

SMART VEHICLE PARKING SYSTEM FOR ANDROID USER

BY

FATEMA JAHAN JUTHI
ID: 151-15-5180

FABLIHA AFIA
ID: 151-15-5194

AND

Md. AHSAN NAEEM RAHMAN MOLLAH
ID: 151-15-4929

This Report Presented in Partial Fulfilment of the Requirements for the Degree of
Bachelor of Science in Computer Science and Engineering

Supervised By

MS. FARAH SHARMIN
Senior Lecturer
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2018

APPROVAL

This Project titled “**SMART VEHICLE PARKING SYSTEM FOR ANDROID USER**”, submitted by Fatema Jahan, Juthi ID No: 151-15-5180, Fabliha Afia , ID: 151-15-5194, Md. Ahsan Naeem Rahman Mollah, ID: 151-15-4929 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfilment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 10th December, 2018.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman

Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Md. Zahid Hasan
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. Mohammad Shorif Uddin
Professor

Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Ms. Farah Sharmin, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Farah Sharmin

Ms. Farah Sharmin

Senior Lecturer

Department of CSE

Daffodil International University

Submitted by:

Fatema Jahan Juthi

ID: 151-15-5180

Department of CSE

Daffodil International University

Fabliha Afia

ID: 151-15-5194

Department of CSE

Daffodil International University

Md. Ahsan Naeem Rahman Mollah

ID: 151-15-4929

Department of CSE

Daffodil International University

ACKNOWLEDGMENT

First let us thank Almighty Allah who allowed us to finish this genuine project successfully.

Second and foremost, we would like to express heartfelt appreciation to our supervisor **Ms. Farah Sharmin**, Computer Science & Engineering, Daffodil International University for her constant support during our study; She has inspired us a lot to work in this project, her motivation encouraged us extremely to complete our project.

We would like to thank the authority of Daffodil International University for providing us a good environment and facilities such as computer laboratory to complete this project with software which we need during the process.

We would like to thank our entire course mate in Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents who have never doubted our efforts and constantly supported us throughout this journey.

ABSTRACT

A proper vacant and secure parking place is the major need for most of us especially those who use their own vehicles for their daily movement. Also when a person goes to a new area, it becomes difficult for him/her to find a parking place. For those reasons, many of us park his vehicles here and there on the road. It causes traffic congestion. Moreover, due to parking, a vehicle on the road sometimes vehicle got stolen. So an application for searching a parking area is needed. Some applications are available in developed countries but almost non-existent in developing countries, where resources are lacking. Therefore, in this project, we made an android based mobile application, “Android-Based Application of Smart Parking System” to help users such as general people to increase the ability to get to know their parking place easily by using a simple Smart Parking Application. General peoples can see their parking spot details and park their vehicles and they can access from anywhere with an internet connection using their smartphone. Thus this application will reduce the troubles and problems of finding the right parking place. This is an android based mobile application so an android based mobile phone is needed to use this application. The application is also connected with the Google Map. The automated nearby parking area will be suggested using GPS technology.

TABLE OF CONTENTS

CONTENTS	PAGE
Approval	ii
Declaration	iii
Acknowledgements	iv
Abstract	v
List of Figures	viii
CHAPTER 1: INTRODUCTION	1-2
1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	1
1.4 Report Layout	2
CHAPTER 2: BACKGROUND	3-5
2.1 Introduction	3
2.2 Related Works	3
2.3 Comparative Studies	4
2.4 Scope of the Problem	5
2.5 Challenges	5
CHAPTER 3: REQUIREMENT SPECIFICATION	6-10
3.1 Smart Parking Process Modelling	6
3.2 Requirement Collection and Analysis	7
3.3 Use Case Modelling and Description	8

3.4 Logical Data Model	9
3.5 Design Requirements	9
CHAPTER 4: DESIGN SPECIFICATION	11-20
4.1 Front-end Design	11
4.2 Back-end Design	19
4.3 Interaction Design and UX	19
4.4 Implementation Requirements	20
CHAPTER 5: IMPLEMENTATION AND TESTING	21-22
5.1 Introduction to Tools and Technologies	21
5.2 Key features of the Proposed System	21
CHAPTER 6: CONCLUSION AND FUTURE SCOPE	23
6.1 Future Scope to Work	23
6.2 Discussion and Conclusion	23
REFERENCES	24
APPENDICES	25-52
Appendix A: Project Reflect	25
Appendix B: Related Codes	26

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.1.1: Smart Parking Process Model	6
Figure 3.3.1: Use Case Model	8
Figure 3.4.1: System Diagram	10
Figure 4.1.1: Home Page Screen	11
Figure 4.1.2: Select District Screen	12
Figure 4.1.3: Searching District Screen	13
Figure 4.1.4: Select Area Screen	14
Figure 4.1.5: Searching Area Screen	15
Figure 4.1.6: Select Vehicle Type Screen	16
Figure 4.1.7: Show Location Screen	17
Figure 4.1.8: Show Cost Screen	18
Figure 4.3.1: UX	19

CHAPTER 1

INTRODUCTION

1.1 Introduction

Finding a vacant and secure parking place in a metropolitan area is the major concern for most of us and it takes a lot of time to find a proper and secure place for parking. Especially when a person goes to a new area, it becomes difficult for him/her to find a parking place. For those reasons, many of us park his vehicles here and there on the road. It causes traffic congestion. Parking in the wrong places creates problems for pedestrians and many accidents also occur like stealing cars or parts of cars. Stations, near airport, theatre etc. as an efficient means to park. Google Wallet can used to make secure payments fast and convenient. A recent survey[1] shows that at the time of search hour, the traffic made by vehicles finding for a proper parking place takes almost 40% of the total traffic jam.

1.2 Motivation

In our country, a large number of people use their own vehicles for their daily movement. To park those vehicles they need a proper and secure parking place. Moreover, when a person goes to an unknown place they get in trouble with finding a proper parking place. As the place is unknown to them they don't have a proper idea about the parking spots of that particular area. As a result, they park their vehicles here and there on the road. This can cause a traffic jam. Also due to park, a vehicle on the roadside can cause the vehicle got stolen. Sometimes they get police case due to park the vehicle on the roadside. So a proper and secure parking spot is needed. So those are the reasons why we became highly motivated about this application.

1.3 Objective

This project will have contains the following features.

- Providing smart mobile application for parking vehicles
- Finding a parking spot at anywhere and at anytime
- Enhance the security while parking vehicles
- Cost efficient system
- Simple system interface
- Reduced traffic

1.4 Report Layout

Chapter 1: Introduction

In this chapter, we have discussed the introduction, motivation, and objectives of the project. Later followed by background.

Chapter 2: Background

We discuss the background circumstances of our project. We also talk about the related works, comparison to other candidate systems, the scope of the problem and the challenges of the project.

Chapter 3: Requirement Specification

This chapter is all about the requirements like business process modeling, the requirement collection and analysis, the use case model of the project and their description, the logical data model and the design requirements.

Chapter 4: Design Specification

In this chapter, there are all the designs of the project. System design, back-end design, interaction design, and UX and the implementation requirements

Chapter 5: Implementation and Testing

This chapter contains the implementation of the database, front-end designs, interactions, test implementation and the test results of the project.

