MINI PROJECT REPORT

On

"**foodie**"

Submitted by

Deepak Parihar (201500206)

Suchita Khare (201500712)

Akash Rathour (201500060)

Omveer Singh (201500459)

Nitish Baghel (201500453)

Department of Computer Engineering & Applications

Institute of Engineering & Technology

GLA University

Mathura- 281406, INDIA 2022-2023

Declaration

We hereby declare that the work which is being presented in the MINI Project "**FOODIE",** in partial fulfillment of the requirements for MINI Project viva voce, is an authentic record of our work carried out by the team members under the supervision of our mentor Mr. **Bhanu Kapoor**.

Group Members:

- Deepak Parihar (201500206)
- Suchita Khare (201500712)
- Akash Rathour (201500060)
- Omveer Singh (201500459)
- Nitish Baghel (201500453)

Course: B.Tech (Computer Science and Engineering)

Year: 3rd

Semester: 5th

Supervised By:
Mr. Bhanu Kapoor, Trainer
Department of Training and Development, GLA University

Department of Computer Engineering and Applications GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,

Certificate

This is to certify that the above statements made by the candidates are correct to the best of my/our knowledge and belief.

_____

Supervisor
Mr. Bhanu Kapoor

_____

Project Coordinator
(**Mr. Bhanu Kapoor)**

_____

Program Coordinator
(**Mr. Shashi Shekar**)

Acknowledgment

We thank the almighty for giving us the courage and perseverance in completing the project. This project is an acknowledgment of all those who have given us their heartfelt cooperation in making this project a grand success. We extend our sincere thanks to Mr. Bhanu Kapoor, Trainer at "GLA University, Mathura" for providing his valuable guidance at every stage of this project work. We are profoundly grateful for the unmatched services rendered by him. And finally, we would like to express our deep sense of gratitude and earnest thanksgiving to our dear parents for their moral support and heartfelt cooperation in doing the main project.

Abstract

The Foodie! application provides a pre-order feature which is one of the latest features which saves the customer's time, increases the sales and maximize customer satisfaction. The role of technology in the restaurant industry is rapidly increasing. This application will help busy corporate customers stepping out for short lunches, customers placing large group orders which causes overcrowding in kitchens or customers who simply don't like waiting. This new feature will help you to order food online in advance which will help you save your time. In this customers can set a particular time according to them and can order the food and can receive it at that particular time. This application focuses on providing customer login and enabling them to order food in advance & skip the line . Front end of the system is developed with the help of flutter which uses dart language. Back end and Database management of this system will be done with the help of Firebase database.

CONTENT

INTRODUCTION

The restaurant industry has changed a lot over the past decade and at a rapid pace, largely due to new technology and evolving customer preferences. Customers have changed what they eat, how they eat, and where they eat it. They want their food fast and don't want to wait long. They want the options for pre-order, takeout, or dining in the restaurant, and they want their food to meet all of their dietary restrictions while living up to expectations and taste preferences. Foodie is an android application which enables a user to pre-order food.It is easy to use and is very helpful.This application is basically used to save your time in an easy way. This application is based on Flutter(Google) which uses Dart programing language. You can order your meal online and get it at the time that you choose. For example if you only have 1 hour for your lunch and want the meal at the beginning of the break time then you can easily order your meal online before the time of break and pick it at the break time. Enjoy your meals without any delay by using Foodie! app.

SOFTWARE REQUIREMENTS

- Flutter
- Dart
- Android Studio

HARDWARE REQUIREMENTS

- 8 GB RAM
- Window 10,8,7
- Processor : Intel i5
- Hard Disk: 256 GB
- Window 10

IMPLEMENTATION DETAILS

PROJECT DESCRIPTION :-

The purpose of this project is to develop an android application for ordering the food in advance in order to save time. This application is based on flutter which uses dart as the programming language. You can pre-order your meal online and can pick up your order at the scheduled time. For example - if you have only 1 hour for your lunch and you also want the meal at the beginning of the break time then you can easily order your meal online before time and pick it from the restaurant at the break time.

Flutter :-

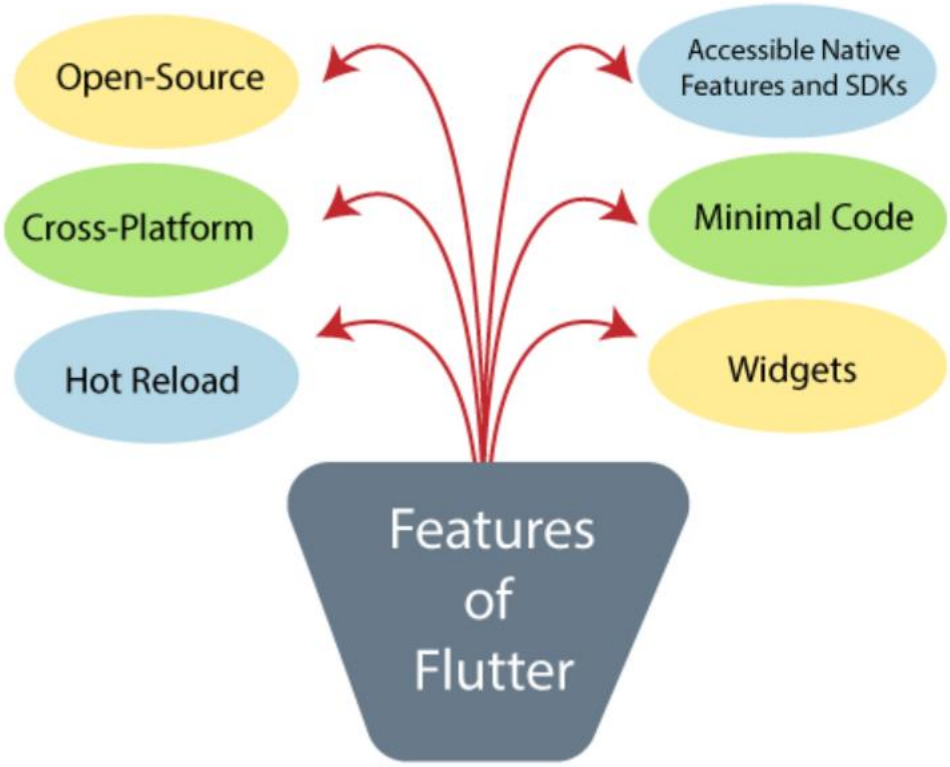
Flutter is a UI toolkit for building fast, beautiful, natively compiled applications for mobile, web, and desktop with one programing language and single codebase. It is free and open-source. Initially, it was developed from **Google** and now manages by an **ECMA standard**. Flutter apps use Dart programming language for creating an app.

