this project design technique used is top-down, object-oriented dynamic modelling technique. A top-down design approach starts by identifying the major components and iterating until the desired level of details is achieved. In objectoriented design technique, the modules in the design represent data abstraction.

8.1 INPUT DESIGN

Input design is the process of converting user-oriented inputs to a computer-based format. The quality of system determines the quality of system outputs. All the data entry screens should be of interactive nature so that the user can directly input data according to prompt messages. The input design determines whether the user can interact directly with computer. Interactive input screens ensure the reliability and accuracy of the system. The goal of designing input data is to make it free from logical errors. The input data is also used for easy calculation of necessary functions.

Different forms are used for data screens in order to input data into the system. The screen formats have options like add, edit and update with buttons, which is used for easy input and retrieving information. All the fields are validated. If the user enters invalid data appropriate messages are displayed. Usually, an incorrect output is produced when the users provide incorrect inputs. The user can select a record by mouse click. Then the output screen displays the selected record.

8.2 OUTPUT DESIGN

One of the most important features of the system for users is the output it produces. Output design should improve the system relationship with the user and helps in decision making. Without quality output, the entire system appears to be unnecessary that users will avoid using. The objective of the output design is to define the control and formats of all printed, documented, reports and screens that will be produced by the system. Computer output is the most important and direct source of information to the user. For many end user's output is the main reason for developing the system and the basis on which they will evaluate the usefulness of the application. Output generally refers to the results that are generated by the system. The user must give a valid input to get an accurate output. The outputs are the total number of function points, total effort required, total time required for development and scheduling to develop that project.

SYSTEM TESTING AND IMPLEMENTATION

9.TESTING

Coding conventions are a set of guidelines for a specific programming language that recommend programming style, practices and methods for each aspect of a piece program written in this language. These conventions usually cover file organization, indentation, comments, declarations, statements, white space, naming conventions, programming practices, programming principles, programming rules of thumb, architectural best practices, etc. These are guidelines for software structural quality. Software programmers are highly recommended to follow these guidelines to help improve the readability of their source code and make software maintenance easier.

9.1 TEST CASES

The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together. The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results with the expected results. The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user must verify the system and give the go-ahead. During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way, users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results. A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferably to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by person other than those who write the original programs. Using persons who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software. Parallel running is often regarded as the final phase of system testing. Since the parallel operation of two systems is very demanding in terms of user resources it should be embarked on only if the user is satisfied with the results of testing, it should not be started if problems are known to exist. Testing is the major quality control measure during software development. Its basic function is to detect errors in the software. Thus, the goal of testing is to uncover requirement design and coding errors in the program. Testing is the process of correcting a program with intends of finding an error.

Different types of testing are,

1. Unit Testing
2. Integrated Testing
3. Validation Testing
4. Output Testing
5. User Acceptance Testing

9.1.1 UNIT TESTING

In computer programming, unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they are fit for use. In this testing we test each module individual and integrated the overall system. Unit testing focuses verification efforts on the smaller unit of software design in the module. This is also known as module testing. The modules of the system are tested separately. The testing is carried out during programming stage itself. In this testing step each module is found to working satisfactory as regard to the expected output from the module. There are some validation checks for verifying the data input given by the user which both the formal and validity of the entered. It is very easy to find error debug the system. These types of tests are usually written by developers as they work on code, to ensure that the specific function is working as expected. Unit testing is a software development process that involves synchronized application of a broad spectrum of defect prevention and detection strategies in order to reduce software development risks, time, and costs. It is performed by the software developer or engineer during the construction phase of the software development lifecycle.

9.1.2 INTEGRATION TESTING

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group.

Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. Data can be lost across an interface; one module can have an adverse effort on the other sub functions when combined by, may not produce the desired major functions. Integrated testing is the systematic testing for constructing the uncover errors within the interface. This testing was done with sample data. The developed system has run success full for this sample data. The need for integrated test is to find the overall system performance. Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. A component, in this sense, refers to an integrated aggregate of more than one unit. Integration testing identifies problems that occur when units are combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining units, you know that any errors discovered when combining units are likely related to the interface between units. This method reduces the number of possibilities to a far simpler level of analysis. Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

9.1.3 VALIDATION TESTING

At the culmination of Black Box testing, software is completely assembled as a package, interface errors have been uncovered and corrected and final series of software tests, Validation tests begins. Validation testing can be defined many was but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.

After validation test has been conducted one of the two possible conditions exists.

1) The function or performance characteristics confirm to specification and are accepted.

2) A derivation from specification uncovered and a deficiency list is created.

