



E-Commerce Application For Local Stores

Marya Anwar, Nuzhat Saba, Isha Admane, Saloni Khandelwal,

Vipul Tembhurne, Jyoti C. Kolte (Asst Prof, P.C.E. Nagpur)

Abstract— The persistence of E-commerce Mobile Application for Local Store is to mechanise the current manual Application by the help of computerized tools and developed Mobile Application, fulfilling their requirements, so that their appreciated data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily presented and easy to work with.

Online E-commerce Application, as described above, can lead to error free, secure, reliable and fast management Application. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help Stores in better utilization of resources. The Stores can maintain computerized records without redundant entries.

Keywords- Android Application, E-commerce, Java.

I. Introduction

The "Online E-commerce Application" has been developed to override the problems usual in the practicing manual Application. This software is supported to eliminate and in some cases reduce the hardships faced by this existing Application. Moreover this Application is designed for the particular need of the company to carry out operations in a smooth and effective manner.

II. Literature Survey

Every Local Store, whether big or small, has challenges to overcome and managing the information of Item Category, Men, woman, Child, Address, Orders. Every Online Ecommerce Application has different Food needs, therefore we design Exclusive employee management Applications that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your Shop is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our Application come with remote access features, which will allow you to manage your labour force anytime. These Application will ultimately allow you to better manage resources

III. Objectives

The main objective of the Project on Online E-commerce Application is to manage the details of Clothes, Item Category, hopping art, Customer, Order. It manages all the information about Clothes or Other type of goods, Delivery Address, Order. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an Application program to reduce the manual work for managing the Clothes, Item Category, Delivery Address, and Shopping Cart. It tracks all the details about the Shopping Cart, Customer Information als

IV. Tools and Technology

1. Android Studio

Android Studio is the official integrated_development environment (IDE) for Google Android Operating_system, built on JetBrains' IntelliJ_IDEA software and designed specifically for Android Development. Android Studio was announced on May 16, 2013 at the Google. It supports application development within the Android operating system, Android Studio uses a Gradle-based build system, emulator, code templates, and Github integration. Every project in Android Studio has one or more modalities with source code and resource files. These modalities include Android app modules, Library modules, and Google App Engine modules which is very helpful for us.

It also helps our team to Building and designing android applications working in coordination with the design and the testing team to develop better products with fewer bugs To work with different API's and data sources Fix bugs Continuous evaluation, discovering new technologies and exploiting them

2. Dart

Google had its first ever release of Flutter 1.0 last December, after having it in beta mode for over 18 months. Dart is the programming language used to code Flutter apps. Dart is another product by Google and released version 2.1, before Flutter, in November. As it is starting out, the Flutter community is not as extensive as React Native, Ionic, or Xamarin. A while back, I discovered a liking for JavaScript. I was ecstatic to be working on a React Native mobile app

for my internship. I enjoy coding hybrid mobile apps too, so wanted to give Flutter a try, as I had done Xamarin sometime last year. At my first glance of Flutter (and Dart), I felt be fuddled and couldn't seem to understand anything. They even had a section on their docs for developers moving from React Native. So, I took to digging deeper on all things Dart. Dart looks a bit like C and is an object-oriented programming language. So, if you prefer the C languages or Java, Dart is the one for you, and you'll likely be proficient in it. Dart is not only used for mobile app development but is a programming language it's used to build just about anything on the web, servers, desktop and of course, mobile applications. Dart, when used in web applications, is transpiled to JavaScript so it runs on all web browsers. The Dart installation comes with a VM as well to run the .dart files from a command-line interface. The Dart files used in Flutter apps are compiled and packaged into a binary file (.apk or .ipa) and uploaded to app stores.

3. Flutter

Flutter is a free and open-source mobile UI framework created by Google and released in May 2017. In a few words, it allows you to create a native mobile application with only one codebase. This means that you can use one programming language and one codebase to create two different apps (for iOS and Android)

To develop with Flutter, you will use a programming language called Dart. The language was created by Google in October 2011, but it has improved a lot over these past years.

