

```
*****
* Competitors: Katharine Nelson and Suraj Masand *
* School: Alpharetta High School in Alpharetta, GA *
* State: Georgia *
* Event: Mobile Application Development *
* 2015 FBLA National Leadership Conference *
* *
* Code PDF for Kalendo *
*****
```

FRONT-END & USER INTERACTIONS CLASSES

```
***** MainActivity.java *****
```

[illegible]

```

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

    //Sets up banner image scroller
    initializeBannerScroller();

    //Displays today's date
    initializeDateText();

    //Checks network before accessing the calendar
    checkInternet();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    //Creates the menu for app info
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    //Opens activity to display app info
    switch (item.getItemId()) {
        case R.id.app_info:
            Intent i = new Intent(this, InfoActivity.class);
            startActivity(i);
            break;
    }
    return super.onOptionsItemSelected(item);
}

public void initializeDateText() {
    //Formats text for the current date
    Date date = new Date();
    todayEvents = (TextView) findViewById(R.id.today_view);
    todayEvents.setGravity(Gravity.CENTER);
    todayEvents.setTextColor(Color.BLACK);
    todayEvents.setTextSize(28);
    todayEvents.setTypeface(null, Typeface.BOLD);
    todayEvents.setText(DateFormat.format("MMMM d, yyyy ", date)+"\n");
}

public void checkInternet() {
    //Internet is required to load calendar data
    if (CheckNetwork.isInternetAvailable(getApplicationContext())) {
        //Okay to proceed with app processes
        hasInternet = true;
        todayEvents.append("Today's Events:\n");
    }
}

```

```

        initializeTodayEvents();
    }
    else {
        hasInternet = false;
        //Cannot do required processes without connection
        todayEvents.append("\nThere is no Internet connection.\n" +
                           "Please connect and restart the app.");
    }
}

public void initializeTodayEvents() {

    //Events will be added to this layout
    LinearLayout todayLayout = (LinearLayout) findViewById(R.id.today_layout);

    String calURL = BuildURL.todayURL();
    RequestCal googleCal = (RequestCal) new RequestCal().execute(calURL);

    try {
        //Saves the returned object as a JSON Object
        JSONObject jsonObj = new JSONObject(googleCal.get());

        ArrayList<EventInfo> allEvents = EventList.analyzeData(jsonObj);

        if(allEvents.size() == 0) {
            TextView noEvents = new TextView(this);
            EventText.noEvents(noEvents, getString(R.string.no_today));
            todayLayout.addView(noEvents);
            return;
        }

        //Loops through events list to analyze them individually
        for (EventInfo indexEvent : allEvents) {

            //Event name
            TextView eventHeader = new TextView(this);
            EventText.header(eventHeader, indexEvent);
            todayLayout.addView(eventHeader);

            //Event time
            TextView eventTime = new TextView(this);
            EventText.time(eventTime, indexEvent);
            todayLayout.addView(eventTime);

            //Event description
            TextView eventDesc = new TextView(this);
            EventText.descrip(eventDesc, indexEvent);
            todayLayout.addView(eventDesc);

            //Skips line between events
            TextView spacer = new TextView(this);
            EventText.spacer(spacer);
            todayLayout.addView(spacer);
        }

    } catch (JSONException e) {

```

```

        Log.e("JSONException", "Error: " + e.toString());
    } catch (ExecutionException e) {
        Log.e("ExecutionException", "Error: " + e.toString());
    } catch (InterruptedException e) {
        Log.e("ExecutionException", "Error: " + e.toString());
    }
}

```

```

public void initializeBannerScroller() {
    //Banner image scroller setup
    horizScroll = (HorizontalScrollView) findViewById(R.id.horiz_scroll);
    horizLayout = (LinearLayout) findViewById(R.id.horiz_layout);
    horizScroll.setHorizontalScrollBarEnabled(false);
    addImagesToView();
    ViewTreeObserver vto = horizLayout.getViewTreeObserver();
    vto.addOnGlobalLayoutListener(new OnGlobalLayoutListener() {
        @Override
        public void onGlobalLayout() {
            resetScroll();
        }
    });
    horizScroll.fullScroll(View.FOCUS_LEFT);
}

```

```

public void addImagesToView() {
    //Adds images to the banner scroller
    //Images are of Alpharetta High School students at various events
    for (int i = 0; i < imageNameArray.length; i++) {
        final Button imageButton = new Button(this);
        int imageResourceId = getResources().getIdentifier(imageNameArray[i],
                                                            "drawable", getPackageName());
        Drawable image = this.getResources().getDrawable(imageResourceId);
        imageButton.setBackground(image);
        imageButton.setTag(i);
        int height = getScreenHeight()/5;
        LinearLayout.LayoutParams params = new LinearLayout.LayoutParams(height,
                                                                              height);

        params.setMargins(1, 0, 1, 5);
        imageButton.setLayoutParams(params);
        horizLayout.addView(imageButton);
    }
}

```

```

public void startAutoScrolling() {
    //Starts the banner scroller and sets speed
    if (scrollTimer == null) {
        scrollTimer = new Timer();
        final Runnable Timer_Tick = new Runnable() {
            public void run() {
                moveScrollView();
            }
        };
        scrollerSchedule = new TimerTask() {

```

```

        @Override
        public void run() {
            runOnUiThread(Timer_Tick);
        }
    };
    scrollTimer.schedule(scrollerSchedule, 30, 30);
}

public void moveScrollView() {
    scrollPos = (horizScroll.getScrollX() + 1);
    if (scrollPos >= scrollMax) {
        //Sets the position back to zero
        scrollPos = 0;
        horizScroll.setScrollX(0);
        resetScroll();
    }
    else {
        //Continues with scrolling
        horizScroll.smoothScrollTo(scrollPos, 0);
    }
}

public void resetScroll() {
    //Get max distance and restart scrolling
    getScrollMaxAmount();
    startAutoScrolling();
}

public void getScrollMaxAmount() {
    //Determines width of the array of images
    int width = ((horizLayout.getWidth()/horizScroll.getWidth()-1)*horizScroll.
        getWidth()) + (horizLayout.getWidth() % horizScroll.getWidth());
    scrollMax = width;
}

private int getScreenHeight() {
    //Gets the height of the screen for formatting page elements
    return getResources().getDisplayMetrics().heightPixels;
}

public void toCal(View view) {
    //Opens the calendar view
    if (hasInternet) {
        Intent i = new Intent(this, CalActivity.class);
        startActivity(i);
    }
    else {
        //No Internet disables app functions
        Toast.makeText(this, "No Internet Connection", Toast.LENGTH_LONG).show();
    }
}

