**JUST ORDER**

**A Mini-Project Report**

**Under**

**Project Workshop**

***Submitted by***

**VIRAJ ANCHAN (B007)**

**SARANG DESHPANDE (B021)**

**DEEP DOSHI (B023)**

***Under The Guidance Of***

**PROF. PALLAVI HALARNKAR**

***in partial fulfillment for the award of the degree***

***of***

**B.TECH**

**IN**

**COMPUTER ENGINEERING**

**at**

**mpstme, NMIMS UNIVERSITY, MUMBAI**

**APRIL 2014**

**CERTIFICATE**

This is to certify that the project entitled “JUST ORDER” is the bonafide work carried out by VIRAJ ANCHAN, SARANG DESHPANDE, DEEP DOSHI of B.Tech (Computer Engineering), MPSTME (NMIMS), Mumbai, during the VI semester of the academic year 2013-2014, in partial fulfillment of the requirements for the award of the Degree of Bachelors of Technology as per the norms prescribed by NMIMS. The mini-project work has been assessed and found to be satisfactory.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Pallavi Halarnkar

Internal Mentor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Examiner 1 Examiner 2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dean

Dr. S. Y. Mhaiskar

**DECLARATION**

We, VIRAJ ANCHAN, SARANG DESHPANDE, DEEP DOSHI, Roll No. B007, B021, B023 B.Tech (Computer Engineering), VI semester understand that plagiarism is defined as anyone or combination of the following:

1. Un-credited verbatim copying of individual sentences, paragraphs or illustration (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.
2. Un-credited improper paraphrasing of pages paragraphs (changing a few words phrases, or rearranging the original sentence order)
3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did wrote what. (Source: IEEE, The institute, Dec. 2004)
4. I have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of my work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.
5. I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such a complaint occurs. I understand fully well that the guide of the seminar/ project report may not be in a position to check for the possibility of such incidences of plagiarism in this body of work.

|  |  |  |
| --- | --- | --- |
| Signature of the Student  Name: Viraj Anchan  Roll No. B007  Place: Mumbai  Date: | Signature of the Student  Name: Sarang Deshpande  Roll No. B021  Place: Mumbai  Date: | Signature of the Student  Name: Deep Doshi  Roll No. B023  Place: Mumbai  Date: |

**ACKNOWLEDGEMENT**

We would like to express our special thanks to our mentor, Ms. Pallavi Halarnkar for her support and guidance throughout the project and while preparing the report for the project.

We would also like to acknowledge our project workshop Prof. Supriya Agrawal who taught us the basics of developing an Android application. This helped us in developing of our application.

We would also like to extend our sincere thanks to the University and individuals who gave us the opportunity to do this project. This helped us to gain more knowledge about the subject.

We would also like to thank our friends who constantly encouraged us towards developing this project and making this idea into reality.

**Table of contents**

**CHAPTER NO. TITLE PAGE NO.**

List of Figures i

Abstract ii

1. INTRODUCTION

1.1 Project Overview

1.2 Hardware Specification

1.3 Software Specification

2. INSTALLATION & SETUP

2.1 Installing ADT

2.2 Set Up

3. ANALYSIS & DESIGN

3.1 ANALYSIS

3.1.1 External Interface Requirements

3.1.2 Functional Requirements

3.1.3 Use Case Diagram

3.1.4 Non-Functional Requirements

3.1.5 Design Constraints

3.1.6 General Constraints

3.1.7 Sequence Diagram

3.1.8 Data Flow Diagram

3.1.9 State Chart Diagram

4. PROJECT IN DETAIL

5. CONCLUSION & FUTURE SCOPE

REFERENCES

**List of Figures**

**CHAPTER NO. TITLE PAGE NO.**

**2. INSTALLATION & SETUP**

Fig 2.1 Permissions Request 3

Fig 2.2 Installing 4

Fig 2.3 Installed 4

**3. ANALYSIS AND DESIGN**

Fig 3.1 Use Case Diagram 5

Fig 3.2 Sequence Diagram 7

Fig 3.3 Data Flow Diagram 7

Fig 3.4 State Chart Diagram 8

**4. PROJECT IN DETAIL**

Fig 4.1 Registration & Login 9

Fig 4.2 Menu 10-11

Fig 4.3 About Us 12-13

Fig 4.4 Gallery 14-15

Fig 4.5 Call Waiter 16-17

Fig 4.6 Menu 18

Fig 4.7 Order 18

Fig 4.8 Menu-3 18

Fig 4.9 Change Order 18

Fig 4.10 View and Confirm Order 18

Fig 4.11 Invalid Entries 18

**Abstract**

The project – Android application named JustOrder is a proprietary application which will cater to simplification of a restaurant’s ordering process. Through this application, the customer can order required items of desired quantity, view menu items as well as call for assistance using server side scripting. The purpose of the application is to reduce the human labor required for a restaurant to take orders from the customers and also to cut down on the usage of papers. With this application, the restaurant can have a simplified, error free and faster ordering process as compared to that involving human labor.

The application will have two parts to it. One is the application itself and the second will be the server side interface which will display the orders placed by the customers. The application will be controlled by the user while the control of the web interface will be with the manager of the restaurant.

1. **INTRODUCTION**
   1. **Project Overview**

Just Order will be a proprietary application developed for restaurants that will allow customers to place an order through a tablet based interface. It provides the user with an interactive and more informative way to place their order. It also helps the restaurant to have a hassle free way to communicate with the customer and also provides a detailed review of the services offered.

