

ภาคผนวก



แบบสอบถาม

เรื่อง

ความพึงพอใจของผู้ใช้บริการแอปพลิเคชันบนมือถือสำหรับค้นหาเส้นทางรถท่องเที่ยว  
กรณีศึกษาจังหวัดเชียงราย

ตอนที่ 1 ข้อมูลเกี่ยวกับสถานภาพของผู้ตอบแบบสอบถามข้อมูลทั่วไป

1. เพศ

ชาย       หญิง

2. อายุ ..... ปี

3. ระดับการศึกษา

นักเรียน     ปริญญาตรี     ปริญญาโท     ปริญญาเอก     อื่นๆ

4. อาชีพ

อาจารย์     นักศึกษา     เจ้าหน้าที่     อื่นๆ

5. ท่านเคยใช้แอปพลิเคชันเพื่อช่วยตัดสินใจในการท่องเที่ยวหรือไม่

ไม่เคย     เคย ระบุ.....

**ตอนที่ 2** ความพึงพอใจของผู้ทดลองใช้แอปพลิเคชันบนมือถือสำหรับค้นหาเส้นทางรถท่องเที่ยว  
กรณีศึกษา จังหวัดเชียงราย

**คำชี้แจง** โปรดทำเครื่องหมาย ✓ ลงในช่องว่างทางขวามือของท่านตามความเป็นจริงจากการ  
ทดลองใช้แอปพลิเคชันบนมือถือค้นหาเส้นทางรถท่องเที่ยว กรณีศึกษา จังหวัดเชียงราย

**ตารางที่ 4** ตารางแบบสอบถาม

ความพึงพอใจในการทดลองใช้ แอปพลิเคชันบนมือถือสำหรับ ค้นหาเส้นทางรถท่องเที่ยว กรณีศึกษา จังหวัดเชียงราย	ระดับความพึงพอใจ				
	น้อยที่สุด(1)	น้อย(2)	ปานกลาง(3)	มาก(4)	มากที่สุด(5)
1. แอปพลิเคชันมีขั้นตอนที่ง่ายไม่ สลับซับซ้อน					
2. แอปพลิเคชันตอบสนองความ ต้องการของผู้ใช้บริการได้ อย่างรวดเร็ว					
3. แอปพลิเคชันมีความสะดวก สำหรับบุคคลทั่วไป					
4. ความประทับใจเกี่ยวกับ แอปพลิเคชัน					
5. ความพึงพอใจในภาพรวมของ รายละเอียดแอปพลิเคชัน					

**ตอนที่ 3** ข้อคิดเห็น และข้อเสนอแนะ เพิ่มเติม

**ข้อคิดเห็น**

.....

**ข้อเสนอแนะเพิ่มเติม**

.....

ขอขอบพระคุณที่ให้ความร่วมมือในการตอบแบบสอบถามครั้งนี้

ภาคผนวก ก

ความพึงพอใจของผู้ทดลองใช้แอปพลิเคชันบนมือถือสำหรับค้นหาเส้นทางการ  
ท่องเที่ยว ทัศนศึกษา จังหวัดเชียงราย

ภาคผนวก ข

โค้ดโปรแกรม

แอปพลิเคชันบนมือถือสำหรับค้นหาเส้นทางการท่องเที่ยว กรณีศึกษา จังหวัดเชียงราย มีทั้งหมด 10 ส่วน แต่ยกส่วนสำคัญมา 4 ส่วนคือ MainActivity , MainActivity2 , MainActivity3 และ O\_DATA

Main Activity ส่วนของหน้าแรกการตั้งค่าของแอปพลิเคชันต่างๆ

```

1  package th.ac.crru.project.travel;
2  import android.app.Activity;
3  import android.os.Bundle;
4  import java.util.ArrayList;
5  import java.util.Date;
6  import edu.asu.emit.qyan.alg.control.YenTopKShortestPathsAlg;
7  import edu.asu.emit.qyan.alg.model.Graph;
8  import edu.asu.emit.qyan.alg.model.Path;
9  import edu.asu.emit.qyan.alg.model.VariableGraph;
10 import android.app.ProgressDialog;
11 import android.content.Context;
12 import android.content.Intent;
13 import android.content.SharedPreferences;
14 import android.database.Cursor;
15 import android.database.sqlite.SQLiteDatabase;
16 import android.util.Log;
17 import android.view.View;
18 import android.view.View.OnClickListener;
19 import android.widget.Button;
20 import android.widget.ImageView;
21 import android.widget.Toast;
22 public class MainActivity extends Activity {
23     private Button tta_button; // type of tourist attraction
24     private Button tv_button; // type of vehicle
25     private Button sel_button; // start/end location

