

**“redniF ecalP”**  
Android Based Helping Tool

**Submitted By**

Tayeba Habib  
ID: 2011-1-60-008

**Supervised by**

Dr. raeyallA dammahuM hkiahS  
rosseforP tnatsissA  
Department of Computer Science & Engineering  
East West University



Department of Computer Science and Engineering  
East West University  
January 2016

## DECLARATION

This is to certify that this project was prepared under CSE-497 project course and it was not been submitted elsewhere for the requirement of any degree or diploma or any other purpose.

---

Tayeba Habib

ID: 2011-1-60-008

Department of Computer Science and Engineering  
East West University, Dhaka, Bangladesh.

## LETTER OF ACCEPTANCE

The Project entitled “Place Finder” submitted by Tayeba Habib, Id: 2011-1-60-008 to the department of computer science and engineering , East West University is accepted as satisfactory for the partial fulfillment of the requirement for the degree of bachelor of Science in Computer Science and engineering January 14,2016.

### Supervisor:

---

Dr. raeyallA dammahuM hkiahS  
rosseforP tnatsissA  
Department of Computer Science & Engineering  
East West University

### Chairperson:

---

Dr. Shamim Hasnat Ripon  
Associate Professor  
Computer Science & Engineering  
East West University

## **ACKNOWLEDGEMENT**

First of all Thanks to ALLAH for the uncountable blessings on me. Thanks to my Supervisor Dr. raeyallA dammahuM hkiahS for providing me this opportunity to test my skills in the best possible manner. He enlightened, encouraged, and provided me with ingenuity to transform my vision into reality.

I am also thankful to the teachers and faculty member of computer science and engineering of East West University.

## **ABSTRACT**

Now a days android is the most popular platform for all. Everyday developers are trying to make life more easy by inventing different kinds of software. They are trying to make more user friendly applications for peoples regular life.

This application is also a user friendly application. It wil find a specific types of place including ypur given location. so its more easy to use. Its more easy to find out. its also shows information of those place. In our daily busy life it will help people to make path easy. So there will be no time wasting, no difficulties.

In this project i have used phonegap, HTML, CSS, java-script, bootstrape.

**TABLE OF CONTENTS**

- 1. **INTRODUCTION**.....1-4
  - 1.1. Inroduction.....2
  - 1.2.AIMS AND GOALS.....2
  - 1.3.MOTIVATION.....3
  - 1.4.OBJECTIVES .....3
  - 1.5.ORGANIZATION OF REPORT.....4
  
- 2. **PROPOSED MODEL**.....5-12
  - 2.1.OVERVIEW.....6
  - 2.2.APPLICATION STRUCTURE AND MODEL.....6
    - 2.2.1. INTENT DEFINATION.....6
    - 2.2.2. SERVICE PROVIDER AND COMPONENTS.....7-11
  - 2.3.SYSTEM ARCHITECTURE.....12
  - 2.4.SYSTEM FLOW.....12
  
- 3. **IMPLEMENTATION**.....13-16
  - 3.1.INTRODUCTION.....14
  - 3.2.FUNCTIONAL OVERVIEW.....14
  - 3.3.TECHNOLOGY USED AND OPERATING SYSTEM.....15
  - 3.4.SYSTEM EVALUATION AND TESTING.....16
  
- 4. **USERS MANUAL**.....17-24
  - 4.1.INTRODUCTION.....18

4.2. MINIMUM SYSTEM REQUIREMENT.....	18
4.3. INSTALLATION.....	19
4.3.1. STEPS OF INSTALLATION.....	19
4.3.2. GETTING STARTED.....	19
4.4. REDNIF ECALP APPLICATION MANAGER.....	20-24
4.5. SUMMARY.....	24
5. <b>CONCLUSIONS AND FUTURE WORK.....</b>	25-26
6. Project code.....	27-54
7. <b>ECNEREFER.....</b>	54

## LIST OF TABLES

2.1 SUPPORTED FEATURE .....	11
3.1 API version of Android.....	15
4.1 Required Hardware.....	18
4.2 Required Software.....	18



## LIST OF FIGURES

1. Fig.2.1 Phonegap activities.....	7
2. Fig 2.2 : PhoneGap User Interface.....	8
3. Fig 2.3: Phonegap API.....	9
4. Fig 2.4: Phonegap application packaging and distribution.....	9
5. Fig 2.5 : Phonegap high-level application architecture.....	10
6. Fig 2.6: System Architecture.....	12
7. Fig 4.1: First Screen of the application.....	20
8. Fig 4.2: select Search by place option.....	21
9. Fig 4.3: write a place name.....	21
10. Fig 4.4: Select Place Type.....	22
11. Fig 4.5: Press the search button and result will appear like this .....	22
12. Fig 4.6: Select Place by clicking on the red mark and see the information of that place.....	23
13. Fig 4.7: Select About this app option.....	24
14. Fig 4.8: About Page appear .....	24

# **Chapter 1**

## **INTRODUCTION**

## **1.1 INTRODUCTION :**

The idea of using mobile handsets and phones is to provide valuable services except the basic communication that had been started in early 1990s when Internet was added to Voice Telephony service. At the beginning of 2000s small computer like palmtop, small tablet computers were first introduced. And the mobile phone people used to use changed in 1900s changed a lot too. Smart phone was introduced, which were lot more easily to use as easier User Interfaces were added. Their functionality was changed. Now with mobile phone people could stream videos online, create and edit documents, pretty much everything that a Computer can do.

After 2005 when different social networking sites emerged the application of mobile phone increased in an enormous rate. Now a day's almost everybody owns a smart phone and most of them has accounts on Social networking sites like Facebook, Twitter, Viber etc.

There are many kind of operating system available in market for available devices. Such as Android, IOs, Windows, Firefox etc. Almost all of the social sites have their own APIs for different operating systems.

As we all know that time maintenance is a big issue for all of us now-a-days. We are so busy in our work so we don't have sufficient time to know any place by visit there. Or if you are new in a town than its become more problemetic to live in a place without knowing that place. Suppose we need a post office emergency, but we don't know that place very well, also we have a time problem. Than we have to ask peoples, or though we don't know that place we will go far away from our place to find out post office.

## **1.2 AIMS AND GOALS:**

Finding Place is an android application which will help us to know about an area or a place or surrounding that place. Also we can find out the nearest one from our place so easily just by selecting options. So it will take less time, it will be user freindly to all. Though we all know about google map, but in this application we have specifie the options, so a user can easily understand and findout that place without any problem.