Flutter is a cross-platform UI toolkit that is designed to allow code reuse across operating systems such as iOS and Android, while also allowing applications to interface directly with underlying platform services. The goal is to enable developers to deliver high-performance apps that feel natural on different platforms, embracing differences where they exist while sharing as much code as possible.

4. Javascript

JavaScript is a scripting or programming language that allows you to implement complex features on web pages every time a web page does more than just sit there and display static information for you to look at displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc. — you can bet that JavaScript is probably involved. It is the third layer of the layer cake of standard web technologies, two of which (HTML and CSS) we have covered in much more detail in other parts of the Learning Area.

5. Node JS

As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications. In the following "hello world" example, many connections can be handled concurrently. Upon each connection, the callback is fired, but if there is no work to be done, Node.js will sleep. This is in contrast to today's more common concurrency model, in which OS threads are employed. Thread-based networking is relatively inefficient and very difficult to use.

Furthermore, users of Node.js are free from worries of dead-locking the process, since there are no locks. Almost no function in Node.js directly performs I/O, so the process never blocks. Because nothing blocks, scalable systems are very reasonable to develop in Node.js.

Node.js is similar in design to, and influenced by, systems like Ruby's Event Machine and Python's Twisted. Node.js takes the event model a bit further. It presents an event loop as a runtime construct instead of as a library. In other systems, there is always a blocking call to start the event-loop. Typically, behavior is defined through callbacks at the beginning of a script, and at the end a server is started through a blocking call like Event Machine::run(). In Node.js, there is no such start-the-event-loop call. Node.js simply enters the event loop after executing the input script. Node.js exits the event loop when there are no more callbacks to perform. This behavior is like browser JavaScript — the event loop is hidden from the user.

6. Mysql

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.

7. MySql-workbench

MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more. MySQL Workbench is available on Windows, Linux and Mac OS X.

MySQL Workbench enables a DBA, developer, or data architect to visually design, model, generate, and manage databases. It includes everything a data modeler needs for creating complex ER models, forward and reverse engineering, and also delivers key features for performing difficult change management and documentation tasks that normally require much time and effort.

V. Activity Diagram

1. Data Flow Diagram

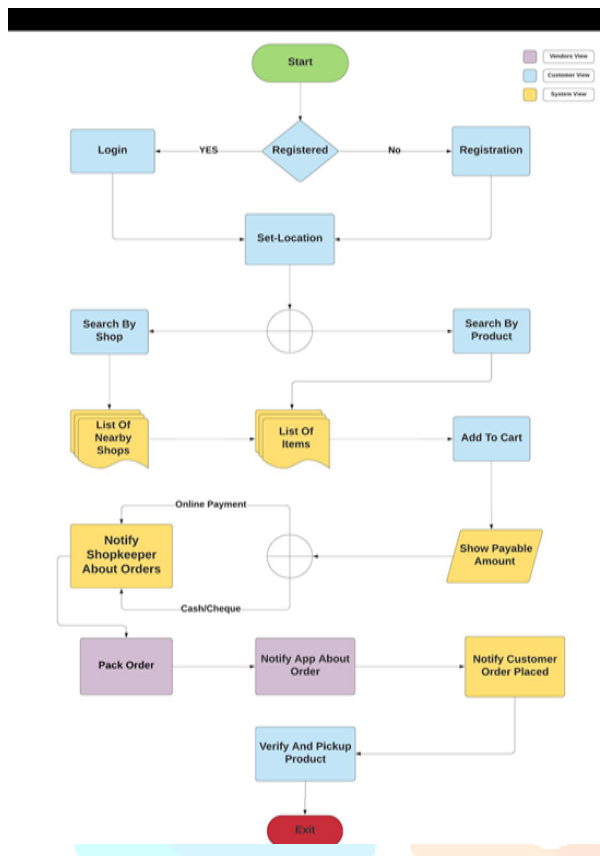


Fig - 1

2. Registration Page

A signup page (also known as a registration page) enables users and organizations to independently register and gain access to your system. It is common to have multiple signup pages depending on the types of people and organizations you want to register. In this article you will learn about the different types of signup pages

