

**IMPLEMENTATION OF RESPONSIVE AUTOMATED ONLINE
“FOOD ORDERING APPLICATION”
AN E-COMMERCE ANDROID APPLICATION**

By

**Md. Al.Amin
172-15-9748**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Mr. Md. Azizul Hakim
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Ms. Nusrat Jahan
Senior Lecturer
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

JUNE 2021

APPROVAL

This Project titled Automated Online “Food Ordering Application” An E-commerce Android Application submitted by Md. Al.Amin, ID No: 172-15-9748 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 02.06.2021.

BOARD OF EXAMINERS

Chairman



Dr. Touhid Bhuiyan
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Moushumi Zaman Bonny
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Md. Sazzadur Ahamed
Senior Lecturer

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

External Examiner



Dr. Md Arshad Ali
Associate Professor

Department of Computer Science and Engineering
Hajee Mohammad Danesh Science and Technology University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Mr. Md. Azizul Hakim, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Mr. Md. Azizul Hakim
Senior Lecturer
Department of CSE
Daffodil International University

Submitted by:



Md. Al.Amin
ID: 172-15-9748
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to Md. Azizul Hakim, Senior Lecturer, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “Android App Development” to carry out this project. His endless patience ,scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Professor Dr. Touhid Bhuiyan, Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

The aim of the project was to build a responsive online application for restaurant which helps customers to order foods online by Online food ordering management application. Online food ordering management system is an android application which used to ease the customer's life. Sometimes you don't feel like cooking or doesn't feel like to go to the restaurants; therefore we here propose an Online Food Ordering Management System which can help the customers to get food delivered immediately. This is mostly designed for a single restaurant having various food items at valuable food price. It gives effective way to order your food and almost within no time food will be delivered. Customer, he/she has login form with password in order to secure the information details and then they can select his/her favorite food items, place the order, also mention the quantity. When the order is placed, it gets stored in the database of the restaurants and then the staffs go through the orders and process it efficiently. To address this functionality, a deep research has been made on the earlier works for automating the food ordering processes. Java script framework has been used to develop the backend & frontend of the system. After developing the system, different testing methods have been used to find bugs in the system and ensure the quality.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
CHAPTER	
CHAPTER 1: INTRODUCTION	1-3
1.1. Overview of the Project	1
1.2. Aims and Objectives of the Project	2
1.3. Justification for the project	2
CHAPTER 2: LITERATURE REVIEW	4-6
2.1. Past work	4
2.2. Mobile Application Base System	5
2.3. Evaluation of the past works	6
CHAPTER 3: METHODOLOGY	7-14
3.1. Research Methodology	7
3.1.1. Initial Research	8
3.1.2. Data Gathering	8
3.1.3. Analysis	9

3.1.4. Actualization	9
3.1.5. System Critical Analysis	9
3.2. Methodology for System Development	10
3.2.1. Waterfall	10
3.2.2. Prototyping of Systems	11
3.2.3. Methodology of Agile	12
3.2.4. Selected SDLC is Reasonable	12
CHAPTER 4: DESIGN & REQUIREMENTS	15-19
4.1. Requisites Invocation	15
4.1.1. Functional Prerequisites	15
4.1.2. Non-Functional Prerequisites	16
4.2. Use Case Diagram	17
4.3. Modeling of Processes	18
4.3.1. Diagram of Data Flow (DFD)	18
CHAPTER 5: DESIGN SPECIFICATIONS	20-29
5.1. Front End Design	20
5.2. Back End Design	20
5.3. UI Design	21
5.4. Activity Layouts	21
5.4.1. Starting Page	22
5.4.2.Registration Page	23

5.4.3. Login Page	24
5.4.4. Product Details	25
5.4.5. Navigation Bar	26
5.4.6. Update Profile	27
5.5. Implementation Tools	28
5.5.1. Android Studio	28
5.5.2. Android SDK	28
5.5.3 JDK	29
CHAPTER 6: TESTING AND RESULT	30
CHAPTER 7: RECOMMENDATIONS AND CONCLUSIONS	31-32
7.1. Realization of the Project	31
7.2. System Limitations	31
7.3. Recommendations for the Future	32
7.4. Conclusions	32
APPENDIX	33-35
REFERENCES	36-37
PLAGIARISM REPORT	38

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.1. : The Waterfall model shows the steps of the research methods.	7
Figure 3.2.4. : Phases of Extreme Programming	13
Figure 4.2.1.: Use Case Diagram	17
Figure 4.3.1. : DFD of processes of customer for ordering food.	19
Figure 5.4.1. : Starting Page	22
Figure 5.4.2. : Registration Page	23
Figure 5.4.3. : Login Page	24
Figure 5.4.4. : Product Details	25
Figure 5.4.5. : Navigation bar	26
Figure 5.4.6. : Update Profile	27
Figure A-1: Starting page view with header code in android studio	33
Figure A-2: Back end using code	34
Figure A-3: Realtime data base	35

CHAPTER 1

Introduction

The project is introduced in this chapter with a high-level summary. The project's basic goals and objectives are then defined. Finally, it evaluates the project's viability and generates a device feasibility study.

1.1. Overview of the Project

The aim of the project is to create an online food ordering system. We may employ technology in a variety of ways. One of them is to build an online presence using an Android app that can also assist with online ordering. That in today's world, the use of internet-based ordering systems is on the rise. Since 2014, about 30% of customers aged 18 to 54 have requested food from a restaurant's takeout menu. An android program on their phone or tablet. Customers, on the other hand, want assurances about the nature of the foods they are purchasing when they purchase products or foods online. People are more likely to consume foods recommended by their friends or family members. This can be accomplished in an online food ordering environment by correctly incorporating and using social media. According to a study conducted by OTX Research [3], two out of every three consumers are affected in their purchasing decisions by knowledge found on social media. As a result, the demand for online food delivery is as big as it's ever been. Despite this, a sizable portion of the industry remains mostly offline.

1.2. Aims and Objectives of the Project

The project's goal is to create a food ordering system based on an Android framework. It will also assist management in managing online orders and tracking their progress. The framework allows management to install menus and take orders. Customers can log in using their mobile number, which can lead to the acquisition of a new customer by word of mouth.

The following goals must be met in order to reach the stated goal:

1. Appropriate methodologies and processes will be used to plan and implement the whole system in a structured manner.
2. The backend and frontend of the framework will be built using Java script.
3. The device would have a mobile number. As a result, customers can log in and register using their phone numbers.
4. The device will be thoroughly checked to ensure that it is of high quality.

1.3. Justification for the project

As mobile and internet technology advances, it is becoming easier to order food from restaurants online. Foodies are increasingly using online ordering systems to place orders because it is simpler and faster. Digital restaurant ordering is growing 300 percent faster than dine-in traffic, according to QSR Web [9]. However, the majority of restaurants do not devote sufficient attention to presenting a professional image on the internet. As a result, restaurants appear in 51% of all mobile searches on Google, but only 5% of them have mobile-optimized websites (Beltis, 2016)[8].

There are also several issues with the conventional food ordering scheme, which causes time waste and conflict. There are a few issues with the current system:

- ❖ Customers must first visit restaurants to learn about food products before making an order and paying. This approach necessitates time and manual labor.
- ❖ Some restaurants accept phone orders. Customer does not have a copy of the available menu items in the restaurant while making an order over the phone.
- ❖ Every restaurant requires specific personnel to take orders over the phone or in person. In today's economy, labor rates are steadily rising, making it difficult to find workers when they are needed (Patel, 2015).

To address these concerns, an Online Food Order System was developed, which was initially intended for small businesses. However, this device can be used in any restaurant. The biggest benefit of this online system is that it makes ordering food much easier for both the customer and the restaurant.

The following are the expected project benefits:

1. The ordering process will be sped up as a result of this.
2. The device would aid in the reduction of labor costs.
3. Because of the speed of execution and the number of optimal screens to meet the highest throughput, this would eliminate long lines at the counter.
4. Since it is a computer, the device would be less likely to make mistakes.
5. The most significant advantage of online ordering was the labor savings, as workers were not required to spend time on the phone or at the counter.

CHAPTER 2

Literature Review

The different approaches for automating food ordering processes are discussed in this chapter. The android application development tools and technologies used in this project are then explained.

