Contents

1 INTRODUCTION

1.1 Project Profile

Online ordering system will be a web based application whose main language of programming will be PHP. Its main aim is to simplify and improve the efficiency of the ordering process for both customer and restaurant, minimize manual data entry and ensure data accuracy and security during order placement process. Customers will also be able to view product menus and there ingredients and be able to have a visual confirmation that the order was place correctly.
What I propose is an online ordering system originally designed for use in college cafeterias, but just as applicable in any food delivery industry. The main advantage of this system is that it greatly simplifies the ordering process for both the customer and the restaurant. The system also greatly lightens the load on the restaurants end, as the entire process of taking orders is automated. Once an order is placed on the webpage that will be designed, it is placed into the database and then retrieved, in pretty much real-time, by a web application on the restaurants end. Within this application, all items in the order are displayed, along with their corresponding options and delivery details, in a concise and easy to read manner. This allows the restaurant employees to quickly go through the orders as they are placed and produce the necessary items with minimal delay and confusion. The greatest advantage of this system is its flexiability.

1.2 Organization Overview

Entrepreneurs such as you are increasingly discovering the need for an excellent website that gather a wide audience. Our job is to create this platform for you- a simple yet creative website to bring together your target demographic. Our services include creative and efficient quality web designing, web development, SEO services, Website maintenance, mobile applications and so forth. Essentially, we are the solution for all your web portal needs. We pride in our analyzing and researching skills that helps us provide you with an easy-to-use, professional website that appeals to both your current and prospective clientele. We are committed to providing the best services in web technology with a vision of satisfied and contend clients. Let us take your business to the web so that millions of people across the world can access it. Allow us to help reach your goal faster. Cazablaze Technologies PVT Lmtd is a major international force in IT consulting and services. Utilizing our broad range of Web-based solutions, we address and resolve the integration & solutions needs of today's IT users for both hardware and software. Our integrated quality information system helps us manage all aspects of high quality software production in our organization. Internal quality assessments are done twice a year with the intention of identifying quality issues and areas of improvement. The assessment focuses on customer rating on product quality, our service levels as well as the effectiveness of the quality system. Cazablaze expert creative services team delivers the following services:

- Branding
- Web Application
- Consultancy
- SEO
- Software Development
- Web Designing
- SMO
- Digital Marketing
- Mobile App Development
- Custom ERP Solutions
- Consulting Services
- Host Services
- Testing

1.3 Overview of The System

Online ordering system will be a web based application whose main language of programming will be PHP. Its main aim is to simplify and improve the efficiency of the ordering process for both customer and restaurant, minimize manual data entry and ensure data accuracy and security during order placement process. Customers will also be able to view product menus and there ingredients and be able to have a visual confirmation that the order was place correctly.

1. Mobile application (android and IOS application)
2. Web portal

1.4 Scope of The Project

Mobile is the future of Software Development Googles Eric Schmidt. Usage of mobile phones has increased in the past year. India stands second in the world, in the number of active mobile phones. Today, out of the 6 billion mobile phones in the world, close to 1 billion is being used in India. This comes to about 70 percentage of our current population. Every month sees an increase of around 6 million subscribers. That, in fact, is a lot of numbers. With the increase in the number and make of mobile phones, there comes a demand for better applications. And in turn, huge scope of android mobile application development in India. Now, this puts a light on why companies like Nokia, BlackBerry, Samsung, HTC, Motorola, Google and many others are going wild with their innovations increase in the need and use of Mobile Applications.Android is an open-source Linux-based operating system designed mainly for smart phones and tablets. It is maintained as an open source project by Google. This open source code and licensing allows the developers and device manufacturers to modify the software according to their needs. Android platform has brought about cutting-edge technologies in app development.Owing to the popularity of Android, Mobile Apps development industries are considering Android Application Development as one of the best remunerative business opportunities. The need to hire knowledgeable mobile application developer is intense.India is considered as a country with several globally recognized IT hubs.One of the main reasons for this is that software as a service is highly cost effective. Before the acceptance of Android, the mobile app development industry was dominated by Proprietary

OS like Symbian and iOS. With Android, came the option for dynamic app development at a lower cost. When thinking of the scope of Android Application Development in India, we can take these three primary notions into consideration:

- Revenue : The need for inventive App Developers are increasing in the current job market. Mobile application development can also be taken up as a part time job, where you can create your own applications and submit it to the Google Play store which can be downloaded. Google adsense ads can be displayed in your application which again provides monetary gains.
- Ease of use : Learning Android Programming is fairly easy and app development is cost effective. Any software developer who can think out of the box will be able to put Android into extraordinary use.
- Support :The most important attraction of Android is backing by Google.

2 ABOUT THE DEVELOPING TOOLS

2.1 Introduction to PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processor, PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or a Common Gateway Interface (CGI) executable. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated webpage; for example, PHP code can generate a web page's HTML code, an image, or some other data. PHP has also evolved to include a command-line interface (CLI) capability and can be used in standalone graphical applications.

The canonical PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge. Despite its popularity, no written specification or standard existed for the PHP language until2014, leaving the canonical PHP interpreter as adefacto standard. Since 2014, there is ongoing work on creating a formal PHP specification.

There are two primary ways for adding support for PHP to a web server as a native web server module, or as a CGI executable. PHP has a direct module interface called Server Application Programming Interface (SAPI), which is supported by many web servers including Apache HTTP Server, Microsoft IIS, Netscape (now defunct) and iPackage set. Some other web servers, such as Omni HTTPd, support the Internet Server Application Programming Interface (ISAPI), which is a Microsoft's web server module interface.
Usage:-

PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runson a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere .It can also be used for command-line scripting and client-side graphical user interface (GUI)applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems (RDBMS). Most web hosting providers support PHP for use by their clients. It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.

2.2 My SQL

Structured Query Language (SQL), in computer science, a database sub language used in querying, updating and managing relational databases. Derived from an IBM research project that created Structured English Query Language (SEQUEL) in the 1970s SQL is accepted standard in database products. Although it is not a programming language in the same sense as C or PASCAL, SQL can either be used in formulating interactive queries or be embedded in as application as for handling data.Microsoft SQL server is a relational database that runs on the Windows NT operating system. SQL (RDBMS) is a widely accepted industry standard for defining , changing and managing data and controlling how changes to the database are made by using tables, index keys rows and columns to store data.

2.3 Android

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touch screen mobile devices such as smart phones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input.

Android's source code is released by Google under an open source license, although most Android devices ultimately ship with a combination of free and open source and proprietary software, including proprietary software required for accessing Google services. Android is popular with technology companies that require a ready-made, low-cost and customizable operating system for high-tech devices. Its open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which deliver updates to older devices, add new features for advanced users or bring Android to devices originally shipped with other operating systems.

2.4 Ionic Framework

Ionic is a complete open-source SDK for hybrid mobile app development. The original version was released in 2013 and built on top of AngularJS and Apache Cordova. The more recent releases, known as Ionic 3 or simply "Ionic", are built on Angular. Ionic provides tools and services for developing hybrid mobile apps using Web technologies like CSS, HTML5, and Sass. Apps can be built with these Web technologies and then distributed through native app stores to be installed on devices by leveraging Cordova. Ionic was created by Max Lynch, Ben Sperry, and Adam Bradley of Drifty Co. in 2013.Ionic provides all the functionality which can be found in native mobile development SDKs.[clarification needed][citation needed] Users can build their apps, customize them for Android or iOS, and deploy through Cordova. Ionic includes mobile components, typography, interactive paradigms, and an extensible base theme. Using Angular, Ionic provides custom components and methods for interacting with them. One such component, collection repeat, allows users to scroll through a list of thousands of items without any performance hits. Another component, scroll-view, creates a scrollable container with which users can interact using a native-influenced delegate system. Ionic Creator is a drag-and-drop interface building tool.

2.5 Netbeans

Netbeans is an integrated development environment (IDE) for developing primarily with Java, but also with other languages, in particular PHP, C/C++, and HTML5.It is also an application platform framework for Java desktop applications and others. The NetBeans IDE is written in Java and can run on Windows, OS X, Linux, Solaris and other platforms supporting a compatible JVM. The Net Beans Platform allows applications to be developed from a set of modular software components called modules. Applications based on the Net Beans Platform (including the Net Beans IDE itself) can be extended by third party developers. Framework for simplifying the development of JavaSwing desktop applications. The Net Beans IDE bundle for Java SE contains what is needed to start developing Net Beans plugging and Net Beans Platform based applications; no additional SDK is required. Applications can install modules dynamically. Any application can include the Update Centre module to allow users of the application to download digitally signed upgrades and new features directly into the running application. Reinstalling an upgrade or a new release does not force users to download the entire application again. The platform offers reusable services common to desktop applications, allowing developers to focus on the logic specific to their application.

2.6 Wordpress

WordPress is an open source Content Management System (CMS), which allows the users to build dynamic websites and blog. WordPress is the most popular blogging system on the web and allows updating, customizing and managing the website from its back-end CMS and components. This tutorial will teach you the basics of WordPress using which you can create websites with ease. The tutorial is divided into various sections for convenience. Each of these sections contain related topics with screenshots explaining the WordPress admin screens.

2.7 GIT

Git is a version control system for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source code management in software development,but it can be used to keep track of changes in any set of files. As a distributed revision control system it is aimed at speed, data integrity,and support for distributed, non-linear workflows.Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development.[12] Its current maintainer since 2005 is Junio Hamano.As with most other distributed version control systems, and unlike most clientserver systems, every Git directory on every computer is a full-fledged repository with complete history and full version tracking abilities, independent of network access or a central server.Git is free software distributed under the terms of the GNU General Public License version 2.

- Characteristics
  Git's design is a synthesis of Torvalds's experience with Linux in maintaining a large distributed development project, along with his intimate knowledge of file system performance gained from the same project and the urgent need to produce a working system in short order. These influences led to the following implementation choices;

  1. Strong support for non-linear development

2. Distributed development
3. Compatibility with extant systems and protocols
4. Efficient handling of large projects
5. Cryptographic authentication of history
6. Garbage accumulates until collected
7. Periodic explicit object packing

Another property of Git is that it snapshots directory trees of files. The earliest systems for tracking versions of source code, Source Code Control System (SCCS) and Revision Control System (RCS), worked on individual files and emphasized the space savings to be gained from interleaved deltas (SCCS) or delta encoding (RCS) the (mostly similar) versions. Later revision control systems maintained this notion of a file having an identity across multiple revisions of a project. However, Torvalds rejected this concept.Consequently, Git does not explicitly record file revision relationships at any level below the source code tree.Git implements several merging strategies; a non-default can be selected at merge time:

1. resolve: the traditional three-way merge algorithm.
2. recursive: This is the default when pulling or merging one branch, and is a variant of the three-way merge algorithm.
3. octopus: This is the default when merging more than two heads.

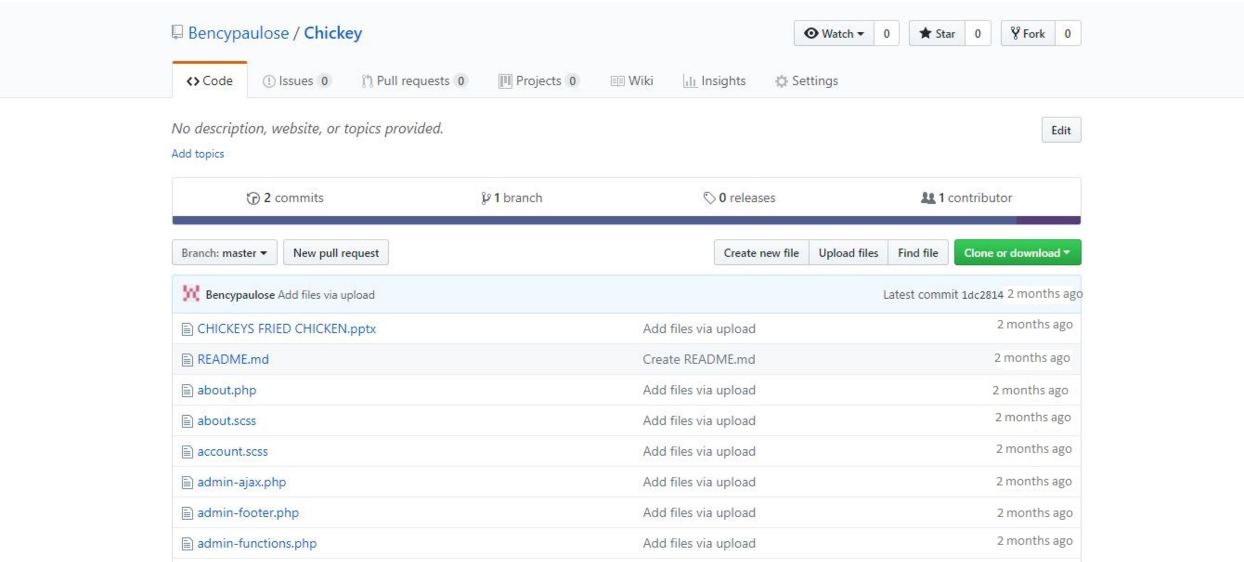
- Implementations
  Git is primarily developed on Linux, although it also supports most major operating systems including BSD, Solaris, macOS, and Windows.The first Microsoft Windows port of Git was primarily a Linux emulation framework that hosts the Linux version. Installing Git under Windows creates a similarly named Program Files directory containing the MinGW port of the GNU Compiler Collection, Perl 5, msys2.0 (itself a fork of Cygwin, a Unix-like emulation environment for Windows) and various other Windows ports or emulations of Linux utilities and libraries. Currently native Windows builds of Git are distributed as 32 and 64-bit installer.The JGit implementation of Git is a pure Java software library, designed to be embedded in any Java application. JGit is used in the Gerrit code review tool and in EGit, a Git client for the Eclipse IDE.The Dulwich implementation of Git is a pure Python software component for Python 2.7, 3.4 and 3.5.The libgit2 implementation

of Git is an ANSI C software library with no other dependencies, which can be built on multiple platforms including Windows, Linux, macOS, and BSD.It has bindings for many programming languages, including Ruby, Python, and Haskell.JS-Git is a JavaScript implementation of a subset of Git.

