Shopping Recommendation App
Code Listing

By

Ayhan Sahin
School of Science Department of Computing &Networking
Institute of Technology Carlow, Kilkenny Road
County Carlow, Republic of Ireland
C00145773@itcarlow.ie

28th April 2014

Contents

Revision History

Date	Version	Description	Author
28/04/2014	1.0	Initial & Final	Ayhan Sahin

1 Introduction

The aim of this document is to present all the code used in the development of the project. This document only presents the Java Classes in the project which directly influences the project. The supplementary code materials such as XML file codes are not included in this code listing. The XML codes are used to manage resources and layouts within this project.

2 Entity Classes

2.1 Comment.java

```
package ie.itcarlow.sra.entity;

import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("Comment")
public class Comment extends ParseObject {
	public Comment() {

	}

	public void setProductId(String productId) {
		put("productId", productId);
	}

	public String getProductId() {
		return getString("productId");
	}

	public void setUserId(String userId) {
		put("userId", userId);
	}

	public String getUserId() {
		return getString("userId");
	}

	public void setCommentContent(String comment) {
		put("content", comment);
	}

	public String getCommentContent() {
```

```
                return getString("content");
        }
}
```

2.2 MessageObject.java

```
package ie.itcarlow.sra.entity;

import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("MessageObject")
public class MessageObject extends ParseObject {

        String senderId;
        String receiverId;
        String messageSubject;
        String messageContent;

        public String getSenderId() {
                return getString("senderId");
        }

        public void setSenderId(String senderId) {
                put("senderId", senderId);
        }

        public String getReceiverId() {
                return getString("receiverId");
        }

        public void setReceiverId(String receiverId) {
                put("receiverId", receiverId);
        }

        public String getMessageSubject() {
                return getString("messageSubject");
```

```
        }

        public void setMessageSubject(String messageSubject) {
                put("messageSubject", messageSubject);
        }

        public String getMessageContent() {
                return getString("messageContent");
        }

        public void setMessageContent(String messageContent) {
                put("messageContent", messageContent);
        }

}
```

2.3 Product.java

```
package ie.itcarlow.sra.entity;

import com.parse.ParseClassName;
import com.parse.ParseFile;
import com.parse.ParseGeoPoint;
import com.parse.ParseObject;

@ParseClassName("Product")
public class Product extends ParseObject {

        public Product() {

        }

        public void setCatId(int catId) {
                put("catId", catId);
        }
```

```
public int getCatId() {
	return getInt("catId");
}

public void setSubCatId(int subCatId) {
	put("subCatId", subCatId);
}

public int getSubCatId() {
	return getInt("subCatId");
}

public void setDescription(String description) {
	put("description", description);
}

public String getDescription() {
	return getString("description");
}

public void setPrice(double price) {
	put("price", price);
}

public double getPrice() {
	return getDouble("price");
}

public void setBussName(String bussName) {
	put("bussName", bussName);
}

public String getBussName() {
	return getString("bussName");
}
```

```
	public void setBussContact(String bussContact) {
		put("bussContact", bussContact);
	}

	public String getBussContact() {
		return getString("bussContact");
	}

	public void setUserId(String userId) {
		put("userId", userId);
	}

	public String getUserId() {
		return getString("userId");
	}

	public void setLocation(ParseGeoPoint location) {
		put("location", location);
	}

	public ParseGeoPoint getLocation() {
		return getParseGeoPoint("location");

	}

	public void setPhoto(ParseFile photo) {
		put("photo", photo);
	}

	public ParseFile getPhoto() {
		return getParseFile("photo");
	}
}
```

2.4 ProductRequest.java

```
package ie.itcarlow.sra.entity;
```

```
import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("ProductRequest")
public class ProductRequest extends ParseObject {
	public ProductRequest() {

	}

	public void setSenderId(String sender) {
		put("senderId", sender);
	}

	public String getSenderId() {
		return getString("senderId");
	}

	public void setReceiverId(String receiver) {
		put("receiverId", receiver);
	}

	public String getReceiverId() {
		return getString("receiverId");
	}

	public void setDescription(String description) {
		put("description", description);
	}

	public String getDescription() {
		return getString("description");
	}
}
```

2.5 Rating.java

```
package ie.itcarlow.sra.entity;

import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("Rating")
public class Rating extends ParseObject {

	public Rating() {

	}

	public void setRecommendationId(String recommendationId) {
		put("recommendationId", recommendationId);
	}

	public String getRecommendationId() {
		return getString("recommendationId");
	}

	public void setProductId(String productId) {
		put("productId", productId);
	}

	public String getProductId() {
		return getString("productId");
	}

	public void setRatingValue(String ratingValue) {
		put("ratingValue", ratingValue);
	}

	public String getRatingValue() {
		return getString("ratingValue");
```

```
        }

}
```

2.6 Recommendation.java

```
package ie.itcarlow.sra.entity;

import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("Recommendation")
public class Recommendation extends ParseObject {

        public Recommendation() {

        }

        public void setProductId(String productId) {
                put("productId", productId);
        }

        public String getProductId() {
                return getString("productId");
        }

        public void setUserId(String userId) {
                put("userId", userId);
        }

        public String getUserId() {
                return getString("userId");
        }
}
```

2.7 ShoppingListItem.java

```
package ie.itcarlow.sra.entity;

import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("ShoppingListItem")
public class ShoppingListItem extends ParseObject {
	public ShoppingListItem() {

	}

	public void setShoppingListId(String shoppingListId) {
		put("shoppingListId", shoppingListId);
	}

	public String getShoppingListId() {
		return getString("shoppingListId");
	}

	public void setItemName(String itemName) {
		put("itemName", itemName);
	}

	public String getItemName() {
		return getString("itemName");
	}

	public void setItemDesc(String itemDescription) {
		put("itemDesc", itemDescription);
	}

	public String getItemDesc() {
		return getString("itemDesc");
	}
```

```
	public void setItemCat(String itemCat) {
		put("itemCat", itemCat);
	}

	public String getItemCat() {
		return getString("itemCat");
	}
}
```