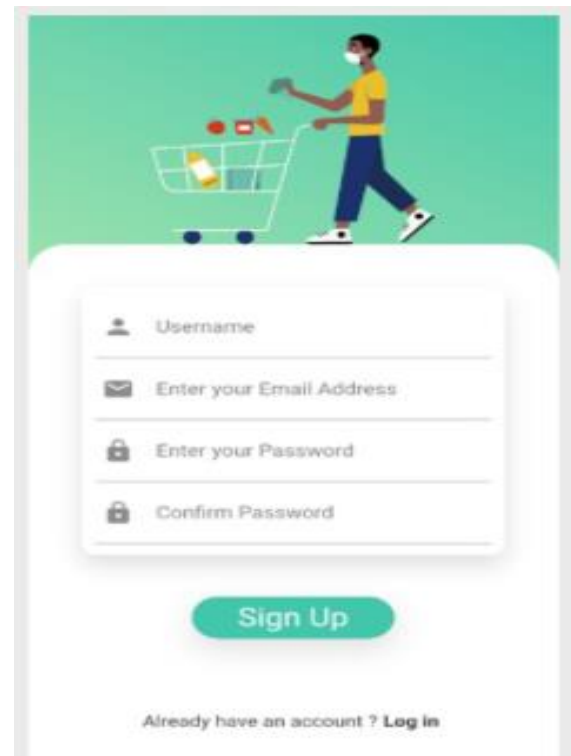


Fig -2

- i. First user will give his user-email and password then this will be saved in redis using key as {email, password → OTP} for 24-hrs
- ii. user will then send otp along with {email, password} so then if otp matches new row will be created in user table with rest of the data as empty string (also before filling user details user will have to set his/her fingerprint and this will be mapped with email and password and stored in flutter local storage)
- iii. then the user will send his profile-detail and current-location and then an update query will be executed to update it to the database.
- iv. then if all this works then will send back token/expiry and this will be stored in flutter local storage the next home page will be called.

3. Login Page

first we will fetch saved {email,password}-(from ingerpint/saved)if exist and get userdetails before going to homepage. in the backend side, first we will check if email exist if does check password by converting it to hash and then match with password in db then all this works then will send back token/expiry and this will be stored in flutter local storage the next home page will be called.

Authentication

First check which page to open(login,home,setuserdetails). If no token -- open login page. If token not expire but no pincode set user details -- means user has validated himself but didnt fill his user detail and did not set his location too If token and not expire -- open home page

