RAMAIAH INSTITUTE OF TECHNOLOGY,
BANGALORE – 560054
(Autonomous Institute, Affiliated to VTU)

Department of Computer Science & Engineering

Internship Report

on

Mobile Application Development
INT411: Intra Institutional Internship

STUDENT NAME: SAHANA BHAGAVANT SAVANTANAVAR

USN : 1MS22EE043

Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)
MSR Nagar, MSRIT Post, Bangalor-560054

October-November, 2023

RAMAIAH
Institute of Technology

RAMAIAH INSTITUTE OF TECHNOLOGY,
BANGALORE – 560054
(Autonomous Institute, Affiliated to VTU)

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that Ms. SAHANA.BHAGAVANT.SAVANTANAVAR , a student of Bachelor of Engineering, bearing USN:1MS22EE043, has successfully completed, 20 Hours: from 25.10.2023 to 8.11.2023 Intra Institutional Internship in Mobile Application Development from the Department of Computer Science & Engineering, M S Ramaiah Institute of Technology, Bangalore.

SL No.	Component	Maximum Marks	Marks Obtained
1	**Continuous Evaluation**	50	
2	**Presentation**	20	
3	**Report**	30	
Total Marks		100	

Signature of the Student with Date **Signature of the Faculty Co- Ordinator**

Signature of Head of the Department

OVERVIEW OF INTERNSHIP ACTIVITIES

DATE	DAY	NAME OF THE TOPIC COMPLETED
25/10/23	Wednesday	Basics of Dart Code
26/10/23	Thursday	Advanced Dart language
27/10/23	Friday	Introduction to Flutter platform.
30/10/23	Monday	Installation of Flutter
31/10/23	Tuesday	Mobile App Development using Flutter
02/11/23	Thursday	Flutter alerts ,icons ,images ,toasts etc.
03/11/23	Friday	Creating Container
06/11/23	Monday	Firebase + Flutter
07/11/23	Tuesday	Assignment
08/11/23	Wednesday	Project Demo

TABLE OF CONTENTS

Contents **Page. No**

BUS BOOKING APP

1. IMPLEMENTATION:

This implementation section provides an overview of the app's structure, user interface, user interaction, and potential areas for further development. The code provided serves as a starting point for a functional seat booking app, and additional features and logic can be incorporated to make it a complete solution.

The Seat Booking App has been developed using the Flutter framework, enabling crossplatform compatibility for Android devices. This section provides a detailed breakdown of the app's implementation.

1. Project Structure

The project follows a structured organization with the primary code components distributed across several classes and widgets. Here's an overview of the key components:

- BusBookingApp Class: This class sets up the initial app configuration, defining the app's title and theme. It serves as the entry point for the app.
- FrontPage Class: This class represents the home page, offering users the option to start the booking process. An "ElevatedButton" widget triggers navigation to the next step.
- CustomerNamePage Class: Users enter their names on this page, utilizing a "TextField" widget for input. The name is stored using the "TextEditingController" and used for personalization.
- SeatSelectionPage Class: This page allows users to select seats by tapping on them. The selected seats are highlighted in red, and a "Selected Seats" list is displayed. Users can proceed to the confirmation page from here.
- Confirmation Page Class: The confirmation page displays the customer's name and the selected seats. Users can confirm the booking, and there's an option to perform booking logic (not implemented in this code).

2 .User Interface

The user interface has been designed with a clean and intuitive layout:

- Home Page: The initial screen with a "Book a Seat" button.
- Customer Name Entry Page: Users can input their names, and this page is wrapped in a "Center" widget for better alignment.
- Seat Selection Page: Users can select seats by tapping. The seats' selection status is dynamically updated with color changes.
- Confirmation Page: This page displays booking details and allows users to confirm their bookings.

3. User Interaction

Users can navigate through the app using the provided buttons. The "onTap" function in the SeatSelectionPage class dynamically updates the selected seats and their color.

4.Data Handling

The app uses a "TextEditingController" to capture and store the user's name. The selected seats are maintained in a list.

5.Navigation

The app utilizes Flutter's "Navigator" for page navigation, ensuring a smooth and structured user experience.