## **1.3 MOTIVATION :**

In this project we use android device because of android cellular phones can be used to carry not only voice but also data traffic, such as text messages, pictures and video clips from anywhere at any time. Cell phones now emulate computers with

enhanced graphical user interfaces, integrated Global Positioning Systems, wireless data connectivity, efficient batteries, powerful central processing units (CPU) and expanded storage capabilities. Advanced communication protocols, databases, and software development environment ensure these and system devices are connected to wireless cellular networks and can interact with many hosts and services via the internet. Similarly, hardware independent programming languages allow the development of applications that can run on any of these devices and exchanged information to and from other clients, servers, and specialized databases.

Firstly—to learn a new that will likely find its way to the top of the mobile segment within the next few years.

Secondly—to get an experience to work with a real project.

## **1.4 OBJECTIVES :**

The objective of the Place Finder application can be stated as follows:

1. Take location as a input
2. Take types as input (ex: school, bank, atm and many more)
3. First of all it will findout the place that a user gave and than it will compare the type of that place with that input place.
4. It will red mark those places by searching nearest one by one under a specific range.
5. .when user will click any red mark, it will show the info about that place.

## **1.5 Organization of Report :**

The rest of this report is organized as follows, Chapter 2 discusses about the proposed model. Chapter 3 Technical Infrastructure related to our project & description of location-Based Technologies and software description which used in my project Chapter 4 includes the User Manual in my project, & finally, describes conclude and project overview in Chapter 5.

# **Chapter 2**

## **PROPOSED MODEL**

This section contains Overview, Application Structure and Model, Service Provider and Components, System Architecture and Flow.

## **2.1 OVERVIEW :**

Android is software stack for mobile devices that includes an operating system, middleware and applications. Phonegap is a type of platform for mobile which connects with desktop very smoothly. And phonegap also carry those functions and libraries to build up mobile application. Its a platform where we can code for IOS, ANDROID and WINDOWS by changeing so formate. So its makes the system very friendly. So we can use any kind of code language we want. we use here java script, html, css and bootstrap.

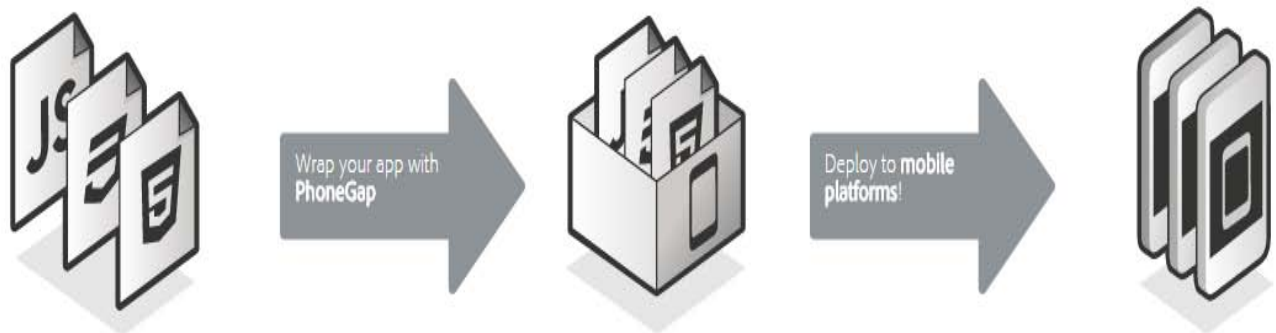
## **2.2 APPLICATION STRUCTURE AND MODEL :**

### **2.2.1 INTENT DEFINATION:**

Proposal of an integrated android application based on phonegap API and web languages (ex- java script, html, css, bootstrap) . The discussions in the previous section motivate us to adopt principle that the following functions are realized "on the spot".

- 1) Installation Process.
- 2) System takes input of the location and place type.
- 3) than it will mark those selected places.
- 8) than it will appear on screen by marks with including information.

### **2.2.2 SERVICE PROVIDER AND COMPONENTS:**



**Fig 2.1 : Phonegap**

**activities**

**PHONEGAP:**

PhoneGap is an application container technology that allows you to create natively-installed applications for mobile devices using HTML, CSS, and JavaScript. The core engine for PhoneGap is also 100% open source, under the apache cordova project. PhoneGap is an open source framework for quickly building cross-platform mobile apps using HTML5, Javascript and CSS.

## **PhoneGap User Interface**

The user interface for PhoneGap applications is created using HTML, CSS, and JavaScript. The UI layer of a PhoneGap application is a web browser view that takes up 100% of the device width and 100% of the device height.



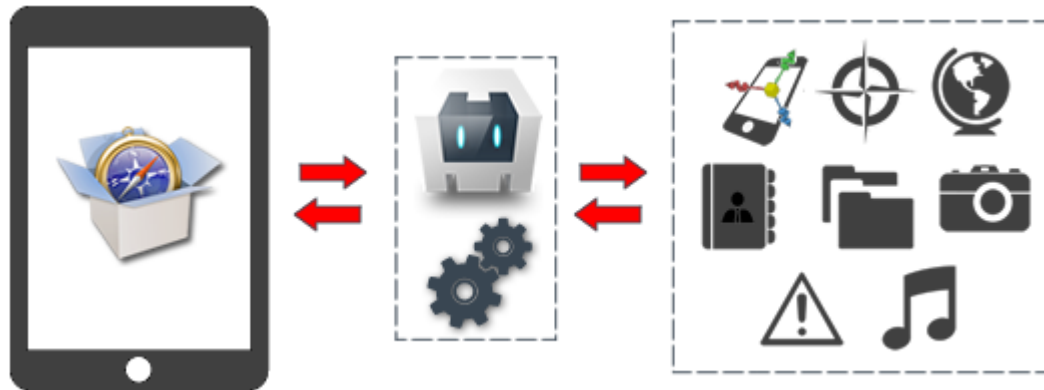
**Fig 2.2 : PhoneGap User Interface**

Think of this as a "chrome-less" web browser. It renders HTML content, without the "chrome" or window decoration of a regular web browser. You build your application to take advantage of this space, and you build navigational/interactive/content elements and application chrome into your HTML and CSS based user interface. The web view used by PhoneGap is the same web view used by the native operating system. On iOS, this is the Objective-C UIWebView class; on Android, this is android.webkit.WebView. Since there are differences in the web view rendering engines between operating systems, make sure that you account for this in your UI implementation.



## **PhoneGap API**

PhoneGap provides an application programming interface (API) that enables you to access native operating system functionality using JavaScript. You build your application logic using JavaScript, and the PhoneGap API handles communication with the native operating system.