Chapter 6: Conclusion and Future Scope

We discussed the conclusion and the scope for further developments which pretty much derive about the project.

Chapter 2

BACKGROUND

2.1 Introduction

Smart Vehicle Parking System for Android User is an android based application and to develop the application it requires GPS(Global Positioning System), smartphone, google map and internet connection for using this application.

In our project, GPS [2] is a space-based route framework that gives area and time data in every climate condition, anyplace on or close to the Earth where there is an unhampered observable pathway to at least four GPS satellites.

Google Maps[3] is a web mapping administration created by Google. It offers satellite symbolism, road maps, and 360° all-encompassing perspectives of boulevards, ongoing movement conditions, and course making arrangements for going by foot, auto, bike, or open transportation.

2.2 Related works

We have searched on Google by the term “smart parking applications” and we found different applications related to smart parking application.

“4Park” a smart parking application [4] that shows the nearby parking spots by using Google Map. The payment system is also included in this application.

Huawei proposed a solution called NB-IoT enables smart parking system [5]. This system will collect parking data and send it to the service provider. This system will include fee collection system and also will provide real-time parking space information.

A hybrid smart parking system by “Street line solution” [6], that uses machine learning techniques to combine several data from different sources for real-time parking instructions.

Zeydin Pala et. al. [7] proposed an RFID based parking management system. RFID innovation is an automation innovation with primary segments as RFID readers, names, PCs, obstructions, programming and so forth. The product is for administration, controlling, exchange detailing and activity undertakings for parking garages. This vehicular information

is brought from the database to check a vehicle when it enters the parking garage utilizing RFID readers. At the point when a vehicle needs to look at, the confirmation is finished by investigating registration subtle elements. The drivers will be informed about the accessibility of parking garages at the passageway itself. Thus, no time is squandered searching for stop space.

ParkNet is a portable framework [8] involving vehicles that gather parking spot inhabitance data by driving by. The vehicle incorporates a GPS collector and traveler side confronting ultrasonic rangefinder to decide empty parking spots. The information gathered is coordinated at the focal server delivering a constant guide of stopping accessibility subsequently serving user's demands. With the end goal to accomplish an exactness of area, an ecological fingerprinting approach is contrived. It makes utilization of a GPS module and ultrasonic sensors for a conclusion to end correspondence. A constant guide is created to mirror the inhabitancies as for a timestamp. The downsides are primarily because of constraints of sensors and multifaceted nature associated with instances of multi-path stopping.

2.3 Comparative Studies

In Bangladesh currently, there are no smart parking mobile applications. In "4park" smart parking application they included payment system but we haven't included any payment system because our system will only find the nearest parking spot for the user.

As our application is an android based application, we don't need any RFID based system. Our system will provide the nearest parking spot using Google map and we don't need any data analysis. So machine learning techniques are not used in our application.

This application is very easy to use one can just download it and can use and he/she doesn't need to pay for using this application.

2.4 Scope of the Problem

The problem will arise when one wants to add a new parking spot but they are can't because of no manual adding option in our application. A person can only find the nearest parking spot but he/she can't check whether there is any available parking spot or not.

2.5 Challenges

The user of this application is general peoples. The user of this application is those who use the personal vehicles for their daily use. They can easily use this application to find their nearby parking place. They faced challenges in many things. Searching all the parking spaces and collecting all the necessary information about all the parking place is quite difficult.

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 Smart Vehicle Parking System Android Application:

In our project, we have made an android application and it is a Smart Parking Application. It is a user- friendly application. Using this android application user can easily find out the parking area and can park their vehicles in a fixed parking spot.

Smart Vehicle Parking System for Android User Process Modelling:

In our project, we have made a processing model. By that, you can understand the whole process of our project Smart Vehicle Parking System For Android User. In this model you can have a total understanding of the total process that how the system is working.

So, the model is giving below:

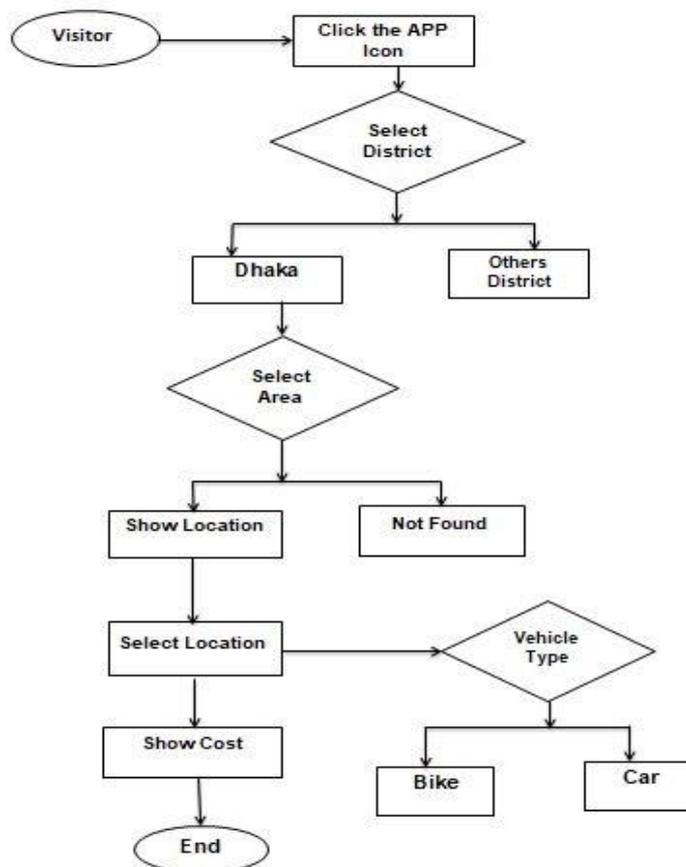


Figure 3.1.1: Smart Vehicle Parking System for Android User Process Model

3.2 Requirement Collection and Analysis:

Collecting requirements and analyzing them for any project is the most important part because the whole system is based on this knowledge. No project can be started without planning the project requirements. Majority of the failed projects have failed due to the wrong or insufficient requirements gathering. If the project team fails to capture all the necessary requirements for the solution, the project will be running with a risk. This may load many problems in the future such as a result the full project can be damaged. After collecting all the requirements and analyzing them, we have found the necessary features that our application needs. It provides all the specifications of the system in details which is very effective for every user.

The current problem is found out to a parking place. But using this system it will be very simple to find out parking place easily. Using this system, we can save out time which was wasted for finding a parking place. It also reduces traffic jam.

There are many different methods in requirement collection and analysis. Every method will have benefits and difficulty also. Cost and time are the two important factors when choosing which method will be used. The method in collecting the requirements may vary depending on the situation but using the various methods it will be very helpful to collect the complete requirements.

Smart Parking Android Application project requirements:-

1. User.
2. An android phone.
3. User-friendly.
4. Time convenience.
5. The System should be work faster in response.
6. Reliable system.
7. Reduce traffic jam.
8. Show the available space.

Hardware Requirements:

- An Android Phone

Software Requirements:

- Operating System: Android Studio.
- Programming Language: Java.
- Google Map.

Analysis:

After analyzing the current situation of our country we decided to make this smart parking android app. our country is one of the overpopulated countries. Here traffic jam is a common problem and we make this traffic jam by parking our vehicle on the roadside. To overcome this problem we need to park our vehicle in a specific vicinity and our app will help to do it. It is a very efficient and reliable app.