2.8 ShoppingListObject.java

```
package ie.itcarlow.sra.entity;

import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("ShoppingListObject")
public class ShoppingListObject extends ParseObject {
	public ShoppingListObject() {

	}

	public void setListName(String listName) {
		put("listName", listName);
	}

	public String getListName() {
		return getString("listName");
	}

	public void setUserId(String userId) {
		put("userId", userId);
	}
```

```
        public String getUserId() {
                return getString("userId");
        }
        /*
         * public void setListDescription(String listDesc){
         * put("listDesc",listDesc); }
         *
         * public String getListDesc(){ return getString("listDesc"); }
         */
}
```

3 Activity Classes

3.1 AppCamera.class

```
package ie.itcarlow.sra;

import java.io.ByteArrayOutputStream;
import java.io.IOException;

import android.app.Fragment;
import android.app.FragmentManager;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Matrix;
import android.hardware.Camera;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.SurfaceHolder;
import android.view.SurfaceHolder.Callback;
import android.view.SurfaceView;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageButton;
import android.widget.Toast;
```

```
import com.parse.ParseException;
import com.parse.ParseFile;
import com.parse.SaveCallback;

public class AppCamera extends Fragment {

	private Camera camera;
	private SurfaceView surfaceView;
	private ParseFile photo;
	private ImageButton snapButton;

	@Override
	public View onCreateView(LayoutInflater inflater, ViewGroup parent,
			Bundle savedInstanceState) {
		View v = inflater.inflate(R.layout.camera, parent, false);
		snapButton = (ImageButton) v.findViewById(R.id.camera_button);

		if (camera == null) {
			try {

				openCamera();
			} catch (Exception e) {
				// Log.e(TAG, "No camera with exception: " + e.getMessage());
				snapButton.setEnabled(false);
				Toast.makeText(getActivity(), "No camera detected",
						Toast.LENGTH_LONG).show();
			}
		}

		snapButton.setOnClickListener(new View.OnClickListener() {

			@Override
			public void onClick(View v) {
				if (camera == null)
					return;
```

```
                camera.takePicture(new Camera.ShutterCallback() {

                        @Override
                        public void onShutter() {
                                // nothing to do
                        }

                }, null, new Camera.PictureCallback() {

                        @Override
                        public void onPictureTaken(byte[] data, Camera camera) {
                                saveScaledPhoto(data);
                        }

                });

        }
});

surfaceView = (SurfaceView) v.findViewById(R.id.camera_surface_view);
SurfaceHolder holder = surfaceView.getHolder();
holder.addCallback(new Callback() {

        @Override
        public void surfaceCreated(SurfaceHolder holder) {
                try {
                        if (camera != null) {
                                camera.setDisplayOrientation(90);
                                camera.setPreviewDisplay(holder);
                                camera.startPreview();
                        }
                } catch (IOException e) {
                        // Log.e(TAG, "Error setting up preview", e);
                }
        }
```

```
                @Override
                public void surfaceChanged(SurfaceHolder holder, int format,
                                int width, int height) {
                        // nothing to do here
                }

                @Override
                public void surfaceDestroyed(SurfaceHolder holder) {
                        // nothing here
                }

        });

        return v;
}

public void openCamera() {

        if (Camera.getNumberOfCameras() < 2) {
                camera = Camera.open(0);
                snapButton.setEnabled(true);
        } else {
                camera = Camera.open(1);
                snapButton.setEnabled(true);
        }

}

private void saveScaledPhoto(byte[] data) {

        // Resize photo from camera byte array
        Bitmap mealImage = BitmapFactory.decodeByteArray(data, 0, data.length);
        Bitmap mealImageScaled = Bitmap.createScaledBitmap(mealImage, 200, 200
                        * mealImage.getHeight() / mealImage.getWidth(), false);

        // Override Android default landscape orientation and save portrait
```

```
        Matrix matrix = new Matrix();
        matrix.postRotate(90);
        Bitmap rotatedScaledMealImage = Bitmap.createBitmap(mealImageScaled, 0,
                        0, mealImageScaled.getWidth(), mealImageScaled.getHeight(),
                        matrix, true);

        ByteArrayOutputStream bos = new ByteArrayOutputStream();
        rotatedScaledMealImage.compress(Bitmap.CompressFormat.JPEG, 100, bos);

        byte[] scaledData = bos.toByteArray();

        // Save the scaled image to Parse
        photo = new ParseFile("product.jpg", scaledData);
        photo.saveInBackground(new SaveCallback() {

                @Override
                public void done(ParseException e) {
                        if (e == null) {
                                addPicToCurrentListing(photo);

                        } else {
                                Toast.makeText(getActivity(),
                                                "Error saving: " + e.getMessage(),
                                                Toast.LENGTH_LONG).show();
                        }
                }
        });
}

private void addPicToCurrentListing(ParseFile photo) {
        ((NewProductActivity) getActivity()).getCurrentProduct()
                        .setPhoto(photo);
        FragmentManager fm = getActivity().getFragmentManager();
        fm.popBackStack("ProductFragment",
                        FragmentManager.POP_BACK_STACK_INCLUSIVE);
}
```

```
}
```