6.Further Development

The current implementation serves as a foundation for a seat booking app. However, for a complete and functional app, the following aspects can be further developed:

Booking Logic: The code for confirming the booking is present, but the booking logic itself is not implemented. This would involve saving booking details, payment integration, and potentially interacting with a backend server.

Error Handling: Implementing error handling and validation for input data and seat selection to enhance user experience and data integrity.

User Authentication: Implementing user accounts and authentication for secure bookings.

Payment Integration: Integrating payment gateways to facilitate ticket payments

2. CODE MODULE:

Code Module Breakdown

1. Main.dart

- Entry point of the application.
- Defines the BusBookingApp class, which sets up the app's initial configuration.

2. BusBookingApp Class

- Configuration class for the Flutter app.
- Sets the app's title and theme.
- Specifies the home page to be displayed.

3. FrontPage Class

- Represents the home page of the app.
- Displays the app name and provides a button to initiate the booking process.

4.BookingApp Class (Not utilized in the code provided)

- A placeholder class that could potentially represent another screen or route.

5.CustomerNamePage Class

- Allows users to enter their name.
- Utilizes a TextEditingController to capture the user's name.
- Provides a button to navigate to the next step.

6.SeatSelectionPage Class

- Enables users to select seats.

- Displays a list of seats, and users can tap on seats to toggle their selection.
- The selected seats are dynamically updated and displayed.

Provides a button to proceed to the confirmation page.

7.ConfirmationPage Class

- Displays the customer's name and the list of selected seats.
- Offers a button for confirming the booking (booking logic is not fully implemented).

■Code Structure

The code is structured with separate classes for different app screens.

Each class is responsible for rendering its respective UI and handling specific user interactions.

Navigation between screens is achieved using the Navigator widget.

User input and data capture are facilitated through the use of TextEditingController.

The code is organized in a modular manner, making it easier to maintain and extend the app's functionality.

■User Interaction

Users can navigate through the app screens by tapping buttons.

The SeatSelectionPage allows users to interact with seat selection by tapping on seats.

The selected seats are tracked dynamically, and their appearance is updated in realtime.

This code provides a foundation for a simple seat booking app. You can further develop and expand upon it to include features like booking logic, payment integration, and user authentication to create a fully functional app.