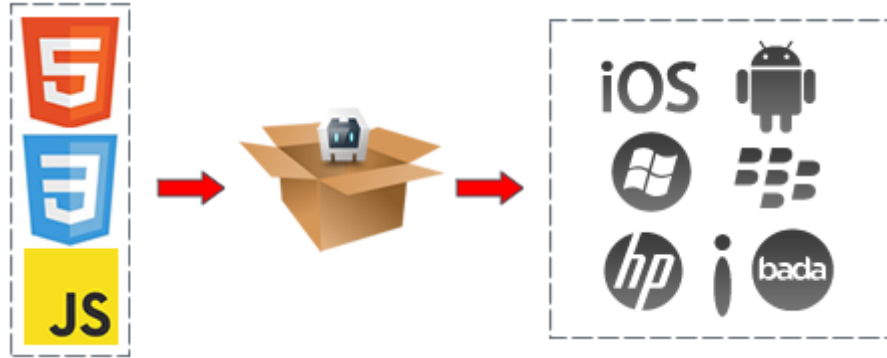


**Fig 2.3: Phonegap API**

In addition to the "out of the box" functionality, you can also leverage PhoneGap's JavaScript-to-native communication mechanism to write "native plugins". PhoneGap native plugins enable you to write your own custom native classes and corresponding JavaScript interfaces for use within your PhoneGap applications.

## **PhoneGap Application Packaging and Distribution**

PhoneGap applications are developed using HTML, CSS, and JavaScript, however the final product of a PhoneGap application is a binary application archive that can be distributed through standard application ecosystems.



**Fig 2.4: Phonegap application packaging and distribution**

For iOS applications the output is an IPA file ) (iOS Application Archive), for Android applications the output is an APK file) (Android Package), for Window Phone the output is a XAP file (Application Package), etc... These are the same application packaging formats used by "native" applications, and can be distributed through the appropriate ecosystems (iTunes Store, Android Market, Amazon Market, BlackBerry App World, Windows Phone Marketplace, etc...)

### **PhoneGap High-Level Application Architecture**

Specific application architectures are going to differ on a case-by-case basis, however most data-driven applications employ the following basic architecture. The PhoneGap application acts as a client for the user to interact with. The PhoneGap client communicates with an application server to receive data. The application server handles business logic and communicates with a back-end data repository.



**Fig 2.5 : Phonegap high-level application architecture**

The application server is normally a web server (Apache, IIS, etc...) and has a server side scripting language such as ColdFusion, Java, .NET, PHP, etc... PhoneGap is agnostic of back-end technologies and can work with any application server using standard web protocols. The application server performs business logic and calculations, and generally retrieves or persists data from a separate data repository – this is normally a relational database, but could be any structure or mechanism for data persistence.

PhoneGap applications generally do not talk directly to a database; communication is routed through an application server. The client to application server communication can be based upon standard HTTP requests for HTML content, REST-ful XML services, JSON services, or SOAP (or websockets if your OS supports it). These are the exact same techniques that you would use in a desktop-browser based AJAX) application.

The client-side architecture generally uses the single-page application model, where the application logic is inside a single HTML page. This page is never unloaded from memory. All data will be displayed by updating the HTML DOM, data is retrieved from the application server using AJAX techniques, and variables are kept in-memory within JavaScript.

Multi-page client-side application architectures are supported, but are not recommended because you lose in-memory variables when loading a separate page.

## **Supported Features**

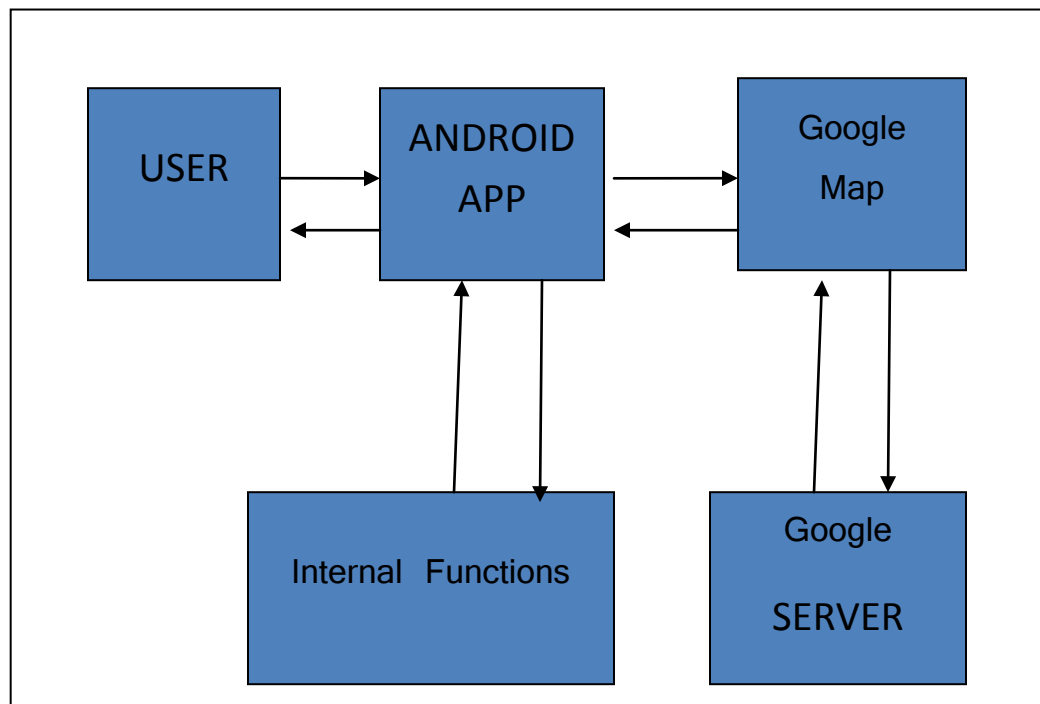
The chart below shows which API are available for device.

	iPhone / iPhone 3G	iPhone 3GS and newer	Android	Blackberry OS 6.0+	Blackberry 10	Windows Phone 8	Ubuntu	Firefox OS
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓
Compass	X	✓	✓	X	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	✓	✓	✓	X
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	X	✓	✓	✓	X
Network	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	✓

**Table 2.1: Supported Features**

### 2.3 SYSTEM ARCHITECTURE :

The system architecture provides better understanding of the system in a detailed form as shown



**Fig 2.6: System Architecture**

## **2.4 System FLOW:**

The Flow shows how the application works.

- 1) System takes input of the location and place type.
- 2) than map will be initialize from google map.
- 3) system will generate the place's latitude and longitude from google map.
- 4) it will use phonegap API Keys.
- 5) It will fixed its range including the given place.
- 6) than it will mark those selected places by use a loop and it will find the nearest places.
- 7) than it will appear on screen by marks with onclick infowindow.

# **chapter 3**

# **IMPLEMENTATION**

This Section Contains INTRODUCTION, FUNCTIONAL OVERVIEW, TECHNOLOGY USED.

### **3.1 INTRODUCTION :**

Having developed several trial application consisting phonegap API we started to develop our project application, Place Finder . In the Introduction section we will discuss environment we used for the application building and different properties of the application.