3.3 Use Case Modelling and Description:

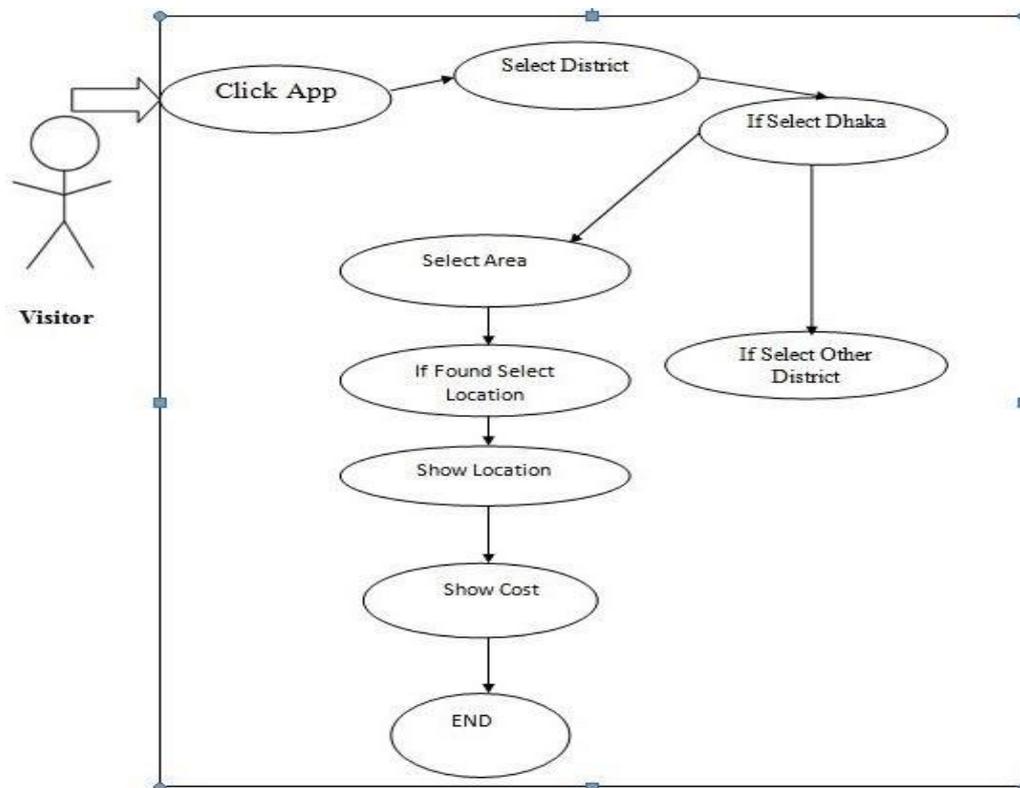


Figure 3.3.1: Use Case of Smart Vehicle Parking System for Android User

Description of Use Case Diagram:

Over the utilization of use case format, there is a working system amongst on-screen character.

Actor: User.

Main Flow of the Diagram:

The user can park their vehicle following these steps:

At first, the user has to click the app. Then have to select the district. After selecting the district there will be two options. If the user selects Dhaka, then it will show the area. After completing it user has to select the location then the location will be shown. If the parking location is available then it will show the parking cost and if the user selects other option it will not find.

3.4 Logical Data Model

Logical Data Modelling is a process used to define and analyze the requirement needed to support the business processes within the scope of corresponding information systems in organizations. The Entity-Relationship model or Entity-Relationship diagram (ERD) is a logical data model; it includes the entity, attributes, table, and relationships.

3.5 Design Requirements:

Design requirements state is the important characteristics that our design must meet in order to be successful. It makes an application more users friendly. To make an application straightforward and easy to understand we must need Design requirement. For this project, the following goals have been saved in thought while designing the system:

Make App Simple and Flexible For Users:

The parking app must be greater adaptable to users. As it is flexible, so the users can expect lots of a stretch experiencing the way towards getting to the app.

Make the app user-friendly:

It is a user-friendly application. Without any hassle, the user can use it.

Efficiency:

This app is capable of guiding the users to park their vehicle safely and quickly. It also gives the space idea of the garage.[9] The performance metrics that we have considered in our proposed work is about finding vacant space for the park the vehicle. Here the information about the free space then it sends that information on the users' android phone through the android application.

Highly available

Maintenance:

The app should be more reliable than any other parking app for maintenance of the app in future.

Now we will show the total system design through a context diagram.

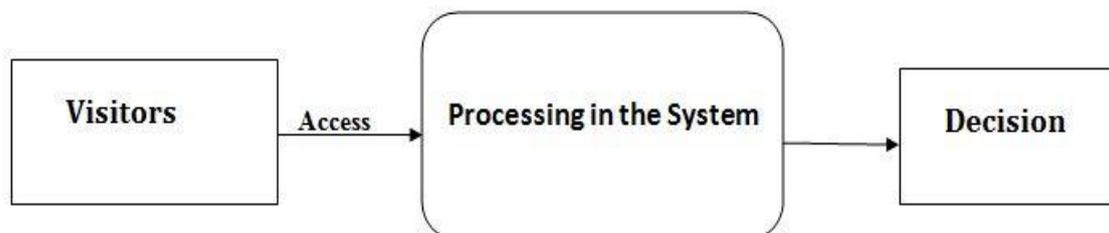


Figure 3.4.1: System Diagram

CHAPTER 4

DESIGN SPECIFICATION

DESIGN SPECIFICATION

4.1 Front-end Design

The front-end plan of a device requires the creation of the Java code that makes up a user interface.

The purpose of designing a machine is to make sure that when a consumer accesses a system they see the facts in a structure that is convenient to read and understand.

The purpose of front-end layout is to enhance the performance and accessibility of a system.

Our device consists of the following front-end design.



Figure 4.1.1: Home page screen

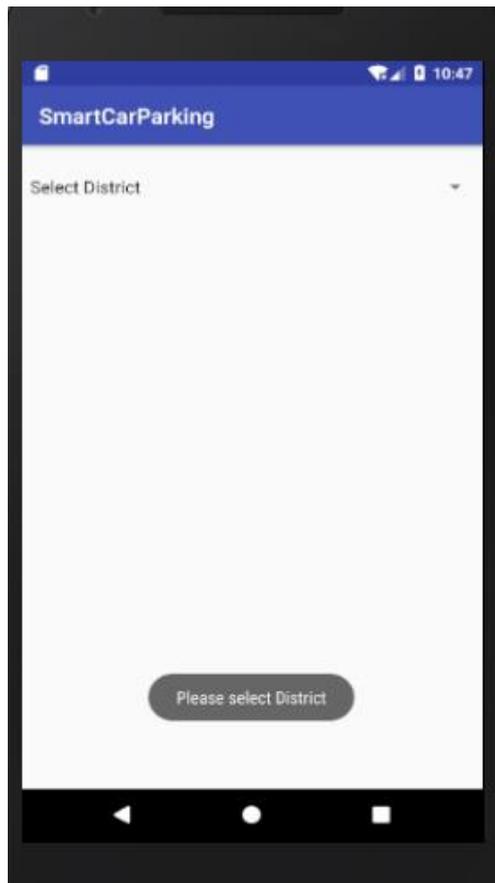


Figure 4.1.2: Select District Screen

This is the first step of our application. In this step, there will be only one district selection option for the user. Users have to click the district option for choosing the district.