CODE FROM THE FLUTTER

```
    import 'dart:async'; import
    'package:flutter/material.dart';
    import'package:video_player/video_player.dr;
    void main() => runApp(BusBookingApp());
    class BusBookingApp extends StatelessWidget
    {

    @override

    Widget build(BuildContext context) {
    return MaterialApp(     title: 'Bus Seat
    Booking',
    Theme:ThemeData(
    primarySwatch: Colors.yellow,     ),
    home:VideoPlayerScreen(),
    debugShowCheckedModeBanner: false,
    ); }}

    class VideoPlayerScreen extends StatefulWidget {

@override

_VideoPlayerScreenStatecreateState() => _VideoPlayerScreenState();}
class _VideoPlayerScreenState extends State<VideoPlayerScreen> {   late
VideoPlayerController _controller; late Timer _timer;
```

```
@override

void initState() {     super.initState();
_controller = VideoPlayerController.network(

'https://www.shutterstock.com/shutterstock/videos/1057221406/preview/stockfootagescho
ol-bus-with-happy-children-cartoon-characters-going-to-school-kanimation-
videomotion.webm') // Replace this URL with your video URL

..initialize().then((_) {

// Ensure the first frame is shown after the video is initialized
setState(() {});
// Play the video after a delay of 10 seconds _timer

= Timer(Duration(seconds: 10), () {

if (mounted) {

_controller.play();}});

_controller.addListener(() {

 if (_controller.value.position >= _controller.value.duration) {

// Video has finished playing, navigate to
FrontPage              Navigator.pushReplacement(
context,

MaterialPageRoute(builder: (context) => FrontPage()),); } });});}
```

```
@override

Widget build(BuildContext context) {
return Scaffold(   appBar: AppBar(
title: Text('SAMYATI'),
centerTitle:true,// Set the app name here
),body: Center(          child:
_controller.value.isInitialized ?
AspectRatio(

aspectRatio: _controller.value.aspectRatio,
child: VideoPlayer(_controller),

) :

 CircularProgressIndicator(),

),floatingActionButton:
FloatingActionButton(         onPressed: () {
setState(() {  if (_controller.value.isPlaying) {
_controller.pause();

} else {

_controller.play();

}  });},          child:
Icon(
_controller.value.isPlaying ? Icons.pause : Icons.play_arrow, ),),);}
```

```
@override
void dispose() {
super.dispose();
_controller.dispose();

_timer.cancel(); // Cancel the timer to prevent memory leaks

}}class FrontPage extends StatelessWidget {

@override

Widget build(BuildContext context) {     return Scaffold(
appBar: AppBar( title: Text('Bus Booking App'),
 // Set the app name here ), body: Center(
child: Column(            mainAxisAlignment:
MainAxisAlignment.center,
children: <Widget>[

Image.network('https://t4.ftcdn.net/jpg/00/15/53/79/360_F_15537925_5qUqgBbDSQ
HCI5DeP7M0z88ouNIHdeKY.jpg',

),ElevatedButton(
onPressed: () {
Navigator.push(              context,
MaterialPageRoute(builder: (context) => CustomerNamePage()),);},
child: Text('Book a Seat'), ),],),),);}} class BookingApp extends StatelessWidget
{

@override
```

```
Widget build(BuildContext context) {      return
MaterialApp(    title: 'Bus Seat Booking',
theme: ThemeData(primarySwatch:Colors.blue,
),home: CustomerNamePage(),);}}  class
CustomerNamePage extends StatefulWidget {
@override

_CustomerNamePageState createState() => _CustomerNamePageState();} class
_CustomerNamePageState extends State<CustomerNamePage> {
TextEditingController _nameController = TextEditingController();

@override

Widget build(BuildContext context) {     return
Scaffold(       appBar: AppBar(         title:
Text('Enter Your Name'),
),body:Container(

color: Colors.black,           child: Stack( // Wrap the Column
in a Center widget         children: <Widget>[
Image.network('https://images.unsplash.com/photo-
1486673748761a8d18475c757?q=80&w=1000&auto=format&fit=crop&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxzZWFyY2h8Nnx8cm9hZHxlbnwwfHwwfHx8MA
%3D%3D',

width: 1000,
height: 3000,

), Center(         child: Padding(
padding: EdgeInsets.all(20.0),           child:
Column(
mainAxisAlignment: MainAxisAlignment.center, // Center the content vertically

children: <Widget>[
 TextField(controller: _nameController,
decoration: InputDecoration(labelText: 'Enter Your Name'),