3.2 DrawerActivity.java

```
package ie.itcarlow.sra;

import java.util.Arrays;
import java.util.List;

import android.app.Activity;
import android.content.Intent;
import android.content.res.Configuration;
import android.os.Bundle;
import android.support.v4.app.ActionBarDrawerToggle;
import android.support.v4.view.GravityCompat;
import android.support.v4.widget.DrawerLayout;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;

import com.parse.ParseUser;

public abstract class DrawerActivity extends Activity {
	private DrawerLayout mDrawerLayout;
	private ListView mDrawerList;
	private ActionBarDrawerToggle mDrawerToggle;

	private CharSequence mDrawerTitle;
	private CharSequence mTitle;
	private String[] mNavigationTitles;
	private int index;

	@Override
```

```
protected void onCreate(Bundle savedInstanceState) {
	super.onCreate(savedInstanceState);

	// Set the draw layout
	DrawerLayout();
	// Highlights the current activity in the drawer
	List<String> navigationList = Arrays.asList(mNavigationTitles);
	index = navigationList.indexOf(mDrawerTitle);
	mDrawerList.setItemChecked(index, true);

}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
	if (mDrawerToggle.onOptionsItemSelected(item)) {
		return true;
	}
	return true;
}

/* The click listener for ListView in the navigation drawer */
private class DrawerItemClickListener implements
		ListView.OnItemClickListener {
	@Override
	public void onItemClick(AdapterView<?> parent, View view, int position,
			long id) {
		selectItem(position);
	}
}

public void selectItem(int position) {

	if (index != position) {
		switch (position) {
		case 0:
			Intent feedIntent = new Intent(this, MainActivity.class);
```

```
            startActivity(feedIntent);
            break;
        case 1:
            Intent searchIntent = new Intent(this, SearchActivity.class);
            startActivity(searchIntent);
            break;
        case 2:
            Intent profileIntent = new Intent(this, Profile.class);
            startActivity(profileIntent);
            break;
        case 3:
            Intent shoppinglistIntent = new Intent(this, ShoppingList.class);
            startActivity(shoppinglistIntent);
            break;
        case 4:
            Intent messagesIntent = new Intent(this, Messages.class);
            startActivity(messagesIntent);
            break;
        case 5:
            Intent requestsIntent = new Intent(this, Requests.class);
            startActivity(requestsIntent);
            break;
        case 6:
            Intent friendSearchIntent = new Intent(this, FriendSearch.class);
            startActivity(friendSearchIntent);
            break;
        case 7:
            ParseUser.logOut();
            Intent loginIntent = new Intent(this, Login.class);
            startActivity(loginIntent);
            finish();
            break;
        default:

            break;
```

```
                }
        }

        mDrawerLayout.closeDrawer(mDrawerList);
        // finish();
}

@Override
public void setTitle(CharSequence title) {
        mTitle = title;
        getActionBar().setTitle(mTitle);
}

@Override
protected void onPostCreate(Bundle savedInstanceState) {
        super.onPostCreate(savedInstanceState);
        // Sync the toggle state after onRestoreInstanceState has occurred.
        mDrawerToggle.syncState();
}

@Override
public void onConfigurationChanged(Configuration newConfig) {
        super.onConfigurationChanged(newConfig);
        // Pass any configuration change to the drawer toggls
        mDrawerToggle.onConfigurationChanged(newConfig);
}

public void DrawerLayout() {
        mTitle = mDrawerTitle = getTitle();
        mNavigationTitles = getResources().getStringArray(
                        R.array.navigation_array);
        mDrawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);
        mDrawerList = (ListView) findViewById(R.id.left_drawer);

        // set a custom shadow that overlays the main content when the drawer
        // opens
```

```
        mDrawerLayout.setDrawerShadow(R.drawable.drawer_shadow,
                        GravityCompat.START);
        // set up the drawer's list view with items and click listener
        mDrawerList.setAdapter(new ArrayAdapter<String>(this,
                        R.layout.drawer_list_item, mNavigationTitles));
        mDrawerList.setOnItemClickListener(new DrawerItemClickListener());

        // enable ActionBar app icon to behave as action to toggle nav drawer
        getActionBar().setDisplayHomeAsUpEnabled(true);
        getActionBar().setHomeButtonEnabled(true);

        // ActionBarDrawerToggle ties together the the proper interactions
        // between the sliding drawer and the action bar app icon
        mDrawerToggle = new ActionBarDrawerToggle(this, /* host Activity */
        mDrawerLayout, /* DrawerLayout object */
        R.drawable.ic_drawer, /* nav drawer image to replace 'Up' caret */
        R.string.drawer_open, /* "open drawer" description for accessibility */
        R.string.drawer_close /* "close drawer" description for accessibility */
        ) {
                @Override
                public void onDrawerClosed(View view) {
                        getActionBar().setTitle(mTitle);
                        invalidateOptionsMenu(); // creates call to
                                                                        // onPrepareOptionsMenu()
                }

                @Override
                public void onDrawerOpened(View drawerView) {
                        getActionBar().setTitle(mDrawerTitle);
                        invalidateOptionsMenu(); // creates call to
                }
        };
        mDrawerLayout.setDrawerListener(mDrawerToggle);

}
```

```
}
```

3.3 Forgot_Password.java

```
package ie.itcarlow.sra;

import android.app.Activity;
import android.app.AlertDialog;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.parse.ParseException;
import com.parse.ParseUser;
import com.parse.RequestPasswordResetCallback;

public class Forgot_password extends Activity {
	EditText userEmailText;
	Button sendButton;
	String userEmailParam;
	String emailSuccess = "A Password Reset Email was successfully sent.";

	@Override
	protected void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.forgot_password);

		userEmailText = (EditText) findViewById(R.id.forgot_email);

		sendButton = (Button) findViewById(R.id.forgot_password_send);

		sendButton.setOnClickListener(new View.OnClickListener() {
```

```
                @Override
                public void onClick(View v) {
                        Validate.hasText(userEmailText);
                        userEmailParam = userEmailText.getText().toString();
                        // TODO Auto-generated method stub
                        ParseUser.requestPasswordResetInBackground(userEmailParam,
                                        new RequestPasswordResetCallback() {
                                                @Override
                                                public void done(ParseException e) {
                                                        if (e == null) {
                                                                showSuccessAlert();
                                                                finish();
                                                        } else {
                                                                Toast.makeText(
                                                                                getApplicationContext(),
                                                                                e.getMessage().toUpperCase()
                                                                                                .toString(),
                                                                                Toast.LENGTH_SHORT).show();
                                                        }
                                                }
                                        });
                }
        });
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.forgot_password, menu);
        return true;
}

public void showSuccessAlert() {
        new AlertDialog.Builder(this).setTitle("Success")
                        .setMessage(emailSuccess).setNeutralButton("OK", null).show();
```

```
        }

        public void showFailAlert(ParseException e) {
                new AlertDialog.Builder(this).setTitle("Error")
                                .setMessage(e.getMessage()).setNeutralButton("OK", null).show();
        }
}
```