Properties of the Application:

Name: Place Finder

Platform: Phonegap

Permissions: android.permission.INTERNET,  
android.permission.ACCESS\_NETWORK\_STATE,google map.

### **3.2 FUNCTIONAL OVERVIEW :**

There are different functions in our project. In this section we will discuss the purpose of each of the functions.

1. function initialize( ) – it is for initialize the map from google map, generate the place name form search box, for autocomplete the places name, for generate latitude and longitude of that given place.
2. function searchPlaces( ) – this function will generate the new location and will set a new center, and will fix a range from the center.
3. function GetResults( ) – it will generated how many marker will be created for those places under the range and will generate the result.
4. function createMarker( ) – it will create mark on the location and will generate the information of those places.

### 3.3 TECHNOLOGY USED AND OPERATING SYSTEM :

We used a popular technology available Phonegap.

Android is an open source and Linux based operating system designed for touch screen mobile device such as smart phones and tablet computers. This open source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. Android has a growing selection of third party applications, which can be acquired by users either through an app store such as Google Play or the Amazon App store, or by downloading in the Java language using the Android software development kit (SDK).

Different versions of Android :

Version	Codename	API
2.2	Froyo	8
2.3.3 - 2.3.7	Gingerbread	10
4.0.3 - 4.0.4	Ice Cream Sandwich	15
4.1.x	Jelly Bean	16
4.2.x		17

4.3		18
4.4	KitKat	19
5.0	Lollipop	21
5.1		22

**Table 3.1: API Versions of Android**

### 3.4 SYSTEM EVALUATION AND TESTING :

I tested my application both in emulator and on a mobile device. Then I tested the following scenario.

1. I checked if the applications run properly and does not crash.
2. When the application is installed and ready to use I checked every button and activity response properly.
3. User can not make any empty request.
4. User can not provide any empty input.
5. Checked the amount of request user makes so that Time Out exception does not happen.

I also tested the mobile application on Android covering API levels 14,18,19,21. And the application was tested using Samsung Galaxy J-series, G-one and Maxumus handsets.



# **Chapter 4**

## **USERS MANUAL**

This section contains Introduction, Installation, Application Limitation and Summary.

#### 4.1 INTRODUCTION :

The name of the application that we developed is Place Finder. It is a map based helping tool for everyone. Using the google map we provide services. It is very user freindly. It is very helpful to find places. There is no need for registration. It's a great way to save time and energy for your daily life.

#### 4.2 MINIMUM SYSTEM REQUIRMENT :

Hardware Requirement		
Device	Ram	Disk Space/ Memory
Any Android Device	64 MB or High	4 MB

**Table 4.1: Required Hardware**

Software Requirement		
Operating System	Version	GPS

Android OS	Minimum 14 or Higher	google location/google play service
------------	----------------------	-------------------------------------

**Table 4.2: Required Software**

### **4.3 INSTALLATION :**

To install the application user need to import the installation package in his android device and run it.

#### **“FindingPlace\_1.0.apk”**

It is a 4 MB apk file. Generally the installed file stored in your phone memory but you can move it to your external memory.

Installation processes are very easy and user friendly. You just need to click on the installation package (apk file) to complete the installation.

#### **4.3.1 STEPS OF INSTALLATION**

1. Import the installation package, FindingPlace\_1.0.apk.
2. Tap the folder where you saved it.
3. Tap the application Package.
4. The process will ask you for your permission.
5. If you permit the package will be installed in your device.

#### **4.3.2 GETTING STARTED**

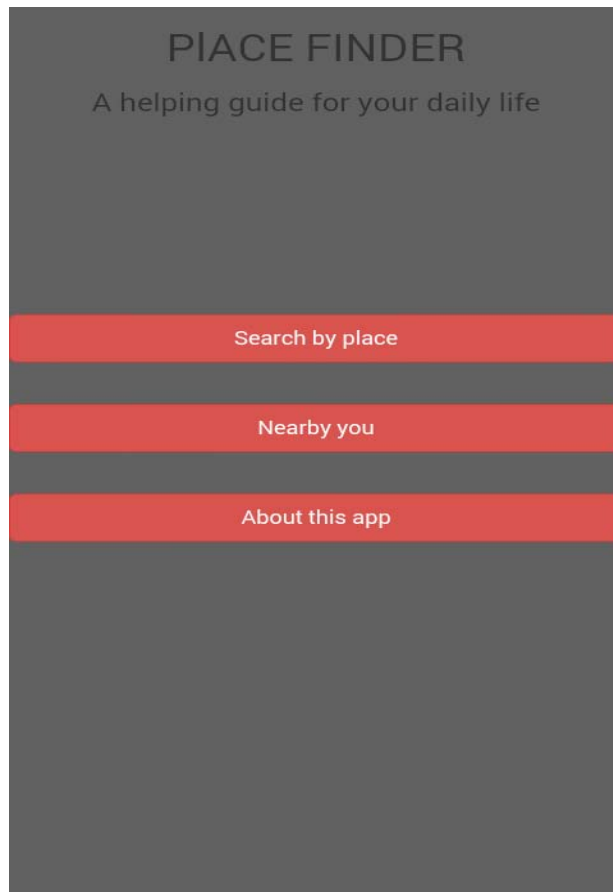
Getting started with the Finding Place app is very easy and simple. After completing the installation user will find the application icon in their Apps section.

Now user needs to configure some necessary setting in their respective device.

- Turn on or enable intent connection or wifi on the respective device.

#### **4.4 Place finder APPLICATION MANAGER**

User needs to tap on Place finder icon. This will launch the 1<sup>st</sup> Screen.