),

SizedBox(height: 20),
ElevatedButton(                onPressed:
() {
String customerName = _nameController.text;
Navigator.push( context,
MaterialPageRoute(builder: (context)
=>SeatSelectionPage(customerName),),); },
child: Text('Next'),
),],),),), ],),),);}}

class SeatSelectionPage extends StatefulWidget {   final
String customerName;
SeatSelectionPage(this.customerName);

@override

_SeatSelectionPageState createState() => _SeatSelectionPageState();

}class _SeatSelectionPageState extends State<SeatSelectionPage> {   List<String>
selectedSeats = []; void _onSeatSelected(String seatNumber) {

setState(() {      if
(selectedSeats.contains(seatNumber)) {
selectedSeats.remove(seatNumber); }
else {selectedSeats.add(seatNumber);

}});}

@override

Widget build(BuildContext context) {
return Scaffold(       appBar: AppBar(
title: Text('Bus Seat Booking'),         ),body:
Center(         child: Column(
mainAxisAlignment:
MainAxisAlignment.center,
children: <Widget>[
Text('Select your seats:',
style: TextStyle(fontSize: 20),    ),
SizedBox(height: 20), Wrap(
spacing: 10,                runSpacing: 10,
children: List.generate(30, (index) {
String seatNumber = 'Seat ${index +
1}';                  return
GestureDetector(
onTap: () {
_onSeatSelected(seatNumber)
;

},child: Container( width: 60,
height: 60,

 decoration: BoxDecoration(                    color:selectedSeats.contains(seatNumber
)

? Colors.red: Colors.green,
borderRadius: BorderRadius.circular(10),),

child: Center(child: Text(
seatNumber,
 style: TextStyle(
color: Colors.white,
fontWeight: FontWeight.bold,
), ),),), );}),),

SizedBox(height: 20),

Text('Selected Seats: ${selectedSeats.join(', ')}',
style: TextStyle(fontSize: 16),  ),
ElevatedButton(  onPressed: () { Navigator.push(

context,

MaterialPageRoute( builder: (context) =>
ConfirmationPage(
widget.customerName,
selectedSeats,
),),);},

child: Text('Confirm Booking'),

),],),),);}}
```

```
class ConfirmationPage extends StatelessWidget {
final String customerName;   final List<String> selectedSeats;
ConfirmationPage(this.customerName, this.selectedSeats);

@override

Widget build(BuildContext context) {
return Scaffold(       appBar: AppBar(
title: Text('Confirmation'), ),
body:Container(

color: Colors.black,
child: Stack( children:<Widget>[
Image.network('https://marketplace.canva.com/EAE6UgJ8HQI/1/0/1600w/canvacongrat
ulations-%28facebook-post%29-vD1uWKCgyeg.jpg',),

 Center(child: Column( mainAxisAlignment: MainAxisAlignment.center,            children:

<Widget>[

Text( 'Customer Name: $customerName',                style:

TextStyle(fontSize: 20, fontWeight: FontWeight.bold),

), Text( 'Seats Selected: ${selectedSeats.join(', ')}',                style:
TextStyle(fontSize: 18),
),

SizedBox(height: 20),
ElevatedButton(               onPressed:
() {
// Perform booking logic here
// You can navigate back to the home page if needed

// Navigator.popUntil(context, (route) => route.isFirst);
Navigator.push( context,
MaterialPageRoute(builder: (context) =>
ThankPage()),
//      widget.customerName,);},

child: Text('Confirm Booking'),

),],),),],),), );}}

class ThankPage extends StatefulWidget {

@override

_ThankPageState createState() => _ThankPageState();

}class _ThankPageState extends State<ThankPage> {

TextEditingController _nameController = TextEditingController();

@override

Widget build(BuildContext context) {

  return Scaffold(

appBar: AppBar title: Text('Your
Seat Is Confirmed'),
), body: Center( // Wrap the Column in a Center widget         child:
Padding(padding: EdgeInsets.all(20.0),
 child: Column( mainAxisAlignment: MainAxisAlignment.center, // Center the
content vertically             children: <Widget>[

Image.network('https://media.istockphoto.com/id/1132817705/photo/paintsplatter-thank

you.jpg?s=612x612&w=0&k=20&c=Yt8LgLuBfQTRqirHrlLk1iXaihXVw2AGPOv

HGlfG8w0=',),

TextField(


//  controller: _nameController,


//decoration: InputDecoration(labelText: 'Welcome to Flutter Platform'),


), SizedBox(height: 20),
ElevatedButton(                onPressed:
() {
String customerName = _nameController.text;


},child: Text('YOU ARE ALWAYS WELCOMED'),


),],),),),),);}}
```

