1.INTRODUCTION

Cibus Partum is a food delivery application which is simply CIPA. To provide quality and variety in food to passengers travelling by bus with CIPA, passengers now have the freedom to order food of their choice. Passengers can order food on any of the bus stations which are already predefined. From a simple combo of daal, roti, and chawal to Mughlai biryani and delicious pizzas, passengers can exercise their freedom and enjoy the yummy food on roadways.

Any passenger having valid reservation ticket can book the meal through this application. In the selection page, enter the service no. Then the stations will be appeared in list. If search is made by service number, the application will seek your phone number for verification purposes before confirming the booking.

- On selecting the bus station at which meal is to be delivered, vendor menu will open along with item prices.
- Select the vendor and then select the meals to be booked. Prices of meal will be displayed alongside.
- On placing the order, the passenger will be required to select the mode of payment i.e. Cash on Delivery. Thereafter, confirmation of booking is to be made.

1.1Existing System

-On trains.

Indian railways expanded over 7,349 stations are a vast expanse and the most trafficked mode of travel. With over 23 Million passengers traveling every day, train travel seems to be driving force of Indian population. Food, no doubt a major concern for people. Getting your favorite food in train journey was a nightmare. People had to suffer and stay contented with the thought that the best food they are bestowed with is the pantry food which was till recently acclaimed as safe for consumption.

1.2Proposed System

-On buses.

The main idea behind this project is to ease the passengers travelling distant locations especially by bus. It overcomes the disadvantage like unavailability of proper and tasty food during their journey. This application will have some pre-identified locations; there the food can be served to the passengers on time. The passengers just need to log-in to the app and select their favorite food near pre-identified location. The menu and all the other information like time, location, and payment are available in the app itself. One can select the location and food of their own preference and wait for the confirmation message. Finally user can enjoy the food of their choice on the wheels and get rid of hunger at a reliable and effective cost.

2.REQUIREMENT ANALYSIS

Requirement analysis is one of the most important thing to make a good software. In this there are two types of requirement analysis.

2.1 H/w & S/w Requirements

2.1.1 Hardware Requirements

The main hardware requirement used in **Cibus Partum** are

- Processor: **Minimum** 1 GHz; Recommended **2GHz** or more.
- Ethernet connection (LAN) OR a wireless adapter (Wi-Fi).
- Memory (RAM): **Minimum** 1 GB.
- Operating system: Android and IOS.

2.1.2 Software Requirements

The main software requirements in **Cibus Partum** are

- Node.Js
- MySQL Workbench
- Eclipse Java
- Java JDK
- Visual Studio code

2.2 Software Requirements Specification

2.2.1. Vision document

- For the people who are travelling by bus can use this application location, the CIBUS PARTUM is a android application location that provides the Users to order food .

- It overcomes the disadvantage of traditional queuing system, we can order food via this application location by selecting the location, preferences and proceed to the payment. Finally, User can enjoy the food of their choice on the wheels.

2.2.2. Scope

Generally, the passengers on bus faces a lot of problems regarding food during their journey. In order to avoid such problems, Cibus Partum provides ease to the passengers travelling distant locations especially by bus. It overcomes the disadvantage like unavailability of proper and tasty food during their journey. This application will have some pre-identified locations; there the food can be served to the passengers on time. The passengers just need to log-in to the app and select their favorite food near pre-identified location. The menu and all the other information like time, location, and payment are available in the app itself. One can select the location and food of their own preference and wait for the confirmation message. Finally user can enjoy the food of their choice on the wheels and get rid of hunger at a reliable and effective cost.

Overview:

In this application location, we will provide Users to order food who are travelling in the bus. Users can also give feedback through this application. So that admin can evaluate the whole system. Users can also make payment by cash on delivery. Also the required information about employees will be saved in the system which can be only accessed by the system admin.

Exclusions:

- Some food items will not be suitable for delivery, so the menus available to users will be a subset of the full food items list.
- Who are travelling only by bus.
- The food on wheels application location shall be used only for those users who have registered.

Assumptions:

- The user should order their food at least one hour before their destination.
- The CIPA application location shall be used only for those users who have registered.

2.2.3. System Functions

S.No	System Function	Description
		Food Order
1	S1.1	Login into the application
2	S1.2	Enter the service number
3	S1.3	Select the delivery location where the food need to be delivered.
4	S1.4	Select the food items to be delivered.
5	S1.5	Proceed for the payment.
		Order approval
6	S2.1	Approve order
7	S2.2	Reject order
8	S2.3	Notify application approval/rejection to user and admin

9	S2.4	Display pending deliveries to the admin
10	S2.5	Check the orders history
		Order administration
11	S3.1	The orders of the Users will be seen, approved and sent to the delivery boy.
12	S3.2	After the food item is ready it is given to the delivery boy
13	S3.3	The place of the delivery is messaged to the delivery boy

2.2.4. Detailed Software Requirements

USE CASE MODEL

Actor Name	**User**
Actor Id	ACT-01
Description	Who wants to order the food by using the application location
Main Activities	The User can order food and see their payment receipt and pay
requency of use	High
Work Environment/Location	Browser
Number of Users	Any number

Actor Name	**Administrator**
Actor Id	ACT-02
Description	Handles all admin related tasks throughout the application location
Main Activities	Uses the system to setup initial data, define access control etc.
Frequency of use	Medium
Work Environment/Location	Browser
Number of Users	1

Actor Name	**Delivery boy**
Actor Id	ACT-03
Description	Handles all food to be delivered to the User
Main Activities	Delivers the food to the User to the desired location.
Frequency of use	High
Work Environment/Location	Road way
Number of Users	Any Number

List of Use Cases:

- Registration
- Order food
- Deliver Order

2.2.5. Functional capabilities

- Notifications should be sent to the User after the order is confirmed/rejected .
- The orders should be automatically updated and only after the order is confirmed.
- Messages should be sent to the delivery boy after the order is placed.

2.2.6. Business Rules / Validations

- Only the User can confirm or cancel the order.
- The User should order the food an hour before the destination.
- If once the order is confirmed then the user cannot cancel the order.

2.2.7. Security Requirements

User Management and Authentication:

- The telephone number of the user should be the identification for the user.
- The user should confirm the order before being processed.

User roles and access control:

- Only the admin can access the whole data.
- The system has to authenticate a user, determine whether he is a valid or not.

2.2.8. Other Non-functional requirements

- The application location should support the case of intermittent network connectivity i.e. the user should be able to order food in multiple sessions (one or more responses at a time). (Reliability and Supportability)
- Usability
- The application location should support interruptions for regular mobile device tasks i.e. receiving a call or a message. (Supportability)
- The form should load within 2 seconds. (Performance)
- The system should be available 24 X 7. (Reliability/ availability)

3.SYSTEM DESIGN

3.1 Data Dictionary

Term	**Definition**
System Admin	System admin is a person who is responsible for managing the whole system and who has full access to the system.
System User	A person who is using or operating the system but with a limited privilege.
Database	Collection of all the information monitored by this system.
Field	A cell within a form.
Software Requirements Specification (SRS)	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document. Stakeholder Any person who is involved in the development process of the software.

3.2 Logical Database Design

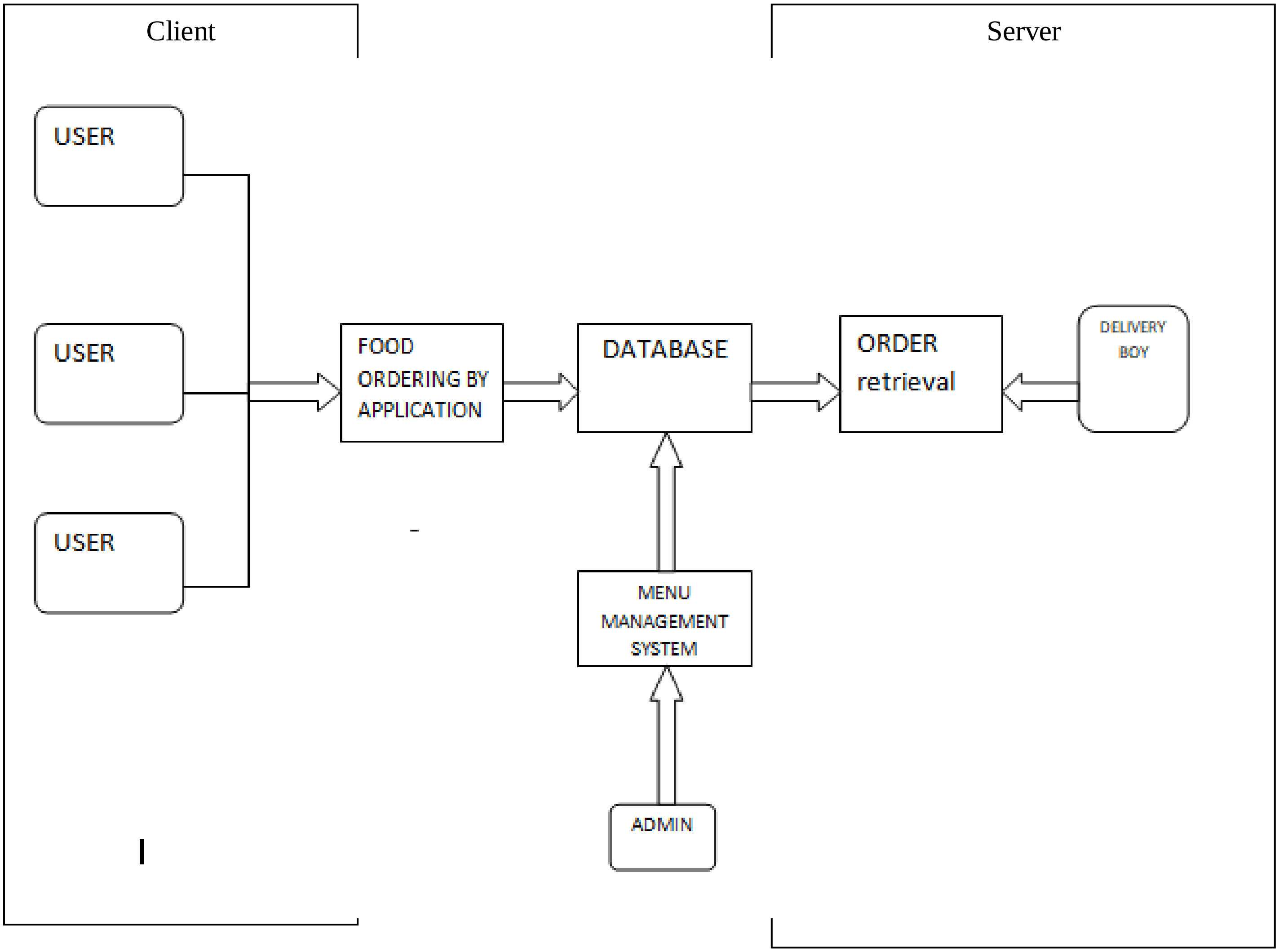


Figure 3.1 Logical Database Design

3.3 UML Diagrams

The unified modeling language allows the software engineers to express an analysis model using the modeling notation that is governed by a set of syntactic semantic and pragmatic rules

3.3.1 Use case Diagram

Use case diagrams are created to visualize the relationships between actors and use cases. A use case is a pattern of behavior the system exhibits. Each use case is a sequence of related transactions performed by an actor and the system. Diagrammatically actor and use case are represented by stick figure and oval respectively.

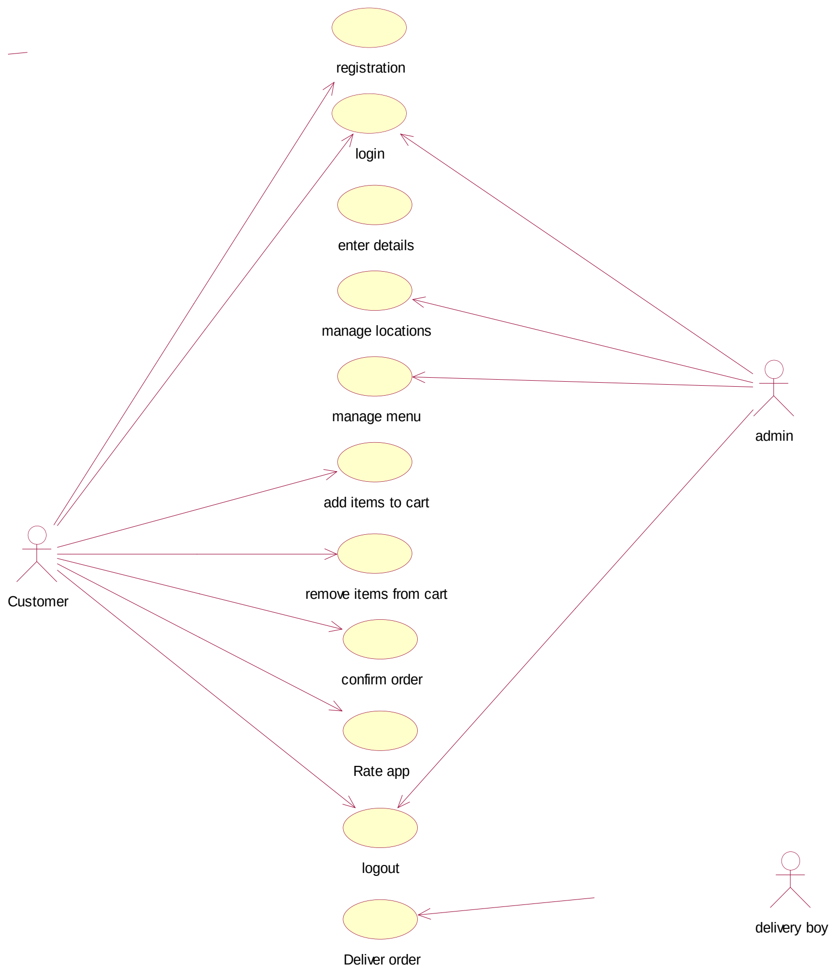


Figure 3.2 Use case diagram

3.3.2 Class Diagram

A Class diagram gives an overview of a system by showing its classes and the relationships among them. UML class is a rectangle divided into: class name, attributes, and operations.

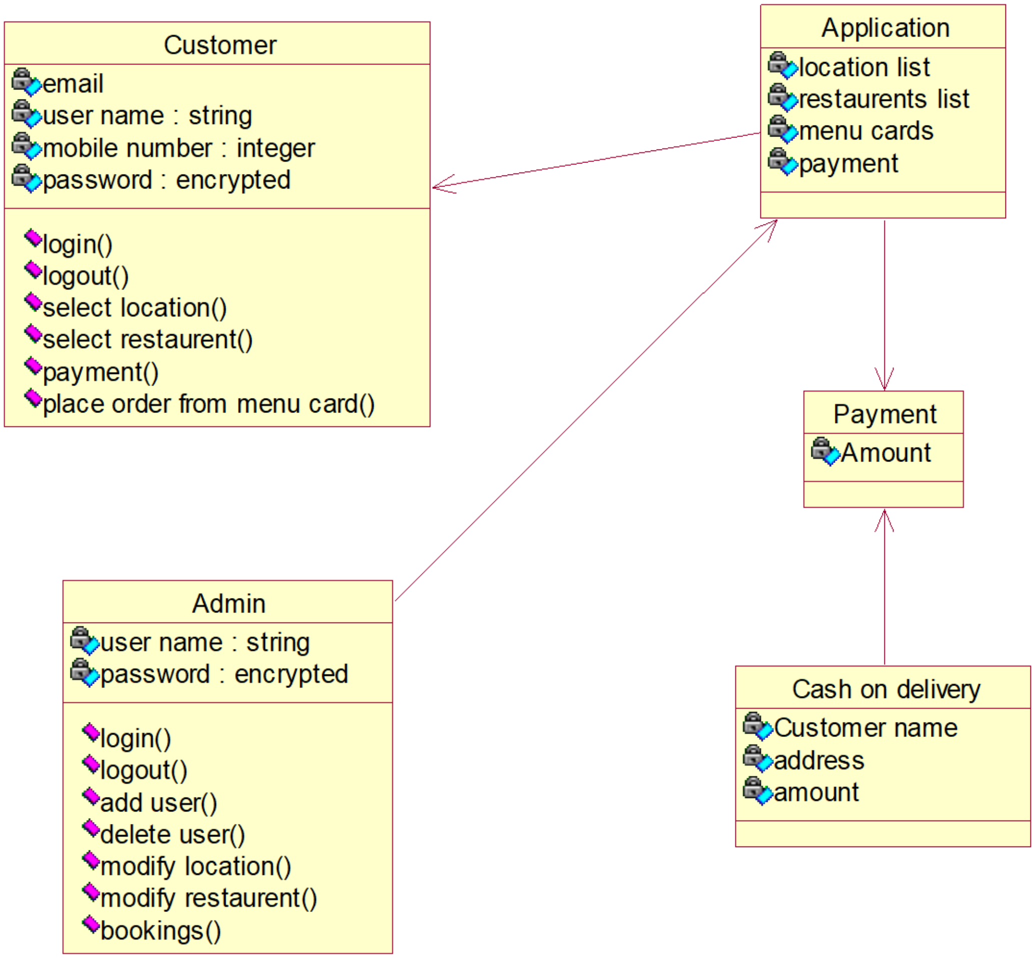


Figure 3.3 Class diagram

3.3.3 Sequence Diagram

Sequence diagram is an interaction diagram which focuses on the time ordering of messages. It shows a set of objects and messages exchange between these objects. This diagram illustrates the dynamic view of a system.

