**A**

***PROJECT REPORT***

*On*

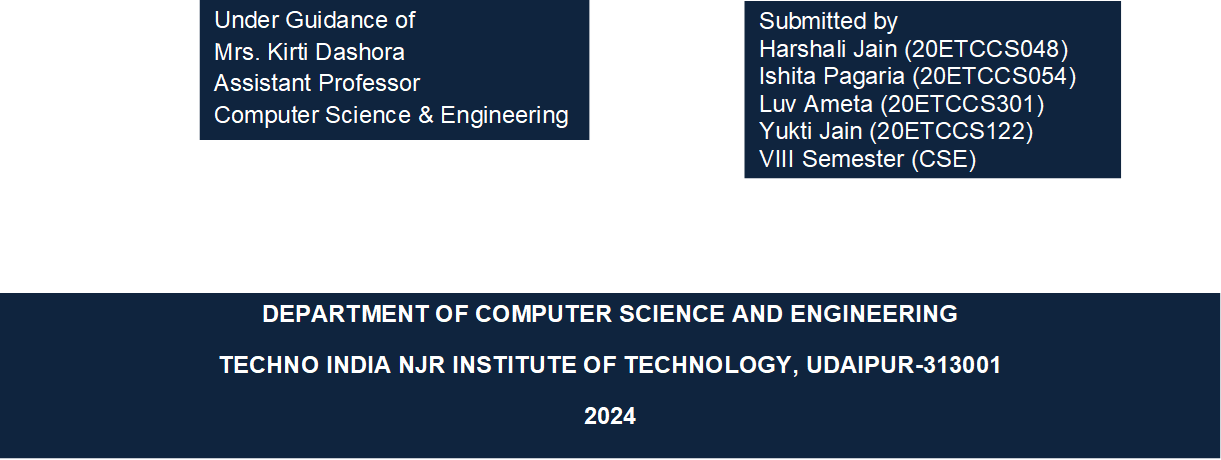
**ONLINE FOOD ORDERING SYSTEM**

*Submitted in partial fulfilment of the requirements for the degree of*

**BACHELOR OF TECHNOLOGY**

****

Session: 2024



**A**

***PROJECT REPORT***

*on*

**ONLINE FOOD ORDERING SYSTEM**

*Submitted in partial fulfilment of the requirements for the degree of*

**BACHELOR OF TECHNOLOGY**



**Session: 2024**

Under Guidance of: Submitted by:

**Mrs. Kirti Dashora** Harshali Jain (20ETCCS048)

Assistant Professor Ishita Pagaria (20ETCCS054)

Computer Science Luv Ameta (20ETCCS301)

& Engineering Yukti Jain (20ETCCS122)

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001**

**2024**



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**CERTIFICATE**

This is to certify that project work titled **ONLINE FOOD ORDERING SYSTEM** by *Harshali Jain* was successfully carried out in the Department of Computer Science and Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in

Computer Science and Engineering.

**Guide Mrs. Kirti Dashora HoD Dr. Rimpy Bishnoi**

Assistant Professor Head of Department

Computer Science Dept. of CSE TINJRIT, Udaipur

& Engineering



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**CERTIFICATE**

This is to certify that project work titled **ONLINE FOOD ORDERING SYSTEM** by *Ishita Pagaria* was successfully carried out in the Department of Computer Science and Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in Computer Science and Engineering.

**Guide Mrs. Kirti Dashora HoD Dr. Rimpy Bishnoi**

Assistant Professor Head of Department

Computer Science Dept. of CSE TINJRIT, Udaipur

& Engineering



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**CERTIFICATE**

This is to certify that project work titled **ONLINE FOOD ORDERING SYSTEM** by *Luv Ameta* was successfully carried out in the Department of Computer Science and Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in Computer Science and Engineering.

**Guide Mrs. Kirti Dashora HoD Dr. Rimpy Bishnoi**

Assistant Professor Head of Department

Computer Science Dept. of CSE TINJRIT, Udaipur

& Engineering



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**CERTIFICATE**

This is to certify that project work titled **ONLINE FOOD ORDERING SYSTEM** by *Yukti Jain* was successfully carried out in the Department of Computer Science and Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in Computer Science and Engineering.

**Guide Mrs. Kirti Dashora HoD Dr. Rimpy Bishnoi**

Assistant Professor Head of Department

Computer Science Dept. of CSE TINJRIT, Udaipur

& Engineering



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**Examiner Certificate**

This is to certify that the following students **Harshali Jain**, **Ishita Pagaria**, **Luv Ameta**, **Yukti Jain** of final year B.Tech. (Computer Science and Engineering), was examined for the

project work titled ***“Online Food Ordering System”*** during the academic year 2022 – 2023 at Techno India NJR Institute of Technology, Udaipur.

**Remarks:**

**Date:**

Signature Signature

**(Internal Examiner)**  **(External Examiner)**

Name : - ……………………… Name: - ……………………

Designation: - ……………….. Designation: - ……………….

Department: - ………………. Department: - ……………….

Organization: - …………… Organization: - ………………

**PREFACE**

Welcome to our guide on developing an Online Food Ordering Website. This project explores the creation of a platform that meets the growing demand for online food services.

1. Introduction: Discusses the project's purpose, objectives, and the significance of online food ordering.

2. Software Requirement Specification: Outlines the specific requirements and functionalities of the website.

3. System Design: Examines the architecture and structure of the website, detailing its design principles and components.

4. Project Planning: Provides a breakdown of the necessary software and technologies for development.

5. Testing: Covers various testing methodologies to ensure the reliability and functionality of the website.

6. Screenshots of the Project: Offers visual representations of the website's user interface and features.

7. Conclusion: Reflects on the project's journey, challenges, achievements, and potential future enhancements.

**ACKNOWLEDGMENT**

We take this opportunity to record our sincere thanks to all who helped us to successfully complete this work. Firstly, we are grateful to our **Supervisor Mrs. Kirti Dashora** for her invaluable guidance and constant encouragement, support and most importantly for giving us the opportunity to carry out this work.

We would like to express our deepest sense of gratitude and humble regards to our **Head of Department Dr. Rimpy Bishnoi** for giving invariable encouragement in our endeavors and providing necessary facilities for the same. Also, a sincere thanks to all faculty members of CSE, TINJRIT for their help in the project directly or indirectly.

Finally, we would like to thank our friends for their support and discussions that have proved very valuable for us. We are indebted to our parents for providing constant support, love and encouragement. We thank them for the sacrifices they made so that we could grow up in a learning environment. They have always stood by us in everything we have done, providing constant support, encouragement and love.

**Harshali Jain (20ETCCS048)**

**Ishita Pagaria (20ETCCS054)**

**Luv Ameta (20ETCCS301)**

**Yukti Jain (20ETCCS122)**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Certificate** | ………………………………………………………………. | ii - v |
| **Preface** ………..…………………………………………………............ | | vi |
| **Acknowledgement** …………………………………………………. | | vii |
| **Table of Contents** ………………………………………………………. | | viii – ix |
| **List of Figures** …………………………………………………………… | | x |
| **List of Tables** ……………………………………………………………. | | xi |
| **Chapter 1** | INTRODUCTION ………………………………………. | 12 |
|  | * 1. Objective ………….……………………………. | 13 |
|  | 1.2 Functionalities ….……………………………………. | 13 |
|  | 1.3 Scope……………………...…………………………… | 13 |
|  | 1.4 Modules………………………………………………... | 14 |
|  | 1.5 Input Data and Validation……………………………. | 14 |
|  | 1.6 Features ………………………………………………. | 15 |
| **Chapter 2** | SOFTWARE REQUIREMENT SPECIFICATION……… | 16 |
|  | * 1. Identifying the Need ……………...…………………... | 16 |
|  | * 1. Feasibility Study …………….………………………… | 17 |
| **Chapter 3** | SYSTEM DESIGN ….……………….………………….... | 18 |
|  | 3.1 General Tasks ………………………...……………… | 18 |
|  | 3.2 User Interface Design ……………………………….. | 19 |
|  | 3.3 Preliminary Product Description ……………………. | 20 |
|  | 3.4 Project Category (RDBMS) ………………………… | 21 |
|  | 3.5 Implementation Methodology ………………………. | 23 |
|  | 3.6 Cost Estimation of the Project ……………………… | 23 |
|  | 3.7 Project Scheduling …………………………………… | 23 |
| **Chapter 4** | PROJECT PLANNING………………………… | 25 |
|  | 4.1 Project Profile ………………………………………… | 25 |
|  | 4.2 PERT Chart …………………………………………… | 27 |
|  | 4.3 Use Case Model ……………………………………… | 28 |
|  | 4.4 Data Flow Diagram …………………………………... | 29 |
|  | 4.5 ER Diagram …………………………………………... | 31 |
| **Chapter 5** | TESTING PROJECT …………………………………….. | 34 |
|  | 5.1 Types of Testing ……………………………………… | 34 |
|  | 5.2 Detailed design of implementation …………………. | 35 |
|  | 5.3 Steps in Software Testing …………………………… | 36 |
|  | 5.4 System Analysis ……………………………………… | 38 |
|  | 5.5 Data Dictionary ……………………………………….. | 39 |
| **Chapter 6** | SCREENSHOTS OF THE PROJECT ………………….. | 41 |
| **Chapter 7** | CONCLUSION ……………………………………………. | 47 |
|  | 7.1 Future Scope …………………………………………. | 47 |
|  | 7.2 Limitations …………………………………………….. | 48 |
| **Chapter 8** | REFERENCES & BIBLIOGRAPHY …………………….. | 50 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| Fig 1 | RDBMS ……………………...……………………………….. | 21 |
| Fig 2 | MVC Diagram …………………….…………………………. | 22 |
| Fig 3 | Use Case Diagram ………………………………………….. | 26 |
| Fig 4 | Class Diagram ………………………………………………. | 26 |
| Fig 5 | PERT Chart ………………………………………………….. | 28 |
| Fig 6 | Data Flow Diagram …………………………………………. | 29 |
| Fig 7 | ER Diagram ………………………………………………….. | 32 |
| Fig 8 | Symbols in ER Diagram ……………………………………. | 33 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| Table 1 | Gantt chart ………………………...…………………... | 24 |
| Table 2 | Software Requirements.….…………………………… | 25 |
| Table 3 | Symbols in DFD………………………………………... | 30 |