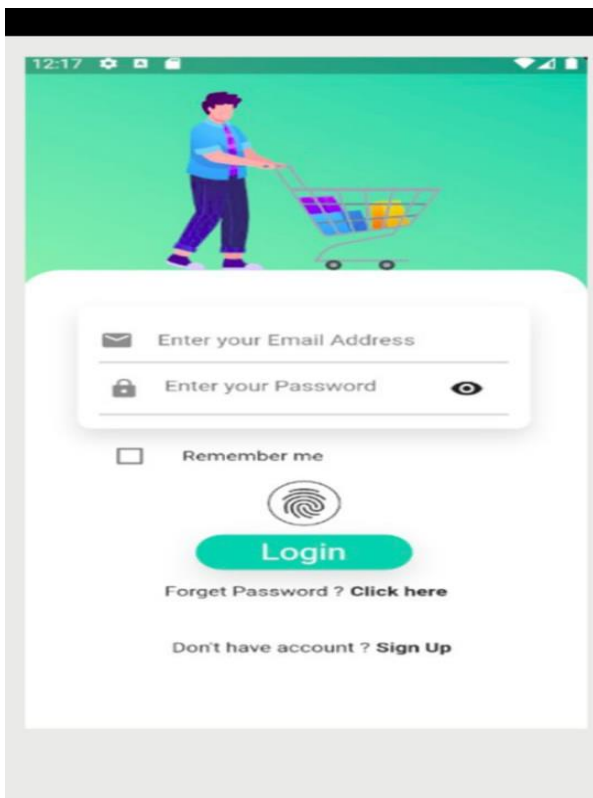


Fig - 3

4. Order Page

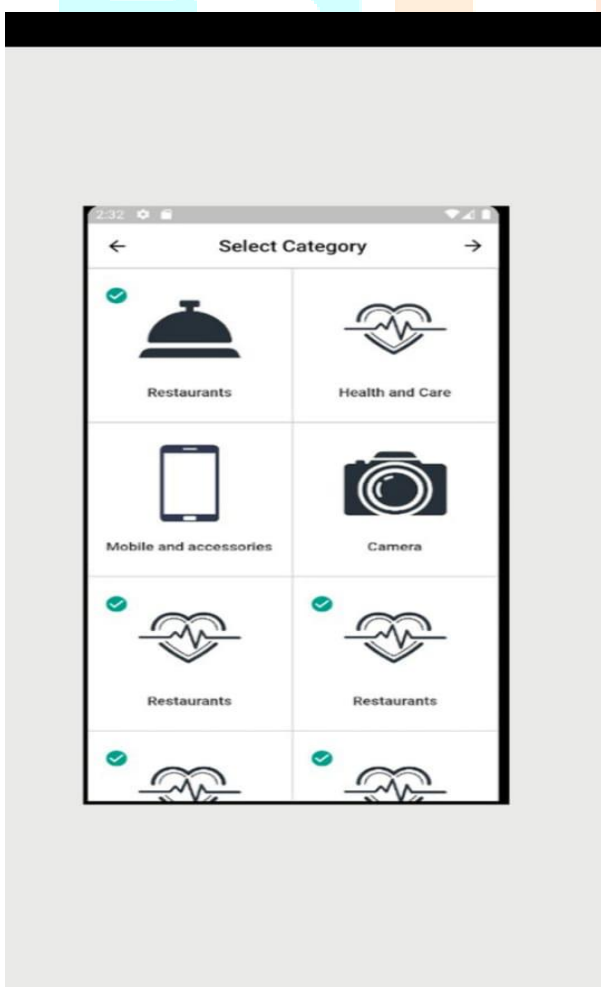


Fig -4

1. firstly we will add the product to the cart and add that product to the cart. On adding to the cart we will add this product to flutter secure storage so that we will know that this product is already added to the cart so we won't add the product again. After that a request will go to

backend to add the product to the cart. On the backend part we will first see if any unplaced order is already present for the current user. If any order is present we will add the current product to that unplaced order. otherwise we will first create an unplaced order and then we will add this product to that unplaced order.

2. on the cart page firstly we will fetch all the unplaced products of the user from the server. On the server side we will first see if any unplaced order is present for the current user. If present we will then fetch all the order-products by that orderid. Again on the frontend after getting these unplaced products we will fetch the quantity of the order (if available) from the storage. [If a user changes the quantity of the product then we will initially change this quantity on the flutter_secure_storage. It is just to make the least no. of requests]. If a user wants to remove the product from the cart a request will go to the backend. On the backend side we will first get the unplaced order by orderid and then remove the particular product.

3. On clicking checkout a request will be sent to the backend. On the backend part we will first get the unplaced order by user id. After getting the order-id we will first change the quantity of every product and after that we will categories these products according to their shops and create new orders as per different shops. After placing these orders we will delete the unplaced order.

4. On the store side first the store will get the order status as pending. After clicking on that order the order will automatically be accepted. After that store has two options either to cancel the order or to click on ready to pick up. Here if the store clicks on ready to pick up an otp will be sent to the user. The user then when picking up the order will show the otp to the shopkeeper and they will confirm the otp by clicking on confirm otp

5. If present we will then fetch all the order products by that orderid. Again on the frontend after getting these unplaced products we will fetch the quantity of the order (if available) from the storage. [If a user changes the quantity of the product then we will initially change this quantity on the flutter_secure_storage. It is just to make the least no. of requests]. If a user wants to remove the product from the cart a request will go to the backend.

