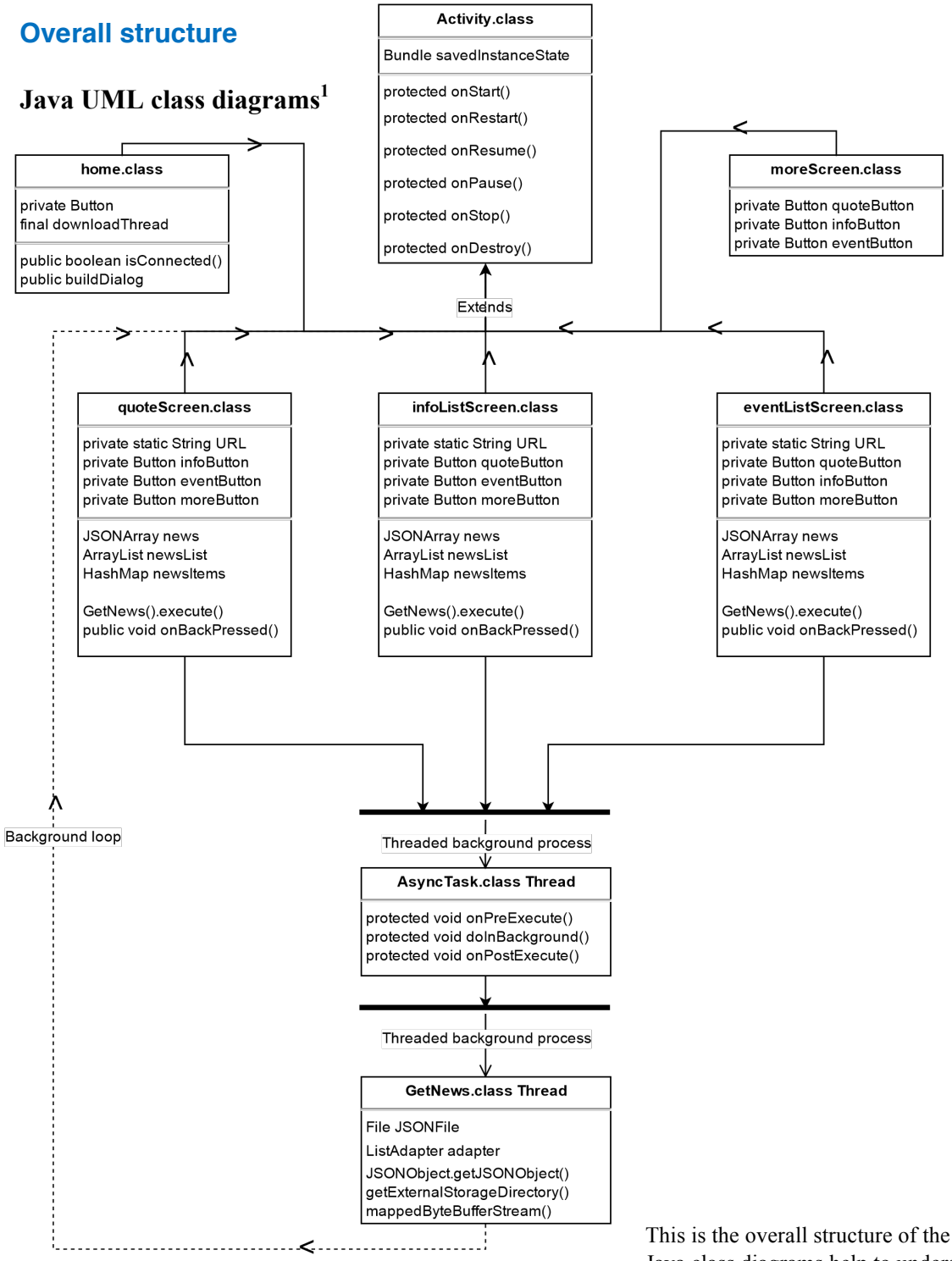
Criterion D: Product design

Overall structure

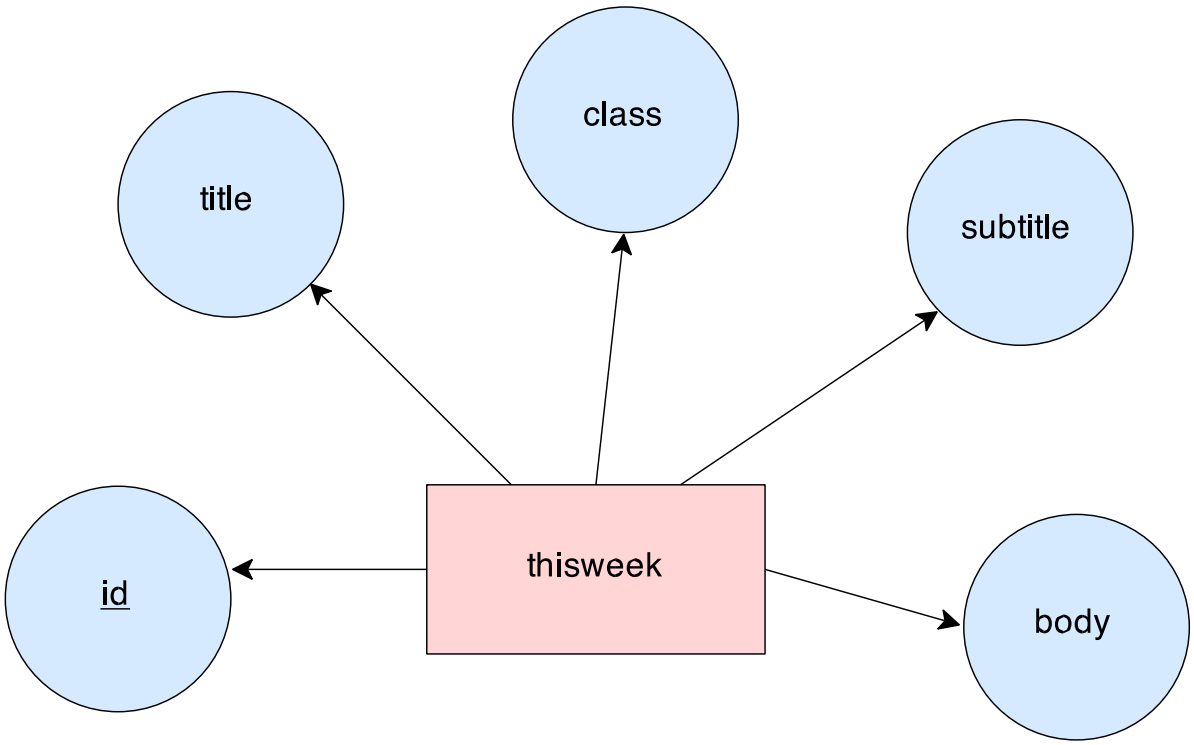
Java UML class diagrams[1]



This is the overall structure of the application. The visual Java class diagrams help to understand how the application will function, and what processes will be done.