```

```
26     private Button dt_button; // duration of travel
27     private Button filter_button; // filter tourist attraction
28     private Button buttonNext1; // Next1
29     private ImageView language;
30     private SharedPreferences.Editor editor;
31     private SharedPreferences sharedPreferences;
32     private Context context;
33     private ArrayList<Integer> listSelectTA;
34     private TADatabaseHelper dbHelper;
35     private SQLiteDatabase db;
36     @Override
37     protected void onCreate(Bundle savedInstanceState) {
38         super.onCreate(savedInstanceState);
39         setContentView(R.layout.activity_main);
40         // context
41         context = this;
42         // view matching
43         tta_button = (Button) findViewById(R.id.settings_tta_button);
44         tv_button = (Button) findViewById(R.id.settings_tv_button);
45         sel_button = (Button) findViewById(R.id.settings_sel_button);
46         filter_button = (Button) findViewById(R.id.settings_filter_button);
47         dt_button = (Button) findViewById(R.id.settings_dt_button);
48         buttonNext1 = (Button) findViewById(R.id.buttonNext1);
49         language = (ImageView) findViewById(R.id.imageViewL);
50         // data preparation
51         O_Data.setData(this);
52         // share preferences
53         sharedPreferences = this.getSharedPreferences(O_Data.FILENAME_LANGUAGE,
54             MODE_PRIVATE);
55         if (O_Data.language.equals("th")) {
56             language.setImageResource(R.drawable.th);
57         } else {
```

```
58     language.setImageResource(R.drawable.eng);
59 }
60 filter_button.setOnClickListener(new OnClickListener() {
61     @Override
62     public void onClick(View v) {
63         Intent intent = new Intent(MainActivity.this,
64             Activity_GridViewTA.class);
65         startActivity(intent);
66     }
67 });
68 language.setOnClickListener(new OnClickListener() {
69     @Override
70     public void onClick(View arg0) {
71         if (O_Data.language.equals("th")) {
72             O_Data.language = "eng";
73             language.setImageResource(R.drawable.eng);
74         } else {
75             O_Data.language = "th";
76             language.setImageResource(R.drawable.th);
77         }
78         // O_Data.clearStartEndTouristAttraction(context);
79         // save to phone (shared preferences)
80         editor = sharedPreferences.edit();
81         editor.putString(O_Data.FILENAME_LANGUAGE, O_Data.language);
82         editor.commit();
83     }
84 });
85 tta_button.setOnClickListener(new OnClickListener() {
86     @Override
87     public void onClick(View arg0) {
88         Intent intent = new Intent(MainActivity.this,
89             Activity_ListviewTTA.class);
```

```
90         startActivity(intent);
91     }
92 });
93 tv_button.setOnClickListener(new OnClickListener() {
94     @Override
95     public void onClick(View arg0) {
96         Intent intent = new Intent(MainActivity.this,
97             Activity_Radio.class);
98         startActivity(intent);
99     }
100 });
101 sel_button.setOnClickListener(new OnClickListener() {
102     @Override
103     public void onClick(View arg0) {
104         Intent intent = new Intent(MainActivity.this,
105             Activity_Spinner.class);
106         startActivity(intent);
107     }
108 });
109 dt_button.setOnClickListener(new OnClickListener() {
110     @Override
111     public void onClick(View arg0) {
112         Intent intent = new Intent(MainActivity.this,
113             Activity_Time.class);
114         startActivity(intent);
115     }
116 });
117 buttonNext1.setOnClickListener(new OnClickListener() {
118     @Override
119     public void onClick(View arg0) {
120         boolean b = true;
121         for (int i = 0; i < O_Data.typeOfTouristAttraction.size(); i++) {
```

```
122         if (O_Data.typeOfTouristAttraction.get(i).isChecked()) {
123             b = false;
124             break;
125         }
126     }
127     if (b) {
128         Toast.makeText(getBaseContext(),
129             "Please select the type of tourist attraction",
130             Toast.LENGTH_LONG).show();
131     } else if (O_Data.typeOfVehicle.equals("")) {
132         Toast.makeText(getBaseContext(),
133             "Please select the type of vehicle",
134             Toast.LENGTH_LONG).show();
135     } else if (O_Data.startLocation.equals("")
136         || O_Data.endLocation.equals("")) {
137         Toast.makeText(getBaseContext(),
138             "Please select the start/end location",
139             Toast.LENGTH_LONG).show();
140     } else if (O_Data.durationH == 0 && O_Data.durationM == 0) {
141         Toast.makeText(getBaseContext(),
142             "Please set the duration of travel",
143             Toast.LENGTH_LONG).show();
144     } else {
145         if(O_Data.touristAttractionFilter==null){
146             O_Data.filterTouristAttraction();
147         }
148         final ProgressDialog dialog = ProgressDialog.show(context,
149             "Please wait", "Computing...", true, false);
150         new Thread(new Runnable() {
151             @Override
152             public void run() {
153                 // TODO Auto-generated method stub
```



```
186         tour.addTour(O_Data.touristAttractionFilter
187             .get(j));
188         break;
189     }
190 }
191 }
192     tour.setDistance(p.get_weight());
193     tour.calTime();
194     tour.calAvgFavorite();
195     if (tour.getTime() > timeSet ) { // limit at 10000
196         // path
197         break;
198     }
199     if (!checkdDuplicateTour(tour)
200         && checkSelectTA(tour.getTour() )){
201         O_Data.tourList.add(tour);
202     }
203 }
204     end = new Date();
205     O_Data.computingTime = end.getTime() - start.getTime();
206     O_Data.numberOfTour=count;
207     Intent intent = new Intent(MainActivity.this,
208         MainActivity2.class);
209     startActivity(intent);
210 } catch (Exception e) {
211     Log.d("MainActivity.onCreate", ""+e);
212 }
213     dialog.dismiss();
214 }
215 }).start();
216 }
217 }
```

```
218     });
219 }
220 private void addSeletTA() {
221     listSelectTA = new ArrayList<Integer>();
222     for (int i = 0; i < O_Data.touristAttractionFilter.size(); i++) {
223         if (O_Data.touristAttractionFilter.get(i).getStatus().equals("s")) {
224             listSelectTA.add(O_Data.touristAttractionFilter.get(i).getId());
225         }
226     }
227 }
228 private boolean checkSelectTA(ArrayList<O_TouristAttraction> tour) {
229     if (listSelectTA.size() > tour.size())
230         return false;
231     for (int i = 0; i < listSelectTA.size(); i++) {
232         boolean b = false;
233         for (int j = 0; j < tour.size(); j++) {
234             if (listSelectTA.get(i) == tour.get(j).getId()) {
235                 b = true;
236                 break;
237             }
238         }
239         if (b == false) {
240             return false;
241         }
242     }
243     return true;
244 }
245 /*
246  * if duplicate, return true;
247  */
248 private boolean checkdDuplicateTour(O_Tour tour) {
249     int count;
```

```
250     for (int i = 0; i < O_Data.tourList.size(); i++) {
251         if (O_Data.tourList.get(i).getSize() == tour.getSize()) {
252             count = 0;
253             for (int j = 0; j < tour.getSize(); j++) { // row
254                 for (int k = 0; k < O_Data.tourList.get(i).getSize(); k++) { // column
255                     if (tour.getTouristAttraction(j).getId() == O_Data.tourList
256                         .get(i).getTouristAttraction(k).getId()) {
257                         count++;
258                         break;
259                     }
260                 }
261             }
262             if (count == tour.getSize()) {
263                 return true;
264             }
265         }
266     }
267     return false;
268 }
269 }
270
```