**INTRODUCTION**

We created an Online Food Ordering System to fix the problems we were having with the old manual system. This new software aims to make things easier and smoother for everyone involved. We've worked hard to make sure it's simple to use and reduces the chances of mistakes.

One of the best things about this system is that it tells you when something doesn't seem right, like if you type in the wrong information. You don't need any special training to use it – it's designed to be friendly and easy for anyone. By using this system, we hope to make managing orders, payments, and everything else related to food ordering much quicker and more reliable. It'll free up time for everyone involved so they can focus on other important tasks.

We understand that every business is unique, so our system can be tailored to fit your specific needs. Whether you're a small local restaurant or a large chain, we've got you covered. Our goal is to help you plan ahead and make sure you have all the information you need to succeed. And for those of you who are always on the move, our system has a feature that lets you access it from anywhere. That way, you can stay on top of things no matter where you are.

The Online Food Ordering System is all about making life easier by switching from manual to computerized systems. With the help of computers and specialized software, we aim to store and manage important data more efficiently and securely.

We've made sure that the software and hardware needed for this system are easy to find and use. Our goal is to create a management system that's error-free, reliable, and fast, so you can focus on running your business smoothly.

By automating tasks like record-keeping, we want to free up your time and energy for other important activities. You won't have to worry about redundant entries or getting bogged down with irrelevant information – our system keeps things organized and accessible.

Ultimately, our aim is to improve performance and provide better service to our clients. We want to make managing your food orders as hassle-free as possible, so you can focus on what really matters.

***Objective of the project***

The main objective of the Project on Online Food Ordering System is to manage the details of Food Item, Category, Customer, Order, Confirm Order. It manages all the information about Food Items, Payment, Confirm Order, Food Item. The project is totally built at the administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Food Item, Category, Payment, Customer. It tracks all the details about the Customer, Order, Confirm Order.

***Functionalities provided by Online Food Ordering System are as follows:***

* Provides the searching facilities based on various factors, such as food item, customer, order, and confirmed orders.
* Online food ordering system also manages the payment details online for order details, confirm order details, food item.
* It tracks all the information of category, payment, order etc.
* Shows the information and description of the food item, customer to increase the efficiency of managing the food item, category
* It deals with monitoring the information and transactions of order.
* Manages the information of food item.
* Editing, adding and updating of records is improved which results in proper resource management of food item data.
* Manage the information of order.
* Integration of all records of confirm order.

# ***Scope***

# It may help collect perfect management in detail. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of the past year perfectly and vividly. It also helps in current work relative to the Online Food Ordering System. It will also reduce the cost of collecting the management & collection procedure will go on smoothly. Our project aims at Business process automation, i.e. we have tried to computerize various processes of Online Food Ordering System.

# ***Modules of Online Food Ordering System***

* Food Item Management Module: Used for managing the Food Item details.
* Confirm Order Module : Used for managing the details of Confirm Order
* Payment Module : Used for managing the details of Payment
* Category Management Module: Used for managing the information and details of the Category.
* Customer Module : Used for managing the Customer details
* Order Module : Used for managing the Order informations
* Login Module: Used for managing the login details
* Users Module : Used for managing the users of the system

***Input Data and Validation of Project on Online Food Ordering System***

* All the fields such as Food Item, Customer, Confirm Order are validated and does not take invalid values
* Each form for Food Item, Category, Payment cannot accept blank value fields
* Avoiding errors in data
* Controlling amount of input
* Integration of all the modules/forms in the system.
* Preparation of the test cases.
* Preparation of the possible test data with all the validation checks.
* Actual testing done manually.
* Recording of all the reproduced errors.
* Modifications done for the errors found during testing.
* Prepared the test result scripts after rectification of the errors.
* Functionality of the entire module/forms.
* Validations for user input.
* Checking of the Coding standards to be maintained during coding.
* Testing the module with all the possible test data.
* Testing of the functionality involving all types of calculations etc.
* Commenting standard in the source files.

***Features***

* Product and Component based
* Creating & Changing Issues at ease
* Query Issue List to any depth
* Reporting & Charting in more comprehensive way
* User Accounts to control the access and maintain security
* Simple Status & Resolutions
* Multi-level Priorities & Severities.
* Targets & Milestones for guiding the programmers
* Attachments & Additional Comments for more information
* Robust database back-end
* Various level of reports available with a lot of filter criteria, it contain better storage capacity.
* Accuracy in work.
* Easy & fast retrieval of information.
* Well-designed reports.
* Decrease the load of the person involved in the existing manual system.
* Access of any information individually.
* Work becomes very speedy.
* Easy to update information

**SOFTWARE REQUIREMENT SPECIFICATION**

The Software Requirements Specification comes after analyzing everything. It takes the tasks and makes them simpler by describing information clearly, detailing how things work, stating what the software needs to do, setting performance standards, and listing any limits on design. It also includes ways to check if everything meets the requirements and other important data.

*Here are the requirements for the proposed system:*

1. The system should be able to store information about new food items.
2. It should help staff members keep track of categories and easily find them using different search methods.
3. The system should maintain records of quantities.
4. It needs to keep track of customers.
5. It should allow updating and deleting of records.
6. There should be a search function.
7. It needs a security feature to protect data.

***Identifying the Need***

The old manual system had a lot of problems. Since everything was done by hand, it was hard to keep, organize, and find information. Records were all over the place, making it tough to connect transactions with their contexts. Finding any information meant digging through various papers, and there were no reports available.

Entering and finding records took a long time. It was also hard to spot mistakes while entering data. Once entered, it was tough to update records.

This was because there was so much to remember and manage in the business. That's why the current system, though partially automated, is still tricky. You have to enter the same info in three different places.

***Feasibility Study***

After completing the Online Food Ordering System project and analyzing all its functionalities, the next step is to conduct a feasibility study for the project.

Every project is technically feasible if you have unlimited resources and time. However, a feasibility study looks at all possible ways to solve a problem. The solution should meet user requirements and be flexible enough for future changes.

*A. Economical Feasibility:*

This is crucial for project development. We chose technology based on minimizing costs. The organization will cover all hardware and software expenses. We estimate that the benefits from the system will outweigh the initial and ongoing costs.

*B. Technical Feasibility:*

We examined the system's functions, performance, and limitations to ensure it's achievable. This involved studying the functionality outlined in the System Requirement Specification (SRS) and testing different frontend and backend platforms.

*C. Operational Feasibility:*

The proposed system has a user-friendly interface that's easy for anyone to understand. Proper training has been provided to users to ensure they are comfortable with the new system. Based on our study, clients are happy with the system as it has reduced their workload.