[1] Flowchart created through: "Flow Chart Maker and Diagrams Online." Flow Chart Maker & Online Diagram Software. https://www.draw.io/.

GoDaddy database ER diagram[2]



thisweek

id	class	title	subtitle	body

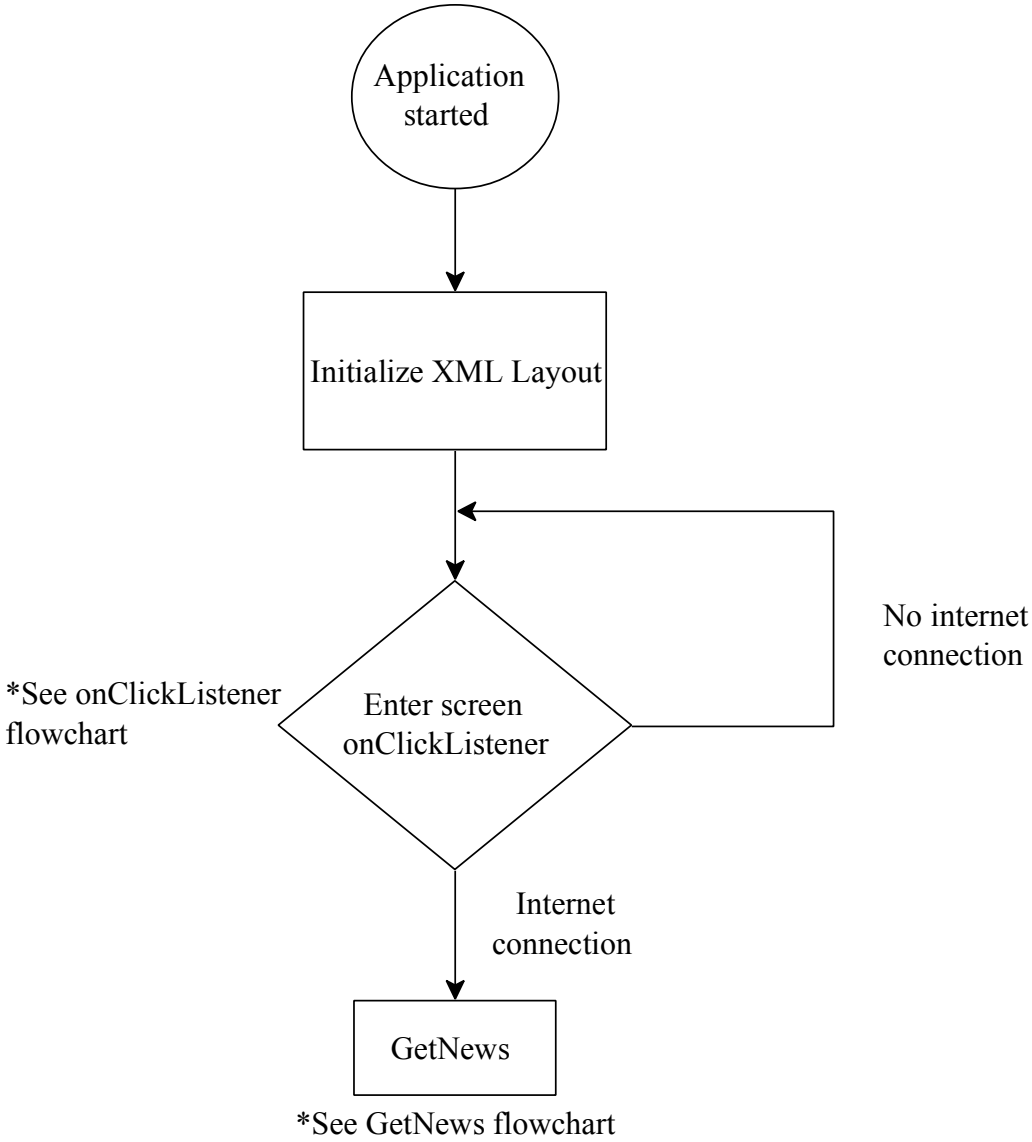
The ER diagram represents the structure of the online GoDaddy news database:

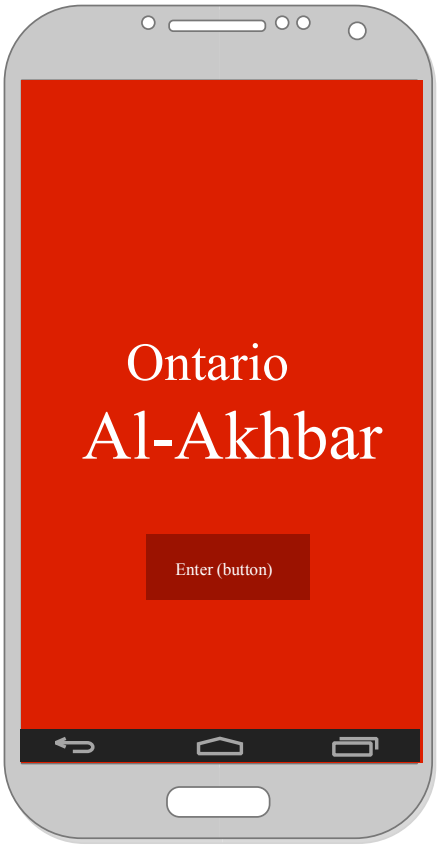
- The key field “id” helps to make every entry unique
- Table is normalized to avoid insertion, deletion, and update anomalies (very important, as this data will be fetched by application)

[2] ER diagram created through: "Flow Chart Maker and Diagrams Online." Flow Chart Maker & Online Diagram Software. https://www.draw.io/.