2.1. Past Works

“An automated IT system that supervises, controls, and encourages the planning operations in restaurants” is what a computerized food ordering system is (Tan, 2013). Orders and payments were handled manually until the automated system was implemented, using register books and other methods. In the 1990s, the point-of-sale (POS) system was implemented. The system began to be used after it gained popularity. The internet and wireless technology advanced at a breakneck pace after 1990. Restaurants have also begun to implement various technology-based systems in order to improve system performance (Sullivan, 2015) [12]. Researchers have also begun to introduce various forms of restaurant solutions. There has been a lot of effort put into automating restaurant operations. Each researcher focused on a specific feature or issue and attempted to come up with a viable solution. The solutions offered by researchers can be classified into several classes. The following parts will go through those systems in detail.

2.2. Mobile Application Based System

As Android based cell phones took over built up an application for Android working framework based cell phones they're located utilized Java Script. This Android app's request handling rate is faster than a PDA-based system's request framework. The program will play a variety of media files client, eatery in addition chief likewise suggested an Android-based system framework to robotize requesting interaction of the café, from which at least one customer can be reached can arrange food, and furthermore make installment distantly. Both of these to works executed practically same functionalities. Leong (2016) additionally built up a framework for Android working framework, which can be utilized for requesting food on the web.

As of late, Saeed et al. (2016) executed an Android based shrewd café framework, which helps clients requesting food varieties and covering bills from their cell phone. Then again, the administration can screen the entire work appropriately. This Smart Food Ordering System has two interfaces for the two types of customers in restaurants: an Android mobile app for customers and a web application for restaurant employees. Clients may order food varieties online from the comfort of their own homes using these proposed and established mobile phone-based frameworks.

These frameworks have likewise a few impediments as follows:

- ❖ Customers need to introduce the café versatile application for getting the advantages of Food Ordering System.
- ❖ Ordering food varieties from PC is preposterous with these application, as versatile applications can't be perused by means of PCs. To get orders from clients utilizing PC, another web application should be created.

2.3. Evaluation of the Past Works

The past lines outline that numerous analysts worked for carrying out IT based frameworks in eateries to acquire income and lessen request handling time. Be that as it may, the majority of them didn't zero in on the online based eatery framework where clients can arrange food over web from their home. With the development of web innovation in this time, it isn't possible to stay with just advancements exhibited above as the greater part of the proposed framework simply attempted to computerize the in-house requesting framework.

The frameworks that are developed to organize food remotely, on the other hand, are gadget or functioning frameworks situated. Clients must use a phone with a specific operating system, such as Android, in order to reap the benefits of the framework.

The proposed arrangements also do not make use of the power of social media, which can be a powerful marketing tool for the café and can influence new customers to try the restaurant's food. The previous section of this paper stated that in rush hour gridlock, computerized café requests are rising 300 percent faster than feast (Beltis, 2016). As a result, eateries should embrace web innovation so that they can appear on the web and provide services to a wider range of clients for the least amount of money and time.

CHAPTER 3

Methodology

This chapter of the study aims to provide a concise picture of the project's methods and methodologies. It begins by describing the methodology and methods chosen for the study, as well as the rationale for using these methods and methodology. The article then goes on to address software development methods and provide justifications for it.

3.1. Research Methodology

Exploration is "a logical and methodical quest for appropriate data on a particular subject" (Kothari, 2004). The examination philosophy is a cycle used to gather data and information to decide. There are comprehensively two methodologies for directing an exploration. Those are: 1) Quantitative investigation 2) Qualitative Analysis. For building up the framework, joined technique has been applied. A precise methodology ought to be utilized for directing any exploration to discover the arrangement of the issue in orderly manner. For this reason, the examination techniques have been applied dependent on cascade model which shown on figure 3.1.

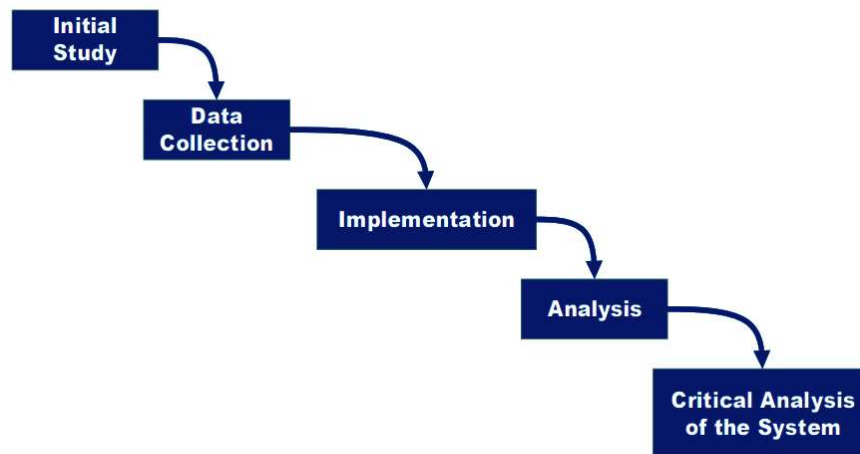


Figure 3.1 : The Waterfall model shows the steps of the research methods.

3.1.1. Initial Research

This is the first step in the analysis process. Throughout this phase, different points and branches of expertise have been examined in order to identify problems. Following the definition of the goal of establishing a café system, the investigation was limited to an online food ordering framework.

3.1.2. Data Gathering

When the goal was set, the data collection measure began gathering information about the online food ordering framework's history and previous chips in this region. The following techniques were used to gather information:

- ❖ **Literary Criticism:** Works of Literature have been explored and fundamentally examined to discover what sort of works have been proposed in past. These works aided in identifying the flaws in previous arrangements as well as defining the project's goal and objectives. Generally, essential information has been gathered to fill the need. Be that as it may, now and again, optional information is additionally utilized.

- ❖ **Observation:** Number of eatery sites have been audited to track down the latest thing in creating web application for eateries. Alongside this, regular café requesting framework has been assessed to get an understanding of online request handling framework.

3.1.3. Analysis

After information assortment, an outline on the framework has been acquired. The context is then dissected and a coherent structure is built using various Structured System Analysis and Design Methodology (SSADM) techniques (Data Flow Diagram, Entity Relationship Diagram, and so on).

3.1.4. Actualization

Following the completion of the framework's consistent construction, the execution phase began. During this stage, the coherent construction gave way to actual design via coding and framework advancement. The framework's frontend and backend have been developed and tested.

3.1.5. System Critical Analysis

After execution, the framework has been basically assessed to comprehend on the off chance that it meets the prerequisites. At that point the outcome is examined. An end and future proposal has been made in this progression.

3.2. Methodology for System Development

Any data structure should be built using a System Creation Methodology since it offers a structured approach to the development of IT-based systems. The acronym SDLC stands for System or Software Design Life Cycle. It is the bringing down of intervals of cycles in order to create a structure that is sufficient. The primary point of SDLC measure is to help give a framework that is successful, costefficient, and of superior grade. SDLC strategies regularly has the accompanying stages:

Examining (needs and plan), designing, evaluating, delivering, and supporting (reaction). The stages in different SDLC legends, however, can be updated. For different types of projects, there are a variety of product development models. In the following lines, only three of them will be discussed.

3.2.1. Waterfall

The cascade model is a well-organized method for improving programming. The entire interaction of framework advancement is isolated into particular stages. The model has been presented in 1970s. Each stage has a special yield. It was the first SDLC model that gained widespread acceptance. As a consequence, SDLC makes a reference to Waterfall at some stage. Where the system requirements are important, creativity is recognized, and the framework is a new version of an existing object, the cascade model is used (Dennis, Wixom and Roth, 2012).

In most cases, the Waterfall model has six phases. If a problem is discovered at either point of the loop, the framework returns to the previous stage. The following are the phases of the Waterfall strategy:

Gathering and Analyzing Necessities: At this point, all possible system prerequisites are identified and recorded in a prerequisite information document.

Application Design: In this step, the requirements from the previous stage are concentrated, and the framework configuration is prepared.

Execution: The system is built in a few joins together, thanks to contributions from the framework plan. The units are then put to the test.

Coordination and Testing: The program's units established in the previous stage are integrated into a system. The whole architecture is then put to the test.