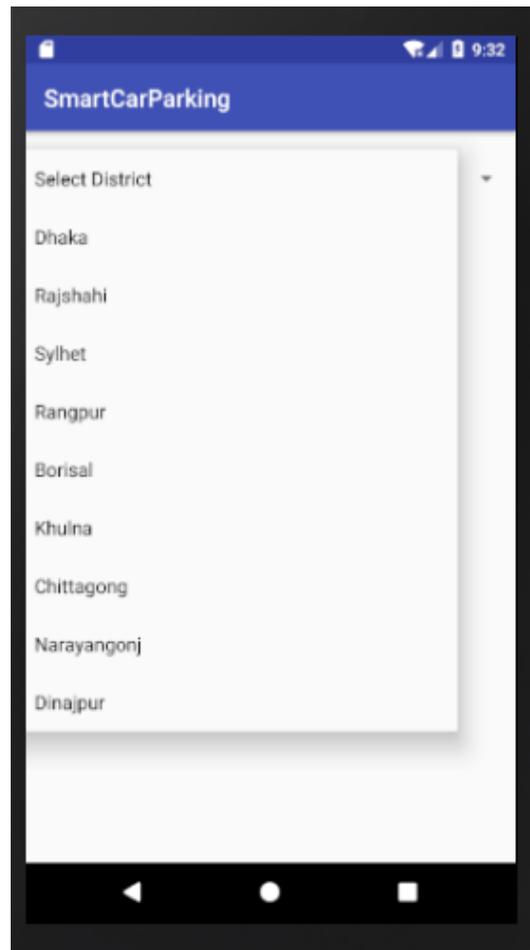


Figure 4.1.3: Searching District Screen

After clicking the select district option there will be showing some district name like Dhaka, Rajshahi, Sylhet etc. Then users have to choose the district name for go to the next option



Figure 4.1.4: Select Area Screen

When the user comes on this step that time users will see the district option. After showing the district name users will click the district. Then clicking the district name user can see all area which is included in Dhaka district. Users can see the area inside Dhaka like Dhanmondi, Gulshan, Mirpur, Science Lab etc.

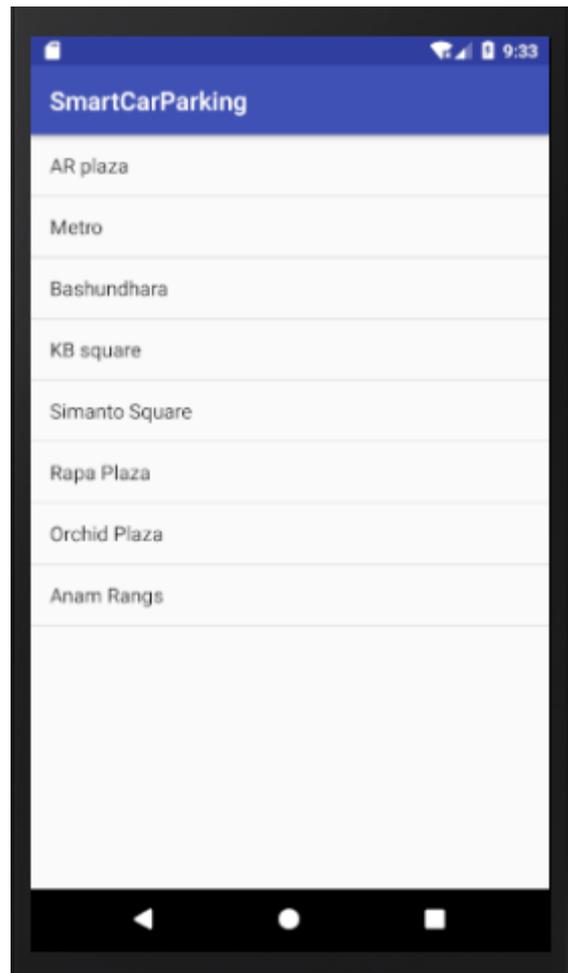


Figure 4.1.5: Searching Area Screen

When users choose Dhanmondi area then will be shown the parking spot near Dhanmondi like AR Plaza, Metro shopping mall, Rapa Plaza, Shimanto square etc. Then the users can easily choose their desired spot.

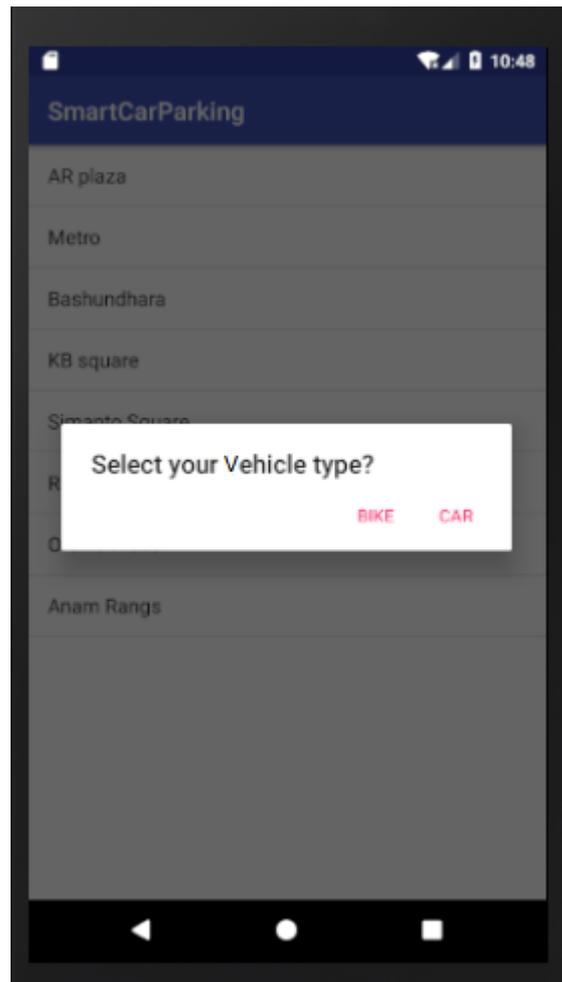


Figure 4.1.6: Select vehicle type screen

Here we can select our vehicle type and there are two options bike and car. If we want to park car then we need to click car and if we want to park bike then we need to click the bike option.