- GIT Server
  As Git is a distributed version control system, it can be used as a server out of the box. Dedicated Git server software helps, amongst other features, to add access control, display the contents of a Git repository via the web, and help managing multiple repositories. Remote file store and shell access: A Git repository can be cloned to a shared file system, and accessed by other persons. It can also be accessed via remote shell just by having the Git software installed and allowing a user to log in.

- Security
  Git does not provide access control mechanisms, but was designed for operation with other tools that specialize in access control.An attacker could perform arbitrary code execution on a target computer with Git installed by creating a malicious Git tree (directory) named .git (a directory in Git repositories that stores all the data of the repository) in a different case (such as .GIT or .Git, needed because Git doesn't allow the all-lowercase version of .git to be created manually) with malicious files in the .git/hooks subdirectory (a folder with executable files that Git runs) on a repository that the attacker made or on a repository that the attacker can modify. If a Windows or Mac user pulls (downloads) a version of the repository with the malicious directory, then switches to that directory, the .git directory will be overwritten (due to the case-insensitive trait of the Windows and Mac filesystems) and the malicious executable files in .git/hooks may be run, which results in the attacker's commands being executed. An attacker could also modify the .git/config configuration file, which allows the attacker to create malicious Git aliases (aliases for Git commands or external commands) or modify extant aliases to execute malicious commands when run. The vulnerability was patched in version 2.2.1 of Git, released on 17 December 2014, and announced on the next day.

Git Upload Details



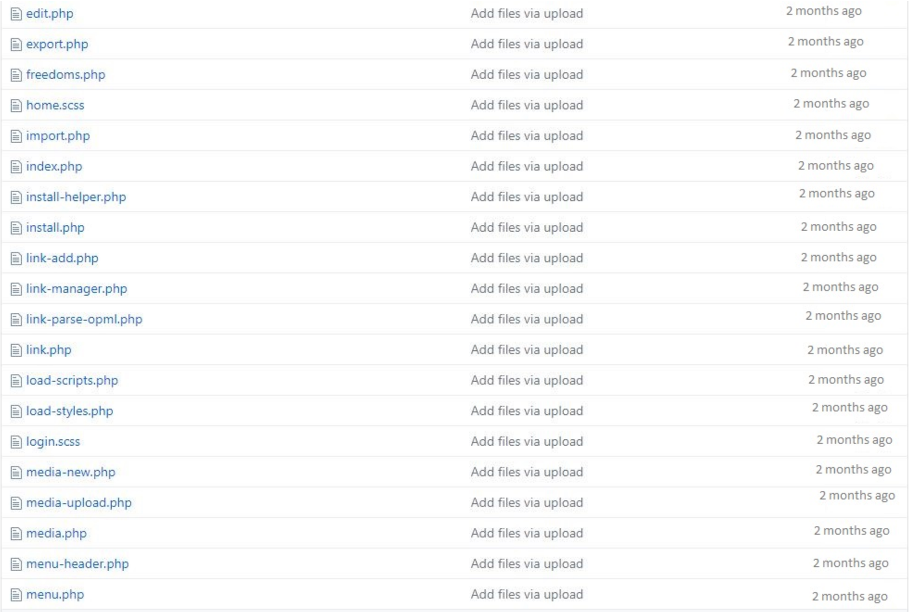
No description, website, or topics provided.

Add topics

2 commits 1 branch 0 releases 1 contributor

Bencypaulose Add files via upload Latest commit 1dc2814 2 months ago

CHICKEYS FRIED CHICKEN.pptx	Add files via upload	2 months ago
README.md	Create README.md	2 months ago
about.php	Add files via upload	2 months ago
about.scss	Add files via upload	2 months ago
account.scss	Add files via upload	2 months ago
admin-ajax.php	Add files via upload	2 months ago
admin-footer.php	Add files via upload	2 months ago
admin-functions.php	Add files via upload	2 months ago



edit.php	Add files via upload	2 months ago
export.php	Add files via upload	2 months ago
freedoms.php	Add files via upload	2 months ago
home.scss	Add files via upload	2 months ago
import.php	Add files via upload	2 months ago
index.php	Add files via upload	2 months ago
install-helper.php	Add files via upload	2 months ago
install.php	Add files via upload	2 months ago
link-add.php	Add files via upload	2 months ago
link-manager.php	Add files via upload	2 months ago
link-parse-opml.php	Add files via upload	2 months ago
link.php	Add files via upload	2 months ago
load-scripts.php	Add files via upload	2 months ago
load-styles.php	Add files via upload	2 months ago
login.scss	Add files via upload	2 months ago
media-new.php	Add files via upload	2 months ago
media-upload.php	Add files via upload	2 months ago
media.php	Add files via upload	2 months ago
menu-header.php	Add files via upload	2 months ago
menu.php	Add files via upload	2 months ago

HOW TO ENTER THE PROJECT VERSION TO GIT

1.Create a new repository on GitHub. To avoid errors, do not initialize the new repository with README, license, or gitignore files. You can add these files after your project has been pushed to GitHub.

2.Open Git Bash

3.Change the current working directory to your local project.

4.Initialize the local directory as a Git repository

```
$ git init
```

5.Add the files in your new local repository. This stages them for the first commit.

```
$ git add
```

6. Commit the files that you've staged in your local repository

```
$ git commit -m ”First commit”
```

7.At the top of your GitHub repository's Quick Setup page, click to copy the remote repository URL.

8.In the Command prompt, add the URL for the remote repository where your local repository will be pushed

```
$git remote add origin remote repository URL
$git remote -v
```

9.Push the changes in your local repository to GitHub

```
$git push origin master
```

3 SYSTEM ANALYSIS

3.1 Introduction

System Analysis works with users to identify goals and build system to achieve them. System Analysis is an important phase of any system development process. System analysis is a step-by-step process used to identify and develop or acquire the software need to control the processing of specific application. System analysis is a continuing activity the stages of the systems development. The system is studied to the minutes details and analyzed. In analysis, a detailed study of these operation performed by a system and their relationships within and outside of the system is done.System analysis is a general term that refers to an orderly, structured process for identifying and solving problems. We call system analysis process life cycle methodology, since it relates to four significant phases .They are:

1. Study phase
2. Design phase
3. Development phase
4. Implementation phase

Analysis implies the process of breaking something into parts so that the whole maybe understood. The definition of system analysis includes not only the process of analysis but also that of synthesis, which implies the process of putting together to form a new one. All activities associated with each life cycle phase must be performance, management,documentation of the activities related to the life cycle phases of a computer based business system. In the study phase a detailed study of the project is made and clear picture of the project should be in mind by this time. In the design phasethe designing of the input, output and table designs are made. In the development phase is where the physical designing of the input- output screens and coding of the system is done. In the system implementation actually implements the system by making necessary testing.

3.2 Existing System

The existing systems has lot of drawbacks.Thus making it time consuming,manual data entry,system is not flexible and does not ensure data accuracy.

3.2.1 Limitations of Existing System

1. Time Consuming.
2. Lack of efficiency.
3. Error Prone.
4. Lack of security.
5. Not informative.

3.3 Feasibilty Study

The prime objective of feasibility study is to ensure that the problem is worth to be solved. At this stage a cost benefit analysis is performed to assertion that the benefit from the system will over rule the cost associated with the whole analysis, design and development of the new system. An important outcome of the preliminary investigation determining whether the system required is feasible.

Feasibility study is a test of proposed system regarding its efficiency, impact on the organization, ability to meet the needs of users and effective use of resources. Thus, when a new project is proposed, it normally goes through a feasibility study before it is approved for development.

All the projects are given unlimited resources and infinite time. Unfortunately, the development process of a computer based system is time bound and feasibility and risk analysis are related in many ways. If project risk is great, the feasibility of producing the quality software is reduced.

There are three aspects in the feasibility study portion of the preliminary investigation.

1. Technical Feasibility
2. Economic Feasibility
3. Behavioural Feasibility

3.3.1 Technical Feasibility

It is a study of function, performance and constraints that may affect the ability to achieve acceptable system. The technical requirements for the CHICKEY are currently available and are widely used. The technology used is found to be optimal for the system and are able to withstand the complexities, which are inherent to the system.

The main points that are considered to prove that the project is technically feasible are:

1. The present technology is sufficient to develop the project.
2. The proposed system provides adequate response to the user.
3. The system can be expanded and developed.
4. The project outputs given are reliable and it is easy to access.

3.3.2 Economic Feasibility

Here an evaluation of development cost weighted against the ultimate income or benefit derived from the developed system. The cost for the development of the project has been evaluated and we want to check that the cost does not exceed beneficial cost of the system. Economic and Financial analysis is used for evaluating the effectiveness of the candidate system.

This project also under gone economic feasibility study and found that it is feasible. Because php is a free open source software and generally runs on a web server. So the cost for development does not exceed its beneficial cost.

3.3.3 Behavioural Feasibility

This is also known as operational feasibility. Operational feasibility is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed and implemented. One of the main problems faced during the development of the new system is getting the acceptance from the user. People are inherently resistant to change and so estimate should be made of how strong a reaction the user is likely to have towards the developing system. The system is much user friendly and the maintenance and working needs much less human effort. Define the urgency of the problem and the acceptability of any solution; if the system is developed, will it be used? It includes people oriented and social issues: internal issues, such as manpower problems, labourobjections, manager resistance, organisational conflicts and policies; also external issues, including social acceptability, legalaspects and government regulations. In case of this system, the organisation is completely in favour of creating the raid based shopping Management system as it saved their precious time, energy and moreover the system when implemented would help to remove inconsistencies, reduce manpower etc. Also there is no specialized training needed, only a few hours of instructed demo needs to be given to the user. So it might hence the system is behavioural feasible. This analysis involves how it will work when it is installed and the assessment of political and managerial environment in which it is implemented. People are inherently resistant to change and computers have been known to facilitate change. The new proposed system is very much useful to the users and therefore it will accept broad audience from around the world.

3.4 Proposed System

Online ordering system will be a web based application whose main language of programming will be PHP. Its main aim is to simplify and improve the efficiency of the ordering process for both customer and restaurant, minimize manual data entry and ensure data accuracy and security during order placement process. Customers will also be able to view product menus and there ingredients and be able to have a visual confirmation that the order was place correctly.

What I propose is an online ordering system originally designed for use in college cafeterias, but just as applicable in any food delivery industry. The main advantage of this system is that it greatly simplifies the ordering process for both the customer and the restaurant. The system also greatly lightens the load on the restaurants end, as the entire process of taking orders is automated. Once an order is placed on the webpage that will be designed, it is placed into the database and then retrieved, in pretty much real-time, by a web application on the restaurants end. Within this application, all items in the order are displayed, along with their corresponding options and delivery details, in a concise and easy to read manner. This allows the restaurant employees to quickly go through the orders as they are placed and produce the necessary items with minimal delay and confusion. The greatest advantage of this system is its flexiability.

3.4.1 Advantages

The system is very simple in design and to implement. It has got following features.

1. High speed
2. Transparency
3. Time saving
4. System dependent
5. Error free
6. Provide service to anyone at any time.
7. Efficient
8. Informative
9. Secure

4 FACT FINDING TECHNIQUES

The success of any project depends upon the accuracy of available data. Accurate information can be collected with the help of certain methods / techniques. These specific methods for finding information of the system are termed as fact finding techniques. Interview, Questionnaire, Record View and Observations are the different fact finding techniques used in this project.

4.1 Interview

This method is used to collect the information from groups or individuals. We select the people who are related with the system for the interview. In this method, we sit face to face with people and record their responses.

4.2 Record View

The information related to the system is available in the source like companys documents, websites and other records. This record review helped me to get valuable information about the system.

4.3 Onsite observation

Unlike the other fact finding techniques, in this method we visit the organization and observe and understand the working of the existing system, flow of the system, the users of the system etc.