```

```
public void toList(View view) {
    //Opens the list view
    if (hasInternet) {
        Intent i = new Intent(this, ListActivity.class);
        startActivity(i);
    }
    else {
        //No Internet disables app functions
        Toast.makeText(this, "No Internet Connection", Toast.LENGTH_LONG).show();
    }
}
```

```
public void toTwitterFeed(View view) {
    //Opens the twitter view
    if (hasInternet) {
        Intent i = new Intent(this, TwitterFeed.class);
        startActivity(i);
    }
    else {
        //No Internet disables app functions
        Toast.makeText(this, "No Internet Connection", Toast.LENGTH_LONG).show();
    }
}
```

```
}
```

```
*****
```

***** CalActivity.java *****

```
package com.alpharettafblla.kalendo;

import java.util.Calendar;

import android.app.Activity;
import android.content.Intent;
import android.graphics.Color;
import android.graphics.Typeface;
import android.os.Bundle;
import android.text.format.DateFormat;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.CalendarView;
import android.widget.CalendarView.OnDateChangeListener;
import android.widget.TextView;
import com.alpharettafblla.kalendo.R;

public class CalActivity extends Activity {

    CalendarView calendar;
    TextView selectedDate;

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

        //Sets up the calendar
        initializeCalendar();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        //Creates the menu for app info
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        //Opens activity to display app info
        switch (item.getItemId()) {
            case R.id.app_info:
                Intent i = new Intent(this, InfoActivity.class);
                startActivity(i);
                break;
        }
        return super.onOptionsItemSelected(item);
    }
}
```



```

public void initializeCalendar() {
    //Prepares the calendar for viewing
    calendar = (CalendarView) findViewById(R.id.calendar_view);
    Calendar today = Calendar.getInstance();
    calendar.setDate(today.getTime().getTime(), false, true);

    //Whether to show the week number
    calendar.setShowWeekNumber(false);

    //Sets the first day of the week according to Calendar
    calendar.setFirstDayOfWeek(1);

    //Initializes the date selected on the calendar
    selectedDate = (TextView) findViewById(R.id.selected_date);
    selectedDate.setTextColor(Color.BLACK);
    selectedDate.setTypeface(null, Typeface.BOLD);
    selectedDate.setText(DateFormat.format("MMMM d, yyyy ", calendar.getDate()));

    //Sets the listener to be notified upon selected date change
    calendar.setOnDateChangeListener(new OnDateChangeListener() {
        @Override
        public void onSelectedDayChange(CalendarView view, int year,
            int month, int day) {
            selectedDate.setText(DateFormat.format("MMMM d, yyyy ",
                calendar.getDate()));
        }
    });
}

public void toDayEvents(View view) {
    //Opens up the specific set of events for the selected date
    Calendar selectDate = Calendar.getInstance();
    selectDate.setTimeInMillis(calendar.getDate());

    //Sends data for the selected date
    Intent i = new Intent(this, DayEvents.class);
    i.putExtra("CalendarDate", selectDate.getTimeInMillis());
    startActivity(i);
}

public void toList(View view) {
    //Opens the list view
    Intent i = new Intent(this, ListActivity.class);
    startActivity(i);
    finish();
}

public void toHome(View view) {
    //Returns to the home page
    Intent i = new Intent(this, MainActivity.class);
    startActivity(i);
    finish();
}

```

```
public void toTwitterFeed(View view) {  
    //Opens the twitter view  
    Intent i = new Intent(this, TwitterFeed.class);  
    startActivity(i);  
    finish();  
}  
  
}
```

***** DayEvents.java *****

```
package com.alpharettafblla.kalendo;

import java.util.ArrayList;
import java.util.Calendar;
import java.util.concurrent.ExecutionException;

import org.json.JSONException;
import org.json.JSONObject;

import android.app.Activity;
import android.content.Intent;
import android.graphics.Color;
import android.graphics.Typeface;
import android.os.Bundle;
import android.text.format.DateFormat;
import android.text.method.ScrollingMovementMethod;
import android.util.Log;
import android.view.Gravity;
import android.view.View;
import android.widget.ImageButton;
import android.widget.LinearLayout;
import android.widget.TextView;
import com.alpharettafblla.kalendo.R;

public class DayEvents extends Activity {

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

        //Collects the selected date's information
        Intent intent = getIntent();
        getDayEvents(intent);

        //Formats back button
        initializeBackBtn();
    }

    public void getDayEvents(Intent intent) {

        //Retrieves the selected date as a millisecond value
        long cal = intent.getLongExtra("CalendarDate",
            Calendar.getInstance().getTimeInMillis());
        Calendar headerDate = Calendar.getInstance();
        headerDate.setTimeInMillis(cal);

        //Displays selected date
        initializeDateText(cal, headerDate);

        //Events will be added to this layout
        LinearLayout eventsView = (LinearLayout) findViewById(R.id.event_view);
```

```

String calURL = BuildURL.dateURL(cal);
RequestCal googleCal = (RequestCal) new RequestCal().execute(calURL);

try {
    //Saves the returned object as a JSON Object
    JSONObject jsonObj = new JSONObject(googleCal.get());

    ArrayList<EventInfo> allEvents = EventList.analyzeData(jsonObj);

    if (allEvents.size() == 0) {
        TextView noEvents = new TextView(this);
        EventText.noEvents(noEvents, getString(R.string.no_selected));
        eventsView.addView(noEvents);
        return;
    }

    //Loops through events list to analyze them individually
    for (EventInfo indexEvent : allEvents) {

        //Event name
        TextView eventHeader = new TextView(this);
        EventText.header(eventHeader, indexEvent);
        eventsView.addView(eventHeader);

        //Event time
        TextView eventTime = new TextView(this);
        EventText.time(eventTime, indexEvent);
        eventsView.addView(eventTime);

        //Event description
        TextView eventDesc = new TextView(this);
        EventText.descrip(eventDesc, indexEvent);
        eventsView.addView(eventDesc);

        //Skips line between events
        TextView spacer = new TextView(this);
        EventText.spacer(new TextView(this));
        eventsView.addView(spacer);
    }

} catch (JSONException e) {
    Log.e("JSONException", "Error: " + e.toString());
} catch (ExecutionException e) {
    Log.e("ExecutionException", "Error: " + e.toString());
} catch (InterruptedException e) {
    Log.e("ExecutionException", "Error: " + e.toString());
}
}

public void initializeDateText(long cal, Calendar headerDate) {
    //Displays the selected date
    TextView eventText = (TextView) findViewById(R.id.event_text);
    eventText.setMovementMethod(new ScrollingMovementMethod());
    TextView dateView = (TextView) findViewById(R.id.date_view);
    dateView.setGravity(Gravity.CENTER_HORIZONTAL);
}