3.4 FriendSearch.java

```
package ie.itcarlow.sra;

import java.util.List;

import android.app.ProgressDialog;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Toast;

import com.parse.FindCallback;
import com.parse.ParseException;
import com.parse.ParseQuery;
import com.parse.ParseUser;

public class FriendSearch extends DrawerActivity {
        Button searchButton;
        String emailAddress;
        EditText emailAddressText;

        @Override
        protected void onCreate(Bundle savedInstanceState) {

                setContentView(R.layout.activity_friend_search);
```

```
        super.onCreate(savedInstanceState);

        searchButton = (Button) findViewById(R.id.friendSearch_search);

        searchButton.setOnClickListener(new View.OnClickListener() {

                @Override
                public void onClick(View v) {
                        // TODO Auto-generated method stub
                        emailAddressText = (EditText) findViewById(R.id.friendSearch_field);
                        emailAddress = emailAddressText.getText().toString();

                        findFriend(emailAddress);

                }
        });
}

private void findFriend(String emailAddress) {

        final ProgressDialog findFriendProgress = ProgressDialog.show(this, "",
                        "Finding Friend...");
        findFriendProgress.show();

        ParseQuery<ParseUser> query = ParseUser.getQuery();
        query.whereEqualTo("email", emailAddress);
        query.findInBackground(new FindCallback<ParseUser>() {

                @Override
                public void done(List<ParseUser> result, ParseException e) {
                        findFriendProgress.dismiss();
                        if (e == null) {
                                // no exception
                                displayListview(result);

                        } else {
```

```
                                Toast.makeText(getApplicationContext(), e.getMessage(),
                                        Toast.LENGTH_LONG).show();
                            }
                        }
                    });

        }

        public void displayListview(List<ParseUser> result) {
                String firstName;
                String lastName;
                String fullName;
                String[] result1 = new String[result.size()];
                int i = 0;
                for (ParseUser user : result) {
                        firstName = user.getString("firstName");
                        firstName = firstName.concat(" ");
                        lastName = user.getString("lastName");
                        fullName = firstName.concat(lastName);
                        fullName.toUpperCase();

                        result1[i] = fullName;
                        i++;
                }
                ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
                                android.R.layout.simple_list_item_1, android.R.id.text1,
                                result1);

                ListView listView = (ListView) findViewById(R.id.friend_search_result_listview);
                listView.setAdapter(adapter);
        }

}
```

3.5 InboxListFragment.java

```
package ie.itcarlow.sra;
```

```
import android.os.Bundle;
import android.support.v4.app.ListFragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.ListAdapter;

public class InboxListFragment extends ListFragment {

	private String inbox[];

	public InboxListFragment() {

		inbox = new String[] { "Inbox 1", "Inbox 2", "Inbox 3", "Inbox 4", };
	}

	@Override
	public void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);

		ListAdapter inboxListAdapter = new ArrayAdapter<String>(getActivity(),
				android.R.layout.simple_list_item_1, inbox);
		setListAdapter(inboxListAdapter);
	}

	@Override
	public View onCreateView(LayoutInflater inflater, ViewGroup container,
			Bundle savedInstanceState) {
		return inflater.inflate(R.layout.messages_list_fragment, container,
				false);
	}
}
```

3.6 Login.java

```
package ie.itcarlow.sra;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.parse.LogInCallback;
import com.parse.ParseException;
import com.parse.ParseUser;

public class Login extends Activity {

	EditText usernameText, passwordText;
	Button loginButton;
	TextView signupTextView, passwordResetTextView;

	@Override
	protected void onCreate(Bundle savedInstanceState) {

		setContentView(R.layout.activity_login);
		super.onCreate(savedInstanceState);

		usernameText = (EditText) findViewById(R.id.username);
		passwordText = (EditText) findViewById(R.id.password);

		loginButton = (Button) findViewById(R.id.sign_in);
		signupTextView = (TextView) findViewById(R.id.register);
		passwordResetTextView = (TextView) findViewById(R.id.forgot_password);
```

```
loginButton.setOnClickListener(new View.OnClickListener() {

	@Override
	public void onClick(View v) {
		// TODO Auto-generated method stub
		String usernameParam = usernameText.getText().toString();
		String passwordParam = passwordText.getText().toString();

		ParseUser.logInInBackground(usernameParam, passwordParam,
				new LogInCallback() {

					@Override
					public void done(ParseUser user, ParseException e) {
						// TODO Auto-generated method stub
						if (e == null) {
							Toast.makeText(getApplicationContext(),
									"Login Successfull",
									Toast.LENGTH_LONG).show();
							gotoMainActivity();
						}

						else
							Toast.makeText(getApplicationContext(),
									e.getMessage().toUpperCase(),
									Toast.LENGTH_LONG).show();
					}
				});

	}
});

signupTextView.setOnClickListener(new View.OnClickListener() {
	@Override
	public void onClick(View v) {
		// TODO Auto-generated method stub
		Intent registerAct = new Intent(getApplicationContext(),
```

```
                                        Signup.class);
                                startActivity(registerAct);
                                finish();
                        }
                });

                passwordResetTextView.setOnClickListener(new View.OnClickListener() {

                        @Override
                        public void onClick(View v) {
                                // TODO Auto-generated method stub
                                Intent passwordReset = new Intent(getApplicationContext(),
                                                Forgot_password.class);
                                startActivity(passwordReset);
                                finish();
                        }
                });
        }

        public void gotoMainActivity() {
                Intent main = new Intent(getApplicationContext(), MainActivity.class);
                startActivity(main);
                finish();
        }
}
```

3.7 MainActivity.java

```
package ie.itcarlow.sra;

import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
```

```
import com.parse.ParseUser;

public class MainActivity extends DrawerActivity {

	@Override
	protected void onCreate(Bundle savedInstanceState) {

		setContentView(R.layout.activity_main);
		super.onCreate(savedInstanceState);

		// Check to see if the user is logged in.If not redirect to Login

		ParseUser sraUser = ParseUser.getCurrentUser();
		if (sraUser == null) {
			Intent registerAct = new Intent(getApplicationContext(),
					Login.class);
			startActivity(registerAct);
			finish();
		}

	}

	@Override
	public boolean onCreateOptionsMenu(Menu menu) {
		getMenuInflater().inflate(R.menu.main, menu);
		return true;
	}

	@Override
	public boolean onOptionsItemSelected(MenuItem item) {
		switch (item.getItemId()) {

		case R.id.new_product: {
			newProduct();
			break;
		}
```

```
                }
                return super.onOptionsItemSelected(item);
        }

        public void newProduct() {
                Intent newProductIntent = new Intent(this, NewProductActivity.class);
                startActivity(newProductIntent);
        }

}
```