Its main purpose is to automate the process of placing an order in the restaurants. Therefore it reduces the amount of time required to place an order and also to avoid confusion. The customer will place an order through a tablet and the manager will be able to view the order details. Since this is a data-centric product it will need somewhere to store the data. For that, a database will be used. Both the mobile application and web portal will communicate with the database, however in slightly different ways. The mobile application will only use the database to get data while the web portal will also add and modify data. All of the database communication will go over the Internet. The mobile application has some restrictions about the resource allocation.

* 1. **Hardware Requirements**

RAM – Minimum 512 MB  
STORAGE SPACE – Minimum 16.3 MB   
NETWORK – Wi-Fi/GPRS/3G (atleast one of these)

* 1. **Software Requirements**

Operating System  
ANDROID 4.0 or higher

Server (Localhost)

WAMP/XAMPP Server for hosting PHP files.

1. **INSTALLATION AND SETUP**

Since the application will be restaurant specific, it will not be uploaded to the Play Store. Instead, the restaurant will have tablets on which the application will be installed.

1. The permissions are required to be accepted by the user to install the application. Following permission are required :

* Full Network Access – For placing orders or for using the assistance call functionality
* Phone calls – This will be primarily used to upload data to the server through the mobile/tablet Wi-Fi Hotspot.

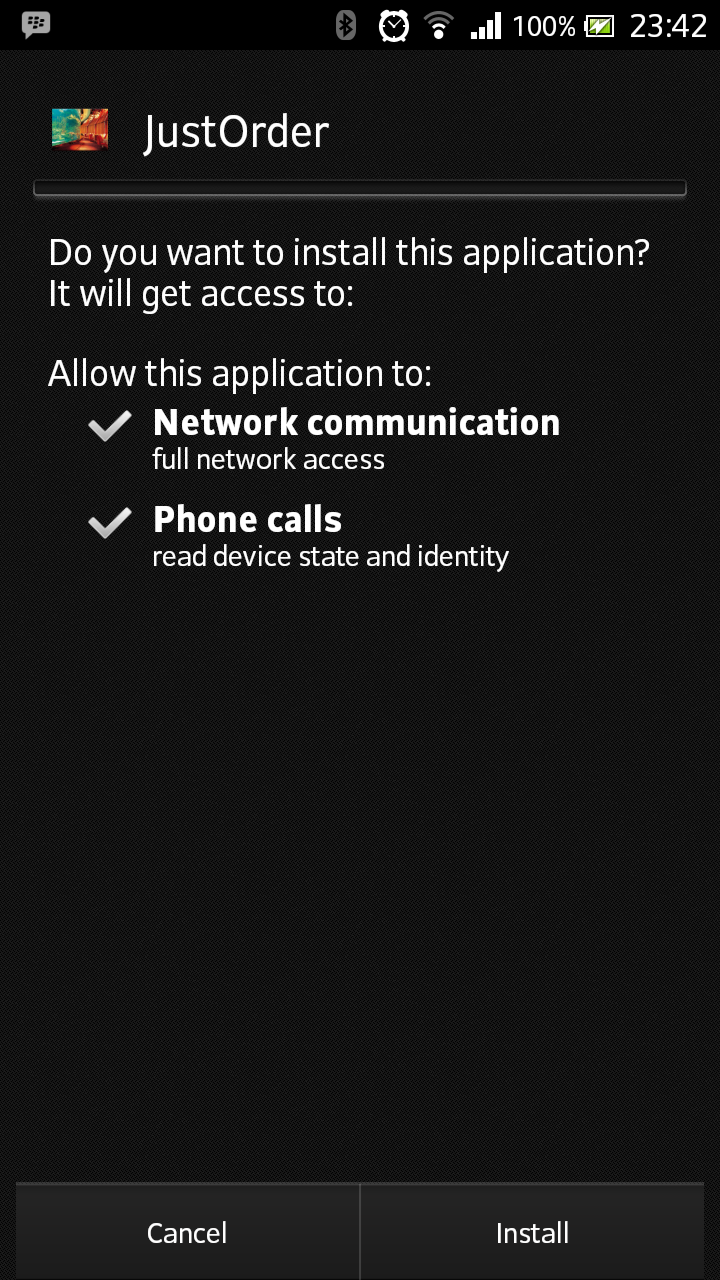


Fig 2.1 Permission Requests

1. After verifying the permissions user is required to click on the ‘Install’ button which will start installing the application on the user’s device.