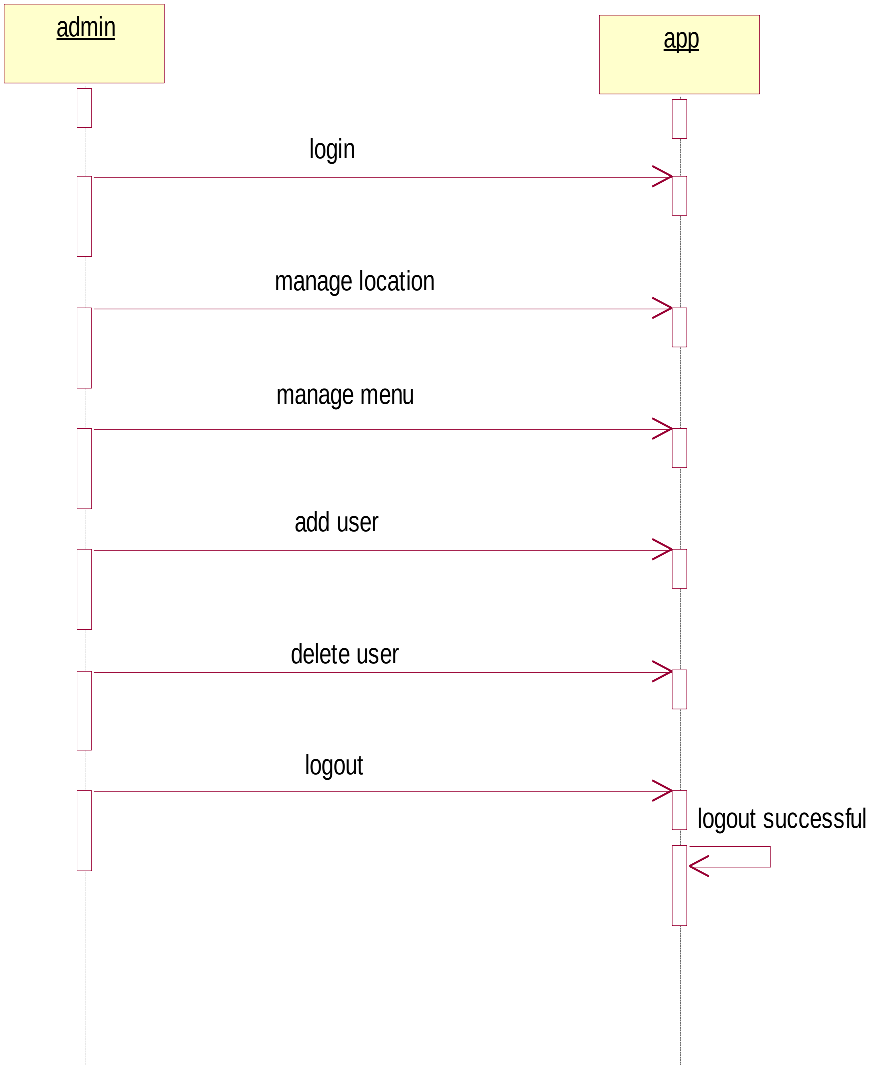


Figure 3.4 Sequence diagram for admin

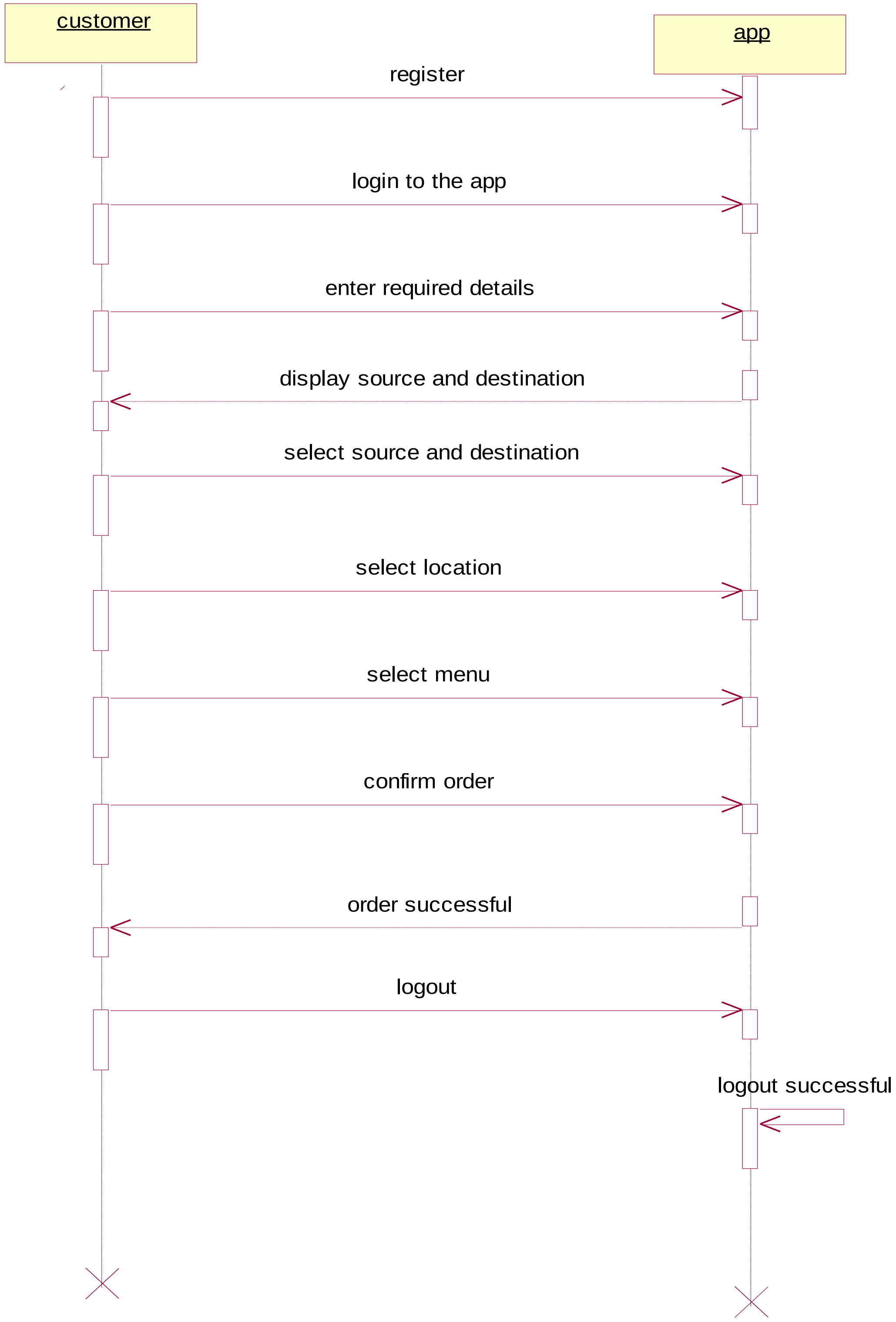


Figure 3.5 Sequence diagram for customer

3.3.4 Activity Diagram

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modelling. They can also describe the steps in a use case diagram. Activities modelled can be sequential and concurrent. In both cases an activity diagram will have a beginning (an initial state) and an end (a final state).

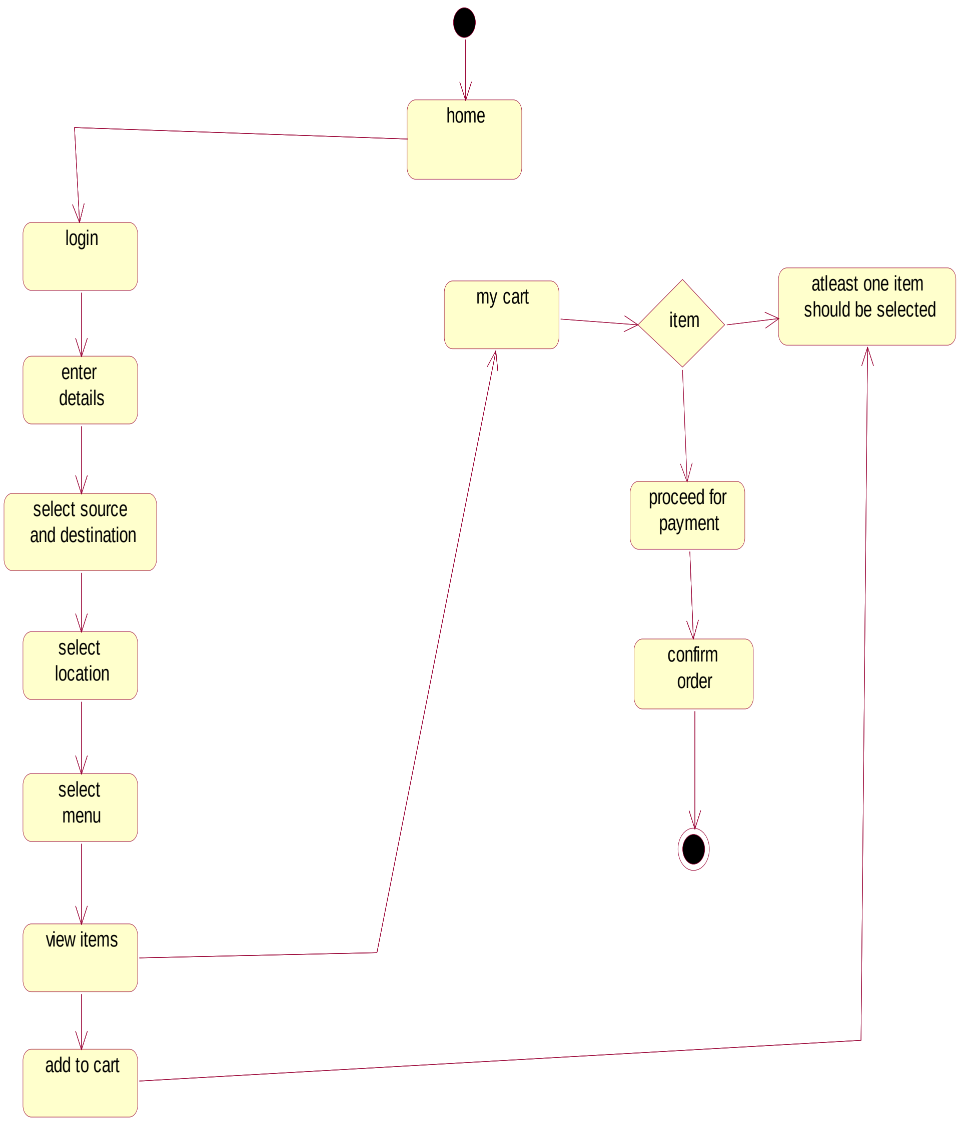


Figure:3.6 Activity diagram

3.3.5 ER diagram

It is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information.

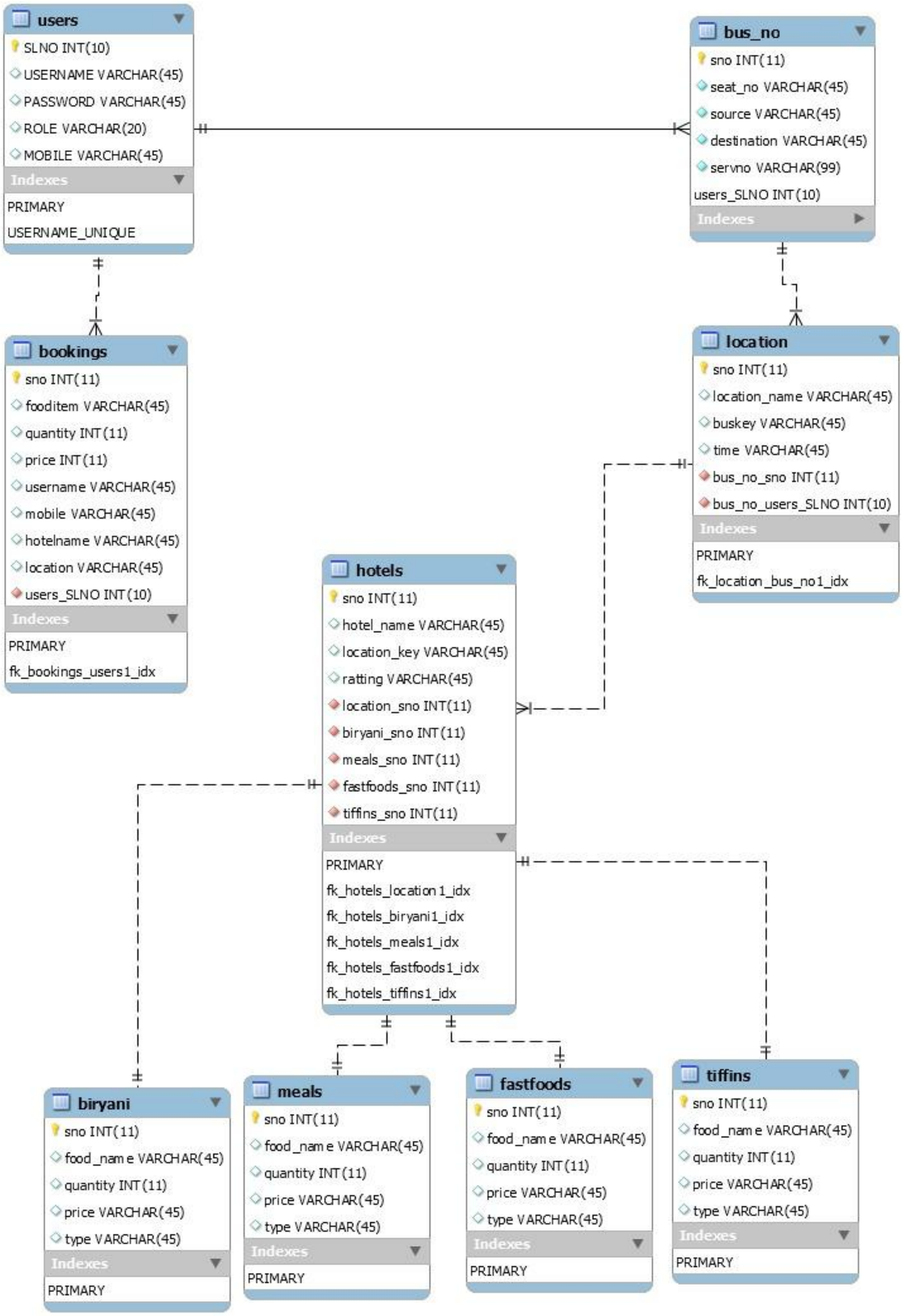


Figure 3.7 ER Diagram

4.SYSTEM IMPLEMENTATION

4.1 SELECTED SOFTWARE

Most software that you buy or download only comes in the **compiled** ready-to-run version. Compiled means that the actual program code that the developer created, known as the **source code,** has run through a special program called a compiler that translates the source code into a form that the computer can understand. It is extremely difficult to modify the compile version of most applications and nearly impossible to see exactly how the developer created different parts of the program. Most commercial software manufacturers see this as an advantage that keeps other companies from copying their code and using it in a competing product. It also gives them control over the quality and features found in a particular product.

Open source software is at the opposite end of the spectrum. The source code is included with the compiled version and modification or customization is actually encouraged. The software developers who support the open source concept believe that by allowing anyone who's interested to modify the source code, the application will be more useful and error-free over the long term.

To be considered as open source software by the software development industry, certain criteria must be met:

1. The program must be freely distributed
2. Source code must be included
3. Anyone must be allowed to modify the source code.
4. Modified versions can be redistributed.
5. The license must not require the exclusion of other software or interface with the operation of other software.

WHAT IS NODE.JS:

Node.js is an open-source, cross-platform JavaScript run-time environment that executes JavaScript code outside of a browser. JavaScript is used primarily for client-side scripting, in which scripts written in JavaScript are embedded in a webpage's HTML and run client-side by a JavaScript engine in the user's web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web application development around a single programming language, rather than different languages for server side and client side scripts.

Features of Node.js

Following is a list of some important features of Node.js that makes it the first choice of software architects.

1. **Extremely fast:** Node.js is built on Google Chrome's V8 JavaScript Engine, so its library is very fast in code execution.
2. **I/O is Asynchronous and Event Driven:** All APIs of Node.js library are asynchronous i.e. non-blocking. So a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call. It is also a reason that it is very fast.
3. **Single threaded:** Node.js follows a single threaded model with event looping.
4. **Highly Scalable:** Node.js is highly scalable because event mechanism helps the server to respond in a non-blocking way.
5. **No buffering:** Node.js cuts down the overall processing time while uploading audio and video files. Node.js applications never buffer any data. These applications simply output the data in chunks.
6. **Open source:** Node.js has an open source community which has produced many excellent modules to add additional capabilities to Node.js applications.
7. **License:** Node.js is released under the MIT license.

VISUAL STUDIO CODE:

Visual Studio Code is a source code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is also customizable, so users can change the editor's theme, keyboard shortcuts, and preferences. The source code is free and open source and released under the permissive MIT License. The compiled binaries are freeware and free for private or commercial use.

Visual Studio Code is based on Electron, a framework which is used to deploy Node.js applications for the desktop running on the Blink layout engine. Although it uses the Electron framework, the software does not use Atom and instead employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

Features:

Visual Studio Code is a source code editor that can be used with a variety of programming languages. It supports a number of programming languages and a set of features that differs per language.

Visual Studio Code can be extended via plug-ins, available through a central repository. This includes additions to the editor and language support. A notable feature is the ability to create extensions that add support for new languages, themes, debuggers, perform static code analysis, add code linters, using the Language Server Protocol and connect to additional services.

Visual Studio Code includes multiple extensions for FTP, allowing the software to be used as a free alternative for web development. Code can be synced between the editor and the server, without downloading any extra software.

JAVA ECLIPSE:

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins, including Ada, ABAP, C, C++, C#, Clojure, COBOL, D, Erlang, Fortran, Groovy, Haskell, Java Script, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Rust, Scala, and Scheme. It can also be used to develop documents with LaTeX (via a TeXlipse plug-in) and packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++, and Eclipse PDT for PHP, among others.

The initial codebase originated from IBM VisualAge. The Eclipse software development kit (SDK), which includes the Java development tools, is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules. Since the introduction of the OSGi implementation (Equinox) in version 3 of Eclipse, plug-ins can be plugged-stopped dynamically and are termed (OSGI) bundles

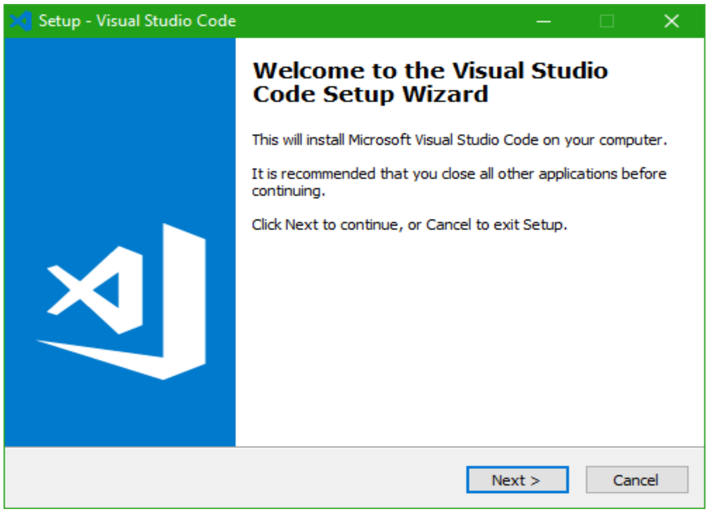
Eclipse software development kit (SDK) is free and open-source software, released under the terms of the Eclipse Public License, although it is incompatible with the GNU General Public License. It was one of the first IDEs to run under GNU Classpath and it runs without problems under IcedTea.

INSTALLATION STEPS

Installing Visual Studio Code

After you hit the download button for the package that matches your needs and configuration the most, the package will start downloading on your machine. Now, there are different ways to install this software on different operating system environments. Concentrate on how to install it on **Windows**. This procedure is purely same for every supported version of Windows Operating System.

So, now after downloading a file named as *VSCodeSetup-version.exe*, double-click on the file to install it. It will just take about a minute to install depending on how powerful hardware you are running. Now a UAC or User Account Control Prompt will be appeared and hit *Yes.*



Welcome to the Visual Studio Code Setup Wizard

This will install Microsoft Visual Studio Code on your computer.

It is recommended that you close all other applications before continuing.

Click Next to continue, or Cancel to exit Setup.

Now, hit *Next* to proceed further.