The first version of Flutter was announced in the year **2015** at the Dart Developer Summit. It was initially known as codename **Sky** and can run on the Android OS. On **December 4, 2018**, the first stable version of the

Flutter framework was released, denoting Flutter 1.0. The current stable release of the framework is Flutter v1.9.1+hotfix.6 on October 24, 2019.

Features of Flutter :-

Flutter gives easy and simple methods to start building beautiful mobile and desktop apps with a rich set of material design and widgets. Here, we are going to discuss its main features for developing the mobile framework.



Dart :-

Dart is a general-purpose, high-level modern programming language which is originally developed by Google. It is the new programming language which is emerged in 2011, but its stable version was released in June 2017. Dart is not so popular at that time, but It gains popularity when it is used by the Flutter.

Dart is a dynamic, class-based, object-oriented programming language with closure and lexical scope. Syntactically, it is quite similar to Java, C, and JavaScript. If you know any of these programming languages, you can easily learn the Dart programming language.

Dart is an open-source programming language which is widely used to develop the mobile application, modern web-applications, desktop application, and the Internet of Things (IoT) using by Flutter framework. It also supports a few advance concepts such as interfaces, mixins, abstract classes, refield generics, and type interface. It is a compiled language and supports two types of compilation techniques.

About the Project

Online ordering has enabled many restaurants to manage their peak business hours very effectively. Thanks to online ordering many people manage to prevent the painful experience of wasting time in a long queue. With the option to order their meals from a mobile app, they can easily place an order when they are stuck in traffic or on the way to pick up the kids.

Mobile apps provide the freedom to order from any place at any time without pausing everything and making a call to the restaurant. The food experience has come a long way it has become a much more hassle-free experience for the customers. And if you can vouch for such an experience then the customers will definitely come your way.

Research shows that customers are more likely to make a purchase if you provide them with a favorable cashless experience.

The lesser you deal in cash more convenient the experience becomes.

Mobile apps help you to achieve that very easily. In this way the customer can choose for the most convenient option.

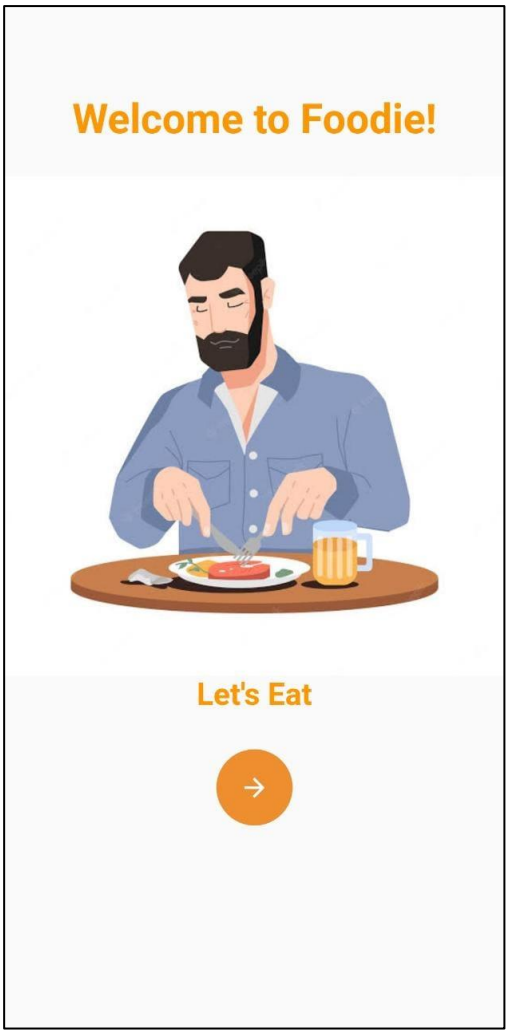
The Foodie! application provides a pre-order feature which is one of the latest features which saves the customer's time, increases the sales and maximize customer satisfaction. This new feature will help you to order food online in advance which will help you save your time. In this customers can set a particular time according to them and can order the food and can receive it at that particular time. This application focuses on providing customer login and enabling them to order food in advance & skip the line .