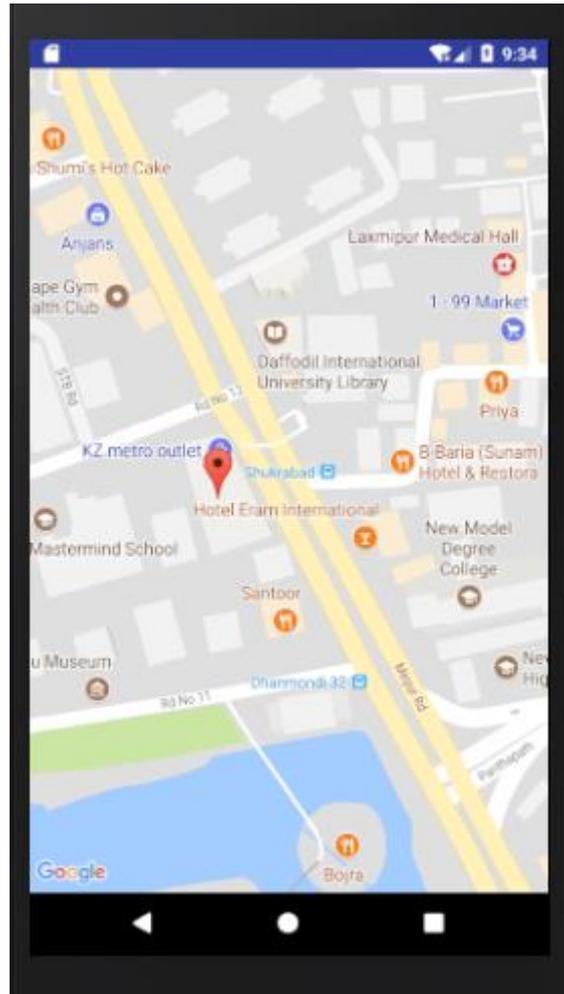


Figure 4.1.7: Show Location Screen

In our project we use GPS. GPS means Global Positioning System. It will help the user to find out the nearest parking spot.

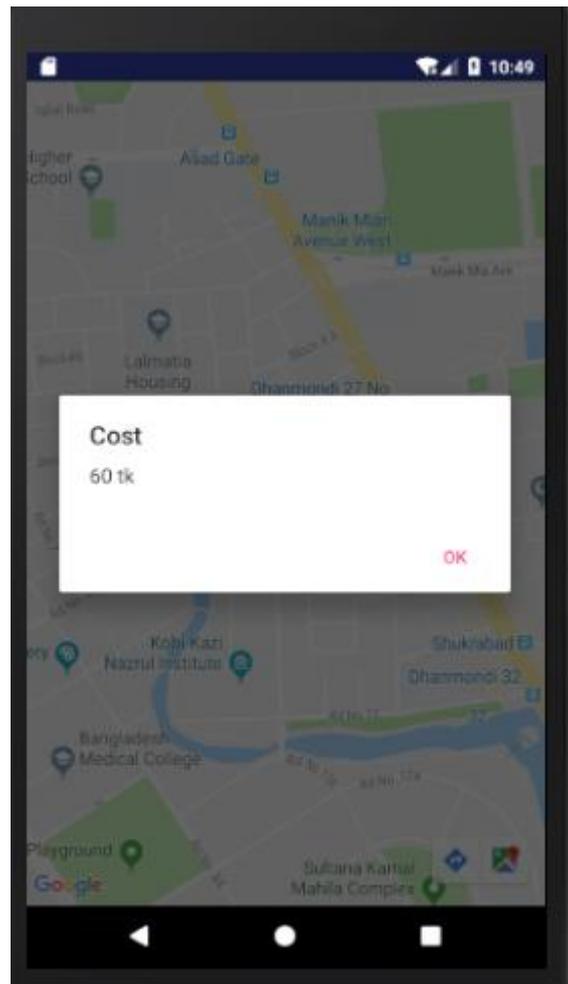


Figure 4.1.8: Show Cost Screen

This is the final step of our application. When the users click the area then cost will be shown against the selective vehicle. The cost will vary depends on the vehicle, area types and parking hour.

4.2 Back-end Design

The back-end design is, actually how the site works, updates and changes. The user doesn't need to know the back-end to use the system. The user only knows how to use the system. This refers to everything the user can't see in the browser, like databases and servers. Back-end design deals with things like security, structure and content management.

4.3 Interaction Design and UX

Interaction design is an important part with user experience (UX). Interaction design is the design of the interaction between users and app. The goal of interaction design is to create an app that helps the user to achieve their objectives in the best way possible.

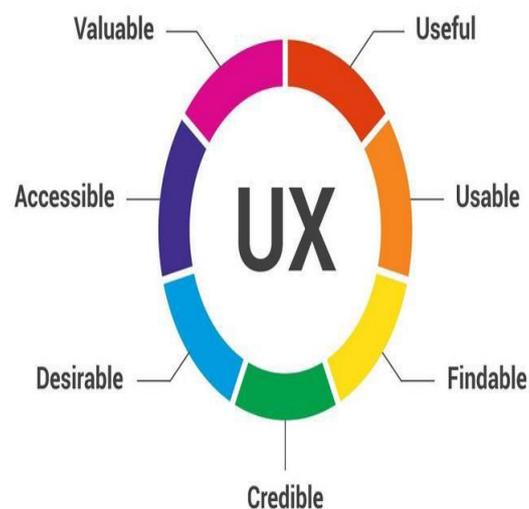


Figure 4.3.1: UX (User Experience)

Home Page Interaction Design and UX:

We have designed a home page for a user. The user can only use this function like selecting the district, area, location. It's an easy system to understand. So, the user will not have any confusion about the application.

User Page Interaction Design and UX:

A user can be anyone who has an android phone. This page shows the district, area, location, available space and cost to the user. So, the user will not have any trouble using the application.

4.4 Implementation Requirements

Implementation requirement is the process of executing a plan, a method, or any design, idea, model for building a system. It is the action that must follow any preliminary thinking in order for something to actually happen.

Our system is developed in such a way that it can readily be modified in response to end-user.

The list of implementation requirement is given below:

- Secure.
- User-friendly
- Time convenience
- Reliable system.

CHAPTER 5

Results

5.1 Introduction to Tools and Technologies

- Java
- Android Studio
- Google Map

5.2 Key features of the Proposed System

The proposed framework has the going with fundamental highlight.

5.2.1 Efficiency

The effectiveness of any framework is worried about the base preparing time and also the ideal utilizes framework assets in outlining the proposed framework: the productivity factor has been pondering over well.

5.2.2 User-friendly interface

The interface of our application will be easy to understand with the goal that a typical client can utilize it effortlessly. Users can jump from one section to another freely.

5.2.3 Data Security and integrity

Data security and integrity is our highest priority. We will ensure that give information of each and every client must stay private and never be imperiled for this reason we will utilize feasible security components.

5.2.4 Flexibility

Our application arrangement is proposed to give,

- Innovation and control

5.2.5 Extensibility

Key highlights of the proposed arrangements would be its extensibility. Our answer empowers another level of remote automation for the improvement of humankind.

5.2.6 Scalability Adaptability can be characterized as the ease with which a framework or part can be altered to fit the issue.

CHAPTER 6

Conclusion

6.1 Future Scope to Work:

The “Smart Car Parking” Application can be developed for other popular mobile operating systems. In the future, our application can be implemented on the existing operating systems like iOS, Windows and BlackBerry also on the upcoming and promising operating systems like Firefox OS, Jolla, and Tizen. Stations, near airport, theatre, etc. as an efficient means to park. Google Wallet can be used to make secure payments fast and convenient.

6.2 Discussion and Conclusion:

Bangladesh is a developing country in the world. In this country, there are eight divisions and sixty four districts. In each division there are several cities, there are a lot of shopping-mall, hospitals, schools, colleges etc. In our country the number of vehicles is increasing day by day. As a result, we are facing so many difficulties. For example, people park their vehicles in front of hospitals, schools, colleges, and shopping-mall. For that reason, the traffic jam creates on the road. That’s why running car cannot move to the disruption. The enforcement agencies of our country have to face many obstacles. Our application is deigned to make these bugs off. So that people can easily find the parking places.