9.1.4 USER ACCEPTANCE TESTING

Acceptance Testing is a level of the software testing process where a system is tested for acceptability. User Acceptance testing is the software testing process where system tested for acceptability & validates the end-to-end business flow. Such type of testing executed by client in separate environment & confirms whether system meets the requirements as per requirement specification or not. UAT is performed after System Testing is done and all or most of the major defects have been fixed. This testing is to be conducted in the final stage of Software Development Life Cycle (SDLC) prior to system being delivered to a live environment. UAT users or end users are concentrating on end-to-end scenarios & typically involves running a suite of tests on the completed system. User Acceptance testing also known as Customer Acceptance testing (CAT), if the system is being built or developed by an external supplier. The CAT or UAT are the final confirmation from the client before the system is ready for production. The business customers are the primary owners of these UAT tests. These tests are created by business customers and articulated in Business domain languages. So ideally it is collaboration between business customers, business analysts, testers and developers. It consists of test suites which involve multiple test cases & each test case contains input data (if required) as well as the expected output. The result of test case is either a pass or fail. Unit testing is the testing of individual forms, like testing of form seat. Here check save, edit, delete, cancel buttons are work or not, if edit and delete is not visible when we save a value, if save and cancel is not visible when we edit. There is another important matter is that validation checking. Check and confirm we can’t save same value more than one time. In integration testing, check the entire project and confirm all forms are integrated. Also test prevention of deletion of values when that value is used in another form.

9.2 TEST REPORTS

Test Report provides a summary of the results of test performed. A Test report must contain the following details: -

- Test Summary: This must include basic information about what was tested and what happened.
- Test Type: This must include basic information about what type of testing (unit testing, integration testing, validation testing etc.) was done and what happened.
- Test Assessment: It should contain a comprehensive assessment of your interpretation of how adequate the test was in light of how thorough the test plan said it should be? It must also specify what wasn't tested well enough.
- Test Results: Summarize the test results. Include a detailed description of any deviations from the original test plan, design, test case, or expected results. Include any issues or bugs discovered during the test.
- Variance: Describe any variances between the testing that was planned and the testing that actually occurred. Also, provide an assessment of the manner in which the test environment may be different from the operational environment and the effect of this difference on the test results.

- Test Instances: Provide a brief description of the unexpected results, problems, or defects that occurred during the testing.

9.3 IMPLEMENTATION

The implementation is the final stage and it is an important phase. It involves the individual programming; system testing, user training and the operational running of developed proposed system that constitutes the application subsystems. A major task of preparing for implementation is education of users, which should really have been taken place much earlier in the project when they were being involved in the investigation and design work.

During the implementation phase system actually takes physical shape. In order to develop a system implemented planning is very essential. The implementation phase of the software development is concerned with translating design specialization into source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from old system to new system. The system can be implemented only after testing is done and is found to be working to specifications. The implementation stage is a systems project in its own right. The implementation stage involves following tasks:

- Careful planning
- Investigation of system and constraints
- Design of method to achieve change over
- Evaluation of the changeover method

In the case of “FOOD SQUARE”, all the screens are designed first. For making it to be run able, codes are written on each screen and performs the implementation by creating the database and connecting to the server. After that the system is checked, whether it performs the transactions correctly. Then databases are cleared and it to be useable to the technicians.

MAINTENANCE

10. MAINTENANCE

Maintenance activity may require the continuing involvement of a large proportion of computer department resources. For computer installations, which have already developed the basic applications for the organization, the main task may be to adapt existing system in a changing environment. Perhaps a better term to describe this activity is system evolution. All systems are dynamic and subject to constantly changing requirements. Efforts must be devoted to adapting them and design should be flexibly specified so that changes are easily implemented. Most changes arise in two ways. A part of the normal running of the system when errors are found, users ask for improvement or external requirements changes and as result of specific investigation and review of the system's performance.

CONCLUSION

11.CONCLUSION

In conclusion, Food Square is more than just a food delivery service – it's a gateway to culinary exploration and convenience. With our diverse selection of top-rated restaurants, easy-to-use platform, and commitment to quality and sustainability, we're redefining the way people experience dining.

Whether you're craving comfort classics, international flavors, or healthy options, Food Square has something for everyone. Our dedication to customer satisfaction means that every meal is delivered fresh, hot, and delicious, right to your doorstep.

Join us on a culinary journey like no other and let Food Square be your go-to destination for memorable dining experiences. With just a few clicks, you can discover new flavors, support local businesses, and indulge in the joy of great food, anytime, anywhere. Welcome to Food Square – where every meal is a celebration of taste, convenience, and culinary delight.