Motivation

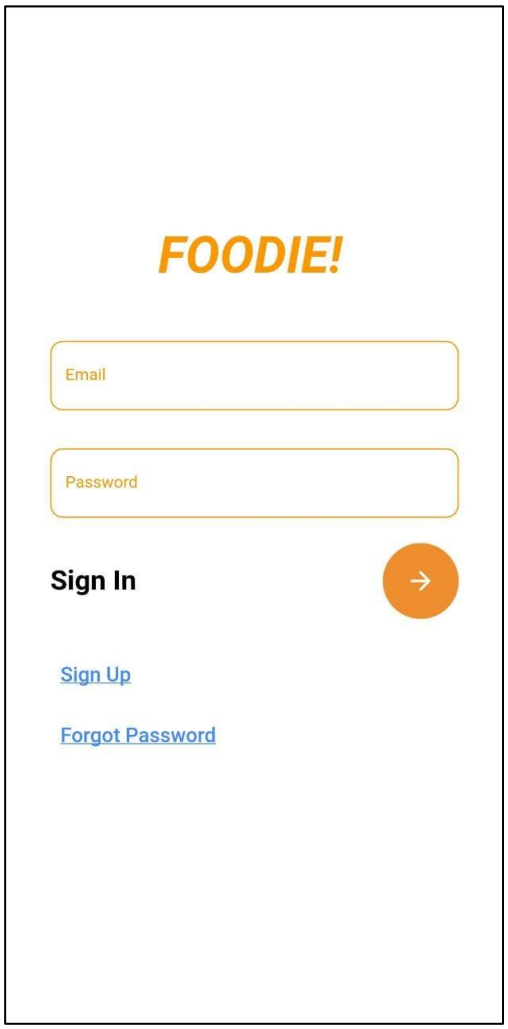
The role of technology in the restaurant industry is rapidly increasing. Our application will help busy corporate customers stepping out for short lunches, customers placing large group orders which causes overcrowding in kitchens or customers who simply don't like waiting. This new feature will help you to order food online in advance which will help you save your time. In this customers can set a particular time according to them and can order the food and can receive it at that particular time. This application focuses on providing customer login and enabling them to order food in advance & skip the line .