```

```
        dateView.setTextColor(Color.BLACK);
        dateView.setTypeface(null, Typeface.BOLD);
        dateView.setText(DateFormat.format("MMMM d, yyyy ", headerDate));
    }
```

```
public void initializeBackBtn() {
    //Resizes the back button
    ImageButton backBtn = (ImageButton) findViewById(R.id.back_btn);
    int btnHeight = getScreenHeight()/8;
    backBtn.setMinimumHeight(0);
    backBtn.setMinimumWidth(0);
    backBtn.setMaxHeight(btnHeight);
    backBtn.setMaxWidth(btnHeight);
}
```

```
public int getScreenHeight() {
    //Gets the height of the screen for formatting page elements
    return getResources().getDisplayMetrics().heightPixels;
}
```

```
public void backToCal(View view) {
    //Closes the event view and returns to calendar
    finish();
}
```

```
}
```

```
*****
```

***** ListActivity.java *****

```
package com.alpharettafblla.kalendo;

import java.util.ArrayList;
import java.util.concurrent.ExecutionException;

import org.json.JSONException;
import org.json.JSONObject;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.LinearLayout;
import android.widget.TextView;
import com.alpharettafblla.kalendo.R;

public class ListActivity extends Activity {

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

        //Retrieve events up to 3 months in advance
        initializeEventsList();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        //Creates the menu for app info
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        //Opens activity to display app info
        switch (item.getItemId()) {
            case R.id.app_info:
                Intent i = new Intent(this, InfoActivity.class);
                startActivity(i);
                break;
        }
        return super.onOptionsItemSelected(item);
    }

    public void initializeEventsList() {
```

```

//Events will be added to this layout
LinearLayout listLayout = (LinearLayout) findViewById(R.id.list_events);
listLayout.setPadding(0, 18, 0, 0);

String calURL = BuildURL.listURL();
RequestCal googleCal = (RequestCal) new RequestCal().execute(calURL);

try {
    //Saves the returned object as a JSON Object
    JSONObject jsonObj = new JSONObject(googleCal.get());

    ArrayList<EventInfo> allEvents = EventList.analyzeData(jsonObj);

    if(allEvents.size() == 0) {
        TextView noEvents = new TextView(this);
        EventText.noEvents(noEvents, getString(R.string.no_upcoming));
        listLayout.addView(noEvents);
        return;
    }

    //Loops through events list to analyze them individually
    for (EventInfo indexEvent : allEvents) {

        //Event name
        TextView eventHeader = new TextView(this);
        EventText.header(eventHeader, indexEvent);
        listLayout.addView(eventHeader);

        //Event time
        TextView eventTime = new TextView(this);
        EventText.dateTime(eventTime, indexEvent);
        listLayout.addView(eventTime);

        //Event description
        TextView eventDesc = new TextView(this);
        EventText.descrip(eventDesc, indexEvent);
        listLayout.addView(eventDesc);

        //Skips line between events
        TextView spacer = new TextView(this);
        EventText.spacer(spacer);
        listLayout.addView(spacer);
    }

} catch (JSONException e) {
    Log.e("JSONException", "Error: " + e.toString());
} // catch (JSONException e)
catch (ExecutionException e) {
    Log.e("ExecutionException", "Error: " + e.toString());
} catch (InterruptedException e) {
    Log.e("ExecutionException", "Error: " + e.toString());
}
}

public void toHome(View view) {

```

```
        //Returns to the home page
        Intent i = new Intent(this, MainActivity.class);
        startActivity(i);
        finish();
    }
```

```
public void toTwitterFeed(View view) {
    //Opens the twitter view
    Intent i = new Intent(this, TwitterFeed.class);
    startActivity(i);
    finish();
}
```

```
public void toCal(View view) {
    //Opens the calendar view
    Intent i = new Intent(this, CalActivity.class);
    startActivity(i);
    finish();
}
```

```
}
```

```
*****
```


***** InfoActivity.java *****

```
package com.alpharettafblla.kalendo;
```

```
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;
import android.widget.TextView;
import com.alpharettafblla.kalendo.R;
```

```
public class InfoActivity extends Activity {
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_info);
```

```
        //Formats the back button
        initializeBackBtn();
```

```
        //Displays info text
        infoText();
    }
```

```
    public void infoText() {
        //Displays info text
        TextView info = (TextView) findViewById(R.id.info_text);
        info.setText("Developed for the 2015 FBLA National Leadership Conference\n\n");
        info.append("Created by: Katharine Nelson and Suraj Masand\n");
        info.append("School: Alpharetta High School\n");
        info.append("Event: Mobile Application Development\n\n");
        info.append("This app was developed to help Alpharetta High School announce " +
            "upcoming events to the students. It connects to the school's Google " +
            "calendar and displays the events in three formats: a today view " +
            "showing events for the current day, a list view showing events for " +
            "the next 90 days, and a calendar allowing students to select any date " +
            "and view its events.");
    }
```

```
    public void initializeBackBtn() {
        //Resizes the back button
        ImageButton backBtn = (ImageButton) findViewById(R.id.back_btn);
        int btnHeight = getScreenHeight()/8;
        backBtn.setMinimumHeight(0);
        backBtn.setMinimumWidth(0);
        backBtn.setMaxHeight(btnHeight);
        backBtn.setMaxWidth(btnHeight);
    }
```

```
    public int getScreenHeight() {
        //Gets the height of the screen for formatting page elements
```

```
        return getResources().getDisplayMetrics().heightPixels;
    }
```

```
    public void goBack(View view) {
        //Closes the event view and returns to the last activity
        finish();
    }
```

```
}
```

```
*****
```