5 SYSTEM SPECIFICATION

5.1 Hardware Specification

1. Processors : Intel Premium Pro or Processor running at 133 MHz
2. Hard Disk : 1.2 GB Hard Disk
3. RAM : Client Level Minimum 128MB

   Recommended Requirements for peak performance
4. RAM : Client Level Minimum 512MB
5. Display Type : SVGA Colour Enhanced Monitor
6. Mouse : PS/2 2 Button

5.2 Software Specification

1. Front End Tool : HTML,JavaScript,CSS, Twitter Bootstrap, Android, Ionic
2. Back End Tool : PHP, MySQL, WORDPRESS
3. Development tools : NetBeans
4. Operating System : Windows 8.1
5. Browse : Google Chrome. Mozilla Firefox

6 SYSTEM DESIGN

6.1 Introduction to System Design

The most creative and challenging phase of the system life cycle is system design. The term design describes a final system and the process by which it is developed. It refers to the technical specifications that will be applied in implementing a candidate system. It also includes the construction of programs and program testing. The question here is: How should the problem is solved?

The first step is to determine how the output is to be produced and in what format. Samples of the output (and input) are also presented. Second input data and master files have to be designed to meet the requirement of the proposed output. The operational phase are handled through program construction and testing, including a list of the programs needed to meet the systems objectives and complete documentation. Finally, details related to justification of the system and an estimate of the impact of the candidate system on the user and the organization are documented and evaluated by management as a step toward implementation. The goal of design process is to produce a model as representation of a system, which can be used later to build that system. The produced model is called the design of the system. The design process for software systems often has two levels. At the first level, the focus is on deciding which modules are needed for the system, the specification of these modules, and how the modules should be interconnected. This is what is called the system design or top level design. In the second level, the internal design of the module can be satisfied, is decided. This design level is often called detailed design of logic design.

6.2 Input Design

Input design phase consists of conversion of user oriented description of the inputs to a computer based business system in to a program oriented specification. An effective input design minimizes error by data entry operators. Taking in to account several input design considerations several interfaces for data entry operations have been created.

6.3 Output Design

Computer output is the most important and direct source of information to the user. Output design is a very important phase because the output will be in an attractive manner. The output should be in such a way that the user can see it from the screen and can take a hard copy from the printer. To make a user friendly output and for better communication the programmer can use the features of window. Efficient, intelligible output design should improve the systems relationship with the user and help in the decision making.

A major form of the output is a hard copy from the printer. The print outs should be designed around the output requirements of the user.

6.4 DataBase Design

The next consideration of the designer after designing the input and output is file design or how data should be organized around user requirements. How data are organized depends on the data and response requirements that determine hardware configurations. An integrated approach to file design is the database. The general theme behind a database is to handle information as an integrated whole. Database is a collection of inter-related data store together data with controlled redundancy to serve one or more application. In a database environment common data are available to the users. A program now requests the data through database management system (DBMS), which determines the data sharing. General objectives are to make information access easy, quick, efficient, inexperience and flexible for the user. Several specific objectives are ease of learning, data independence, integrity and recovery from failure, privacy and security, performance.

In a database environment, Database Management System (DBMS) is the software that provides the interface between the data file on disk and the program that requires processing. Although all DBMSs have a common approach to data management, they differ in the way they structure data. The three types of data structure are hierarchical, network and relational. Here we use relational structuring in which all data and relationships are represented in a flat, two-dimensional table called a relation. A relation equivalent to a file, where each line represents a record. Data structuring is refined through a process called normalization. Data are grouped in the simplest way possible so that later changes can be made with a minimum of impact on the data structure.

6.5 Normalization

Normalization is the process of efficiently organizing data in a database. There are two goals of the normalization process: eliminating redundant data (for example, storing the same data in more than one table) and ensuring data dependencies make sense (only storing related data in a table). Both of these are worthy goals as they reduce the amount of space a database consumes and ensure that data is logically stored.

The database community has developed a series of guidelines for ensuring that databases are normalized. These are referred to as normal forms and are numbered from one (the lowest form of normalization, referred to as first normal form or 1NF) through five (fifth normal form or 5NF). In practical applications, you'll often see 1NF, 2NF, and 3NF along with the occasional 4NF. Fifth normal form is very rarely seen.

6.5.1 First Normal Form(1NF)

First normal form (1NF) sets the very basic rules for an organized database.Create separate tables for each group of related data and identify each row with the primary key.

6.5.2 Second Normal Form(2NF)

Second normal form (2NF) further addresses the concept of removing duplicative data.

6.5.3 Third Normal Form(3NF)

Third normal form (3NF) goes one large step further .Meet all the requirements of the second normal form and remove columns that are not dependent upon the primary key.

6.5.4 Boyce-Codd Normal Form (BCNF or 3.5NF)

The Boyce-Codd Normal Form, also referred to as the "third and half (3.5) normal form.

6.5.5 Fourth Normal Form(4NF)

Finally, fourth normal form (4NF) has one additional requirement.A relation is in 4NF if it has no multi-valued dependencies.

6.6 Architectural Design

Architectural design is of crucial importance in software engineering during which the essential requirements like reliability, cost, and performance are dealt with. Architectural design is the responsibility of developers, some other people like user representatives, systems engineers, hardware engineers, and operations personnel are also involved. All these stakeholders must also be consulted while reviewing the architectural design in order to minimize the risks and errors.

6.7 System Modules

The different modules that will make the whole system are as follows.The project contains 2 modules :

1. MOBILE APPLICATION (ANDROID , IOS APPLICATION)
2. WEB PORTAL

Modules Included in Mobile Application are:

1. SIGNUP

   A new user can sign up by entering the following fields o Name o Mobile Number o Street o Zip code o Email o Password o Confirm Password

2. SIGN IN

   User can sign in by entering the username and password

3. PRODUCT CATEGORY

   Categorize the product based on market

4. PRODUCT LISTING

   List the product based on different category and filter the product based on the category.

5. PRODUCT SINGLE PAGE

   Detail description of the product Product details Product recipe Product Price Option to add or minimize the product quantity Option to select corresponding add-on Option to add more products Display total price

6. ADDON PAGE

   Display all addons We can choose the quantity Display addon price Option to add or minimize the quantity of addon Display total price of addon

7. CART PAGE

   See all products we listed See quantity we ordered Price of each product Total price Option to add or minimize product Option to remove product

8. PAYMENT GATEWAY INTEGERATION

   Option to give online payment Paypal (A gateway for the payment) is the API used for it

9. HOME DELIVERY

   User can enter his/her address a list User can select location by using map Google Map API Integration

10. PICK UP

   User can enter his/her address a list User can select location by using map Google Map API Integration

Modules Included in Web Portal are:

1. SIGN IN

   To Sign in to web portal by entering the username and password

2. SEE ALL REGISTERED USERS

   Option to see all Users who registered by using mobile app, Option to remove user, Option to send message to user.

3. ADD/EDIT/DELETE PRODUCTS

   Option to add new products, Option to edit a current product , Option to delete a new product.

4. ADD/EDIT/DELETE CATEGORIES

   Option to add new categories, Option to edit a current categories, Option to delete a new categories.

5. VIEW PENDING ORDERS

   Option to see all pending orders, Option to change status from pending to complete

6. OPTION TO FILTER PENDING/COMPLETED ORDERS

   Option to filter the orders, Option to see all completed orders, Option to see all pending orders.

6.8 Form Design

A form designing means deciding the contents and layout of forms for the purpose of collecting and processing the required information economically and efficiently. The importance of forms designing can be understood because of the following points:

1. Forms are used to collect record and communicate the required information according to the expectations of the needy persons. Therefore, forms are treated as tools of office work. If the forms are badly designed, it reduces the speed of operation of office work.

2. The forms create psychological impact on the people who use it. The people may be frustrated and get tired if the forms are not designed properly.

3. The badly designed forms results in more number of mistakes in clerical work. Hence, there is a need of well-designed forms to avoid mistakes in clerical work.

4. Sometimes, the designed form may project a poor image in the minds of the customers. This may adversely affect the good will of the company.

5. System is the basis for form design. Hence, forms are designed according to the needs of the system. If forms are badly designed, they can ruin a whole system.

6. The well-designed forms contribute much to the efficiency of employees of an organization and efficiency of the system.

7. The cost of forms is less than the cost of completing office forms, transporting and filling of office forms. The ratio will be greater if the forms are badly designed.

6.9 Table Structure

1. Table name:Tbl_registration
   primary key: id

Field Name	Type	Description
id	bigint(10)	id
name	varchar(100)	Name
street	varchar(50)	not null
phone	varchar(100)	Phone
email	varchar(100)	Email
password	varchar(100)	password
confirm password	varchar(100)	confirm password

2. Table nameTb_product category
   primary key: id

Field Name	Type	Description
id	bigint(10)	Id
categorytitle	varchar(100)	Category Titlel
categorydesc	text	Category Description
type	varchar(100)	Type
order	bigint(10)	order

3. Table name:Tb_contact
   primary key:id

Field Name	Type	Description
id	bigint(10)	Id
name	varchar(100)	Name
phone	varchar(30)	Phone
categoryid	bigint(10)	Category id

4. Table name:Tbl_addon page
   primary key:id

Field Name	Type	Description
id	bigint(10)	id
addon	varchar(100)	addon
addon quantity	int(50)	quantity
addonprice	int(100)	addonprice

5. Table name:Tbl_cart Page
   primary key:id

Field Name	Type	Description
id	bigint(10)	Id
addproducts	varchar(100)	products
minimizeproducts	varchar(100)	products
removeproducts	varchar(100)	products

6. Table name:Tbl_delivery
   primary key:id

Field Name	Type	Description
id	bigint(10)	Id
address	varchar(100)	Address details
location	varchar(100)	location details

7. Table name:Tbl_locations
   primary key:id

Field Name	Type	Description
id	bigint(10)	id
location name	varchar(100)	Location name
address	text	Address

8. Table name:Tbl_comments
   primary key:comment-id

Field Name	Type	Description
comment-id	bigint(10)	Id
comment-post-d	bigint(10)	foreignkey
comment-date	varchar(50)	comments
comment-content	varchar(50)	comment content
comment-approvedt	varchar(50)	comment approved

9. Table name:Tbl_posts
primary key:id

Field Name	Type	Description
id	bigint(10)	Id
title	varchar(100)	Title
date	datetime	Date
contents	text	Content Text
title	varchar(100)	title
postmodified	varchar(100)	modified
status	varchar(100)	Status

10. Table name:Tbl_options
primary key:id

Field Name	Type	Description
id	bigint(10)	Id
option	ivarchar(100)	Option
id	bigint(10)	Id
value	varchar(50)	value

11. Table name: Tbl_postmeta
primary key:id

Field Name	Type	Description
meta-id	bigint(10)	Id
post-id	bigint(10)	Foreignkey
meta-value	varchar(100)	metavalue

12. Table name: Tbl_termmeta
primary key:id

Field Name	Type	Description
meta-id	intbigint(10)	Id
term-id	bigint(10)	term Id
meta-key	bigint(10)	Metakey

13. Table name: Tbl_pickup
    primary key:id

Field Name	Type	Description
id	bigint(10)	Id
address	varchar(100)	address details
location	varchar(200)	location
locationdetails	varchar(200)	locationdetails