BIBLIOGRAPHY

11. BIBLIOGRAPHY

Visual Studio Code	- VSCode
Google Chrome	- Google
Microsoft Edge	- Microsoft

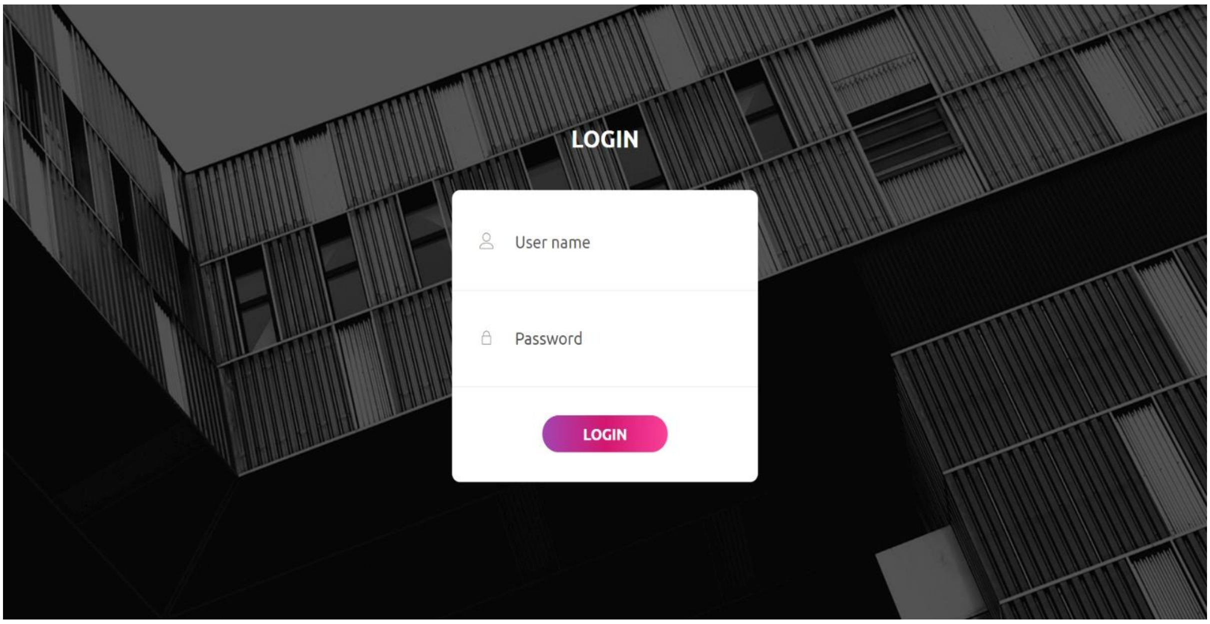
REFERENCES

12. REFERENCES

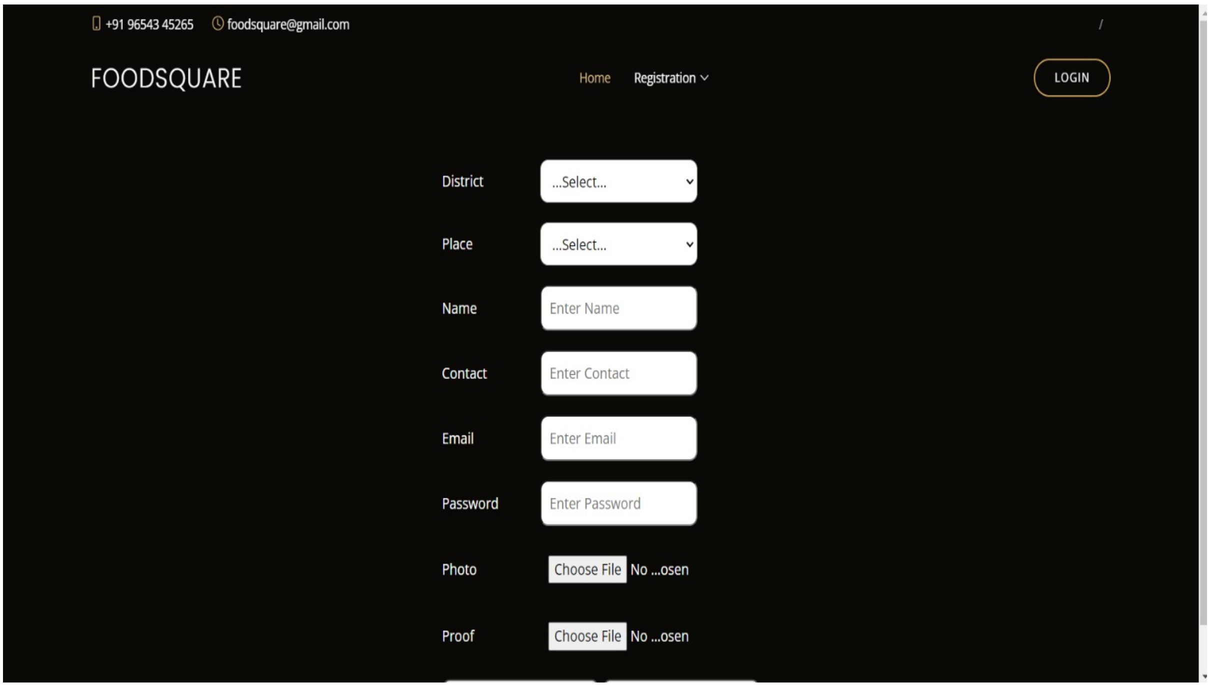
1. https://www.youtube.com/
2. https://www.wikipedia.org/
3. https://codepen.io/
4. https://www.w3schools.com/
5. https://www.javatpoint.com/
6. https://stackoverflow.com/