Sending the System: After all of the testing is completed, the item is sent to the client.

Maintenance: There are a few problems that arise in the customer setting. Patches are sent out to resolve the problems.

3.2.2. Prototyping of Systems

Rapid Action Development (RAD) is a methodology that involves prototyping. In this process, the investigation, plan and execution stages carried out simultaneously and over and again in a cycle until the framework is finished. With this approach, the rudiments of examination and configuration are finished. At that point the work on the framework model starts right away. So that, numerous bugs and issues stayed on the framework. From that point onward, the clients or venture supports give remark on the framework. At that point, the framework is reanalysed, updated and implemented in response to criticisms This relationship continues until the clients or project sponsors are satisfied with the framework (Gould, 2016).

3.2.3. Methodology of Agile

Light-footed is a method of progress that is iterative and steady. In 2001, at a meeting in Utah, USA, the Agile procedure was born. The required programming begins with a simple plan, after which small capacities and modules are coded. For each life cycle, augmentation or sprint, work on these capacities and modules is completed in weeks. During these runs, blunders will be identified, and client feedback will be incorporated into the next augmentation plan (Ben-Zahia and Jaluta, 2014).

This theory is associated with a variety of models. Scrum, lean, and Extreme Programming are among the most well-known.

3.2.4. Selected SDLC is Reasonable

Since no SDLC is perfect, there are a variety of criteria to consider when selecting the best SDLC for constructing an IT system. Each SDLC has its own set of advantages and disadvantages. In this way, the SDLC ought to be picked painstakingly dependent on the kind of the undertaking. Dennis, Wixom and Roth (2012) examined about various models for choosing the proper advancement system for a task.

Then again, Waterfall approach is useful for creating mind boggling and dependable framework. In any case, its exhibition is helpless when the timetable is short and cutoff time is noticeable. The System Prototyping technique has a magnificent exhibition with hazy prerequisites and inside brief timeframe plan. Along these lines, it very well may be a decent decision for building this eatery framework. Be that as it may, the unwavering quality of item created with System Prototyping is poor. Nobody needs to assemble a framework that the person in question can't depend on. Agile Development performs admirably in these areas. As a result, the Agile Methodology has been chosen as the framework's product development process.

There are a variety of ways to implement Agile Methodology. Outrageous Programming (XP) has been chosen as the winner. As shown in Figure 3.2.4, the Outrageous Programming (XP) model has four phases for constructing a task.

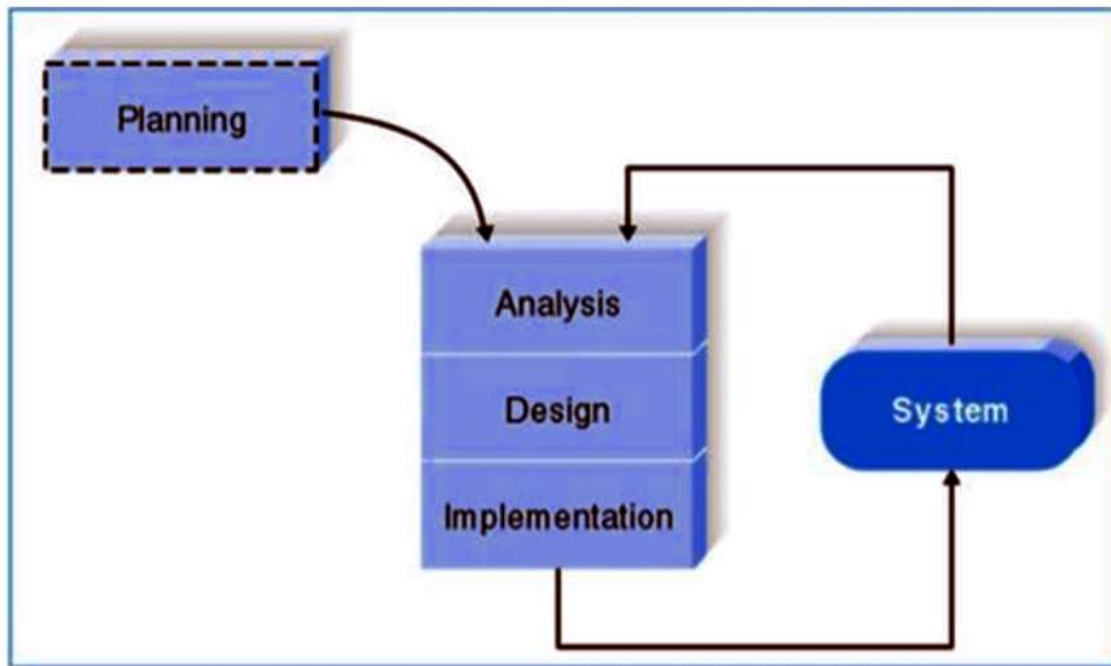


Figure 3.2.4. : Phases of Extreme Programming

The Extreme Programming method has four stages, as seen in the diagram above. These stages are covered in the following sections:

- ❖ **Organizing:** The masterminding stage is where Extreme Programming begins. The structure's requirements have been assembled and documented at this stage. The game plan, time, and cost of completing the cycles are all planned during this movement.

- ❖ **Examination:** The real model of the system has been created at this stage. SSADM is used to build the structure's cognizant plan. A granular perspective is used for analysis of the structure, since there isn't any previous scheme. There will be no planning involved in the construction of the building. Close by, the UI requirements are also investigated.

- ❖ **Plan:** The actual model of the data base and the structure's interface are set up at this point in the SDLC. In this point, the data base example's normalization is completed.

- ❖ **Execution:** The structure has been revealed through coding at this stage. Since incredible programming is an iterative method, it is possible to use a test-driven development approach with unit testing. After the framework has been installed, the structure will be surveyed using beginning to end testing (disclosure testing, customer affirmation testing). It is important to remember that the stages of analysis, planning, and execution are all iterative. Following the completion of one loop, an analysis was performed. By that time, all of the stages for the subsequent accentuation had begun.

CHAPTER 4

Design & Requirements

The methodologies used to plan the structure for the café are seen in this section. The first section addresses a variety of system requirements. At that point it examines about the framework plan and gives an outline of the frameworks measures. At that point, the information base diagram of the framework is shown. In conclusion, the UI configuration has been created.

4.1. Requisites Invocation

The prerequisites of a framework are the characteristics that a framework must have. The requirements were gathered during the SDLC planning phase. To obtain the framework's requirements, various types of information collection strategies were used.

4.1.1. Functional Prerequisites

As indicated by IIBA, utilitarian prerequisites are "the item abilities, or things that the item should accomplish for its clients". Following are the utilitarian necessities of the task:

- ❖ Client enrollment and login options should be available in the application.
- ❖ The application should include a shopping basket where users can request different types of food.
- ❖ The application should include a framework for administrator enrollment and login.
- ❖ The administrator should be able to add and change options in the application's menu.

4.1.2. Non-Functional Prerequisites

IIBA characterizes non practical prerequisites as "the quality ascribes, plan, and execution imperatives, and outside interfaces which an item should have. Following are the non-practical necessities of the venture.

- ❖ The program should have a user interface.
- ❖ The User Interface should be portable cordial.
- ❖ Special case dealing with techniques should be utilized.
- ❖ Administrator.

4.2. Use Case Diagram

To know the connection between the client and the application a utilization case chart is exceptionally fundamental. We will see how the client and the application will cooperate with one another through the utilization case graph. The utilization case outline of our venture appeared in figure 4.2.1.