OUTPUT SNAPSHOT:

Home page: This page plays video and next navigate to frontpage.

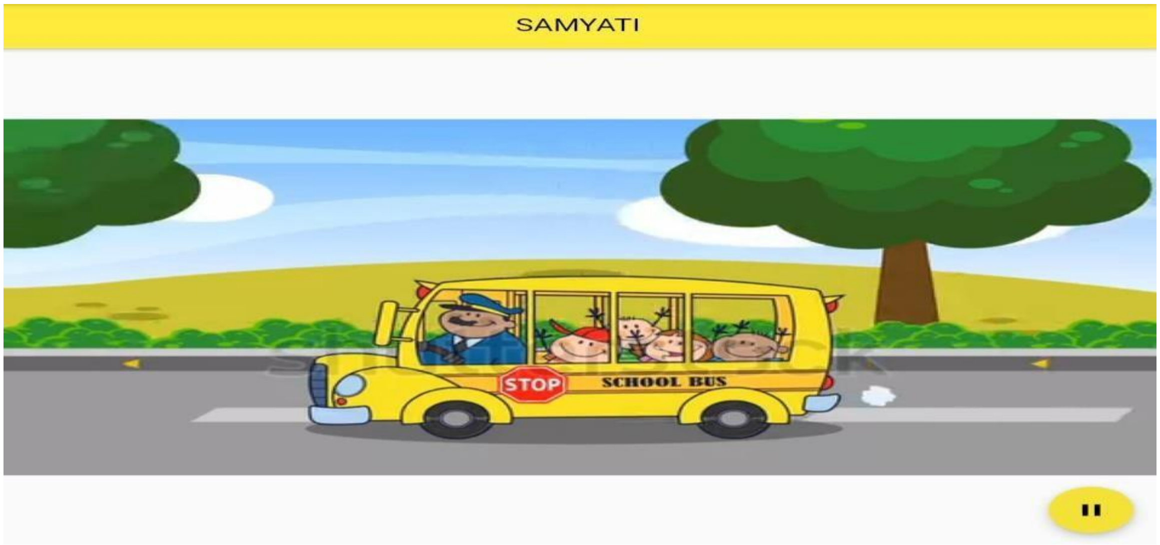


Figure 3.1

Booking Home page: Front page shows a button which leads us to customer name entry page.

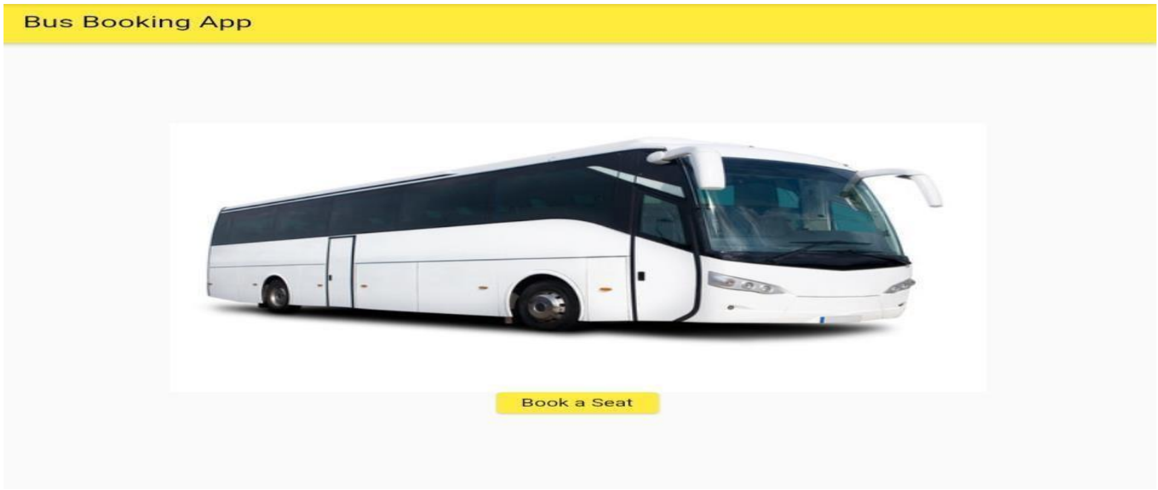
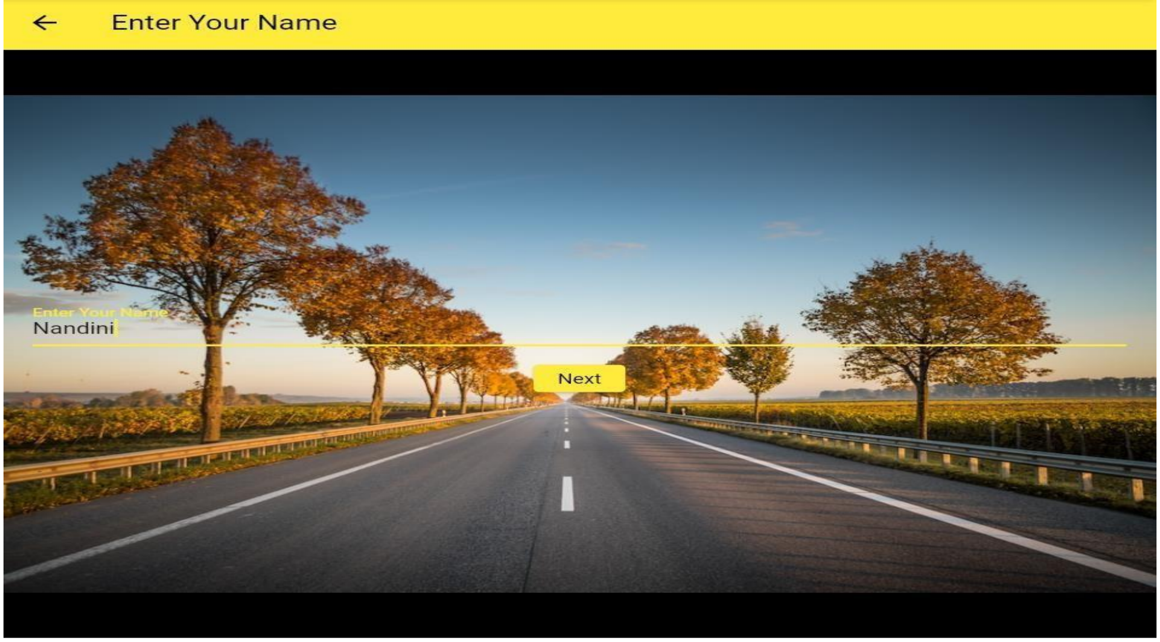