List of UI

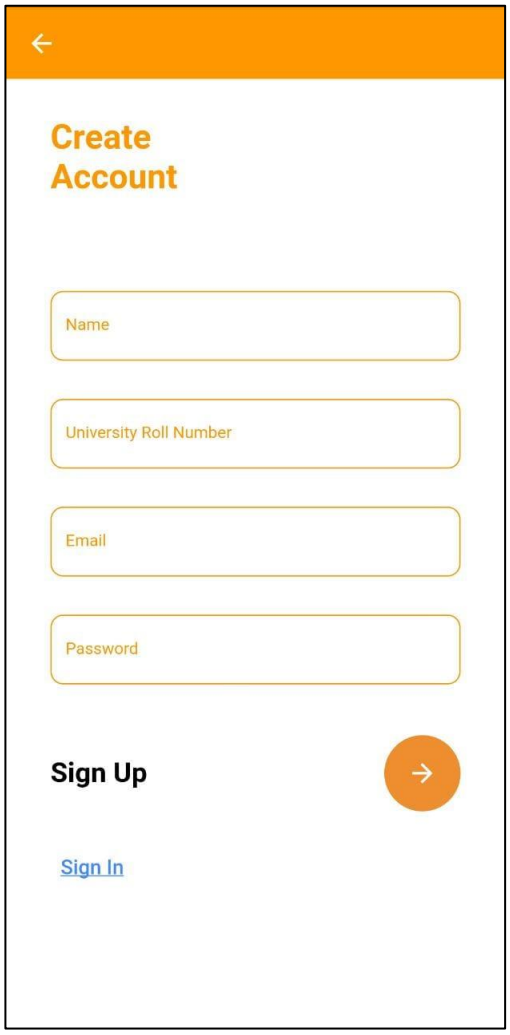
Welcome Page :-



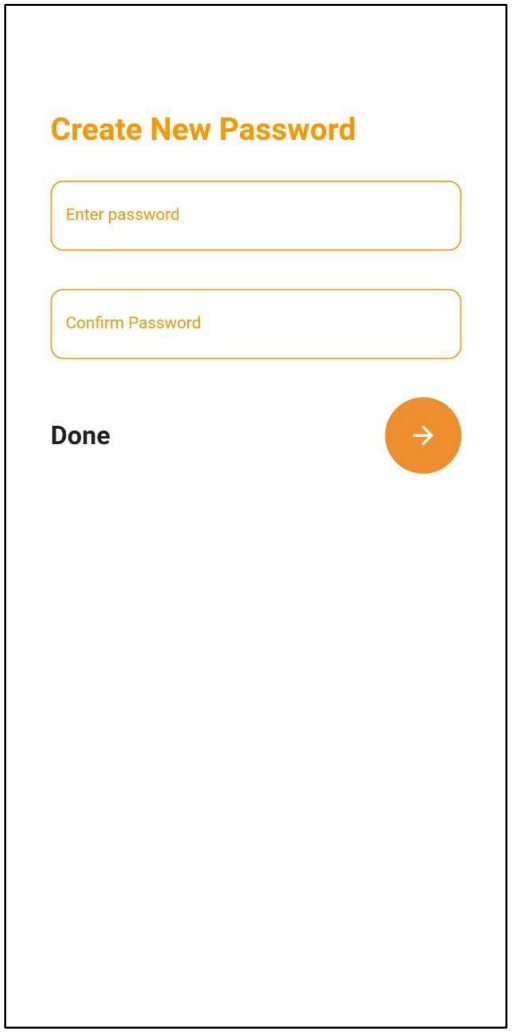
LogIn Page:-



SignUp Page : -

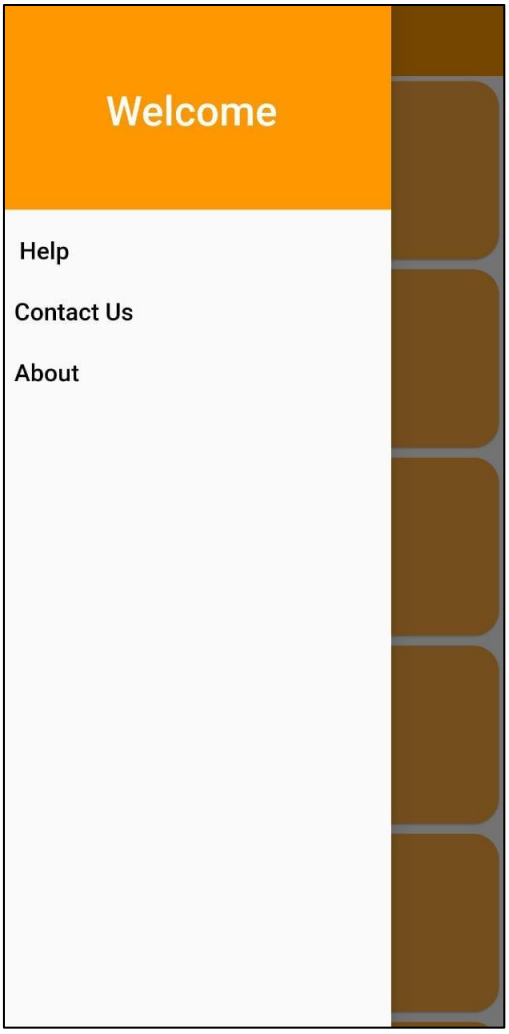


Forget Password :-

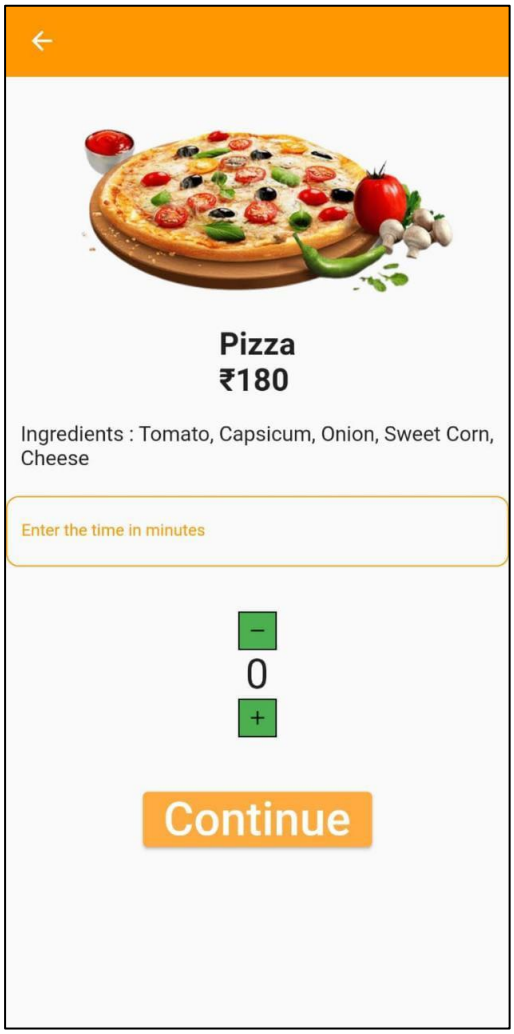


Home Page:-

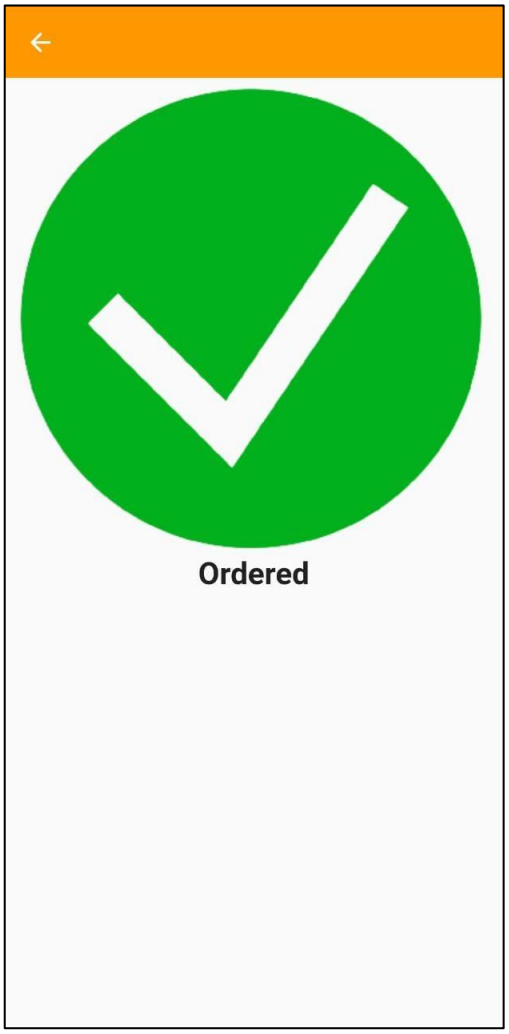
Drawer Page:-



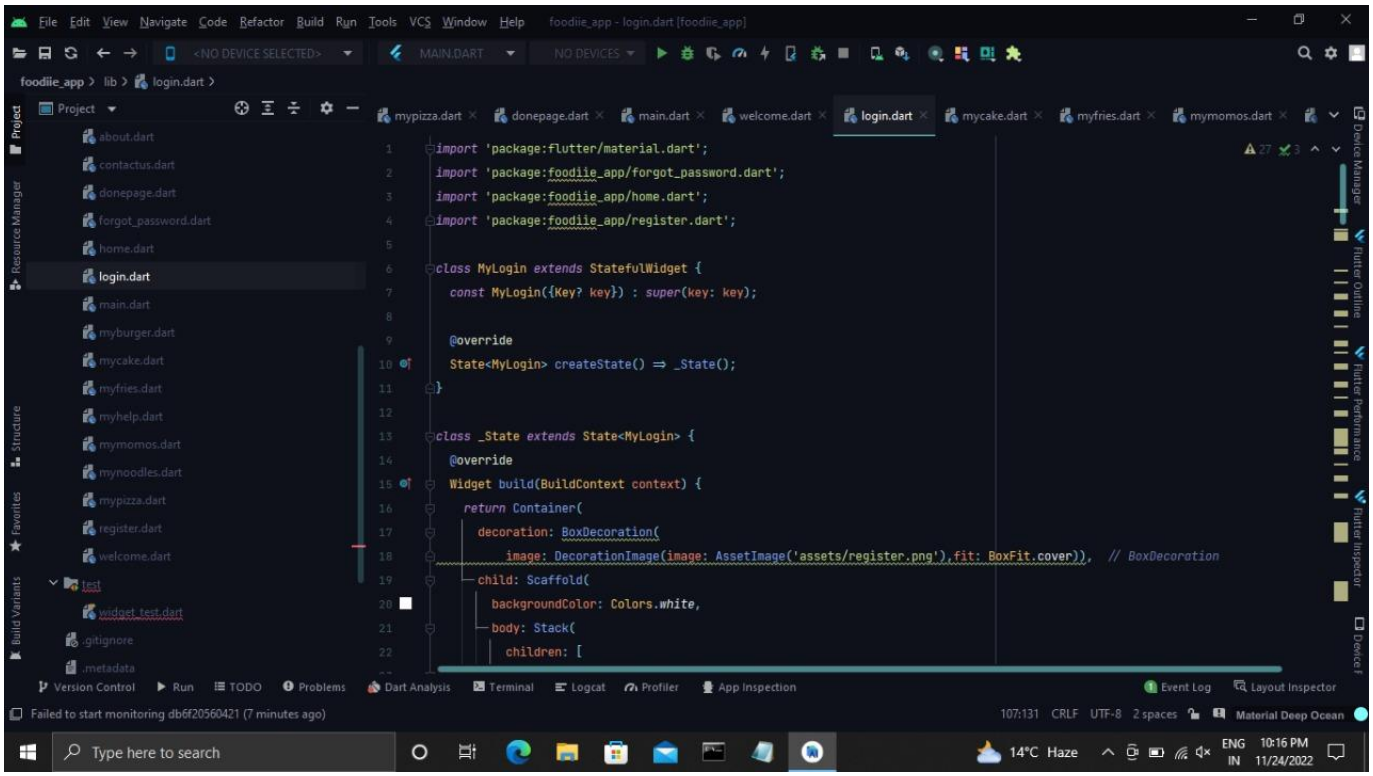
Order Quantity Page:-

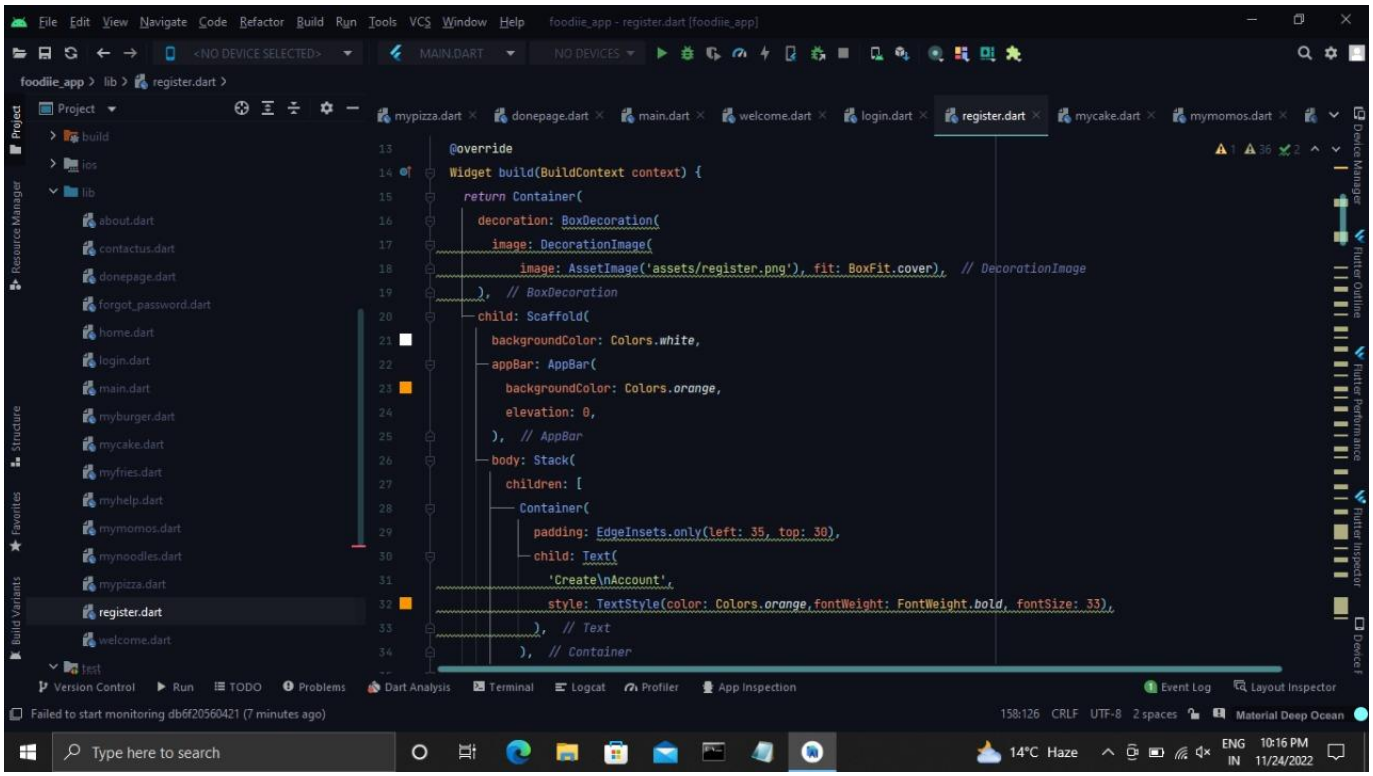


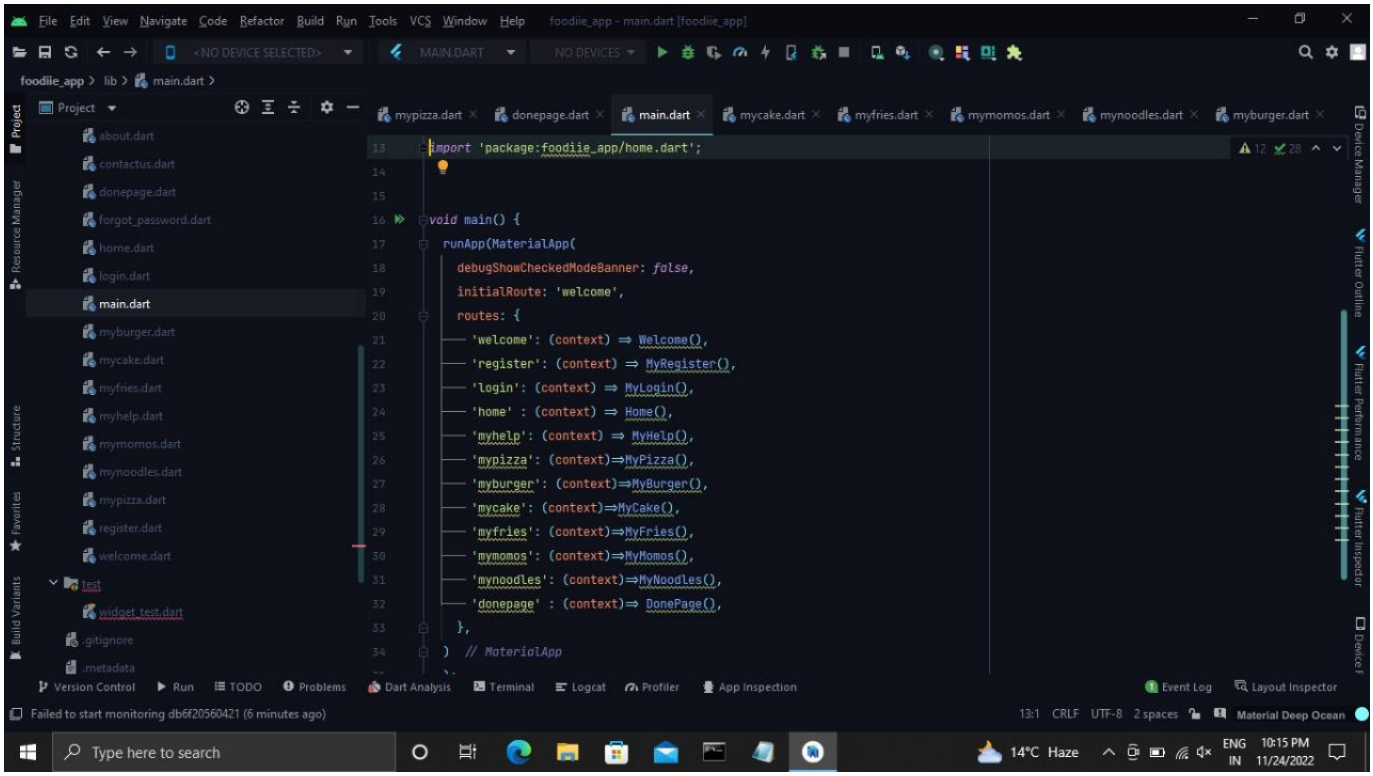
Order confirmation Page:-

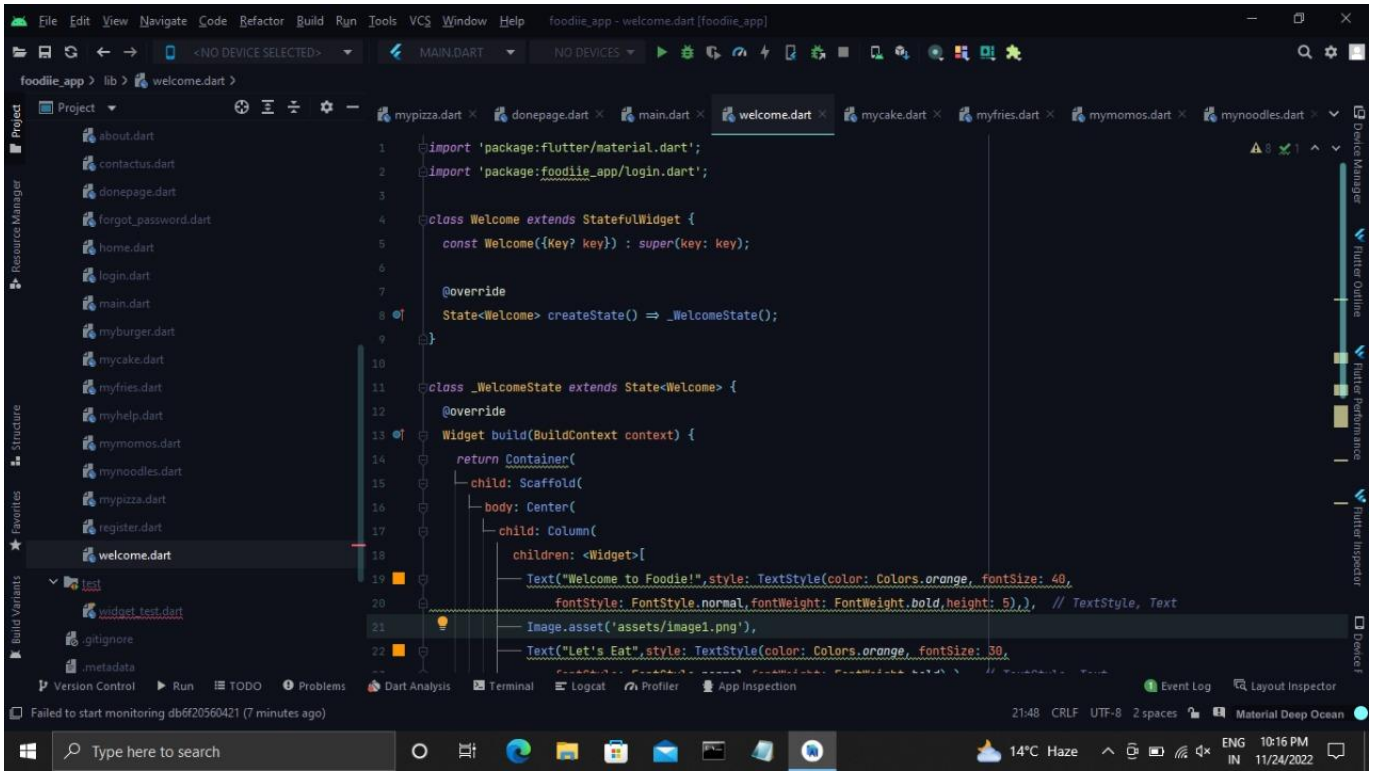


Some Code Pages:-






File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