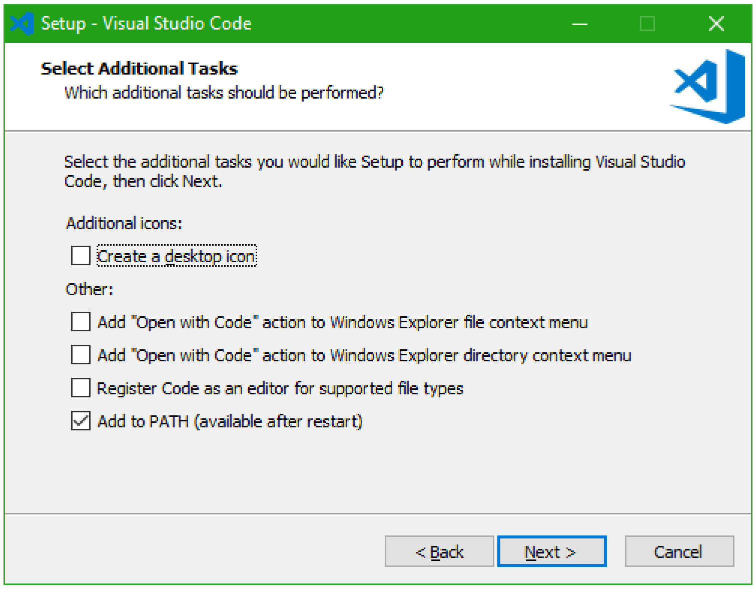
Then the License Agreement page will be appeared. Click on the *I accept the agreement* radio button and then hit *Next.*

Then a page that will ask and show the default path where Visual Studio Code will be installed. By default it is set to *C:\Program Files\Microsoft VS Code*.

Now hit *Next* to proceed further.

Now the page will ask to create a Start Menu folder entry for Visual Studio Code. According to the preferences, select a start menu entry for VS Code if needed.

After hitting the *Next* button, the next page will show some additional options as shown in the screenshot below that can be choosen. After completion, hit *Next.*



Select Additional Tasks
Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Visual Studio Code, then click Next.

Additional icons:

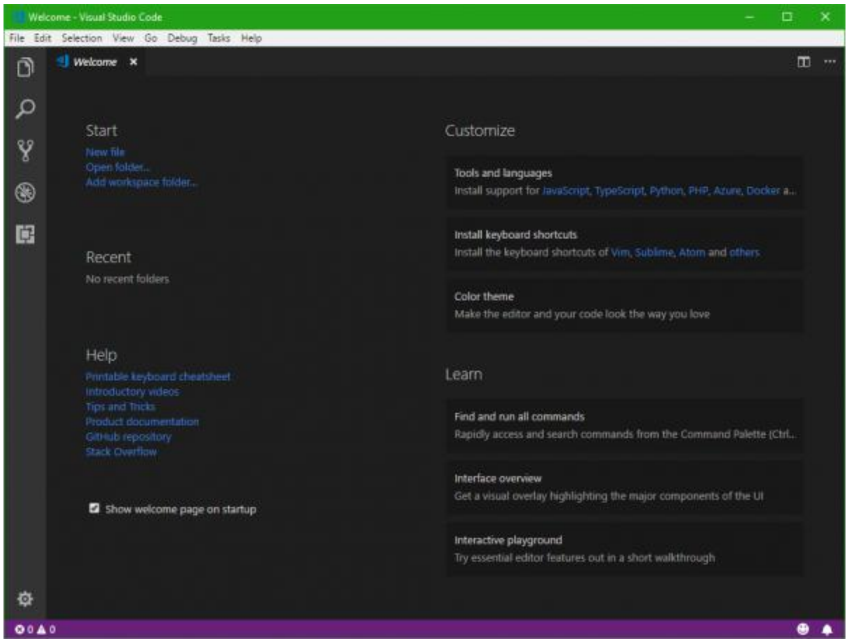
- □ Create a desktop icon

Other:

- □ Add "Open with Code" action to Windows Explorer file context menu
- □ Add "Open with Code" action to Windows Explorer directory context menu
- □ Register Code as an editor for supported file types
- ☑ Add to PATH (available after restart)

Then it will show a summary of what preference have to be selected to install VS Code, and according to that one can navigate either way and proceed with the installation by hitting the *Install* button.

Now, it will show an installation progress bar. After the installation is done, hit *Finish* to launch start using Visual Studio Code.


Welcome - Visual Studio Code
File Edit Selection View Go Debug Tasks Help
Welcome ×
0 0

Start
New file
Open folder...
Add workspace folder...

Recent
No recent folders

Help
Printable keyboard cheatsheet
Introductory videos
Tips and Tricks
Product documentation
GitHub repository
Stack Overflow

☑ Show welcome page on startup

Customize

Tools and languages
Install support for JavaScript, TypeScript, Python, PHP, Azure, Docker a...

Install keyboard shortcuts
Install the keyboard shortcuts of Vim, Sublime, Atom and others

Color theme
Make the editor and your code look the way you love

Learn

Find and run all commands
Rapidly access and search commands from the Command Palette (Ctrl...

Interface overview
Get a visual overlay highlighting the major components of the UI

Interactive playground
Try essential editor features out in a short walkthrough

Now, this is the home page of a fresh instance of Visual Studio Code. New projects can be created or use existing project folders. A wide variety of useful extensions are available for the IDE for you to be productive.

Installing Ionic

Ionic apps are created and developed primarily through the Ionic command-line utility. The Ionic CLI is the preferred method of installation, as it offers a wide range of dev tools and help options along the way. It is also the main tool through which to run the app and connect it to other services, such as Ionic Appflow.

Install the Ionic CLI

Before proceeding, make sure the latest version of Node.js and npm are installed. See Environment Setup for details. Install the Ionic CLI globally with npm:

```
$ npm install -g ionic
```

The `-g` means it is a global install. For Window’s it's recommended to open an Admin command prompt. For Mac/Linux, run the command with `sudo`.

Start an App

Create an Ionic app using one of the pre-made app templates, or a blank one to start fresh. The three most common starters are the `blank` starter, `tabs` starter, and `sidemenu` starter. Get started with the `ionic start` command:

```
$ ionic start cipa
```

Run the App

The majority of Ionic app development can be spent right in the browser using the `ionic serve` command:

```
$ cd myApp
$ ionic serve
```

There are a number of other ways to run an app, it's recommended to start with this workflow. To develop and test apps on devices and emulators, see the Running an App Guide.

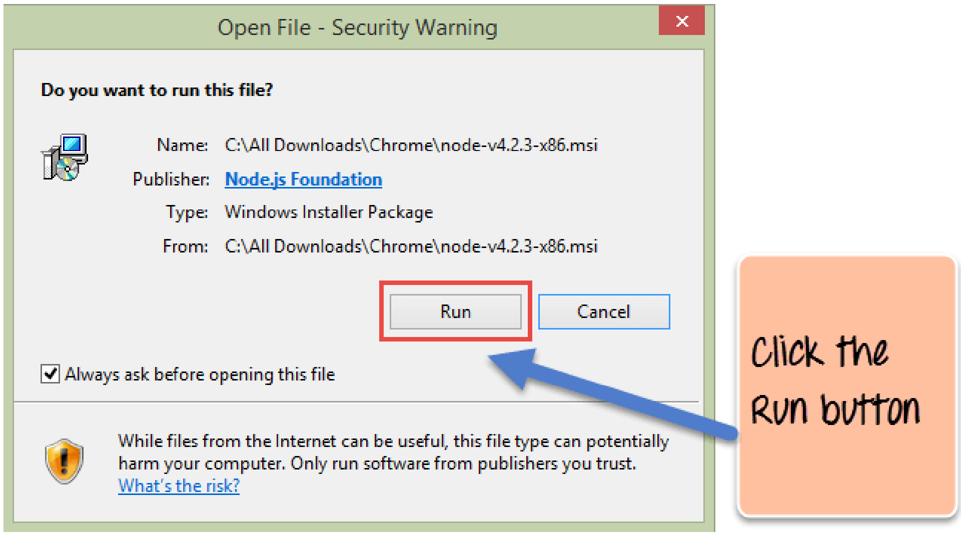
Installation of Node.js

To start building Node.js applications, the first step is the installation of the node.js framework. The Node.js framework is available for a variety of os right from Windows to Ubuntu and OS X. Once the Node.js framework is installed then start building the first Node.js applications.

The first steps in using Node.js is the installation of the Node.js libraries on the client system. To perform the installation of Node.js, perform the below steps;

Step 1) Go to the site https://nodejs.org/en/download/ and download the necessary binary files. In our example, we are going to the download the 64-bit setup files for Node.js.

Step 2) Double click on the downloaded .msi file to start the installation. Click the Run button in the first screen to begin the installation.



Open File - Security Warning

Do you want to run this file?

Name: C:\All Downloads\Chrome\node-v4.2.3-x86.msi

Publisher: Node.js Foundation

Type: Windows Installer Package

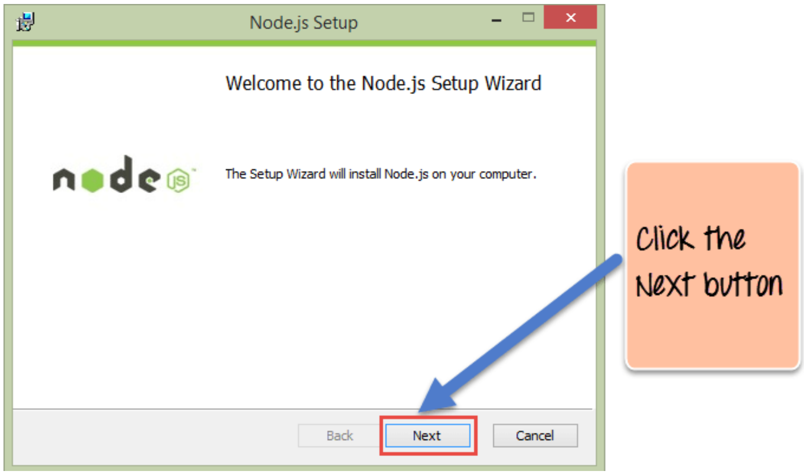
From: C:\All Downloads\Chrome\node-v4.2.3-x86.msi

☑ Always ask before opening this file

While files from the Internet can be useful, this file type can potentially harm your computer. Only run software from publishers you trust. What's the risk?

Click the Run button

Step 3) In the next screen, click the "Next" button to continue with the installation.



Welcome to the Node.js Setup Wizard

The Setup Wizard will install Node.js on your computer.

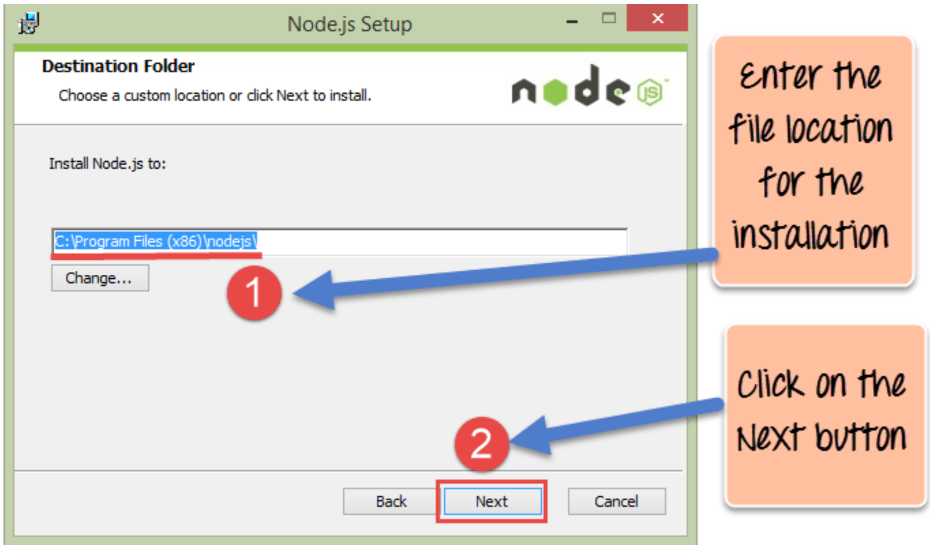
Click the Next button

Step 4) In the next screen Accept the license agreement and click on the Next button.

Step 5) In the next screen, choose the location where Node.js needs to be installed and then click on the Next button.

1.First enter the file location for the installation of Node.js. This is where the files for Node.js will be stored after the installation.

2.Click on the Next button to proceed ahead with the installation.



Destination Folder

Choose a custom location or click Next to install.

Install Node.js to:

C:\Program Files (x86)\nodejs\

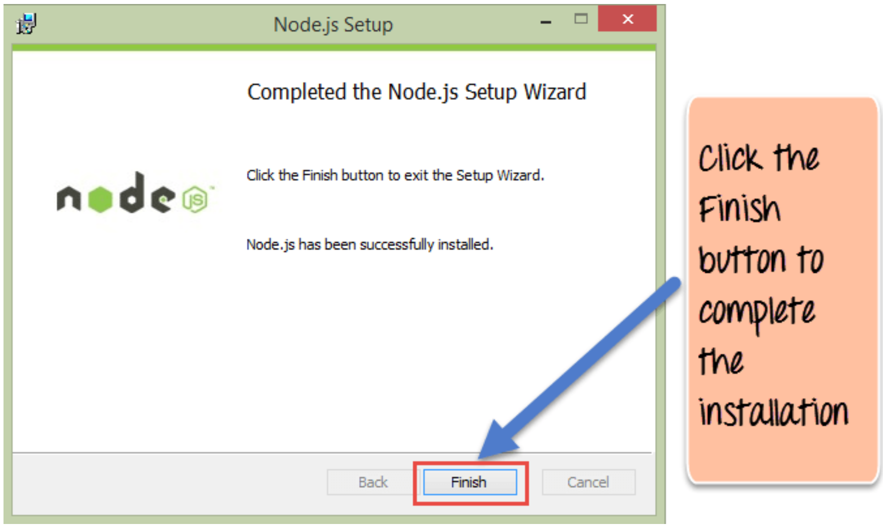
Enter the file location for the installation

Click on the Next button

Step 6) Accept the default components and click on the next button.

Step 7) In the next screen, click the Install button to start the installation.

Step 8) Click the Finish button to complete the installation.



Completed the Node.js Setup Wizard

Click the Finish button to exit the Setup Wizard.

Node.js has been successfully installed.

Click the Finish button to complete the installation

Step 10) *Verify That Node.js was Properly Installed*

To double check that Node.js was installed fully on your PC, now test the following command in the Command Prompt (regardless of if you're using `cmd.exe`, `Powershell`, or any other command prompt):

```
$ node -v
```

If Node.js was installed fully, the command prompt will print something similar to (but probably not *exactly*) this:

```
$ node -v // The command we ran - prints out the version of Node.js that's currently installed
v6.9.5 // The printed version of Node.js that's currently installed - v6.9.5 was the most current LTS release at the time of writing.
```

Step11) *Update the Local npm Version*

As the final step in getting Node.js installed, update your version of npm - the package manager that comes bundled with Node.js.

Node.js always ships with a specific version of npm - Node.js doesn't (and shouldn't!) automatically update npm. The release cycle of the npm CLI client isn't in sync with the Node.js releases. Due to this, there's almost *certainly* going to be a newer version of npm available than the one that is installed as a default in any given Node release.

To quickly and easily update `npm`, run the following command:

```
npm install npm --global // Update the `npm` CLI client
```

Installation Eclipse

Download eclipse from http://www.eclipse.org/downloads/. The download page lists a number of flavors of eclipse.

The capabilities of each packaging of eclipse are different. Java developers typically use Eclipse Classic or Eclipse IDE for developing Java applications.

The drop down box in the right corner of the download page allows you to set the operating system on which eclipse is to be installed. Choose between Windows, Linux and Mac. Eclipse is packaged as a zip file.

Installing Eclipse

To install on windows, a tool is needed that can extract the contents of a zip file.

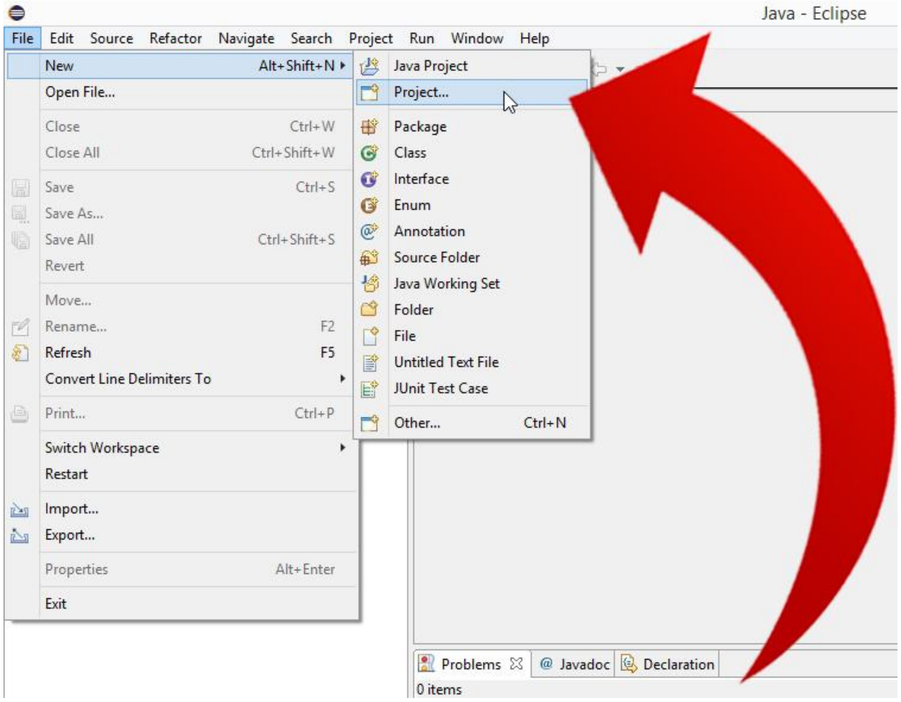
Using any one of these tools, extract the contents of the eclipse zip file to any folder.

Launching Eclipse

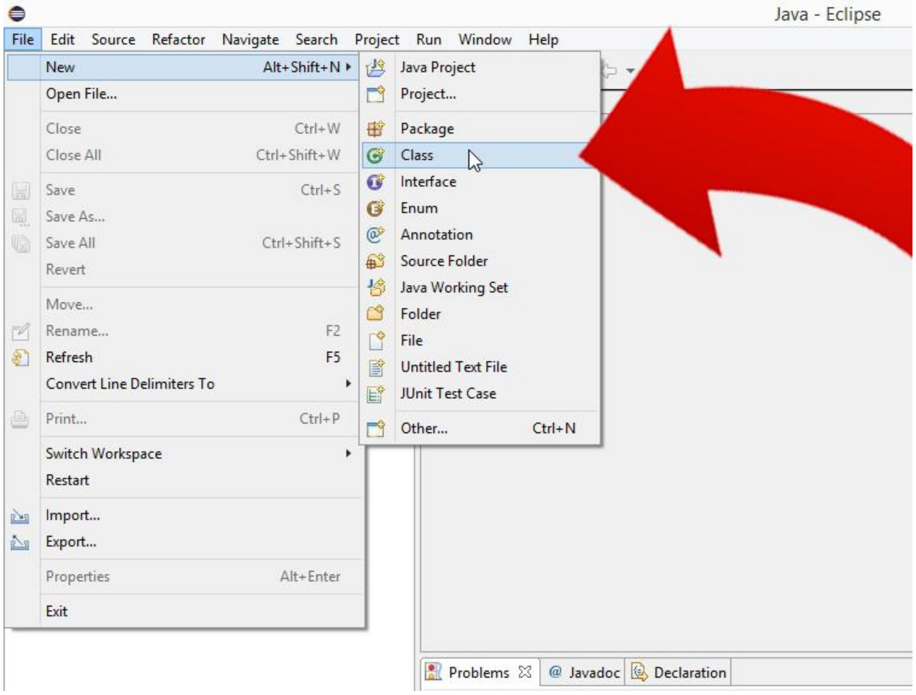
On the windows platform, if you extracted the contents of the zip file to c:\, then start eclipse by using c:\eclipse\eclipse.exe

When eclipse starts up for the first time it prompts for the location of the workspace folder. All the data will be stored in the workspace folder.