Internal structure

Overall application process flowchart[3]





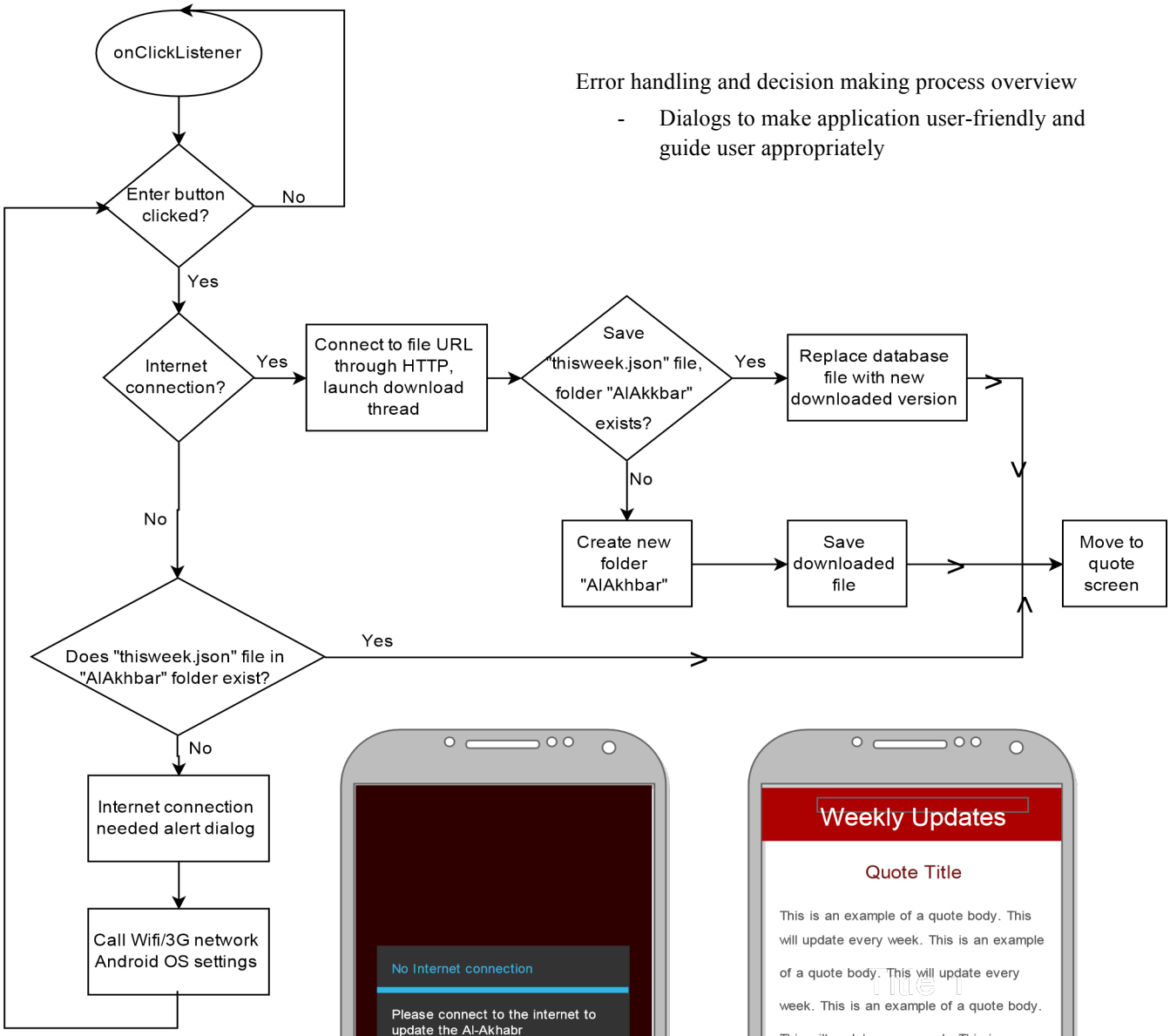
Home page design

Overall process flowchart of the application:

- Home page design shows the enter screen layout on an Android phone
- Button must be size of one finger for easy usability

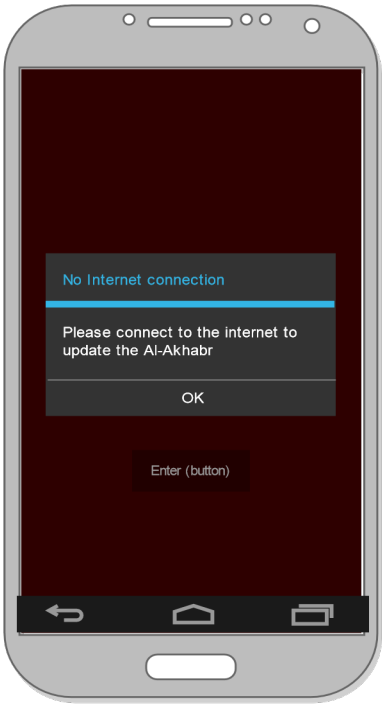
[3] Flowchart and screen layout created through: "Flow Chart Maker and Diagrams Online." Flow Chart Maker & Online Diagram Software. https://www.draw.io/.

Button clicked activity process flowchart[4]