foodiie_app - main.dart [foodiie_app]
foodiie_app > lib > main.dart >
mypizza.dart donepage.dart main.dart mycake.dart myfries.dart mymomos.dart mynoodles.dart myburger.dart
import 'package:foodiie_app/home.dart';
void main() {
runApp(MaterialApp(
debugShowCheckedModeBanner: false,
initialRoute: 'welcome',
routes: {
'welcome': (context) => Welcome(),
'register': (context) => MyRegister(),
'login': (context) => MyLogin(),
'home' : (context) => Home(),
'myhelp': (context) => MyHelp(),
'mypizza': (context)=>MyPizza(),
'myburger': (context)=>MyBurger(),
'mycake': (context)=>MyCake(),
'myfries': (context)=>MyFries(),
'mymomos': (context)=>MyMomos(),
'mynoodles': (context)=>MyNoodles(),
'donepage' : (context)=> DonePage(),
},
) // MaterialApp
Failed to start monitoring db6f20560421 (6 minutes ago)
Type here to search
14°C Haze
ENG IN 10:15 PM 11/24/2022


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
foodiie_app - welcome.dart [foodiie_app]
foodiie_app > lib > welcome.dart >
mypizza.dart donepage.dart main.dart welcome.dart mycake.dart myfries.dart mymomos.dart mynoodles.dart
import 'package:flutter/material.dart';
import 'package:foodiie_app/login.dart';
class Welcome extends StatefulWidget {
const Welcome({Key? key}) : super(key: key);
@override
State<Welcome> createState() => _WelcomeState();
}
class _WelcomeState extends State<Welcome> {
@override
Widget build(BuildContext context) {
return Container(
child: Scaffold(
body: Center(
child: Column(
children: <Widget>[
Text("Welcome to Foodie!",style: TextStyle(color: Colors.orange, fontSize: 40,
fontStyle: FontStyle.normal,fontWeight: FontWeight.bold,height: 5),), // TextStyle, Text
Image.asset('assets/image1.png'),
Text("Let's Eat",style: TextStyle(color: Colors.orange, fontSize: 30,
Failed to start monitoring db6f20560421 (7 minutes ago)
Type here to search
14°C Haze
ENG IN 10:16 PM 11/24/2022