6.10 UML

6.10.1 Use case diagram

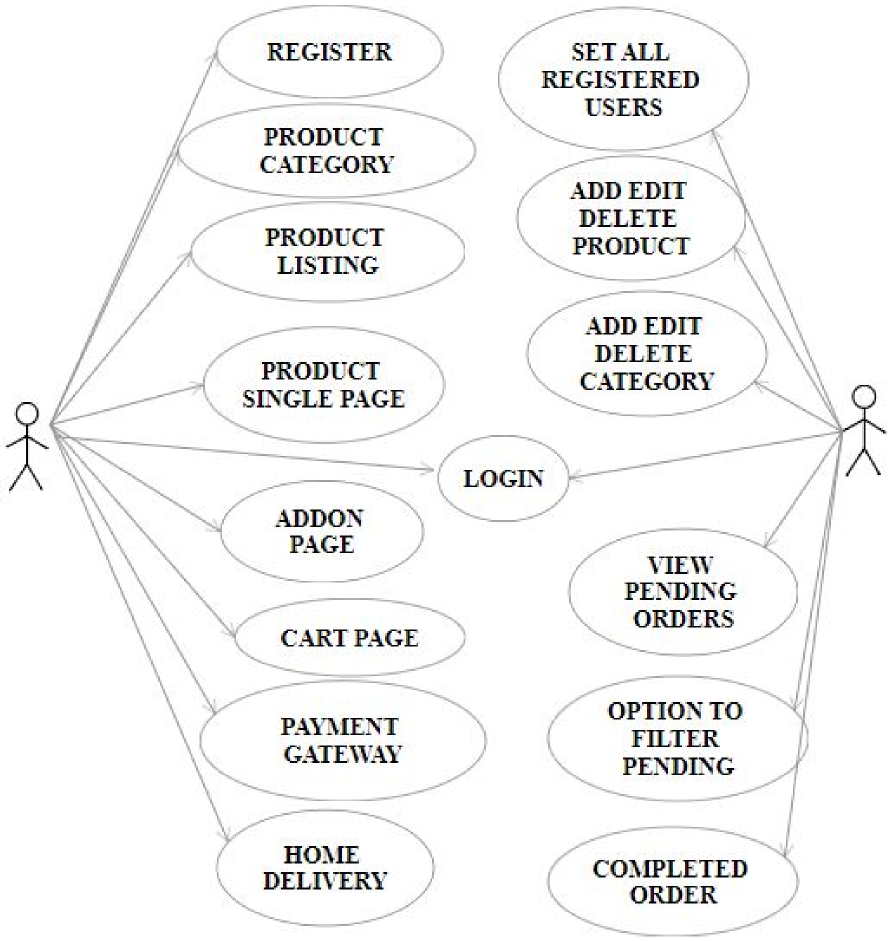


Figure 1: uml diagram

6.10.2 Activity Diagram

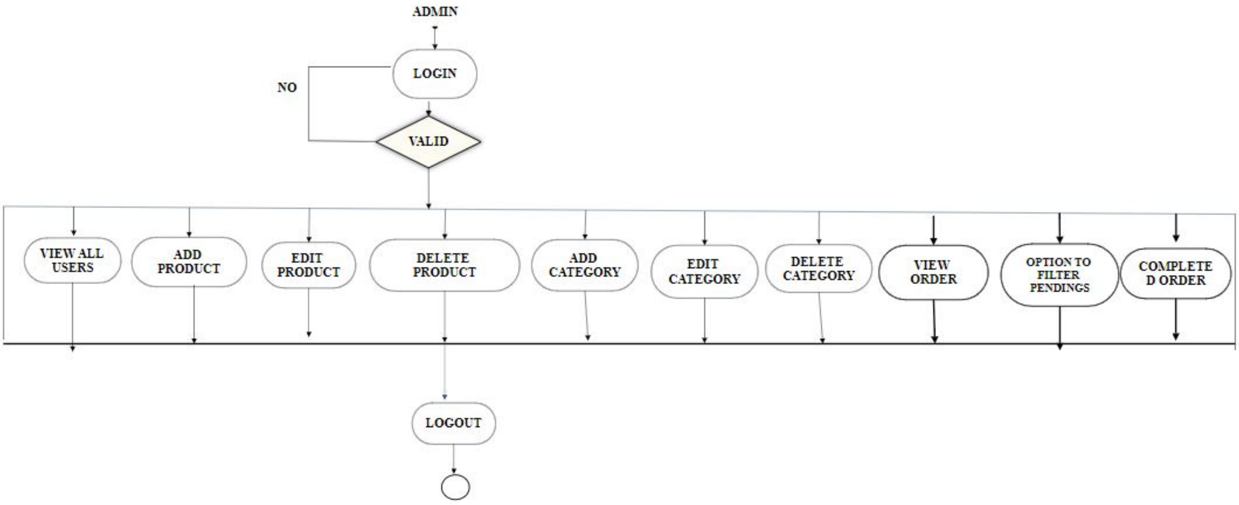


Figure 2: Activity Diagram-Admin

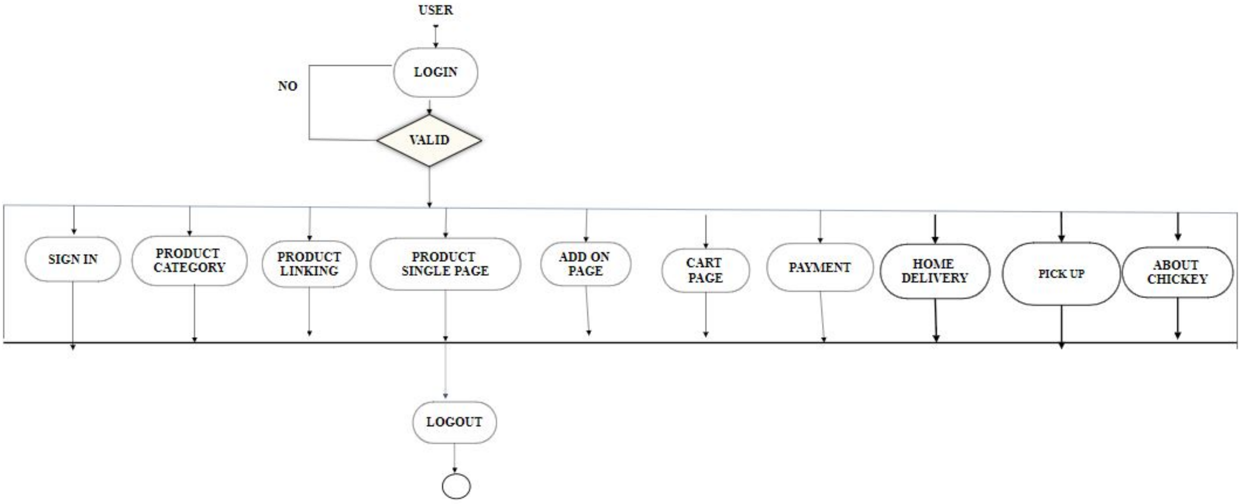


Figure 3: Activity Diagram-User

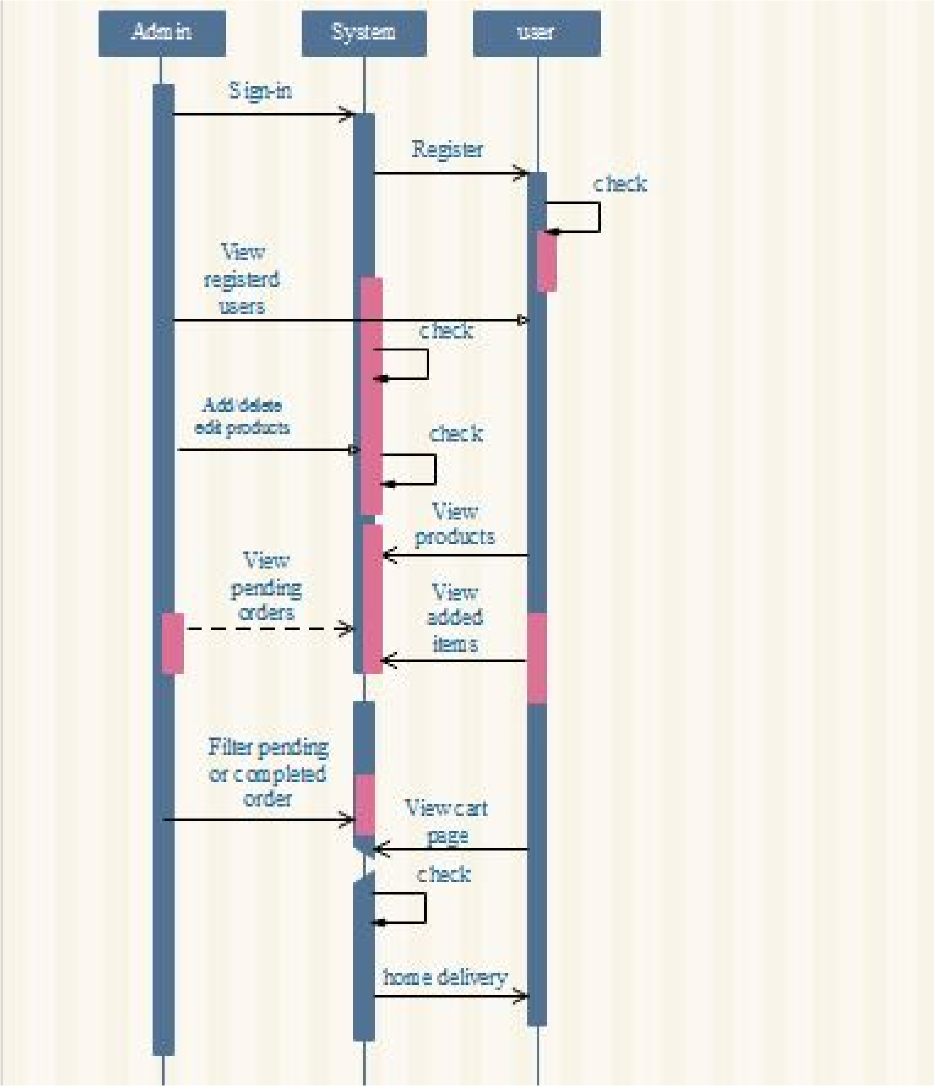


Figure 4: Sequence Diagram

7 SYSTEM TESTING

7.1 Introduction to System Testing

Testing is the process of examining the software to compare the actual behavior with that of the excepted behavior. The major goal of software testing is to demonstrate that faults are not present. In order to achieve this goal the tester executes the program with the intent of finding errors. Though testing cannot show absence of errors but by not showing their presence it is considered that these are not present.

System testing is defined as the process by which one detects the defects in the software. Any software development organization or team has to perform several processes. Software testing is one among them. It is the final opportunity of any programmer to detect and rectify any defects that may have appeared during the software development stage. Testing is a process of testing a program with the explicit intention of finding errors that makes the program fail. In short system testing and quality assurance is a review in software products and related documentation for completion, correctness, reliability and maintainability.

System testing is the first stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct and the goal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptance testing.System testing can be broadly classified into:

1. Syntax testing
2. Unit testing
3. Integration testing
4. Validation testing
5. Output testing
6. User acceptance testing

7.2 Syntax Testing

System testing involves unit testing, integration testing, acceptance testing. Careful planning and scheduling are required to ensure that modules will be available for integration into the evolving software product when needed. A test plan has the following steps:

1. Prepare test plan
2. Specify conditions for user acceptance testing
3. Prepare test data for program testing
4. Prepare test data for transaction path testing
5. Plan user training
6. Compile/assemble programs
7. Prepare job performance aids
8. Prepare operational documents

7.3 Unit Testing

In computer programming, unit testing is a procedure used to validate that individual units of source code are working properly. A unit is the smallest testable part of an application. In procedural programming a unit may be an individual program, function, procedure, etc., while in object-oriented programming, the smallest unit is a method, which may belong to a base/super class, abstract class or derived/child class.Ideally, each test case is independent from the others; mock or fake objects as well as test harnesses can be used to assist testing a module in isolation. Unit testing is typically done by software developers to ensure that the code they have written meets software requirements and behaves as the developer intended.

In this we test each module individually but not integrate the whole system. It focuses verification efforts even in the smallest unit of software design in each module. This is also known as Module Testing. The testing is carried out in the programming style itself. In this testing each module is focused to work satisfactorily as regard to the expected output from the module. There are some validation checks for the fields.

7.4 Integration Testing

Integration testing (sometimes called Integration and Testing, abbreviated I and T) is the phase of software testing in which individual software modules are combined and tested as a group. It follows unit testing and precedes system testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.Data can be lost across an interface, one module can have adverse effect on the other sub-functions, when combined may not produce the desired functions. Integration testing is the systematic testing to uncover the errors within the interface. This testing is done with simple data .The need for an integrated system is to find the overall performance.

The purpose of integration testing is to verify functional, performance and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface. Test cases are constructed to test that all components within assemblages interact correctly, for example across procedure calls or process activations, and this is done after testing individual modules, i.e. unit testing.

7.5 Validation Testing

At the culmination of black box testing (Here the structure of the program is not considered), software is completely assembled as a package .Interface errors have been uncovered and correct and final series of tests, i.e., and validation test begins. The customer defines validation with a simple definition and validation succeeds When the software functions in manner than can be reasonably accepted.

7.6 Output Testing

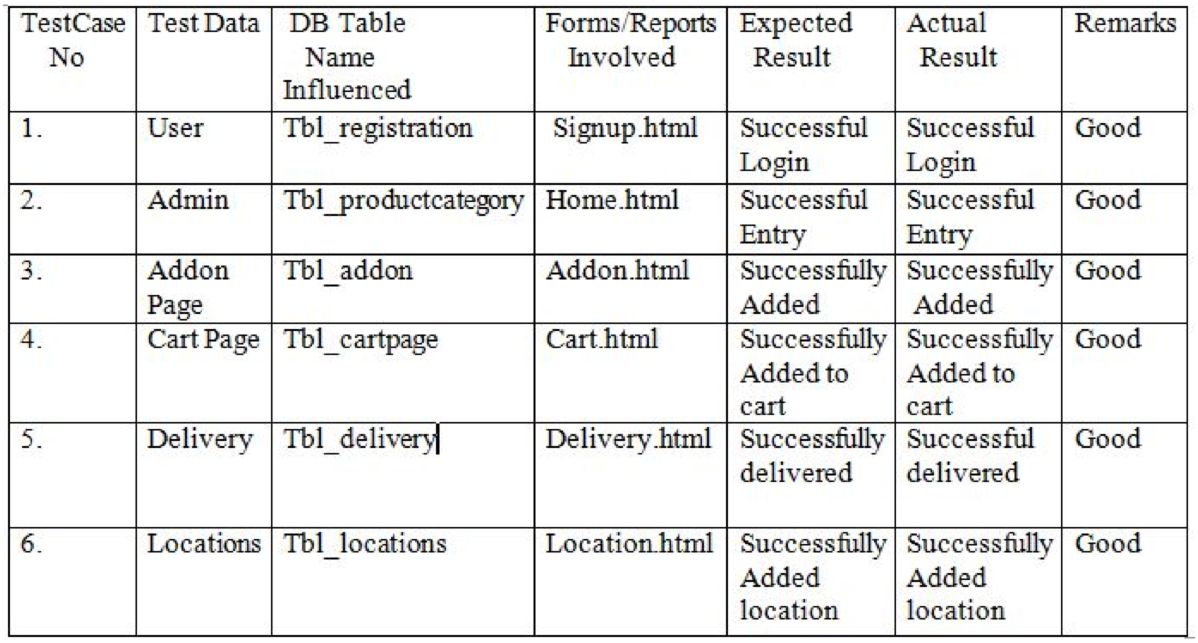
The output of the software should be acceptable to the system user. The output requirement is defined during the system analysis. Testing of the software system is done against the output and the output testing was completed with success.