Create a New Java Project.



Create a new class with following File > New > Class.



Enter name of the class and press finish.

Install MySQL Workbench:

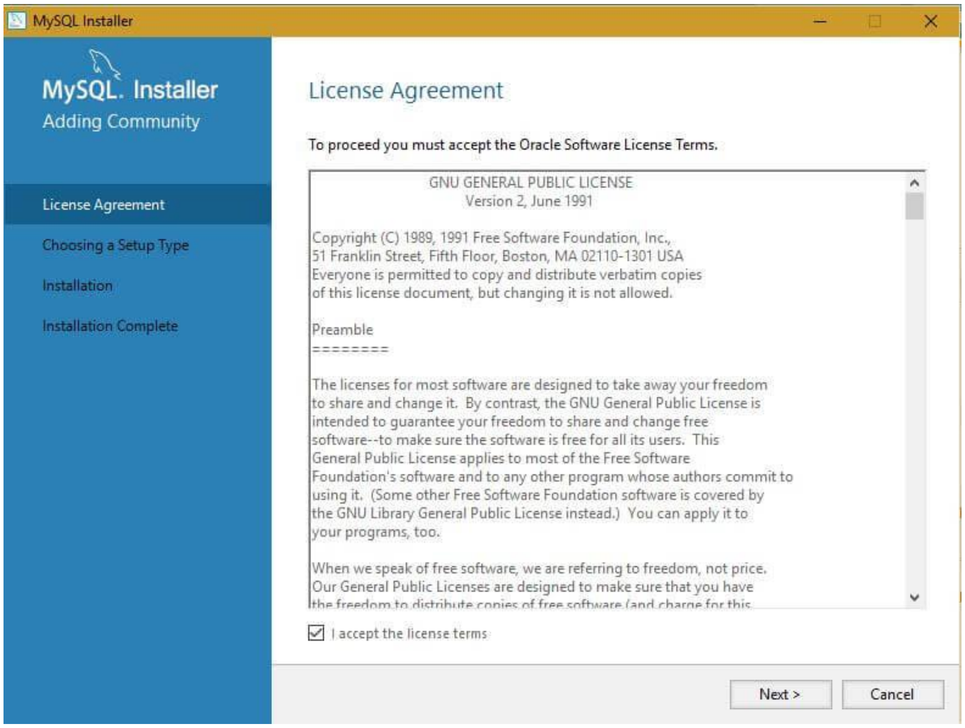
STEP 1: Download MySQL installer from below link and install it into pc.
http://dev.mysql.com/downloads/mysql/
After loading the above URL, a web page will be appeared. From this page click on the download link button to download MySQL installer.

STEP 2: Once download gets completed, observe a file with the below file name. It will be of around 320 MB. Double click on this file to initiate the installation of MySQL package. The package here means that the installer will install MySQL database server, MySQL Workbench, connectors and other software that we will see in later steps.

mysql-installer-community-5.7.13.0.msi

STEP 3: MySQL installer will start configuring the file to prepare the setup for installation of the MySQL package.

STEP 4: Next, it will open up a dialogue box to accept the license. Check the checkbox and click on the 'Next' button.



License Agreement

To proceed you must accept the Oracle Software License Terms.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

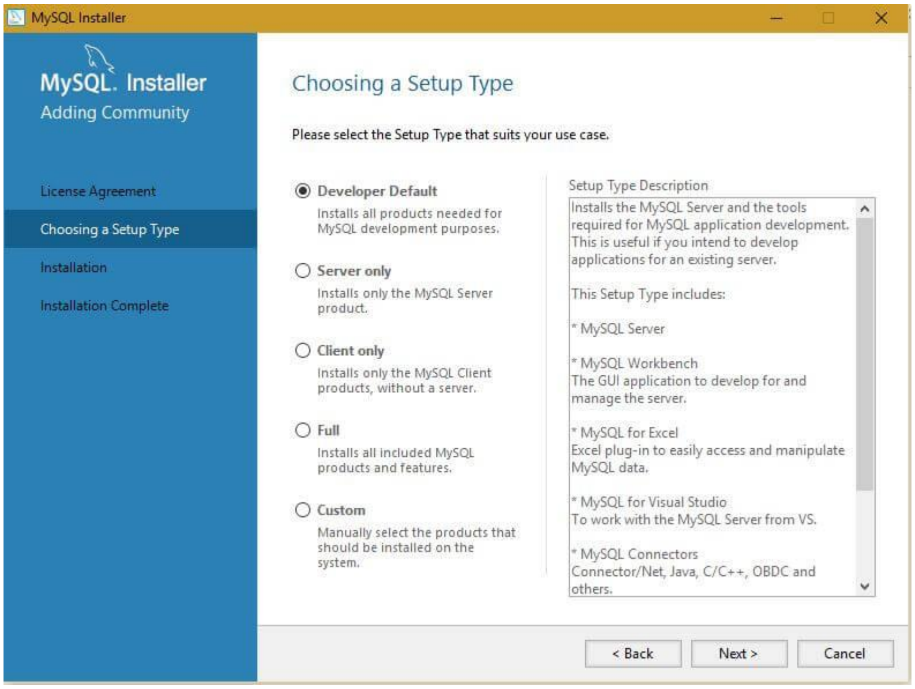
Preamble
========

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this

☑ I accept the license terms

STEP 5: Choose a setup as ‘Developer Default’ and click on the ‘Next’ button as shown below.



Choosing a Setup Type

Please select the Setup Type that suits your use case.

● Developer Default
Installs all products needed for MySQL development purposes.

○ Server only
Installs only the MySQL Server product.

○ Client only
Installs only the MySQL Client products, without a server.

○ Full
Installs all included MySQL products and features.

○ Custom
Manually select the products that should be installed on the system.

Setup Type Description

Installs the MySQL Server and the tools required for MySQL application development. This is useful if you intend to develop applications for an existing server.

This Setup Type includes:

* MySQL Server

* MySQL Workbench
The GUI application to develop for and manage the server.

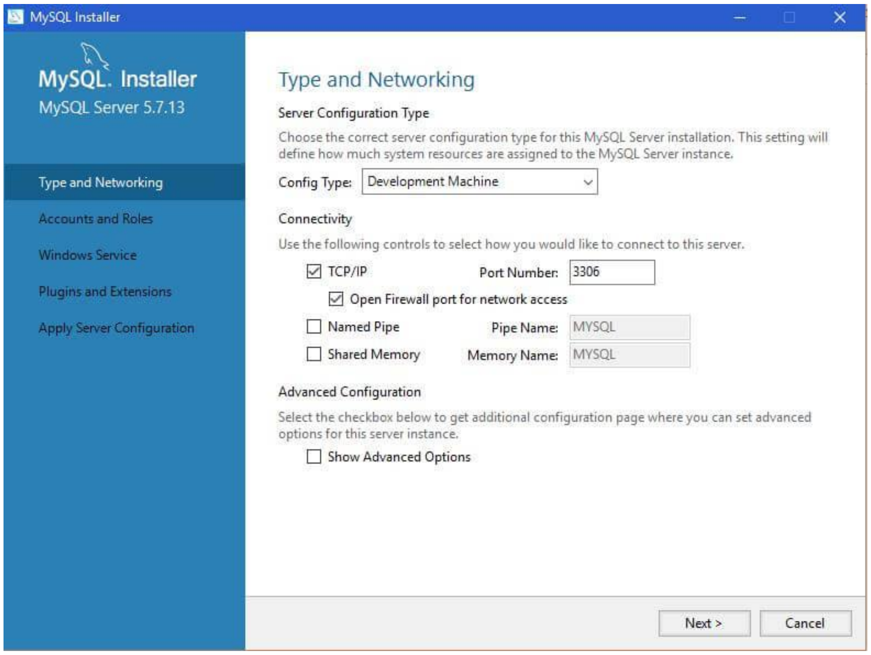
* MySQL for Excel
Excel plug-in to easily access and manipulate MySQL data.

* MySQL for Visual Studio
To work with the MySQL Server from VS.

* MySQL Connectors
Connector/Net, Java, C/C++, OBDC and others.

STEP 6: Click on the ‘Execute’ button present at the bottom of the dialogue box. This will kick-start the installation of the software items present in the MySQL package.

STEP 7: After installation of the product, the system will ask to choose the Config type and other connectivity directives. Here, select Config type as ‘Development Machine’, Connectivity as TCP/IP and Port number is 3306 as shown below.



Type and Networking

Server Configuration Type

Choose the correct server configuration type for this MySQL Server installation. This setting will define how much system resources are assigned to the MySQL Server instance.

Config Type: Development Machine

Connectivity

Use the following controls to select how you would like to connect to this server.

☑ TCP/IP Port Number: 3306

☑ Open Firewall port for network access

☐ Named Pipe Pipe Name: MYSQL

☐ Shared Memory Memory Name: MYSQL

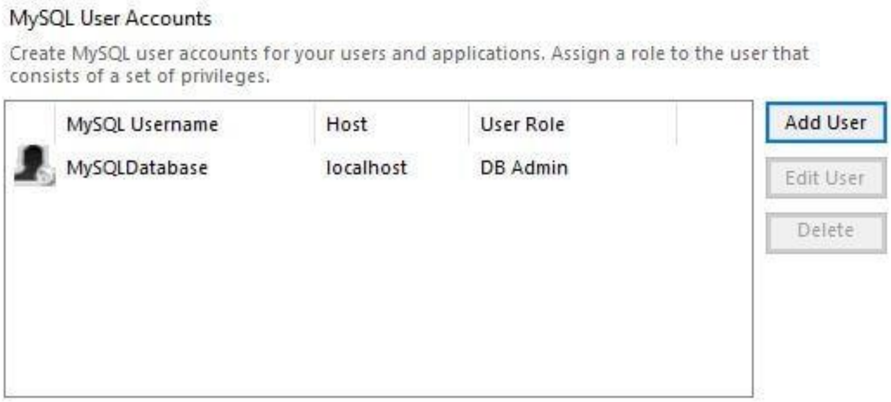
Advanced Configuration

Select the checkbox below to get additional configuration page where you can set advanced options for this server instance.

☐ Show Advanced Options

STEP 8: Next step it will ask to choose the account, username and password as shown below. Here, we are entering username as ‘Root’, host as ‘localhost’, Role as ‘Admin’ and password as ‘12345’. As shown below. After entering these details click on the OK button.

STEP 9: Once the account is created successfully, we can see the detail in current dialogue box.



MySQL User Accounts

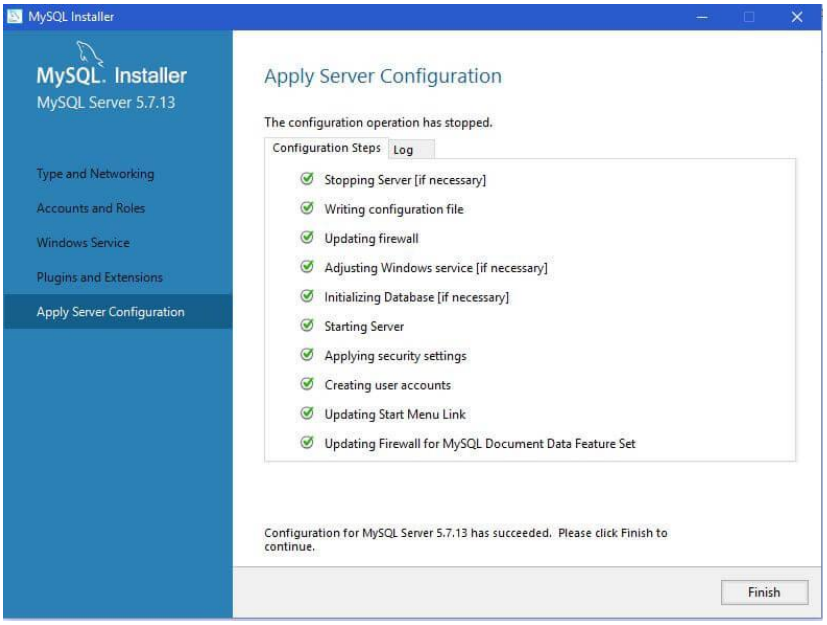
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL Username	Host	User Role
MySQLDatabase	localhost	DB Admin

STEP 10: Next, the system will ask to configure Windows Service. Just keep the default setup and click on the ‘Next’ button.

STEP 11: Next, the system will ask to configure *Plugins and Extensions*. Just keep the default setup and click on the ‘Next’ button.

STEP 12: Next, the system will ask to apply server configuration. Just click on the ‘Execute’ button, the below screen once the configuration has completed. Click on the ‘Finish’ button to complete the MySQL package installation.



Apply Server Configuration

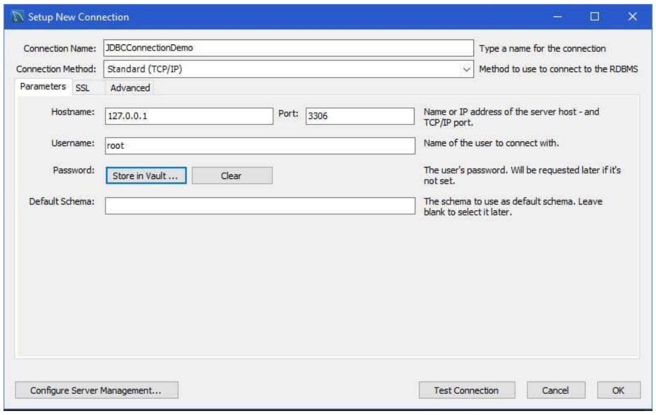
The configuration operation has stopped.

- ✓ Stopping Server [if necessary]
- ✓ Writing configuration file
- ✓ Updating firewall
- ✓ Adjusting Windows service [if necessary]
- ✓ Initializing Database [if necessary]
- ✓ Starting Server
- ✓ Applying security settings
- ✓ Creating user accounts
- ✓ Updating Start Menu Link
- ✓ Updating Firewall for MySQL Document Data Feature Set

Configuration for MySQL Server 5.7.13 has succeeded. Please click Finish to continue.

STEP 13: Next, the system will prompt to check the connectivity of the database. Just click on the ‘Check’ button to test the connectivity with the username as ‘root’ and password as ‘12345’. It will display ‘connection successful’.

STEP 14: Open the MySQL Workbench and create a connection as shown below. Once entering below details, feel free to click on the *TestConnection* button to test the database connectivity.



Connection Name: JDBCConnectionDemo — Type a name for the connection

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

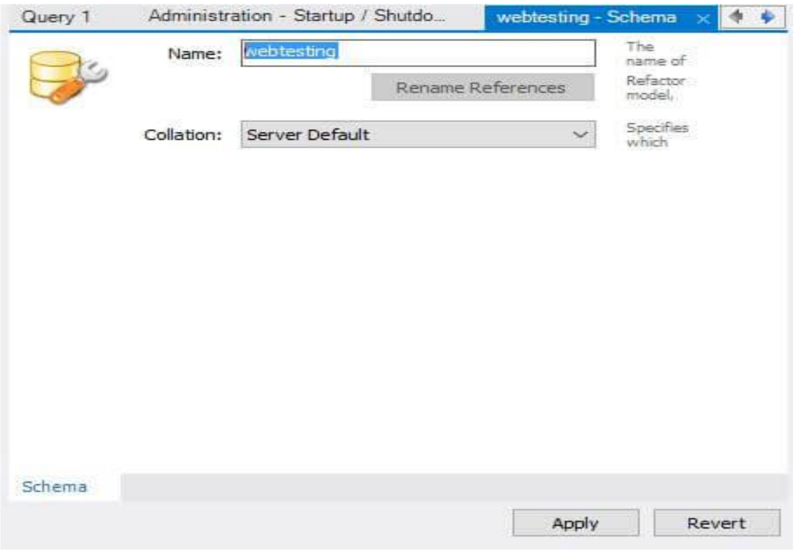
Hostname: 127.0.0.1 Port: 3306 — Name or IP address of the server host - and TCP/IP port.

Username: root — Name of the user to connect with.

Password: — The user's password. Will be requested later if it's not set.

Default Schema: — The schema to use as default schema. Leave blank to select it later.

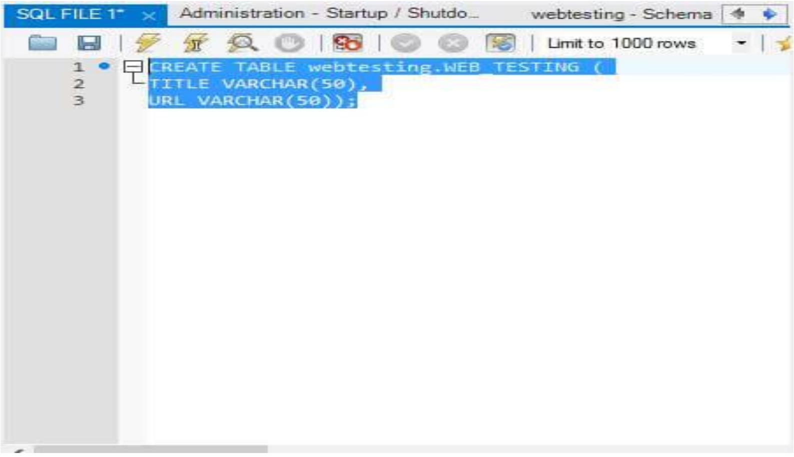
STEP 15: Create a new scheme 'webtesting' from the MySQL workbench as shown below. The schema is the database region where all tables are located. After entering the schema name and choosing the 'Server Default', click on the Apply button to complete the creation of schema.