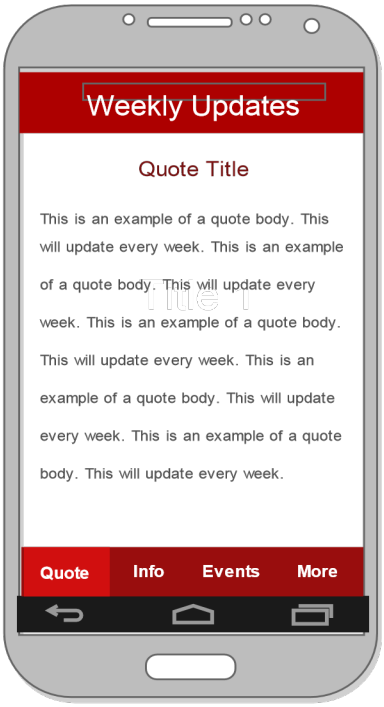


Error handling and decision making process overview

- Dialogs to make application user-friendly and guide user appropriately



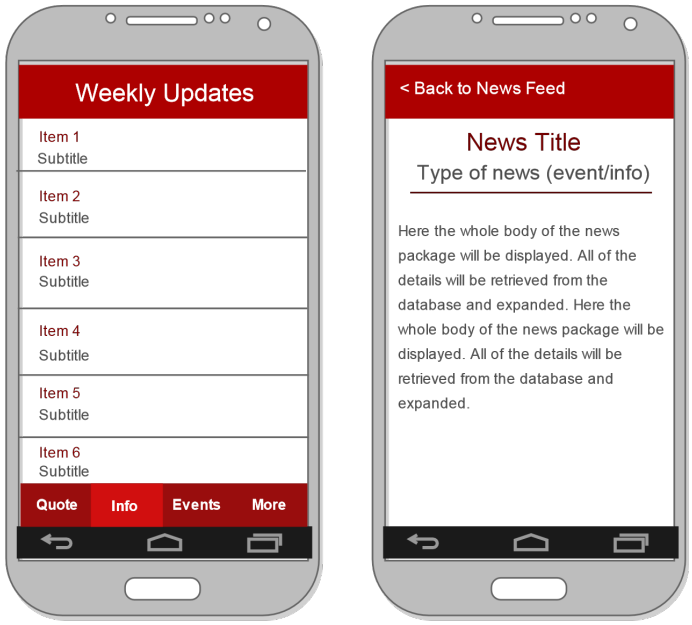
Home page (network dialog) design



Quote page design

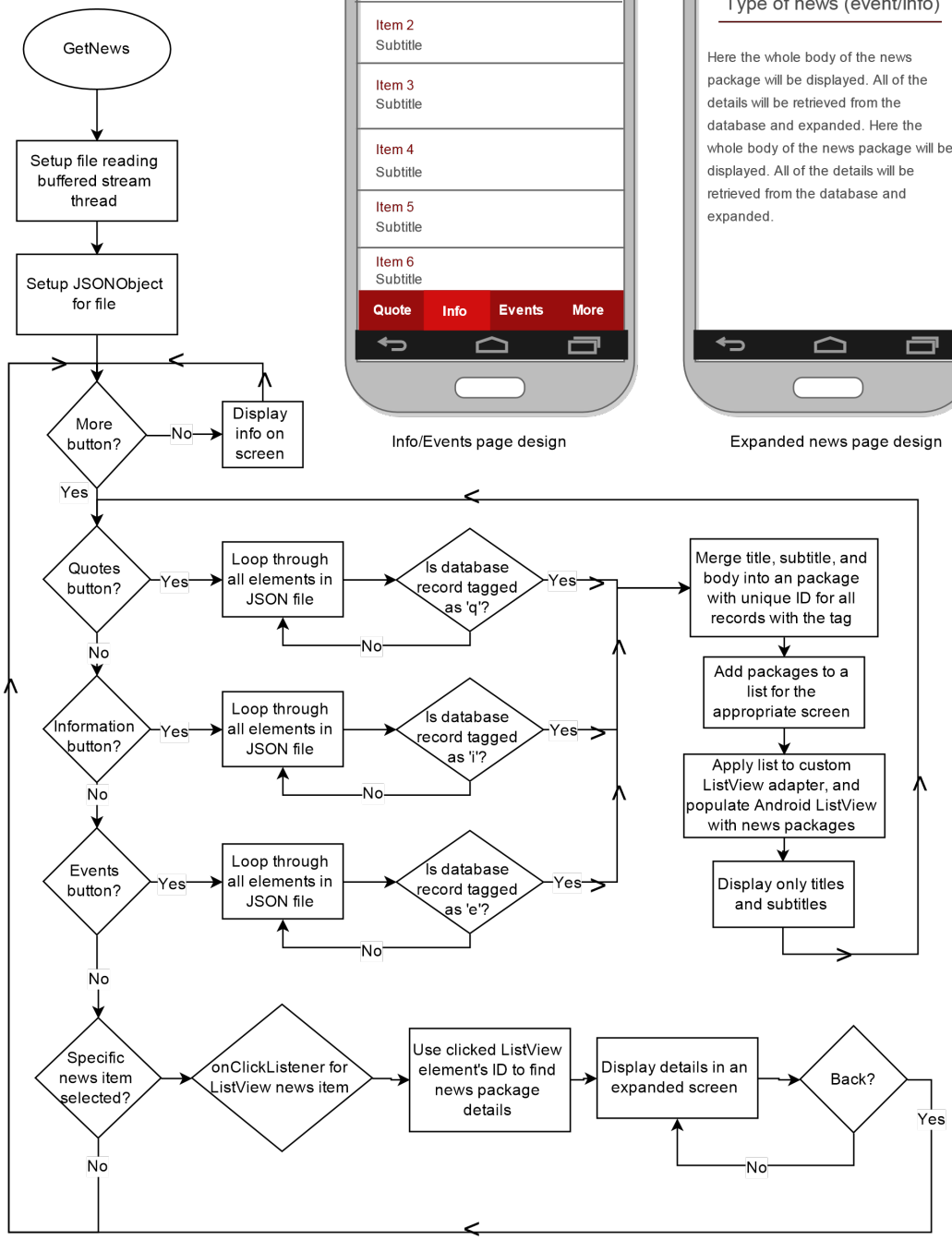
[4] Flowchart and screen layout created through: "Flow Chart Maker and Diagrams Online." Flow Chart Maker & Online Diagram Software. https://www.draw.io/.