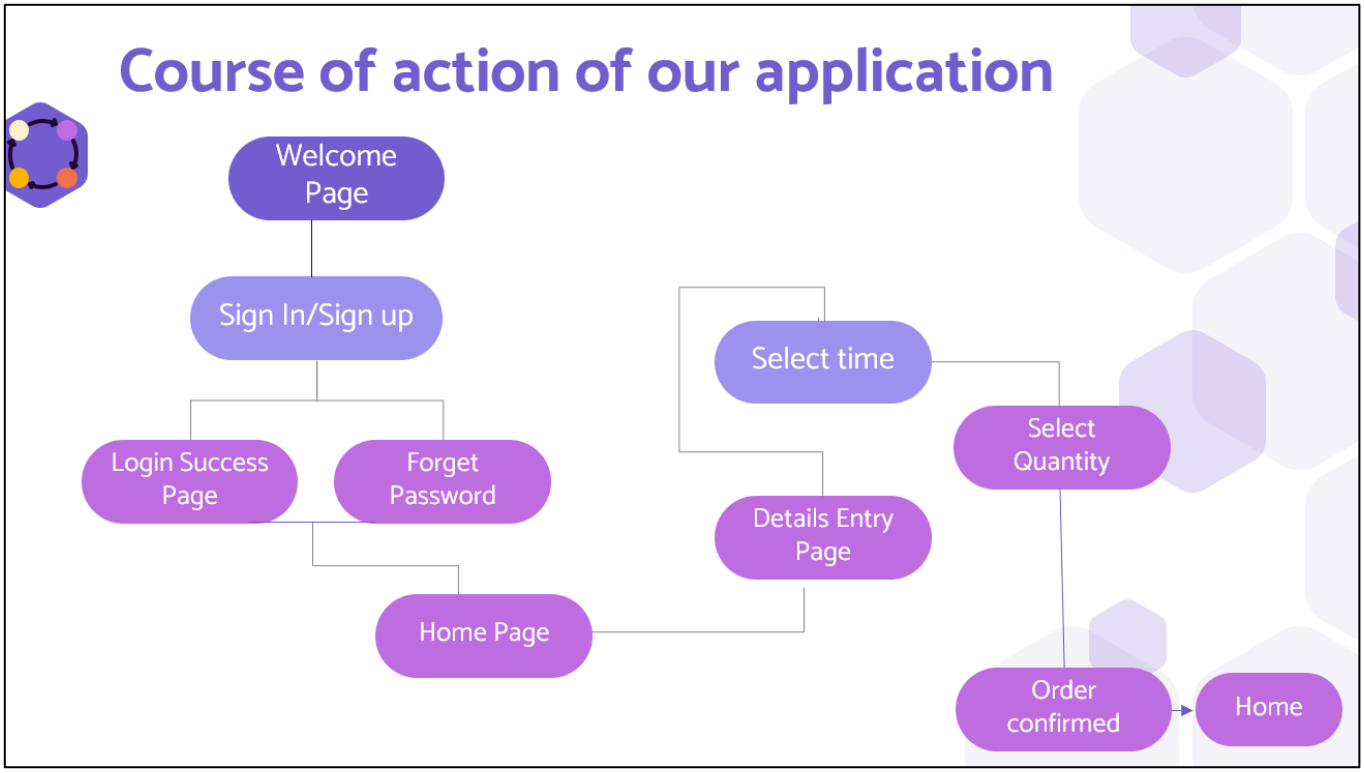
Software Testing :-

Once source code has been generated, software must be tested to uncover as many errors as possible before delivery. It is very important to work the system successfully and achieve high quality of software. Testing include designing at series of test cases that have a high likelihood of finding errors by applying software-testing techniques. System testing makes logical assumptions that if all the parts of the system are correct, the goal will be successfully achieved. The system should be checked logically. Validations and cross checks should be there. Avoid duplications of record that cause redundancy of data. In other Words, Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. It is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

The preliminary goal of implementation is to write source code and internal documentation so that conformance of the code to its specifications can be easily verified, and so that debugging, testing and modifications are eased. This goal can be achieved by making the source code as clear and straightforward as possible. Simplicity, clarity and elegance are the hallmark of good programs, obscurity. cleverness, and complexity are indications of inadequate design and misdirected thinking. Source code clarity is enhanced by structured coding techniques, by good coding style, by, appropriate supporting documents, by good internal comments, and by feature provided in modern programming languages. The implementation team should be provided with a well-defined set of software requirement, an architectural design specification, and a

detailed design description. Each team member must understand the objectives of implementation.

RoadMap :-



REFRENCES

- www.w3school.com
- www.javatpoint.com
- www.projectdeveloper.com
- www.dart.dev
- www.tutorialspoint.com
- www.geeksforgeeks.org

Faculty Guidelines:

Mr. Bhanu Kapoor (Technical Trainer in GLA University)

GitHub Repository link:

https://github.com/deepakparihar10/Mini_project_1