# **SYSTEM DESIGN**

# 

# During the design phase of software development, we take the client's requirements and turn them into a system that actually works. This phase involves two main steps:

1. *Primary Design Phase:*

Here, we start by designing the system at a broad level, kind of like creating the skeleton of a building. We identify different functions that the system needs to perform and create blocks for each of them. The goal is to keep things organized and make sure that different parts of the system don't have to interact too much. So, activities that need to work closely together are grouped into the same block.

1. *Secondary Design Phase:*

Once we have the basic structure in place, we dive deeper into each block. This is where we work out all the tiny details and figure out exactly how each part of the system will function.

Basically, it's like sketching out the blueprint for a house and then going in to plan out every room and corner.

*The general tasks involved in the design process are the following:*

* Design various blocks for overall system processes.
* Design smaller, compact and workable modules in each block.
* Design various database structures.
* Specify details of programs to achieve desired functionality.
* Design the form of inputs, and outputs of the system.
* Perform documentation of the design.
* System reviews.

***User Interface Design***

User Interface Design is concerned with the dialogue between a user an the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

The following steps are various guidelines for User Interface Design:

1. The system user should always be aware of what to do next.
2. The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.
3. Messages, instructions or information should be displayed long enough to allow the system user to read them.
4. Use display attributes sparingly.
5. Default values for fields and answers to be entered by the user should be specified.
6. A user should not be allowed to proceed without correcting an error.
7. The system user should never get an operating system message or fatal error.

***Preliminary Product Description***

The first step in the system development life cycle is the preliminary investigation to determine the feasibility of the system. The purpose of the preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the business system in all respects. Rather, it is the collecting of information that helps committee members to evaluate the merits of the project request and make an informed judgment about the feasibility of the proposed project.

Analysts working on the preliminary investigation should accomplish the following objectives:

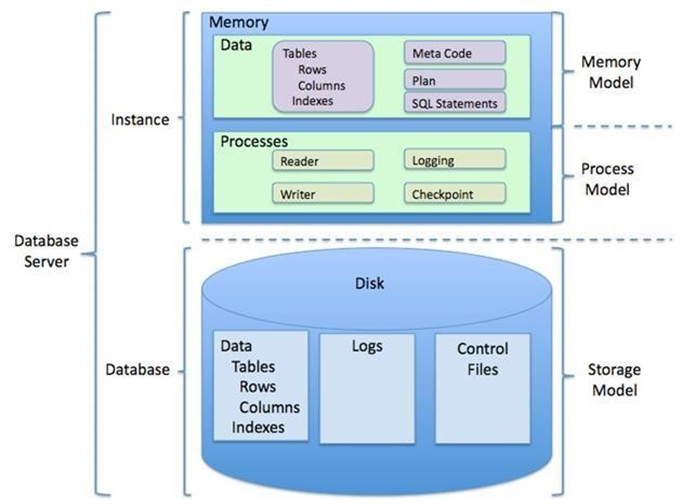
* Benefit to Organization: The organization will obviously be able to gain benefits such as savings in operating cost, reduction in paperwork, better utilization of human resources and more presentable image increasing goodwill.
* The Initial Cost: The initial cost of setting up the system will include the cost of hardware software (OS, add-on software, utilities) & labor (setup & maintenance). The same has to bear by the organization.
* Running Cost: Besides, the initial cost and the long-term cost will include the running cost for the system including the AMC, stationary charges, cost for human resources, cost for update/renewal of various related software.
* Need for Training: The users along with the administrator need to be trained at the time of implementation of the system for smooth running of the system. The client will provide the training site.

***Project Category (RDBMS)***

Relational Database Management System (RDBMS): This is an RDBMS based project which is currently using MySQL for all the transaction statements. MySQL is an open-source RDBMS System.

A relational database management system (RDBMS) is a database management system (DBMS) that is based on the relational model as invented by E. F. Codd, of IBM's San Jose Research Laboratory. Many popular databases currently in use are based on the relational database model.

RDBMSs have become a predominant choice for the storage of information in new databases used for financial records, manufacturing and logistical information, personnel data, and much more since the 1980s. Relational databases have often replaced legacy hierarchical databases and network databases because they are easier to understand and use. However, relational databases have been challenged by object databases, which were introduced in an attempt to address the object-relational impedance mismatch in relational databases, and XML databases.



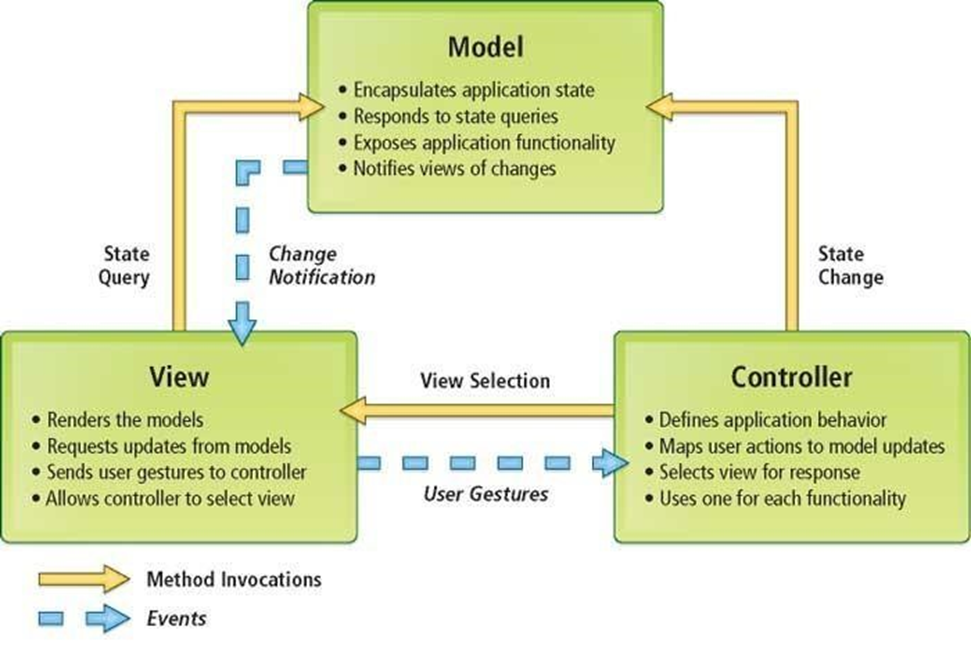
**Fig. 1: RDBMS**

***Implementation Methodology***

Model View Controller or MVC as it is popularly called, is a software design pattern for developing web applications. A Model View Controller pattern is made up of the following three parts:

* Model: The lowest level of the pattern which is responsible for maintaining data.
* View: This is responsible for displaying all or a portion of the data to the user.
* Controller: Software Code that controls the interactions between the Model and View.

MVC is popular as it isolates the application logic from the user interface layer and supports separation of concerns. Here the Controller receives all requests for the application and then works with the Model to prepare any data needed by the View. The View then uses the data prepared by the Controller to generate a final presentable response. The MVC abstraction can be graphically represented as follows.

****

**Fig. 2: MVC (Model View Controller Flow) Diagram**

For a successful software project, the following steps can be followed:

* Select a project
  + Identifying project’s aims and objectives
  + Understanding requirements and specification
  + Methods of analysis, design and implementation
  + Testing techniques
  + Documentation
* Project milestones and deliverables
* Budget allocation
  + Exceeding limits within control
  + Project Estimates
  + Cost
  + Time
  + Size of code
  + Duration
* Resource Allocation
  + Hardware
  + Software
  + Previous relevant project information
  + Digital Library
* Risk Management
  + Risk avoidance
  + Risk detection

***Cost estimation of the project***

Software cost comprises a small percentage of overall computer-based system cost. There are a number of factors, which are considered, that can affect the ultimate cost of the software such as - human, technical, Hardware and Software availability etc.

The main point that was considered during the cost estimation of project was its sizing. In spite of complete software sizing, function point and approximate lines of code were also used to size each element of the Software and their costing.

The cost estimation done by us for project also depend upon the baseline metrics collected from past projects and these were used in conjunction with estimation variables to develop cost and effort projections. We have basically estimated this project mainly on two bases:

1. Effort Estimation**:** This refers to the total man-hours required for the development of the project. It even includes the time required for doing documentation and user manual.
2. Hardware Required Estimation**:** This includes the cost of the PCs and the hardware cost required for development of this project.