3.8 Messages.java

```
package ie.itcarlow.sra;

import android.app.ActionBar;
import android.app.ActionBar.Tab;
import android.app.FragmentTransaction;
import android.os.Bundle;
import android.support.v4.app.FragmentActivity;

public class Messages extends FragmentActivity implements ActionBar.TabListener {

        private static final String STATE_SELECTED_NAVIGATION_ITEM = "selected_navigation_item";

        @Override
        protected void onCreate(Bundle savedInstanceState) {

                super.onCreate(savedInstanceState);
                setContentView(R.layout.activity_messages);

                final ActionBar actionbar = getActionBar();
                actionbar.setNavigationMode(ActionBar.NAVIGATION_MODE_TABS);

                actionbar.addTab(actionbar.newTab().setText("Inbox")
                                .setTabListener(this));
                actionbar.addTab(actionbar.newTab().setText("Outbox")
```

```
                        .setTabListener(this));

        }

        @Override
        public void onRestoreInstanceState(Bundle savedInstanceState) {
                if (savedInstanceState.containsKey(STATE_SELECTED_NAVIGATION_ITEM)) {
                        getActionBar().setSelectedNavigationItem(
                                        savedInstanceState.getInt(STATE_SELECTED_NAVIGATION_ITEM));
                }
        }

        @Override
        public void onTabReselected(Tab tab, FragmentTransaction ft) {
                // TODO Auto-generated method stub

        }

        @Override
        public void onSaveInstanceState(Bundle outState) {
                outState.putInt(STATE_SELECTED_NAVIGATION_ITEM, getActionBar()
                                .getSelectedNavigationIndex());
        }

        @Override
        public void onTabSelected(Tab tab, FragmentTransaction ft) {
                // TODO Auto-generated method stub
                if (tab.getPosition() == 0) {
                        InboxListFragment inboxFragment = new InboxListFragment();
                        getSupportFragmentManager().beginTransaction()
                                        .replace(R.id.content_frame, inboxFragment).commit();
                } else {
                        OutboxListFragment outboxFragment = new OutboxListFragment();
                        getSupportFragmentManager().beginTransaction()
                                        .replace(R.id.content_frame, outboxFragment).commit();
                }
```

```
	}

	@Override
	public void onTabUnselected(Tab tab, FragmentTransaction ft) {
		// TODO Auto-generated method stub

	}

}
```

3.9 NewProductActivity.java

```
package ie.itcarlow.sra;

import ie.itcarlow.sra.entity.Product;
import android.app.Activity;
import android.app.Fragment;
import android.app.FragmentManager;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.Window;

public class NewProductActivity extends Activity {
	private Product product;

	@Override
	protected void onCreate(Bundle savedInstanceState) {
		product = new Product();
		requestWindowFeature(Window.FEATURE_NO_TITLE);
		// getWindow().addFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN);
		super.onCreate(savedInstanceState);

		setContentView(R.layout.activity_new_product);
		FragmentManager manager = getFragmentManager();
		Fragment fragment = manager.findFragmentById(R.id.container);
		if (fragment == null) {
```

```
                        fragment = new ProductFragment();
                        getFragmentManager().beginTransaction()
                                        .add(R.id.container, fragment).commit();
                }
        }

        @Override
        public boolean onCreateOptionsMenu(Menu menu) {

                // Inflate the menu; this adds items to the action bar if it is present.
                getMenuInflater().inflate(R.menu.new_product, menu);
                return true;
        }

        @Override
        public boolean onOptionsItemSelected(MenuItem item) {
                // Handle action bar item clicks here. The action bar will
                // automatically handle clicks on the Home/Up button, so long
                // as you specify a parent activity in AndroidManifest.xml.
                int id = item.getItemId();
                if (id == R.id.action_settings) {
                        return true;
                }
                return super.onOptionsItemSelected(item);
        }

        public Product getCurrentProduct() {
                // TODO Auto-generated method stub
                return product;
        }

}
```

3.10 OutboxListFragment.java

```
package ie.itcarlow.sra;
```

```
import android.os.Bundle;
import android.support.v4.app.ListFragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.ListAdapter;

public class OutboxListFragment extends ListFragment {

	private String outbox[];

	public OutboxListFragment() {

		outbox = new String[] { "Outbox 1", "Outbox 2", "Outbox 3", "Outbox 4", };
	}

	@Override
	public void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);

		ListAdapter outboxListAdapter = new ArrayAdapter<String>(getActivity(),
				android.R.layout.simple_list_item_1, outbox);
		setListAdapter(outboxListAdapter);
	}

	@Override
	public View onCreateView(LayoutInflater inflater, ViewGroup container,
			Bundle savedInstanceState) {
		return inflater.inflate(R.layout.messages_list_fragment, container,
				false);
	}
}
```

