

PROJECT REPORT

SOFTWARE DESIGN HANDHELD
DEVICES

DIPTANU DAS

OVERVIEW

Search Simplifier is an android application which is used to search various categories of location depending upon user's interest and save the point of interests in a local file named as favorite. The user can later on directly see his favorite spots from his folder without the need of searching for that location. This application provides information related to Restaurants, Stores and Bookstore depending upon what location user enters.

STRATEGY

This application is developed using Eclipse classic 3.7, JDK 6, and SDK Android platform 2.3.3 Eclipse plug-in. Eclipse plug-in for android is used to provide and emulator to show the application.

XML layout are used to design the GUI of the application. Few XMLs are designed using LinearLayout and others are designed using RelativeLayout to make the UI look more fascinating. Bitmap is used to place the background image to the home screen.

All the buttons were actually image which acts like a button. These image icons are taken from internet to give the application a better look and feel.

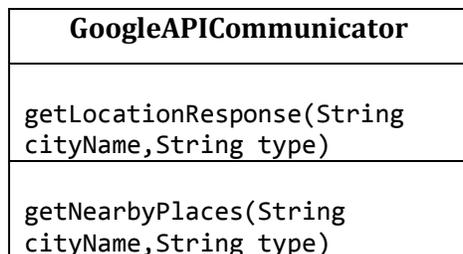
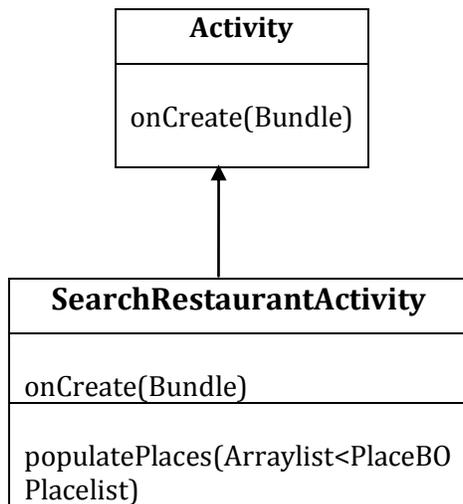
This application uses two APIs: Google Places API and Google Geo Location API. Google Places API retrieves nearby point of interest and corresponding details based on keyword and location inputs. Since, Google Places API gives information based on longitude and Latitude, the Google Geo Location API is used to fetch the location pointing to that particular longitude and latitude. This shows the point of interest within 500 radius. The request is sent in the form of HTTP URL which in return sends an XML or JSON form response. This application receives the XML response.

This application is using DOM Parser and XPath. DOM Parser is used to parse XML from Google Places API and Google Geo Location API and XPath is used to navigate through elements and attributes in an XML document.

The user is able to select and object and save it as a favorite. This data is stored in a file system which can be seen anytime depending upon user's interest. The FileOutputStream writes bytes to a file. The same information is retrieved and shown back to the user once he goes to favorite screen.

CLASS DIAGRAM

The following class diagram is an example of one section of searching restaurants. In a similar way, the same flow is implemented for searching Stores and Book store.



As you can see in this class diagram, SearchRestaurantActivity class extends Activity Class. The entire lifetime of an activity starts with the first Call to onCreate(Bundle). An activity will do all the setup of “global” state in onCreate().

SearchRestaurantActivity class extends Activity Class. Therefore, it will inherit all the properties of super class Activity. The populatePlaces(Arraylist<PlaceBO Placelist) gets all the values from the XML provided by Google Places API and displays it in the screen. In this application, for restaurants, it is populating the values of Restaurant Name, Location and Rating.

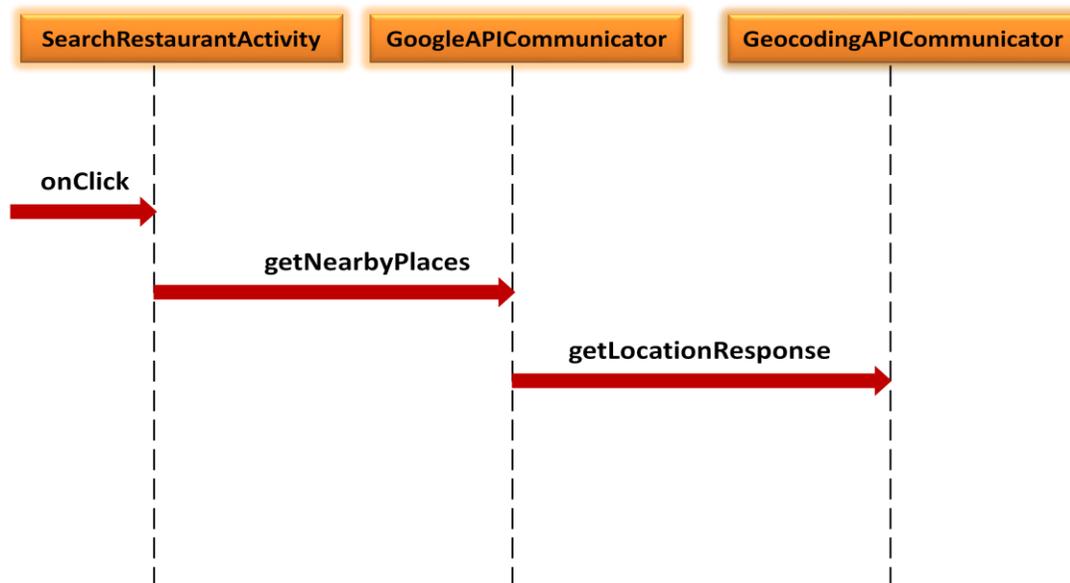
GoogleAPICommunicator class fetches the information from Google Places API. A place search request is sent in the form of a HTTP URL with certain parameter passed along with it. (example, key, type, radius etc). Which in return sends the response in the form of XML or JSON. This application is getting the response in the form of XML.