SCREENSHOTS

Login Page

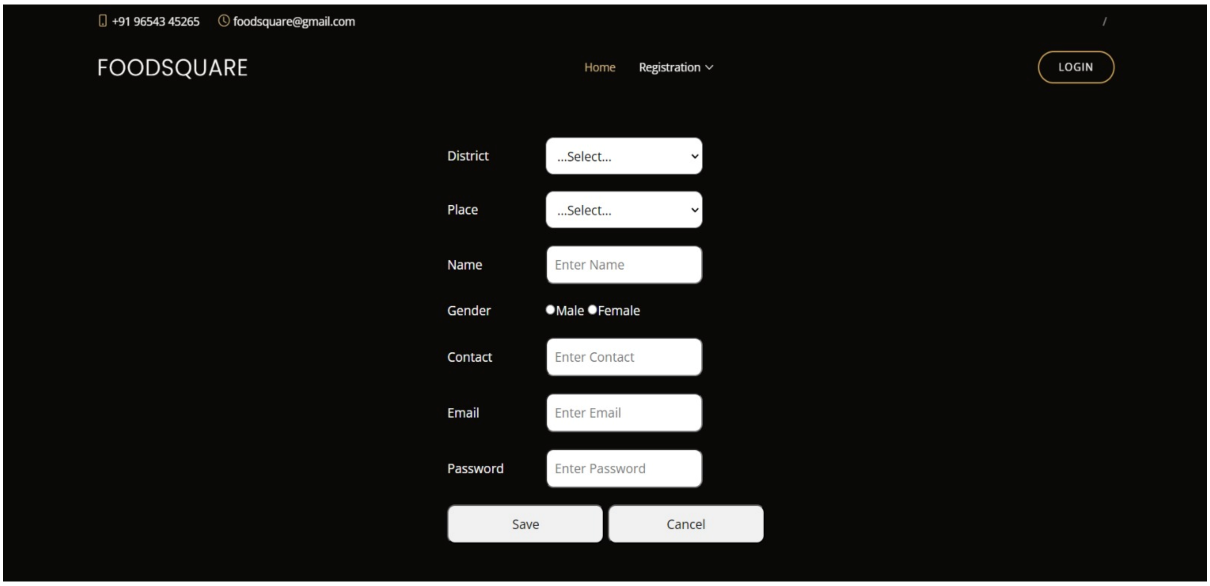


Restaurant Registration



District	...Select...
Place	...Select...
Name	Enter Name
Contact	Enter Contact
Email	Enter Email
Password	Enter Password
Photo	Choose File No ...osen
Proof	Choose File No ...osen

Delivery Boy Registration



District ...Select...

Place ...Select...