***Project Scheduling***

An elementary Gantt chart or Timeline chart for the development plan is given below. The plan explains the tasks versus the time (in weeks) they will take to complete.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | January | | | | February | | | | March | | | |
| Requirement  Gathering |  | |  | |  | | | |  | | | |
| Analysis |  | |  | |  | | | |  | | | |
| Design |  | | | |  |  | | |  | | | |
| Coding |  | | | |  |  | | |  |  | | |
| Testing |  | | | |  | | | |  |  | |  |
| Implement |  | | | |  | | | |  | | |  |
|  | W1 | W2 | W3 | W4 | W1 | W2 | W3 | W4 | W1 | W2 | W3 | W4 |

W*i ‘*s are weeks of the months, for i =1, 2, 3, 4

**Table 1: Gantt Chart**

**PROJECT PLANNING**

|  |  |
| --- | --- |
| ***Component*** | ***Specification*** |
| Operating System | Windows 11 |
| Language | PHP |
| Database | MySQL |
| Browser | Chrome, Brave, Firefox, etc. |
| Web Server | XAMPP |
| Software Development Kit | Visual Studio Code |

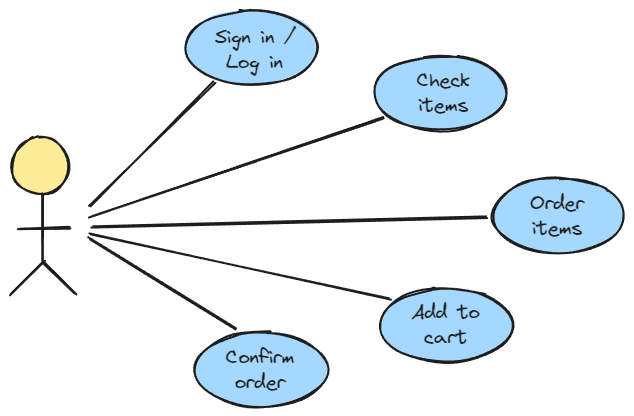
**Table 2: Software Requirements**

***Project Profile***

The software development landscape is constantly evolving, with developers facing challenges as technology trends change. Software re-engineering has become crucial in this environment. This involves understanding existing software systems and adapting them to new environments, often requiring extensive manual effort.

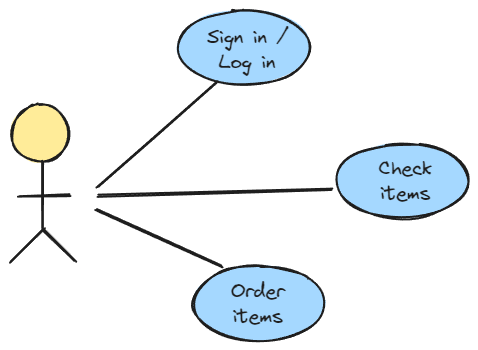
This project aims to tackle the challenges of program analysis and diagram generation to depict program structures more effectively. The use of UML (Unified Modeling Language) is increasingly becoming an industry standard in the software engineering design process. It essentially provides several diagramming tools that can express different aspects or characteristics of a program:

1. Use Case Diagrams: These helps elicit requirements from users in meaningful chunks. Construction planning revolves around delivering these use cases for system testing on an interaction basis.



**Fig3: Use Case Diagram**

1. Class Diagrams: These illustrate the static structure of concepts, types, and classes. Concepts represent how users think about the world; types show interfaces of software components, and classes depict the implementation of software components.

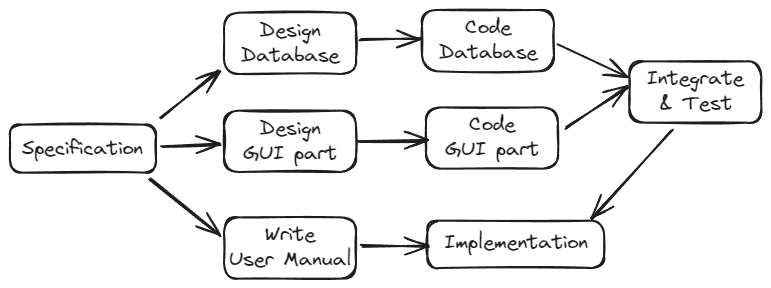


**Fig4: Class Diagram**

1. Interaction Diagrams: Interaction diagrams describe how groups of objects collaborate in specific behaviors, typically aligned with individual use cases. Sequence and collaboration diagrams are two common types, illustrating the flow of messages between objects.
2. Package Diagrams: Package diagrams serve to organize classes into higher-level units, addressing the challenge of breaking down large systems into manageable components. They depict dependencies among packages and classes, facilitating modular design and understanding.
3. State Diagrams: State diagrams describe the behavior of a system by delineating the possible states an object can assume and how it transitions between states in response to events. They are commonly used to model the lifecycle of individual objects, often based on David Harel’s state chart semantics.
4. Activity Diagrams: These show behavior with control structures and can depict many objects over many uses, many objects in a single use case, or methods that encourage parallel behavior.
5. Association: Associations represent relationships between instances of classes, capturing conceptual relations between classes. Each association comprises two roles, each with a multiplicity indicating how many objects may participate in the relationship.
6. Generalization: Generalization involves identifying commonalities among classes and organizing them into a hierarchy of inheritance. For instance, personal and corporate customers in a business exhibit both differences and similarities, allowing for the formation of subtypes under a generalized superclass.
7. Aggregation: Aggregation denotes part-whole relationships, such as a car having components like an engine and wheels. Distinguishing between aggregation and association can be nuanced.

***PERT Chart***

The Program Evaluation Review Technique (PERT) chart is a tool used to organize events, activities, or tasks in a project. It serves as a scheduling device that visually depicts the sequence of tasks to be completed.



**Fig. 5: PERT Chart**

One of its key functions is to facilitate the calculation of the critical path. The critical path represents the sequence of tasks that collectively require the longest duration to complete. By identifying the critical path, project managers can pinpoint where delays would have the most significant impact on the overall project timeline.

Additionally, PERT charts allow for the estimation of both time and cost associated with each path or sequence of tasks. This enables project managers to allocate resources effectively and make informed decisions regarding scheduling and budgeting.

***Use Case Model of the Project***

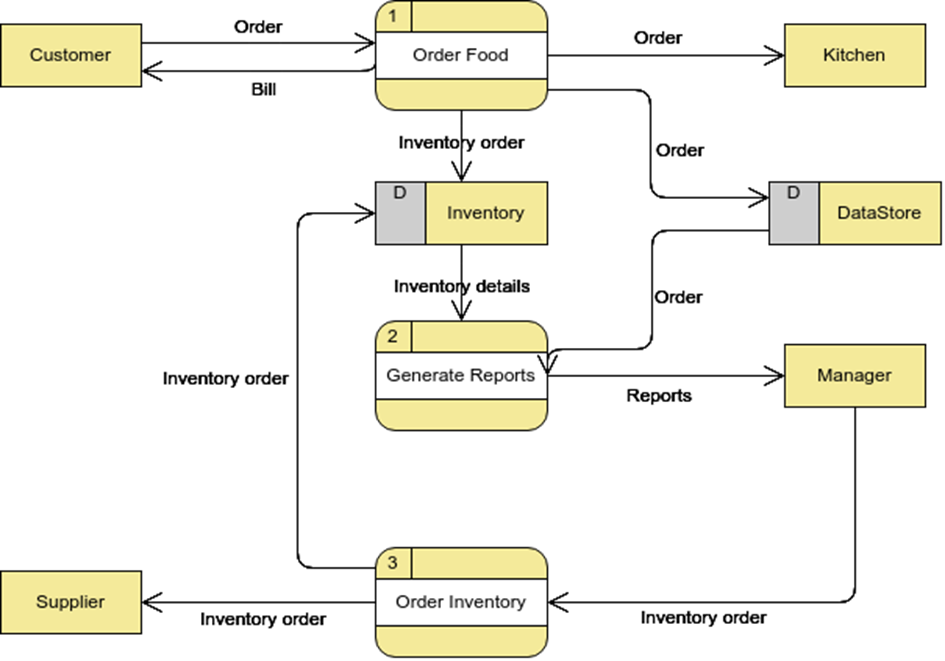
The use case model for any system consists of “use cases”. Use cases represent different ways in which the system can be used by the user. A simple way to find all the use case of a system is to ask the questions “What the user can do using the system?” The use cases partition the system behavior into transactions such that each transaction performs some useful action from the users’ point of view.

The purpose of the use case is to define a piece of coherent behavior without revealing the internal structure of the system. A use case typically represents a sequence of interaction between the user and the system. These interactions consist of one main line sequence that represents the normal interaction between the user and the system. The use case model is an important analysis and design artifact (task). Use cases can be represented by drawing a use case diagram and writing an accompanying text elaborating the drawing.