MainActivity2 คือ ส่วนของการแสดงโปรแกรมทัวร์ขอสถานที่ท่องเที่ยว

```
1 package th.ac.crru.project.travel;
2 import java.util.ArrayList;
3 import java.util.Collections;
4 import java.util.List;
5 import android.app.Activity;
6 import android.app.AlertDialog;
7 import android.content.Context;
8 import android.content.DialogInterface;
9 import android.content.Intent;
10 import android.content.SharedPreferences;
11 import android.database.Cursor;
12 import android.database.sqlite.SQLiteDatabase;
13 import android.os.Bundle;
14 import android.util.Log;
15 import android.view.LayoutInflater;
16 import android.view.View;
17 import android.view.ViewGroup;
18 import android.widget.AdapterView;
19 import android.widget.AdapterView.OnItemClickListener;
20 import android.widget.ArrayAdapter;
21 import android.widget.BaseAdapter;
22 import android.widget.Button;
23 import android.widget.ImageView;
24 import android.widget.ListView;
25 import android.widget.Spinner;
26 import android.widget.TextView;
27 import android.widget.Toast;
28 import android.widget.AdapterView.OnItemClickListener;
29 import edu.asu.emit.qyan.alg.control.YenTopKShortestPathsAlg;
30 import edu.asu.emit.qyan.alg.model.Graph;
```

```
30 import edu.asu.emit.qyan.alg.model.Path;
31 import edu.asu.emit.qyan.alg.model.VariableGraph;
32 public class MainActivity2 extends Activity {
33     private Button buttonNext2, buttonPrevious2, buttonViewDetails;
34     private Spinner spinnerOrder;
35     // context
36     private Context context;
37     // list view & adapter & data
38     private ListView listView;
39     private ArrayList<O_SettingsItem> tourList;
40     private SettingsAdapter listAdapter;
41     // share preferences
42     private SharedPreferences sharePreferences;
43     private SharedPreferences.Editor editor;
44     @Override
45     protected void onCreate(Bundle savedInstanceState) {
46         // TODO Auto-generated method stub
47         super.onCreate(savedInstanceState);
48         setContentView(R.layout.activity_main2);
49         // context
50         context = this;
51         // view matching
52         spinnerOrder = (Spinner) findViewById(R.id.spinnerOrderTour);
53         buttonNext2 = (Button) findViewById(R.id.buttonNext2);
54         buttonPrevious2 = (Button) findViewById(R.id.buttonPrevious2);
55         buttonViewDetails = (Button) findViewById(R.id.buttonViewDetails);
56         listView = (ListView) findViewById(R.id.listViewTour);
57         sharePreferences = this.getSharedPreferences(
58             O_Data.FILENAME_SELOCATION,
59             MODE_PRIVATE);
60         // Log.d("MainActivity2.onCreate ", "size TA = " +
61         // O_Data.touristAttraction.size());
```

```

62         AlertDialog.Builder builder = new AlertDialog.Builder(this);
63         builder.setTitle("Computing time")
64         .setIcon(getResources().getDrawable(R.drawable.ic_launcher))
65         .setMessage(O_Data.computingTime+" ms.\nNumber of tours (before
66 remove)           "+O_Data.numberOfTour)
67         .setPositiveButton("OK", new DialogInterface.OnClickListener() {
68
69             @Override
70             public void onClick(DialogInterface dialog, int which) {
71                 // TODO Auto-generated method stub
72                 dialog.dismiss();
73             }
74         });
75         AlertDialog alert = builder.create();
76         alert.show();
77         //         Toast.makeText(getBaseContext(), "Computing time:
78 "+O_Data.computingTime+" ms.", Toast.LENGTH_LONG)
79         //             .show();
80         showTour("");
81         // _____ spinner
82         order_____
83         // share preferences
84         sharedPreferences =
85         this.getSharedPreferences(O_Data.FILENAME_ORDERTOUR,
86             MODE_PRIVATE);
87         ArrayAdapter<String> dataAdapterS = new ArrayAdapter<String>(this,
88             android.R.layout.simple_spinner_item,
89         O_Data.orderTour);
90         dataAdapterS
91
92         .setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
93         spinnerOrder.setAdapter(dataAdapterS);