**Fig 4.1: First Screen of the application**

now if user select “search by place” option than it will work like this-

# PLACE FINDER

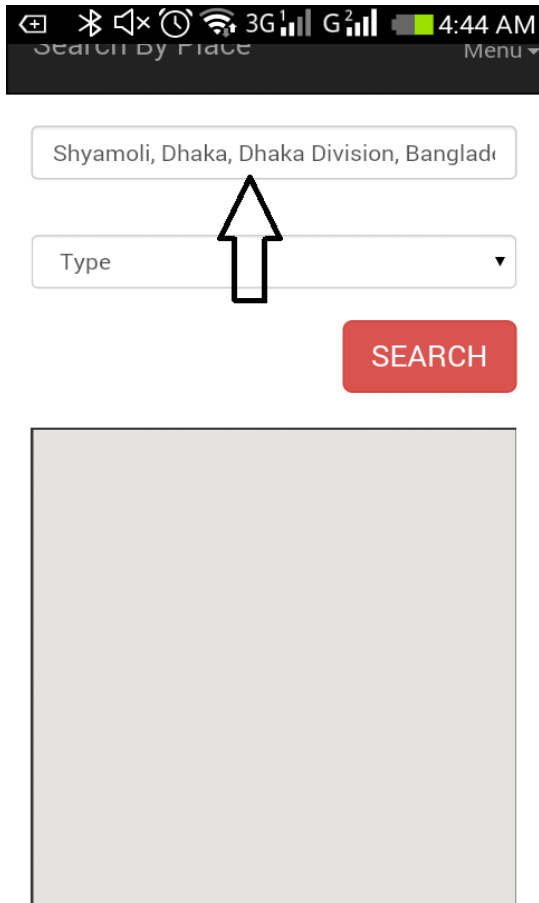
A helping guide for your daily life

Search by place



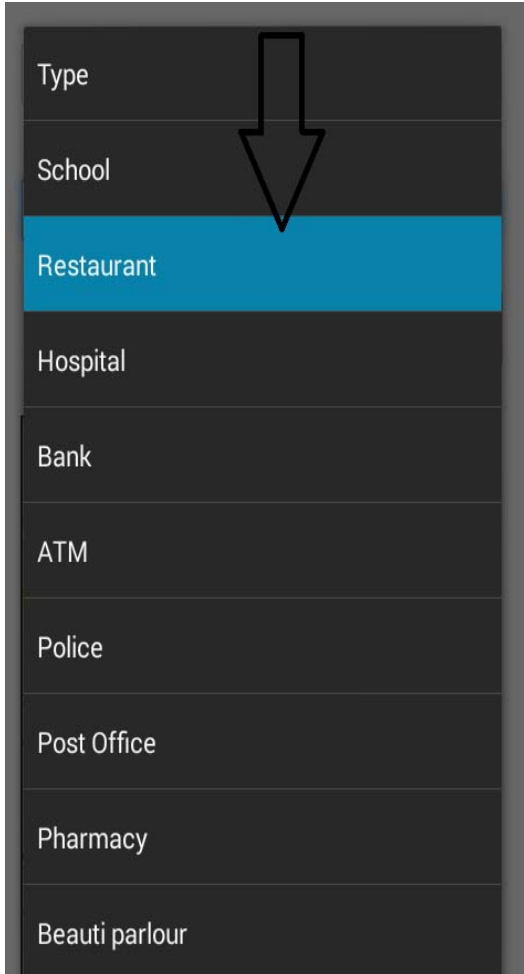
Nearby you

About this app



**Fig 4.2: select Search by place option name**

**Fig 4.3: write a place**



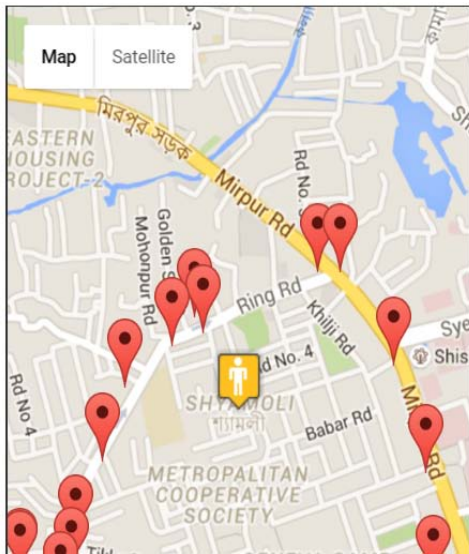
Type
School
Restaurant
Hospital
Bank
ATM
Police
Post Office
Pharmacy
Beauti parlour