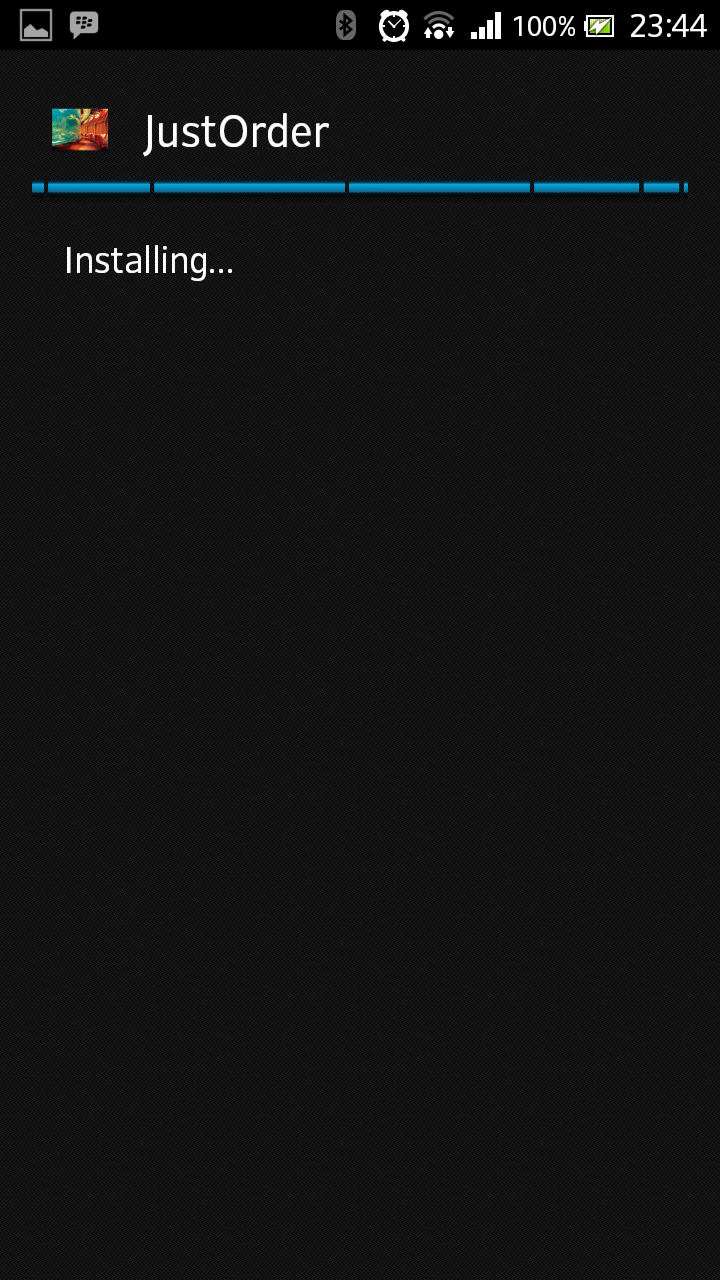


Fig 2.2 Installing

1. Once the application is installed the user has to click open button to view the application.

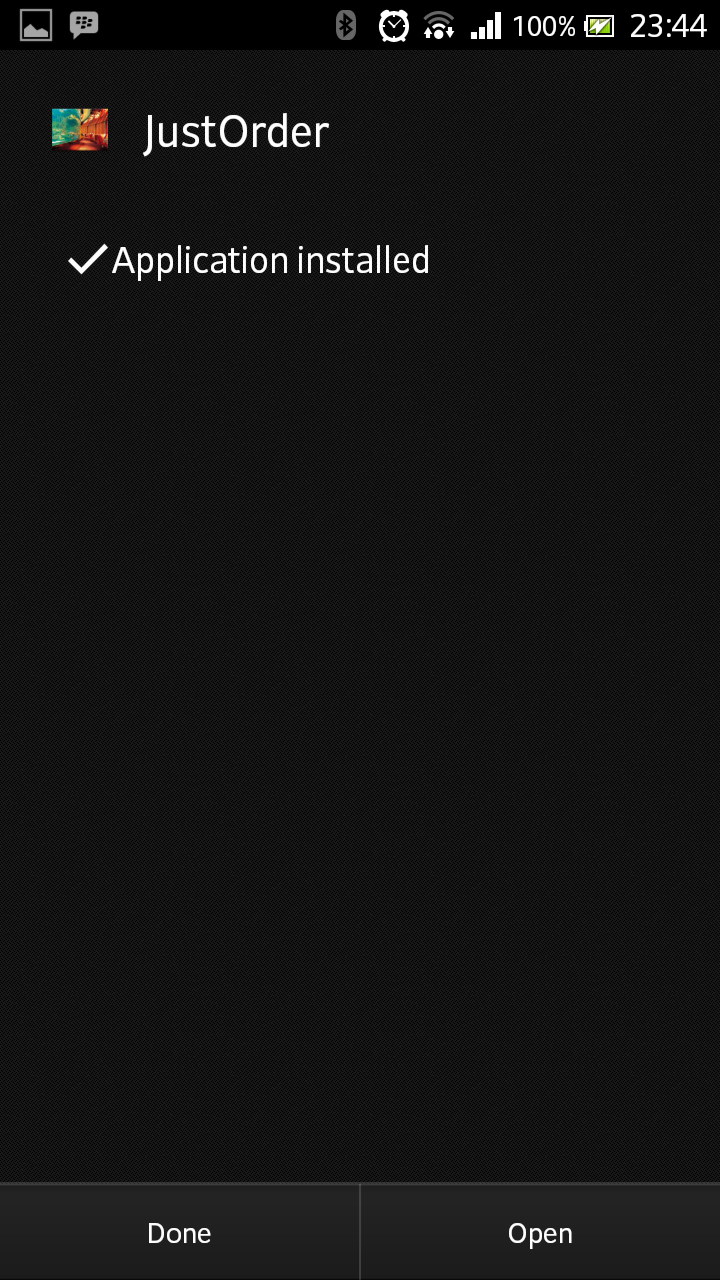


Fig 2.3 Installed

1. **ANALYSIS & DESIGN**
   1. **ANALYSIS**

**3.1.1 External Interface Requirements**

**User Interfaces**

When the user opens the application, the user gets to see various tabs that allow the user to see various options like Menu, About us, Order and Feedback. The menu tab will have many sub options like Starters, Main Course, Beverages and Desserts. Each sub options will show various food items. Each food item will have their own page which will provide detailed info and allow the customer to place an order. Upon submitting the order, the food item will be added to the orders tab. The orders tab will show all the food items that have been ordered by the customer. After the user has selected all food items, then he can click on pay option to get the detailed bill.