***** TwitterFeed.java *****

```
package com.alpharettafblla.kalendo;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.MotionEvent;
import android.view.View;
import android.view.View.OnClickListener;
import android.webkit.WebView;
import android.widget.Button;
import android.widget.Spinner;
import com.alpharettafblla.kalendo.R;

public class TwitterFeed extends Activity {

    public static final String TAG = "TwitterFeed";
    private static final String baseUrl = "https://twitter.com";
    private static String twitterAcc = "AHSRaiderFBLA";
    private Spinner spinner1;

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

        //Sets up the twitter feed drop-down menu
        addListenerOnButton();

        //Sets spinner to the saved state
        spinnerSavedState();

        //Displays the twitter feed
        initializeTwitter();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        //Creates the menu for app info
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        //Opens activity to display app info
        switch (item.getItemId()) {
            case R.id.app_info:
                Intent i = new Intent(this, InfoActivity.class);
                startActivity(i);
                break;
        }
    }
}
```

```

    }
    return super.onOptionsItemSelected(item);
}

public void addListenerOnButton() {
    spinner1 = (Spinner) findViewById(R.id.spinner1);
    Button submitBtn = (Button) findViewById(R.id.submit_btn);

    //Waits for new twitter feed selection
    submitBtn.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v) {
            WebView webView = (WebView) findViewById(R.id.timeline_webview);
            twitterAcc = String.valueOf(spinner1.getSelectedItem());
            webView.loadDataWithBaseURL(baseUrl, BuildURL.twitterURL(twitterAcc),
                "text/html", "UTF-8", null);
        }
    });
}

public void spinnerSavedState() {
    //Sets spinner to the saved value
    spinner1 = (Spinner) findViewById(R.id.spinner1);
    if (twitterAcc.equals("AHSRaiderFBLA"))
        spinner1.setSelection(0);
    else if (twitterAcc.equals("FultonCoSchools"))
        spinner1.setSelection(1);
    else
        spinner1.setSelection(2);
}

public void initializeTwitter() {
    //Must load background color for webView
    loadBackgroundColor();

    WebView webView = (WebView) findViewById(R.id.timeline_webview);
    webView.getSettings().setDomStorageEnabled(true);
    webView.getSettings().setJavaScriptEnabled(true);

    //Forces webView to be scrolling but not clickable
    //No links can be chosen but more tweets can be loaded
    webView.setLongClickable(false);
    webView.setOnTouchListener(new View.OnTouchListener() {
        @Override
        public boolean onTouch(View arg0, MotionEvent arg1)
        {
            return arg1.getAction() == MotionEvent.ACTION_UP;
        }
    });

    //Loads twitter feed
    webView.loadDataWithBaseURL(baseUrl, BuildURL.twitterURL(twitterAcc), "text/html",
        "UTF-8", null);
}

```

```
private void loadBackgroundColor() {  
    WebView webView = (WebView) findViewById(R.id.timeline_webview);  
    //Sets webView to be transparent  
    webView.setBackgroundColor(0);  
}
```

```
public void toList(View view) {  
    //Opens the list view  
    Intent i = new Intent(this, ListActivity.class);  
    startActivity(i);  
    finish();  
}
```

```
public void toHome(View view) {  
    //Returns to the home page  
    Intent i = new Intent(this, MainActivity.class);  
    startActivity(i);  
    finish();  
}
```

```
public void toCal(View view) {  
    //Opens the calendar view  
    Intent i = new Intent(this, CalActivity.class);  
    startActivity(i);  
    finish();  
}
```

```
}
```

```
*****
```

***** BuildURL.java *****

```
package com.alpharettafblla.kalendo;

import java.text.DecimalFormat;
import java.util.Calendar;
import java.util.TimeZone;

public class BuildURL {

    //School's google calendar id
    private static final String cal_id =
        "b3ugdgdvqf5ldgns33drkn314%40group.calendar.google.com";

    //API Key to allow API call
    private static final String myApiKey = "AIzaSyA21YKyARNKTxphFz95aYmGv7SQSWnEX3k";

    public static String todayURL() {
        Calendar todayDate = Calendar.getInstance();
        Calendar tomorrowDate = Calendar.getInstance();

        //Sets up max to end of current date
        tomorrowDate.add(Calendar.DATE, 1);

        //Gets time zone info
        String tz = timeZone(todayDate);

        //Limits results based on today
        String minTime = todayDate.get(Calendar.YEAR) + "-"
            + (todayDate.get(Calendar.MONTH) + 1) + "-"
            + todayDate.get(Calendar.DATE) + tz;
        String maxTime = tomorrowDate.get(Calendar.YEAR) + "-"
            + (tomorrowDate.get(Calendar.MONTH) + 1) + "-"
            + tomorrowDate.get(Calendar.DATE) + tz;

        return createBase(minTime, maxTime);
    }

    public static String listURL() {
        Calendar todayCal = Calendar.getInstance();
        Calendar endDateCal = Calendar.getInstance();

        //Sets up max to 90 days in the future
        endDateCal.add(Calendar.DATE, 90);

        //Gets time zone info
        String tz = timeZone(todayCal);
```

```

//Limits results to 90 days in the future
String minTime = todayCal.get(Calendar.YEAR) + "-"
    + (todayCal.get(Calendar.MONTH) + 1) + "-"
    + todayCal.get(Calendar.DATE) + tz;
String maxTime = endDateCal.get(Calendar.YEAR) + "-"
    + (endDateCal.get(Calendar.MONTH) + 1) + "-"
    + endDateCal.get(Calendar.DATE) + tz;

return createBase(minTime, maxTime);
}

public static String dateURL(long cal) {
    Calendar startDateCal = Calendar.getInstance();
    Calendar endDateCal = Calendar.getInstance();

    //Determines selected date using parameter (millisecond value)
    startDateCal.setTimeInMillis(cal);
    endDateCal.setTimeInMillis(cal);

    //Sets up max to end of selected date
    endDateCal.add(Calendar.DATE, 1);

    //Gets time zone info
    String tz = timeZone(startDateCal);

    //Limits results based on selected date
    String minTime = startDateCal.get(Calendar.YEAR) + "-"
        + (startDateCal.get(Calendar.MONTH) + 1) + "-"
        + startDateCal.get(Calendar.DATE) + tz;
    String maxTime = endDateCal.get(Calendar.YEAR) + "-"
        + (endDateCal.get(Calendar.MONTH) + 1) + "-"
        + endDateCal.get(Calendar.DATE) + tz;

    return createBase(minTime, maxTime);
}

public static String twitterURL(String twitterAcc) {
    if (twitterAcc.equals("AHSRaiderFBLA"))
        return "<a unselectable='on' class='twitter-timeline' " +
            "data-dnt='true' href='https://twitter.com/AHSRaiderFBLA' " +
            "data-widget-id='594912738868129792'>Tweets by @AHSRaiderFBLA</a>" +
            "<script>!function(d,s,id){var js,fjs=d.getElementsByTagName(s)[0], " +
            "p=/^http:/.test(d.location)?'http':'https';if(!d.getElementById(id)) " +
            "{js=d.createElement(s);js.id=id;js.src=p+'://platform.twitter.com/' + " +
            "widgets.js';fjs.parentNode.insertBefore(js,fjs);}}(document, " +
            "'script','twitter-wjs');</script>";
    else if (twitterAcc.equals("FultonCoSchools"))
        return "<a unselectable='on' class='twitter-timeline' " +
            "data-dnt='true' href='https://twitter.com/FultonCoSchools' " +
            "data-widget-id='594912738868129792'>Tweets by @FultonCoSchools</a>" +
            "<script>!function(d,s,id){var js,fjs=d.getElementsByTagName(s)[0], " +
            "p=/^http:/.test(d.location)?'http':'https';if(!d.getElementById(id)) " +
            "{js=d.createElement(s);js.id=id;js.src=p+'://platform.twitter.com/' + " +
            "widgets.js';fjs.parentNode.insertBefore(js,fjs);}}(document, " +
            "'script','twitter-wjs');</script>";
}