Figure 3.2

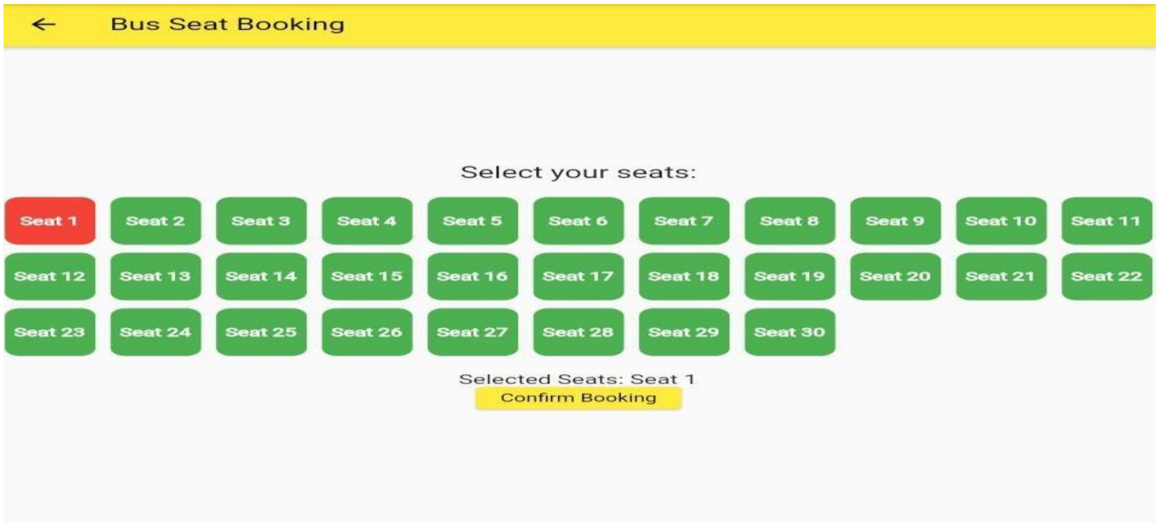
Entering Customer Name page: This page takes user name as input.



Enter Your Name: Nandini

Figure 3.3

Seat Selection page: This page allows user to select their desired seats.