Name Enter Name

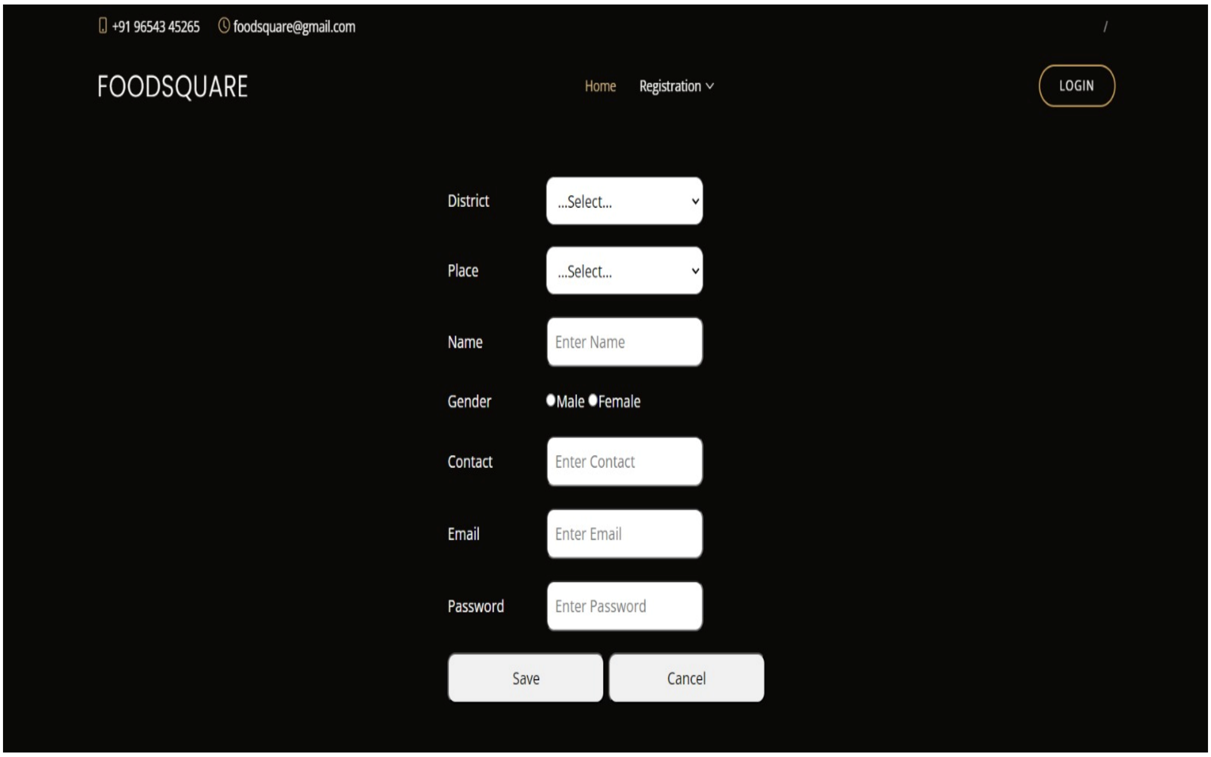
Gender ●Male ●Female

Contact Enter Contact

Email Enter Email

Password Enter Password

User Registration



District ...Select...

Place ...Select...