```

```

else
    return "<a unselectable=\"on\" class=\"twitter-timeline\" +
        \"data-dnt=\"true\" href=\"https://twitter.com/RaiderSport\" +
        \" data-widget-id=\"594912738868129792\">Tweets by @RaiderSport</a>\" +
        \"<script>!function(d,s,id){var js,fjs=d.getElementsByTagName(s)[0],\" +
        \"p=/^http:/.test(d.location)?'http':'https';if(!d.getElementById)\" +
        \"{js=d.createElement(s);js.id=id;js.src=p+\"://platform.twitter.com/\" +
        \"widgets.js\";fjs.parentNode.insertBefore(js,fjs);}}(document,\" +
        \"script\", \"twitter-wjs\");</script>\";
}

```

```

private static String timeZone(Calendar cal) {
    //Gets the user's time zone
    int tz = TimeZone.getDefault().getOffset(cal.getTimeInMillis());
    int offsetHours = tz/3600000;
    int offsetMinutes = (tz%3600000)/60000;
    String timeZone = "T00%3A00%3A00" +
        (new DecimalFormat("00").format(offsetHours)) + ":" +
        (new DecimalFormat("00").format(offsetMinutes));
    return timeZone;
}

```

```

private static String createBase(String min, String max) {
    //Calendar URL with all parameters filled in
    return "https://www.googleapis.com/calendar/v3/calendars/" +
        cal_id + "/events?timeMin=" + min + "&timeMax=" + max +
        "&max-results=100&singleEvents=true&orderBy=startTime&key=" + myApiKey;
}

```

```

}

```

```

*****

```


***** CheckNetwork.java *****

```
package com.alpharettafblla.kalendo;
```

```
import android.content.Context;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.util.Log;
```

```
public class CheckNetwork {
```

```
    private static final String TAG = CheckNetwork.class.getSimpleName();
```

```
    public static boolean isInternetAvailable(Context context) {
        //Checks for available Internet connection
        NetworkInfo info = (NetworkInfo) ((ConnectivityManager) context.
            getSystemService(Context.CONNECTIVITY_SERVICE)).getActiveNetworkInfo();
```

```
        if (info == null) {
            Log.d(TAG,"no internet connection");
            return false;
        }
```

```
        else {
            if(info.isConnected()) {
                Log.d(TAG," internet connection available...");
                return true;
            }
            else {
                Log.d(TAG," internet connection");
                return true;
            }
        }
    }
```

```
}
```

***** EventInfo.java *****

```
package com.alpharettafbia.kalendo;

import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

import org.json.JSONException;
import org.json.JSONObject;

public class EventInfo {

    private String name;
    private String descrip;
    private Date startDate;
    private Date endDate;
    private String startTime;
    private String endTime;
    private boolean isAllDay = false;

    SimpleDateFormat forDate = new SimpleDateFormat("yyyy-MM-dd", Locale.getDefault());
    SimpleDateFormat forDateTime = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ssZ");

    public EventInfo (JSONObject singleEvent) {
        //Will store the name, description, and timings for each event
        try {

            //Gets event name
            name = singleEvent.getString("summary");

            //Checks if description is available
            if (singleEvent.has("description"))
                descrip = singleEvent.getString("description");
            else
                descrip = "There is no description for this event";

            //Fetches timings of the event
            if (singleEvent.getJSONObject("start").has("date")) {
                //If the event has a start date then it is an all-day event
                isAllDay = true;
                String s = singleEvent.getJSONObject("start").getString("date");
                startDate = forDate.parse(s);
            }
            else {
                //Else, there must be a start dateTime and end dateTime
                startTime = singleEvent.getJSONObject("start").getString("dateTime");
                startDate = forDateTime.parse(startTime);
                endTime = singleEvent.getJSONObject("end").getString("dateTime");
                endDate = forDateTime.parse(endTime);
            }

        } catch (JSONException | ParseException e) {
            e.printStackTrace();
        }
    }
}
```

```
    }  
}
```

```
public String getName() {  
    return name;  
}
```

```
public Date getStartDate() {  
    return startDate;  
}
```

```
public Date getEndDate() {  
    return endDate;  
}
```

```
public String getStartTime() {  
    return startTime;  
}
```

```
public String getEndTime() {  
    return endTime;  
}
```

```
public String getDescrip() {  
    return descrip;  
}
```

```
public boolean isAllDay() {  
    return isAllDay;  
}
```

```
}
```

```
*****
```

***** EventList.java *****

```
package com.alpharettafbia.kalendo;
```

```
import java.util.ArrayList;
```

```
import org.json.JSONArray;
```

```
import org.json.JSONException;
```

```
import org.json.JSONObject;
```

```
public class EventList {
```

```
    public static ArrayList<EventInfo> analyzeData(JSONObject jsonObj) {  
        ArrayList<EventInfo> listOfEventInfo = new ArrayList<EventInfo>();  
        try {  
            //Get the array of events  
            JSONArray jArray = jsonObj.getJSONArray("items");  
  
            //Get the name, description, start time, and end time for each event  
            for (int i = 0; i < jArray.length(); i++) {  
                JSONObject jobject = jArray.getJSONObject(i);  
                listOfEventInfo.add(new EventInfo(jobject));  
            }  
        } catch (JSONException e) {  
            e.printStackTrace();  
        }  
        return listOfEventInfo;  
    }  
}
```

```
}
```