Fetch news content flowchart[5]



Info/Events page design

Expanded news page design



Complex download thread and news fetching process outlined through a flowchart

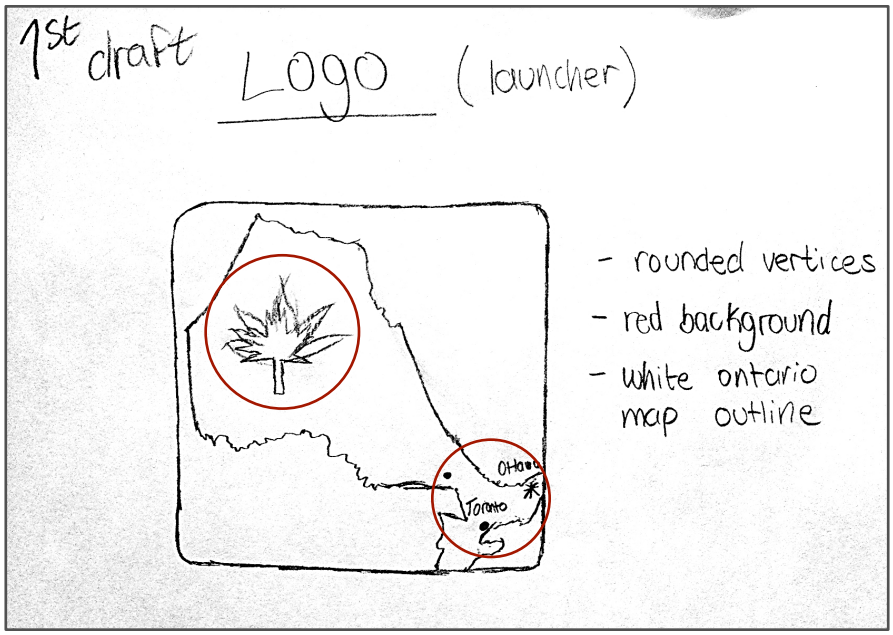
- News items organized in appropriate tabs for easy access
- Screen layouts should be optimized for high content density

[5] Flowchart and screen layout created through: "Flow Chart Maker and Diagrams Online." Flow Chart Maker & Online Diagram Software. https://www.draw.io/.

Pseudo code for the whole application activity process

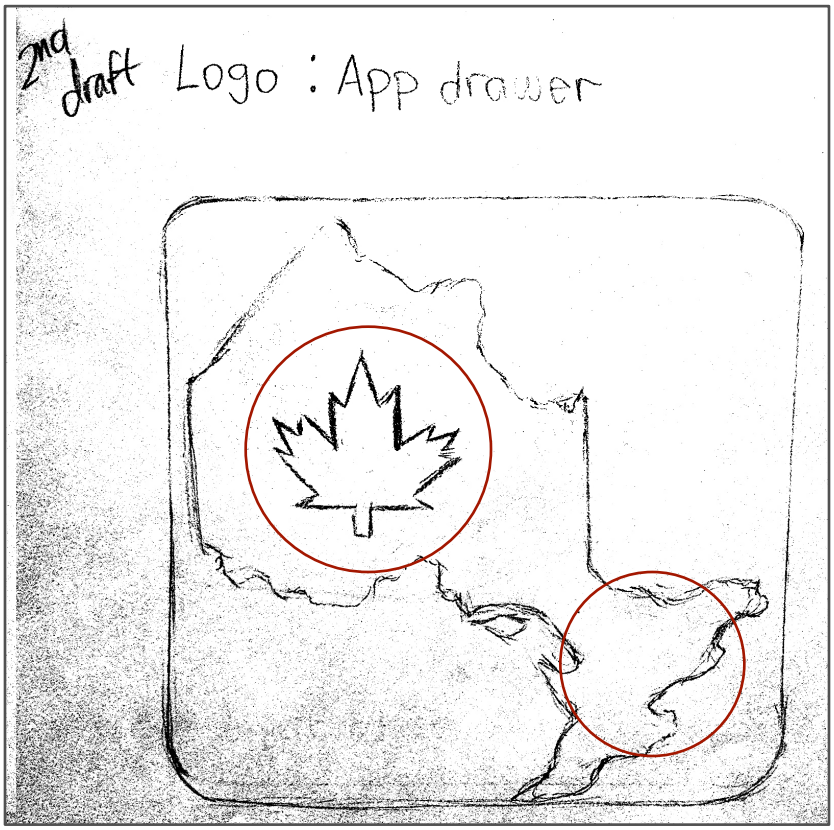
1. Launch application through logo
2. Start onCreate() for first Activity: enter page
3. Set XML layout for activity, and initialize the enter button and a download thread
4. onClickListener checks to see if the user clicks the button
    a. If clicked, the application checks for any connection to the internet (WiFi/3G)
    b. If connected, the application launches the background download thread: connects to the URL: http://zainzulfiqar.com/thisweek.json to download and save the database file
        i. Create new folder "AlAkhbar" in storage. If folder already exists, replace it
        ii. Update local storage (SD/internal). The button process is finished and next screen/activity is displayed
    c. If no internet connection, the application checks to see if there is an already existing "thisweek.json" database file in the folder "AlAkhbar" in the device's local storage (SD/Internal)
        i. If file does exist, the enter button process ends and the button process is finished and the next screen/activity is displayed
        ii. If file does not exist (no internet + no database file = first time use of app), present a dialog box to establish a network connection
        iii. Launch the WiFi/3G networks settings screen for the user
        iv. If download successful, go through the process in steps 4.b.i. – 4.b.ii.
5. Quote screen activity is launched and started
6. Set XML layout for activity, and initialize 4 buttons at the bottom for tabs
7. Initialize JSONObject object to hold the parsed data
8. Back button method is overwritten to "do nothing"
9. GetNews method is launched
10. The database file in "AlAkhabr/thisweek.json" is buffered into a readable stream
11. The stream is parsed into the JSONObject
12. JSONObject parses the JSON formatted information into a string format.
13. String is filtered through if statements to find all the appropriate information tagged as 'q' for quote.
14. This information is stored into the appropriate different string variables
15. The strings are paired with key-value hashmap (key ID-based list/array) to a list
16. List is read by the quote activity, and the appropriate string variable in the hashmap key-value pair updates the appropriate XML attribute to display the information
17. HTML Java code extension helps to visually format data
18. Different buttons navigate between different types of news feed (Quote, Events, Information, More)
19. Each button has a custom XML layout assigned to it, and has a unique listener
20. Based on the onClickListener activated, the activity switches to the appropriate tab activity
21. Steps 9a to 11 are repeated once more, but for the appropriate activity/screen. NOTE: the information tagged will be different: 'i', or 'e'

Logo/icon design



First rough sketch with icon specifications.

App for Ontario users. So the icon was based on Ontario map.



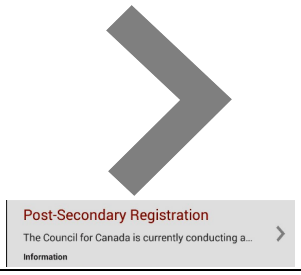
Ideas were skillfully transformed into a final sketch with the help of an artist (friend named Annika).