```

```

94         // set selection
95         if(!O_Data.seletedOrderTour.equals("")) {
96             spinnerOrder.setSelection(dataAdapterS
97                 .getPosition(O_Data.seletedOrderTour));
98         }
99         spinnerOrder.setOnItemClickListener(new OnItemSelectedListener() {
100             @Override
101             public void onItemClick(AdapterView<?> parent, View view,
102                 int position, long id) {
103                 editor = sharedPreferences.edit();
104                 editor.putString(O_Data.FILENAME_ORDERTOUR,
105                     O_Data.orderTour.get(position));
106                 editor.commit();
107                 O_Data.seletedOrderTour =
108 O_Data.orderTour.get(position);
109                 O_Data.selectedTour = -1;// set default value
110                 showTour(O_Data.orderTour.get(position));
111             }
112             @Override
113             public void onNothingSelected(AdapterView<?> arg0) {
114                 // TODO Auto-generated method stub
115             }
116         });
117         buttonNext2.setOnClickListener(new View.OnClickListener() {
118             @Override
119             public void onClick(View v) {
120                 if (O_Data.selectedTour < 0) {
121                     Toast.makeText(getApplicationContext(), "Please
122 select a tour",
123
124 Toast.LENGTH_LONG).show();
125                 } else {

```

```

126             Intent intent = new Intent(MainActivity2.this,
127                 MainActivity3.class);
128             startActivity(intent);
129         }
130     }
131 });
132 buttonPrevious2.setOnClickListener(new View.OnClickListener() {
133     @Override
134     public void onClick(View v) {
135         Intent intent = new Intent(MainActivity2.this,
136             MainActivity.class);
137         startActivity(intent);
138     }
139 });
140 buttonViewDetails.setOnClickListener(new View.OnClickListener() {
141     @Override
142     public void onClick(View v) {
143         if (O_Data.selectedTour == -1) {
144             Toast.makeText(getApplicationContext(), "Please
145 select a tour",
146             Toast.LENGTH_LONG).show();
147         } else {
148             Intent intent = new Intent(MainActivity2.this,
149                 Activity_ListImage.class);
150             startActivity(intent);
151         }
152     }
153 });
154 // list view event
155 listView.setOnItemClickListener(new
156 AdapterView.OnItemClickListener() {

```

```

158         @Override
159         public void onItemClick(AdapterView<?> adapter, View view,
160             int position, long id) {
161             // change checker value
162             for (int i = 0; i < tourList.size(); i++) {
163                 tourList.get(i).setChecked(false);
164             }
165             tourList.get(position).setChecked(true);
166             O_Data.selectedTour = position;
167             listAdapter.notifyDataSetChanged();
168             // save to phone (shared preferences)
169             // editor = sharedPreferences.edit();
170             // editor.putBoolean(O_Data.FILENAME_TYPEOFTA
171 + position, data
172             // .get(position).isChecked());
173             // editor.commit();
174         }
175     });
176 }
177 private void showTour(String orderBy) {
178     if (orderBy.equals("Sort by Time")) {
179         Collections.sort(O_Data.tourList, new O_Tour.CompareTime());
180     } else if (orderBy.equals("Sort by Favorite")) {
181         Collections.sort(O_Data.tourList, new
182 O_Tour.CompareFavorite());
183     } else {
184         Collections.sort(O_Data.tourList, new O_Tour.CompareSize());
185     }
186     tourList = new ArrayList<O_SettingsItem>();
187     String s = "";
188     if (O_Data.language.equals("th")) {
189         s = " สถานที่ท่องเที่ยว, ";

```

```

190         } else {
191             s = " Tourist Attractions, ";
192         }
193     for (int i = 0; i < O_Data.tourList.size(); i++) {
194         // Log.d("MainActivity2.onCreate ", "time "
195         // + O_Data.tourList.get(i).getTime());
196         String header = ""+(i+1)+" ";
197         if (O_Data.indexEndLocation == O_Data.ENDTA) {
198             header += O_Data.tourList.get(i).getSize() - 1;
199         } else {
200             header += O_Data.tourList.get(i).getSize();
201         }
202         header += s + O_Data.tourList.get(i).getTimeFormat();
203         tourList.add(new O_SettingsItem(header, O_Data.tourList.get(i)
204             .getDetail(), false));
205     }
206     // list view & adapter
207     listAdapter = new SettingsAdapter(); // use tourList variable
208     listView.setAdapter(listAdapter);
209 }
210 private class SettingsAdapter extends BaseAdapter {
211     private ViewHolder viewHolder;
212     @Override
213     public int getCount() {
214         // TODO Auto-generated method stub
215         return tourList.size();
216     }
217     @Override
218     public Object getItem(int arg0) {
219         // TODO Auto-generated method stub
220         return null;
221     }

```

```
222         @Override
223         public long getItemId(int arg0) {
224             // TODO Auto-generated method stub
225             return 0;
226         }
227         @Override
228         public View getView(int position, View view, ViewGroup parent) {
229             if (view == null) {
230                 view = LayoutInflater.from(context).inflate(
231                     R.layout.layout_listview_item, null);
232                 viewHolder = new ViewHolder();
233                 // view matching
234                 viewHolder.title = (TextView) view
235
236                 .findViewById(R.id.listview_item_title);
237                 viewHolder.description = (TextView) view
238
239                 .findViewById(R.id.listview_item_description);
240                 viewHolder.checker = (ImageView) view
241
242                 .findViewById(R.id.listview_item_checker);
243                 // set tag
244                 view.setTag(viewHolder);
245             } else {
246                 viewHolder = (ViewHolder) view.getTag();
247             }
248             // assign
249             viewHolder.title.setText(tourList.get(position).getTitle());
250             viewHolder.description.setText(tourList.get(position)
251                 .getDescription());
252             if (tourList.get(position).isChecked() {
```

```
253
254     viewHolder.checker.setImageDrawable(getResources().getDrawable(
255         R.drawable.tick1));
256     } else {
257         viewHolder.checker.setImageDrawable(null);
258     }
259     return view;
260 }
261 private class ViewHolder {
262     public TextView title, description;
263     public ImageView checker;
264 }
265 }
266 }
```