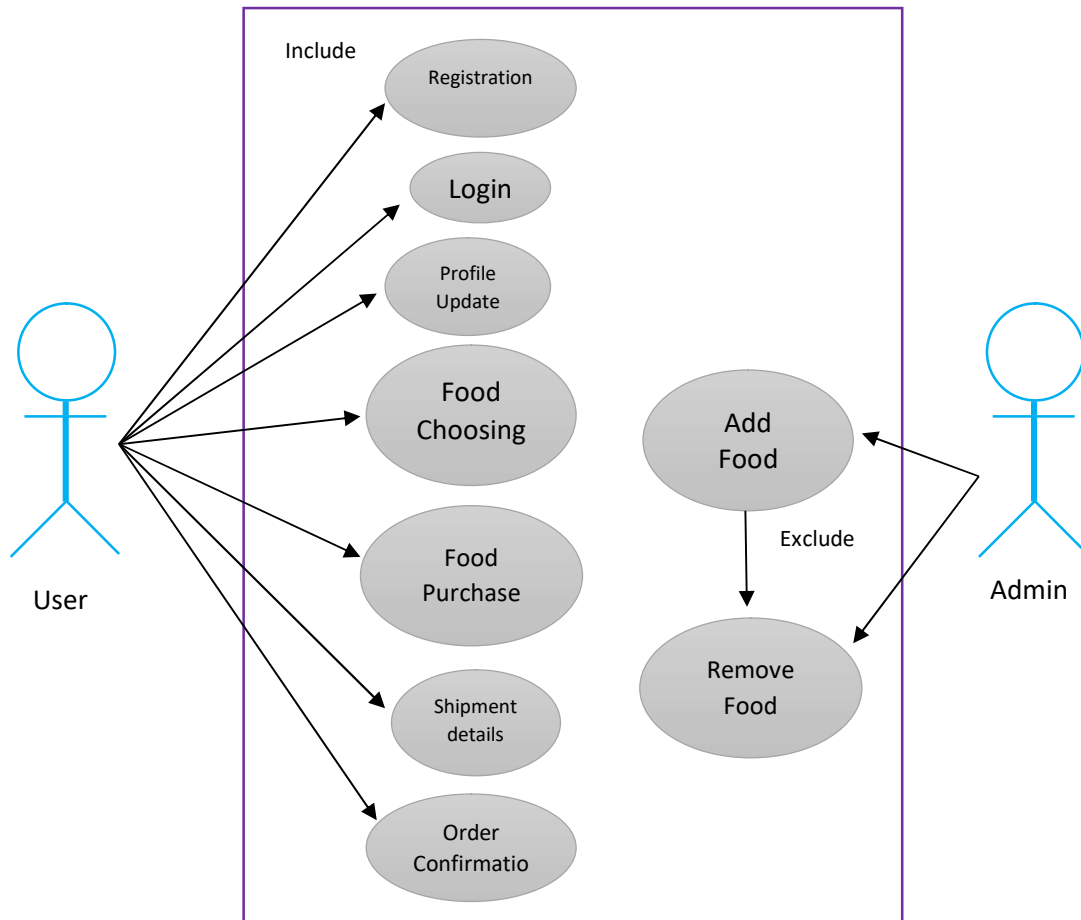


Figure 4.2.1.: Use Case Diagram

For use this application user has to open an account by registration step. Then user can log in the application by log in button. After that user can update his/her profile by uploading his/her image and also fill up other details. Now by the section of product details user can chose food for purchase the food. At the last have to fill up the shipment details for confirmation then it will show order its confirm.

4.3. Modeling of Processes

In a project, process modeling is used to represent the data processing processes in an application. The Model-View Controller design pattern will be used to create and execute the food ordering program. The majority of these processes are implemented in application controllers as business logic. The system's processes will be modeled using a data flow diagram.

4.3.1. Diagram of Data Flow (DFD)

This is a data flow diagram (DFD) that depicts the method of placing an online food order. An object or consumer is represented by the oval shape. Phase boxes are the boxes in the diagram's centre. The number of processes and their descriptions are shown in the process box. Data is stored in the boxes on the right side of the diagram. When a piece of data is stored, the box represents it. The flow of data is depicted by the space between the boxes and the oval shape. A DFD's main purpose is to depict the flow of data while performing an operation, as shown in figure 4.3.1.

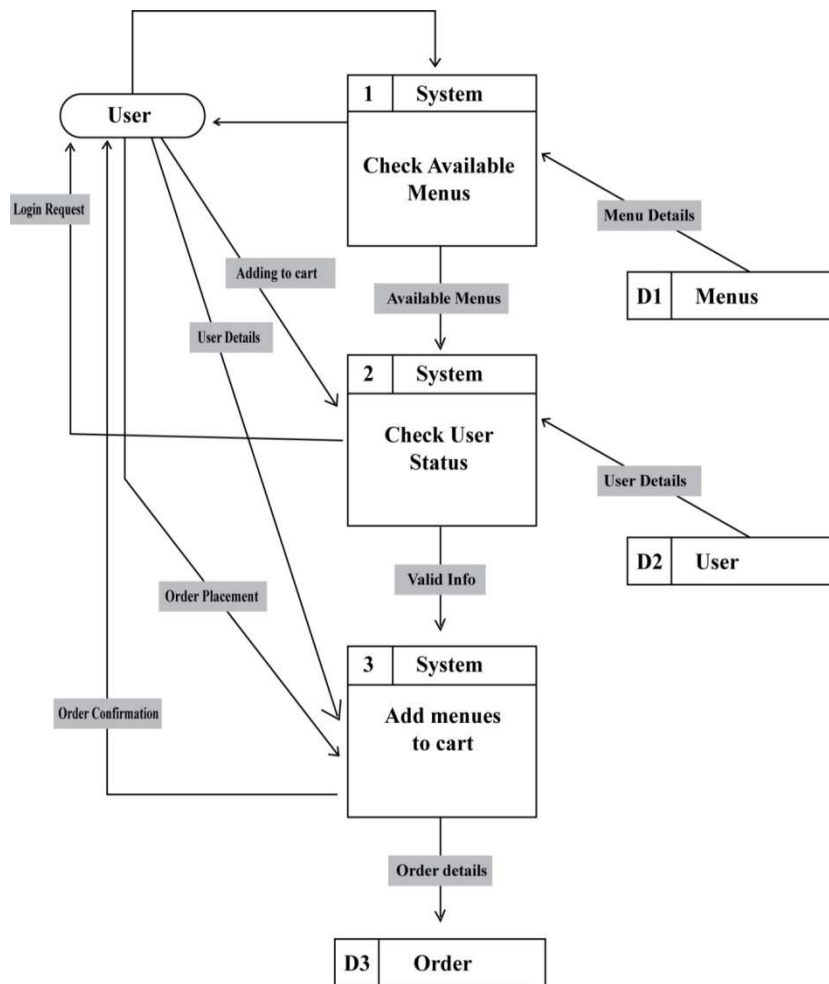


Figure 4.3.1 : DFD of processes of customer for ordering food.

When a visitor requests the menu page, Machine 1 performs a query in the database's menu table, retrieves available menus, and returns the menu list to the visitor. If a visitor wants to add a menu to their cart, they click the "Add to Cart" button. The second device checks to see if the user is logged in. The framework prompts the user to log in if he or she is not already logged in. The machine compares the user's credential to the user account after receiving it. If the user credentials in system 3 match those saved in the database's User table, the system adds the menu to Cart table 35 and redirects to the cart list. If the user orders the menu in Process 4, the order information are saved in the Order table.

CHAPTER 5

Design Specifications

5.1 Front End Design

A framework's graphical structure is implied by front-end plan. At the point when client opens an application or any product whatever he/she sees as the interface is fundamentally the front-end plan. It has the most significance in light of the fact that the clients can be pulled in by this. That is the reason front-end plan of any application or programming ought to be alluring and delightful. With the goal that it can make individuals intrigued toward the application. We buckled down toward the front plan to make our application so easy to use just as alluring. Since, in such a case that the front-end plan which in other word we can say the UI is mind boggling then the client cannot utilize the application appropriately and might wind up losing interest on our application. We needed to work with Java language to do the front-end planning of our venture.

5.2 Back End Design

The conduct of a framework or an application is controlled by the back-end plan which is otherwise called worker side plan. It controls each activity of the framework so it is the primary piece of a framework. If there should be an occurrence of our task it was the most basic part. The security of the framework, structure and different substance are relied upon the back-end plan. For the back end planning of our undertaking we have utilized Java language.

5.3 UI Design

At the point when the front-end configuration addresses graphically it is known as the (UI). At the point when a client opens an application, he/she sees the (UI). What's more, box the UI the client needs to utilize that application. We zeroed in on our application's UI plan when we concluded the prerequisites. We attempted our level best to make the UI generally basic and easy to use.