Name Enter Name

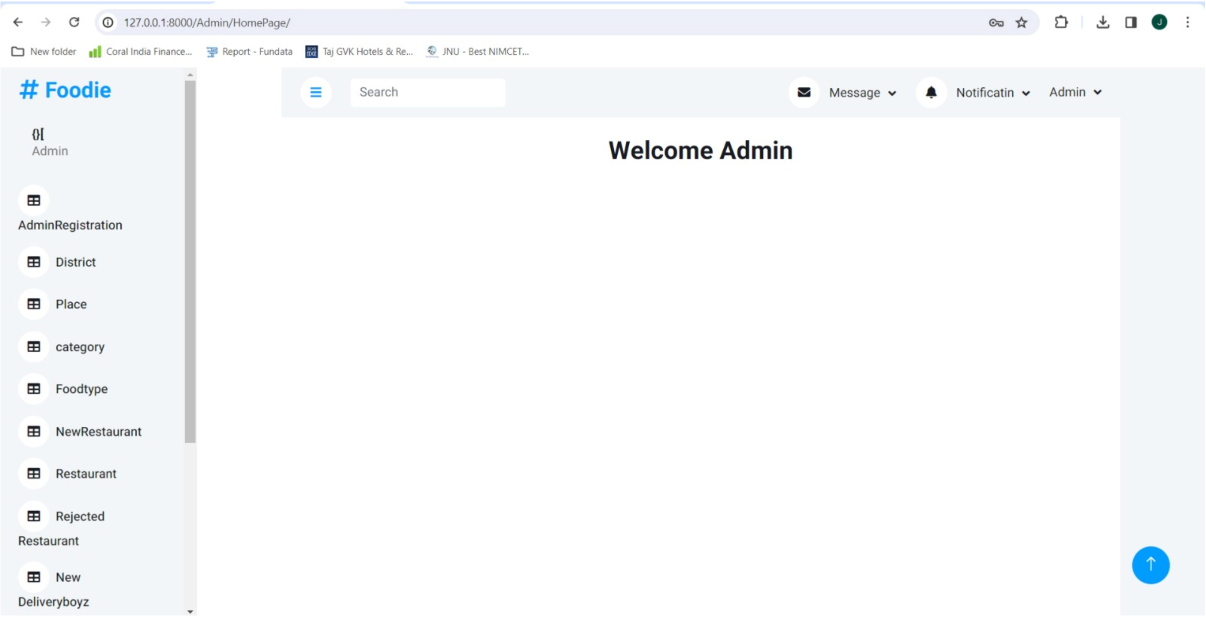
Gender ●Male ●Female

Contact Enter Contact

Email Enter Email

Password Enter Password

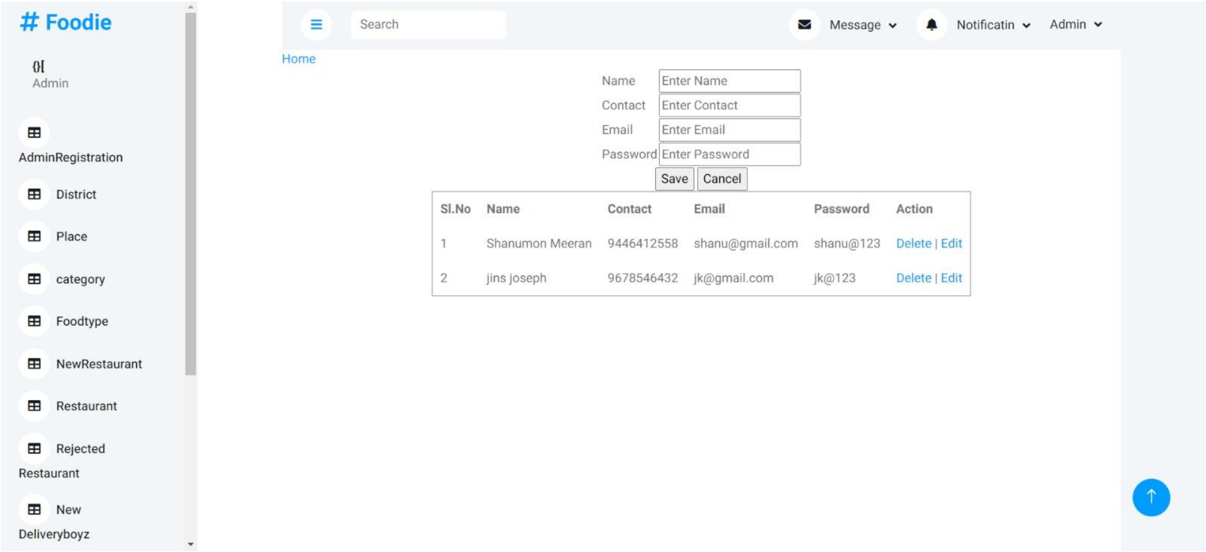
Admin Homepage



127.0.0.1:8000/Admin/HomePage/

Welcome Admin

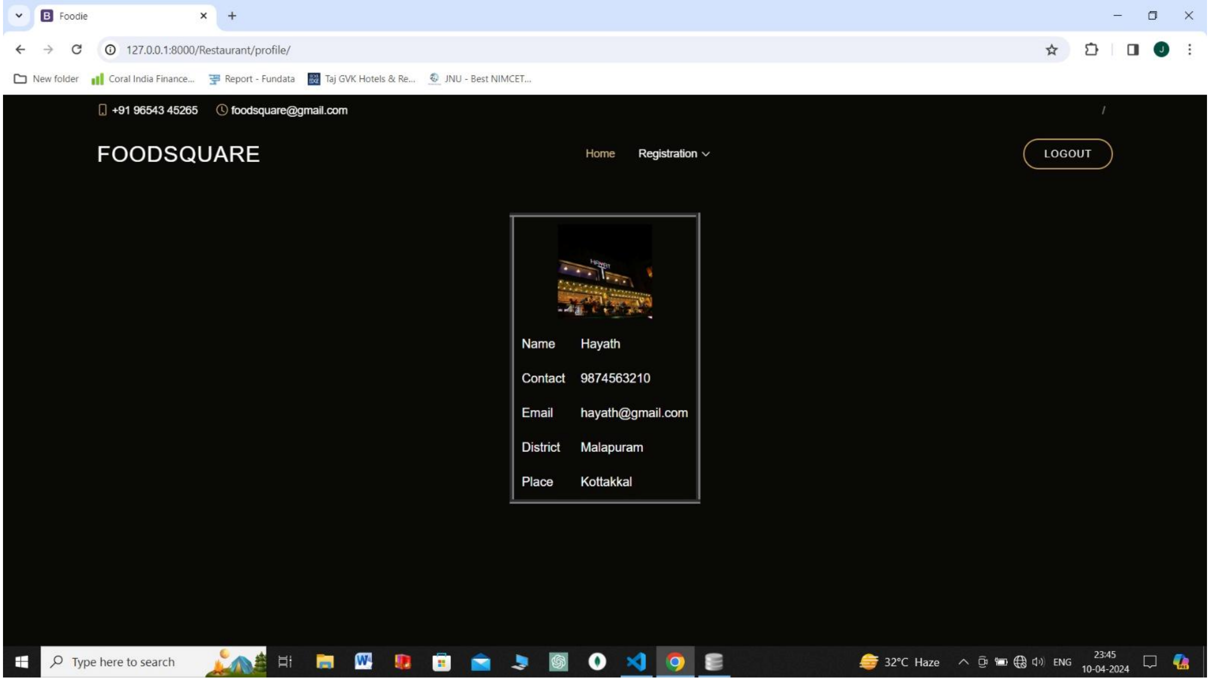
Admin Registration



Name Enter Name
Contact Enter Contact
Email Enter Email
Password Enter Password
Save Cancel