In the use case diagram, each use case is represented by an ellipse with the name of the use case written inside the ellipse. All the ellipses of the system are enclosed within a rectangle which represents the system boundary. The name of the system being moduled appears inside the rectangle. The different users of the system are represented by using a stick person icon. The stick person icon is normally referred to as an Actor. The line connecting the actor and the use cases is called the communication relationship. When a stick person icon represents an external system, it is annotated by the stereotype <<external system>>.

***Dataflow Diagram***

Data flow diagram is the starting point of the design phase that functionally decomposes the requirements specification.



**Fig6: Data flow diagram**

A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flows in the system. A DFD describes what data flow rather than how they are processed, so it does not hardware, software and data structure.

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing (structured design). A data flow diagram (DFD) is a significant modeling technique for analyzing and constructing information processes.

DFD literally means an illustration that explains the course or movement of information in a process. DFD illustrates this flow of information in a process based on the inputs and outputs. A DFD can be referred to as a Process Model.

The data flow diagram is a graphical description of a system’s data and how to Process transform the data is known as Data Flow Diagram (DFD). Unlike detailed flow charts, DFDs don’t supply detailed descriptions of modules that graphically describe a system’s data and how the data interact with the system. Data flow diagram number of symbols and the following symbols are by DeMarco.

|  |  |  |
| --- | --- | --- |
|  | Dataflow | Arrows showing directions of flow |
|  | Process | Circles |
|  | File | Horizontal pair of lines |
|  | Data source, Sink | Rectangular box |

**Table 3: Symbols used in Data flow diagram**

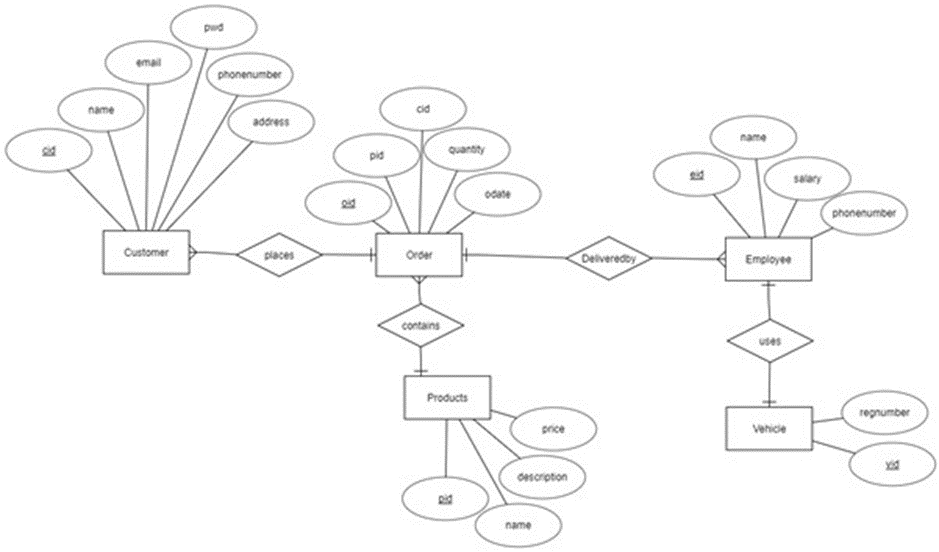
*There are seven rules for constructing a data flow diagram.*

1. Arrows should not cross each other.
2. Squares, circles and files must wear names.
3. Decomposed data flows must be balanced.
4. No two data flows, squares or circles can be the same names.
5. Draw all data flows around the outside of the diagram.
6. Choose meaningful names for data flows, processes & data stores.
7. Control information such as record units, password and validation requirements are not penitent to a data flow diagram.

On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process. It is common practice to draw a [context-level data flow diagram f](http://en.wikipedia.org/wiki/System_context_diagram)irst, which shows the interaction between the system and external agents, which act as data sources and data sinks. On the context diagram (also known as the Level 0 DFD’), the system's interactions with the outside world are modeled purely in terms of data flows across the system boundary. The context diagram shows the entire system as a single process, and gives no clues as to its internal organization.

***Entity Relationship Diagram***

E-R Model is a popular high level conceptual data model. This model and its variations are frequently used for the conceptual design of database application and many database design tools employ its concept.



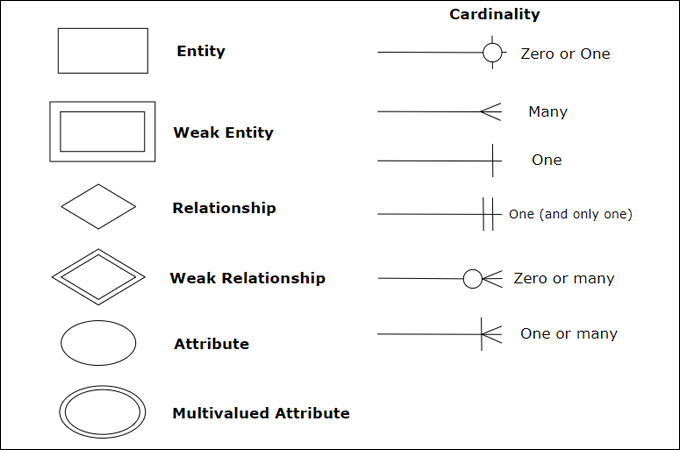
**Fig. 7: ER diagram**

A database that confirms an E-R diagram can be represented by a collection of tables in the relational system. The mapping of E-R diagram to the entities are:

* Attributes
* Relations
  + Many-to-many
  + Many-to-one
  + One-to-many
  + One-to-one
* Weak entities
* Subtype and supertype

The entities and their relationships between them are shown using the following conventions.

* An entity is shown in a rectangle.
* A diamond represents the relationship among a number of entities.
* The attributes shown as ovals are connected to the entities or relationship by lines.
* Diamond, oval and relationships are labeled.



**Fig. 8: Symbols in ER diagram**

**TESTING PROJECT**

Testing is vital for the success of any software. no system design is ever perfect. Testing is also carried in two phases. first phase is during the software engineering that is during the module creation. second phase is after the completion of software. this is system testing which verifies that the whole set of programs hanged together.

***Types of Testing***

* White Box Testing: In this technique, the close examination of the logical parts through the software are tested by cases that exercise species sets of conditions or loops. All logical parts of the software checked once. errors that can be corrected using this technique are typographical errors, logical expressions which should be executed once may be getting executed more than once and error resulting by using wrong controls and loops. When the box testing tests all the independent part within a module a logical decisions on their true and the false side are exercised , all loops and bounds within their operational bounds were exercised and internal data structure to ensure their validity were exercised once.
* Black Box Testing: This method enables the software engineer to device sets of input techniques that fully exercise all functional requirements for a program. black box testing tests the input, the output and the external data. it checks whether the input data is correct and whether we are getting the desired output.
* Alpha Testing: Acceptance testing is also sometimes called alpha testing. Be spoke systems are developed for a single customer. The alpha testing proceeds until the system developer and the customer agree that the provided system is an acceptable implementation of the system requirements.
* Beta Testing:On the other hand, when a system isto be marked as a software product, another process called beta testing is often conducted. During beta testing, a system is delivered among a number of potential users who agree to use it. The customers then report problems to the developers. This provides the product for real use and detects errors which may not have been anticipated by the system developers.
* Unit Testing: Each module is considered independently. it focuses on each unit of software as implemented in the source code. it is white box testing.
* Integration Testing: Integration testing aims at constructing the program structure while at the same constructing tests to uncover errors associated with interfacing the modules. modules are integrated by using the top down approach.
* Validation Testing: Validation testing was performed to ensure that all the functional and performance requirements are met.
* System Testing: It is executing programs to check logical changes made in it with intention of finding errors. a system is tested for online response, volume of transaction, recovery from failure etc. System testing is done to ensure that the system satisfies all the user requirements.

***Detailed Design of Implementation***

This phase of the systems development life cycle refines hardware and software specifications, establishes programming plans, trains users and implements extensive testing procedures, to evaluate design and operating specifications and/or provide the basis for further modification.