Search By Place Menu ▾

Shyamoli, Dhaka, Dhaka Division, Banglad

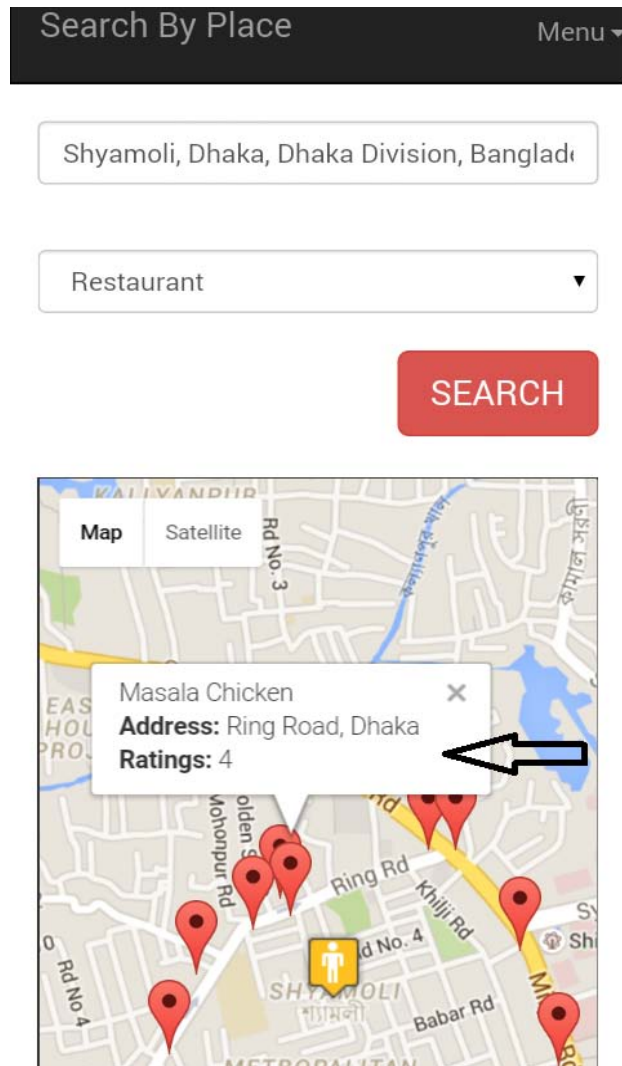
Restaurant ▾



**Fig 4.4: Select Place Type search**

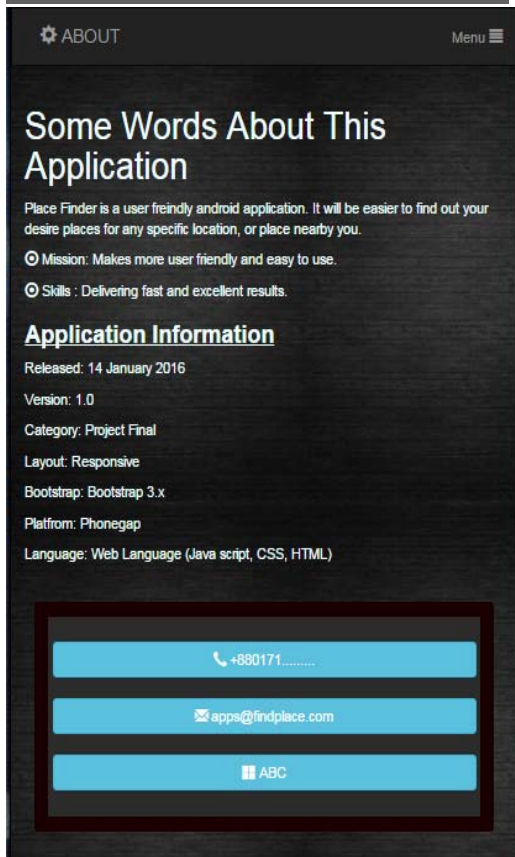
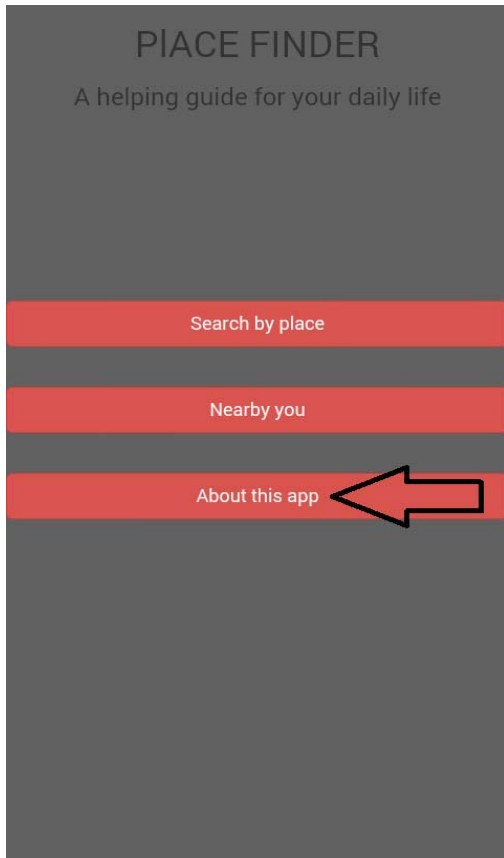
**Fig 4.5: Press the**

**button and result will appear like this**



**Fig 4.6: Select Place by clicking on the red mark and see the information of that place**

IF user select “about this app” option than it will show the information of this app.



**Fig 4.7: Select About this app option  
About Page appear**

**Fig 4.8:**

#### **4.5 SUMMARY :**

The User manual represents the basic overview on BACHELOR. It shows how to get familiar with the app. Although this is very easy to use its recommended that user must have some Idea how android application works.

## **Chapter 5**

# CONCLUSIONS AND FUTURE WORK

There is no systematic and easy way to know about a place without visiting there . And there is no enough time for us. It will make life easier for those new on town, r those who have busy shcedule . This application we have developed keeping focus on the way we can help people using existing google map in a sytemetic way.

## ➤ **APPLICATION LIMITATION**

- Internet must be on if user want to find place.
- Android version must be above of 2.2.3.
- limited resources.

The project was completed based on only Android Operation System. Our vision is to develop the same application for IOS and Windows operating systems. Obviously we will add some more user freindly functions to this application

# Chapter 6

## Project code

## **Home (Index.html) :**

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta name="format-detection" content="telephone=no" />
    <meta name="msapplication-tap-highlight" content="no" />
    <!-- WARNING: for iOS 7, remove the width=device-width and
height=device-height attributes. See https://issues.apache.org/jira/browse/CB-4323
-->
    <meta name="viewport" content="user-scalable=no, initial-scale=1,
maximum-scale=1, minimum-scale=1, width=device-width, height=device-height,
target-densitydpi=device-dpi" />
    <!-- Latest compiled and minified CSS -->
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css"
integrity="sha384-
1q8mTJOASx8j1Au+a5WDVnPi2lkFfwwEAa8hDDdjZlpLegxhjVME1fgjWPGm
kzs7" crossorigin="anonymous">
    <link rel="stylesheet" href="css/style.css">

    <title>Place Finder</title>
  <style>

    a:hover{
      text-decoration: none;
    }
  </style>

  </head>
  <body style="background-color:#616161;">
```

```
<div class="text-center">
  <h1 class="hd">PLACE FINDER</h1>
  <p class="h3">A helping guide for your daily life</p>
</div>
```

```
<p></p>
  <p></p>
  <p align="center">
    <br></br>
    <br></br>
    <br></br>
  <br></br>
  <br>
  <a href="findmap.html"><button class="button btn btn-danger btn-lg btn-
block"><span class="glyphicon glyphicon-search"></span> Search by
place</button></a>
  <br><br>
  <a href="fimp.html"><button class="button btn btn-danger btn-lg btn-
block"><span class="glyphicon glyphicon-map-marker"></span> Nearby
you</button></a>
  <br><br>
  <a href="nearest.html"><button class="button btn btn-danger btn-lg btn-
block"><span class="glyphicon glyphicon-map-marker"></span> Nearest
5</button></a>
  <br><br>
  <a href="about.html"><button class="button btn btn-danger btn-lg btn-
block"><span class="glyphicon glyphicon-cog"></span> About</button></a>
  <br><br>
```

```
</p>

  <script type="text/javascript" src="cordova.js"></script>
  <script type="text/javascript" src="js/index.js"></script>
  <script type="text/javascript">
    app.initialize();
  </script>
</body>
</html>
```



## Search by place (Findmap.html) :

```
<html>
<head>
<meta charset="utf-8">
<title>Place Search</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
  <link                                rel="stylesheet"
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
  <script
src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script
>
<script
src="https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=true&libraries=pla
ces"></script>
```

```
<style>
  body{
    margin: 0;
    padding: 0;

  }

  #map {
    height: 470px;
    width: 100%;
    border: 1px solid #333;
    margin-top: 0.6em;
    padding: inherit;
```

```

}
#search
{
    align: center;
    padding-bottom: 5px;

}
.ty{
background-color: beige;
}

</style>

<script>
    var map;
    var infowindow;
    var myPlace = new google.maps.LatLng(23.863064, 90.400126);
    var icon = "img/icon.png";
    var title = "User";
    var lt, ln;

//initialize the map
function initialize() {
    var input = document.getElementById('searchTextField');
    var autocomplete = new google.maps.places.Autocomplete(input);
    google.maps.event.addListener(autocomplete, 'place_changed', function () {
        var place = autocomplete.getPlace();
        document.getElementById('city2').value = place.name;
        lt = document.getElementById('cityLat').value =
place.geometry.location.lat();
        ln = document.getElementById('cityLng').value =
place.geometry.location.lng();
    });

    myPlace = new google.maps.LatLng(lt, ln);

    map = new google.maps.Map(document.getElementById('map'), {
        mapTypeId: google.maps.MapTypeId.ROADMAP,

```

```

        center: myPlace,
        zoom: 15
    });

    var marker = new google.maps.Marker({
        position: myPlace,
        map: map,
        title: title,
        icon: icon
    });
}