7.7 User Acceptance Testing

The system is validated by negotiating the existing and proposed system. This test evaluates the system in the real time environment with live data and finds it to be satisfied. This is done by the user. The various possibilities of the data are entered and response from the system is tested once the acceptance testing is signed off by the user.

7.8 Test Cases

A Test Case is a script, program, or other mechanism that exercises a software component to ascertain that a specific correctness assertion is true. In general, it creates a specified initial state, invokes the tested component in a specified way, observes its behavior, and checks to ensure that the behavior was correct.



TestCase No	Test Data	DB Table Name Influenced	Forms/Reports Involved	Expected Result	Actual Result	Remarks
1.	User	Tbl_registration	Signup.html	Successful Login	Successful Login	Good
2.	Admin	Tbl_productcategory	Home.html	Successful Entry	Successful Entry	Good
3.	Addon Page	Tbl_addon	Addon.html	Successfully Added	Successfully Added	Good
4.	Cart Page	Tbl_cartpage	Cart.html	Successfully Added to cart	Successfully Added to cart	Good
5.	Delivery	Tbl_delivery	Delivery.html	Successfully delivered	Successful delivered	Good
6.	Locations	Tbl_locations	Location.html	Successfully Added location	Successfully Added location	Good

7.9 Manual Quality Assurance(QA)

QA includes activities that ensure the implementation of processes,procedures and standards in context to verification of developed software and intended requirements. Manual testing is flexible, has low entry costs, and offers the very important human factor when testing usability.Manual software quality assurance is the process where the tester is performing a series of tests using the software like the end user would to compare the behavior of the developed code against expected behavior and make sure it works according to the specified requirements.

The type of testing, manual or automation, depends on various factors, including project requirements, budget, timeline, expertise, and suitability. Organizations need to understand these requirements to be able to decide when using manual testing tools best suit the situation.

8 SYSTEM SECURITY

Any mobile-based application that manages sensitive information or causes actions that can improperly harm individuals is a target for improper or illegal penetration. Penetration spans a broad range of activities : hackers who attempt to penetrate systems for sport; disgruntled employees who attempt to penetrate for revenge; dishonest individuals who attempt to penetrate for illicit personal gain.

The proposed system, provide security technology attempts to verify that protection mechanisms built into a application will, in fact, protect it from improper penetration. The systems security must, of course, be tested for invulnerability from frontal attack, but must also be tested for invulnerability from rear attack.

The security and integrity of data is ensured in proposed application. The access to data is restricted to those who register with the admin web portal. While registering with the system, the users are provided with a username and password. Only the registered person can login into to the system.

9 SYSTEM IMPLEMENTATION

9.1 Introduction to System Implementation

The implementation is the final state 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. A software implementation method is a systematically structured approach to effectively integrate software based service or component into the workflow of an organizational structure or an individual end-user. This entry focuses on the process modeling (Process Modeling), a process model is a description of a process at the type level, side of the implementation of large product software, using the implementation of Enterprise Resource Planning systems as the main example to elaborate on. A product software implementation method is a blueprint to get users and/or organizations running with a specific software product. The method is a set of rules and views to cope with the most common issues that occur when implementing a software product: business alignment from the organizational view and acceptance from the human view. The implementation of product software, as the final link in the deployment chain of software production, is in a financial perspective of a major issue. The implementation phase of the software development is concerned with translating design specification into source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented. Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period of time. Implementation is the third phase of the system process.

9.2 Training

An analysis of user training focuses on two factors:

1. User capabilities

2. Nature of the system to be installed.

Users range from the native to the highly sophisticated. They approach it as concrete learners, learning how to use the system without trying to understand which abstract principles determine which function. The distinction between concrete and formal (student type) learning says about what one can expect from trainees in general.These project also sophisticated the user capabilities and the corresponding nature of the system to be installed.

9.3 Conversion

Conversion refers to changing from one design to another system. The main objective of conversion is to put tested system into operation while holding costs, risks, and personal irritation to minimum. The various tasks involved in conversion are:

1. Creating computer compatible files.

2. Training the operating staffs.

3. Installing terminals and hardware.

The project entitled Ilahia Exam Cell Automation agreed the conversion phases that begin with a review of the project plan, the system test documentation and the implementation plan. And also conversion portion of the implementation plan is finalized and approved. Files are converted.

9.4 Post Implementation Review

Every system requires periodic evaluation after implementation. A post implementation review measures the systems performance against predefined requirements. Unlike system testing, which determines where the system fails so that the necessary adjustments can be made, a post-implementation review determines how well the system continues to meet performances specifications. It is done after design and conversion are complete. It also provides information to determine whether major redesign is necessary.

9.5 System Maintenance

Software maintenance is the modification of a software product after delivery to correct faults, to improve performance or other attributes. This section describes the six software maintenance processes as:

1. The implementation processes contains software preparation and transition activities, such as the conception and creation of the maintenance plan, the preparation for handling problems identified during development, and the follow-up on product configuration management.

2. The problem and modification analysis process, which is executed once the application has become the responsibility of the maintenance group. The maintenance programmer must analyze each request, confirm it (by reproducing the situation) and check its validity, investigate it and propose a solution, document the request and the solution proposal, and, finally, obtain all the required authorizations to apply the modifications.

3. The process considering the implementation of the modification itself.

4. The process acceptance of the modification, by confirming the modified work with the individual who submitted the request in order to make sure the modification provided a solution.

5. The migration process is exceptional, and is not part of daily maintenance tasks. If the software must be ported to another platform without any change in functionality, this process will be used and a maintenance project team is likely to be assigned to this task.

6. Finally, the last maintenance process, also an event which does not occur on a daily basis, is the retirement of a piece of software.

10 SYSTEM EVALUATION

Although system evaluation is an ongoing process throughout the performance testing effort, it offers greater value when conducted early in the test project. The intent of system evaluation is to collect information about the project as a whole, the functions of the system, the expected user activities, the system architecture, and any other details that are helpful in guiding performance testing to achieve the specific needs of the project.

1. Your need to evaluate and select software that meets your business requirements.

2. Your need to evaluate and select a partner that is capable of delivering the most benefit to your business from your software investment, as well as managing the risks inherent in system implementation projects.

3. Your time and ours is valuable; at each step along the way we will each decide whether or not it is beneficial to proceed.

To help you with your selection, this evaluation process is designed to give us both a clear understanding of the systems to be implemented and the corresponding benefits of the partnership.This information provides a foundation for collecting the performance goals and requirements, characterizing the workload, creating performance-testing strategies and plans, and assessing project and system risks. A thorough understanding of the system under test is critical to a successful performance-testing effort. The measurements gathered during later stages are only as accurate as the models that are developed and validated in this stage. The evaluation provides a foundation for determining acceptable performance; specifying performance requirements of the software, system, or component(s); and identifying any risks to the effort before testing even begins. System evaluation providing in these project is needed to evaluate and select the requirements and managing the risk in system implementation on project. Also it is valuable in time so that way it is beneficial in each step.

11 CONCLUSION

What I propose is an online ordering system originally designed for use in college cafeterias, but just as applicable in any food delivery industry. The main advantage of this system is that it greatly simplifies the ordering process for both the customer and the restaurant. The system also greatly lightens the load on the restaurants end, as the entire process of taking orders is automated. Once an order is placed on the webpage that will be designed, it is placed into the database and then retrieved, in pretty much real-time, by a web application on the restaurants end. Within this application, all items in the order are displayed, along with their corresponding options and delivery details, in a concise and easy to read manner. This allows the restaurant employees to quickly go through the orders as they are placed and produce the necessary items with minimal delay and confusion. The greatest advantage of this system is its Flexibility.

11.1 Future Enhancement

The project entitled chickey is currently now available only in UAE.The future scope of this application is to extend the availability of chickey product to other regions world wide,this app can manage the purchase and delivery processes more extremely in the future.There will be updation to the current version to make the application more customer friendly.
Though there is bulk quantity of data is handled by the system. While developing a particular system, the designer should bother about the damages and modification that may appear in the system. Then only the maintenance of the particular system can be carried out. After developing a particular system, it should give to a client. While using the system, the client may modify or may able to cause any damage. This is known as maintenance. So the developing system must be user friendly. Then only the client can use the system by their own wish. The data can be updated and regulated accordingly.

12 APPENDIX

12.1 Appendix A

12.1.1 CODING

- SIGNIN

```
<ion-header>
<ion-navbar>
<button ion-button end menuToggle>
<ion-icon name="menu"></ion-icon>
</button>
<ion-title><img src="assets/images/logo.png" /></ion-title>
</ion-navbar>
</ion-header>
<ion-content no-padding>
<ion-grid no-padding>
<ion-row>
<div class="profile_pics"><img src="assets/images/regpr.png"
class="prof_pic"></div>
<div class="profile_fields">
<form class="list" [formGroup]="signupForm">
<ion-input type="text" [(ngModel)]="reg.username" formControlName="username"
 class="prof_input" placeholder="Username"></ion-input>
<ion-input type="text" [(ngModel)]="reg.name" formControlName="name"
class="prof_input" placeholder="Name"></ion-input>
<ion-input type="tel" [(ngModel)]="reg.mobile" formControlName="mobile"
class="prof_input" placeholder="Mobile Number"></ion-input>
<ion-input type="text" [(ngModel)]="reg.street" formControlName="street"
 class="prof_input" placeholder="Street"></ion-input>
<ion-input type="text" [(ngModel)]="reg.zip" formControlName="zip"
class="prof_input" placeholder="Zip Code"></ion-input>
<ion-input type="email" [(ngModel)]="reg.email" formControlName="email"
 class="prof_input" placeholder="Email"></ion-input>
<ion-input type="password" [(ngModel)]="reg.password"
formControlName="password"
class="prof_input"
 placeholder="Password" validateEqual="confirm_password"
reverse="true"></ion-input>
```

```
<ion-input type="password" [(ngModel)]="reg.confirmpass"
formControlName="confirm_password" class="prof_input"
 validateEqual="password" placeholder="Confirm Password">
</ion-input>
<button class="sign_in_btn" (click)="opensuccess()"
type="submit" [disabled]="!signupForm.valid"> Register</button>
</form>
</div>
</ion-row>

</ion-grid>
</ion-content>
<!--<ion-footer>arersdrsdrds</ion-footer>-->
<ion-footer>
<ion-toolbar>
<ion-list style="margin: 0;">
<ion-grid no-padding>
<ion-row>
<ion-col col>
<button [navPush]="LocationsPage" class="footer-menu" ion-item>
<img src="assets/images/location.png" />
</button>
</ion-col>
<ion-col col>
<button [navPush]="CartPage" class="footer-menu" ion-item>
<img src="assets/images/cart.png" />
</button>
<ion-badge class="count_cart_foot" *ngIf="this.cartItemsCount">
{{this.cartItemsCount}}</ion-badge>

</ion-col>
<ion-col col>
<button (click)="openhomepage()" class="footer-menu" ion-item>
<img src="assets/images/home.png" />
</button>
</ion-col>
<ion-col col>
<button [navPush]="DeliveryPage" class="footer-menu" ion-item>
<img src="assets/images/track.png" />
</button>
</ion-col>
```

```
<ion-col col>
<button (click)="shareapp()" class="footer-menu" ion-item>
<img src="assets/images/more.png" />
</button>
</ion-col>
</ion-row>
</ion-grid>
</ion-list>
</ion-toolbar>
</ion-footer>
```