Select your seats:

Selected Seats: Seat 1

Figure 3.4

Confirmation page: Your seats get booked or confirmed.

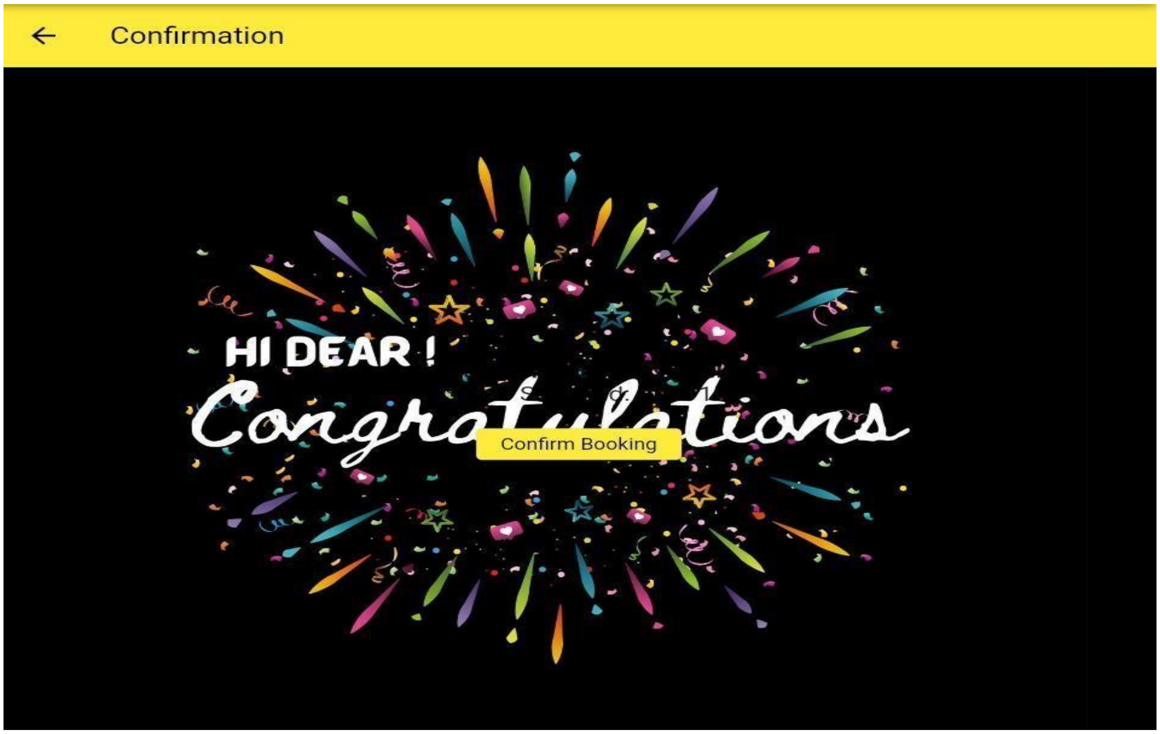


Figure 3.5

Thanking page: Thanking the customer for booking through our app.