Name: webtesting

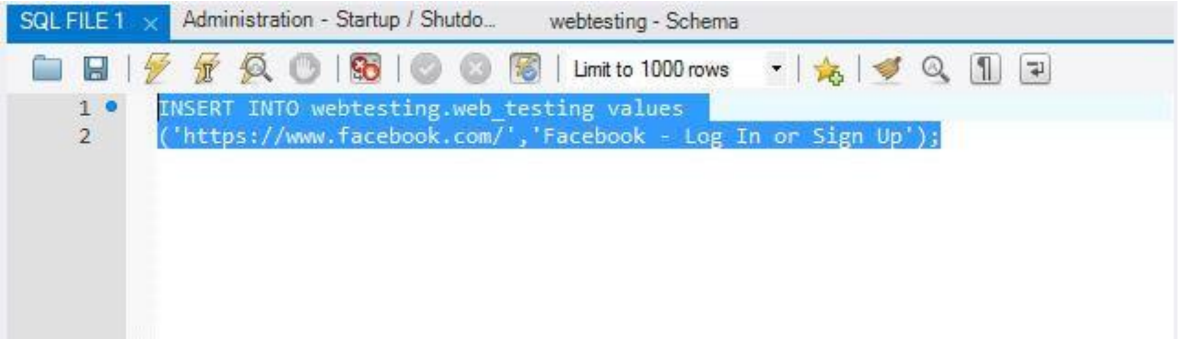
Collation: Server Default

STEP 16: Now create a table WEB_TESTING in the webtesting schema by executing the below script from MySQL workbench. This table has two columns, the first column stores the website URL and the second column stores the home page title.



```
CREATE TABLE webtesting.WEB_TESTING (
TITLE VARCHAR(50),
URL VARCHAR(50));
```

STEP 18: Insert the records into the table after executing the below DB script.



```
INSERT INTO webtesting.web_testing values
('https://www.facebook.com/','Facebook - Log In or Sign Up');
```

MYSQL WORKBENCH:

MYSQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system. It is the successor to DBDesigner 4 from fabFORCE.net, and replaces the previous package of software, MySQL GUI Tools Bundle.

MySQL Workbench 5.2

Starting with MySQL Workbench 5.2 the application has evolved to a general database GUI application. Apart from physical database modeling it features an SQL Editor, database migration tools, and a database server administration interface, replacing the old MySQL GUI Tools Bundle.

Features:

- Prominent features of MySQL Workbench are:
- General
  - Database Connection & Instance Management
  - Wizard driven action items
- SQL Editor
  - Schema object browsing, inspection, and search
  - SQL syntax highlighter and statement parser

- Data modeling
    - ER diagramming
    - Drag'n'Drop visual modeling
- Database administration
    - Start and stop of database instances
    - Instance configuration

4.2 SAMPLE CODE

admin-login.html

```
<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-back-button></ion-back-button>
    </ion-buttons>
    <ion-title>AdminLogin</ion-title>
  </ion-toolbar>
</ion-header>
<ion-content padding>
  <ion-input placeholder="Enter Username"></ion-input>
  <ion-input placeholder="Enter Password"></ion-input>
  <ion-button color="primary">Login</ion-button>
</ion-content>
```

Adminlogin.ts

```
import { Component, OnInit } from '@angular/core';
@Component({  selector: 'app-admin-login',templateUrl: './admin-login.page.html',
  styleUrls: ['./admin-login.page.scss'],})
export class AdminLoginPage implements OnInit {
  constructor() {
}

  ngOnInit() {

  }

}
```

userlogin.html

```
<ion-header>

  <ion-toolbar>

    <ion-buttons slot="start">

      <ion-back-button></ion-back-button>

    </ion-buttons>

    <ion-title>UserLogin</ion-title>  </ion-toolbar>

</ion-header>

<ion-content padding>

  <ion-input [(ngModel)]="username" placeholder="Enter Username"></ion-input>

  <ion-input  [(ngModel)]="password" placeholder="Enter Password"></ion-input>

  <ion-button (click)="login()" color="primary">Login</ion-button>

  <ion-button (click)="Registration()" color="primary">Sign up</ion-button> </ion-content>
```

Userlogin.ts

```
import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import { AuthService } from '../_services/auth-service';

import { Common } from '../_services/common';
```

```
@Component({

  selector: 'app-user-login',

  templateUrl: './user-login.page.html',

  styleUrls: ['./user-login.page.scss'],

})

export class UserLoginPage implements OnInit {

  password: any;

  username: any;

  resposeData:any= {};

  constructor(private authservice : AuthService,private common: Common,private router:Router)
{

  }

  Registration()

  {

this.router.navigate(['registration']);

  }

  ngOnInit() {

  }

  login(){

     this.common.presentLoading();

        if(this.password)

        {

            this.authservice.postData(", "Login?user="+this.username+"&pass="+this.password
                ).then((result) =>{
        //console.log("hi how r u i nthe");
        this.resposeData = result;
       //console.log("this is the"+this.resposeData.id);
       if(this.resposeData.status=="200"){
        this.router.navigate(['selection']);
        this.common.closeLoading();
       }
     else{
      alert("invalid credentials");
       this.common.closeLoading();
     }
      }, (err) => {
        this.common.closeLoading();
       });
      }
      else{
       this.common.closeLoading();}
}
}
}
```

5.TESTING

Software testing is a process of executing a program or application with the intent of finding the software bugs. It can also be stated as the process of validating and verifying that a software program or application or product: Meets the business and technical requirements that guided it's design and development. To evaluate the functionality of a software application with an intent to find whether the developed software met the specified requirements or not and to identify the defects to ensure that the product is defect free in order to produce the quality product.

Testing Levels:

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing

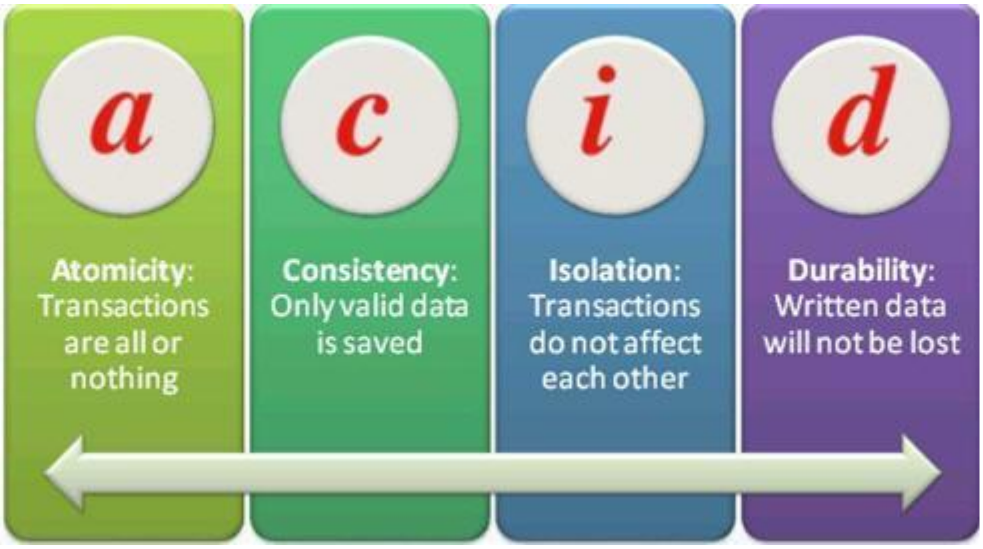
Unit Testing: Unit Testing is done to check whether the individual modules of the source code are working properly. i.e. testing each and every unit of the application separately by the developer in the developer's environment. It is AKA Module Testing or Component Testing

Integration Testing: Integration Testing is the process of testing the connectivity or data transfer between a couple of unit tested modules. It is AKA I&T Testing or String Testing. It is subdivided into Top-Down Approach, Bottom-Up Approach and Sandwich Approach (Combination of Top Down and Bottom Up).

System Testing (end to end testing): It's a black box testing. Testing the fully integrated application this is also called as end to end scenario testing. To ensure that the software works in all intended target systems. Verify thorough testing of every input in the application to check for desired outputs. Testing of the users experiences with the application.

Acceptance Testing: To obtain customer sign–off so that software can be delivered and payments received. Types of acceptance testing are alpha, beta & gamma testing.

DATABASE TESTING:


a
c
i
d
Atomicity: Transactions are all or nothing
Consistency: Only valid data is saved
Isolation: Transactions do not affect each other
Durability: Written data will not be lost

- **Atomicity** means that a transaction either fails or passes. This means that even if a single part of the transaction fails- it means that the entire transaction has failed. Usually, this is called the “all-or-nothing” rule.
- **Consistency**: A transaction will always result in a valid state of the DB
- **Isolation**: If there are multiple transactions and they are executed all at once, the result/state of the DB should be the same as if they were executed one after the other.
- **Durability**: Once a transaction is done and committed, no external factors like power loss or crash should be able to change it

5.2 Test cases

Testcase-login:

Test case ID	Input	Description	Expected Results
CIPA_01	Username blank Password Blank	A blank user name and password given	Please enter user name and password message will be displayed
CIPA_02	Valid user name Password blank	A valid user name and a blank password given	Please enter password
CIPA_03	Blank user name Password valid	A blank user name and a valid password given	Invalid user name
CIPA_04	Valid user name Valid password	A valid user name and password given	Redirected to next page
CIPA_05	Invalid user name Invalid password	A invalid user name and password given	Invalid credentials message will be displayed

Table 5.1 Test cases for logins

Testcase-register:

Test case ID	Input	Description	Expected Results
CIPA_01	Any one field in the form is left blank	One or more fields in the form were not entered	Please enter the particular field input message will be displayed
CIPA_02	Entered all fields correctly	All the fields are entered correctly by the user	Registration successful message will be displayed
CIPA_03	User name already exists	If the particular user name has already exists in the database	User name already exists message will be displayed
CIPA_04	Password is not greater than 5	Entered password is less than 5 characters	Password should be greater than 5 message will be displayed
CIPA_05	Phone number is not equal to 10 digits	Respective phone number less/greater than contains 10 digits	Phone number should be 10 digits message will be displayed

Table 5.2 Test cases for Registration

Testcase-selection:

Test case ID	Input	Description	Expected Results
CIPA_01	Service number blank	A service number given	Invalid Service number
CIPA_02	Valid service number Seat number greater than 60	A valid service number and seat number greater than 60 given	Seat number should be less than 60 message will be displayed
CIPA_03	Valid servicenumber Seat number less than 60	A valid service number and seat number less than 60 given	Source and destinations will be displayed

Table 5.3 Test cases for Seat number and Service number

Testcase-food items:

Test case ID	Input	Description	Expected Results
CIPA_01	Cart empty	None of the items were selected by the user	Please select one message will be displayed
CIPA_02	One or more items in the cart	One or more items have been selected by the user	Do you want to confirm order message will be displayed

Table 5.4 Test cases for Checking the Cart.

Testcase-confirm order:

Test case ID	Input	Description	Expected Results
CIPA_01	Order confirmed	The order has been confirmed by the user	Booking successful message will be displayed
CIPA_02	Order not confirmed	The order has been cancelled by the user	Redirect to login page

Table 5.5 Test cases for Confirmation.

6.SCREENS AND REPORTS

6.1. SCREEN SHOTS

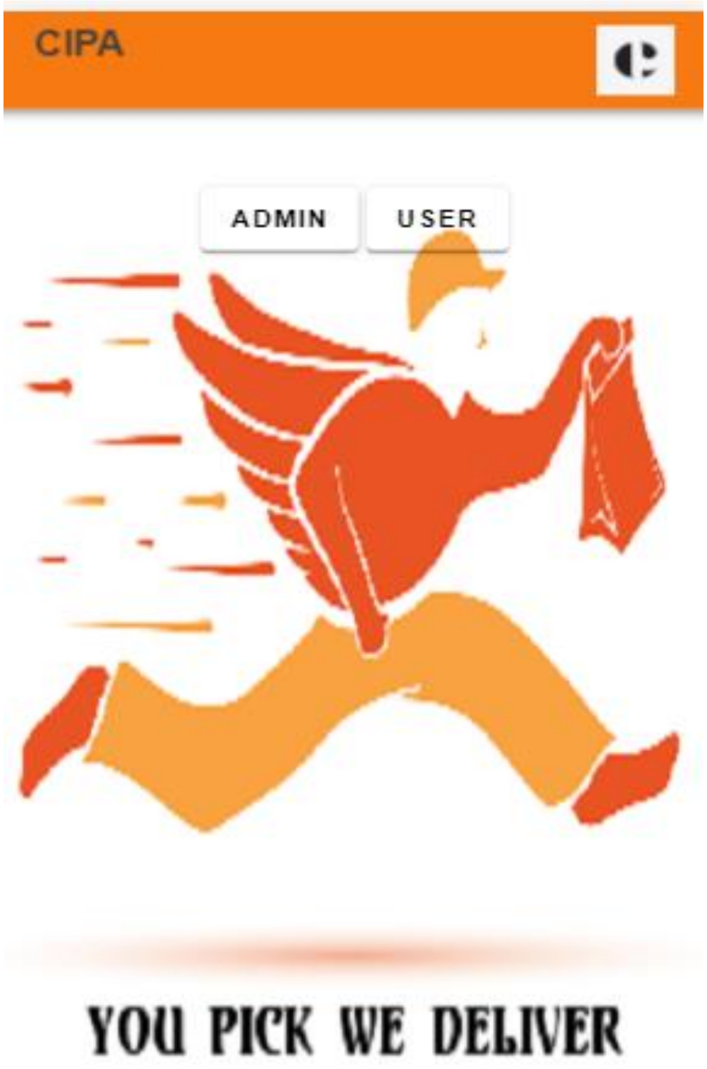


Figure 6.1 Home Page

Represents the landing page of the application.

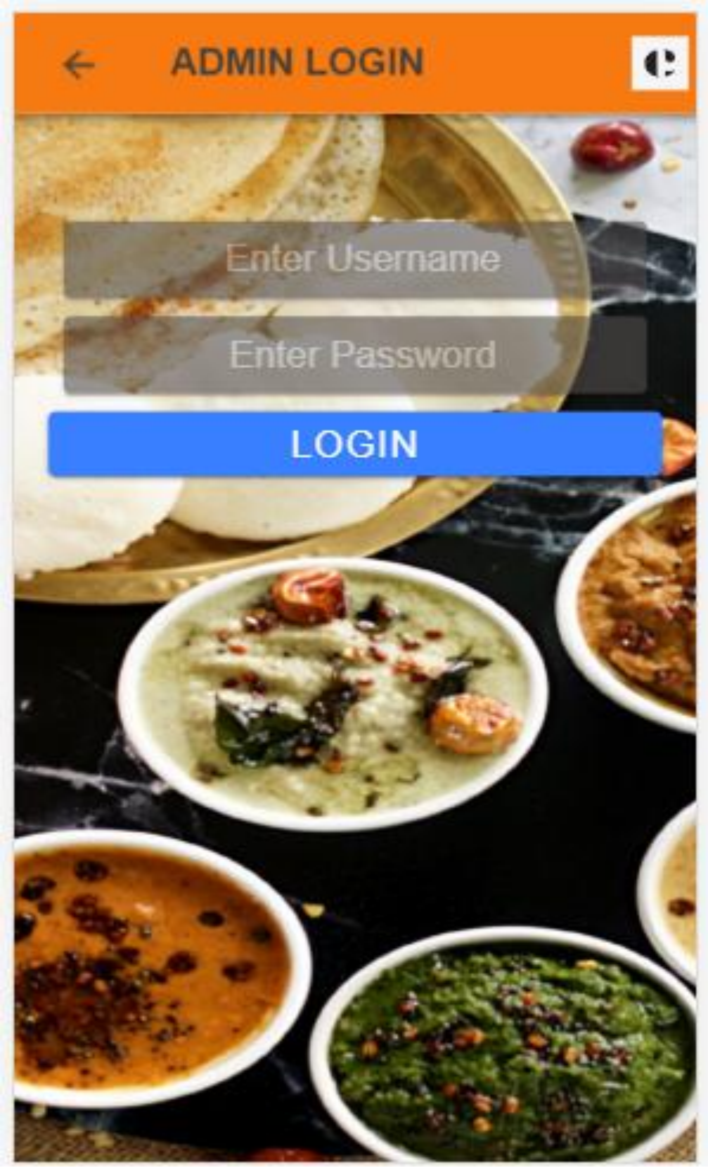
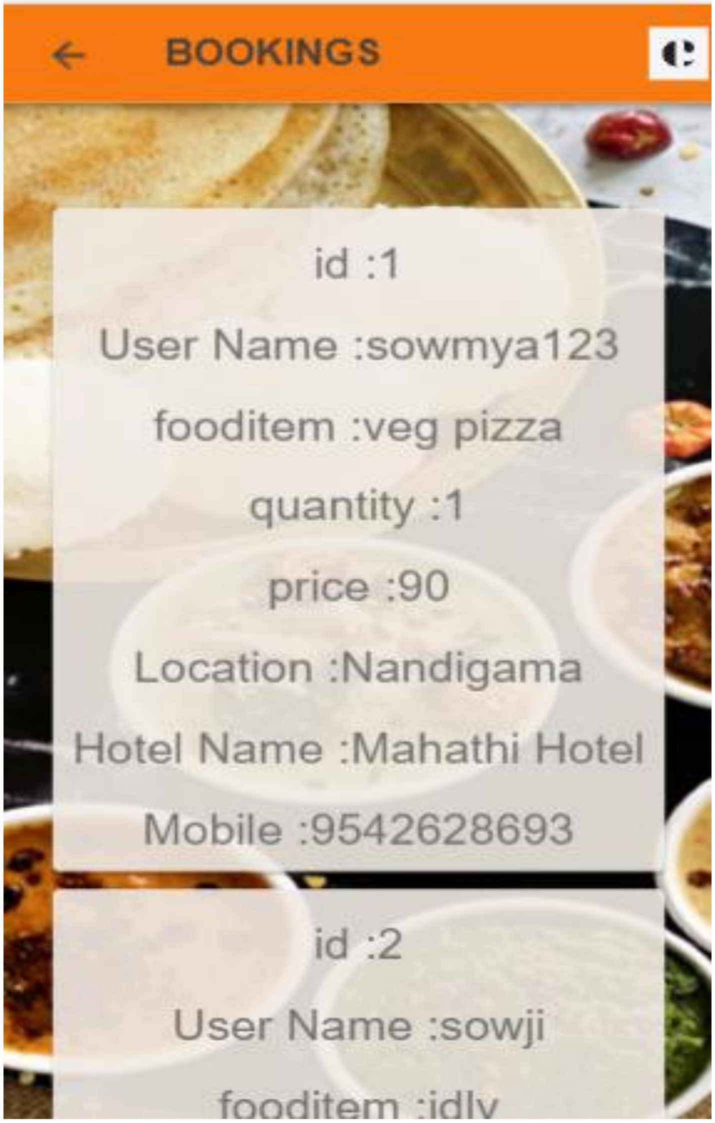


Figure 6.2 Admin Login Page

Used by the admin to get log-on to the application.



id :1

User Name :sowmya123

fooditem :veg pizza

quantity :1

price :90

Location :Nandigama

Hotel Name :Mahathi Hotel

Mobile :9542628693

id :2

User Name :sowji

fooditem :idly

Figure 6.3 Bookings Page

Helps the admin to view the booking history.