3.11 ProductFragment.java

```
package ie.itcarlow.sra;
```

```
import ie.itcarlow.sra.entity.Product;

import java.io.File;

import android.app.Activity;
import android.app.Fragment;
import android.app.FragmentTransaction;
import android.app.ProgressDialog;
import android.content.Context;
import android.os.Bundle;
import android.os.Looper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.view.inputmethod.InputMethodManager;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import com.parse.LocationCallback;
import com.parse.ParseException;
import com.parse.ParseGeoPoint;
import com.parse.ParseImageView;
import com.parse.ParseUser;
import com.parse.SaveCallback;

public class ProductFragment extends Fragment {
	private ImageButton photoButton, geoLocationButton;
	private EditText productDesc, productPrice, storeName, storeContact;
```

```
private Button submitButton, cancelButton;

private TextView mealName;
private Spinner categorySpinner, subCatSpinner;
private ParseImageView productPreview;
private String productDescParam, storeNameParam, storeContactParam,
               locationParam;
double productPriceParam;
ParseGeoPoint currentLocation = new ParseGeoPoint();
ArrayAdapter<CharSequence> spinnerAdapter2;

// String[] categories = {"Sporting Goods","Software","Home&Garden"};
// String[] subCategories = {"abc","123","def"};

@Override
public void onCreate(Bundle savedInstanceState) {
       super.onCreate(savedInstanceState);
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup parent,
               Bundle SavedInstanceState) {
       View v = inflater.inflate(R.layout.new_product_fragment, parent, false);

       productDesc = ((EditText) v.findViewById(R.id.desc_input));

       productPrice = ((EditText) v.findViewById(R.id.price_input));

       storeName = ((EditText) v.findViewById(R.id.bus_name_input));

       storeContact = ((EditText) v.findViewById(R.id.store_contact_input));

       categorySpinner = ((Spinner) v.findViewById(R.id.category_spinner));
       subCatSpinner = ((Spinner) v.findViewById(R.id.subcategory_spinner));

       ArrayAdapter<CharSequence> spinnerAdapter = ArrayAdapter
```

```
                .createFromResource(getActivity(), R.array.cat_array,
                                android.R.layout.simple_spinner_dropdown_item);
categorySpinner.setAdapter(spinnerAdapter);

categorySpinner.setOnItemSelectedListener(new OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> arg0, View arg1,
                        int position, long arg3) {
                if (position == 1) {
                        spinnerAdapter2 = ArrayAdapter.createFromResource(
                                        getActivity(), R.array.elect_subcat,
                                        android.R.layout.simple_spinner_dropdown_item);
                } else if (position == 2) {
                        spinnerAdapter2 = ArrayAdapter.createFromResource(
                                        getActivity(), R.array.vehicles_subcat,
                                        android.R.layout.simple_spinner_dropdown_item);
                }
                subCatSpinner.setAdapter(spinnerAdapter2);
        }

        @Override
        public void onNothingSelected(AdapterView<?> arg0) {
                // TODO Auto-generated method stub

        }
});

geoLocationButton = ((ImageButton) v.findViewById(R.id.current_loc));
geoLocationButton.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {
                // TODO Auto-generated method stub
                final ProgressDialog gpsProgress = ProgressDialog.show(
                                getActivity(), "", "Finding GPS Location...");
                gpsProgress.show();
```

```
                        ParseGeoPoint.getCurrentLocationInBackground(50000,
                                        new LocationCallback() {

                                                @Override
                                                public void done(ParseGeoPoint point,
                                                                ParseException e) {
                                                        // TODO Auto-generated method stub
                                                        gpsProgress.dismiss();
                                                        if (e == null) {
                                                                setCurrentLocation(point);

                                                        } else {
                                                                Toast.makeText(getActivity(),
                                                                                e.getMessage(),
Toast.LENGTH_LONG)
                                                                                .show();
                                                        }
                                                }
                                        });
                        }
                });

                photoButton = ((ImageButton) v.findViewById(R.id.camera));
                photoButton.setOnClickListener(new View.OnClickListener() {

                        @Override
                        public void onClick(View v) {
                                InputMethodManager imm = (InputMethodManager) getActivity()
                                                .getSystemService(Context.INPUT_METHOD_SERVICE);
                                // imm.hideSoftInputFromWindow(mealName.getWindowToken(), 0);

                                startCamera();
                        }
                });

                submitButton = ((Button) v.findViewById(R.id.submit));
```

```
submitButton.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {
                Product newProduct = ((NewProductActivity) getActivity())
                                .getCurrentProduct();

                /*
                 * Validate.hasText(productDesc);
                 * Validate.hasText(productPrice); Validate.hasText(storeName);
                 * Validate.hasText(storeContact);
                 */
                productDescParam = productDesc.getText().toString();

                // productPriceParam = Double.parseDouble(productPrice.getText()
                // .toString());

                storeNameParam = storeName.getText().toString();

                storeContactParam = storeContact.getText().toString();

                newProduct.setDescription(productDescParam);
                newProduct.setPrice(productPriceParam);
                newProduct.setBussName(storeNameParam);
                newProduct.setBussContact(storeContactParam);
                newProduct.setUserId(ParseUser.getCurrentUser().getObjectId());
                newProduct.setLocation(currentLocation);

                newProduct.saveInBackground(new SaveCallback() {

                        @Override
                        public void done(ParseException e) {
                                if (e == null) {
                                        getActivity().setResult(Activity.RESULT_OK);
                                        getActivity().finish();
                                } else {
```

```
                                        Toast.makeText(
                                                getActivity().getApplicationContext(),
                                                "Error saving: " + e.getMessage(),
                                                Toast.LENGTH_SHORT).show();
                                    }
                                }

                            });

                    }
            });

            cancelButton = ((Button) v.findViewById(R.id.cancel));
            cancelButton.setOnClickListener(new View.OnClickListener() {

                    @Override
                    public void onClick(View v) {
                            getActivity().setResult(Activity.RESULT_CANCELED);
                            getActivity().finish();
                    }
            });

            // Until the user has taken a photo, hide the preview
            productPreview = (ParseImageView) v
                            .findViewById(R.id.product_preview_image);
            productPreview.setVisibility(View.INVISIBLE);

            return v;
    }

    public void setCurrentLocation(ParseGeoPoint point) {

            currentLocation = point;
    }

    public void startCamera() {
```

```
        Fragment cameraFragment = new AppCamera();
        FragmentTransaction transaction = getActivity().getFragmentManager()
                        .beginTransaction();
        transaction.replace(R.id.container, cameraFragment);
        transaction.addToBackStack("ProductFragment");
        transaction.commit();
}

/*
 * On resume, check and see if a meal photo has been set from the
 * CameraFragment. If it has, load the image in this fragment and make the
 * preview image visible.
 */
// @Override
/*
 * public void onResume() { super.onResume(); ParseFile photoFile =
 * ((NewProductActivity) getActivity()) .getCurrentProduct().getPhoto(); if
 * (photoFile != null) { // productPreview.setParseFile(photoFile); //
 * productPreview.loadInBackground(new GetDataCallback() {
 *
 * @Override public void done(byte[] data, ParseException e) { //
 * productPreview.setVisibility(View.VISIBLE); } }); } }
 */
public void findGpsThread(final File dir, final String defaultPerm,
                final boolean value) {
        // show progress
        final ProgressDialog progress = ProgressDialog.show(getActivity(), "",
                        "Finding GPS Location...");
        progress.show();
        Runnable myRun = new Runnable() {
                @Override
                public void run() {
                        Looper.myLooper();
                        Looper.prepare();

                        // Finally
```

```
                        getActivity().runOnUiThread(new Runnable() {
                                @Override
                                public void run() {
                                        progress.dismiss();
                                        // bring back default perm
                                        if (value == true) {

                                        }

                                }
                        });
                }
        };
        Thread T = new Thread(myRun);
        T.start();
    }
}
```