Reference:

- [1] I. Ajzen, "Belief , attitude , intention and behaviour : An introduction to theory and research," no. May 1975, 2015.
- [2] "GPS." [Online], Available at: https://en.wikipedia.org/wiki/Global_Positioning_System, [Last Accessed on 08-Sep-2018 at 11:35am].
- [3] "Googlemap." [Online], Available at: <https://www.google.com.bd/maps/@22.3765595,91.8393838,15z>. [Last Accessed on 08-Sep-2018 12:00 pm].
- [4] "4-park-smart-parking-app" [Online], Available at: <http://smartparkingsystems.com/en/4-park-smart-parking-app/> [Last Accessed on 10-Sep-2018 12:30 pm].
- [5] "IoT Enables Smart Parking" [Online], Available at: <https://www.huawei.com/minisite/iot/en/smart-parking>. [Last Accessed on 10-Sep-2018 12:45 pm].
- [6] "The Street line Solution" [Online], Available at: <https://www.streetline.com/how-it-works/> [Last Accessed on 11-Sep-2018 10:00 am].
- [7] Z. Pala and N. Inanc, "Smart parking applications using RFID technology" in 1st Annual Eurasia RFID conference, September 2007
- [8] Brabham, "Crowdsourcing as a model for problem solving: An introduction and Cases" Convergence: The International Journal of Research into New Media Technological Studies.
- [9] "Design and Implementation of a Prototype Smart parking. (SPARK) System Using Sensor Networks and Android App"[2013-2014]

APPENDICES

Appendix A: Project Reflection

The purpose of this appendix is to provide an introduction to Project Reflection. The research project was a challenging and enjoyable experience typical of the course as a whole.

We have had little exposure to group work at a university so it was a nice change to be part of an effective and dynamic team. I think we complemented one another quite well both in bringing together interdisciplinary perspectives and in balancing the work at hand.

The experience taught us that planning and crafting responses take a longer time in teams than on your own. The extensive effort required was ultimately a good thing. When working alone, you can end up with a result that is identical to your initial plans. In our group, we were constantly developing and refining one another's ideas. It was fascinating just how productive our group meetings were. The time seemed to fly and yet we always got a lot done and managed to help another along the way towards the end point of having a substantive policy.

Appendix B: Related Codes

We had to write many codes in order to finish this project. We cannot show all our codes in our report. But there are some important parts which should be included to in our report in order to understand the functionality or how it started working.

1. activity_welcome.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:background="#ffffff"
    tools:context="com.example.mollah.smartcarparking.WelcomeActivity">

    <ImageView
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:src="@drawable/carpariking"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome to Smart Parking"
        android:textSize="25dp"
        android:textColor="#000000"
        android:textStyle="bold"
        android:layout_marginTop="10dp"
        android:layout_gravity="center"/>
    <Button
        android:id="@+id/nextButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Next"
        android:textColor="#000000"
        android:layout_marginTop="30dp"
        android:background="#d2d2d2"
        android:layout_marginBottom="20dp"/>

</LinearLayout>
```

2. activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"
```

```

android:layout_width="match_parent"

android:layout_height="match_parent"

android:orientation="vertical"

tools:context="com.example.mollah.smartcarparking.MainActivity">

<Spinner

    android:id="@+id/districSpinner"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:gravity="center"

    android:layout_marginTop="15dp"/>

<Spinner

    android:id="@+id/districtArea"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:gravity="center"

    android:layout_marginTop="15dp"

    android:visibility="gone"/>

</LinearLayout>

```

3. activity_maps.xml:

```

<fragment xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:map="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

    android:id="@+id/map"

    android:name="com.google.android.gms.maps.SupportMapFragment"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

```

```
tools:context="com.example.mollah.smartcarparking.MapsActivity" />
```

4. activity_places.xml:

```
<?xml version="1.0" encoding="utf-8"?>  
  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
  
    xmlns:tools="http://schemas.android.com/tools"  
  
    android:layout_width="match_parent"  
  
    android:layout_height="match_parent"  
  
    android:orientation="vertical"  
  
    tools:context="com.example.mollah.smartcarparking.places">  
  
    <ListView  
  
        android:id="@+id/parkingplaceslist"  
  
        android:layout_width="match_parent"  
  
        android:layout_height="wrap_content"/>  
  
</LinearLayout>
```

5. WelcomeActivity.java:

```
package com.example.mollah.smartcarparking;  
  
import android.content.Intent;  
  
import android.support.v7.app.AppCompatActivity;  
  
import android.os.Bundle;  
  
import android.view.View;  
  
import android.widget.Button;  
  
  
public class WelcomeActivity extends AppCompatActivity {  
  
    private Button nextBtn;  
  
    @Override
```

```

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_welcome);

    nextBtn = (Button) findViewById(R.id.nextButton);

    nextBtn.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View view) {

            Intent intent = new Intent(WelcomeActivity.this , MainActivity.class);

            startActivity(intent);

        }

    });

}
}

```

6. MainActivity.java:

```

package com.example.mollah.smartcarparking;

import android.content.Intent;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.ArrayAdapter;

import android.widget.Spinner;

import android.widget.Toast;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

```

```

private Spinner districSpinner, districtArea;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    districSpinner = (Spinner)findViewById(R.id.districtSpinner);

    districtArea = (Spinner)findViewById(R.id.districtArea);

    ArrayList<String> districtArray = new ArrayList<>();

    districtArray.add("Select District");

    districtArray.add("Dhaka");

    districtArray.add("Rajshahi");

    districtArray.add("Sylhet");

    districtArray.add("Rangpur");

    districtArray.add("Borisa");

    districtArray.add("Khulna");

    districtArray.add("Chittagong");

    districtArray.add("Narayangonj");

    districtArray.add("Dinajpur");

    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
        R.layout.support_simple_spinner_dropdown_item, districtArray);

    districSpinner.setAdapter(adapter);

    districSpinner.setOnItemClickListener(new AdapterView.OnItemClickListener() {

        @Override

        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

            switch (i){

                case 0:

```

```

Toast.makeText(MainActivity.this, "Please select District", Toast.LENGTH_LONG).show();

break;

case 1:

    districtArea.setVisibility(View.VISIBLE);

    final ArrayList<String> districtAreaArray = new ArrayList<>();

    districtAreaArray.add("Select area");

    districtAreaArray.add("Dhanmondi");

    districtAreaArray.add("Science Lab");

    districtAreaArray.add("Mirpur");

    districtAreaArray.add("Gulshan");

    districtAreaArray.add("Bashundhara Residential Area");

    ArrayAdapter<String> adapter = new ArrayAdapter<String>(MainActivity.this,
R.layout.support_simple_spinner_dropdown_item, districtAreaArray);

    districtArea.setAdapter(adapter);

    districtArea.setOnItemClickListener(new AdapterView.OnItemClickListener() {

        @Override

        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

            if ( i != 0){

                Intent intent = new Intent(MainActivity.this, places.class);

                intent.putExtra("districtArea", districtArea.getSelectedItem().toString());

                startActivity(intent);

            }

            else{

                Toast.makeText(getApplicationContext(), "Select Area",
Toast.LENGTH_LONG).show();