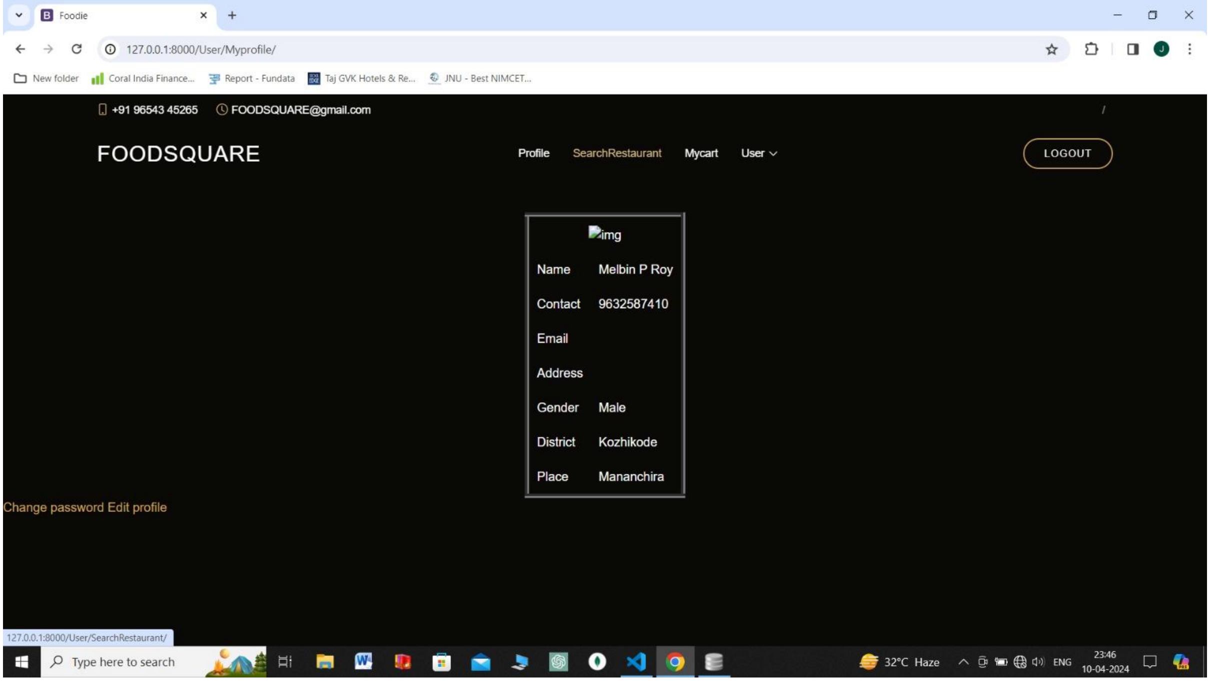
Sl.No	Name	Contact	Email	Password	Action
1	Shanumon Meeran	9446412558	shanu@gmail.com	shanu@123	Delete | Edit
2	jins joseph	9678546432	jk@gmail.com	jk@123	Delete | Edit

Restaurant Registration



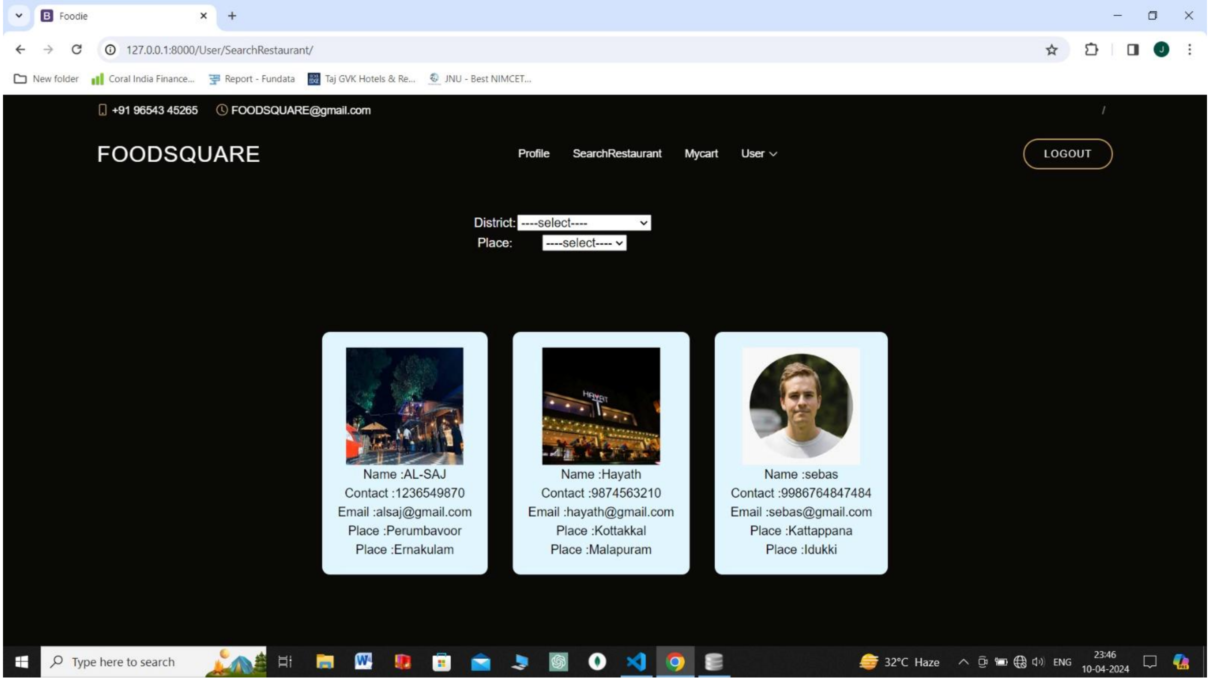
Name	Hayath
Contact	9874563210
Email	hayath@gmail.com
District	Malapuram
Place	Kottakkal