***** EventText.java *****

```
package com.alpharettafblla.kalendo;

import java.text.SimpleDateFormat;
import java.util.Locale;

import android.graphics.Color;
import android.graphics.Typeface;
import android.util.TypedValue;
import android.view.Gravity;
import android.widget.TextView;

public class EventText {

    private static SimpleDateFormat displayDate = new SimpleDateFormat("MMMM d, yyyy",
        Locale.getDefault());
    private static SimpleDateFormat displayTime = new SimpleDateFormat("h:mm a",
        Locale.getDefault());
    private static int bgCol = Color.argb(255, 132, 218, 209);

    public static void header(TextView textView, EventInfo indexEvent) {
        //Formats the text for the event name
        textView.setBackgroundColor(bgCol);
        textView.setTextSize(TypedValue.COMPLEX_UNIT_SP, 24);
        textView.setTypeface(null, Typeface.BOLD);
        textView.setGravity(Gravity.CENTER);
        textView.setPadding(5, 5, 5, 2);
        textView.setText(indexEvent.getName());
    }

    public static void time(TextView textView, EventInfo indexEvent) {
        //Formats the text for the event time
        textView.setBackgroundColor(bgCol);
        textView.setTextSize(TypedValue.COMPLEX_UNIT_SP, 18);
        textView.setTypeface(null, Typeface.BOLD);
        textView.setGravity(Gravity.CENTER);
        textView.setPadding(5, 2, 5, 2);
        if (indexEvent.isAllDay())
            textView.setText("All-day event");
        else
            textView.setText(displayTime.format(indexEvent.getStartDate()) + " - "
                + displayTime.format(indexEvent.getEndDate()));
    }

    public static void dateTime(TextView textView, EventInfo indexEvent) {
        //Formats the text for the event date and time
        textView.setBackgroundColor(bgCol);
        textView.setTextSize(TypedValue.COMPLEX_UNIT_SP, 18);
        textView.setTypeface(null, Typeface.BOLD);
        textView.setGravity(Gravity.CENTER);
        textView.setPadding(5, 2, 5, 2);
        if (indexEvent.isAllDay())
```

```

        textView.setText(displayDate.format(indexEvent.getStartDate())
            + "\nAll-day event");
    else
        textView.setText(displayDate.format(indexEvent.getStartDate()) + "\n"
            + displayTime.format(indexEvent.getStartDate()) + " - "
            + displayTime.format(indexEvent.getEndDate()));
}

```

```

public static void describ(TextView textView, EventInfo indexEvent) {
    //Formats the text for the event description
    textView.setBackgroundColor(bgCol);
    textView.setTextSize(TypedValue.COMPLEX_UNIT_SP, 18);
    textView.setTypeface(null, Typeface.ITALIC);
    textView.setGravity(Gravity.CENTER);
    textView.setPadding(5, 2, 5, 10);
    textView.setText(indexEvent.getDescrip());
}

```

```

public static void noEvents(TextView textView, String noVal) {
    //Text to show if there are no events
    textView.setTextSize(TypedValue.COMPLEX_UNIT_SP, 28);
    textView.setTextColor(Color.BLACK);
    textView.setGravity(Gravity.CENTER);
    textView.setText(noVal);
}

```

```

public static void spacer(TextView spacer) {
    //Spaces between the events
    spacer.setBackgroundColor(Color.TRANSPARENT);
    spacer.setTextSize(TypedValue.COMPLEX_UNIT_SP, 18);
}

```

```

}

```

```

*****

```

***** RequestCal.java *****

```
package com.alpharettafblla.kalendo;

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

import org.apache.http.HttpResponse;
import org.apache.http.HttpStatus;
import org.apache.http.StatusLine;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.DefaultHttpClient;

import android.os.AsyncTask;

class RequestCal extends AsyncTask<String, String, String> {

    @Override
    protected String doInBackground(String... uri) {
        //Makes connections with the API
        HttpClient client = new DefaultHttpClient();
        HttpResponse response;
        String responseString = null;
        try {
            response = client.execute(new HttpGet(uri[0]));
            StatusLine status = response.getStatusLine();
            if (status.getStatusCode() == HttpStatus.SC_OK) {
                ByteArrayOutputStream out = new ByteArrayOutputStream();
                response.getEntity().writeTo(out);
                responseString = out.toString();
                out.close();
            }
            else {
                //Closes the connection
                response.getEntity().getContent().close();
                throw new IOException(status.getReasonPhrase());
            }
        } catch (ClientProtocolException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
        return responseString;
    }

    @Override
    protected void onPostExecute(String result) {
        super.onPostExecute(result);
    }

}
```

LAYOUT XML FILES

***** activity_main.xml *****

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/rel_layout"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="@drawable/bg">

    <HorizontalScrollView
        android:id="@+id/horiz_scroll"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fadingEdge="none">

        <LinearLayout
            android:id="@+id/horiz_layout"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:background="@android:color/transparent"
            android:gravity="center_vertical"
            android:orientation="horizontal"
            android:paddingBottom="5dip"
            android:paddingTop="5dip">

            </LinearLayout>

        </HorizontalScrollView>

    <ScrollView
        android:id="@+id/today_scroll"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/horiz_scroll"
        android:layout_above="@id/bottom_bar"
        android:scrollbars="vertical">

        <LinearLayout
            android:id="@+id/today_layout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:orientation="vertical">

            <TextView
                android:id="@+id/today_view"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:textColor="#FFFFFF"
```



```

        android:textSize="24sp"/>

</LinearLayout>

</ScrollView>

<LinearLayout
    android:id="@+id/bottom_bar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_gravity="bottom"
    android:background="#343434"
    android:orientation="horizontal">

    <ImageButton
        android:id="@+id/list_btn"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_gravity="start|bottom"
        android:layout_weight="1"
        android:adjustViewBounds="true"
        android:contentDescription="@string/list_view"
        android:cropToPadding="false"
        android:onClick="toList"
        android:padding="2dp"
        android:scaleType="fitXY"
        android:src="@drawable/list"/>

    <ImageButton
        android:id="@+id/home_btn"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|center_horizontal"
        android:layout_weight="1"
        android:adjustViewBounds="true"
        android:contentDescription="@string/home_view"
        android:cropToPadding="false"
        android:padding="2dp"
        android:scaleType="fitXY"
        android:src="@drawable/home_on"/>

    <ImageButton
        android:id="@+id/twitter_btn"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_gravity="end|bottom"
        android:layout_weight="1"
        android:adjustViewBounds="true"
        android:contentDescription="@string/twitter_view"
        android:cropToPadding="false"
        android:onClick="toTwitterFeed"
        android:padding="2dp"
        android:scaleType="fitXY"
        android:src="@drawable/twitter"/>

    <ImageButton

```

```
android:id="@+id/cal_btn"  
android:layout_width="0dp"  
android:layout_height="wrap_content"  
android:layout_gravity="end|bottom"  
android:layout_weight="1"  
android:adjustViewBounds="true"  
android:contentDescription="@string/cal_view"  
android:cropToPadding="false"  
android:onClick="toCal"  
android:padding="2dp"  
android:scaleType="fitXY"  
android:src="@drawable/cal"/>
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

```
*****
```

***** activity_cal.xml *****

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/rel_layout"
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:background="#EEEEEE">
```

```
<TextView
    android:id="@+id/selected_date"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBottom="@+id/show_events"
    android:layout_alignLeft="@+id/calendar_view"
    android:layout_alignStart="@+id/calendar_view"
    android:layout_alignTop="@+id/show_events"
    android:layout_toLeftOf="@+id/show_events"
    android:layout_toStartOf="@+id/show_events"
    android:gravity="center"
    android:padding="5dp"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="#ffffff"
    android:textSize="20sp"/>
```