**Hardware Interfaces**

A stable network connection is required for the tablets to connect to. A router is also required for a wireless network to be set up. All tablets in the restaurant that have the application will connect to this wireless network. Once connected to the same network, data transfer between the tablets will be possible via the router over the network connection.

**Communications Interfaces**

After the connection on all tablets is established, there will be a server on which data will be stored for further transmission to the other tablets. The data will be transferred from the tablet to the server via the router. The tablets will communicate with the server by connecting to the same network.

**3.1.2 Functional Requirements**

The application is designed for individual restaurants. Design of the application will differ by each restaurant. The customer will place an order from the tablet that will given to him. The order placed will go to the tablet that is handled by the manager, via the router, on the network. The order will then be processed. The payment process can be an added functionality in the application where the customer will be allowed to make final bill payment of his order via credit or debit card. The customer can then make secure payments to the restaurant. Secure payment gateways need to be established so as to successfully process all transactions.

**3.1.3 Use Cases**



Fig 3.1 Use case Diagram

**3.1.4 Non-Functional Requirements**

The designed software will perform smoothly unless there is some undesired action by the user on the interface of the application. Enough memory is allocated to the application so that there is no lack of memory for proper functioning of the application.

**Reliability**

The application will provide reliable information regarding the menu items available in the restaurant. Since the database will be provided by the restaurant itself, the information can be treated as 100% reliable.

**Availability**

The application is designed to be a proprietary application. Hence, it will not be available on the Play Store. It will only be distributed to those restaurants who will buy the application separately.

**Security**

The information is stored only within the application and on a secured server through which data from one client tablet is sent to the primary (host) tablet. Since the data is stored only on a network, it will be password protected and hence a secure connection is established.

**Maintainability**

The application will have to be maintained from time to time. Operations like update of databases, checking for any errors on the server, checking the network connection (which is a primary requirement of the application) are mandatory. These operations will ensure the smooth working of the designed application.

**Portability**

The application is portable from one device to the other as far as the minimum android version required is complied with. The application can be transferred from one device to the other easily and without any constraints.

**3.1.5 Design Constraints**

As this is a proprietary application, the design of the application will vary from restaurant to restaurant. Designs given by each restaurant will have to be adhered to while the basic working of the application remains the same. Constraints like the menu displayed, certain display specifications will also differ for each restaurant.

**3.1.6 General Constraints**

Other requirements of the application include a basic network connection on which the tablets will be connected. Router will be used to create wireless network so that the tablets can communicate with each other and data can be transferred as part of the basic working of the application.

**3.1.6 Sequence Diagram**

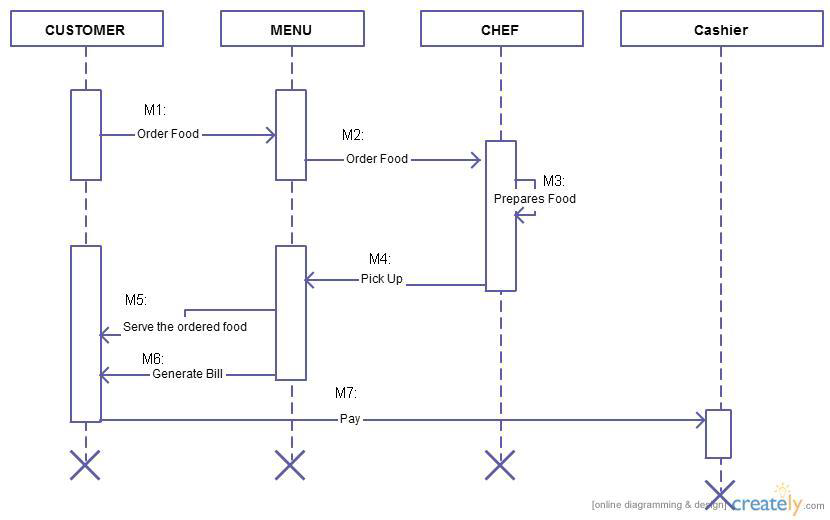


Fig. 3.2: Sequence Diagram

**3.1.7 Data Flow Diagram (DFD)**

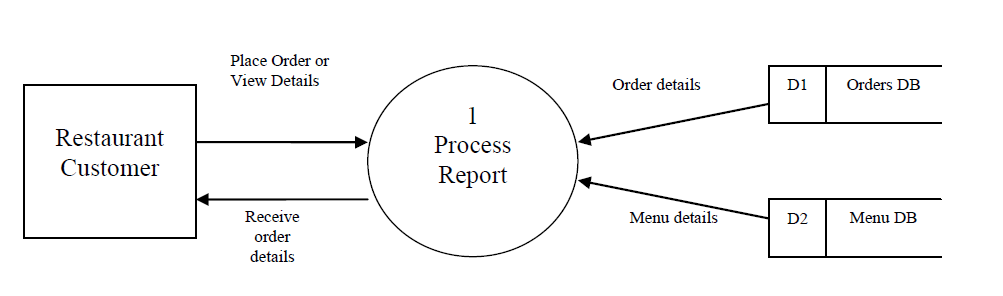


Fig 3.3: Data Flow Diagram

**3.1.7 State Chart Diagram:**

**States:**

1. Place Order – This state is reached on event of selecting the menu. Here, the customer creates the order to be placed.