User Profile



Name	Melbin P Roy
Contact	9632587410
Email	
Address	
Gender	Male
District	Kozhikode
Place	Mananchira

Searching Restaurant



District: ----select----

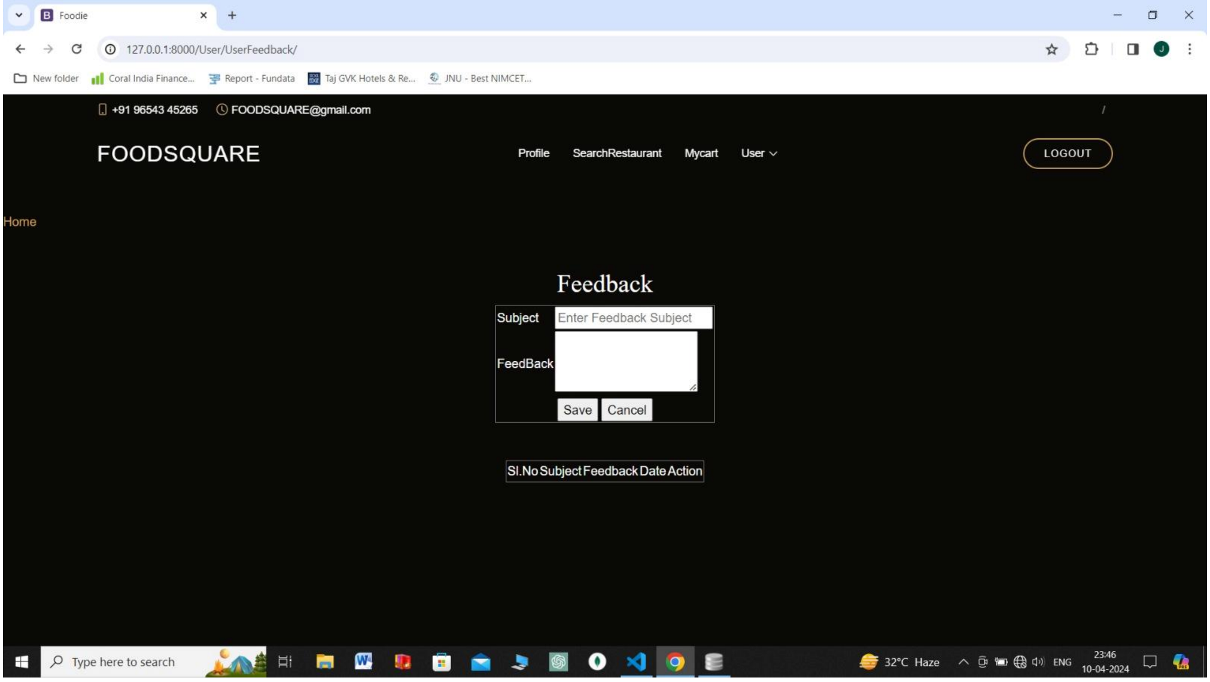
Place: ----select----

Name :AL-SAJ
Contact :1236549870
Email :alsaj@gmail.com
Place :Perumbavoor
Place :Ernakulam

Name :Hayath
Contact :9874563210
Email :hayath@gmail.com
Place :Kottakkal
Place :Malapuram

Name :sebas
Contact :9986764847484
Email :sebas@gmail.com
Place :Kattappana
Place :Idukki

User Feedback



Feedback

Subject: Enter Feedback Subject

FeedBack:

Sl.No	Subject	Feedback	Date	Action