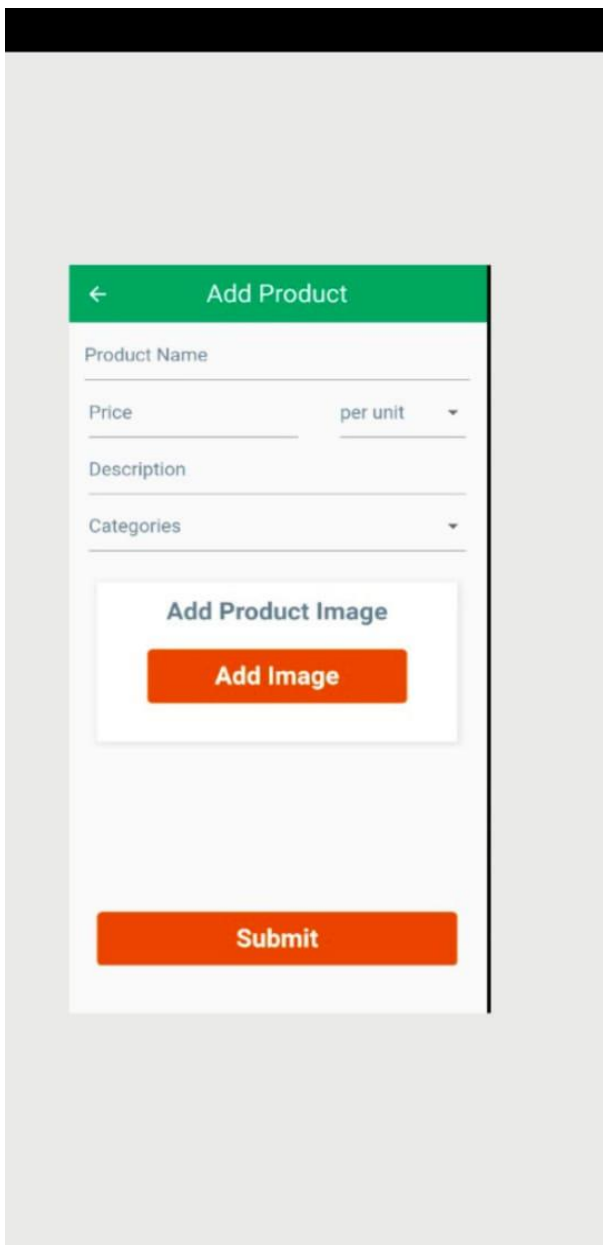


Fig-5

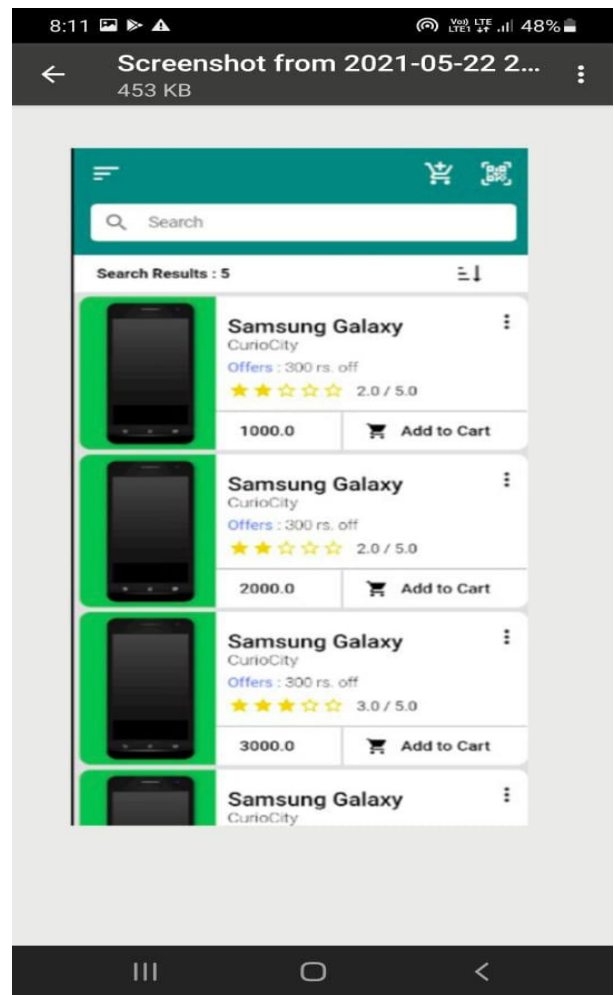


Fig-7

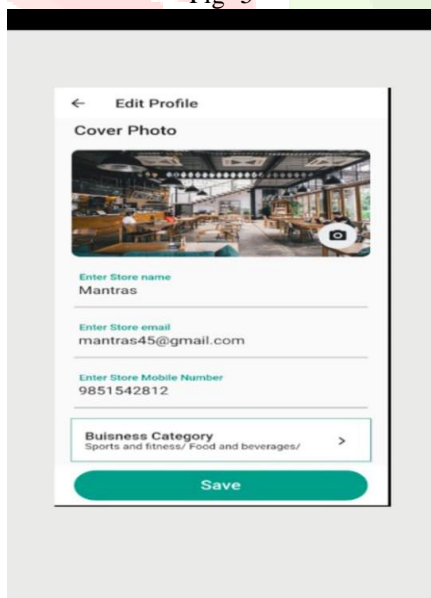


Fig-6



Fig-8

VI. Application

It is applicable in Local City for getting the Information or any goods items from people in same city and provide them the best service to families with no Delay.

VII. Advantages

Buying/selling a variety of goods and services from one's home or business → Anywhere, anytime transaction → Can look for lowest cost for specific goods or service → Businesses can reach out to worldwide clients - can establish business partnerships → Order processing cost reduced → Electronic funds transfer faster → Supply chain management is simpler, faster, and cheaper using ecommerce - Can order from several vendors and monitor supplies. - Production schedule and inventory of an organization can be inspected by cooperating supplier who can in-turn schedule their work

VIII. Future Scope

To increase the applicability and usage of this project. Here we can maintain the records of Food and Items, Category. Also, as it can be seen that now-a-days the players are versatile, i.e. so there is a scope for introducing a method to maintain the Online E-commerce Application. Enhancements can be done to maintain all the Food, Item Category, Shopping Cart, Customer, Orders etc.