5.4 Activity Layouts

- ❖ Starting Page
- ❖ Registration Page
- ❖ Login Page
- ❖ Product Details
- ❖ Navigation bar
- ❖ Update Profile

5.4.1. Starting Page

The accompanying figure is the main movement. At the point when another client opens the application, this movement will show up. From here the client should go to the login or enlistment action.

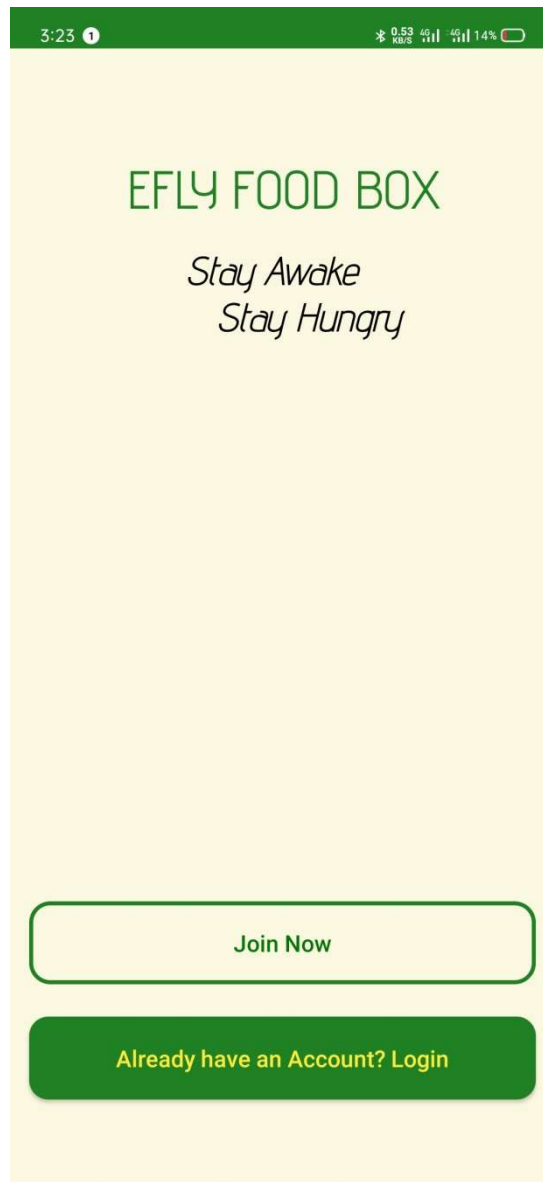


Figure 5.4.1. : Starting Page

5.4.2. Registration Page

The accompanying figure is the registration Page. User have to registration here for enlistment action.

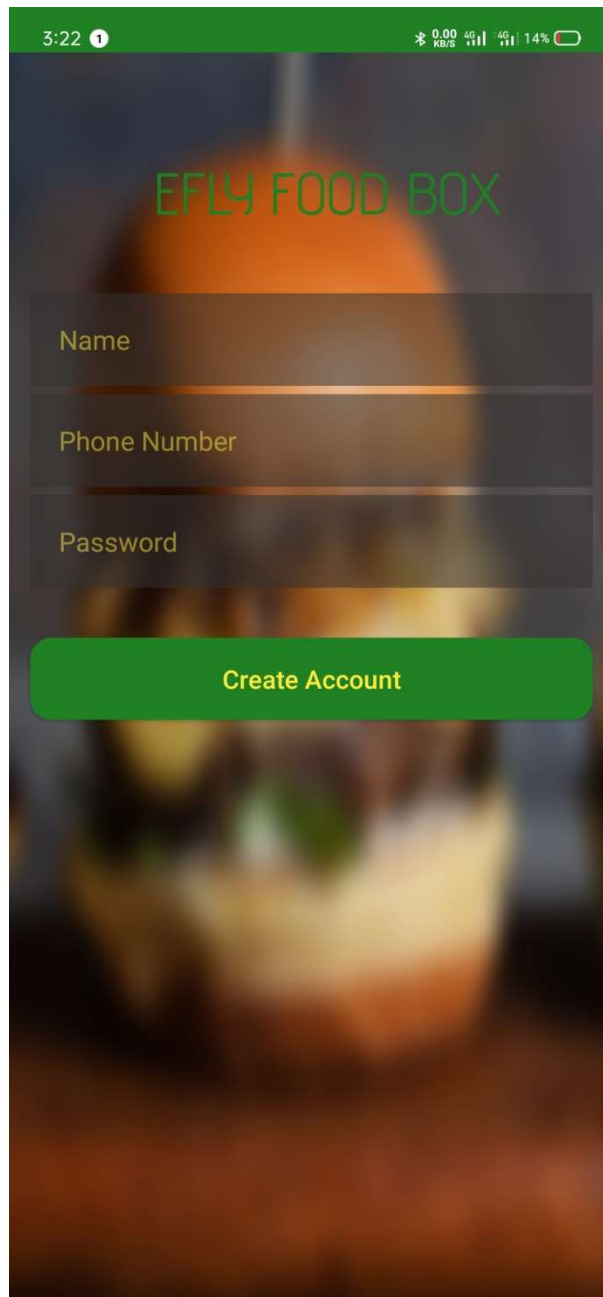


Figure 5.4.2. : Registration Page

5.4.3. Login Page

The accompanying figure is the login Page. User can login from here.

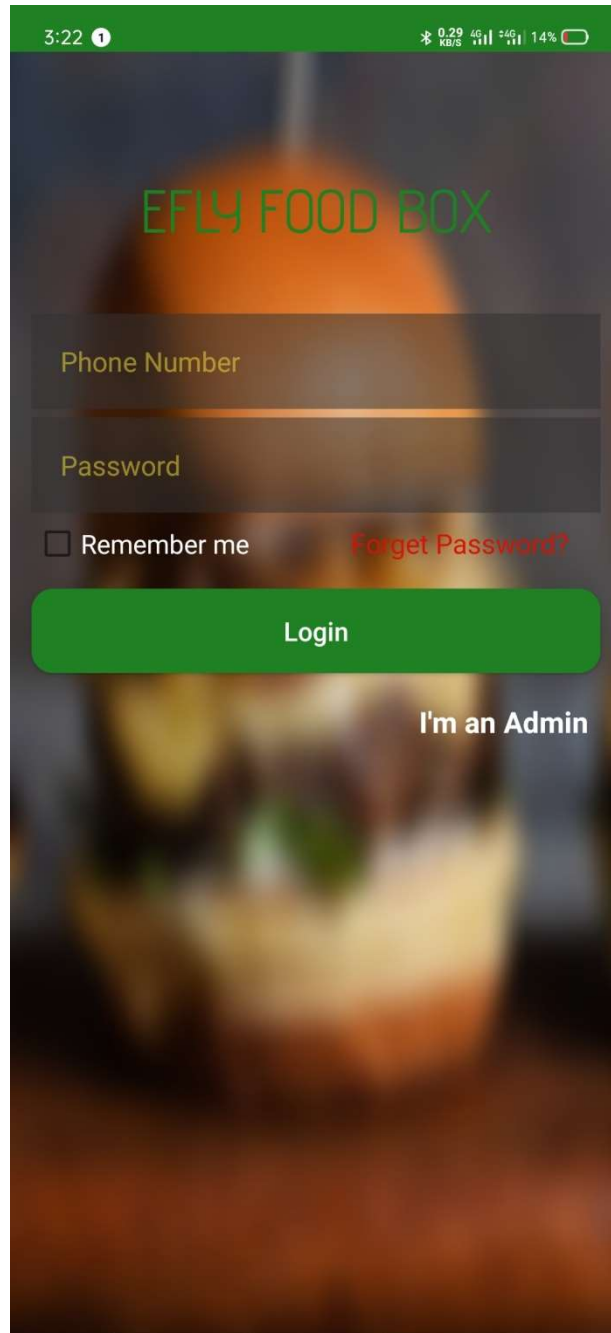


Figure 5.4.3. : Login Page

5.4.4. Product Details

The accompanying figure is the product details Page. User can choose his/her requirement product from here.