- LOGIN PAGE

```
<ion-header>
<ion-navbar>
<button ion-button end menuToggle>
<ion-icon name="menu"></ion-icon>
</button>
<ion-title><img src="assets/images/logo.png" /></ion-title>
</ion-navbar>
</ion-header>
<ion-content no-padding>
<ion-grid no-padding>
<ion-row>
<div class="profile_pics"><img src="assets/images/prof-pic.png"
class="prof_pic"></div>
<div class="profile_fields">
<form class="list" [formGroup]="loginForm">
<input type="text" [(ngModel)]="reg.username" formControlName="username"
 class="prof_input" placeholder="Username">
<input type="password" [(ngModel)]="reg.password"
formControlName="password"
 class="prof_input" placeholder="Password">
<button class="sign_in_btn" (click)="user_login()"> Sign In</button>
</form>
</div>
</ion-row>
<ion-row class="forgot_signin">
<span class="forgot_pwd">Forgot password ?</span>
<span class="line_strgt"></span>
<span class="forgot_pwd">Not a member yet ?</span>
```

```
<button class="sign_in_btn" [navPush]="SignupPage"> Sign Up</button>
</ion-row>
</ion-grid>
</ion-content>
<!--<ion-footer>arersdrsdrds</ion-footer>-->
<ion-footer>
<ion-toolbar>
<ion-list style="margin: 0;">
<ion-grid no-padding>
<ion-row>
<ion-col col>
<button [navPush]="DeliveryPage" class="footer-menu" ion-item>
<img src="assets/images/location.png" />
</button>
</ion-col>
<ion-col col>
<button [navPush]="CartPage" class="footer-menu" ion-item>
<img src="assets/images/cart.png" />
</button>
<ion-badge class="count_cart_foot" *ngIf="this.cartItemsCount">
{{this.cartItemsCount}}</ion-badge>

</ion-col>
<ion-col col>
<button (click)="openhomepage()" class="footer-menu" ion-item>
<img src="assets/images/home.png" />
</button>
</ion-col>
<ion-col col>
<button [navPush]="DeliveryPage" class="footer-menu" ion-item>
<img src="assets/images/track.png" />
</button>
</ion-col>
<ion-col col>
<button (click)="shareapp()" class="footer-menu" ion-item>
<img src="assets/images/more.png" />
</button>
</ion-col>
</ion-row>
</ion-grid>
</ion-list>
```

```
</ion-toolbar>
</ion-footer>
```

- ADDON PAGE

```
<ion-header>
<ion-toolbar>
<span class="price_head">Adon Price <small>SAR</small>
 {{adonstotal}}/-</span>
<ion-title>
Adons
</ion-title>
</ion-toolbar>
</ion-header>
<ion-content>
<ion-list>
<ion-item *ngFor="let p of adons; let i = index">
<ion-grid no-padding>
<ion-row>
<ion-col col-2 class="img_wrap">
<img class="adon_img" title="{{p.title.rendered}}"
 src="{{p.featured_image_urls.product_large}}" />
</ion-col>
<ion-col col-7 class="adon_details">
<h3>{{p.title.rendered}}</h3>
<h4 class="adon_price">SR {{p.acf.price}}/-</h4>
<span class="adon_info">Add this Ketchup for {{p.title.rendered}}
 {{p.acf.price}}/-</span>
</ion-col>
<ion-col col-3>
<ul class="add_cart_amount">
<li><button (click)="decrement(i)" class="decrement">
<img src="assets/images/decrese.png"/></button></li>
<li class="countspc" ><span>{{p.qty}}</span></li>
<li><button (click)="increment(i)" class="increment">
<img src="assets/images/increse.png"/></button></li>
</ul>
</ion-col>
</ion-row>
</ion-grid>
</ion-item>
```

```
</ion-list>
</ion-content>
<ion-footer>
<ion-toolbar>
<ion-buttons start>
<button class="bottom_addon" ion-button (click)="dismiss()">
<span ion-text color="primary">Cancel</span>
</button>
<button class="bottom_addon" ion-button (click)="addAdons()">
<span ion-text color="primary">OK</span>
</button>
</ion-buttons>
</ion-toolbar>
</ion-footer>
```

- DELIVERY PAGE

```
<ion-header>
<ion-navbar>
<button ion-button end menuToggle>
<ion-icon name="menu"></ion-icon>
</button>
<ion-title><img src="assets/images/logo.png" /></ion-title>
</ion-navbar>
</ion-header>
<ion-content no-padding>
<ion-grid no-padding>
<ion-row>
<div class="outer_sec_del">
<span class="strp_top">
</span>
<div class="tab_top_delivery">
<button class="shopping_del">
<img src="assets/images/shop.png"/></button>
<button class="location_del">
<img src="assets/images/loc_del.png"/></button>
<button class="cart_del">
<img src="assets/images/cart_del.png"/></button>
</div>
</div>
</ion-row>
```

```
<!--<ion-row>-->
<div class="tab_del_sec">
<ion-segment [(ngModel)]="user.deliveryop" (ionChange)
="onSegmentChanged
(user.deliveryop,sub_delivery,sub_pickup)">
<ion-segment-button value="delivery">
Delivery
</ion-segment-button>
<ion-segment-button value="pickup">
Pickup
</ion-segment-button>
</ion-segment>
</div>
<div [ngSwitch]="user.deliveryop">
<div *ngSwitchCase="'delivery'">
<div [ngSwitch]="sub_delivery">
<div *ngSwitchCase="'map'">
<div id="map_main" [ngInit]="loadMap">
<!--<img class="img_reponsive" src="assets/images/map.png"/>-->
</div>
<div class="delivery_fields">
<input type="hidden" value="deliver" class="del_input" [(ngModel)]
="user.action" >
<input type="text" class="del_input" [(ngModel)]
="user.name" placeholder="Name">
<input type="tel" class="del_input" [(ngModel)]
="user.contact" placeholder="Contact Number">
<input type="text" class="del_input" [(ngModel)]
="user.address1" placeholder="Address">
<input type="text" class="del_input" [(ngModel)]
="user.address2" placeholder="Address 2">

</div>
</div>
<div *ngSwitchCase="'list_address'">
<!--<ion-item>-->
<div class="delivery_fields">
<h3 class="address_head">Delivery Address</h3>
<span class="address_sub">
Enter Your Address where you want to deliver</span>
<input type="hidden" value="deliver"
```

```
class="del_input" [(ngModel)]="user.action" >
<input type="text" class="del_input" [(ngModel)]
="user.name" placeholder="Name">
<input type="tel" class="del_input" [(ngModel)]
="user.contact" placeholder="Contact Number">
<input type="text" class="del_input" [(ngModel)]
="user.city" placeholder="City">
<input type="text" class="del_input" [(ngModel)
]="user.address1" placeholder="Address">
<input type="text" class="del_input" [(ngModel)]
="user.address2" placeholder="Address 2">
<!-- <ion-list>
<ion-item class="city_address">
<ion-label class="city_labl">City</ion-label>
<input type="text" class="del_input" placeholder="Name">
</ion-item>
</ion-list>-->
</div>
<!--</ion-item>-->
</div>
</div>
<div padding class="del_inner_seg">
<ion-segment [(ngModel)]="sub_delivery"
 (ionChange)="onSegmentChanged
(user.deliveryop,sub_delivery,sub_pickup)">
<ion-segment-button value="map" class="inner_seg_btn">
Map
</ion-segment-button>
<ion-segment-button value="list_address" class="inner_seg_btn">
List
</ion-segment-button>
<button class="inner_seg_btn_proceed
 segment-md segment-button" (click)="checkout()">
Proceed
</button>
</ion-segment>
</div>
</div>
<div *ngSwitchCase="'pickup'">
<div [ngSwitch]="sub_pickup">
<div *ngSwitchCase="'map'">
```

```
<div id="map_pickup" [ngInitNew]="loadMapPickup">
<!--<img class="img_reponsive" src="assets/images/map.png"/>-->
</div>
<div class="delivery_fields">
<input type="hidden" value="pickup"
class="del_input" [(ngModel)]="user.action" >
<input type="text" class="del_input"
[(ngModel)]="user.name" placeholder="Name">
<input type="text" class="del_input"
 [(ngModel)]="user.contact" placeholder="Contact Number">
</div>
</div>
<div *ngSwitchCase="'list_address'">
<!--<ion-item>-->
<div class="delivery_fields">
<h3 class="address_head">Select outlet</h3>
<span class="address_sub">
Enter your nearest outlet where you want to pick up</span>
<input type="hidden" value="pickup"
 class="del_input" [(ngModel)]="user.action" />
<ion-item class="city_address">
<ion-label class="city_labl">City</ion-label>
<ion-select [(ngModel)]="user.selected_city"
class="city_selct" (ionChange)=citySelected()>
<ion-option value="{{p.id}}"
*ngFor="let p of cities">{{p.name}}</ion-option>
</ion-select>
</ion-item> <ion-item class="city_address">
<ion-label class="city_labl">Area/Branch</ion-label>
<ion-select [(ngModel)]="user.Selected_branch"
class="city_selct">
<ion-option value="{{p.id}}"
*ngFor="let p of filterdstores">
{{p.title.rendered}}</ion-option>
</ion-select>
</ion-item>
<input type="text" class="del_input"
 [(ngModel)]="user.name" placeholder="Name">
<input type="text" class="del_input"
 [(ngModel)]="user.contact" placeholder="Contact Number">
<!-- <ion-list>
```

```
<ion-item class="city_address">
<ion-label class="city_labl">City</ion-label>
<ion-select [(ngModel)]="gaming2" class="city_selct">
<ion-option value="nes">Kochi</ion-option>
<ion-option value="n64">Kottayam</ion-option>
<ion-option value="ps">TVM</ion-option>
<ion-option value="genesis">Kozhikode</ion-option>
<ion-option value="saturn">Malappuram</ion-option>
<ion-option value="snes">Pala</ion-option>
</ion-select>
</ion-item>
</ion-list>-->
</div>
<!--</ion-item>-->
</div>
</div>
<div padding class="del_inner_seg">
<ion-segment [(ngModel)]="sub_pickup"
(ionChange)="onSegmentChanged
(user.deliveryop,sub_delivery,sub_pickup)">
<ion-segment-button value="map"
 class="inner_seg_btn">
Map
</ion-segment-button>
<ion-segment-button value="list_address"
class="inner_seg_btn">
List
</ion-segment-button>
<button class="inner_seg_btn_proceed segment
-md segment-button"
(click)="checkout()">
Proceed
</button>
</ion-segment>
</div>
</div>
</div>
</ion-grid>
</ion-content>
<!--<ion-footer>arersdrsdrds</ion-footer>-->
<ion-footer>
```

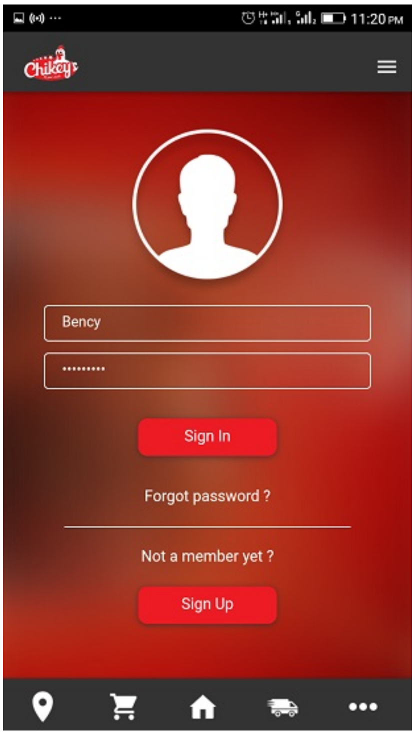
```
<ion-toolbar>
<ion-list style="margin: 0;">
<ion-grid no-padding>
<ion-row>
<ion-col col>
<button [navPush]="LocationsPage"
 class="footer-menu" ion-item>
<img src="assets/images/location.png" />
</button>
</ion-col>
<ion-col col>
<button [navPush]="CartPage"
 class="footer-menu" ion-item>
<img src="assets/images/cart.png" />
</button>
<ion-badge class="count_cart_foot"
 *ngIf="this.cartItemsCount">
{{this.cartItemsCount}}</ion-badge>

</ion-col>
<ion-col col>
<button (click)="openhomepage()"
 class="footer-menu" ion-item>
<img src="assets/images/home.png" />
</button>
</ion-col>
<ion-col col>
<button [navPush]="DeliveryPage"
 class="footer-menu" ion-item>
<img src="assets/images/track.png" />
</button>
</ion-col>
<ion-col col>
<button (click)="shareapp()"
class="footer-menu" ion-item>
<img src="assets/images/more.png" />
</button>
</ion-col>
</ion-row>
</ion-grid>
</ion-list>
```

```
</ion-toolbar>
</ion-footer>
```

12.2 Appendix B

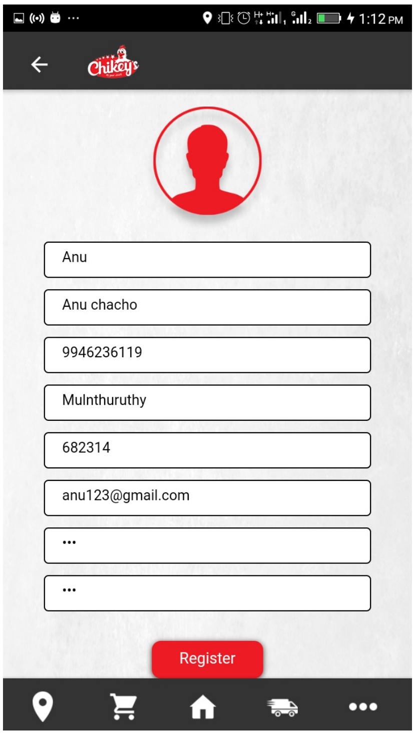
12.2.1 FORM DESIGN



Bency

•••••••••

Figure 5: Login



Anu

Anu chacho

9946236119

Mulnthuruthy

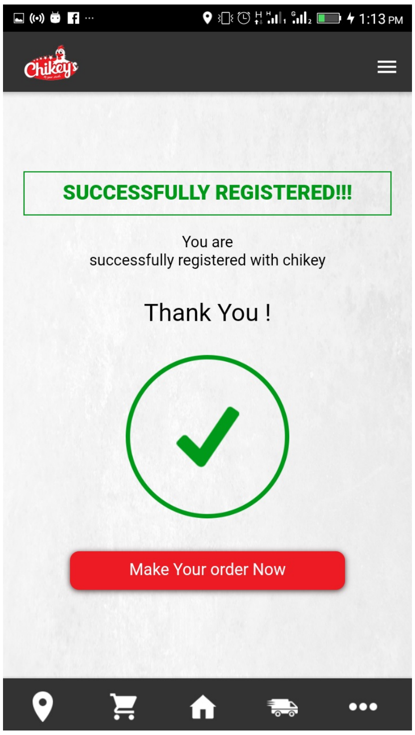
682314

anu123@gmail.com

•••

•••

Figure 6: Registration



1:13 PM

SUCCESSFULLY REGISTERED!!!