2. Display Full Order – In this state, the customer is shown the order which is finally created. Further state is determined based on whether the customer wants to edit the order or is done with adding items to the order.

3. Order Placed – Once the customer is satisfied with the order, he places the order and this state is reached. Here, the order is sent to the server. From here, the order is delivered to the customer and then system goes to end state.



Fig 3.4 State Chart Diagram

1. **PROJECT IN DETAIL**

**Working of our Application:**

JustOrder is a client-server application. For server coding, PHP scripts have been used. MySQL database has been used to store the orders placed by the customers. The Android application makes a connection to the PHP scripts on the server using HTTPClient and HTTPPost methods. The PHP scripts further establishes a connection to the MySQL database to insert the values into the table. The values from the Android application are passed to the PHP scripts which pass it to the tables in the database. For all functionalities of the application, a working Internet connection is a prerequisite. An alternative is to connect all the tablets to a common network.

Working of the application along with the screenshots is explained below:

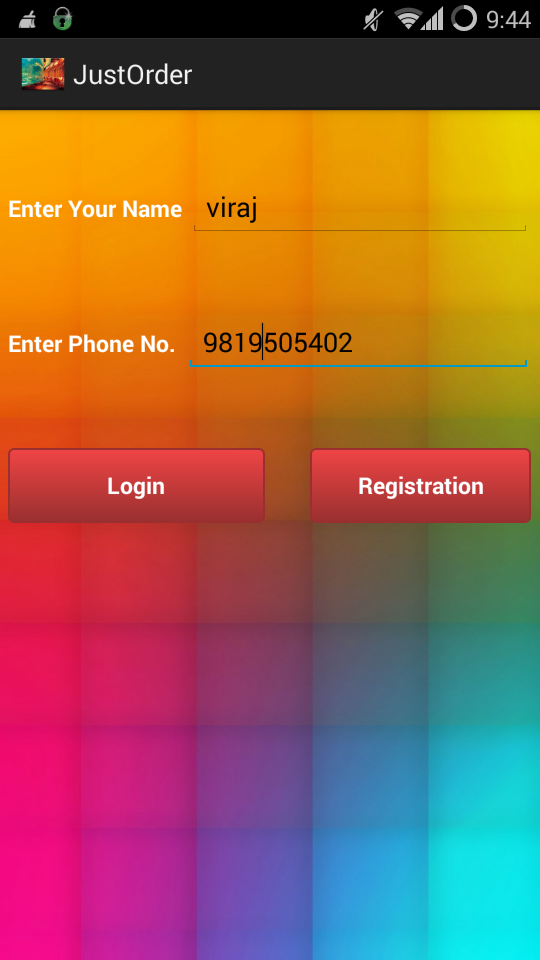
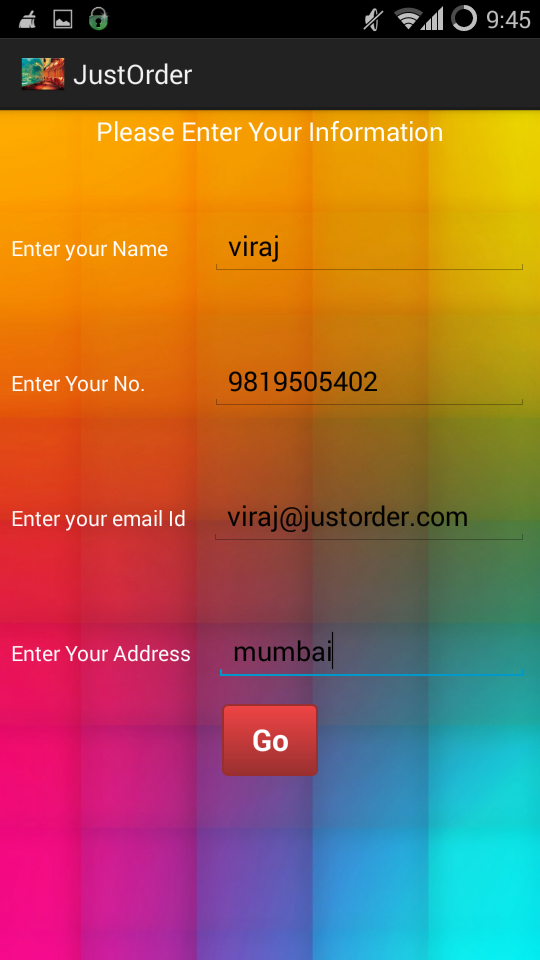


Fig 4.1 Registration & Login

The above screenshots show the registration page (on the left) and the login page (on the right). When the application is opened, the customer is prompted to login. If it is a first time login, the customer has to first register to the application for him to use it. After deployment of the application, the tablets will be logged in before handing them to the customers. The mobile number and email id fields are validated so that they take only valid values into their respective fields. If the details do not match, login fails.