* Technical Design: This activity builds upon specifications produced during new system design, adding detailed technical specifications and documentation.
* Test Specifications and Planning: This activity prepares detailed test specifications for individual modules and programs, job streams, subsystems, and for the system as a whole.
* Programming and Testing: This activity encompasses actual development, writing, and testing of program units or modules.
* User Training: This activity encompasses writing user procedure manuals, preparation of user training materials, conducting training programs, and testing procedures.
* Acceptance Test: A final procedural review to demonstrate a system and secure user approval before a system becomes operational.
* Installation Phase: In this phase the new Computerized system is installed, the conversion to new procedures is fully implemented, and the potential of the new system is explored.
* System Installation: The process of starting the actual use of a system and training user personnel in its operation.
* Review Phase: This phase evaluates the successes and failures during a systems development project, and to measure the results of a new Computerized Tran system in terms of benefits and savings projected at the start of the project.
* Development Recap: A review of a project immediately after completion to find successes and potential problems in future work.
* Post-Implementation Review: A review, conducted after a new system has been in operation for some time, to evaluate actual system performance against original expectations and projections for cost-benefit improvements. Also identifies maintenance projects to enhance or improve the system.

***Steps In the Software Testing***

1. Preparation of the test cases.
2. Preparation of the possible test data with all the validation checks.
3. Complete code review of the module.
4. Actual testing done manually.
5. Modifications done for the errors found during testing.
6. Prepared the test result scripts.

The unit testing done included the testing of the following items:

1. Functionality of the entire module/forms.
2. Validations for user input.
3. Checking of the Coding standards to be maintained during coding.
4. Testing the module with all the possible test data.
5. Testing of the functionality involving all type of calculations etc.
6. Commenting standard in the source files.

After completing the Unit testing of all the modules, the whole system is integrated with all its dependencies in that module. While System Integration, We integrated the modules one by one and tested the system at each step. This helped in reduction of errors at the time of the system testing.

The steps involved during System testing are as follows:

1. Integration of all the modules/forms in the system.
2. Preparation of the test cases.
3. Preparation of the possible test data with all the validation checks.
4. Actual testing done manually.
5. Recording of all the reproduced errors.
6. Modifications done for the errors found during testing.
7. Prepared the test result scripts after rectification of the errors.

The System Testing done included the testing of the following items:

1. Functionality of the entire system as a whole.
2. User Interface of the system.
3. Testing the dependent modules together with all the possible test data scripts.
4. Verification and Validation testing.
5. Testing the reports with all its functionality.

After the completion of system testing, the next following phase was the Acceptance Testing. Clients at their end did this and accepted the system with appreciation. Thus, we reached the final phase of the project delivery. There are other six tests, which fall under special category. They are described below:

* Peak Load Test: It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand. For example, test the system by activating all terminals at the same time.
* Storage Testing: It determines the capacity of the system to store transaction data on a disk or in other files.
* Performance Time Testing: it determines the length of time system used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response.
* Recovery Testing: This testing determines the ability of user to recover data or re-start system after failure. For example, load backup copy of data and resume processing without data or integrity loss.
* Procedure Testing: It determines the clarity of documentation on operation and uses of system by having users do exactly what manuals request. For example, powering down system at the end of week or responding to paper-out light on printer.
* Human Factors Testing: It determines how users will use the system when processing data or preparing reports.

***System Analysis***

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information about the Online Food Ordering System to recommend improvements on the system. It is a problem-solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action. A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal. Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies.

In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

***Data Dictionary***

This is normally represented as the data about data. It is also termed as metadata some times which gives the data about the data stored in the database. It defines each data term encountered during the analysis and design of a new system. Data elements can describe files or the processes. Following are some major symbols used in the data dictionary

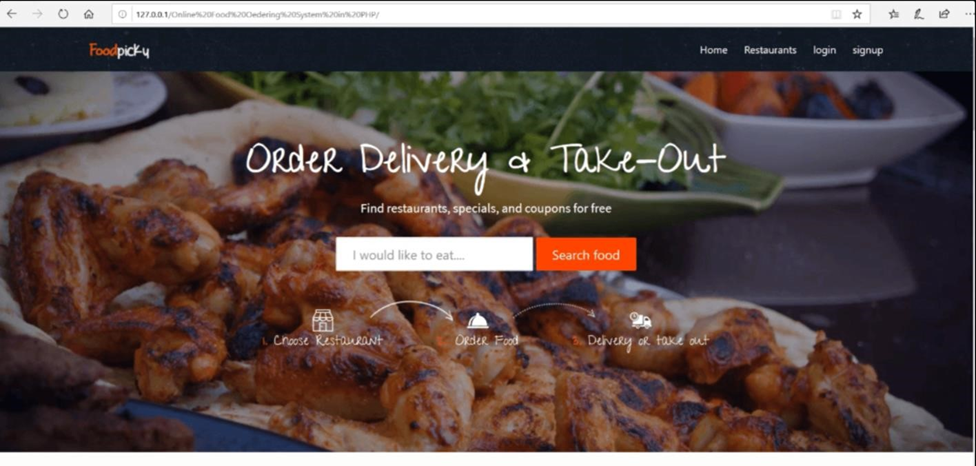
* = equivalent to
* + and
* [] either/ or
* () Optional entry

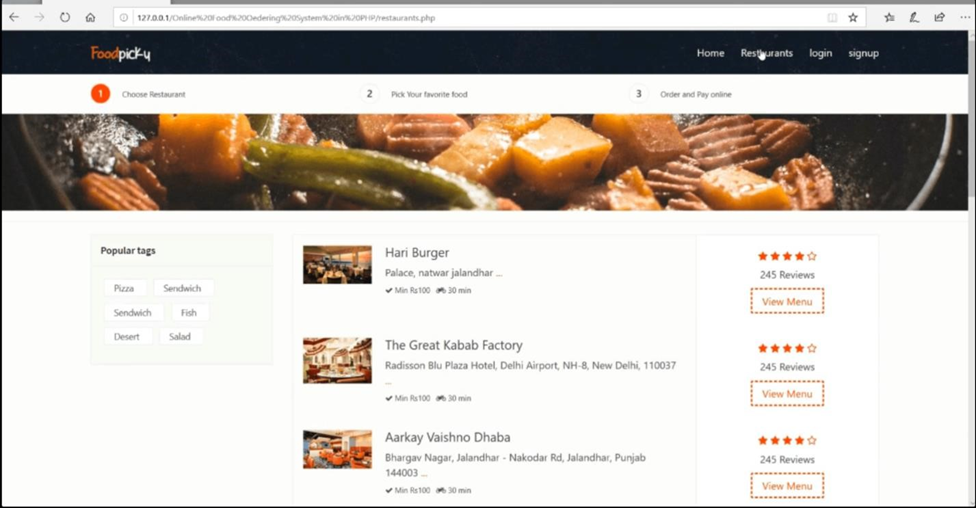
Following are some rules, which defines the construction of data dictionary entries:

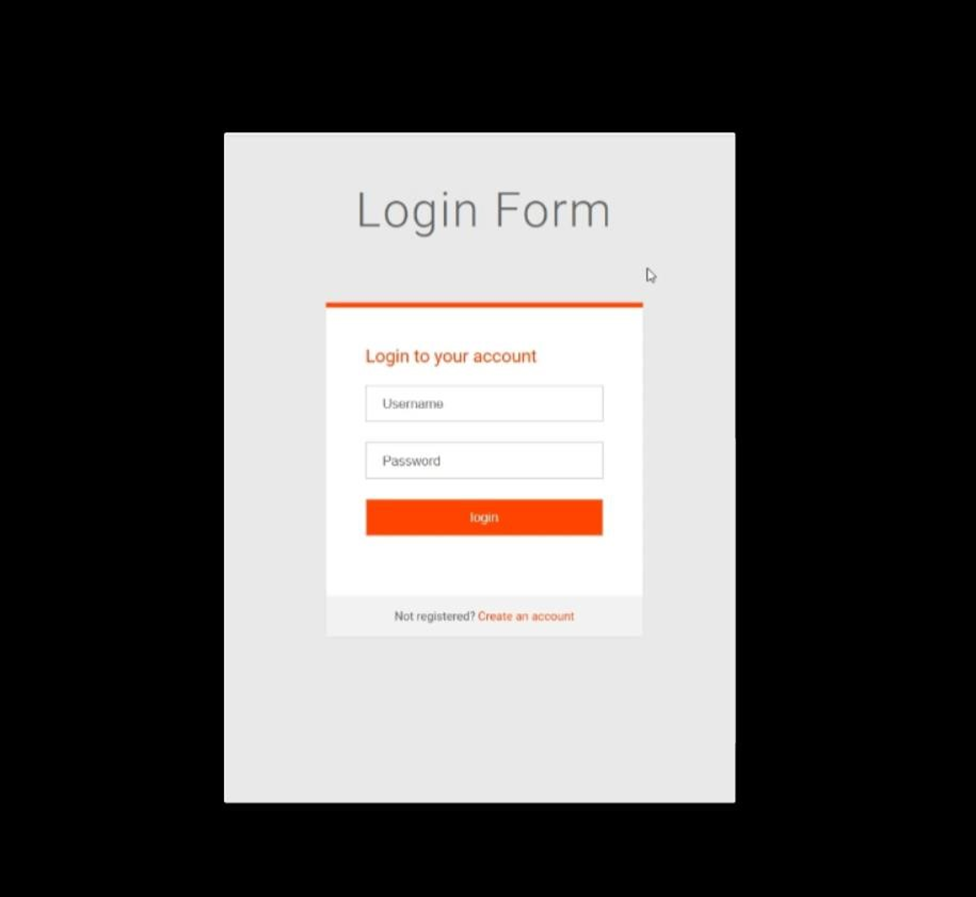
1. Words should be defined to understand for what they need and not the variable need by which they may be described in the program.
2. Each word must be unique. We cannot have two definitions of the same client.
3. Aliases or synonyms are allowed when two or more enters shows the same meaning. For example, a vendor number may also be called as customer number.
4. A self-defining word should not be decomposed. It means that the reduction of any information in to subpart should be done only if it is really required that is it is not easy to understand directly.