MainActivity3 คือ ส่วนของการแสดงแผนที่

```
1 package th.ac.crru.project.travel;
2 import java.util.ArrayList;
3 import android.app.Activity;
4 import android.content.Intent;
5 import android.graphics.Color;
6 import android.net.Uri;
7 import android.os.Bundle;
8 import android.util.Log;
9 import android.view.View;
10 import android.webkit.WebSettings;
11 import android.webkit.WebView;
12 import android.widget.AdapterView;
13 import android.widget.AdapterView.OnItemClickListener;
14 import android.widget.ArrayAdapter;
15 import android.widget.Button;
16 import android.widget.Spinner;
17 import android.widget.TextView;
18 public class MainActivity3 extends Activity {
19     private Spinner spinnerTour;
20     private Button buttonRoute;
21     private O_Tour tour;
22     private ArrayList<String> listTA;
23     private int positionSeleted;
24     @Override
25     protected void onCreate(Bundle savedInstanceState) {
26         // TODO Auto-generated method stub
27         super.onCreate(savedInstanceState);
28         setContentView(R.layout.activity_main3);
29         // view matching
```

```

30         spinnerTour = (Spinner) findViewById(R.id.spinnerTour);
31         buttonRoute = (Button) findViewById(R.id.buttonRoute);
32         tour = O_Data.tourList.get(O_Data.selectedTour);
33         addList();
34         showMap(tour);
35         // _____ start
36         location_____
37         ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this,
38             android.R.layout.simple_spinner_item, listTA);
39         dataAdapter
40         .setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
41         spinnerTour.setAdapter(dataAdapter);
42         spinnerTour.setOnItemClickListener(new OnItemSelectedListener() {
43             @Override
44             public void onItemClick(AdapterView<?> parent, View view,
45                 int position, long id) {
46                 // TODO Auto-generated method stub
47                 // Toast.makeText(getApplicationContext(),
48                 // "" + O_Data.startEndLocation.get(position),
49                 // Toast.LENGTH_SHORT).show();//
50                 positionSeleted = position;
51                 ((TextView)
52                 parent.getChildAt(0)).setTextColor(Color.rgb(0, 30,
53                     10));
54                 ((TextView) parent.getChildAt(0)).setTextSize(18);
55                 if(position>0){
56                     O_Tour t = new O_Tour();
57                     t.addTour(tour.getTouristAttraction(position-1));
58                     t.addTour(tour.getTouristAttraction(position));
59                     showMap(t);

```

```

60             }else{
61                 showMap(tour);
62             }
63         }
64         @Override
65         public void onNothingSelected(AdapterView<?> arg0) {
66             // TODO Auto-generated method stub
67         }
68     });
69     buttonRoute.setOnClickListener(new View.OnClickListener() {
70         @Override
71         public void onClick(View v) {
72             double latitude =
73     tour.getTouristAttraction(positionSeleted).getLatitude();
74             double longitude =
75     tour.getTouristAttraction(positionSeleted).getLongitude();
76             String label = "";
77             if(O_Data.language.equals("th")){
78                 label =
79     tour.getTouristAttraction(positionSeleted).getNameTh();
80             }else{
81                 label =
82     tour.getTouristAttraction(positionSeleted).getNameEng();
83             }
84             String uriBegin = "geo:" + latitude + "," + longitude;
85             String query = latitude + "," + longitude + "(" + label +
86     ")";
87             String encodedQuery = Uri.encode(query);
88             String uriString = uriBegin + "?q=" + encodedQuery +
89     "&z=16";

```

```

90         Uri uri = Uri.parse(uriString);
91         Intent intent = new Intent(Intent.ACTION_VIEW, uri);
92         startActivity(intent);
93     }
94 });
95 }
96 private void addList() {
97     listTA = new ArrayList<String>();
98     for (int i = 0; i < tour.getSize(); i++) {
99         if (i == 0) {
100             if (O_Data.language.equals("th")) {
101                 listTA.add("แสดงทั้งหมด");
102             } else {
103                 listTA.add("Show All");
104             }
105         } else {
106             if (O_Data.language.equals("th")) {
107                 listTA.add(i + " "
108                     + tour.getTouristAttraction(i -
109 1).getNameTh()
110                     + "->"
111                     +
112 tour.getTouristAttraction(i).getNameTh());
113             } else {
114                 listTA.add(i + " "
115                     + tour.getTouristAttraction(i -
116 1).getNameEng()
117                     + "->"
118                     +
119 tour.getTouristAttraction(i).getNameEng());

```

```

120         }
121     }
122 }
123 }
124 private void showMap(O_Tour tour) {
125     String source = "", destination = "", l = "";
126     for (int i = 0; i < tour.getSize(); i++) {
127         l = tour.getTouristAttraction(i).getLatitude() + ","
128             + tour.getTouristAttraction(i).getLongitude();
129         if (i == 0) {
130             source = l;
131         } else {
132             if (i >= 2)
133                 destination += "+to:";
134             destination += l;
135         }
136     }
137     // Log.d("MainActivity3.onCreate ", source + "---" + destination);
138     String data = "<body style='margin: 0px; padding: 0px;'> <iframe
139 style='margin: 0px; padding: 0px; height: 100%; width: 100%;' frameborder='0'
140 scrolling='no' src='https://maps.google.co.th/maps?f=d&source=s_d&saddr="
141         + source
142         + "&daddr="
143         + destination
144         + "&hl=en&aq=&sll="
145         + source
146         +
147         "&sspn=0.104777,0.155697&mra=ls&ie=UTF8&ll="
148         + source