eCommerce Application is used to assist people making purchases online. The Business-to-Customer aspect of electronic commerce (e-commerce) is the most visible business use of the World Wide Web. The primary goal of an e-commerce site is to sell goods and services online.

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web site providing functionality for performing commercial transactions over the web. It is reasonable to say that the process of shopping on the web is becoming commonplace

IX. References

- Jawed Siddiqi, Babak Akghar, Carl Davies and Samir Al-Khayatt, E-Commerce: Continuous Growth or Leveling Out, *Proceedings of IEEE International Conference on Information Technology: Coding and Computing*, 2002, pp. 491-496
- Daniel A. Menasci, A Reference Model for Designing an E-Commerce curriculum, *IEEE Concurrency*, Vol.8, No.1, 2000, pp. 82-85.
- Chiwei Lan, Chunchou Chien, Mengyen Hsieh and Irene Chen, A Mobile E-Commerce Solution, *Proceedings of IEEE International Symposium on Multimedia Software Engineering*, 2000, pp. 215-222
- Dongkyu Kim, Jaebum Kim and Sanggoo Lee, Catalog Integration for Electronic Commerce through Category-Hierarchy Merging Technique, *Proceedings of IEEE 12th International Workshop on Research Issues in Data Engineering: Engineering E-Commerce/ e-Business Systems*, 2002, pp. 28-33.
- Howard Bennett and Melissa L. Jaramillo, Assessing Web-Enabled Call Center Technologies, *IT Professional*, Vol.3, No.3, 2001, pp. 24-30.