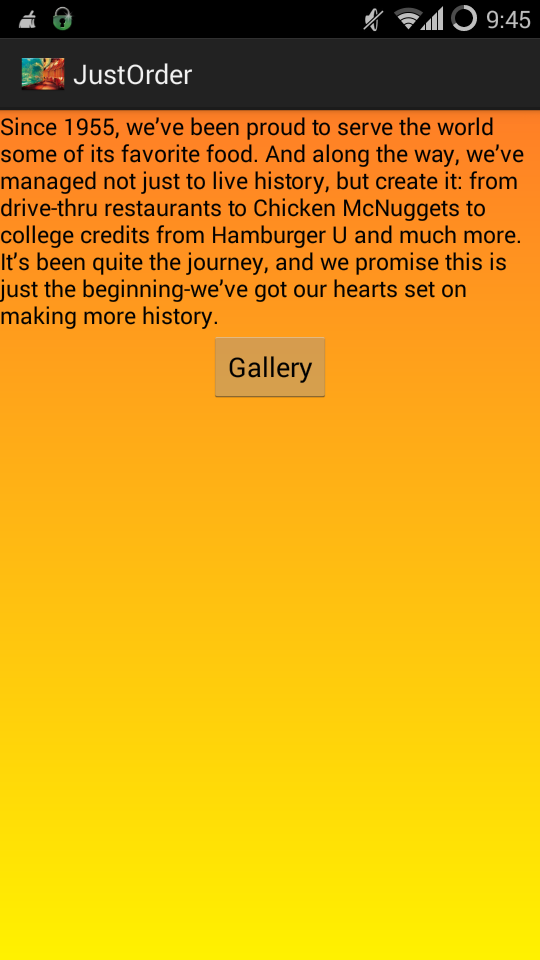
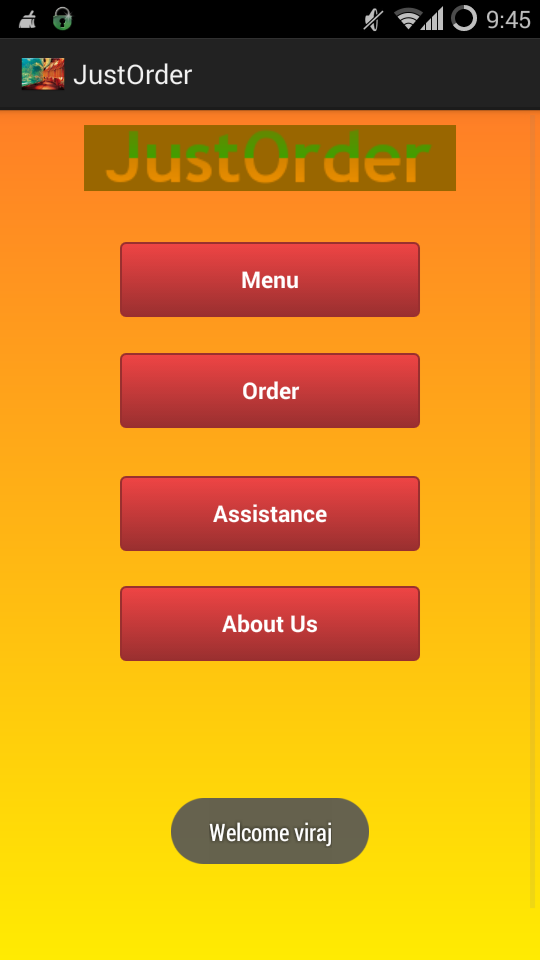


Fig 4.2 Menu Fig 4.3 About Us

The above screenshots show the main menu (on the left) and the About us page (on the right). The name, with which the application is logged in, is shown as toast on the main page. The buttons on the main menu will navigate to different pages such as the menu, the current order details and assistance call. The About us page will show some information about the restaurant where this application will be deployed. The button on the About us page will display the photos of the items provided by the restaurant.

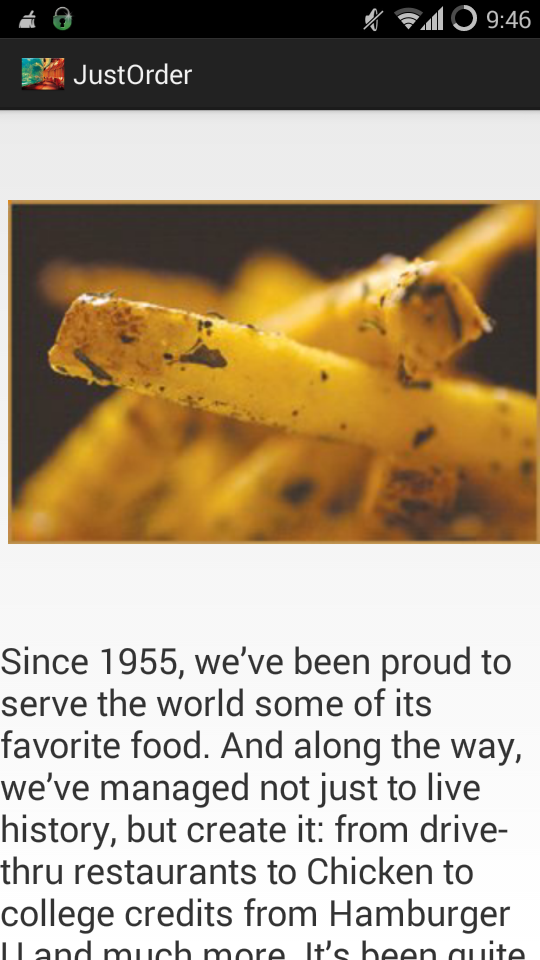
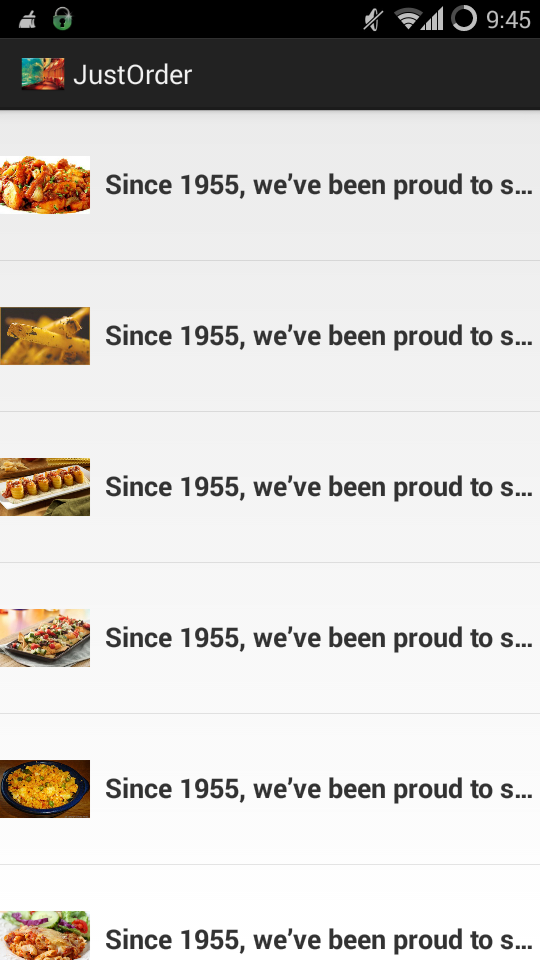