```

```

    }

}

@Override

public void onNothingSelected(AdapterView<?> adapterView) {

    Toast.makeText(MainActivity.this, "Please select
District",Toast.LENGTH_LONG).show();

}

});

break;

case 2:

    Toast.makeText(MainActivity.this, "work perfectly this is
Rajshahi",Toast.LENGTH_LONG).show();

    break;

case 3:

    Toast.makeText(MainActivity.this, "work perfectly this is
sylhet",Toast.LENGTH_LONG).show();

    break;

case 4:

    Toast.makeText(MainActivity.this, "work perfectly this is
Rnagpur",Toast.LENGTH_LONG).show();

    break;

}

}

@Override

```

```

        public void onNothingSelected(AdapterView<?> adapterView) {

            Toast.makeText(MainActivity.this, "Please select District",Toast.LENGTH_LONG).show();

        }

    });

}

}

```

7. MapsActivity.java:

```

package com.example.mollah.smartcarparking

import android.content.Context;

import android.os.Bundle;

import android.support.v4.app.FragmentActivity;

import android.support.v7.app.AlertDialog;

import com.google.android.gms.maps.CameraUpdateFactory;

import com.google.android.gms.maps.GoogleMap;

import com.google.android.gms.maps.OnMapReadyCallback;

import com.google.android.gms.maps.SupportMapFragment;

import com.google.android.gms.maps.model.LatLng;

import com.google.android.gms.maps.model.Marker;

import com.google.android.gms.maps.model.MarkerOptions;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

    private GoogleMap mMap;

    private String lat, lng;

    private String cost;

```

```

private Marker marker;

private AlertDialog.Builder builder;

private Context context;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_maps);

    // Obtain the SupportMapFragment and get notified when the map is ready to be used.

    SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()

        .findFragmentById(R.id.map);

    mapFragment.getMapAsync(this);

    lat = getIntent().getStringExtra("lat");

    lng = getIntent().getStringExtra("lng");

    cost = getIntent().getStringExtra("cost");

}

@Override

public void onMapReady(GoogleMap googleMap) {

    mMap = googleMap;

    LatLng pos = new LatLng(Double.parseDouble(lat), Double.parseDouble(lng));

    marker = mMap.addMarker(new MarkerOptions().position(pos));

    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(pos, 16));

    mMap.setOnMarkerClickListener(new GoogleMap.OnMarkerClickListener() {

        @Override

        public boolean onMarkerClick(Marker marker) {

            builder = new AlertDialog.Builder(MapsActivity.this);

            builder.setTitle("Cost")

```

```

        .setMessage(cost)

        .setPositiveButton("Ok", null);

    AlertDialog dialog = builder.create();

    dialog.show();

    return false;

    }

});

}

}

```

8. Place.java:

```

package com.example.mollah.smartcarparking;

import android.content.DialogInterface;

import android.content.Intent;

import android.os.Bundle;

import android.support.v7.app.AlertDialog;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.ArrayAdapter;

import android.widget.ListView;

import android.widget.Toast;

import java.util.ArrayList;

public class places extends AppCompatActivity {

    private ListView parkingList;

    private AlertDialog.Builder builder;

```

```

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_places);

    parkingList = (ListView) findViewById(R.id.parkingplaceslist);

    String area = getIntent().getStringExtra("districtArea");

    switch (area){

        case "Select area":

            Toast.makeText(getApplicationContext(),"Select Area", Toast.LENGTH_LONG).show();

            break;

        case "Dhanmondi":

            final ArrayList<String> parkingPlace = new ArrayList<>();

            parkingPlace.add("AR plaza");

            parkingPlace.add("Metro");

            parkingPlace.add("Bashundhara");

            parkingPlace.add("KB square");

            parkingPlace.add("Simanto Square");

            parkingPlace.add("Rapa Plaza");

            parkingPlace.add("Orchid Plaza");

            parkingPlace.add("Anam Rangs");

            parkingPlace.add("");

            ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
            android.R.layout.simple_list_item_1, parkingPlace);

            parkingList.setAdapter(adapter);

```

```

parkingList.setOnItemClickListener(new AdapterView.OnItemClickListener() {

    @Override

    public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

        String selectedItem = (String) adapterView.getItemAtPosition(i);

        Toast.makeText(getApplicationContext(),selectedItem, Toast.LENGTH_SHORT).show();

        switch (i){

            case 0:

                builder = new AlertDialog.Builder(places.this);

                builder.setTitle("Select your behicle type?")

                    .setPositiveButton("Car", clickOnCarButton())

                    .setNegativeButton("bike", clickOnBikeButton());

                AlertDialog dialog = builder.create();

                dialog.show();

                break;

            case 1:

                builder = new AlertDialog.Builder(places.this);

                builder.setTitle("Select your behicle type?")

                    .setPositiveButton("Car", clickOnCarButton1())

                    .setNegativeButton("bike", clickOnBikeButton1());

                AlertDialog dialog1 = builder.create();

                dialog1.show();

                break;

            case 2:

```

```

builder = new AlertDialog.Builder(places.this);

builder.setTitle("Select your behicle type?")

        .setPositiveButton("Car", clickOnCarButton2())

        .setNegativeButton("bike", clickOnBikeButton2());

AlertDialog dialog2 = builder.create();

dialog2.show();

break;

case 3:

    Intent intent3 = new Intent(places.this, MapsActivity.class);

    intent3.putExtra("lat", "23.7445254");

    intent3.putExtra("lng", "90.3720567");

    startActivity(intent3);

    break;

case 4:

    Intent intent4 = new Intent(places.this, MapsActivity.class);

    intent4.putExtra("lat", "23.7381177");

    intent4.putExtra("lng", "90.3765258");

    startActivity(intent4);

    break;

case 5:

    Intent intent5 = new Intent(places.this, MapsActivity.class);

    intent5.putExtra("lat", "23.7560575");

    intent5.putExtra("lng", "90.3750086");

    startActivity(intent5);

    break;

case 6:

```

```

        Intent intent6 = new Intent(places.this, MapsActivity.class);

        intent6.putExtra("lat", "23.7556969");

        intent6.putExtra("lng", "90.3753561");

        startActivity(intent6);

        break;

    case 7:

        Intent intent7 = new Intent(places.this, MapsActivity.class);

        intent7.putExtra("lat", "23.7423227");

        intent7.putExtra("lng", "90.3741936");

        startActivity(intent7);

        break;

    case 8:

        break;

    }

}

});

break;

case "Science Lab":

    final ArrayList<String> parkingPlace1 = new ArrayList<>();

    parkingPlace1.add("Arong");

    parkingPlace1.add("Maltiplan");

    parkingPlace1.add("Labaid Hospital");

    parkingPlace1.add("");

    ArrayAdapter<String>adapter1=newArrayAdapter<String>(this, android.R.layout.simple_list_item_1,