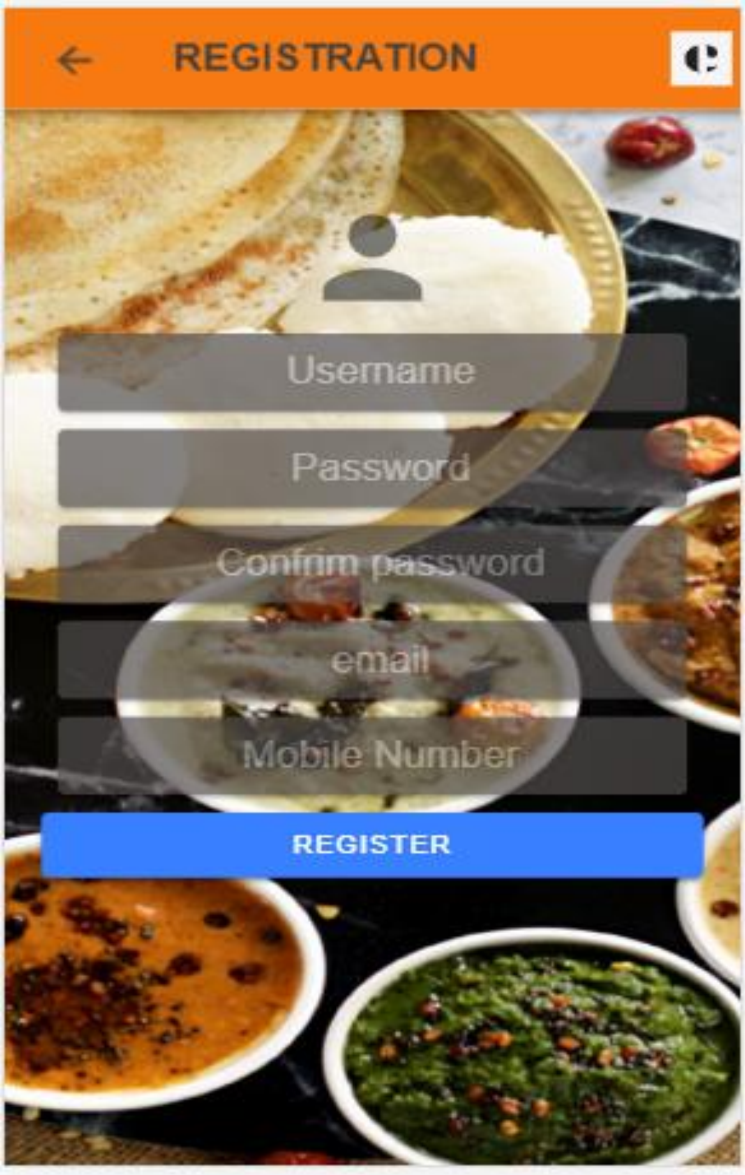
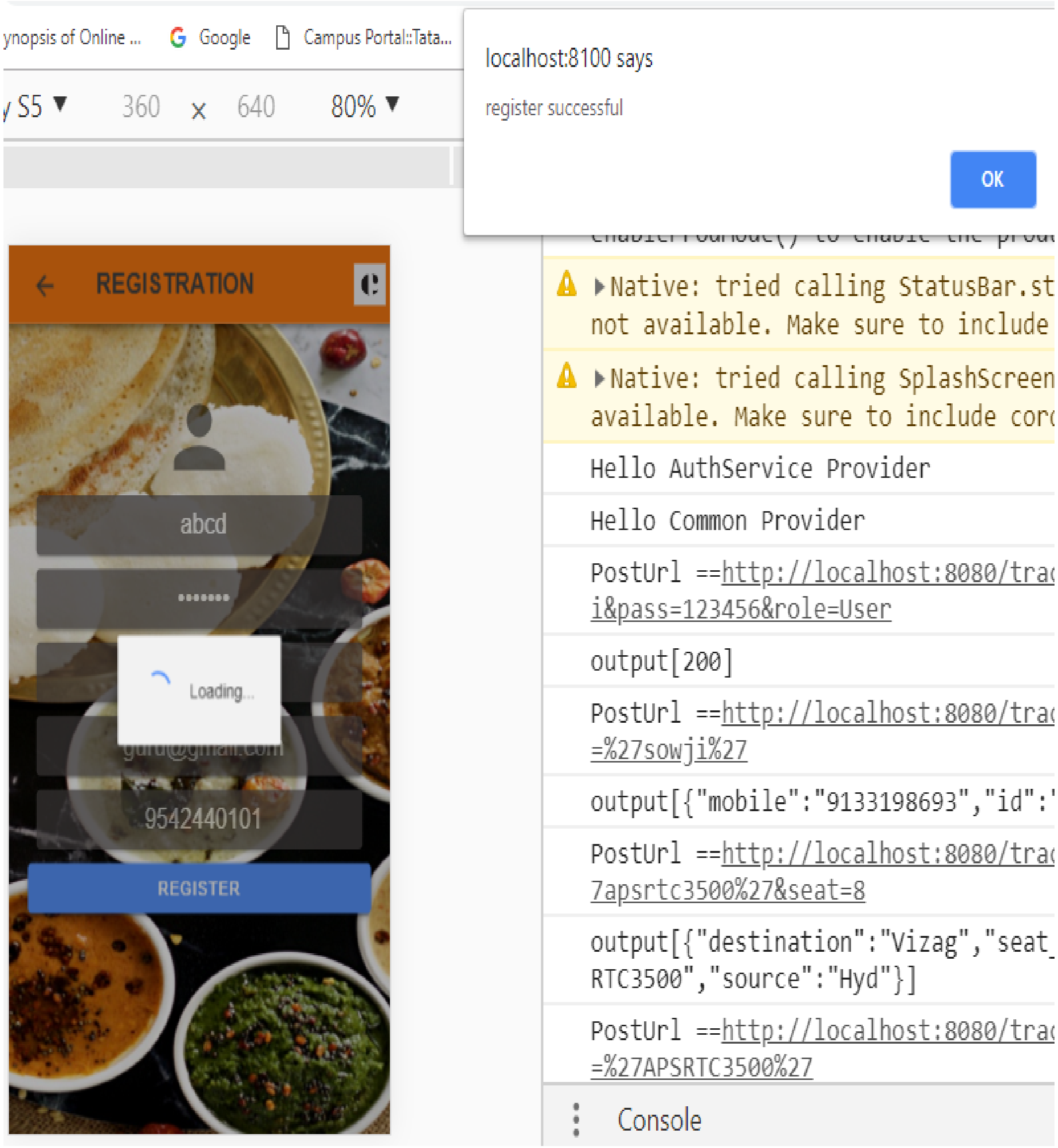


Figure 6.4 Registration Page

If user has no account then the registeration page is appeared.



localhost:8100 says

register successful

OK

REGISTRATION

abcd

•••••••

Loading...

guru@gmail.com

9542440101

REGISTER

```
Native: tried calling StatusBar.st
not available. Make sure to include
Native: tried calling SplashScreen
available. Make sure to include cor
Hello AuthService Provider
Hello Common Provider
PostUrl ==http://localhost:8080/tra
i&pass=123456&role=User
output[200]
PostUrl ==http://localhost:8080/tra
=%27sowji%27
output[{"mobile":"9133198693","id":'
PostUrl ==http://localhost:8080/tra
7apsrtc3500%27&seat=8
output[{"destination":"Vizag","seat_
RTC3500","source":"Hyd"}]
PostUrl ==http://localhost:8080/tra
=%27APSRTC3500%27
```

Console

Figure 6.5 Registration Page

This window is appeared when the registration is successful

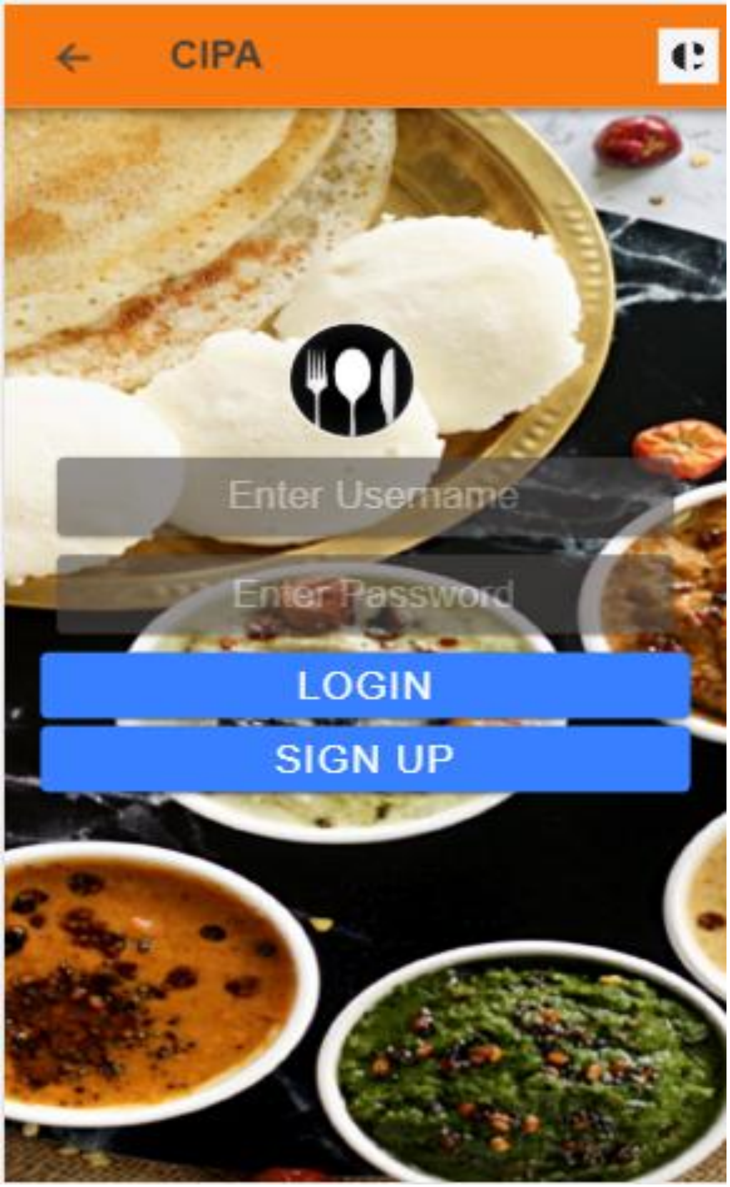
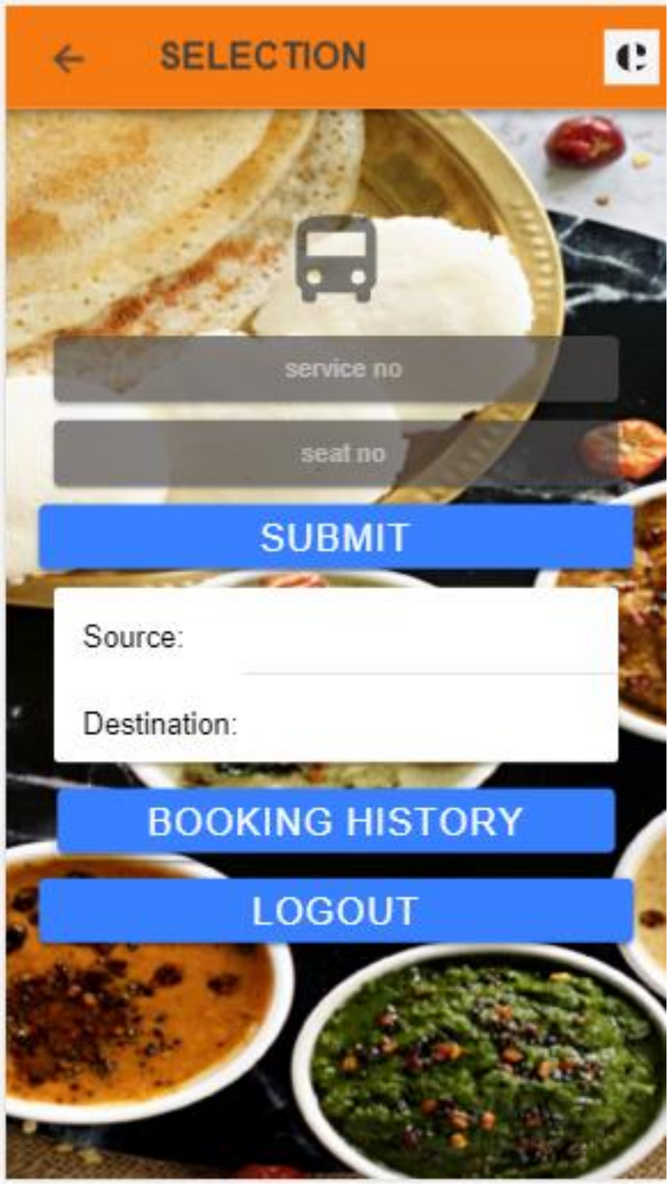


Figure 6.6 User Login Page

Allows user to enter the application.

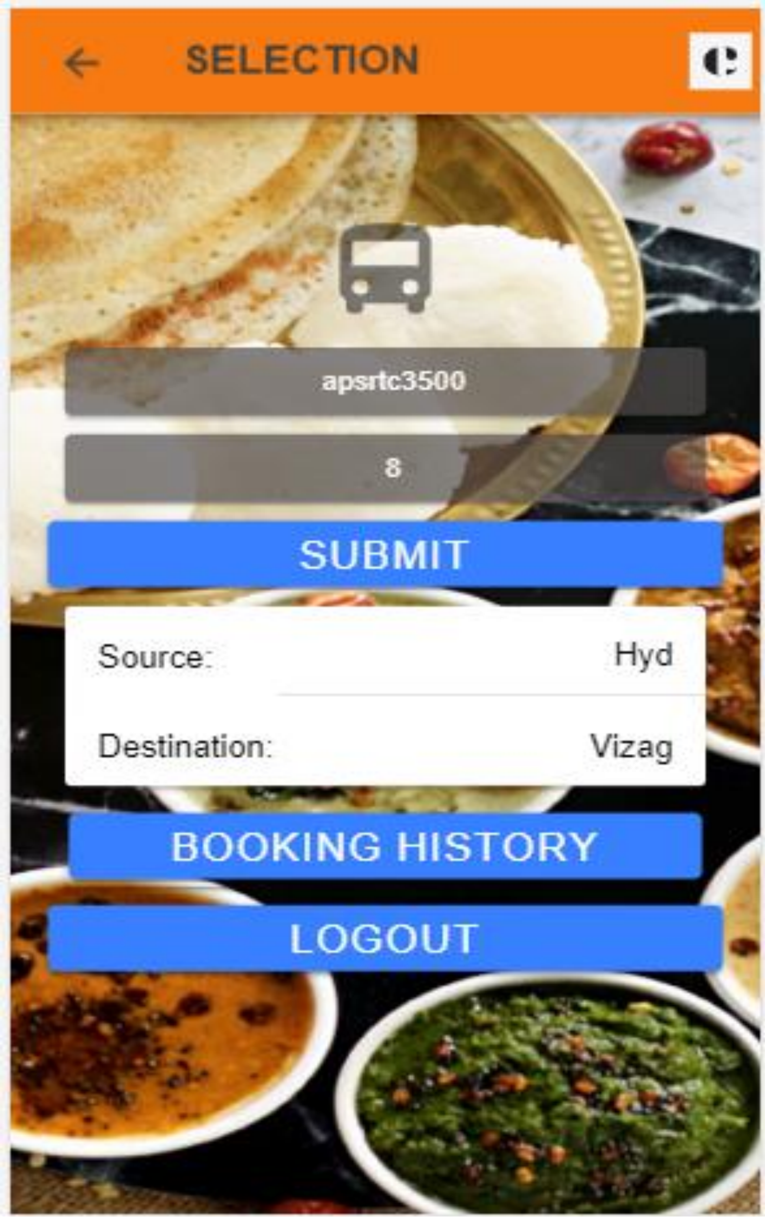


Source:

Destination:

Figure 6.7 Selection Page

The user has to provide service number in order to go further.

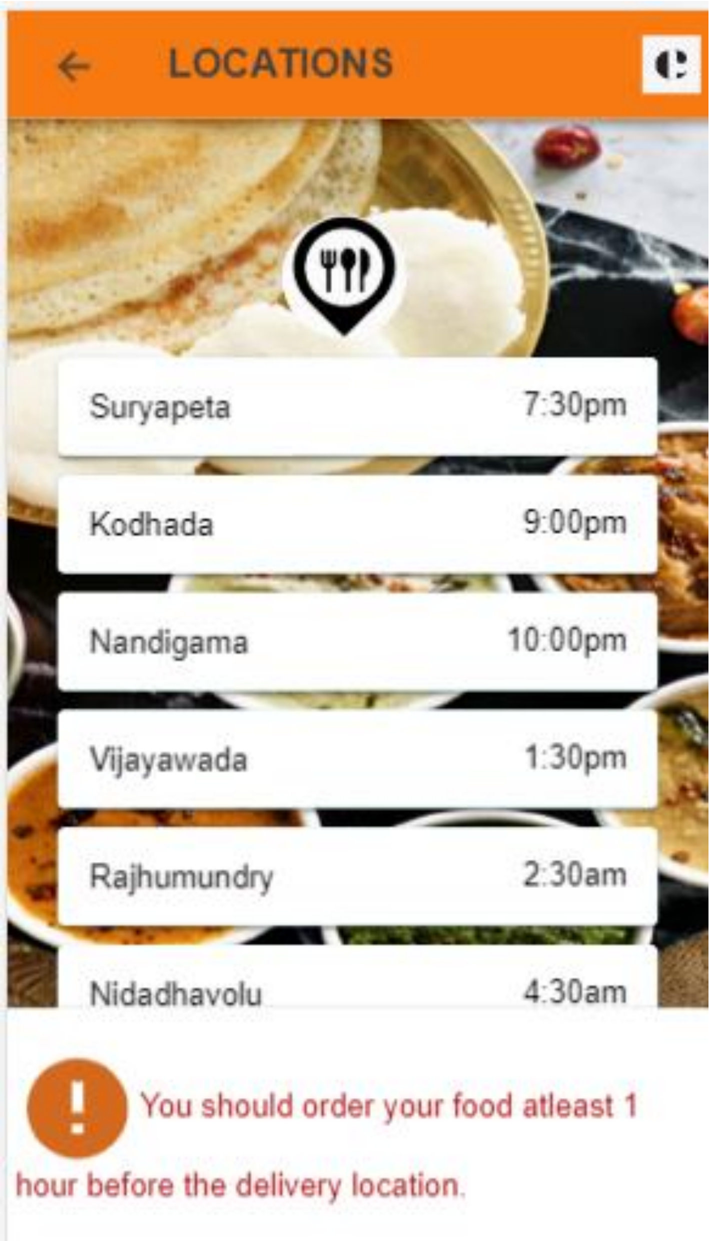


Source: Hyd

Destination: Vizag

Figure 6.8 Selection Page

This window displays the source and destination.