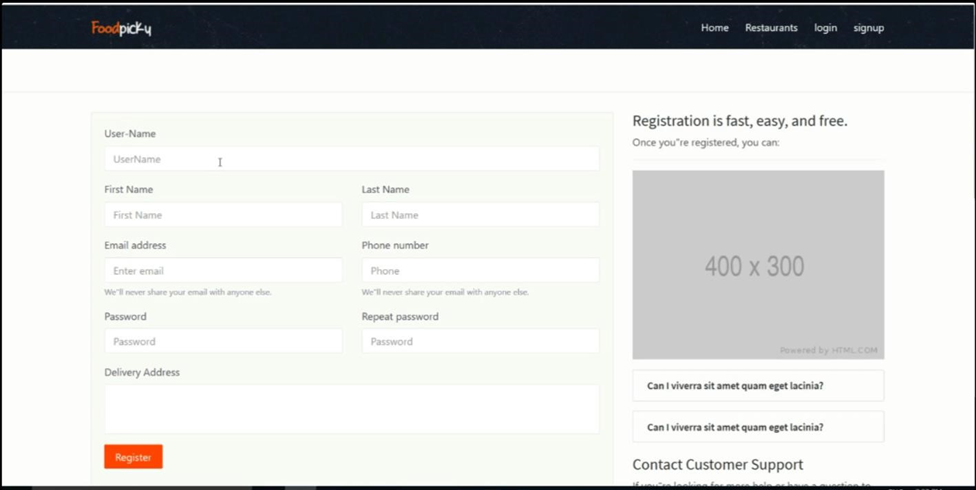
Data dictionary includes information such as the number of records in file, the frequency a process will run, security factor like pass word which user must enter to get excess to the information.

