

```
//author(s): Alex Berliner
package com.cyf.challengeyourfriends;

import java.util.ArrayList;

import twitter4j.TwitterException;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.os.AsyncTask;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import com.hintdesk.core.util.StringUtil;

public class ChallengeSingleView extends Activity implements OnClickListener,
    OnItemSelectedListener {
    private TextView TVName;
    private ListView LVRules;
    private TextView TVDaysLeft;
    private Button completeButton;
    private ChallengeManager cm;
    private FriendManager friendsList;
    private Challenge targetChallenge;
    private TextView messagePreview;
    private EditText sendToFriends;

    private final Context theC = this;
    private final ConstantValues theValues = new ConstantValues(theC);
    private Spinner friendSpinner;
    private boolean startup = true;
    private ArrayList<String> spinnerList;
    private EditText linkText;
    private String stinklink;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_challenge_single_view);
        Bundle extras = getIntent().getExtras();
        String chalName=null;
        if (extras != null) {
            chalName = extras.getString("touse");
        }
    }
}
```

```
    }
    friendsList = new FriendManager(this);
    cm = new ChallengeManager(this, false);
    targetChallenge = cm.getChallengeNamed(chalName);
    TVName = (TextView) findViewById(R.id.viewchallenge_name);
    TVName.setText("Challenge name: " + targetChallenge.name);
    LVRules = (ListView) findViewById(R.id.viewchallenge_lvrules);

    //populate friendslist
    ArrayAdapter<String> av = new ArrayAdapter<String>(this, android.R.layout
        .simple_list_item_1, targetChallenge.rules);
    ListView friendsListView = (ListView) findViewById(R.id.
        viewchallenge_lvrules);
    friendsListView.setAdapter(av);

    TVDaysLeft = (TextView) findViewById(R.id.viewchallenge_completedays);
    TVDaysLeft.setText("Days to complete challenge: " + targetChallenge.
        completeDays);

    completeButton = (Button) findViewById(R.id.
        viewchallenge_btncompletechallenge);
    completeButton.setOnClickListener(this);

    friendSpinner = (Spinner) findViewById(R.id.viewchallenge_friendspinner);
    // Create an ArrayAdapter using the string array and a default spinner
    layout

    spinnerList = (ArrayList<String>) friendsList.getFriendsList().clone();
    spinnerList.add(0, "");
    ArrayAdapter<String> friendAdapter = new ArrayAdapter<String>(this,
        android.R.layout.simple_list_item_1, spinnerList);

    friendAdapter.setDropDownViewResource(android.R.layout.
        simple_spinner_dropdown_item);

    friendSpinner.setAdapter(friendAdapter);
    friendSpinner.setOnItemSelectedListener(this);
    sendToFriends = (EditText) findViewById(R.id.viewchallenge_allfriends);
    linkText = (EditText) findViewById(R.id.viewchallenge_edittextlink);
}

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

@Override
public void onClick(View v) {
    int viewID = v.getId();
    switch(viewID){
    case R.id.viewchallenge_btncompletechallenge:
        //when sending challenge
        if(sendToFriends.getText().length() < 2){
            Toast.makeText(getApplicationContext(),"Please select friends to
            send this challenge to.", Toast.LENGTH_SHORT).show();
            return;
        }
        //passed checks so send challenge
        stinklink = linkText.getText()+"";
        if(stinklink==null || stinklink.length()<2 )
            stinklink="";
        AlertDialog.Builder alert2 = new AlertDialog.Builder(this);

        alert2.setTitle("Challenge Preview");

        // Set an EditText view to get user input
        messagePreview = new TextView(this);
        messagePreview.setText("Challenge preview: <name>"+targetChallenge.
            challengeTweet(stinklink));
        alert2.setView(messagePreview);

        alert2.setPositiveButton("Confirm",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int
                    whichButton) {

                    for(String str : (sendToFriends.getText()
                        + "").split("\n")){
                        final String status = "@"+str+targetChallenge.
                            challengeTweet(stinklink);
                        //open twitter stream
                        if (!StringUtil.isNullOrEmpty(status)) {
                            new TwitterUpdateStatusTask().execute(status)
                                ;
                        }
                    }
                }
            });

        alert2.setNegativeButton("Cancel",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,
                    int whichButton) {
                    // Canceled.
                }
            });

        alert2.show();

```

```
        break;
    }
}

class TwitterUpdateStatusTask extends AsyncTask<String, String, Boolean> {
    //threaded twitter interfacing object
    @Override
    protected void onPostExecute(Boolean result) {
        if (result) {
            Toast.makeText(theC, "Challenge Sent", Toast.LENGTH_SHORT).show()
                ;
        }
        else {
            Toast.makeText(theC, "Failed to send challenge", Toast.
                LENGTH_SHORT).show();
        }
    }

    @Override
    protected Boolean doInBackground(String... params) {
        try {
            theValues.getTwitter().updateStatus(params[0]);
            return true;
        }
        catch (TwitterException e) {
            e.printStackTrace();
            return false;
        }
    }
}

public void log(String str){
    Log.w("chalsing",str);
}

@Override
public void onItemClick(AdapterView<?> parent, View view, int position,
    long id) {
    if(startup){
        startup = false;
        return;
    }
    if(position == 0)
        return;
    if(sendToFriends.getText().length() < 1)
        sendToFriends.setText(spinnerList.get(position));
    else
        sendToFriends.setText(sendToFriends.getText()+"\n"+spinnerList.get
            (position));
}

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

```
}
```