The getLocationResponse(String cityName, String type) gets the places in terms of longitude and latitude. The getNearbyPlaces (String cityName, String type) gets the XML Response from Google Places API. It gets the value of restaurant name, vicinity (location) and rating. This values get set in PlaceBO list in the form of an array.

StorageHelper
<code>saveInfo(String info,FileOutputStream fos)</code>

This StorageHelper Class captures the information for a particular restaurant and store it in the favorite once the user clicks on the save button. This values are stored in the internal file system of Android in pipe separated tokens. The FileOutputStream writes bytes to a file. The same information is retrieved and shown back to the user once he goes to favorite screen.

SEQUENCE DIAGRAM:



PROJECT SCREEN SHOT WITH DETAILS

HOME SCREEN:



This screen shot is the home page of the application “Search Simplifier”. Currently this application provides three types of location that user can search for. They are: Restaurants, Store, Book Store. User can click on the image and the new page will open where user will be prompted to enter the location and this application will give the list. The user can select one or many location and save it in favorite. Favorite tab bookmarks whatever user wishes to save. That way, user do not have to run the search again.

Here, Bitmap is used to add the background image. This home page is built in XML using Relative Layout provided by Android. All the four images of Restaurant, Store, Books and Favorite are acting as a button.

RESTAURANT LOCATION:



This is the screen shot of Restaurant Location. This page is opened once the user clicks on the restaurant icon on the home page. The user will be prompted to type the location where he/she wants to search the list of location. In this screen shot, the list of restaurants is showing based on the location, Washington DC. The information of restaurant comes in three fields: 1) Restaurant name, 2) Address of Restaurant, 3) Rating. At the end there is save button. If the user clicks on the save button, that particular information about restaurant will be saved in favorite folder.

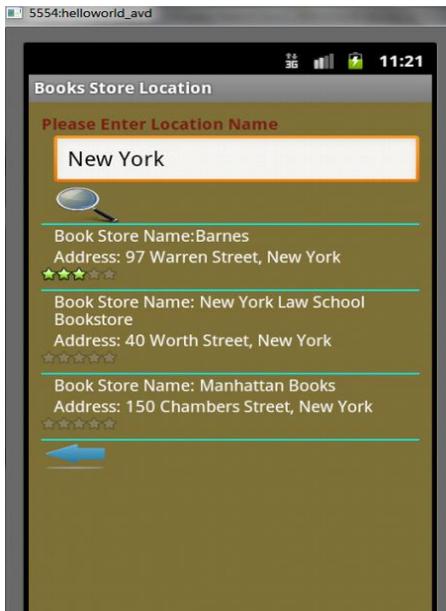
This information is fetched from Google places API whose type is set equal to restaurants.

FAVORITE SCREEN:



Once the user clicks on the save button as mentioned in the earlier screen, that particular information gets saved in the favorite folder. This is nothing but book-marking the information so that user don't have to search again. This gives the user flexibility to use this application in a more friendly way. This captures following information: 1) Name of Restaurant, 2) Address of Restaurant, 3) Star rating of the restaurant. By clicking on the back button the user can go to the home page to add more favorites.

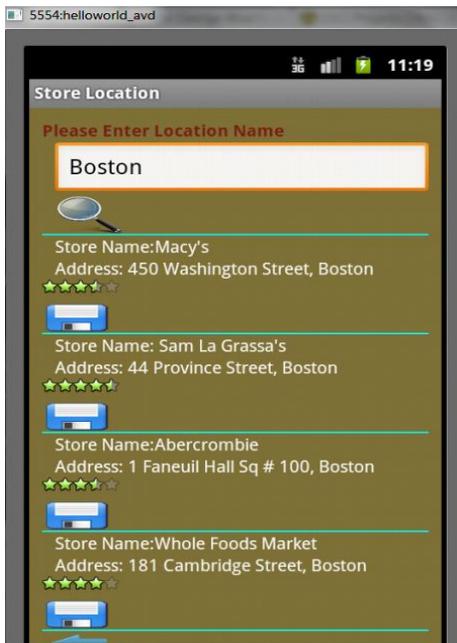
BOOK STORE LOCATION:



This is the screen shot of Book Store Location. This page is opened once the user clicks on the Book icon on the home page. The user will be prompted to type the location where he/she wants to search the list of location. In this screen shot, the list of book stores is showing based on the location, New York. The information of book store comes in three fields: 1) Book Store Name, 2) Address of book Store, 3) Rating of the book store.

This information is fetched from Google places API whose type is set equal to book_type. The back button takes the user to the home page.

STORE LOCATION:



This is the screen shot of Stores Location. This page is opened once the user clicks on the store icon on the home page. The user will be prompted to type the location where he/she wants to search the list of store location. In this screen shot, the list of stores is showing based on the location, Boston. The information of store comes in three fields: 1) Store name, 2) Address of Store, 3) Rating of the store. At the end there is save button. If the user clicks on the save button, that particular information about store will be saved in favorite folder.

This information is fetched from Google places API whose type is set equal to store.

DIFFICULTIES:

Figuring out the layout at the start was difficult but as I explored more it became easy for me. I also got a memory leak error when I tried to add new images. Creating a new avd helped me removing the problem.

CONCLUSION

Overall I am very much pleased with the final outcome of my project. The Application accomplishes all the functionalities that I have outlined in my proposal. Search Simplifier is fully functional and achieves all of my initial goals, but going forward I intended to add many other features like user can use GPS navigation to locate the selected location, User can check-in at place, linking it to facebook and twitter so that the user can post the check-in place, improve the graphics and port it to Android and potentially IOS.