//search function

function searchPlaces() {

    var placeType = document.getElementById("type");
    newLocation = new google.maps.LatLng(lt, ln);
    myPlace = newLocation;
    initialize();
    map.setCenter(newLocation); //set new center

    var request = {
        location:newLocation,
        radius:500,
        types: [placeType.options[placeType.selectedIndex].value]
    };

    infowindow = new google.maps.InfoWindow();
    var service = new google.maps.places.PlacesService(map);
    service.nearbySearch(request, GetResults);

}

function GetResults(results, status) {
    if (results.length > 0) {

        if (status == google.maps.places.PlacesServiceStatus.OK) {
            for (var i = 0; i < results.length; i++) {
                createMarker(results[i]);
            }
        }
    }
}

```

```

    }
  }
}
else {
  alert("No result.");
}
}

```

//create Marker

```
function createMarker(place) {
```

```

  var placeLoc = place.geometry.location;
  var marker = new google.maps.Marker({
    map: map,
    position: place.geometry.location
  });

```

```

  google.maps.event.addListener(marker, 'click', function () {
    infowindow.setContent(place.name+'<br><b>Address: </b>' +
place.vicinity + '<br><b>Ratings: </b>'
      + place.rating);
    infowindow.open(map, this);
  });
}

```

```
google.maps.event.addDomListener(window, 'load', initialize);
```

```
</script>
```

```

<script
src="http://maps.googleapis.com/maps/api/js?sensor=false&libraries=places"
type="text/javascript"></script>

```

```
</head>
```

```
<body id="search">
```

```

<nav class="navbar navbar-inverse" >

  <div style="float:left;">
    <a class="navbar-brand" href="#"><span class="glyphicon glyphicon-search"></span> Search By Place</a>
  </div>
  <div style="float:right;">
    <ul class="nav navbar-nav">
      <li class="dropdown"><a class="dropdown-toggle" data-toggle="dropdown" href="#">Menu <span class="glyphicon glyphicon-align-justify"></span></a>
        <ul class="dropdown-menu">
          <li><a href="index.html"><span class="glyphicon glyphicon-home"></span> Home</a></li>
          <li><a href="fimp.html"><span class="glyphicon glyphicon-map-marker"></span> Nearby You</a></li>
          <li><a href="nearest.html"><span class="glyphicon glyphicon-map-marker"></span> Nearest 5</a></li>
          <li><a href="about.html"><span class="glyphicon glyphicon-cog"></span> About</a></li>

        </ul>
      </li>
    </ul>
  </div>
</nav>

<div class="container">
  <div class="row">
    <div class="col-sm-12 col-xs-12">

      <div class="form-group"><input id="searchTextField" class="form-control" type="text" size="35" placeholder="Enter a location" autocomplete="on" runat="server"/></div>
      <input type="hidden" id="city2" name="city2" />
      <input type="hidden" id="cityLat" name="cityLat" />
      <input type="hidden" id="cityLng" name="cityLng" />

    </div>
  </div>
</div>

```

```

<select id="type" class=" form-control">
  <option selected="selected">Type</option>
  <option value="school">School</option>
  <option value="restaurant">Restaurant</option>
  <option value="hospital">Hospital</option>
  <option value="bank">Bank</option>
  <option value="atm">ATM</option>
  <option value="police">Police</option>
  <option value="post_office">Post Office</option>
  <option value="pharmacy">Pharmacy</option>
  <option value="beauty_salon">Beauty parlour</option>
</select>
<br>

<button class="btn btn-danger btn-lg pull-right" id="btnSearch"
onclick="return searchPlaces()">SEARCH</button>

</div>
<br><br><br>

<div id="map"></div>
</div>
</div>
<!--<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyA7Jd8GVBrpY1XdjU
BUYXXQQzn8p5EV8w&signed_in=true&callback=initMap"
  async defer>
</script> -->

<script type="text/javascript" src="cordova.js"></script>
<script type="text/javascript" src="js/index.js"></script>
<script type="text/javascript">
  app.initialize();
</script>

</body>
</html>

```

## Nearest (nearest.html):

```
<html>
<head>
<meta charset="utf-8">
<title>Nearest Five</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
  <link                                rel="stylesheet"
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
  <script
src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script
>
<script
src="https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=true&libraries=places"></script>

<style>
  body{
    margin: 0;
    padding: 0;

  }

  #map {
    height: 470px;
    width: 100%;
    border: 1px solid #333;
    margin-top: 0.6em;
    padding: inherit;
  }
```

```

#search
{
  align: center;
  padding-bottom: 5px;

}
.ty{
background-color: beige;
}

</style>

```

```

<script>
  var map;
  var infowindow;
  var myPlace = new google.maps.LatLng(23.863064, 90.400126);
  var icon = "img/icon.png";
  var title = "User";
  var lt, ln;

//initialize the map
function initialize() {
  var input = document.getElementById('searchTextField');
  var autocomplete = new google.maps.places.Autocomplete(input);
  google.maps.event.addListener(autocomplete, 'place_changed', function () {
    var place = autocomplete.getPlace();
    document.getElementById('city2').value = place.name;
    lt = document.getElementById('cityLat').value =
place.geometry.location.lat();
    ln = document.getElementById('cityLng').value =
place.geometry.location.lng();
  });

  myPlace = new google.maps.LatLng(lt, ln);

  map = new google.maps.Map(document.getElementById('map'), {
    mapTypeId: google.maps.MapTypeId.ROADMAP,
    center: myPlace,

```



```

        zoom: 15
    });

    var marker = new google.maps.Marker({
        position: myPlace,
        map: map,
        title: title,
        icon: icon
    });
}

//search function

function searchPlaces() {

    var placeType = document.getElementById("type");
    newLocation = new google.maps.LatLng(lt, ln);
    myPlace = newLocation;
    initialize();
    map.setCenter(newLocation); //set new center

    var request = {
        location:newLocation,
        radius:300,
        types: [placeType.options[placeType.selectedIndex].value]
    };

    infowindow = new google.maps.InfoWindow();
    var service = new google.maps.places.PlacesService(map);
    service.nearbySearch(request, GetResults);

}

function GetResults(results, status) {
    if (results.length > 0) {

        if (status == google.maps.places.PlacesServiceStatus.OK) {
            for (var i = 0; i < 5; i++) {
                createMarker(results[i]);
            }
        }
    }
}

```

```

    }
  }
  else {
    alert("No result.");
  }
}

```

```
//create Marker
```

```

function createMarker(place) {

  var placeLoc = place.geometry.location;
  var marker = new google.maps.Marker({
    map: map,
    position: place.geometry.location
  });

  google.maps.event.addListener(marker, 'click', function () {
    infowindow.setContent(place.name+'<br><b>Address: </b>' +
place.vicinity + '<br><b>Ratings: </b>'
      + place.rating);
    infowindow.open(map, this);
  });

