Abstract

This project is an online bakery shop that allows users to search for and purchase various bakery items available at the online shop. The project offers a variety of bakery products that are displayed online in various categories. These items can be viewed by the user. If the user wishes to purchase a product or products, he or she may do so by adding them to his or her shopping cart. An online bakery shop software project, while retaining the features of an e-commerce site, acts as a central database containing various bakery products. It allows customers to shop online from the comfort of their own homes.

A customer can sign up for free, log in to his or her account, browse items of interest, view prices and other details of selected items, place items with preferred weights into the shopping cart, and choose from payment options. Following that, the user can check out. The user can choose any payment method, such as credit/debit card or cash on delivery. This mobile application project is built with XML as the front end and JAVA as the back end. The FIREBASE database will store all information about the users and various bakery items, as well as their respective categories, and will send notifications or updates on their orders.

The items, along with the ordered weights and the total amount to be paid, are presented to the customer as a ready order at checkout. More information will be required to complete the transaction at that time. Typically, the customer will be asked to fill out a very short and simple form containing information about the shipping address and other details, as well as payment information such as cash on delivery, etc. As soon as the order is placed, an email will be sent to the customer. After a successful transaction, the user receives a copy of the shopping receipt via e-mail.

Chapter 1

Introduction

1.1 About the Project

The purpose of Cool Cakes Bakery is to provide a platform for users to check and purchase various bakery products available online. User can add the selected items it to his shopping cart.User needs to register on the site before checking out.He can then login using same id password next time. Now he may pay through a Stripe.In addition, the admin can add new items in the existing list of cakes or appetisers by selecting the category where he wants to add new items. The desserts are organised into categories based on birthday cakes and anniversary cakes, cupcakes and the snacks can be categorised into appetisers ,pizza and burger The customer is identified through a unique login authentication by passwords. They can add desired items to their carts or wish lists, and are directed to the payment gateway after their order is placed. Our back-end technology is reliable and efficient enough to not create any hurdles from the time the user logs into the website to the time when he/she has found the information needed. We hope that the users have a seamless experience.

1.1.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.

Android Studio supports all the same programming languages of IntelliJ (and CLion) e.g. Java, C++, and more with extensions, such as Go; and Android Studio 3.0 or later supports Kotlin and "all Java 7 language features and a subset of Java 8 language features that vary by platform version."

External projects backport some Java 9 features. While IntelliJ states that Android Studio supports all released Java versions, and Java 12, it's not clear to what level Android Studio supports Java versions up to Java 12 (the documentation mentions partial Java 8 support). At least some new language features up to Java 12 are usable in Android.

1.1.2 Android SDK

The Android SDK is a software development kit that includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows 7 or later. As of March 2015, the SDK is not available on Android itself, but software development is possible by using specialized Android applications.

Until around the end of 2014, the officially-supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) Plugin.As of 2015, Android Studio, is the official IDE; however, developers are free to use others, but Google made it clear that ADT was officially deprecated since the end of 2015 to focus on Android Studio as the official Android IDE. Additionally, developers may use any text editor to edit Java and XML files, then use command line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android devices (e.g., triggering a reboot, installing software package(s) remotely).

1.1.3 Emulator

The Android Emulator simulates Android devices on your computer so that you can test your application on a variety of devices and Android API levels without needing to have each physical device.

The emulator provides almost all of the capabilities of a real Android device. You can simulate incoming phone calls and text messages, specify the location of the device, simulate different network speeds, simulate rotation and other hardware sensors, access the Google Play Store, and much more.

Testing your app on the emulator is in some ways faster and easier than doing so on a physical device. For example, you can transfer data faster to the emulator than to a device connected over USB.

The emulator comes with predefined configurations for various Android phone, tablet, Wear OS, and Android TV devices.