```

```
149                                     + "&spn=0.25,0.25&t=m&output=embed'  
150     /></body>";  
151     //         Log.d("MainActivity3.onCreate ", data);  
152         WebView webView = (WebView) findViewById(R.id.webViewMap);  
153         // Enable Javascript  
154         WebSettings webSettings = webView.getSettings();  
155         webSettings.setJavaScriptEnabled(true);  
156         webView.loadData(data, "text/html", "UTF-8");  
157     }  
158 }
```

O\_DATA คือ ส่วนของการเก็บข้อมูลเอาไว้ใน O\_DATA

```

1  package th.ac.crru.project.travel;
2  import java.util.ArrayList;
3  import edu.asu.emit.qyan.alg.control.YenTopKShortestPathsAlg;
4  import android.content.Context;
5  import android.content.SharedPreferences;
6  import android.database.Cursor;
7  import android.database.sqlite.SQLiteDatabase;
8  import android.util.Log;
9  public class O_Data {
10     public static int numberOfTour=0;
11         public static long computingTime=0;
12         public static int ENDTA = 555;
13         public static String language = "th";
14         public static ArrayList<O_Tour> tourList;
15         public static int selectedTour = -1;
16         public static ArrayList<String> orderTour;
17         public static final String FILENAME_ORDERTOUR = "orderTour";
18         public static String seletedOrderTour;
19         public static ArrayList<O_TouristAttraction> touristAttraction;
20         public static ArrayList<O_TouristAttraction> touristAttractionFilter;
21         // public static YenTopKShortestPathsAlg yenAlg;
22         public static ArrayList<O_DistanceTA> tADistance; // tourist attraction
23             // distance, only
24                 // selected
25         public static ArrayList<O_SettingsItem> typeOfTouristAttraction;
26         public static final String FILENAME_TYPEOFTA = "typeOfTA";
27         public static String typeOfVehicleSelect;
28         public static ArrayList<String> typeOfVehicle;
29         public static final String FILENAME_TYPEOFV = "typeOfV";

```

```
30     public static ArrayList<String> startEndLocation;
31     public static String startLocation;
32     public static String endLocation;
33     public static int indexStartLocation;
34     public static int indexEndLocation;
35     public static final String FILENAME_SELOCATION = "startEndLocation";
36     public static int durationH;
37     public static int durationM;
38     public static final String FILENAME_DURATION = "durationOfTravel";
39     public static final String FILENAME_LANGUAGE = "language";
40     public static void setData(Context context) {
41         // set language
42         setLanguage(context);
43         // set type of tourist attraction
44         setTypeOfTouristAttraction(context);
45         // set start/end location, must set after set the type of tourist
46         // attraction
47         setTouristAttraction(context);
48         // set type of vehicle
49         setTypeOfVehicle(context);
50         // set duration of travel(H:M)
51         setDurationOfTravel(context);
52         // set order tour
53         setOrderTour(context);
54     }
55     private static void setLanguage(Context context) {
56         // share preferences
57         SharedPreferences sharedPreferences = context.getSharedPreferences(
58             FILENAME_LANGUAGE, 0);
```

```

59         sharePreferences =
60     context.getSharedPreferences(FILENAME_LANGUAGE, 0);
61         String l = sharePreferences.getString(FILENAME_LANGUAGE, "th");
62         O_Data.language = l;
63     }
64     private static void setOrderTour(Context context) {
65         SharedPreferences sharePreferences = context.getSharedPreferences(
66             FILENAME_ORDERTOUR, 0);
67         sharePreferences =
68     context.getSharedPreferences(FILENAME_ORDERTOUR, 0);
69         seletedOrderTour =
70     sharePreferences.getString(FILENAME_ORDERTOUR, "");
71         // set default
72         // seletedOrderTour="Sort by #Tourist Attraction";
73         if (seletedOrderTour.equals(""))
74             seletedOrderTour = "Sort by #Tourist Attraction";
75         orderTour = new ArrayList<String>();
76         orderTour.add("Sort by #Tourist Attraction");
77         orderTour.add("Sort by Time");
78         orderTour.add("Sort by Favorite");
79     }
80     private static void setTypeOfTouristAttraction(Context context) {
81         SharedPreferences sharePreferences = context.getSharedPreferences(
82             FILENAME_TYPEOFTA, 0);
83         // data preparation
84         typeOfTouristAttraction = new ArrayList<O_SettingsItem>();
85         TADatabaseHelper dbHelper;
86         SQLiteDatabase db;
87         dbHelper = new TADatabaseHelper(context);
88         db = dbHelper.getReadableDatabase();

```

```

89         Cursor cursor;
90         cursor = db.rawQuery(
91             "select type from touristAttraction group by type", null);
92         int i = 0;
93         if (cursor.getCount() != 0) {
94             cursor.moveToFirst();
95             do {
96                 String t =
97 cursor.getString(cursor.getColumnIndex("type"));
98                 typeOfTouristAttraction.add(new O_SettingsItem(t, "",
99
100         sharedPreferences.getBoolean(FILENAME_TYPEOFTA + i,
101                                     false)));
102                 i++;
103             } while (cursor.moveToNext());
104         } else {
105             Log.d("Database tourist attraction", "No result of select");
106         }
107     }
108     public static void setTouristAttraction(Context context) {
109         SharedPreferences sharedPreferences = context.getSharedPreferences(
110             FILENAME_TYPEOFTA, 0);
111         sharedPreferences =
112 context.getSharedPreferences(FILENAME_SELOCATION, 0);
113         startLocation = sharedPreferences.getString(FILENAME_SELOCATION
114             + "_start", "");
115         endLocation = sharedPreferences.getString(FILENAME_SELOCATION +
116         "_end",
117             "");
118         TADatabaseHelper dbHelper;