You are
successfully registered with chikey

Thank You !

Figure 7: Registration sucessfull

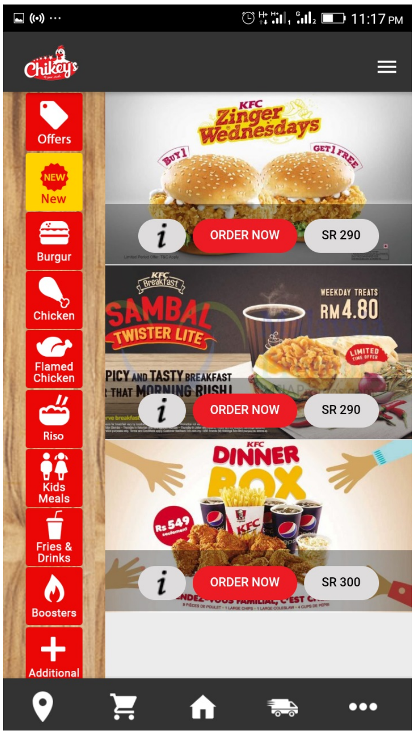
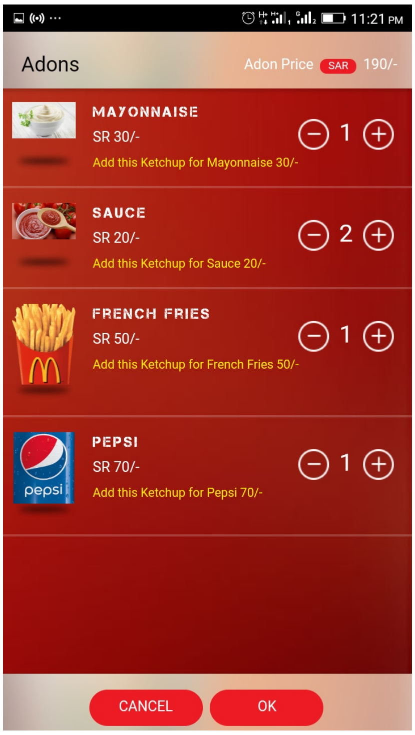


Figure 8: Home Page



Adons

Adon Price SAR 190/-

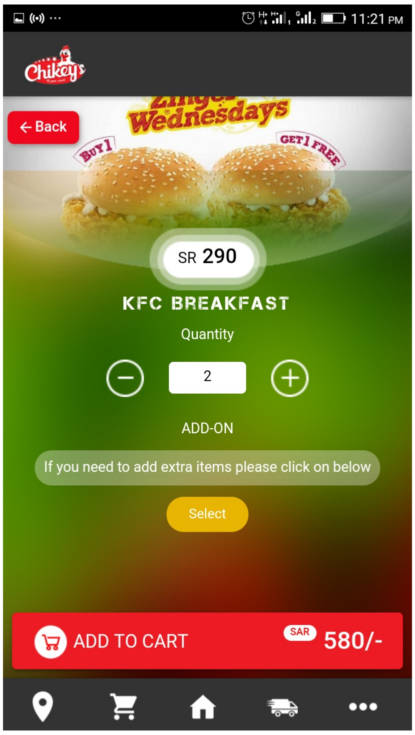
MAYONNAISE
SR 30/-
Add this Ketchup for Mayonnaise 30/-
1

SAUCE
SR 20/-
Add this Ketchup for Sauce 20/-
2

FRENCH FRIES
SR 50/-
Add this Ketchup for French Fries 50/-
1

PEPSI
SR 70/-
Add this Ketchup for Pepsi 70/-
1

Figure 9: Add on



11:21 PM

← Back

SR 290

KFC BREAKFAST

Quantity

2

ADD-ON

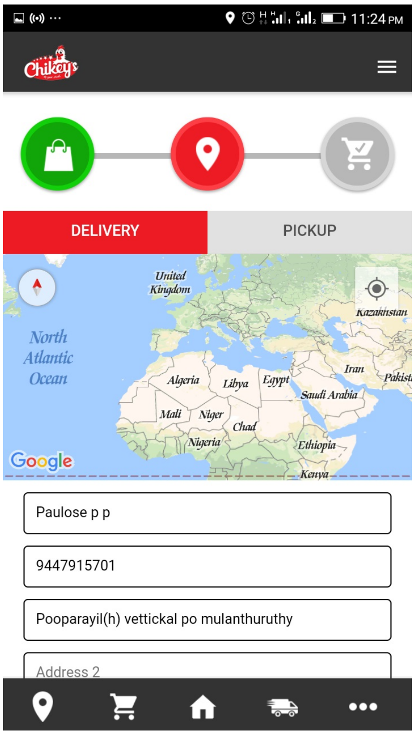
If you need to add extra items please click on below

Select

ADD TO CART

SAR 580/-

Figure 10: Cart Page



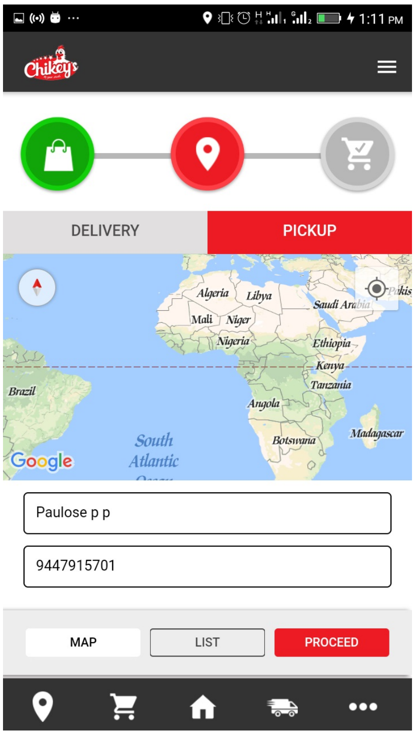
Paulose p p

9447915701

Pooparayil(h) vettickal po mulanthuruthy

Address 2

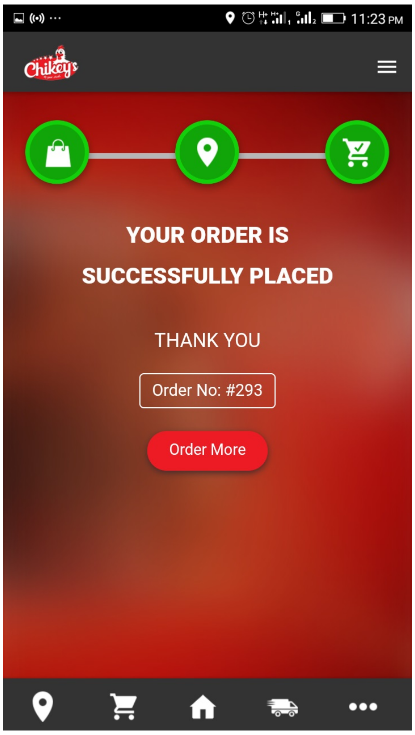
Figure 11: Delivery



Paulose p p

9447915701

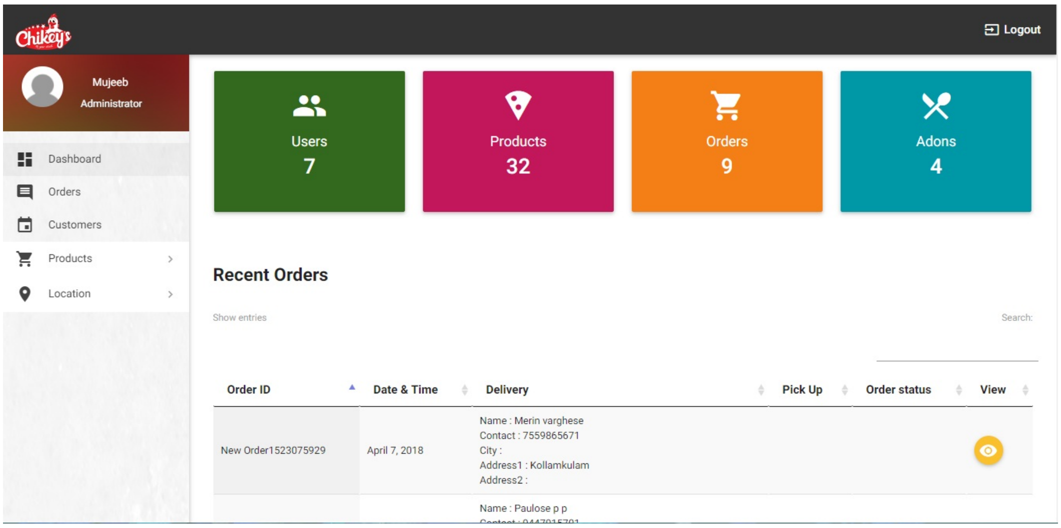
Figure 12: Pick up



YOUR ORDER IS SUCCESSFULLY PLACED

THANK YOU

Figure 13: Order sucess



Users 7

Products 32

Orders 9

Adons 4

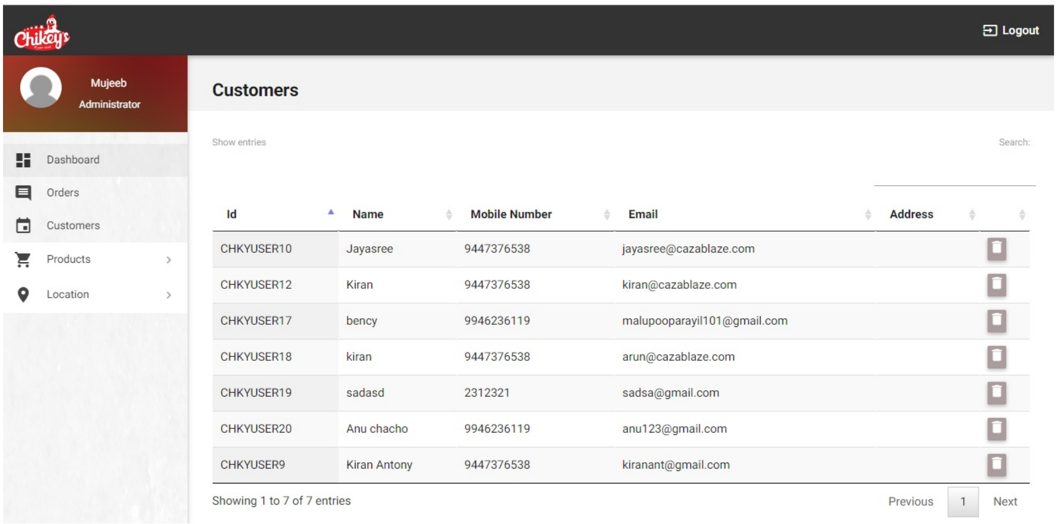
Recent Orders

Show entries

Search:

Order ID	Date & Time	Delivery	Pick Up	Order status	View
New Order1523075929	April 7, 2018	Name : Merin varghese Contact : 7559865671 City : Address1 : Kollamkulam Address2 :			
		Name : Paulose p p Contact : 9447015701			