Digital design with Adobe Photoshop.

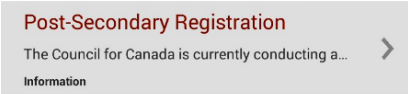
Icon is red to follow IIC's color scheme.

NOTE: Image is pixelated as the standard Android logo specification is: HDPI screens, 96 x 96 pixels

List of resources

Resource type	Description
News items	All news item descriptions will be provided by the IIC through their email alerts.
Application font	Default Android font, Roboto, used for all Android devices/systems, sans-serif for easy readability on a digital screen.
Logo/icon	Idea was described to Annika. The artist replicated the ideas into a final sketch. Replicated into a digital design through Adobe Photoshop.
Arrow graphic on news item in ListView 	"Right_grey_arrow Clip Art." Right Grey Arrow Clip Art. Accessed December 27, 2014. http://www.clker.com/clipart-right-grey-arrow.html.
Java and XML code libraires	Code was found through the help of multiple sources providing guides. The exact code was changed and adapted for my purposes, so only the idea/structure of the code was replicated, not the code: - "Codepath/android_guides." GitHub. Accessed January 15, 2015. https://github.com/codepath/android_guides/wiki/Using-an-ArrayAdapter-with-ListView. - "Android Custom ListView with Image and Text." Android Custom ListView with Image and Text. Accessed January 16, 2015. http://www.androidhive.info/2012/02/android-custom-listview-with-image-and-text/. - "AsyncTask." AsyncTask | Android Developers. Accessed January 17, 2015.http://developer.android.com/reference/android/os/AsyncTask.html. - "BufferedInputStream." BufferedInputStream | Android Developers. Accessed January 18, 2015. http://developer.android.com/reference/java/io/BufferedInputStream.html - "How to Create REST API for Android App Using PHP, Slim and MySQL - Day 1/2." How to Create REST API for Android App Using PHP, Slim and MySQL - Day 1/2. Accessed January 18, 2015. http://www.androidhive.info/2014/01/how-to-create-rest-api-for-android-app-using-php-slim-and-mysql-day-12-2/. - "How to Build a Mobile App with an App Engine Backend (Tutorial)." Google Developers. Accessed January 18, 2015. https://cloud.google.com/developers/articles/how-to-build-mobile-app-with-app-engine-backend-tutorial/.

List of techniques

Technique type	Description
Designing graphics	Adobe Photoshop used to design the application icon/logo, and edit the arrow graphic.
Button clicked/tapped color	Color of button changes to a different shade when tapped. Helps user know if the button was tapped.
HTML Java tags	HTML tags are placed in with database entries to style parts of content. Hyperlinks in the news were formatted this way.
ListView XML custom adapter design	A custom list item template was designed to change the way list elements show the news items. 
Object-Oriented Java	With the help of classes, objects, and threads, a strong application with many extra features was developed.
Tab colors	The tab on which the user views information is a brighter red than the other tabs so that the user knows what page he/she is on.
Background threads	AsyncTask threads used to download the news in the background so that the user does not need to wait for the download to finish.
If-statements	Helped to create specific conditions for the offline system: if internet was not available and no database file, if internet not available but database file available, if internet available. Helped to control the direction of the application.
Local storage read/write	Helps to make an offline application by saving the downloaded database file into storage, and then loading it to display app content.
Client-server connection	The database file is downloaded from the domain server to update application.
Independent Android activity based application	Using activities, every screen can run independently. Helps free RAM and make application stable.
XML based screen GUI	Helped ensure a strong, consistent design for the application. XML set the design, while Java set the algorithms of the application.

Test plan

The following scans show testing of the Android application. Improvements were made based on the feedback from the testers in my ITGS class.

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
Opening page: Enter button	White box: Enter button to launch app	Button, onClickListener	Transition to Quote Page	Correct Enter button leads to home screen	Works fine No delay and works correctly	Opening Page
Network Connection, Opening page	White box: See if app connects to any WiFi/3G network	Network Listener	If no network, asks to join a network. If connected, goes to Quotes page (file download is already completed)	Correct Correct	Works fine If network is working, app functions correctly	Opening Page
File reading, Opening page	White box: See if App can read existing database file in the SD, or else it downloads	Storage, and Network	If not connected, and no file - it asks to join network and downloads file. If not connected and file is there - it uses the file and continues. If connected and no file/file exists, it updates the storage for the newer verison	Correct Functions as described	Works fine Small delay on Quotes page after first connecting to network	Opening Page
Tabbed buttons working on each screen	White box: The individual buttons linking all of the other screens/ pages for each screen	Buttons, Activity launching, Tabbed navigation	All of the buttons should link to their respective page, no matter what screen. Make sure they all work.	Correct Functions as described	Works fine No delay between tabs	Content page

Test 1

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
Download activity	White box: Content is loaded into the application	Threads, Download Manager	The AsyncTask thread should download the content and update the UI at the same time, when launching app for the first time - or when updating the database. This should happen simultaneously	Correct Functions as described	Works fine, a bit slow - depending on the network connection.	Content page
News View	White box: click the list to open it	ListView	The related title and subtitle and body of the news headline should be viewed in a single screen, highlighting the details.	Correct Correct	Works fine Perfect display	Single screen

Zayne

Zayne Thower

Feb/4/15

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref In Product
Enter screen works as planned	1. Click app icon launcher 2. Click "Enter" button for new screen	White box testing of launching the application and clicking Enter button	(1) for each test case - Application should launch to title page "Ontario Weekly Al-Akhbar" - (2) Enter button should advance to news list screen	App icon and enter button works as planned	Slight delay when entering the app	Launching/ Enter screen
Internet connection dialog works	1. Launch application process without internet connection, and delete any files in SD. 2. Launch application process without internet, keeping existing files. 3. Launch application process with any type of Internet connection (WiFi/ 3G)	White box testing of internet connection for application	- (1) If no internet connection and no database file, network settings dialog should appear - (2) If no internet connection and database file exists, move to next screen - (3) If internet connection, download new database file and move to next screen	-If no internet, screen is re-directed to network connection page -If there is internet, app works correctly	No delay when redirecting	Launching/ Enter screen
Tabbed screens work appropriately	1. Click all of the tabs for the different news feed categories 2. Check if all of the buttons work on EACH different tab 3. Click back button to see if process switches or remains the same	White box testing of the tab-buttons and their screens in the application	- All of the buttons should take the users to the corresponding news category. - The tabs should respond to the user click in a fast manor. - Content displayed should switch, different list of information - Current page's tab should be in bright red (aid the user)	Buttons correspond to correct page Each page displays correct information Current tab is differentiated from the others	No delay when switching between tabs	Quote, Information, Events, and More screens