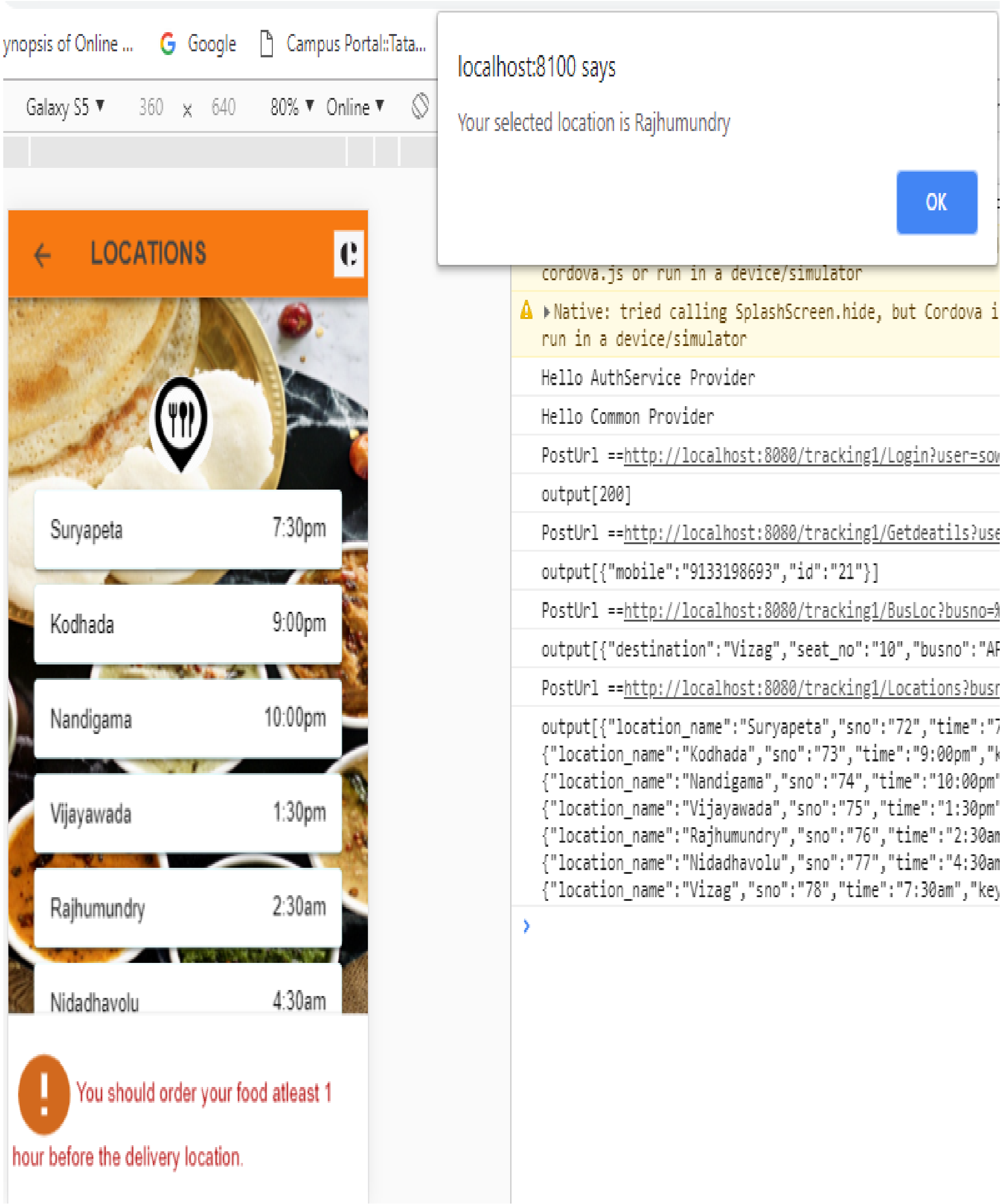
LOCATIONS

Suryapeta	7:30pm
Kodhada	9:00pm
Nandigama	10:00pm
Vijayawada	1:30pm
Rajhumundry	2:30am
Nidadhavolu	4:30am

! You should order your food atleast 1 hour before the delivery location.

Figure 6.9 Locations Page

Displays all the predefined locations to the user.



localhost:8100 says

Your selected location is Rajhumundry

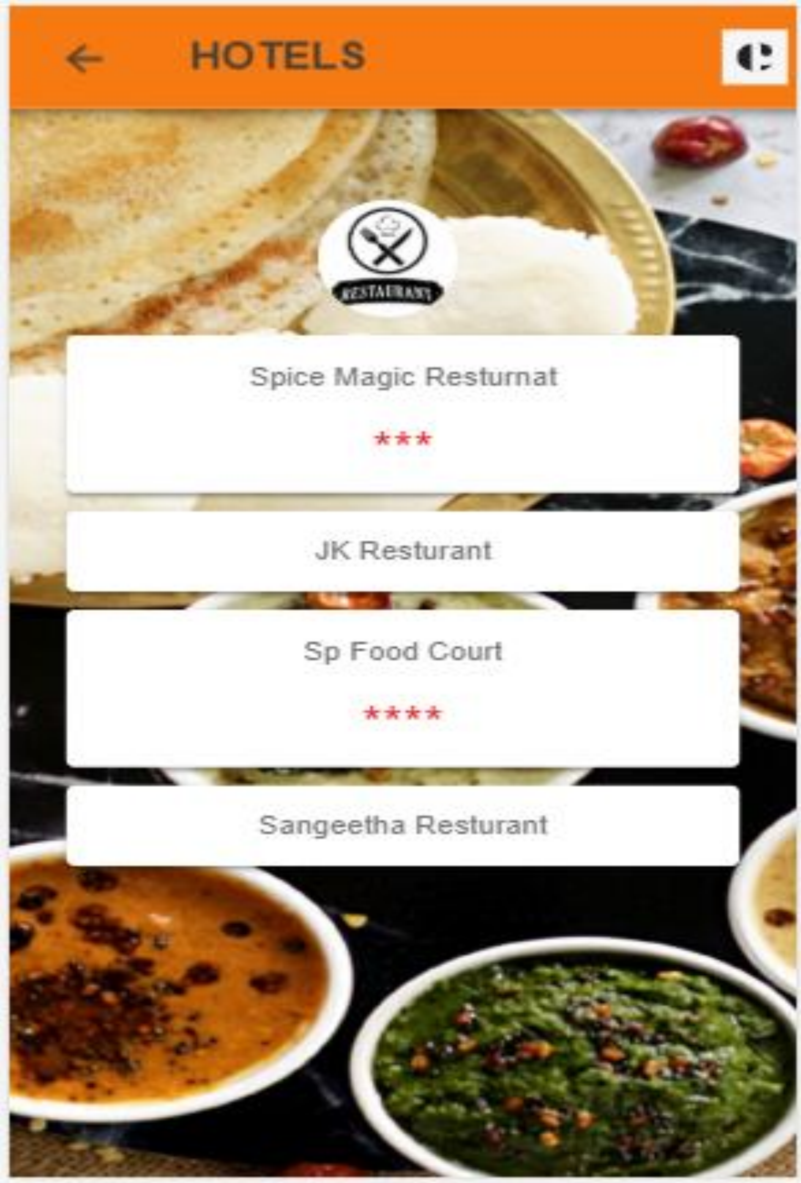
Suryapeta	7:30pm
Kodhada	9:00pm
Nandigama	10:00pm
Vijayawada	1:30pm
Rajhumundry	2:30am
Nidadhavolu	4:30am

You should order your food atleast 1 hour before the delivery location.

```
cordova.js or run in a device/simulator
▸Native: tried calling SplashScreen.hide, but Cordova i
run in a device/simulator
Hello AuthService Provider
Hello Common Provider
PostUrl ==http://localhost:8080/tracking1/Login?user=sow
output[200]
PostUrl ==http://localhost:8080/tracking1/Getdeatils?use
output[{"mobile":"9133198693","id":"21"}]
PostUrl ==http://localhost:8080/tracking1/BusLoc?busno=
output[{"destination":"Vizag","seat_no":"10","busno":"A
PostUrl ==http://localhost:8080/tracking1/Locations?bus
output[{"location_name":"Suryapeta","sno":"72","time":"
{"location_name":"Kodhada","sno":"73","time":"9:00pm","
{"location_name":"Nandigama","sno":"74","time":"10:00pm"
{"location_name":"Vijayawada","sno":"75","time":"1:30pm"
{"location_name":"Rajhumundry","sno":"76","time":"2:30a
{"location_name":"Nidadhavolu","sno":"77","time":"4:30a
{"location_name":"Vizag","sno":"78","time":"7:30am","ke
>
```

Figure 6.10 Locations Page

Re-checks the selected location given by the user.



HOTELS

Spice Magic Resturnat

***

JK Resturant

Sp Food Court

****

Sangeetha Resturant

Figure 6.11 Hotels Page

Displays the hotels which are nearby the selected location.

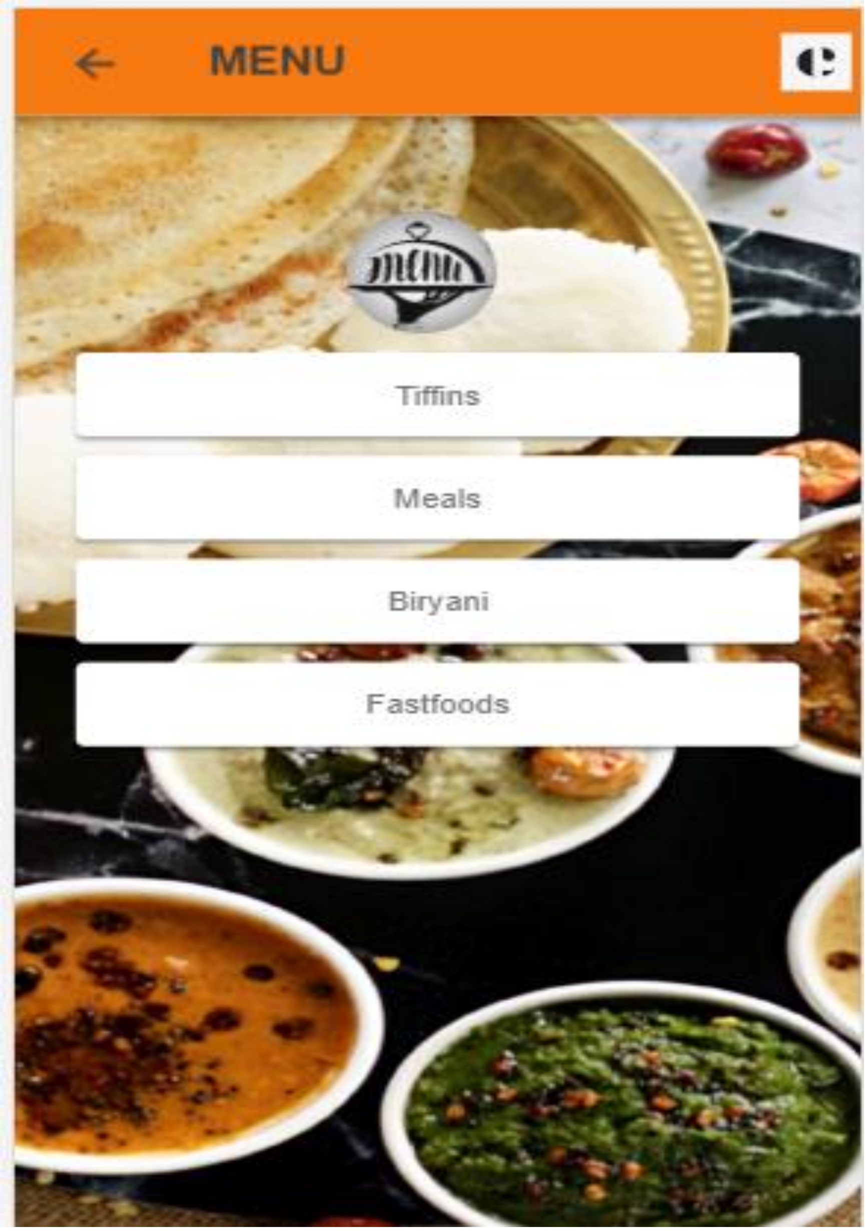
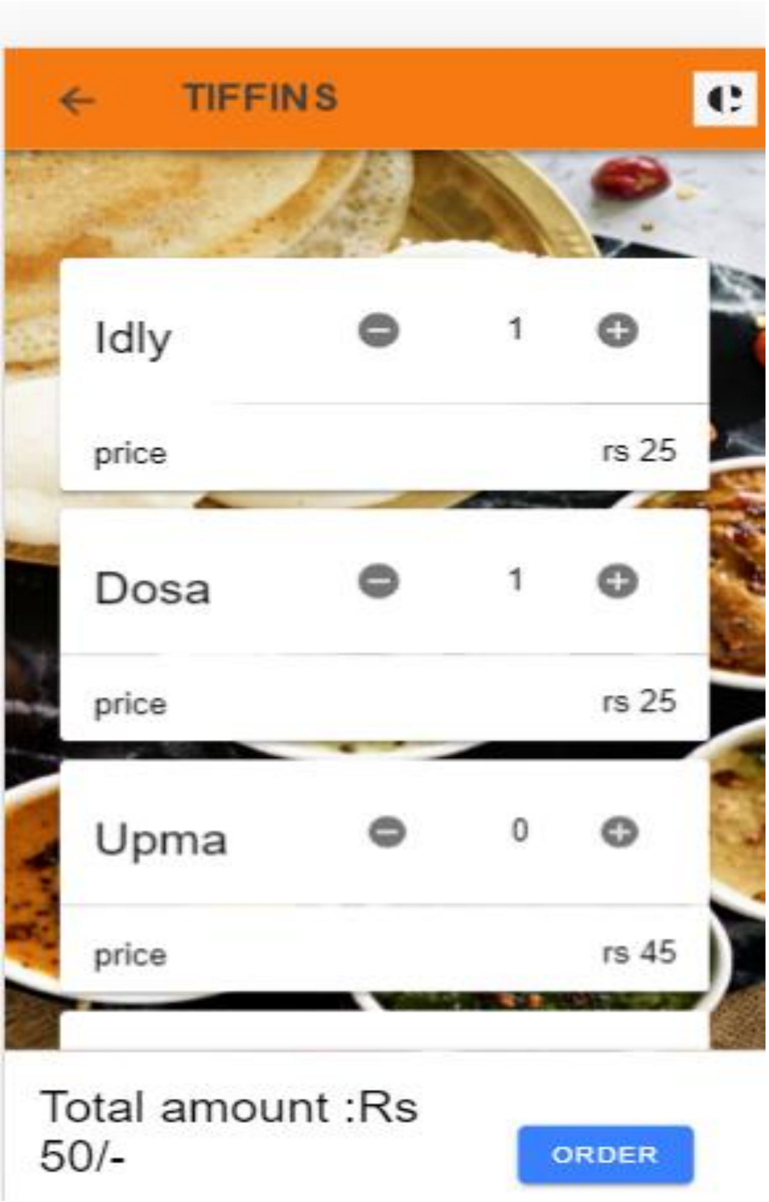


Figure 6.12 Menu Page

Shows all the categories of the menu.



Idly 1

price rs 25

Dosa 1

price rs 25

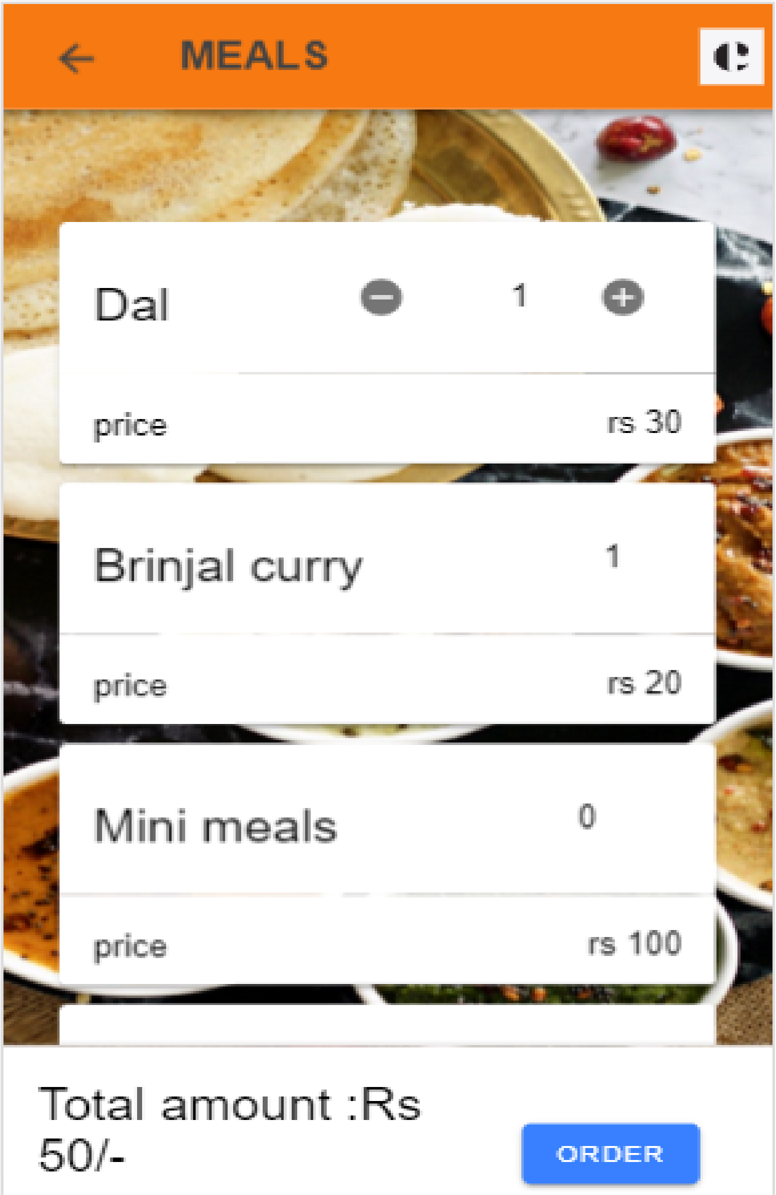
Upma 0

price rs 45

Total amount :Rs 50/-

Figure 6.13 Tiffins Page

Displays a variety of tiffins.