1.1.4 Firebase

Firebase is a product of Google which helps developers to build, manage, and grow their apps easily. It helps developers to build their apps faster and in a more secure way. No programming is required on the firebase side which makes it easy to use its features more efficiently. It provides services to android, ios, web, and unity. It provides cloud storage. It uses NoSQL for the database for the storage of data. In this architecture, Firebase sits between the server and clients. Your servers can connect to Firebase and interact with the data just like any other client would. In other words, your server communicates with clients by manipulating data in Firebase. Our Security and Firebase Rules language lets you assign full access to your data to your server. Your server code can then listen for any changes to data made by clients, and respond appropriately. In this configuration, even though you're still running a server, Firebase is handling all of the heavy lifting of scale and real-time updates. Firebase initially was an online chat service provider to various websites through API and ran with the name Envolve. It got popular as developers used it to exchange application data like a game state in real time across their users more than the chats. This resulted in the separation of the Envolve architecture and it's chat system. The Envolve architecture was further evolved by it's founders James Tamplin and Andrew Lee,to what modern day Firebase is in the year 2012. In this architecture, Firebase sits between the server and clients. Your servers can connect to Firebase and interact with the data just like any other client would. In other words, your server communicates with clients by manipulating data in Firebase. Our Security and Firebase Rules language lets you assign full access to your data to your server. Your server code can then listen for any changes to data made by clients, and respond appropriately. In this configuration, even though you're still running a server, Firebase is handling all of the heavy lifting of scale and real-time updates.

1.1.5 Java

Java is a programming language independent of all platforms and can be used for multiple operating systems.Keeping security in mind, all other programming languages are developed, including the interpreter, compiler, and run-time environment. A lot of concentration is put on testing to ensure potential early errors are caught.Java is the first choice of android app developers because of ease of use, robustness, security features, and cross-platform development capabilities.

1.1.6 XML

Extensible Markup Language (XML) is a markup language and file format for storing, transmitting, and reconstructing arbitrary data. It defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The World Wide Web Consortium's XML 1.0 Specification of 1998 and several other related specifications—all of them free open standards—define XML.

The design goals of XML emphasize simplicity, generality, and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services.

Several schema systems exist to aid in the definition of XML-based languages, while programmers have developed many application programming interfaces (APIs) to aid the processing of XML data.

1.2 Existing System

Since we are advancing in the use of technology there are many bakeries converting to this convenient method where in they can virtually receive orders and the hassle for the customers also decreases since all the need is their phone and the app to order rather than physically going over to the bakery if not to several bakeries because there are chances of anyone to decline the order.

1.2.1 Limitation of Existing System

The Bakery App could be improved by adding some more products in the order section. "Order"can be improved by allowing user to maintain multiple carts. Further changes can be easily done by changing the code. The front end can be made more attractive by using attractive layout. Due to the practical challenges, instant logistics is facing many difficulties, while developing rapidly.This app providers face the complex problem as to control the cost of scheduling riders, while maintaining high quality of customers' service. That is, takeout service providers struggle to efficiently assign orders to riders for instant deliveries. The issue is an inherent contradiction between the undulated distribution of order time, space locations, and available riders' stability. We would further like to improve the app by providing more options for editing user profile.

Some of the major limitations that persons with bakery app are:

- To make customer service interactions so that the administrators aware of the problems faced by the customer.

- To list the products if they are out of stock.
- To add an option to modify or update the products after the order is confirmed.

1.3 Problem Statement

As all face the inconvenience of visiting the bakeries every single time to buy or place order due to hectic life. It would be very helpful to have a cool cake bakery app which allows one to place order online anywhere anytime . This also simplifies the work for any bakery as they will have a expanded market as well.

Chapter 2

Requirement Analysis

2.1 Hardware Requirements

The hardware requirements are very minimal and the program can be run on most of the machines