Test (2)

Tested by: Zayne Thawer
Zayne

Feb 12/15

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref In Product
ListView elements work as planned	1. Click the listed items in the "Information", "Events", and "More" screens 2. Scroll through the content of the new screen opened 3. Click the different list elements in the different tabs 4. Scroll through items to make sure all items are accessible.	White box testing of ListView elements	- The corresponding list item should expand its details to another screen containing the body. - The relating title should be carried forward to the next screen - No duplicate list items in multiple tabs, everything should be categorized	Headings a lign with correct article No duplication of list items	No listed items on "More" screen	Information, Events, and More screens
Expanded news screen should display all details	1. Click a news item, it should open up a new screen with all the details 2. Click the back button	White box testing of ListView/expanded screen buttons.	- A new screen should be launched containing all of the details - When the back button is pressed, the user should continue with the last activity (last tab screen)	New screen is launched Back button functions correctly	Buttons have no delay	Expanded news screen
Database entries should be smooth	1. Enter data into the phpMyAdmin text boxes 2. Enter data through SQL queries (if you can) 3. Enter different types of information for each field 4. Enter a different tag into the system (anything other than 'i', 'q', 'e')	Black box testing of database entries, and SQL statements	- All of the data should construct a unique record into the database - Errors should occur if the data entered is out of bounds, or is not entered correctly. - If everything is entered right, there should not be any formatting errors in the application - If the news items are tagged with something other than 'i', 'e', or 'q', it will not appear on any of the tabbed news lists. - Through SQL, the user should be able to create a whole database, tables, fields, and enter records.	Not accessible to user at the moment	User is not allowed to add or edit information in database	GoDaddy phpMyAdmin database

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
HTML tags should format the text	1. Place HTML tags into the database when entering the news details into phpMyAdmin 2. Use different HTML tags for different styles 3. Design the title, subtitle, and body	Black box testing of the database entries, and HTML tags.	- Not all of the tags will work. Only about 50% percent are parsed through Java. - Important information should be bolded, underlined, or italicized. - Every news item should have unique formatting based on the right HTML tags	User is not allowed to access "phpMyAdmin"	Tags are not accessible	GoDaddy phpMyAdmin database, expanded news screen
Hyperlinks all lead to the right location	1. Click the hyperlinks underlined in blue 2. Browse through the link that opens in the default internet browser 3. Click device return button	White box testing of all the hyperlinks leading to external pages in the application.	- When the user clicks the link, they should be taken to their default browser - The appropriate content should load up - When the return button on the device is pressed, the user should be taken back to the application	Link leads to correct website Return button leads user back to app	Delay when accessing the website (based on network connection)	Expanded news screen, internet browser
Special characters should be parsed by XML/Java	1. Enter any special characters with their unicode numbers into the database 2. Enter any Java special modifiers into the database	Black box testing of the database entries, and XML/Java special character parsing	- The unicode number should translate to the special character to display when reading the content (like bullet points)	User is not allowed to access "phpMyAdmin"	Tags are not accessible	Expanded news screen
XML custom adapter should work well	1. Enter all sorts of values into the title field in the phpMyAdmin database 2. Enter HTML tags into the field	Black box testing of the XML ListView adapter	- No matter how long the title, there should be 3 dots at the end to indicate more information - No HTML modifier applied to title, so noting should happen with HTML tags - The theme should remain the same, nothing can be modified from the outside. - If the body is long, then the screen should begin to scroll	User is not allowed to access "phpMyAdmin"	XML ListView adapter is not accessible	ListView list in all tabs

Fixing the errors

The following screen snapshots of the code show the before and after changes made to correct the major errors present in the test plan.

Test plan #2, test item #4 – fixing more screen details

```java
//initialize the quote screen - setup the layout
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.more_screen);

    //buttons leading to the other activities in the application
    infoButton = (Button) findViewById(R.id.infoButton);
    infoButton.setOnClickListener((view) -> {
            Intent i = new Intent(view.getContext(), infoListScreen.class);
            i.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
            overridePendingTransition(0,0); //0 for no animation
            startActivity(i);
    });
    eventButton = (Button) findViewById(R.id.eventButton);
    eventButton.setOnClickListener((view) -> {
            Intent i = new Intent(view.getContext(), eventListScreen.class);
            i.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
            overridePendingTransition(0,0); //0 for no animation
            startActivity(i);
    });
    quoteButton = (Button) findViewById(R.id.quoteButton);
    quoteButton.setOnClickListener((view) -> {
            Intent i = new Intent(view.getContext(), quoteScreen.class);
            i.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
            overridePendingTransition(0,0); //0 for no animation
            startActivity(i);
    });
}
```