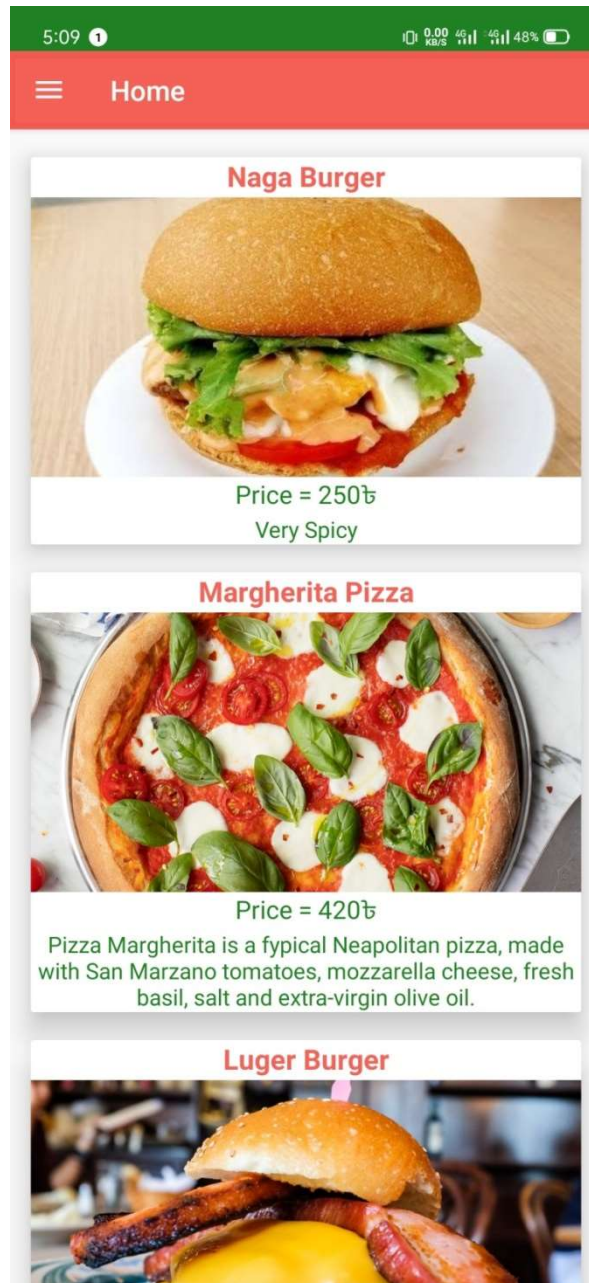


Figure 5.4.4. : Product Details

5.4.5. Navigation bar

The accompanying figure is the Navigation Bar.

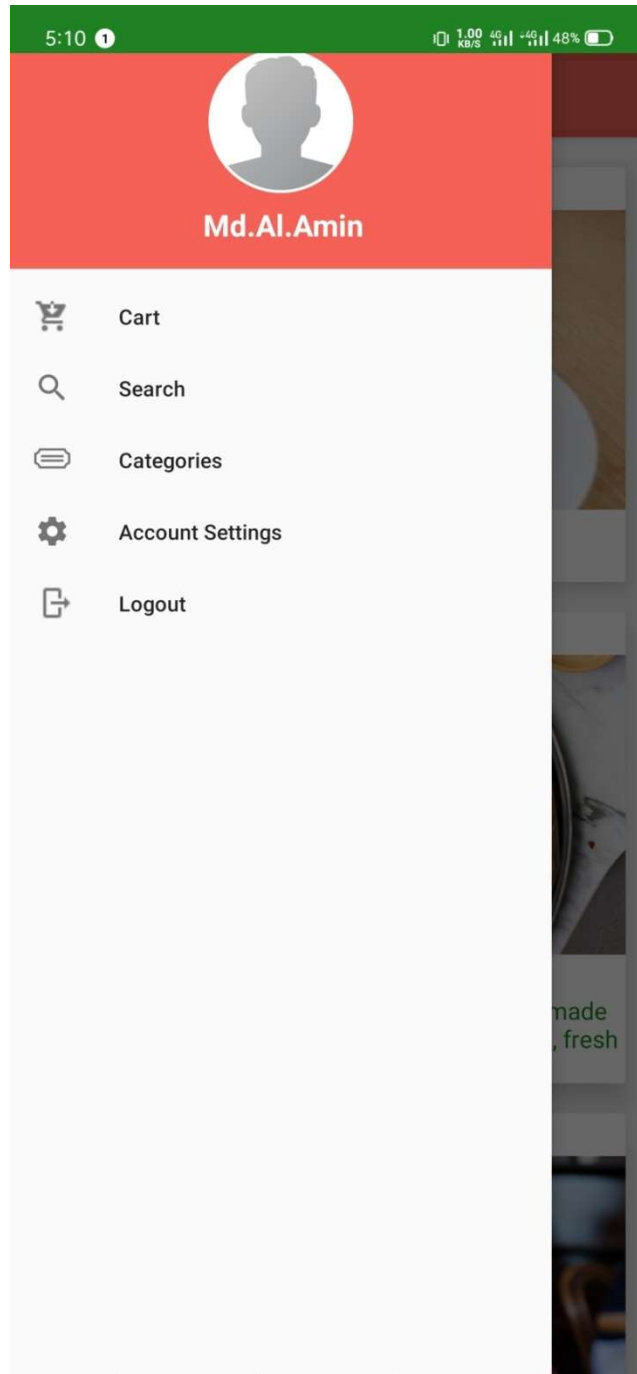


Figure 5.4.5. : Navigation bar

5.4.6. Update Profile

The accompanying figure is update profile page. Here user can upload picture and also update his/her information.

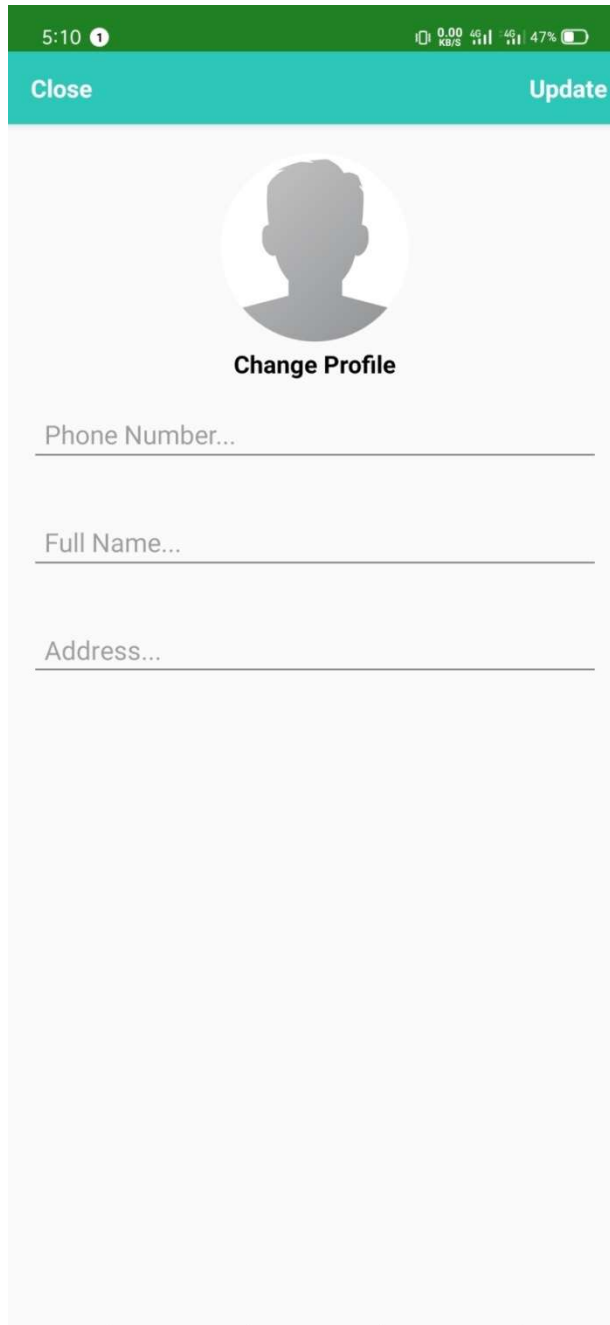


Figure 5.4.6. : Update Profile

5.5. Implementation Tools

To chip away at some venture the main things that are must are the execution devices. In our undertaking we required a great deal of them. Here we have depicted about the absolute most significant execution devices of our task.