```
<Button
    android:id="@+id/show_events"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentTop="true"
    android:background="@drawable/mybutton"
    android:onClick="toDayEvents"
    android:padding="10dp"
    android:text="@string/show_events"
    android:textColor="#000000"/>
```

```
<View
    android:id="@+id/line"
    android:layout_width="fill_parent"
    android:layout_height="1dp"
    android:layout_below="@+id/selected_date"
    android:padding="2dp"
    android:background="#000000"/>
```

```
<CalendarView
    android:id="@+id/calendar_view"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_above="@+id/bottom_bar"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_below="@+id/line"
    android:background="#EEEEEE"
    android:selectedDateVerticalBar="@drawable/vertical_bar"
    android:selectedWeekBackgroundColor="#5BCEC1"
```

```
android:weekSeparatorLineColor="#1EBBAA"/>
```

```
<LinearLayout
```

```
    android:id="@id/bottom_bar"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignParentBottom="true"  
    android:layout_gravity="bottom"  
    android:background="#343434"  
    android:orientation="horizontal">
```

```
    <ImageButton
```

```
        android:id="@+id/list_btn"  
        android:layout_width="0dp"  
        android:layout_height="wrap_content"  
        android:layout_gravity="start|bottom"  
        android:layout_weight="1"  
        android:adjustViewBounds="true"  
        android:contentDescription="@string/list_view"  
        android:cropToPadding="false"  
        android:onClick="toList"  
        android:padding="2dp"  
        android:scaleType="fitXY"  
        android:src="@drawable/list"/>
```

```
    <ImageButton
```

```
        android:id="@+id/home_btn"  
        android:layout_width="0dp"  
        android:layout_height="wrap_content"  
        android:layout_gravity="bottom|center_horizontal"  
        android:layout_weight="1"  
        android:adjustViewBounds="true"  
        android:contentDescription="@string/home_view"  
        android:cropToPadding="false"  
        android:onClick="toHome"  
        android:padding="2dp"  
        android:scaleType="fitXY"  
        android:src="@drawable/home"/>
```

```
    <ImageButton
```

```
        android:id="@+id/twitter_btn"  
        android:layout_width="0dp"  
        android:layout_height="wrap_content"  
        android:layout_gravity="end|bottom"  
        android:layout_weight="1"  
        android:adjustViewBounds="true"  
        android:contentDescription="@string/twitter_view"  
        android:cropToPadding="false"  
        android:onClick="toTwitterFeed"  
        android:padding="2dp"  
        android:scaleType="fitXY"  
        android:src="@drawable/twitter"/>
```

```
    <ImageButton
```

```
        android:id="@+id/cal_btn"  
        android:layout_width="0dp"  
        android:layout_height="wrap_content"
```

```
android:layout_gravity="end|bottom"
android:layout_weight="1"
android:adjustViewBounds="true"
android:contentDescription="@string/cal_view"
android:cropToPadding="false"
android:padding="2dp"
android:scaleType="fitXY"
android:src="@drawable/cal_on"/>
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

```
*****
```

***** activity_day_events.xml *****

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context="com.example.api_restcalls.DayEvents"
    android:id="@+id/rel_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg">

    <TextView
        android:id="@+id/date_view"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:padding="10dp"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:textSize="28sp"/>

    <ScrollView
        android:id="@+id/event_scroll"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_above="@+id/back_btn"
        android:layout_below="@id/date_view"
        android:background="@android:color/transparent"
        android:scrollbars="vertical">

        <LinearLayout
            android:id="@+id/event_view"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:id="@+id/event_text"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:maxLines="10000"
                android:scrollbars="vertical"/>

        </LinearLayout>

    </ScrollView>

    <ImageButton
        android:id="@+id/back_btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_gravity="start|bottom"
        android:adjustViewBounds="true"
        android:background="@android:color/transparent"
        android:contentDescription="@string/back"
        android:cropToPadding="false"
        android:onClick="backToCal"
```

```
android:paddingLeft="1dp"  
android:paddingRight="1dp"  
android:paddingBottom="8dp"  
android:paddingTop="1dp"  
android:scaleType="fitXY"  
android:src="@drawable/back"/>
```

```
</RelativeLayout>
```

```
*****
```

***** activity_info.xml *****

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context="com.example.api_restcalls.DayEvents"
    android:id="@+id/rel_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg">
```

```
<ScrollView
    android:id="@+id/info_scroll"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:layout_above="@+id/back_btn"
    android:background="@android:color/transparent"
    android:gravity="top"
    android:scrollbars="vertical">
```

```
<LinearLayout
    android:id="@+id/info_view"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="10dp">
```

```
<TextView
    android:id="@+id/info_text"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:maxLines="10000"
    android:scrollbars="vertical"
    android:textColor="#000000"
    android:textSize="20sp"/>
```

```
</LinearLayout>
```

```
</ScrollView>
```

```
<ImageButton
    android:id="@+id/back_btn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_gravity="start|bottom"
    android:adjustViewBounds="true"
    android:background="@android:color/transparent"
    android:contentDescription="@string/back"
    android:cropToPadding="false"
    android:onClick="goBack"
    android:paddingLeft="1dp"
    android:paddingRight="1dp"
    android:paddingBottom="8dp"
    android:paddingTop="1dp"
    android:scaleType="fitXY"
```



```
android:src="@drawable/back"/>
```

```
</RelativeLayout>
```

```
*****
```

***** activity_list.xml *****

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/rel_layout"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="@drawable/bg">

    <ScrollView
        android:layout_width="wrap_content"
        android:layout_height="fill_parent"
        android:layout_above="@+id/bottom_bar"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentTop="true"
        android:scrollbarAlwaysDrawVerticalTrack="true"
        android:scrollbars="vertical"
        android:visibility="visible">

        <LinearLayout
            android:id="@+id/list_events"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            </LinearLayout>

        </ScrollView>

        <LinearLayout
            android:id="@+id/bottom_bar"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:layout_gravity="bottom"
            android:background="#343434"
            android:orientation="horizontal">

            <ImageButton
                android:id="@+id/list_btn"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_gravity="start|bottom"
                android:layout_weight="1"
                android:adjustViewBounds="true"
                android:contentDescription="@string/list_view"
                android:cropToPadding="false"
                android:padding="2dp"
                android:scaleType="fitXY"
                android:src="@drawable/list_on"/>

            <ImageButton
                android:id="@+id/home_btn"
                android:layout_width="0dp"
```

```
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|center_horizontal"
    android:layout_weight="1"
    android:adjustViewBounds="true"
    android:contentDescription="@string/home_view"
    android:cropToPadding="false"
    android:onClick="toHome"
    android:padding="2dp"
    android:scaleType="fitXY"
    android:src="@drawable/home"/>
```