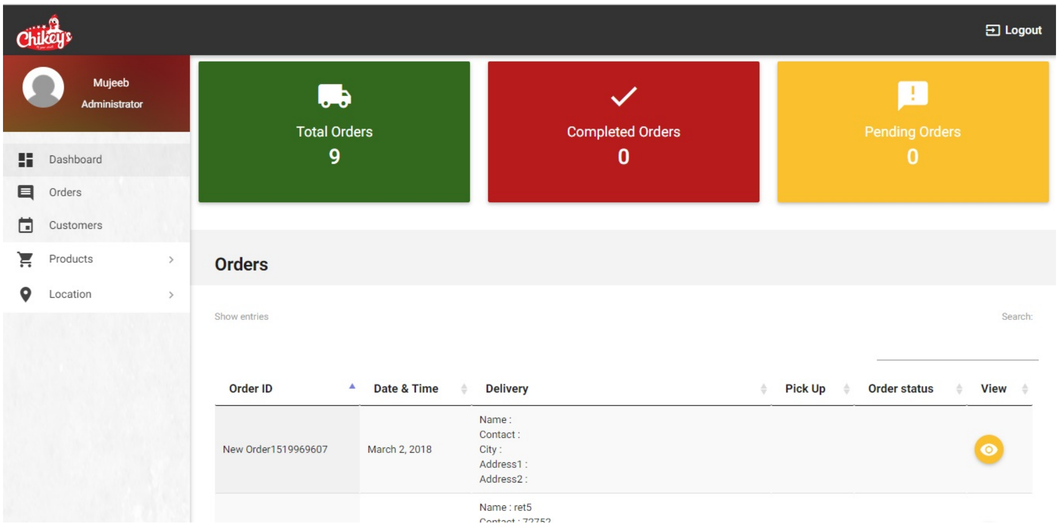
Figure 14: Dashboards



Id	Name	Mobile Number	Email	Address	
CHKYUSER10	Jayasree	9447376538	jayasree@cazablaze.com		
CHKYUSER12	Kiran	9447376538	kiran@cazablaze.com		
CHKYUSER17	bency	9946236119	malupooparayil101@gmail.com		
CHKYUSER18	kiran	9447376538	arun@cazablaze.com		
CHKYUSER19	sadasd	2312321	sadsa@gmail.com		
CHKYUSER20	Anu chacho	9946236119	anu123@gmail.com		
CHKYUSER9	Kiran Antony	9447376538	kiranant@gmail.com		

Showing 1 to 7 of 7 entries

Figure 15: Customers



Total Orders
9

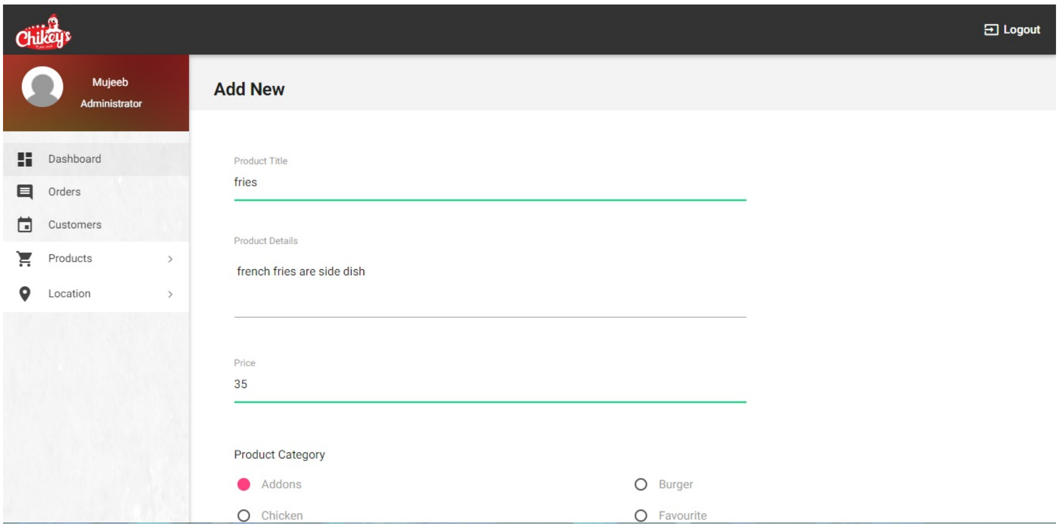
Completed Orders
0

Pending Orders
0

Orders

Order ID	Date & Time	Delivery	Pick Up	Order status	View
New Order1519969607	March 2, 2018	Name : Contact : City : Address1 : Address2 :			👁
		Name : ret5 Contact : 72752			

Figure 16: Orders



Add New

Product Title: fries

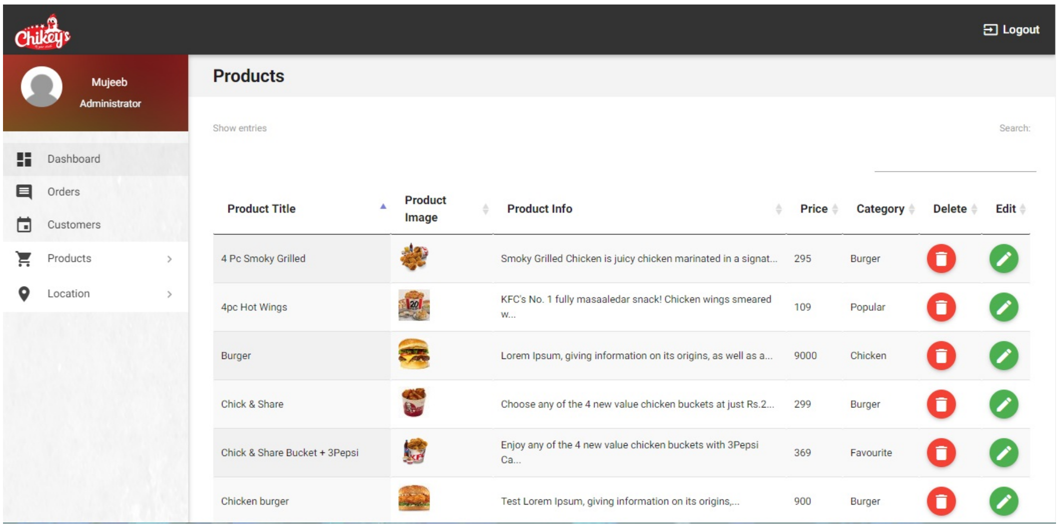
Product Details: french fries are side dish

Price: 35

Product Category

● Addons
○ Burger
○ Chicken
○ Favourite

Figure 17: Add New Product



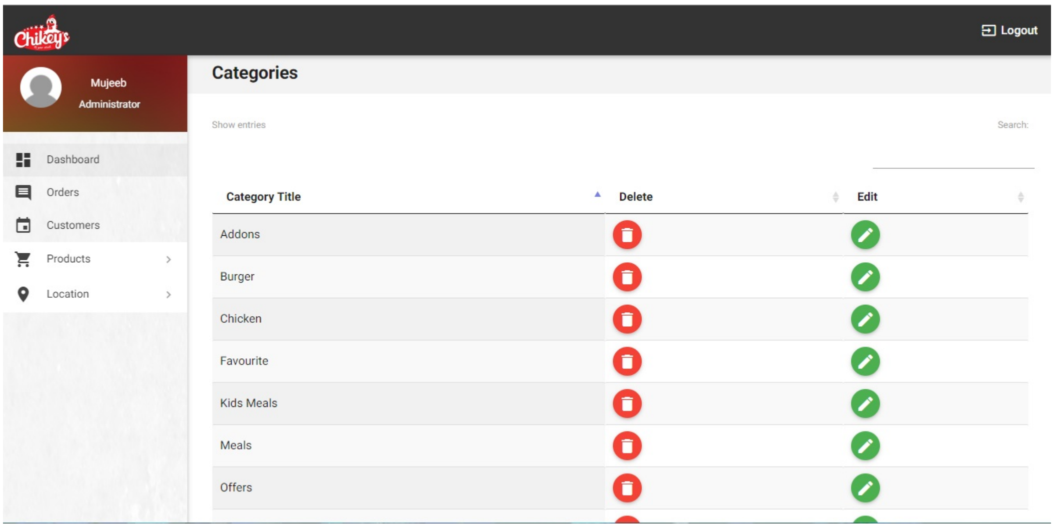
Products

Show entries

Search:

Product Title	Product Image	Product Info	Price	Category	Delete	Edit
4 Pc Smoky Grilled		Smoky Grilled Chicken is juicy chicken marinated in a signat...	295	Burger		
4pc Hot Wings		KFC's No. 1 fully masaaledar snack! Chicken wings smeared w...	109	Popular		
Burger		Lorem Ipsum, giving information on its origins, as well as a...	9000	Chicken		
Chick & Share		Choose any of the 4 new value chicken buckets at just Rs.2...	299	Burger		
Chick & Share Bucket + 3Pepsi		Enjoy any of the 4 new value chicken buckets with 3Pepsi Ca...	369	Favourite		
Chicken burger		Test Lorem Ipsum, giving information on its origins,...	900	Burger		

Figure 18: Products

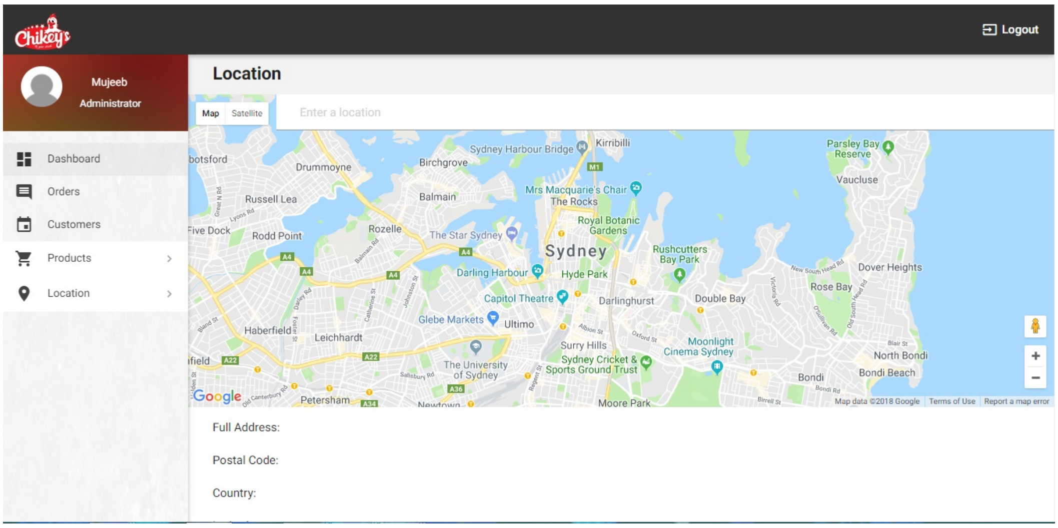


Categories

Category Title	Delete	Edit
Addons	🗑	✎
Burger	🗑	✎
Chicken	🗑	✎
Favourite	🗑	✎
Kids Meals	🗑	✎
Meals	🗑	✎
Offers	🗑	✎

Figure 19: Categories

v



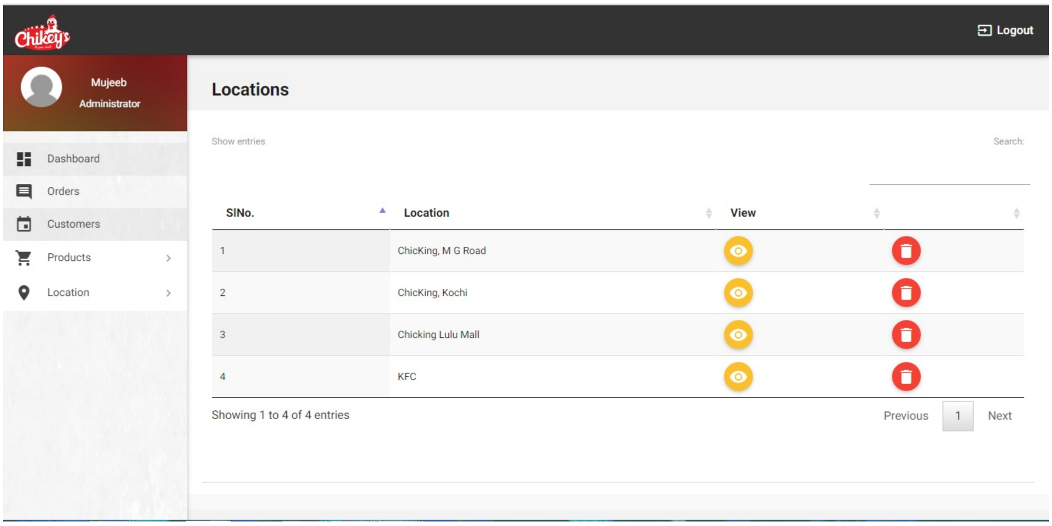
Location

Full Address:

Postal Code:

Country:

Figure 20: Location Map



Locations

Show entries

Search:

SlNo.	Location	View	
1	ChicKing, M G Road		
2	ChicKing, Kochi		
3	Chicking Lulu Mall		
4	KFC		

Showing 1 to 4 of 4 entries

Figure 21: Locations

12.3 Appendix C

12.3.1 Acronyms

UML - Unified Modeling Language
HTML - Hyper Text Markup Language
PHP - Hyper Text PreProcessor
XML - Extensible Markup Language
JVM - Java Virtual Machine
CLI - Command Line Interface
GUI - Graphical User Interface
CSS - Cascading Style Sheet
GB - Giga Bytes
MB - Mega Bytes
QA - Quality Assurance
CGI - Common Gateway Inteface

12.3.2 Bibliography

References:

1 System Analysis and Design - ELIAS M.AWAD
2-Software Engineering -ROGER.S.PRESSMAN
3-Android A Beginners Guide -by Vikram Vaswani

Websites:

1. http://www.google.co.in
2. http://www.wikipedia.org
3. http://www.seminarsonly.com
4. http://www.1000projects.org