5.5.1. Android Studio

Android studio is the authority improvement device for Google's android working system[6]. Without android studio we were unable to have constructed our undertaking. In this way, it was the main execution apparatus for our task.

5.5.2. Android SDK

The Android SDK is a software development kit for creating Android apps. It comes with a collection of tools that developers would need to create Android apps. The following are the tools provided by the Android SDK:

- Libraries that are needed
- An emulator
- A debugger
- Sample source code
- Android OS tutorials
- Relevant documentation for the Android application program interfaces (APIs)

5.5.3 JDK

JDK implies java improvement pack which is a climate for programming advancement. To create java applications, engineers need to utilize JDK. It incorporates the accompanying segments:

- Runtime Environment for Java (JRE)
- Compiler
- Java interpreter/loader (javac)
- Documentation generator • Archiver (jar) (javadoc)

CHAPTER 6

Testing and Result

Testing can be partitioned into two sorts extensively: practical testing and primary testing. Primary testing, otherwise called white-box testing, includes analyzing the inner execution. It tests the plan utilized by the execution to check it rightness. Interestingly, Functional testing, in some cases alluded to discovery testing, will be trying on the usefulness of the framework dependent on the predetermined prerequisite. The actual test has little information about the testing objective's given below. The given details about the experiments for discovery testing.

Test-1

T. N. : Check adding to cart functionality for logged in user.
Work : (1) Go to any menu details page, (2) Click on add to menu button
Result : The user is redirected to cart page where the subtotal and total price shown.
Main Result : As expected
Status : Pass

Test-2

T. N. : Add new menu into database
Work : (1) Fill up the add menu form, (2) Press add button
Result : A confirmation message will be shown. New menu added to database. The image of the menu uploaded to images folder with menu name.
Main Result : As expected
Status : Pass

CHAPTER 7

Recommendations and Conclusions

This chapter brings the project's study to a close. This chapter begins with a discussion of the project's accomplishments. It then goes on to explain the system's limitations. After that, it suggests and recommends some device enhancements. Finally, the chapter comes to a close with some closing remarks.

7.1. Realization of the Project

The following methods were used to achieve these goals with the prototype software:

- ❖ Objective #1 was met by analyzing previous works for automating the food ordering process. In addition, the technologies for developing Android applications are briefly discussed.
- ❖ The prototype program was used to achieve these goals using the following methods: The first goal was accomplished by reviewing previous work on automating the food ordering process. Furthermore, the technologies used to create Android apps are briefly discussed.

7.2. System Limitations

There are some drawbacks to the method as well. The system's shopping cart only has basic features and does not allow for advanced cart customization. Validation features are also available. Validating data using client-side languages like JavaScript or HTML 5 will help to overcome this limitation. In addition, the order model has been developed. The controllers and functions for moving data into the order table, on the other hand, have not been published. As a result, the instructions that have been placed cannot be viewed.

7.3. Recommendations for the Future

Aside from the unfinished specifications, there are other ways to improve the project. The user acceptance research respondents also offered several suggestions for change. Improvements may include:

- ❖ Payment mechanism that is safe and accepts a variety of payment types.
- ❖ Using a graphical floor plan to handle tables and reservations.
- ❖ Support for food order delivery monitoring has been added.
- ❖ Converting the system to a progressive web application. Advanced inventory management with material storage and expiry details. Managing customer loyalty membership and discount vouchers.

7.4. Conclusions

The aim of this project is to create an Android application for food ordering. A systematic methodology has been used in the development of the program. The framework was created using Android Studio's Extreme Programming process. The Java script system was used to build the program.

A working complex prototype of an online food ordering framework was successfully introduced as part of the project. The prototype program has been thoroughly tested to demonstrate the system's quality and efficiency. All related research information and decision-making processes were also recorded in this study. In conclusion, the project met its goals and served its intent. I'm hoping that the application would be able to fulfill the majority of the criteria for online food ordering.

APPENDIX

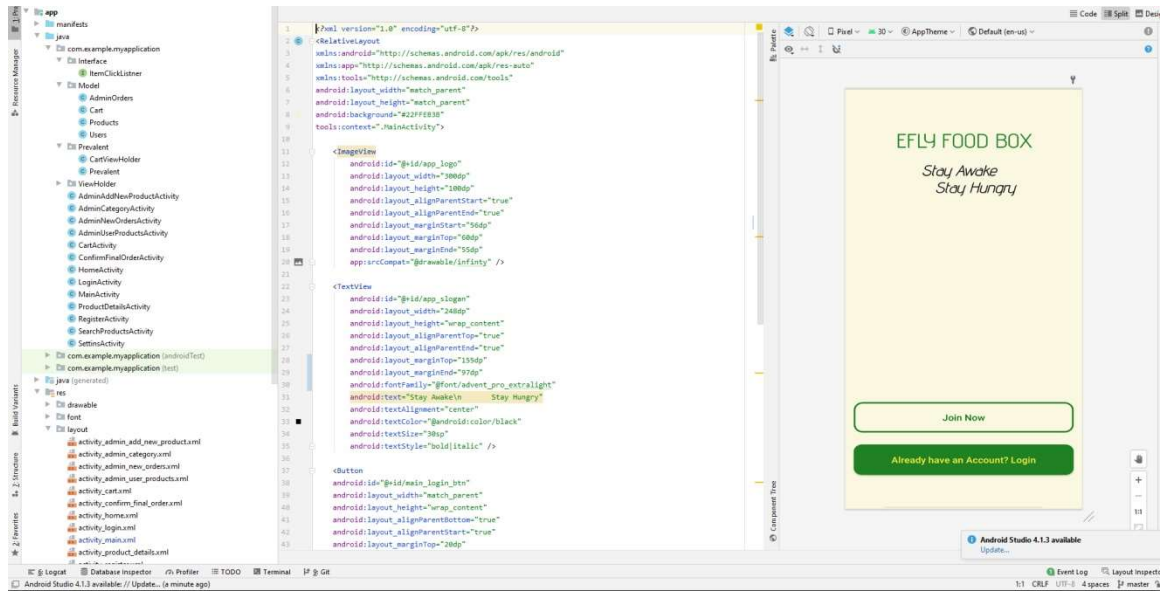


Figure A-1: Starting page view with header code in android studio

```

1 package com.example.myapplication;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6
7 import androidx.appcompat.widget.Toolbar;
8
9 import android.os.Bundle;
10
11 import androidx.appcompat.app.AppCompatActivity;
12
13 import android.os.Bundle;
14
15 import androidx.appcompat.app.AppCompatActivity;
16
17 import android.os.Bundle;
18
19 import androidx.appcompat.app.AppCompatActivity;
20
21 import android.os.Bundle;
22
23 import androidx.appcompat.app.AppCompatActivity;
24
25 public class LoginActivity extends AppCompatActivity {
26     private EditText InputPhoneNumber, InputPassword;
27     private Button loginButton;
28     private ProgressDialog loadingBar;
29     private TextView AdminLink, NotAdminLink;
30     private String parentDBName = "Users";
31     private CheckBox checkBoxRememberMe;
32     @Override
33     protected void onCreate(Bundle savedInstanceState) {
34         super.onCreate(savedInstanceState);
35         setContentView(R.layout.activity_login);
36         loginButton = (Button) findViewById(R.id.login_btn);
37         InputPassword = (EditText) findViewById(R.id.login_password_input);
38         InputPhoneNumber = (EditText) findViewById(R.id.login_phone_number_input);
39         AdminLink = (TextView) findViewById(R.id.admin_panel_link);
40         NotAdminLink = (TextView) findViewById(R.id.not_admin_panel_link);
41         loadingBar = new ProgressDialog(this);
42         checkBoxRememberMe = (CheckBox) findViewById(R.id.remember_me_chk);
43         Paper.init(this);
44         loginButton.setOnClickListener(v -> { loginUser(); });
45
46         AdminLink.setOnClickListener(v -> {
47             loginButton.setText("Login Admin");
48             AdminLink.setVisibility(View.INVISIBLE);
49             NotAdminLink.setVisibility(View.VISIBLE);
50             parentDBName = "Admins";
51         });
52         NotAdminLink.setOnClickListener(v -> {
53             loginButton.setText("Login");
54             AdminLink.setVisibility(View.VISIBLE);
55             NotAdminLink.setVisibility(View.INVISIBLE);
56             parentDBName = "Users";
57         });
58     }
59
60     private void loginUser() {
61         String phone = InputPhoneNumber.getText().toString();
62         String password = InputPassword.getText().toString();
63         if (TextUtils.isEmpty(phone))
64     }