Fig 4.4 Gallery

The above screenshots show the gallery as a list (left) and individual description (right).

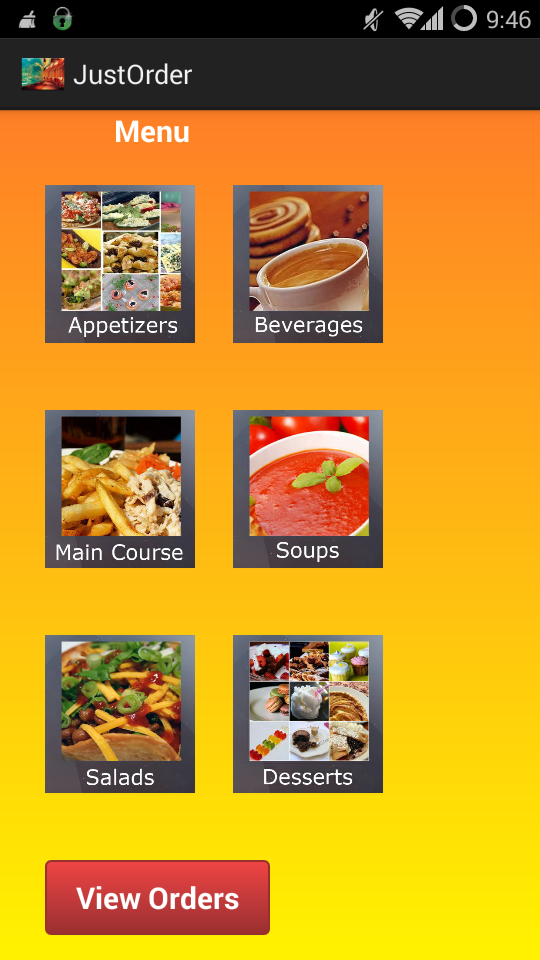


Fig 4.5 Call Waiter Fig 4.6 Menu

The left screenshot shows the call for assistance option. When the button is pressed, a call will be made to the PHP script and the current date and time will be displayed on the web interface of the PHP script. The screenshot on the right shows the menu of the restaurant divided into various sections. The button at the bottom shows the items currently in the order list.



Fig 4.7 Order

The screenshot on the left shows the page after clicking one of the sections of the menu. The right image shows the page after one of the items is selected. The quantity can then be mentioned and it is added to the order by clicking the ‘Add’ button.

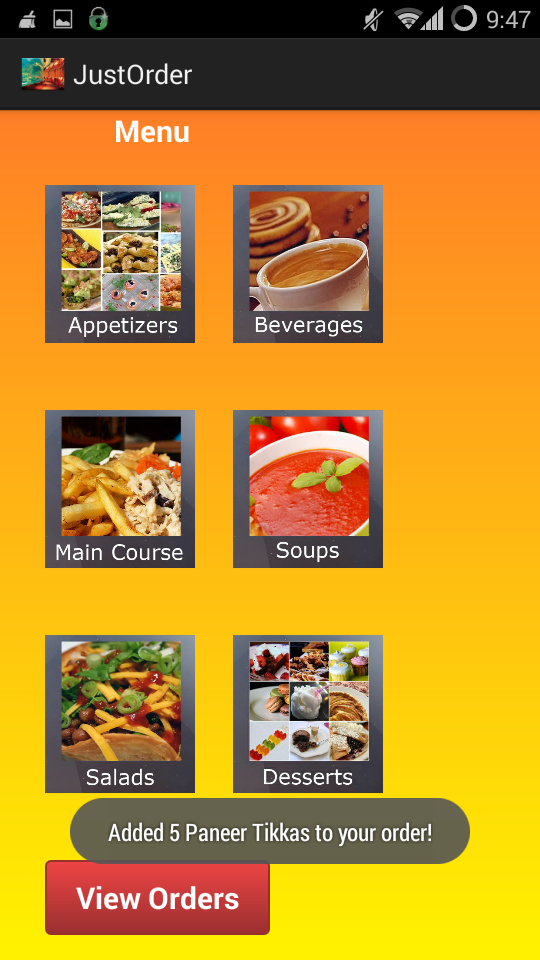


Fig 4.8 Menu-3

After adding the desired quantity, a toast is shown on the menu page which displays the quantity ordered by the customer, which is added to the order list. The same process for a different section of the menu is shown on the right.