```

```

119         SQLiteDatabase db;
120         dbHelper = new TADatabaseHelper(context);
121         db = dbHelper.getReadableDatabase();
122         Cursor cursor;
123         ArrayList<String> list = new ArrayList<String>();
124         for (int i = 0; i < typeOfTouristAttraction.size(); i++) {
125             if (typeOfTouristAttraction.get(i).isChecked()) {
126                 list.add(typeOfTouristAttraction.get(i).getTitle());
127             }
128         }
129         String s = "(";
130         for (int i = 0; i < list.size(); i++) {
131             s += "" + list.get(i) + """;
132             if (i < list.size() - 1) {
133                 s += ",";
134             }
135         }
136         s += ")";
137         cursor = db.rawQuery("select * from touristAttraction where type in "
138             + s, null);
139         touristAttraction = new ArrayList<O_TouristAttraction>(); // selectedTA
140         // only in
141         // the
142         // selected
143         // types
144         if (cursor.getCount() != 0) {
145             cursor.moveToFirst();
146             do {
147                 double latitude = cursor.getDouble(cursor
148                     .getColumnIndex("latitude"));

```

```
149         double longitude = cursor.getDouble(cursor
150             .getColumnIndex("longitude"));
151         int id = cursor.getInt(cursor.getColumnIndex("id"));
152         int time = cursor.getInt(cursor.getColumnIndex("time"));
153         int favorite =
154     cursor.getInt(cursor.getColumnIndex("favorite"));
155         String name_th = cursor.getString(cursor
156             .getColumnIndex("name_th"));
157         String name_eng = cursor.getString(cursor
158             .getColumnIndex("name_eng"));
159         String type =
160     cursor.getString(cursor.getColumnIndex("type"));
161         String description_th = cursor.getString(cursor
162             .getColumnIndex("description_th"));
163         String description_eng = cursor.getString(cursor
164             .getColumnIndex("description_eng"));
165         String image =
166     cursor.getString(cursor.getColumnIndex("image"));
167         String status = cursor.getString(cursor
168             .getColumnIndex("status"));
169         touristAttraction.add(new O_TouristAttraction(latitude,
170             longitude, id, time, favorite, name_th,
171     name_eng, type,
172             description_th, description_eng, image,
173     status));
174         } while (cursor.moveToNext());
175         cursor.close();
176         db.close();
177     } else {
178         Log.d("Database tourist atteaction", "No result of select");
```

```

179         }
180     }
181     public static void removeDestination() {
182         for (int i = touristAttractionFilter.size() - 1; i >= 0; i--) {
183             if (touristAttractionFilter.get(i).getType().equals("Destination")) {
184                 if (touristAttractionFilter.get(i).getId() !=
185 indexStartLocation
186                                     && touristAttractionFilter.get(i).getId()
187 != indexEndLocation) {
188                     touristAttractionFilter.remove(i);
189                     // Log.d("Database filterTouristAttraction", "id-
190 >"+i);
191                 }
192             }
193         }
194     }
195     public static void filterTouristAttraction() {
196         touristAttractionFilter = new ArrayList<O_TouristAttraction>();
197         try {
198             for (int i = 0; i < touristAttraction.size(); i++) {
199                 if (!touristAttraction.get(i).getStatus().equals("n")) {
200                     O_TouristAttraction touristAttractionClone =
201 (O_TouristAttraction) touristAttraction
202                                     .get(i).clone();
203
204                 touristAttractionFilter.add(touristAttractionClone);
205             }
206         }
207     } catch (CloneNotSupportedException e) {
208         // TODO Auto-generated catch block

```

```
209             e.printStackTrace();
210         }
211     }
212     public static void setStartEndTouristAttraction() {
213         filterTouristAttraction();
214         startEndLocation = new ArrayList<String>();
215         for (int i = 0; i < touristAttractionFilter.size(); i++) {
216             if (O_Data.language.equals("th")) {
217                 startEndLocation
218
219             .add(touristAttractionFilter.get(i).getNameTh());
220             } else {
221                 startEndLocation.add(touristAttractionFilter.get(i)
222                     .getNameEng());
223             }
224         }
225     }
226     public static void clearStartEndTouristAttraction(Context context) {
227         findIndexStartLocation();
228         findIndexEndLocation();
229         if (indexStartLocation == -1 || indexEndLocation == -1) {
230             startEndLocation = null;
231             SharedPreferences sharePreferences;
232             SharedPreferences.Editor editor;
233             sharePreferences = context.getSharedPreferences(
234                 O_Data.FILENAME_SELOCATION, 0);
235             editor = sharePreferences.edit();
236             editor.putString(O_Data.FILENAME_SELOCATION + "_start",
237                 "");
```

```
238         editor.putString(O_Data.FILENAME_SELOCATION + "_end",
239     "");
240         editor.commit();
241         startLocation = "";
242         endLocation = "";
243     }
244 }
245 private static void setTypeOfVehicle(Context context) {
246     SharedPreferences sharePreferences = context.getSharedPreferences(
247         FILENAME_TYPEOFTA, 0);
248     sharePreferences =
249     context.getSharedPreferences(FILENAME_TYPEOFV, 0);
250     typeOfVehicleSelect =
251     sharePreferences.getString(FILENAME_TYPEOFV, "");
252     typeOfVehicle = new ArrayList<String>();
253     typeOfVehicle.add("Walking");
254     typeOfVehicle.add("Bicycle");
255     typeOfVehicle.add("Motorcycle");
256     typeOfVehicle.add("Car");
257 }
258 private static void setDurationOfTravel(Context context) {
259     SharedPreferences sharePreferences = context.getSharedPreferences(
260         FILENAME_TYPEOFTA, 0);
261     sharePreferences =
262     context.getSharedPreferences(FILENAME_DURATION, 0);
263     durationH = sharePreferences.getInt(FILENAME_DURATION + "H", 0);
264     durationM = sharePreferences.getInt(FILENAME_DURATION + "M",
265     0);
266 }
267 public static void setDistanceOfTouristAttraction(Context context) {
```

```

268         if (indexStartLocation == indexEndLocation) {
269             int index = -1;
270             for (int i = 0; i < touristAttractionFilter.size(); i++) {
271                 if (touristAttractionFilter.get(i).getId() ==
272 indexStartLocation) {
273                     index = i;
274                     break;
275                 }
276             }
277             try {
278                 O_TouristAttraction touristAttractionClone =
279 (O_TouristAttraction) touristAttractionFilter
280                     .get(index).clone();
281                 touristAttractionClone.setId(ENDTA);
282                 touristAttractionClone.setTime(0);
283                 touristAttractionFilter.add(touristAttractionClone);
284                 indexEndLocation = ENDTA;
285             } catch (Exception e) {
286                 Log.d("O_Data.setDistanceOfTouristAttraction", "" + e);
287             }
288         }
289         ArrayList<String> list = new ArrayList<String>();
290         for (int i = 0; i < touristAttractionFilter.size(); i++) {
291             list.add("" + touristAttractionFilter.get(i).getId());
292             // Log.d("O_Data.setDistanceOfTouristAttraction", ""
293             // + touristAttractionFilter.get(i).getId()
294             // + touristAttractionFilter.get(i).getNameTh());
295         }
296         String s = "(";
297         for (int i = 0; i < list.size(); i++) {