Processor : Qualcomm Snapdragon processor

Processor Speed : 1.4 GHz

RAM : 2 GB

Storage Space : 10 GB

Display Resolution : 1024*768

I/O Elements : Camera, Speaker, Microphone, GPS

Network : 5 Mbps

2.2 Software Requirement

Operating System : Android / iOS

Chapter 3

System Design

3.1 System Architecture

The project consists of the following parts as shown in figure 3.1

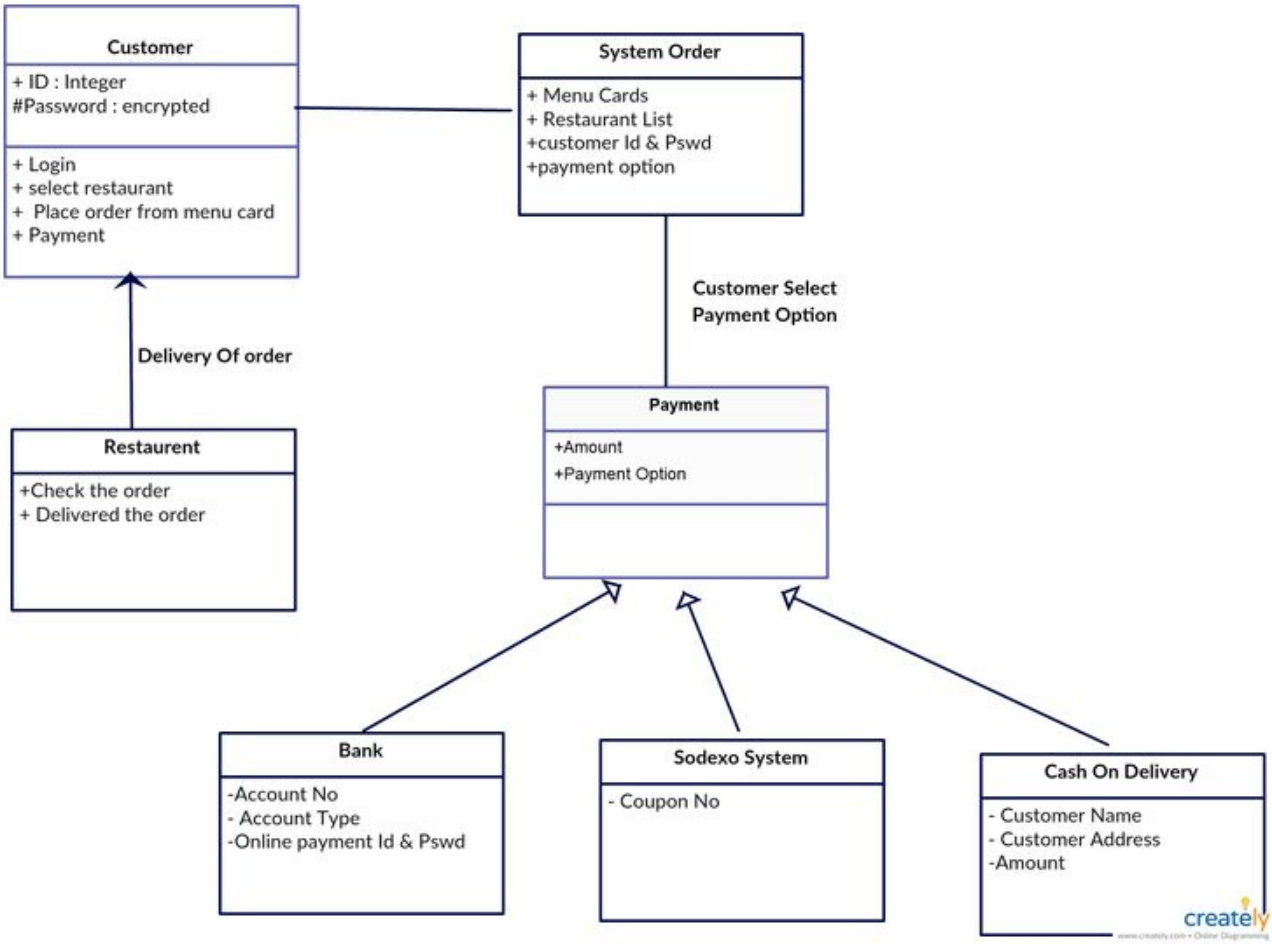

Figure 3.1: Flowchart of the application

The above figure (Figure 3.1) represents the system design of the Cool Cakes Bakery. The movement of the bakery application is shown and it also shows how the customer logs in and what the customer can access such as order and the payments. The admin logs in and can add the products,he can also cancel the orders if the product is not available and he can accept the same and send a notification (firebase is used for notifications) to the customer that their order is in process. The desserts are organised into categories based on birthday cakes and anniversary cakes, cupcakes and the snacks can be categorised into appetisers ,pizza and burger.They can add desired items to their carts or wish lists, and are directed to the payment gateway after their order is placed.

Chapter 4

Implementation

4.1 Creating a User Login

```
public User(String name, String id, String phoneNumber, String emailId)
{
Name = name;
Id = id;
PhoneNumber = phoneNumber;
EmailId = emailId;
}
```

4.2 Displays the cart activity

```
public Cart(String date, String desc_enter, String devprice, String flavours
String img_url, String pid, String pname, String price, String quantity,
String time, String uid, String wordings)
{
this.date = date;
this.desc_enter = desc_enter;
this.devprice = devprice;
this.flavours = flavours;
this.img_url = img_url;
this.pid = pid;
this.pname = pname;
```

```
this.price = price;
this.quantity = quantity;
this.time = time;
this.uid = uid;
this.wordings = wordings;
}
```

4.3 Displays the confirmed order.

```
public void onClick(View v)
{
Toast.makeText(ConfirmOrderActivity.this,"Deliver Only Within
Karnataka Bangalore",Toast.LENGTH_LONG).show();
   }
```