```
<ImageButton
    android:id="@+id/twitter_btn"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="end|bottom"
    android:layout_weight="1"
    android:adjustViewBounds="true"
    android:contentDescription="@string/twitter_view"
    android:cropToPadding="false"
    android:onClick="toTwitterFeed"
    android:padding="2dp"
    android:scaleType="fitXY"
    android:src="@drawable/twitter"/>
```

```
<ImageButton
    android:id="@+id/cal_btn"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="end|bottom"
    android:layout_weight="1"
    android:adjustViewBounds="true"
    android:contentDescription="@string/cal_view"
    android:cropToPadding="false"
    android:onClick="toCal"
    android:padding="2dp"
    android:scaleType="fitXY"
    android:src="@drawable/cal"/>
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

```
*****
```

***** activity_twitter_feed.xml *****

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context=".TwitterFeed"
    android:id="@+id/rel_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg">
```

```
<LinearLayout
    android:id="@+id/twitter_options"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="2dp">
```

```
<Spinner
    android:id="@+id/spinner1"
    android:layout_width="0dp"
    android:layout_height="fill_parent"
    android:layout_weight="1"
    android:entries="@array/twitters"
    android:gravity="start"
    android:popupBackground="#ffffff"
    android:prompt="@string/twitter_prompt"
    android:spinnerMode="dialog"/>
```

```
<Button
    android:id="@+id/submit_btn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@drawable/mybutton"
    android:onClick="updateTwitter"
    android:padding="10dp"
    android:text="@string/submit"/>
```

```
</LinearLayout>
```

```
<WebView
    android:id="@+id/timeline_webview"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/twitter_options"
    android:layout_above="@id/bottom_bar"/>
```

```
<LinearLayout
    android:id="@+id/bottom_bar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_gravity="bottom"
    android:background="#343434"
    android:orientation="horizontal">
```

```
<ImageButton
    android:id="@+id/list_btn"
```

```
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="start|bottom"
    android:layout_weight="1"
    android:adjustViewBounds="true"
    android:contentDescription="@string/list_view"
    android:cropToPadding="false"
    android:onClick="toList"
    android:padding="2dp"
    android:scaleType="fitXY"
    android:src="@drawable/list"/>
```

```
<ImageButton
    android:id="@+id/home_btn"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|center_horizontal"
    android:layout_weight="1"
    android:adjustViewBounds="true"
    android:contentDescription="@string/home_view"
    android:cropToPadding="false"
    android:onClick="toHome"
    android:padding="2dp"
    android:scaleType="fitXY"
    android:src="@drawable/home"/>
```

```
<ImageButton
    android:id="@+id/twitter_btn"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="end|bottom"
    android:layout_weight="1"
    android:adjustViewBounds="true"
    android:contentDescription="@string/twitter_view"
    android:cropToPadding="false"
    android:padding="2dp"
    android:scaleType="fitXY"
    android:src="@drawable/twitter_on"/>
```

```
<ImageButton
    android:id="@+id/cal_btn"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="end|bottom"
    android:layout_weight="1"
    android:adjustViewBounds="true"
    android:contentDescription="@string/cal_view"
    android:cropToPadding="false"
    android:onClick="toCal"
    android:padding="2dp"
    android:scaleType="fitXY"
    android:src="@drawable/cal"/>
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

```
*****
```

***** main.xml *****

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:tools="http://schemas.android.com/tools"
      tools:context="com.example.fb1a1.MainActivity">
```

```
    <item
        android:id="@+id/app_info"
        android:title="@string/app_info"
        app:showAsAction="never"/>
```

```
</menu>
```

***** strings.xml *****

<?xml version="1.0" encoding="utf-8"?>

<resources>

```
<string name="app_name">Kalendo</string>
<string name="action_settings">Settings</string>
<string name="app_info">About Kalendo</string>
<string name="list_view">List View</string>
<string name="cal_view">Calendar View</string>
<string name="home_view">Home View</string>
<string name="twitter_view">Twitter View</string>
<string name="see_twitter">View Twitter Feed</string>
<string name="title_activity_cal">Calendar View</string>
<string name="title_events">Events For Selected Day</string>
<string name="title_activity_list">Upcoming Events</string>
<string name="no_today">There are no events for today.</string>
<string name="no_selected">There are no events for the selected day.</string>
<string name="no_upcoming">There are no upcoming events.</string>
<string name="show_events">Show Events</string>
<string name="back">Back</string>
<string name="submit">Update Twitter</string>
<string name="twitter_prompt">Choose a Twitter Feed to View</string>
<string-array name="twitters">
    <item>AHSRaiderFBLA</item>
    <item>FultonCoSchools</item>
    <item>RaiderSport</item>
</string-array>
```

</resources>

***** styles.xml *****

<resources>

<style name="AppBaseTheme" parent="Theme.AppCompat.Light">
</style>

<style name="AppTheme" parent="AppBaseTheme">
 <item name="android:actionBarStyle">@style/GrayActionBar</item>
</style>

<style name="GrayActionBar" parent="@android:style/Widget.Holo.Light.ActionBar">
 <item name="android:titleTextStyle">@style/TitleBarTextColor</item>
 <item name="android:background">#343434</item>
</style>

<style name="TitleBarTextColor"
 parent="@style/TextAppearance.AppCompat.Widget.ActionBar.Title">
 <item name="android:textColor">#FFFFFF</item>
</style>

</resources>

ANDROID MANIFEST

***** AndroidManifest.xml *****

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xml>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.alpharettafblla.kalendo"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="17"
        android:targetSdkVersion="22" />

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme"
        android:supportsRtl="false">
        <activity
            android:name="com.alpharettafblla.kalendo.MainActivity"
            android:configChanges="orientation"
            android:label="@string/app_name"
            android:screenOrientation="portrait" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name="com.alpharettafblla.kalendo.TwitterFeed"
            android:configChanges="orientation"
            android:label="@string/twitter_view"
            android:screenOrientation="portrait" >
        </activity>
        <activity
            android:name="com.alpharettafblla.kalendo.CalActivity"
            android:configChanges="orientation"
            android:label="@string/title_activity_cal"
            android:screenOrientation="portrait" >
        </activity>
        <activity
            android:name="com.alpharettafblla.kalendo.ListActivity"
            android:configChanges="orientation"
            android:label="@string/title_activity_list"
            android:screenOrientation="portrait" >
```

```
</activity>
<activity
    android:name="com.alpharettafbla.kalendo.DayEvents"
    android:configChanges="orientation"
    android:label="@string/title_events"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name="com.alpharettafbla.kalendo.InfoActivity"
    android:configChanges="orientation"
    android:label="@string/app_info"
    android:screenOrientation="portrait" >
</activity>
</application>
```

```
</manifest>
```

```
*****
```