3.12 Profile.java

```
package ie.itcarlow.sra;

import android.os.Bundle;

public class Profile extends DrawerActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        setContentView(R.layout.activity_profile);
        super.onCreate(savedInstanceState);

    }

}
```

3.13 Requests.java

```
package ie.itcarlow.sra;

import android.os.Bundle;

public class Requests extends DrawerActivity {

	@Override
	protected void onCreate(Bundle savedInstanceState) {

		setContentView(R.layout.activity_requests);
		super.onCreate(savedInstanceState);

	}

}
```

3.14 SearchActivity.java

```
package ie.itcarlow.sra;

import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;

public class SearchActivity extends DrawerActivity {
	private EditText whatToSearch, whereToSearch;
	private String whatToSearchParam, whereToSearchParam;
	private Button searchButton;

	@Override
	protected void onCreate(Bundle savedInstanceState) {

		setContentView(R.layout.activity_search);
```

```
            super.onCreate(savedInstanceState);

            whatToSearch = (EditText) findViewById(R.id.search_what_input);
            whereToSearch = (EditText) findViewById(R.id.search_where_input);
            searchButton = (Button) findViewById(R.id.search);

            searchButton.setOnClickListener(new OnClickListener() {

                @Override
                public void onClick(View arg0) {
                    // TODO Auto-generated method stub

                    findProduct(whatToSearchParam, whereToSearchParam);
                }
            });
        }

        public void findProduct(String whatToSearch, String whereToSearch) {

        }
}
```

3.15 ShoppingList.java

```
package ie.itcarlow.sra;

import ie.itcarlow.sra.entity.ShoppingListObject;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
```

```
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Toast;

import com.parse.ParseQueryAdapter;
import com.parse.ParseUser;

public class ShoppingList extends DrawerActivity {

	private ParseQueryAdapter<ShoppingListObject> shoppingListAdapter;
	ListView listview;

	@Override
	protected void onCreate(Bundle savedInstanceState) {

		setContentView(R.layout.activity_shopping_list);
		super.onCreate(savedInstanceState);

		shoppingListAdapter = new ParseQueryAdapter<ShoppingListObject>(this,
				ShoppingListObject.class);
		shoppingListAdapter.setTextKey("listName");

		listview = (ListView) findViewById(R.id.shopping_list);
		listview.setAdapter(shoppingListAdapter);

	}

	@Override
	public boolean onCreateOptionsMenu(Menu menu) {
		MenuInflater inflater = getMenuInflater();
		inflater.inflate(R.menu.shopping_list, menu);
		return true;
	}

	@Override
```

```
public boolean onOptionsItemSelected(MenuItem item) {
	switch (item.getItemId()) {
	case R.id.add_shoppinglist:
		inputAlertDialog();
		return true;
	case R.id.refreshList:
		refreshList();
		return true;
	case R.id.remove_shoppinglist:

	default:
		return super.onOptionsItemSelected(item);
	}
}

public void inputAlertDialog() {
	AlertDialog.Builder alert = new AlertDialog.Builder(this);
	alert.setTitle("New Shopping List").setMessage("Name of Shopping List");

	final EditText shoppingListNameInput = new EditText(this);
	// final EditText ListDescriptionInput = new EditText(this);

	alert.setView(shoppingListNameInput);
	// alert.setView(ListDescriptionInput);

	alert.setPositiveButton("OK", new DialogInterface.OnClickListener() {

		@Override
		public void onClick(DialogInterface dialog, int which) {
			// TODO Auto-generated method stub
			String shoppingListName = shoppingListNameInput.getText()
					.toString();
			// String shoppingListDescription =
			// ListDescriptionInput.getText().toString();

			if (shoppingListName.length() > 0) {
```

```
                                ShoppingListObject shoppingList = new ShoppingListObject();
                                shoppingList.setListName(shoppingListName);
                                shoppingList.setUserId(ParseUser.getCurrentUser()
                                                .getObjectId().toString());
                                // shoppingList.setListDescription(shoppingListDescription);
                                shoppingList.saveInBackground();
                        }

                }
        });
        alert.setNegativeButton("Cancel",
                        new DialogInterface.OnClickListener() {
                                @Override
                                public void onClick(DialogInterface dialog, int whichButton) {
                                        // User has Cancelled the Operation.
                                }
                        });

        alert.show();

}

@Override
public void onResume() {
        super.onResume();

        shoppingListAdapter.loadObjects();
        listview.setAdapter(shoppingListAdapter);
}

public void refreshList() {
        shoppingListAdapter.loadObjects();
        listview.setAdapter(shoppingListAdapter);
}

public void removeList() {
```

```
                listview.setOnItemSelectedListener(new OnItemSelectedListener() {

                        @Override
                        public void onItemSelected(AdapterView<?> arg0, View arg1,
                                        int position, long arg3) {
                                // TODO Auto-generated method stub
                                Toast.makeText(
                                                getApplicationContext(),
                                                (CharSequence) shoppingListAdapter
                                                                .getItem((Integer) arg0.getSelectedItem()),
                                                Toast.LENGTH_LONG).show();
                        }

                        @Override
                        public void onNothingSelected(AdapterView<?> arg0) {
                                // TODO Auto-generated method stub

                        }
                });
        }
}
```