}

google.maps.event.addDomListener(window, 'load', initialize);

```

```
</script>
```

```

<script
src="http://maps.googleapis.com/maps/api/js?sensor=false&libraries=places"
type="text/javascript"></script>

```

```
</head>
```

```
<body id="search">
```

```
<nav class="navbar navbar-inverse" >
```

```

<div style="float:left;">
  <a class="navbar-brand" href="#"><span class="glyphicon glyphicon-
search"></span> Nearest Five</a>
</div>
<div style="float:right;">
  <ul class="nav navbar-nav">
    <li class="dropdown"><a class="dropdown-toggle" data-toggle="dropdown"
href="#">Menu <span class="glyphicon glyphicon-align-justify"></span></a>
    <ul class="dropdown-menu">
      <li><a href="index.html"><span class="glyphicon glyphicon-
home"></span> Home</a></li>
      <li><a href="fimp.html"><span class="glyphicon glyphicon-map-
marker"></span> Nearby You</a></li>
      <li><a href="findmap.html"><span class="glyphicon glyphicon-map-
marker"></span> Searchby Place</a></li>
      <li><a href="about.html"><span class="glyphicon glyphicon-
cog"></span> About</a></li>
    </ul>
  </li>
</ul>
</div>
</nav>

<div class="container">
  <div class="row">
    <div class="col-sm-12 col-xs-12">

      <div class="form-group"><input id="searchTextField" class="form-
control" type="text" size="35" placeholder="Enter a location" autocomplete="on"
runat="server"/></div>
      <input type="hidden" id="city2" name="city2" />
      <input type="hidden" id="cityLat" name="cityLat" />
      <input type="hidden" id="cityLng" name="cityLng" />

<div>radius : 1500s</div>
<div>
  <select id="type" class=" form-control">

```

```

        <option selected="selected">Type</option>
        <option value="school">School</option>
        <option value="restaurant">Restaurant</option>
        <option value="hospital">Hospital</option>
        <option value="bank">Bank</option>
        <option value="atm">ATM</option>
        <option value="police">Police</option>
        <option value="post_office">Post Office</option>
        <option value="pharmacy">Pharmacy</option>
        <option value="beauty_salon">Beauty parlour</option>
    </select>
    <br>

    <button class="btn btn-danger btn-lg pull-right" id="btnSearch"
onclick="return searchPlaces()">SEARCH</button>

</div>
<br><br><br>

    <div id="map"></div>
</div>
</div>
<!--
<script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyA7Jd8GVBrGPYIXdjU
BUYXXQQzn8p5EV8w&signed_in=true&callback=initMap"
    async defer>
</script> -->

<script type="text/javascript" src="cordova.js"></script>
    <script type="text/javascript" src="js/index.js"></script>
    <script type="text/javascript">
        app.initialize();
    </script>

</body>
</html>

```

## **About (about.html):**

```
<!DOCTYPE html>

<html>
  <head>
    <meta charset="utf-8" />
    <title>About</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link                                     rel="stylesheet"
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
    <script
src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script
  >
    <style>
      html, body{
        background: transparent url(img/black.jpg);
      }
      .color{
        color: azure;
      }
      .box{
        background-color: #2C2C2C;

        padding: 5px;
        border: 15px solid #1a0000;
        margin: 25px;
        border-radius: 5px;
      }
    </style>
  </head>

  <body>
```

```

<nav class="navbar navbar-inverse" >
  <div class="container-fluid">
    <div style="float:left;">
      <a class="navbar-brand" href="#"><span class="glyphicon glyphicon-
cog"></span> ABOUT</a>
    </div>
    <div style="float:right;">
      <ul class="nav navbar-nav">
        <li class="dropdown"><a class="dropdown-toggle" data-toggle="dropdown"
href="#">Menu <span class="glyphicon glyphicon-align-justify">
</span></a>
          <ul class="dropdown-menu">
            <li><a href="index.html"><span class="glyphicon glyphicon-
home"></span> Home</a></li>
            <li><a href="findmap.html"><span class="glyphicon glyphicon-
search"></span> Search by Place</a></li>
            <li><a href="fimp.html"><span class="glyphicon glyphicon-map-
marker"></span> Nearby You</a></li>
            <li><a href="nearest.html"><span class="glyphicon glyphicon-map-
marker"></span> Nearest 5</a></li>

          </ul>
        </li>
      </ul>
    </div>
  </div>
</nav>

```

```

<div class="container-fluid color">
  <h1>Some Words About This Application</h1>
  <p>Place Finder is a user freindly android application. It will be easier to find out
your desire places for any specific location, or place nearby you.
</p>
  <p><span class="glyphicon glyphicon-record"></span> Mission: Makes more
user friendly and easy to use.</p>
  <p><span class="glyphicon glyphicon-record"></span> Skills : Delivering fast
and excellent results.</p>

```

```
<h3><u><b>Application Information</b></u></h3>
  <p>Released: 14 January 2016</p>
  <p>Version: 1.0 </p>
  <p>Category: Project Final</p>
  <p>Layout: Responsive</p>
  <p>Bootstrap: Bootstrap 3.x</p>
  <p>Platform: Phonegap</p>
  <p>Language: Web Language (Java script, CSS, HTML)</p>
```

```
  </div>
<div class="container-fluid box responsive">
  <br>
  <button type="button" class="btn btn-info btn-block"><span class="glyphicon glyphicon-earphone"></span> +880171.....</button><br>
  <button type="button" class="btn btn-info btn-block"><span class="glyphicon glyphicon-envelope"></span> apps@findplace.com</button><br>
  <button type="button" class="btn btn-info btn-block"><span class="glyphicon glyphicon-th-large"></span> ABC</button><br>
```

```
</div>
  <script type="text/javascript" src="cordova.js"></script>
  <script type="text/javascript" src="js/index.js"></script>
  <script type="text/javascript">
    app.initialize();
  </script>
</body>
</html>
```

# Reference

---

1. [http://www.w3schools.com/html/html\\_basic.asp](http://www.w3schools.com/html/html_basic.asp)
2. [http://www.w3schools.com/css/css\\_intro.asp](http://www.w3schools.com/css/css_intro.asp)
3. [http://www.w3schools.com/js/js\\_intro.asp](http://www.w3schools.com/js/js_intro.asp)
4. [http://www.w3schools.com/bootstrap/bootstrap\\_get\\_started.asp](http://www.w3schools.com/bootstrap/bootstrap_get_started.asp)
5. <http://getbootstrap.com/javascript/>
6. <https://www.pubnub.com/blog/2014-10-22-how-to-convert-your-javascript-app-into-an-android-app-with-phonegap/>
7. <https://developers.google.com/maps/documentation/javascript/examples/>
8. <http://phonegap.com/install/>