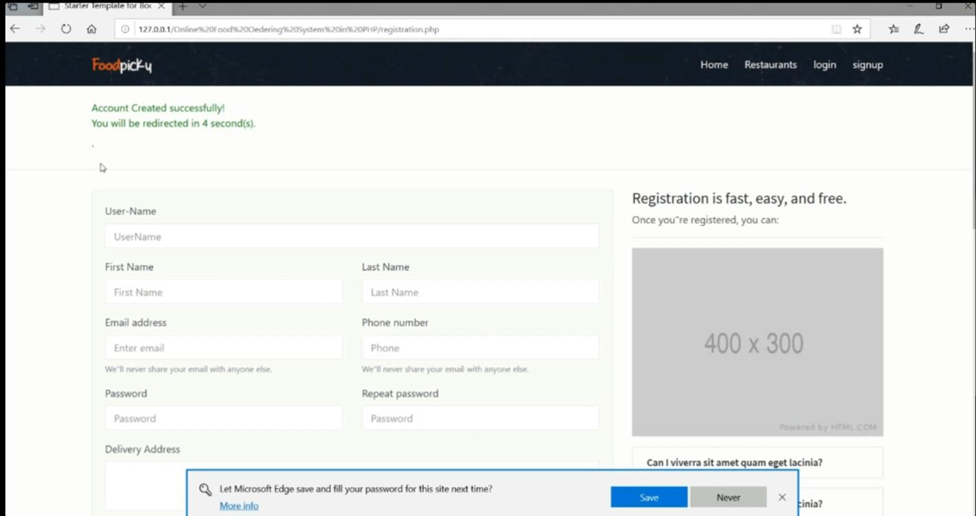
**SCREENSHOTS OF THE PROJECT**

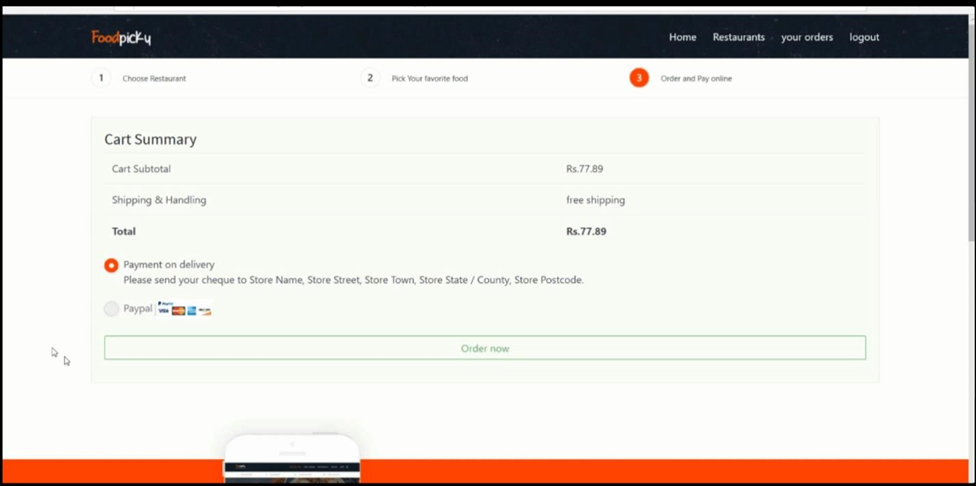


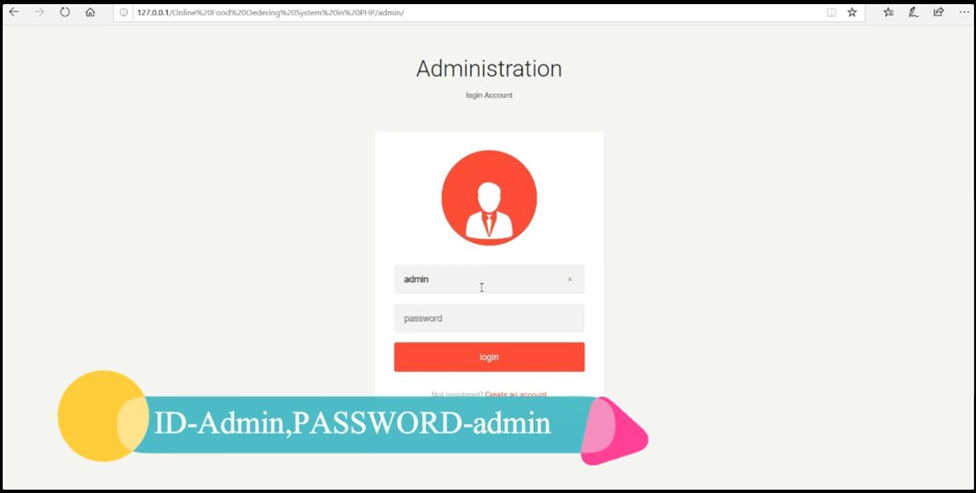


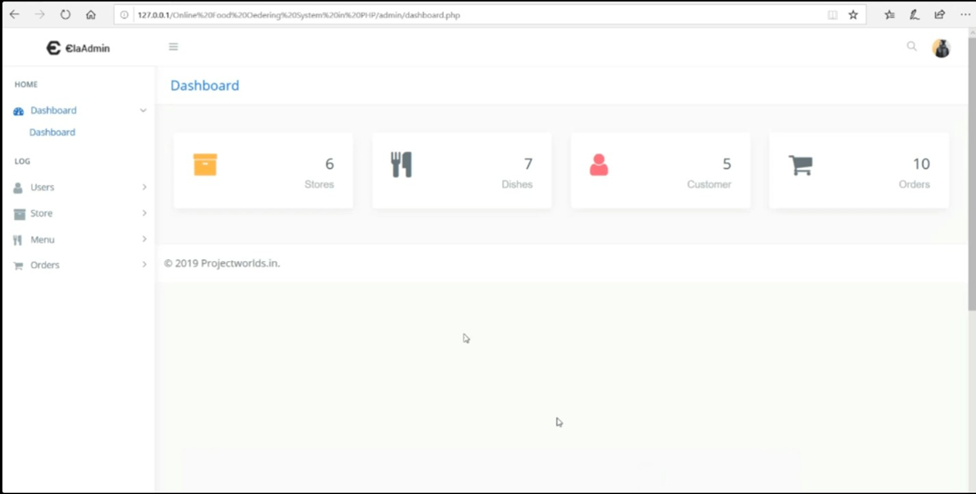


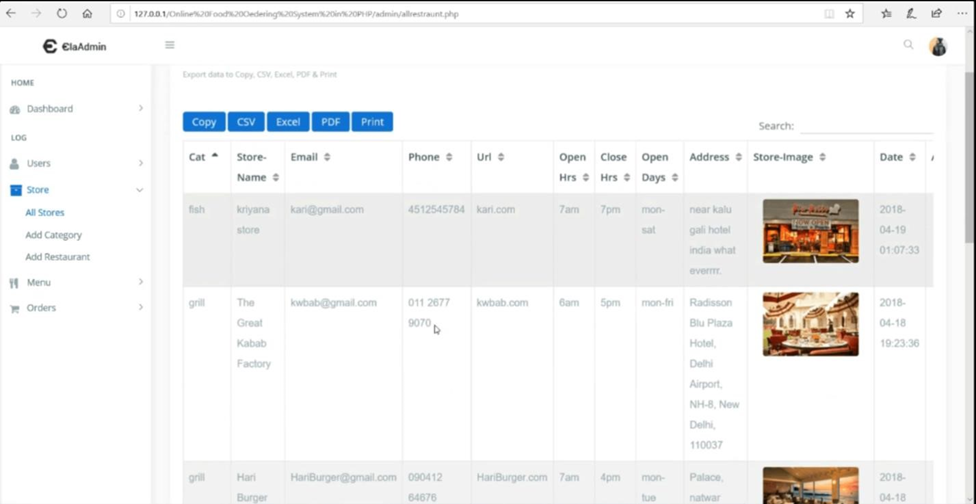


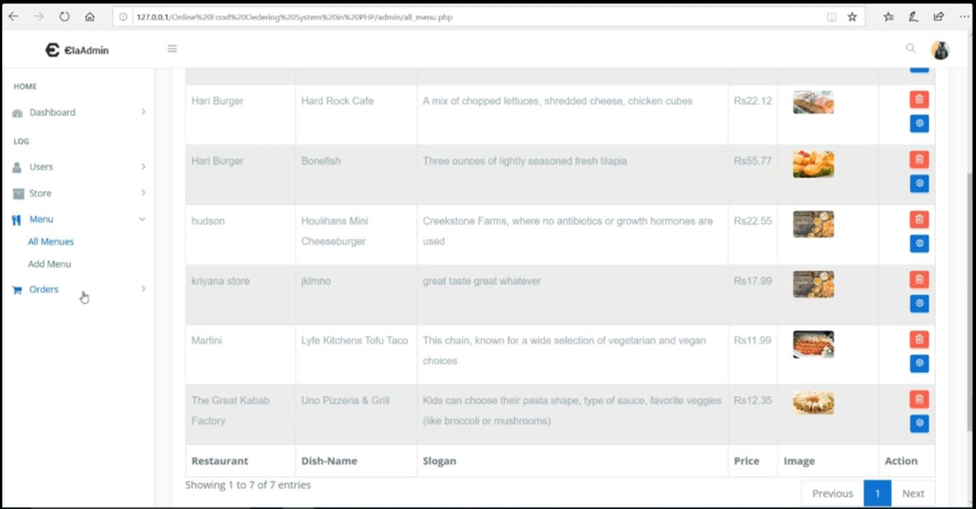


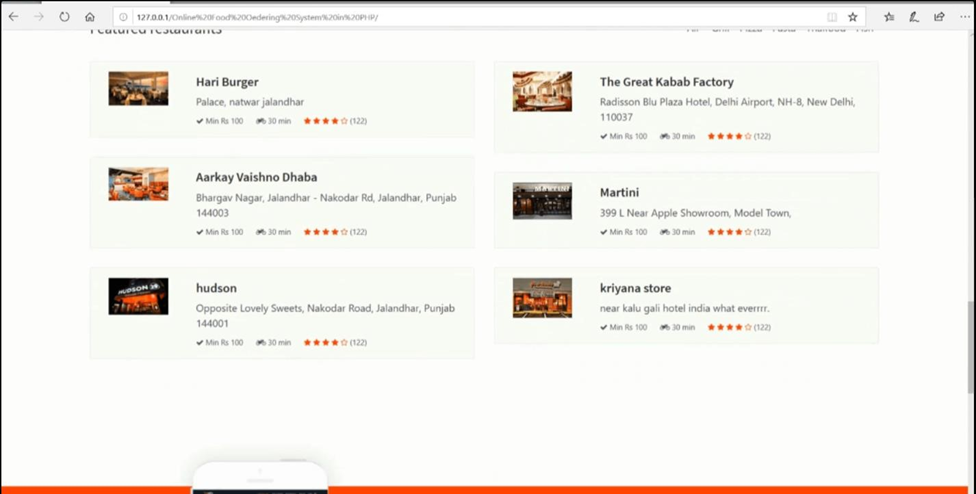












**CONCLUSION**

Our project is only a humble venture to satisfy the needs to manage their project work. Several user friendly coding have also adopted. This package shall prove to be a powerful package in satisfying all the requirements of the school. The objective of software planning is to provide a frame work that enables the manger to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

*At the end it is concluded that we have made effort on following points:*

* A description of the background and context of the project and its relation to work already done in the area.
* Made statement of the aims and objectives of the project.
* The description of Purpose, Scope, and applicability.
* We define the problem on which we are working in the project.
* We describe the requirement Specifications of the system and the actions that can be done on these things.
* We understand the problem domain and produce a model of the system, which describes operations that can be performed on the system.
* We included features and operations in detail, including screen layouts.
* We designed user interface and security issues related to system.
* Finally, the system is implemented and tested according to test cases.

*Future Scope*

In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:

* We can add printer in future.
* We can give more advance software for Online Food Ordering System including more facilities
* We will host the platform on online servers to make it accessible worldwide
* Integrate multiple load balancers to distribute the loads of the system
* Create the master and slave database structure to reduce the overload of the database queries
* Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers

The above-mentioned points are the enhancements which can be done to increase the applicability and usage of this project. Here we can maintain the records of Food Item and Category. Also, as it can be seen that now-a-days the players are versatile, i.e. so there is a scope for introducing a method to maintain the Online Food Ordering System. Enhancements can be done to maintain all the Food Item, Category, Customer, Order, Confirm Order.

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them. In the last we would like to thanks all the persons involved in the development of the system directly or indirectly. We hope that the project will serve its purpose for which it is develop there by underlining success of process.

*Limitations*

Although we have put my best efforts to make the software flexible, easy to operate but limitations cannot be ruled out even by me. Though the software presents a broad range of options to its users some intricate options could not be covered into it; partly because of logistic and partly due to lack of sophistication.

Paucity of time was also major constraint; thus, it was not possible to make the software foolproof and dynamic. Lack of time also compelled me to ignore some part such as storing old result of the candidate etc.

Considerable efforts have made the software easy to operate even for the people not related to the field of computers but it is acknowledged that a layman may find it a bit problematic at the first instance. The user is provided help at each step for his convenience in working with the software.

List of limitations which is available in the Online Food Ordering System:

* Excel export has not been developed for Food Item, Category due to some criticality.
* The transactions are executed in off-line mode, hence on-line data for Customer, Order capture and modification is not possible.
* Off-line reports of Food Item, Confirm Order, Customer cannot be generated due to batch mode execution.

## **REFERENCES AND BIBLIOGRAPHY**

* Google for problem solving
* http://www.Phpworld.com/phpworld/jw-01-1998/php-01-Credentialreview.html
* Database Programming with mysql
* Head First Xampp 5.6
* http://www.php-tutorial.com/
* php and Software Design Concepts by A press
* https://www.tutorialspoint.com/php/
* http://www.phpttpoint.com/php-tutorial
* https://docs.oracle.com/javase/tutorial/
* http://www.php.net/
* http://www.tutorialspoint.com/mysql/
* httpd.apache.org/docs/2.0/misc/tutorials.html