```

Figure A-2: Back end using code

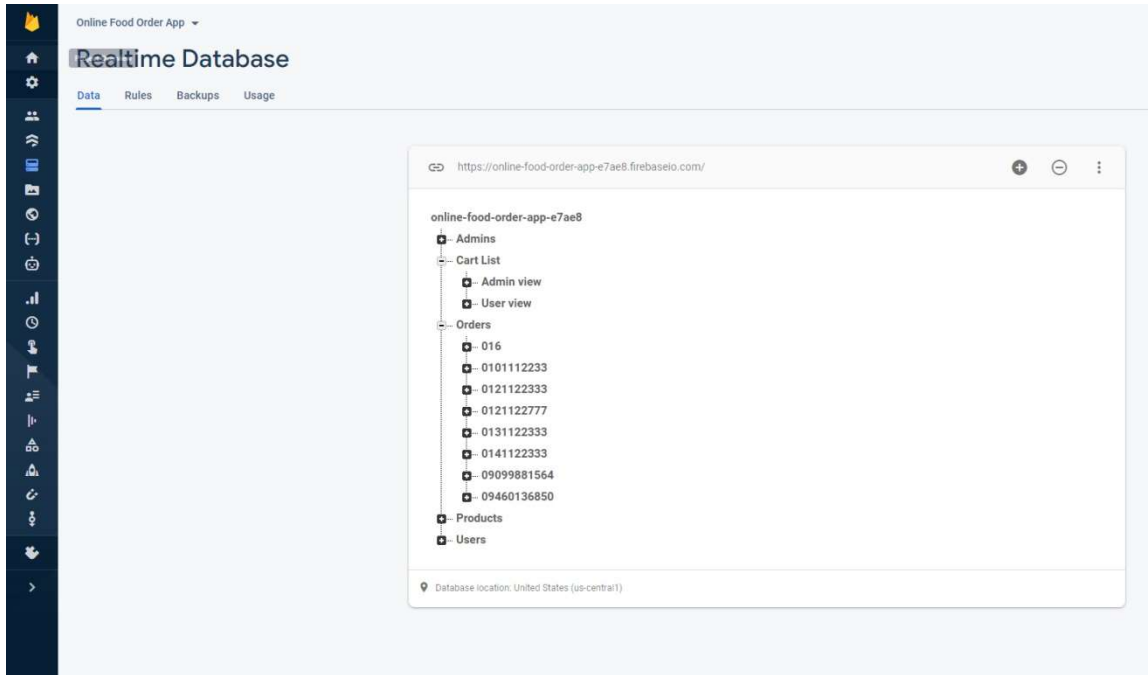


Figure A-3: Realtime data base

REFERENCE

- [1] Alagoz, S. M. and Hekimoglu, H. (2012) 'A Study on Tam: Analysis of Customer Attitudes in Online Food Ordering System', *Procedia - Social and Behavioral Sciences*, 62, pp. 1138–1143. doi: 10.1016/j.sbspro.2012.09.195.
- [2] Bean, M. (2015) *Laravel 5 Essentials*. Packt Publishing Ltd.
- [3] Beltis, A. J. (2016) 9 Advantages of an Online Food Ordering System. Available at: <https://pos.toasttab.com/blog/online-food-ordering-system> (Accessed: 7 July 2017).
- [4] Ben-Zahia, M. A. and Jaluta, I. (2014) 'Criteria for selecting software development models', in: 2014 Global Summit on Computer Information Technology (GSCIT), pp. 1–6. doi: 10.1109/GSCIT.2014.6970099.
- [5] Bhaumik, S. (2015) *Bootstrap Essentials*. Birmingham, UK: Packt Publishing.
- [6] Bounnady, K. et al. (2016) 'Comparison the processing speed between PHP and ASP.NET', in: 2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp. 1–5. doi: 10.1109/ECTICon.2016.7561484.
- [7] Cheong, S. N., Chiew, W. W. and Yap, W. J. (2010) 'Design and development of Multi-touchable E-restaurant Management System', in: 2010 International Conference on Science and Social Research (CSSR 2010), IEEE, pp. 680–685. doi: 10.1109/CSSR.2010.5773867.
- [8] Das, R. and Saikia, L. P. (2016) 'Comparison of Procedural PHP with Codeigniter and Laravel Framework', *International Journal of Current Trends in Engineering & Research*, 2(6).
- [9] Dawson, C. W. (2009) *Projects in Computing and Information Systems: A Student's Guide*. Addison-Wesley.
- [10] Dennis, A., Wixom, B. H. and Roth, R. M. (2012) *Systems Analysis and Design*. 5th Edition. John wiley & sons.
- [11] Dhore, V. B. et al. (2014) 'Digital Table Booking and Food Ordering System Using Android Application', *International Journal of Emerging Engineering Research and Technology*, 2(7), pp. 76–81.
- [12] Dockins, K. (2016) *Design Patterns in PHP and Laravel*. Apress.
- [13] AndroidStudio tutorial on firebase application, available at: <https://www.youtube.com/watch?v=zKBGjGoeid0&t=880s> [Last accessed: September 28, 2019].

- [17] Android Studio tutorial on simple login application, available at:<https://www.youtube.com/watch?v=tTbd1Mfi-Sk>[Last accessed: October 22, 2019].
- [18] Database and storage, available at:<https://console.firebase.google.com/project/help/database/help/data> [Last accessed: October 29, 2019].
- [19] Google APIs and Services, available at:<https://console.developers.google.com/projectselector2/apis/dashboard?pli=1&supportedpurview=project&project&folder&organizationId>[Last accessed: October24, 2019].
- [20] Google Cloud Platform, available at:<https://console.cloud.google.com/getting-started?pli=1>[Last accessed: October25, 2019].
- [21] Detail information about android studio, available at:https://en.wikipedia.org/wiki/Android_Studio [Last accessed: October 26, 2019].
- [22] Definition and details about android SDK, available at:<https://www.techopedia.com/definition/4220/android-sdk>[Last accessed: October 26, 2019].
- [23] Definition and details about JDK, available at:<https://www.techopedia.com/definition/5594/java-development-kit-jdk>[Last accessed: October 26, 2019].

PLAGIARISM REPORT

5/1/2021

Turnitin

Document Viewer

Turnitin Originality Report

Processed on: 01-May-2021 21:15 +06
ID: 1575309243
Word Count: 5157
Submitted: 1

IMPLEMENTATION OF
RESPONSIVE AUTOMATED
ONLINE... By Md. Al. Amin

Similarity Index	Similarity by Source	
13%	Internet Sources:	N/A
	Publications:	N/A
	Student Papers:	13%

[exclude quoted](#) [exclude bibliography](#) [exclude small matches](#) mode:

quickview (classic) report [print](#) [refresh](#) [download](#)

6% match (student papers from 04-Feb-2018)

[Submitted to University of Derby on 2018-02-04](#)



3% match (student papers from 03-Apr-2018)

[Submitted to Daffodil International University on 2018-04-03](#)



1% match (student papers from 31-Jan-2018)

[Submitted to University of Derby on 2018-01-31](#)



1% match (student papers from 27-Nov-2020)

[Submitted to University of Hertfordshire on 2020-11-27](#)



1% match (student papers from 05-Feb-2018)

[Submitted to Buckinghamshire Chilterns University College on 2018-02-05](#)



<1% match (student papers from 19-Feb-2021)

[Submitted to Asia Pacific University College of Technology and Innovation \(UCTI\) on 2021-02-19](#)



<1% match (student papers from 25-Aug-2017)

[Submitted to St. Patrick's College on 2017-08-25](#)



<1% match (student papers from 11-Mar-2021)

[Submitted to Universiti Tunku Abdul Rahman on 2021-03-11](#)



<1% match (student papers from 06-Jan-2021)

[Submitted to Uganda Christian University on 2021-01-06](#)



https://www.turnitin.com/newreport_classic.asp?lang=en_us&oid=1575309243&ft=1&bypass_cv=1