Dal 1

price rs 30

Brinjal curry 1

price rs 20

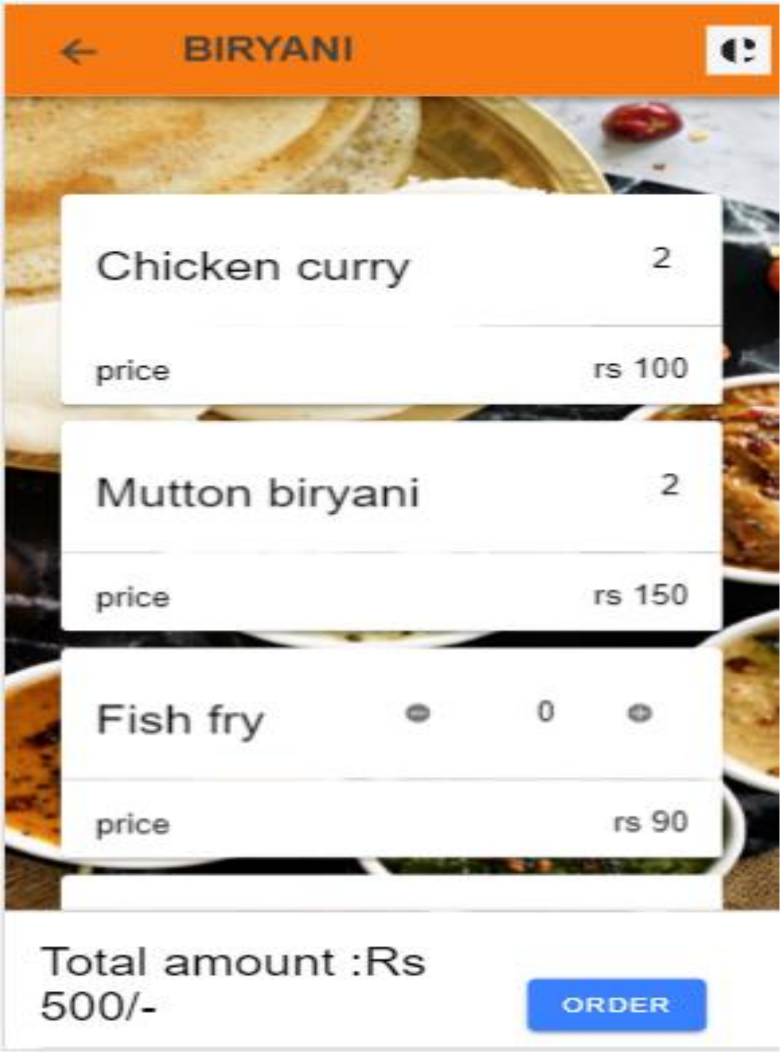
Mini meals 0

price rs 100

Total amount :Rs 50/-

Figure 6.14 Meals Page

Displays a variety of Meals.



BIRYANI

Chicken curry 2

price rs 100

Mutton biryani 2

price rs 150

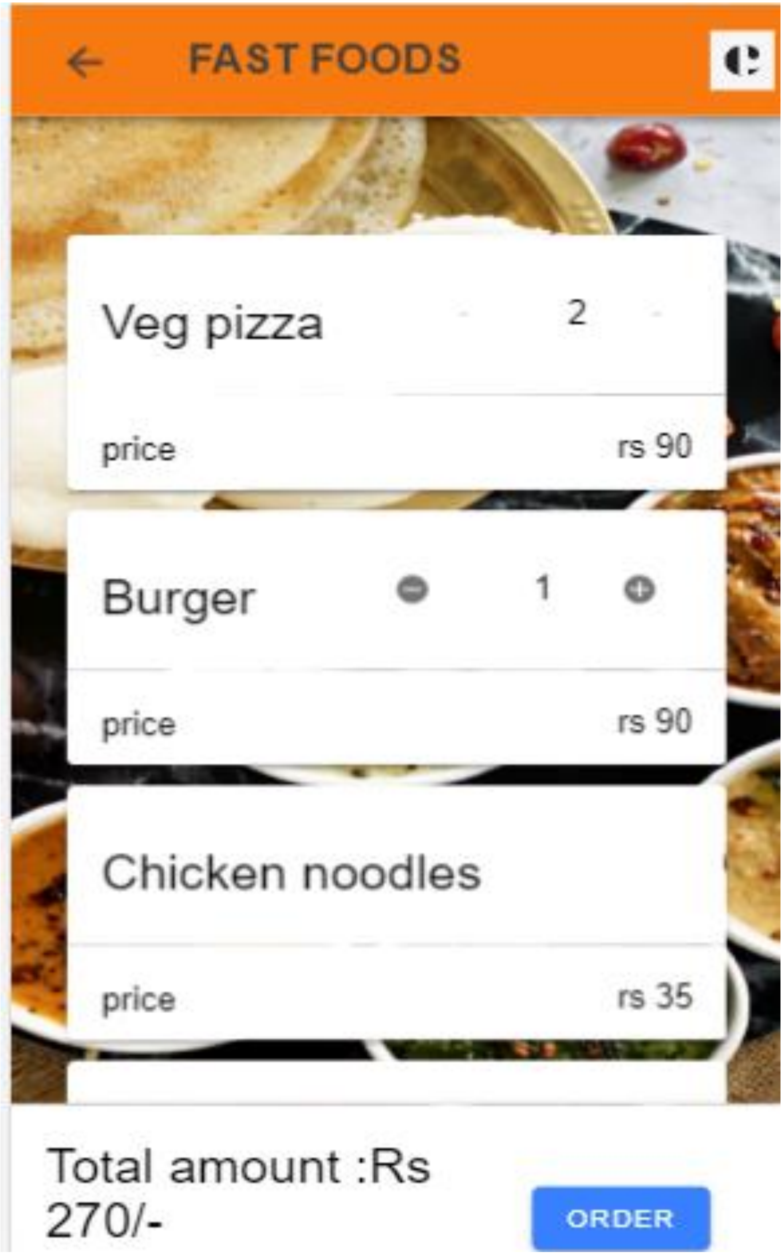
Fish fry 0

price rs 90

Total amount :Rs 500/-

Figure 6.15 Biryani Page

Displays a variety of Biryani.



Veg pizza 2

price rs 90

Burger 1

price rs 90

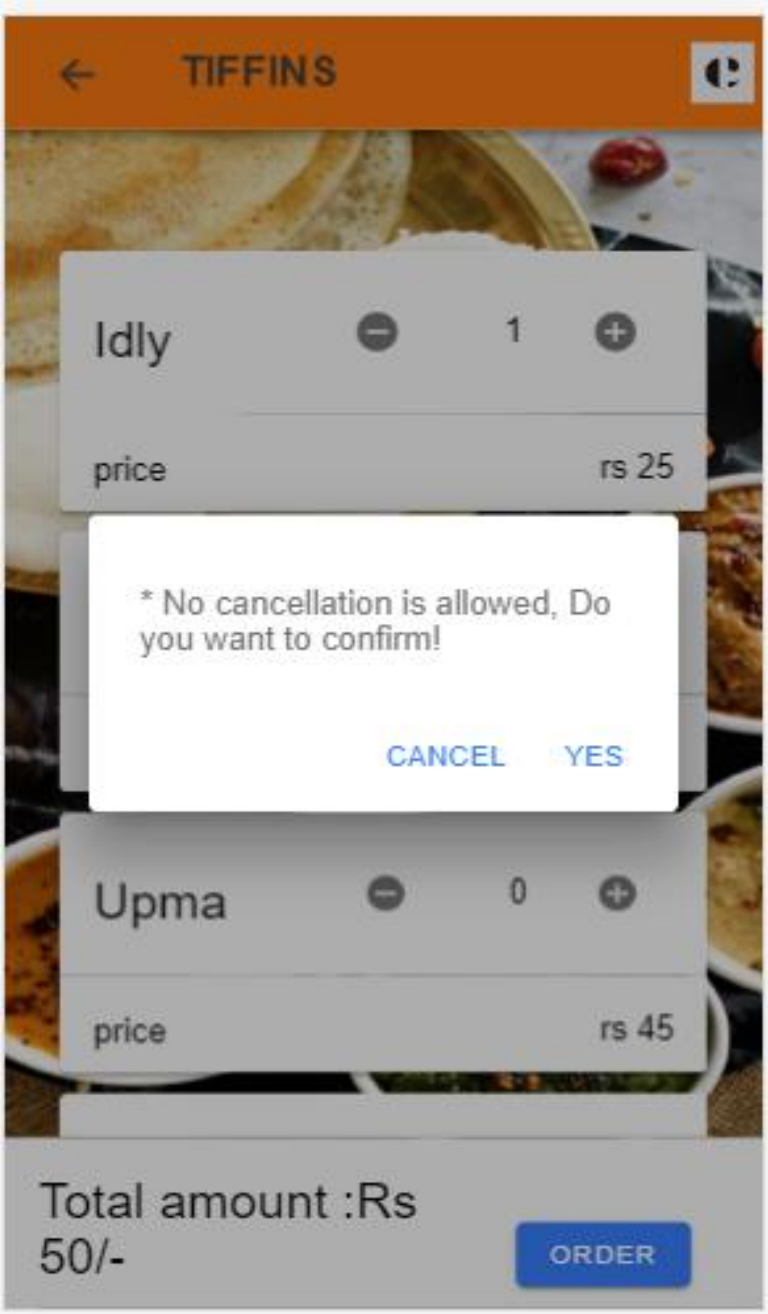
Chicken noodles

price rs 35

Total amount :Rs 270/-

Figure 6.16 Fast Foods Page

Displays a variety of Fast Foods.

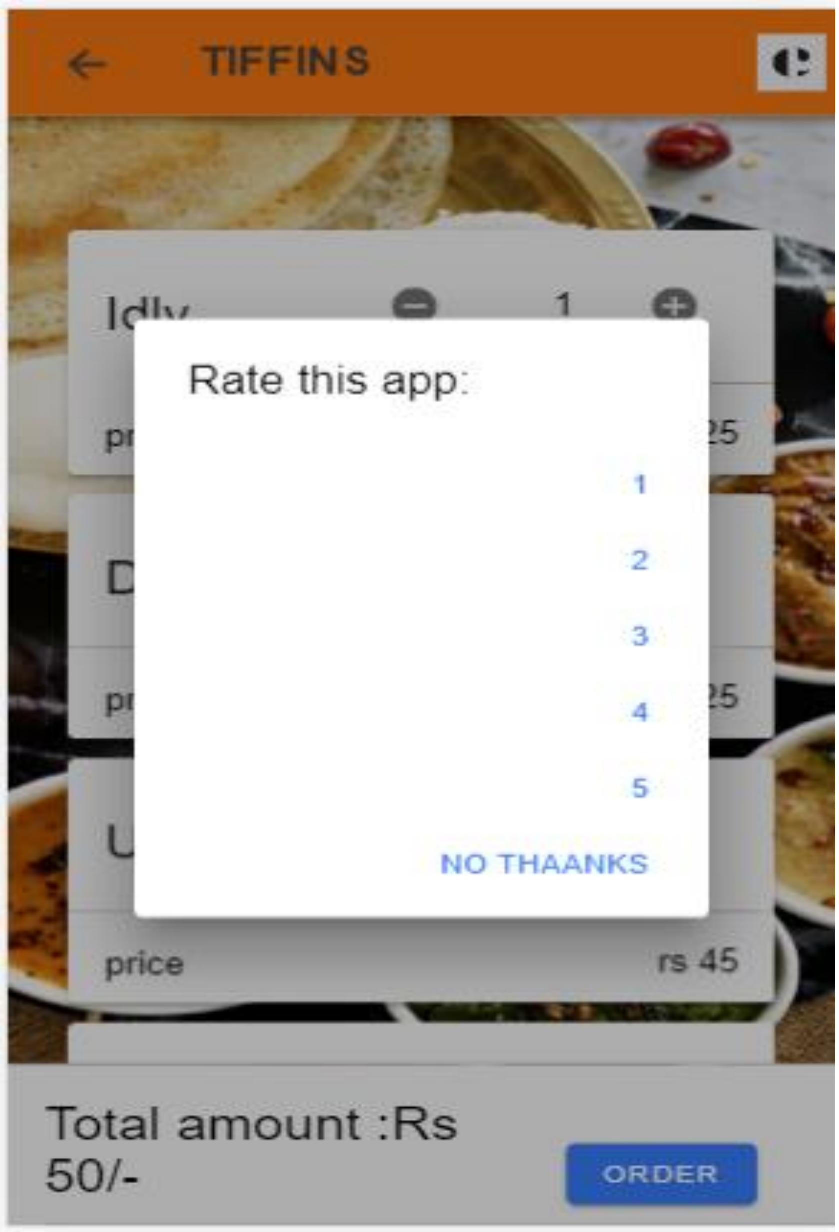


* No cancellation is allowed, Do you want to confirm!

Total amount :Rs 50/-

Figure 6.17 Confirmation Page

Asks for the order confirmation.

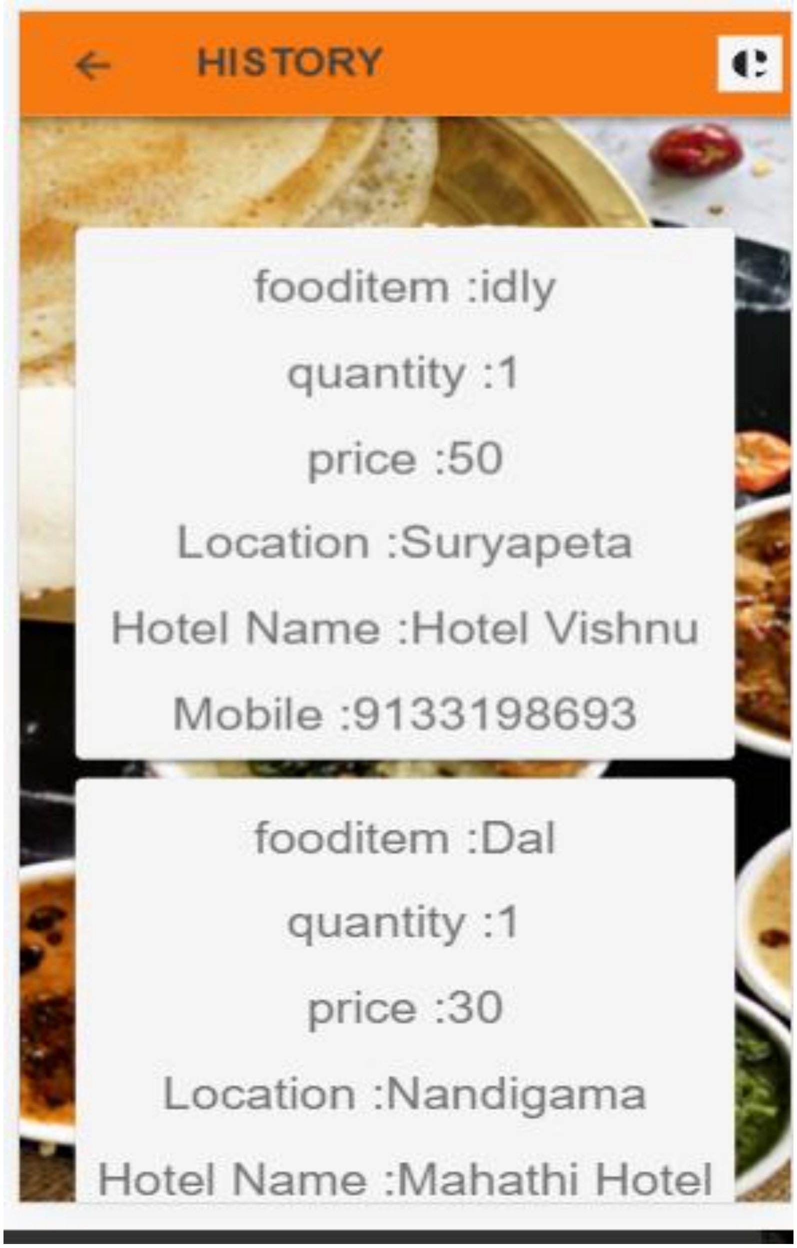


Rate this app:

Total amount :Rs 50/-

Figure 6.18 Rating Page

Requests to rate the application.



fooditem :idly

quantity :1

price :50

Location :Suryapeta

Hotel Name :Hotel Vishnu

Mobile :9133198693

fooditem :Dal

quantity :1

price :30

Location :Nandigama

Hotel Name :Mahathi Hotel

Figure 6.19 User History Page

Displays the history of the user.

6.2 Report

Report-login:

Test case ID	Input	Description	Expected Results	Reports
CIPA_01	Username blank Password Blank	A blank user name and password given	Please enter user name and password message will be displayed	Pass
CIPA_02	Valid user name Password blank	A valid user name and a blank password given	Please enter password	Pass
CIPA_03	Blank user name Password valid	A blank user name and a valid password given	Invalid user name	Pass
CIPA_04	Valid user name Valid password	A valid user name and password given	Redirected to next page	Pass
CIPA_05	Invalid user name Invalid password	A invalid user name and password given	Invalid credentials message will be displayed	Pass

Table 6.1 Report for logins

Report-register:

Test case ID	Input	Description	Expected Results	Reports
CIPA_01	Any one field in the form is left blank	One or more fields in the form were not entered	Please enter the particular field input message will be displayed	Pass
CIPA_02	Entered all fields correctly	All the fields are entered correctly by the user	Registration successful message will be displayed	Pass
CIPA_03	User name already exists	If the particular user name has already exists in the database	User name already exists message will be displayed	Pass
CIPA_04	Password is not greater than 5	Entered password is less than 5 characters	Password should be greater than 5 message will be displayed	Pass
CIPA_05	Phone number is not equal to 10 digits	Respective phone number less/greater than contains 10 digits	Phone number should be 10 digits message will be displayed	Pass

Table 6.2 Report for Registration

Report-selection:

Test case ID	Input	Description	Expected Results	Reports
CIPA_01	Service number blank	A service number given	Invalid Service number	pass
CIPA_02	Valid service number Seat number greater than 60	A valid service number and seat number greater than 60 given	Seat number should be less than 60 message will be displayed	Pass
CIPA_03	Valid servicenumber Seat number less than 60	A valid service number and seat number less than 60 given	Source and destinations will be displayed	Pass

Table 6.3 Report for Seat number and Service number

Report-food items:

Test case ID	Input	Description	Expected Results	Reports
CIPA_01	Cart empty	None of the items were selected by the user	Please select one message will be displayed	pass
CIPA_02	One or more items in the cart	One or more items have been selected by the user	Do you want to confirm order message will be displayed	pass

Table 6.4 Report for Checking the Cart.

Report-confirm order:

Test case ID	Input	Description	Expected Results	Reports
CIPA_01	Order confirmed	The order has been confirmed by the user	Booking successful message will be displayed	pass
CIPA_02	Order not confirmed	The order has been cancelled by the user	Redirect to login page	pass

Table 6.5 Report for Confirmation.

7.1Conclusion

Cibus Partum provides a simple and convenient way for users to purchase food online, without any burden.

This system is enabled by the internet, where users visit the application, and browse through the various food items before purchasing or ordering the ones he or she wants on the application. The users can then have the food delivered to their location Payments can be made through cash or card on delivery. This method is completely safe, secure and reliable, and is revolutionizing the present restaurant industry.

7.2Future Scope :

The section describe the work that will implemented with future releases of this application.

- Enhance User Interface by adding more user interactive features. Provide Deals and promotional Offer details to home page. Provide Recipes of the Week/Day to Home Page.
- Payment Options: Add different payment options such as PayPal, Cash, Gift Cards etc. Allow to save payment details for future use.
- Order Process Estimate: Provide customer a visual graphical order status bar
- Order Status: Shows the status of the order to the user.
- Order Ready notification: Send an Order Ready notification to the customer.
- Intimating the bus location to the user.

8. BIBLIOGRAPHY

Websites referred:

- https://ionicframework.com/
- https://www.guru99.com/introduction-to-mysql-workbench.html
- www.vogella.com/tutorials/Eclipse/article.html
- https://www.eclipse.org/getting_started/
- https://www.tutorialspoint.com/nodejs/
- www.nodeclipse.org/
- https://www.slideshare.net
- https://www.scribd.com
- https://www.youtube.com
- https://dev.mysql.com
- https://www.guru99.com