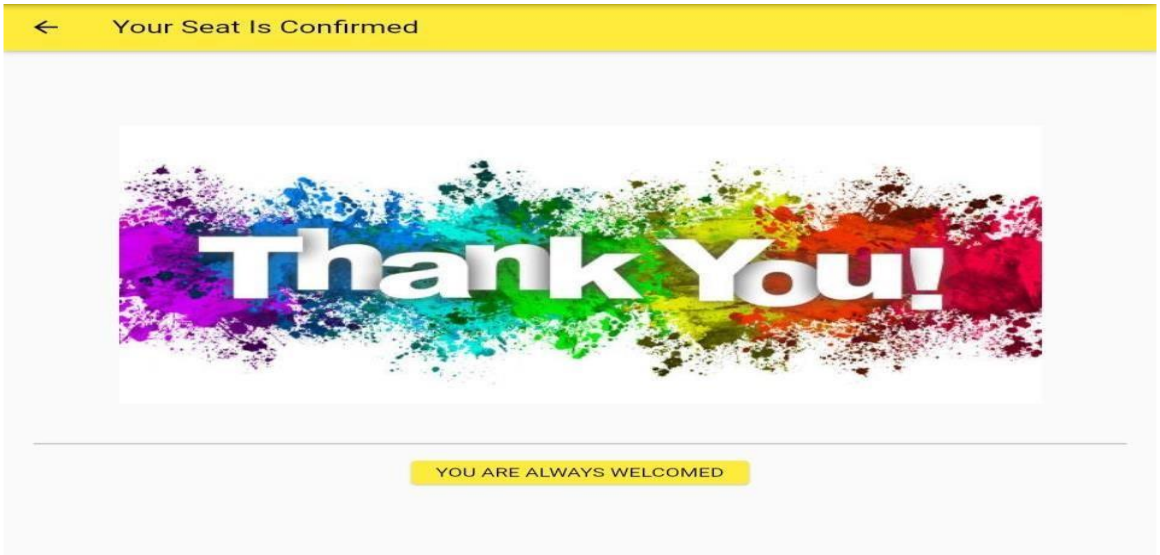


Figure 3.6

APPLICATION:

A bus seat booking app developed using Dart language and Flutter can be applied in various areas to enhance transportation services and streamline the booking process. Here are different areas where bus seat booking apps can be applied:

1. Intercity Travel: Booking seats for travel between different cities or states.

2. Airport Shuttle Services: Providing transportation between airports and nearby cities.

3. Corporate Commuting: Allowing employees to book seats for daily commuting between home and the workplace.

4. Tourism and Sightseeing: Organizing city tours, sightseeing trips, and guided tours for tourists.

5. Educational Institutions: Facilitating transportation for students and staff of schools, colleges, and universities.

6. Event Transportation: Offering transportation services for events like conferences, concerts, weddings, and sports events.

7. Hotel Shuttle Services: Providing shuttle services for hotel guests, connecting airports, tourist attractions, and other points of interest.

8. Group Travel: Booking entire buses for group travel, corporate outings, or family events.

9. Charter Services: Offering charter bus services for private events, parties, and special occasions.

10. Rural Connectivity: Connecting rural areas with urban centers, improving accessibility for residents.

11. Specialized Transportation: Providing specialized buses, such as luxury coaches, sleeper buses, and eco-friendly options.

12. Bus Pooling and Ride-Sharing: Allowing users to share bus rides, reducing costs and promoting eco-friendly commuting.

13. Travel Agencies: Travel agencies can integrate bus booking services into their offerings, providing a complete travel solution for customers.

14. Community Transportation: Serving community needs, such as senior citizen transport or community events.

15. Hospital and Healthcare Services: Facilitating transportation for patients, staff, and visitors to hospitals and healthcare facilities.

16. Festival and Event Transport: Providing temporary transportation solutions for festivals, fairs, and large events.

By applying bus seat booking apps in these diverse areas, businesses and organizations can cater to specific transportation needs, making travel more convenient, efficient, and accessible for people in various contexts.

4 .CONCLUSION:

As travelling has become a part our life, we need to travel from one part of globe to the different location because of work, education, business, family and friends. This seat booking app makes this job easy to travel by selecting our own transport mode and our desired seats within a second using this app. In this busy world people having no free time to get bus and go can just book their seats and can also reserve it for the day they need to travel.

It just takes a second for them to book a seat with tip of their finger. In conclusion, the development and implementation of the bus seat booking app have significantly improved the efficiency and convenience of the booking process for passengers and bus operators alike. The app's user-friendly interface, seamless payment system, and real-time seat availability updates have enhanced the overall travel experience for users.

Additionally, the app has proven to be a cost-effective solution for bus operators, reducing manual workload and streamlining operations. The feedback from both passengers and operators has been overwhelmingly positive, highlighting the app's positive impact on the transportation industry.

Moving forward, continuous updates and improvements based on user feedback will be essential to ensure the app remains competitive and continues to meet the evolving needs of its users. Overall, the bus seat booking app stands as a testament to the power of technology in transforming traditional processes and enhancing customer satisfaction in the travel sector.