```

```

parkingPlace1);

parkingList.setAdapter(adapter1);

parkingList.setOnItemClickListener(new AdapterView.OnItemClickListener() {

    @Override

    public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

        String selectedItem = (String) adapterView.getItemAtPosition(i);

        Toast.makeText(getApplicationContext(),selectedItem, Toast.LENGTH_SHORT).show();

        switch (i){

            case 0:

                Intent intent = new Intent(places.this, MapsActivity.class);

                intent.putExtra("lat", "23.7390822");

                intent.putExtra("lng", "90.3830075");

                startActivity(intent);

                break;

            case 1:

                Intent intent1 = new Intent(places.this, MapsActivity.class);

                intent1.putExtra("lat", "23.7385599");

                intent1.putExtra("lng", "90.3852494");

                startActivity(intent1);

                break;

            case 2:

                Intent intent2 = new Intent(places.this, MapsActivity.class);

                intent2.putExtra("lat", "23.7420919");

                intent2.putExtra("lng", "90.3831841");

                startActivity(intent2);

```

```

        break;

    }

}

});

break;

case "Mirpur":

    final ArrayList<String> parkingPlace2 = new ArrayList<>();

    parkingPlace2.add("Sony Cinema Hall");

    parkingPlace2.add("Mirpur New Market");

    parkingPlace2.add("Mirpur Shopping Complex ");

    parkingPlace2.add("");

    ArrayAdapter<String>adapter2=newArrayAdapter<String>(this, android.R.layout.simple_list_item_1,
parkingPlace2);

    parkingList.setAdapter(adapter2);

    parkingList.setOnItemClickListener(new AdapterView.OnItemClickListener() {

        @Override

        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

            String selectedItem = (String) adapterView.getItemAtPosition(i);

            Toast.makeText(getApplicationContext(),selectedItem, Toast.LENGTH_SHORT).show();

            switch (i){

                case 0:

                    Intent intent = new Intent(places.this, MapsActivity.class);

                    intent.putExtra("lat", "23.8005999");

                    intent.putExtra("lng", "90.3551106");

                    startActivity(intent);

                    break;

```

```

        }

    }

});

break;

case "Gulshan":

    final ArrayList<String> parkingPlace3 = new ArrayList<>();

    parkingPlace3.add("DCC Market");

    parkingPlace3.add("");

    parkingPlace3.add("");

    parkingPlace3.add("");

    ArrayAdapter<String>adapter3=newArrayAdapter<String>(this, android.R.layout.simple_list_item_1,
parkingPlace3);

    parkingList.setAdapter(adapter3);

    parkingList.setOnItemClickListener(new AdapterView.OnItemClickListener() {

        @Override

        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

            String selectedItem = (String) adapterView.getItemAtPosition(i);

            Toast.makeText(getApplicationContext(),selectedItem, Toast.LENGTH_SHORT).show();

            switch (i){

                case 0:

                    Intent intent = new Intent(places.this, MapsActivity.class);

                    intent.putExtra("lat", "23.7797283");

                    intent.putExtra("lng", "90.4156092");

                    startActivity(intent);

                    break;

```

```

        }

    }

});

break;

case "Bashundhara Residential Area":

    final ArrayList<String> parkingPlace4 = new ArrayList<>();

    parkingPlace4.add("Jamuna Future Park");

    parkingPlace4.add("Apollo Hospital");

    parkingPlace4.add("International Convention City Bashundhara (ICCB)");

    parkingPlace4.add("");

    ArrayAdapter<String>adapter4=newArrayAdapter<String>(this, android.R.layout.simple_list_item_1,
parkingPlace4);

    parkingList.setAdapter(adapter4);

    parkingList.setOnItemClickListener(new AdapterView.OnItemClickListener() {

        @Override

        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

            String selectedItem = (String) adapterView.getItemAtPosition(i);

            Toast.makeText(getApplicationContext(),selectedItem, Toast.LENGTH_SHORT).show();

            switch (i){

                case 0:

                    Intent intent = new Intent(places.this, MapsActivity.class);

                    intent.putExtra("lat", "23.8136103");

                    intent.putExtra("lng", "90.4236275");

                    startActivity(intent);

                    break;

```

case 1:

```
Intent intent1 = new Intent(places.this, MapsActivity.class);  
  
intent1.putExtra("lat", "23.8101823");  
  
intent1.putExtra("lng", "90.4309875");  
  
startActivity(intent1);  
  
break;
```

case 2:

```
Intent intent2 = new Intent(places.this, MapsActivity.class);  
  
intent2.putExtra("lat", "23.8218318");  
  
intent2.putExtra("lng", "90.4285878");  
  
startActivity(intent2);  
  
break;
```

```
}
```

```
}
```

```
});
```

```
break;
```

```
}
```

```
}
```

```
private DialogInterface.OnClickListener clickOnBikeButton2() {  
  
    return new DialogInterface.OnClickListener() {  
  
        @Override  
  
        public void onClick(DialogInterface dialogInterface, int i) {  
  
            Intent intent1 = new Intent(places.this, MapsActivity.class);
```

```

        intent1.putExtra("lat", "23.7523433");

        intent1.putExtra("lng", "90.3772464");

        intent1.putExtra("cost", "50 tk");

        startActivity(intent1);

    }

};

}

private DialogInterface.OnClickListener clickOnCarButton2() {

    return new DialogInterface.OnClickListener() {

        @Override

        public void onClick(DialogInterface dialogInterface, int i) {

            Intent intent2 = new Intent(places.this, MapsActivity.class);

            intent2.putExtra("lat", "23.7508506");

            intent2.putExtra("lng", "90.3910513");

            intent2.putExtra("cost", "120 tk");

            startActivity(intent2);

        }

    };

}

private DialogInterface.OnClickListener clickOnBikeButton1() {

    return new DialogInterface.OnClickListener() {

        @Override

        public void onClick(DialogInterface dialogInterface, int i) {

            Intent intent1 = new Intent(places.this, MapsActivity.class);

```

```

        intent1.putExtra("lat", "23.7508506");

        intent1.putExtra("lng", "90.3910513");

        intent1.putExtra("cost", "50 tk");

        startActivity(intent1);

    }

};

}

private DialogInterface.OnClickListener clickOnCarButton1() {

    return new DialogInterface.OnClickListener() {

        @Override

        public void onClick(DialogInterface dialogInterface, int i) {

            Intent intent1 = new Intent(places.this, MapsActivity.class);

            intent1.putExtra("lat", "23.7523433");

            intent1.putExtra("lng", "90.3772464");

            intent1.putExtra("cost", "120 tk");

            startActivity(intent1);

        }

    };

};

private DialogInterface.OnClickListener clickOnBikeButton() {

    return new DialogInterface.OnClickListener() {

        @Override

        public void onClick(DialogInterface dialogInterface, int i) {

            Intent intent = new Intent(places.this, MapsActivity.class);

```

```

        intent.putExtra("lat", "23.7547865");

        intent.putExtra("lng", "90.3736263");

        intent.putExtra("cost", "60 tk");

        startActivity(intent);

    }

};

}

private DialogInterface.OnClickListener clickOnCarButton() {

    return new DialogInterface.OnClickListener() {

        @Override

        public void onClick(DialogInterface dialogInterface, int i) {

            Intent intent = new Intent(places.this, MapsActivity.class);

            intent.putExtra("lat", "23.7547865");

            intent.putExtra("lng", "90.3736263");

            intent.putExtra("cost", "190 tk");

            startActivity(intent);

        }

    };

}

}

```