3.16 Signup.java

```
package ie.itcarlow.sra;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
```

```
import com.parse.ParseException;
import com.parse.ParseUser;
import com.parse.SignUpCallback;

public class Signup extends Activity {
	String usernameParam, passwordParam, firstnameParam, lastnameParam,
			userEmailParam;
	Boolean mstatus = true;
	Integer userid;
	Button signupButton;
	EditText usernameEditText, passwordEditText, passConfirmText,
			firstnameEditText, lastnameEditText, userEmailEditText;

	// These will be used for Object fields.
	final String FIRSTNAME = "firstName";
	final String LASTNAME = "lastName";
	final Boolean MSTATUS = true;

	@Override
	protected void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.activity_signup);

		signupButton = (Button) findViewById(R.id.button_signup);

		signupButton.setOnClickListener(new View.OnClickListener() {

			@Override
			public void onClick(View v) {
				// TODO Auto-generated method stub

				usernameEditText = (EditText) findViewById(R.id.signup_username);
				Validate.hasText(usernameEditText);
				usernameParam = usernameEditText.getText().toString();

				passwordEditText = (EditText) findViewById(R.id.signup_password);
```

```
passConfirmText = (EditText) findViewById(R.id.signup_passconfirm);
Validate.hasText(passwordEditText);
Validate.isPasswordSame(passwordEditText, passConfirmText);
passwordParam = passwordEditText.getText().toString();

firstnameEditText = (EditText) findViewById(R.id.signup_firstname);
Validate.hasText(firstnameEditText);
firstnameParam = firstnameEditText.getText().toString();

lastnameEditText = (EditText) findViewById(R.id.signup_lastname);
Validate.hasText(lastnameEditText);
lastnameParam = lastnameEditText.getText().toString();

userEmailEditText = (EditText) findViewById(R.id.signup_email);
Validate.isEmailAddress(userEmailEditText, true);
userEmailParam = userEmailEditText.getText().toString();

ParseUser newUser = new ParseUser();

newUser.setUsername(usernameParam);
newUser.put(FIRSTNAME, firstnameParam);
newUser.put(LASTNAME, lastnameParam);
newUser.setPassword(passwordParam);
newUser.setEmail(userEmailParam);
newUser.put("mStatus", true);
newUser.signUpInBackground(new SignUpCallback() {
       @Override
       public void done(ParseException e) {
              if (e == null) {
                     Toast.makeText(getApplicationContext(),
                                   "Signup Successful", Toast.LENGTH_LONG)
                                   .show();
                     goToMainActivity();
              } else {

                     Toast.makeText(getApplicationContext(),
```

```
                                                    e.getMessage(), Toast.LENGTH_LONG).show();
                                            }
                                        }
                                    });
                                }

                            });

        }

        @Override
        public boolean onCreateOptionsMenu(Menu menu) {
                // Inflate the menu; this adds items to the action bar if it is present.
                getMenuInflater().inflate(R.menu.signup, menu);
                return true;
        }

        public void goToMainActivity() {
                Intent main = new Intent(getApplicationContext(), MainActivity.class);
                startActivity(main);
                finish();
        }
}
```

4 Supplementary Classes

4.1 App.java

```
package ie.itcarlow.sra;

import ie.itcarlow.sra.entity.Comment;
```

```
import ie.itcarlow.sra.entity.Product;
import ie.itcarlow.sra.entity.ProductRequest;
import ie.itcarlow.sra.entity.Rating;
import ie.itcarlow.sra.entity.Recommendation;
import ie.itcarlow.sra.entity.ShoppingListItem;
import ie.itcarlow.sra.entity.ShoppingListObject;
import android.app.Application;

import com.parse.Parse;
import com.parse.ParseObject;

public class App extends Application {

	@Override
	public void onCreate() {

		super.onCreate();
		// Initialize the Parse.com Keys
		Parse.initialize(this, "ATvRI18FPtvTj57nsbK9CyOCfv1gJAqbPGz18dK0",
				"LKis4dFnLbuCrditS37pRvY3C29IsYBdcSymUU4S");
		// Register the Custom Classes to be used within the Application
		ParseObject.registerSubclass(ShoppingListObject.class);
		ParseObject.registerSubclass(ShoppingListItem.class);
		ParseObject.registerSubclass(Product.class);
		ParseObject.registerSubclass(Comment.class);
		ParseObject.registerSubclass(Rating.class);
		ParseObject.registerSubclass(ProductRequest.class);
		ParseObject.registerSubclass(Recommendation.class);
	}
}
```

4.2 Validate.java

```
package ie.itcarlow.sra;

import java.util.regex.Pattern;
```

```
import android.widget.EditText;

public class Validate {
	// Regex
	private static final String EMAIL_REGEX = "^[_A-Za-z0-9-\\+]+(\\.[_A-Za-z0-9-]+)*@[A-Za-z0-9-]+(\\.[A-Za-z0-
9]+)*(\\.[A-Za-z]{2,})$";

	private static final String REQUIRED_MSG = "Field is Required";
	private static final String EMAIL_MSG = "Invalid Email Address";
	private static final String NO_MATCH = "Passwords do not match";

	public static boolean isEmailAddress(EditText editText, boolean required) {
		return isValid(editText, EMAIL_REGEX, EMAIL_MSG, required);
	}

	public static boolean isValid(EditText editText, String regex,
			String errMsg, boolean required) {

		String text = editText.getText().toString().trim();

		editText.setError(null);

		if (required && !hasText(editText))
			return false;

		if (required && !Pattern.matches(regex, text)) {
			editText.setError(errMsg);
			return false;
		}
		;

		return true;
	}

	public static boolean hasText(EditText editText) {

		String text = editText.getText().toString().trim();
		editText.setError(null);
```

```
                if (text.length() == 0) {
                        editText.setError(REQUIRED_MSG);
                        return false;
                }

                return true;
        }

        public static boolean isPasswordSame(EditText editText, EditText editText2) {
                String text = editText.getText().toString().trim();
                editText.setError(null);
                String text2 = editText2.getText().toString().trim();
                editText2.setError(null);

                if (text.equals(text2) == false) {
                        editText2.setError(NO_MATCH);
                        return false;
                }
                return true;
        }
}
```