4.4 Display the date of the order.

```
public Dialog onCreateDialog(@Nullable Bundle savedInstanceState)
{
return new DatePickerDialog(getActivity(),
(DatePickerDialog.OnDateSetListener) getActivity(),
year,month,day);
   }
```

4.5 Displays the history of the orders by the user.

```
public HistoryCustomerAdapter(ArrayList<History> list, Context context)
{
this.list=list;
this.context=context;
}
```

4.6 Displays the order activity:

```
public void onClick(View v)
{
Intent intent= new Intent(ShowOrderActivity.this,ShowHistoryActivity.class)
startActivity(intent);
}
```

4.7 Displays all the orders of the day

```
public void onDataChange(@NonNull DataSnapshot snapshot)
{
for(DataSnapshot snapshot1:snapshot.getChildren())
{
productsconfirm productsconfirm=snapshot1.getValue(productsconfirm.class);
list.add(productsconfirm);
} myadapter.notifyDataSetChanged();
}
```

Chapter 5

Result Analysis

5.1 Testing

Table 5.1 gives details of validation.

Table 5.1: Test Case Validation

Test Case No.	**Input**	**Expected Output**	**Actual Output**
1	User enters email address and password	User gets verified and app outputs screen to order ice cream	User gets access to the app only if email id and password matches with the one stored in system
2	User gives an invalid email	The app outputs login failed	The app outputs invalid command and redirects user to login page
3	User gives a valid command	The app performs desired functionality	The app performs desired function as per user's request
4	User places the order	The selected products get added to the cart	The selected products get added to the cart
5	User receives confirmation notification	The app performs the desired functionality	User receives notification

Chapter 6

Conclusion

The purpose of Bakery App is to provide a platform for users to check and purchase various bakery products available online. User can add the selected items it to his shopping cart.User needs to register on the site before checking out.He can then login using same id password next time. Now he may pay through a Stripe.In addition, the admin can add new items in the existing list of cakes or appetisers by selecting the category where he wants to add new items. The desserts are organised into categories based on birthday cakes and anniversary cakes, cupcakes and the snacks can be categorised into appetisers ,pizza and burger The customer is identified through a unique login authentication by passwords. They can add desired items to their carts or wish lists, and are directed to the payment gateway after their order is placed. This mobile application project is built with XML as the front end and JAVA as the back end. The FIREBASE database will store all information about the users and various bakery items, as well as their respective categories, and will send notifications or updates on their orders.

The items, along with the ordered weights and the total amount to be paid, are presented to the customer as a ready order at checkout. More information will be required to complete the transaction at that time. Typically, the customer will be asked to fill out a very short and simple form containing information about the shipping address and other details, as well as payment information such as cash on delivery, etc. As soon as the order is placed, an email will be sent to the customer. After a successful transaction, the user receives a copy of the shopping receipt via e-mail. Our back-end technology is reliable and efficient enough to not create any hurdles from the time the user logs into the website to the time when he/she has found the information needed. We hope that the users have a seamless experience.

References

[1] Abrar Ahmed Chhipa, et al., *Adaptive Neuro-fuzzy Inference System Based Maximum Power Tracking Controller for Variable Speed WECS*, 2021 *Energies*, Vol. 14, No. 19, pp.6275. https://doi.org/10.3390/en14196275

[2] Abrar Ahmed Chhipa, et al., *MPPT optimisation techniques and power electronics for renewable energy systems: wind and solar energy systems*, 2022 *Int. J. Swarm Intelligence (IJSI)*, Vol. 7, No. 2. https://doi.org/10.1504/IJSI.2021.10041290

[3] Abrar Ahmed Chhipa and Vinod Kumar, *DC-Microgrid Voltage Regulation using Dual Active Bridge based SVR,2021 IEEE 7th International Conference on Electrical Energy Systems (ICEES)*, 2021, pp. 490-495, doi: 10.1109/ICEES51510.2021.9383696.

[4] Abrar Ahmed Chhipa and Vinod Kumar, *Grid-Connected PV System Power Forecasting Using Nonlinear Autoregressive Exogenous Model*, *The 2nd Electric Power and Renewable Energy Conference (EPREC-2021)*, 28-30 May, 2021. (In Process)

[5] @online Raspberry pi, `https://www.raspberrypi.org/` Online; accessed 10-June-2019

[6] HU, Yun Chao, et al., *Mobile edge computing?A key technology towards 5G*, ETSI white paper, 2015, vol. 11, no 11, p. 1-16.