****

Online Bakery Management System

**SUBMITTED BY: SUBMITTED TO:**

VINEETA SHARMA MR. AMIT SRIVASTAVA SIR

718028

HARSHITA PANDEY

718029

BCA –III

**Introduction and Objectives**

1. **Statement of the Proposed System**

Our proposed system is an online bakery system that enables the ease for the customers. It overcomes the disadvantages of the traditional queuing system. Our proposed system is a medium to order online cakes hassle free from bakery. This system improves the method of taking the order from customer. This online bakery system sets up a menu online and customers can easily place the order as per the wish. Also with menu, customers can easily track the orders. This system also provides a feedback system in which user can rate the items. Also, the proposed system can recommend bakery, cakes based on the ratings given by the user. The payment can be made online or pay-on-delivery system. For more secured ordering, separate accounts are maintained for each user by providing them an ID and a password.

**2.Problem with the Existing System**

* **Selection**

Having so many options to choose from often overwhelm the customers, and many times results in a dropped sale. Having sorting system is really important.

* **Ordering Process**

Convenience is one reason why more customers than ever before prefer ordering online these days. But there are still plenty of websites which have been created without giving any thought to[**optimizing the user experience**](https://restolabs.com/blog/restaurant-technology/top-uiux-must-haves-online-ordering-websites-restaurants). Sometimes it can even be difficult to find the ‘Menu’ button on a site! At times there’s an information overload and at others too little. Most customers also consider websites that require customers to register on the site before placing an order troublesome.

* **Payment**

A recurring problem that customers face when ordering cakes online is regarding payment. Some apps do not accept multiple payment methods, forcing customers to drop their orders.

Security is another factor that people consider when purchasing anything online and if your payment gateway doesn’t assure it, they’ll have no reason to proceed further with their order.

* **Customer Services**

Whether it’s regarding delays in delivery, dissatisfaction with the food queries about payments and refunds or any general questions and complaints, customers will want to get in touch with you and want to feel like they’re being heard.

* **Delivery & Packaging**

When it comes to issues in delivery, they not only include delays, but also the quality and quantity of the food, packaging and unpleasant behavior by delivery persons.

In your online menu, feel free to throw in some information on preparation and delivery time, as well as portion size so that the customer knows what to expect.

Lastly, having a live tracking option for customers allows them to check where exactly their order is.  Adding a tracker to track the movement is a good way to keep the customers updated about their order status. You can use push notifications or text messages to send updates.

**3.Objective of the Project**

The main objective of online bakery system is to develop an application which gives provision to the bakery owners to flourish their business by uploading menus at no cost and will invariably lead o higher customer retention and acquisition rates.

The next important objective of is to make the process of ordering quick, easy and convenient.

The System is User Friendly so that the person using it will not face any difficulties in operating it.

The System has facilities that allow users to view menu card, select the items as required, add them to cart, view special discounts on food items, select a time for food delivery, rate the items and services provided.

Data Redundancy is avoided.

Less item consuming hence increase the efficiency of the system.

**4.Project Description**

This project Online Bakery System has been developed on Android Studio and Firebase. The main objective of this project is to sell bakery stuff online. This project intends different types of forms with different variety provides to user to buy online. In this users can give order from place and pay cash on delivery. This project provides a lot of features to manage In very many manner.

**Methodology Used**

**Technologies Used:**

**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 was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0.

On May 7, 2019, Kotlin replaced Java as Google's preferred language for Android app development. Java is still supported, as is C++.

**Java** is a class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let application developers *write once, run anywhere* (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages.

**Extensible Markup Language** (**XML**) is a markup language that 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.

**Firebase** is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development.

**Software and Hardware Requirement**

**Software Requirement**

**Server**

Database Firebase

Operating System Windows 8 or later

**Developer**

Database Firebase

Operating System Windows 10

Implementation Tool Android Studio

**Hardware Requirement**

**Client**

Processor Octa Core or later

RAM 4.00 GB

HDD 64 GB

Display 720x1440

Operating System Android 7 or later

**Developer**

Processor 2.30GHz Intel® Core™ i3 7th Gen

RAM 4.00GB

HDD 1 TB

Display 1366x768

**Data Flow Diagram (DFD)**

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFD’s can also be used for the visualization of data processing (structured design). On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process. A DFD provides no information about the timing of processes, or about whether processes will operate in sequence or in parallel. It is therefore quite different from a flowchart, which shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored. It is common practice to draw a context-level data flow diagram first, which shows the interaction between the system and external agents which act as data sources and data sinks. On the context diagram (also known as the Level 0 DFD) the system's interactions with the outside world are modeled purely in terms of data flows across the system boundary. The context diagram shows the entire system as a single process, and gives no clues as to its internal organization.

**Fig: 0-Level DFD:-**

**Fig: 1-Level DFD:-**

**E R Diagram:-**