```

```

298         s += "" + list.get(i) + "";
299         if (i < list.size() - 1) {
300             s += ",";
301         }
302     }
303     s += ")";
304     TADatabaseHelper dbHelper;
305     SQLiteDatabase db;
306     dbHelper = new TADatabaseHelper(context);
307     db = dbHelper.getReadableDatabase();
308     Cursor cursor;
309     cursor = db.rawQuery("select * from distance where ta1 in " + s
310         + " and ta2 in " + s, null);
311     tADistance = new ArrayList<O_DistanceTA>();
312     if (cursor.getCount() != 0) {
313         cursor.moveToFirst();
314         do {
315             int ta1 = cursor.getInt(cursor.getColumnIndex("ta1"));
316             int ta2 = cursor.getInt(cursor.getColumnIndex("ta2"));
317             double distance = cursor.getDouble(cursor
318                 .getColumnIndex("distance"));
319             // _____ cal distance time _____
320             double rate = 10.00;// Km/hour
321             if (O_Data.typeOfVehicleSelect.equals("Walking")) {
322                 rate = 4.00;
323             } else if (O_Data.typeOfVehicleSelect.equals("Bicycle")) {
324                 rate = 12.00;
325             } else if (O_Data.typeOfVehicleSelect.equals("Motorcycle")) {
326                 rate = 50.00;
327             } else if (O_Data.typeOfVehicleSelect.equals("Car")) {

```

```

328         rate = 50.00;
329     }
330     double timeD = (distance / rate) * 60.00;
331         if (indexEndLocation == ENDTA) {
332             if (ta1 == indexStartLocation) {
333                 tADistance.add(new
334 O_DistanceTA(ENDTA, ta2, timeD+getTouristAttraction(ta2).getTime()));
335                 tADistance.add(new O_DistanceTA(ta2, ENDTA,
336 timeD+getTouristAttraction(ta2).getTime()));
337             } else if (ta2 == indexStartLocation) {
338                 tADistance.add(new O_DistanceTA(ta1,
339 ENDTA, timeD+getTouristAttraction(ta1).getTime()));
340                 tADistance.add(new O_DistanceTA(ENDTA, ta1,
341 timeD+getTouristAttraction(ta1).getTime()));
342             }
343         }
344         tADistance.add(new O_DistanceTA(ta1, ta2,
345 timeD+getTouristAttraction(ta2).getTime()));
346         tADistance.add(new O_DistanceTA(ta2, ta1,
347 timeD+getTouristAttraction(ta1).getTime()));
348     } while (cursor.moveToNext());
349     } else {
350         Log.d("Database distance", "No result of select");
351     }
352 }
353 static private O_TouristAttraction getTouristAttraction(int id){
354     for(int i=0;i<touristAttraction.size();i++){
355         if(id==touristAttraction.get(i).getId()){
356             return touristAttraction.get(i);
357         }

```

```
358     }
359     return null;
360 }
361     static public void findIndexStartLocation() {
362         int id = -1;
363         for (int i = 0; i < touristAttractionFilter.size(); i++) {
364             if (touristAttractionFilter.get(i).getNameEng()
365                 .equals(startLocation)
366                 || touristAttractionFilter.get(i).getNameTh()
367                 .equals(startLocation)) {
368                 id = touristAttractionFilter.get(i).getId();
369                 break;
370             }
371         }
372         indexStartLocation = id;
373     }
374     static public void findIndexEndLocation() {
375         int id = -1;
376         for (int i = 0; i < touristAttractionFilter.size(); i++) {
377             if
378 (touristAttractionFilter.get(i).getNameEng().equals(endLocation)
379                 || touristAttractionFilter.get(i).getNameTh()
380                 .equals(endLocation)) {
381                 id = touristAttractionFilter.get(i).getId();
382                 break;
383             }
384         }
385         indexEndLocation = id;
386     }
387 }
```