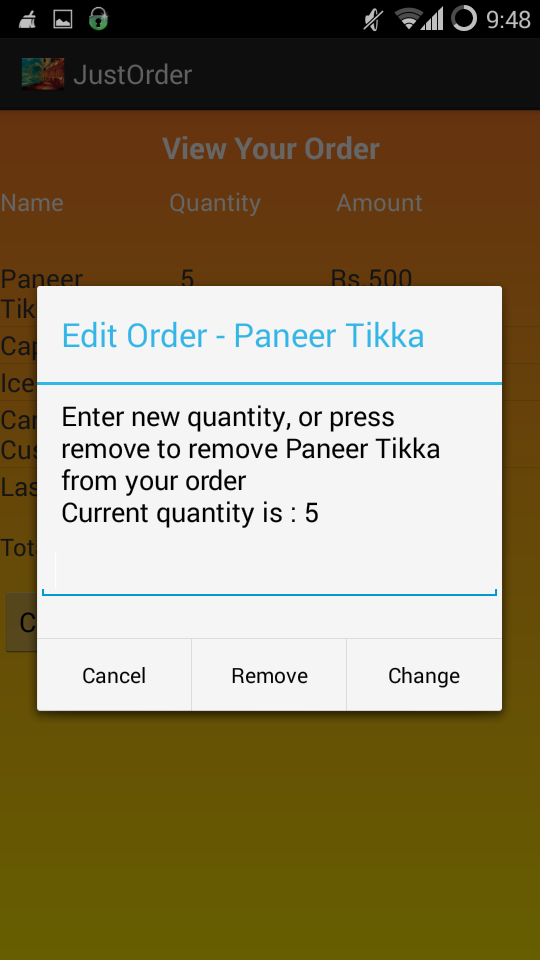


Fig 4.9 Change Order

The screenshot on the right is seen when the ‘View Order’ button is pressed on the menu page. Here, the individual items can be selected and the quantity for the same can be changed or the item can be completely removed from the order list.



Fig 4.10 View and Confirm Order

The screenshot on the left is seen as the final order list created by the customer. It shows all the items added along with the quantity and the total price of each item. The screenshot on the right is seen when the ‘Confirm Order’ button is pressed. A toast is shown which displays the message, confirming order. The ‘End’ button ends the current session and logs out of the current account. It takes the customer back to the login page.

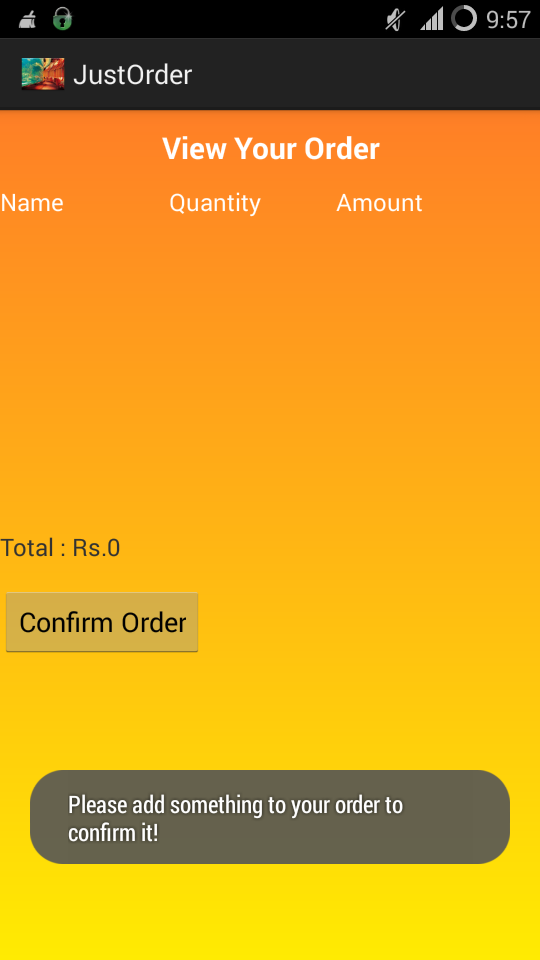


Fig 4.11 Invalid Entries

The screenshot on the left shows a toast displaying ‘Please enter a valid quantity’. This is see when the quantity of the item ordered is zero. The field is validated to accept only non-zero values. The screenshot on the right can be seen when there is nothing added to the order list and still the customer tries to confirm the order or place the order. A toast is shown which shows the message ‘Please add something to your order to confirm it’.

1. **CONCLUSION & FUTURE SCOPE**
   1. **CONCLUSION**

Thus we have created an application which will help to simplify and automate the ordering process of a restaurant. It aims at reducing human labour required by the restaurant. The interface provided is simple and user-friendly. It will also provide a new and enriching experience to the customers visiting the restaurant.

* 1. **FUTURE SCOPE**

The application can be deployed at restaurants individually. The user interface and the menu of the application will vary from restaurant to restaurant. It will be designed based on their requirements.

While deploying the application, the PHP files can be hosted on an actual server instead of server on a local machine. This will make the execution faster and smoother.