No details about the IIC present in the moreScreen.java file

Before

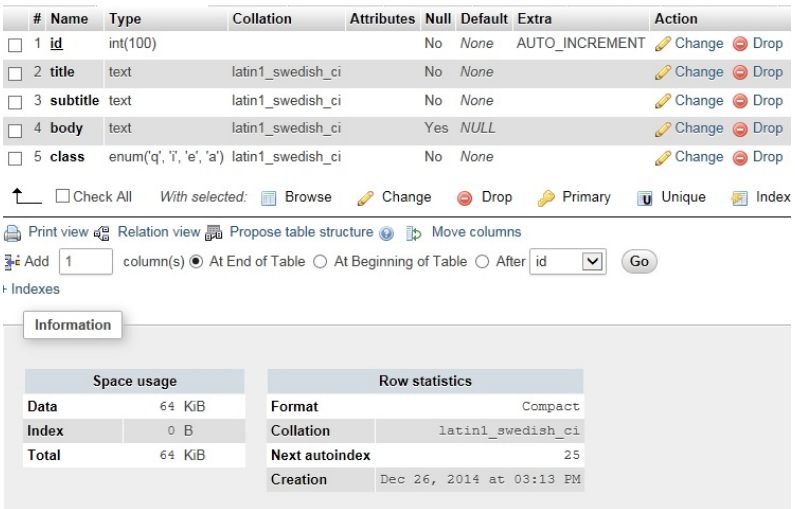
After

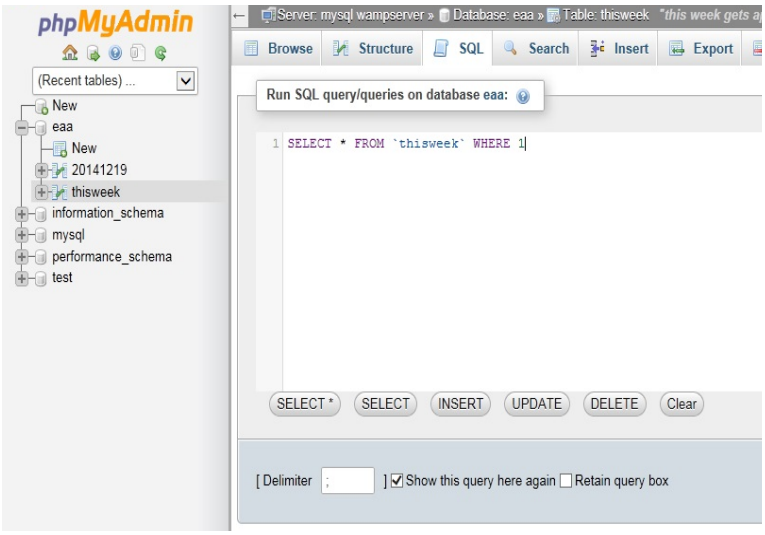
```java
});
eventButton = (Button) findViewById(R.id.eventButton);
eventButton.setOnClickListener((view) -> {
        Intent i = new Intent(view.getContext(), eventListScreen.class);
        i.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
        overridePendingTransition(0,0); //0 for no animation
        startActivity(i);
});
quoteButton = (Button) findViewById(R.id.quoteButton);
quoteButton.setOnClickListener((view) -> {
        Intent i = new Intent(view.getContext(), quoteScreen.class);
        i.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
        overridePendingTransition(0,0); //0 for no animation
        startActivity(i);
});

TextView body = (TextView) findViewById(R.id.moreBody);
body.setText(Html.fromHtml("<b>Ismaili Council for Canada (IIC)</b><br>" +
        "49 Wynford Drive<br>" +
        "Toronto, Ontario M3C 1K1<br>" +
        "CANADA<br>" +
        "Tel: +1 (416) 646-6965<br><br>" +
        "<b>Ismaili Council for Ontario</b><br>" +
        "149-151 Bartley Avenue<br>" +
        "Toronto, Ontario M4A 1C9<br>" +
        "CANADA<br>" +
        "Tel: +1-416-751-4001<br><br><br><br>" +
        "<i>Application developed by Zain Zulfiqar</i>"));
```

Hard-coded details about the IIC in the More screen tab. HTML tags are used to style the layout

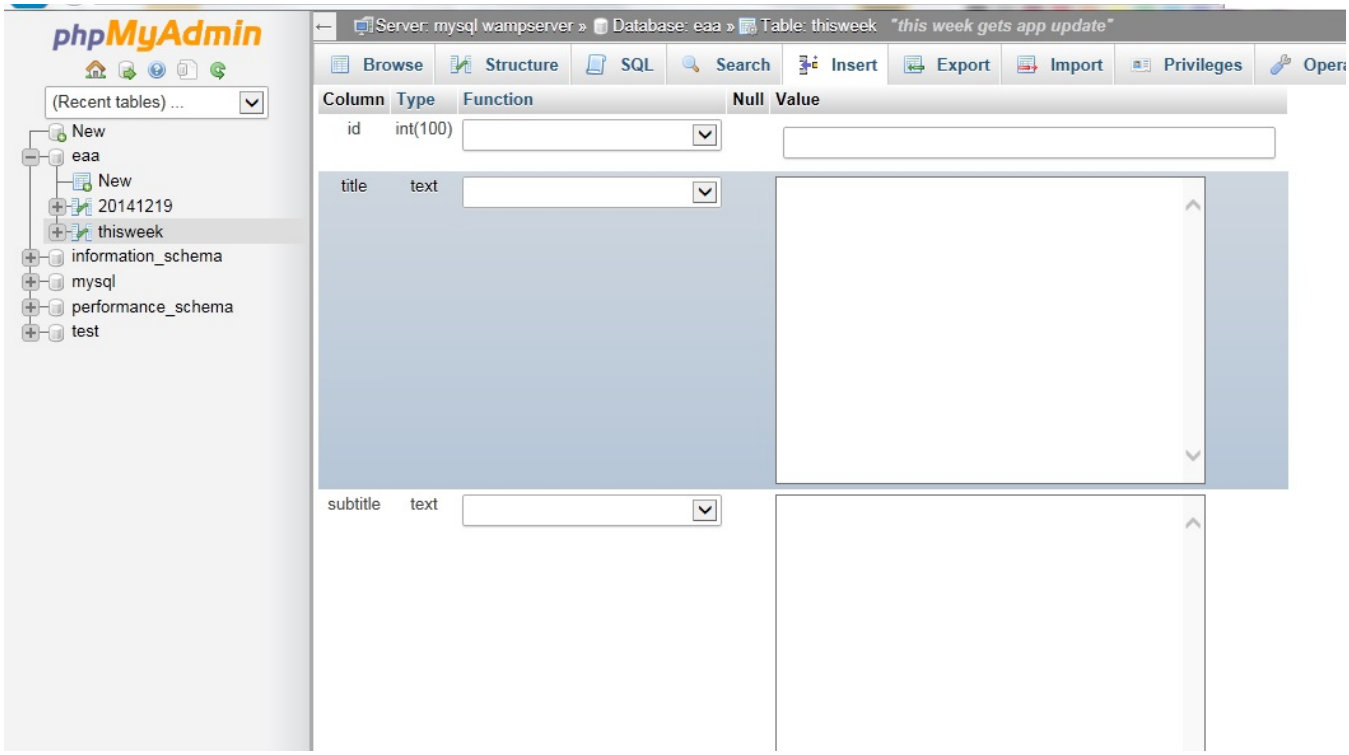
Test plan #2, test item #6 – allow IIC admin to edit news database





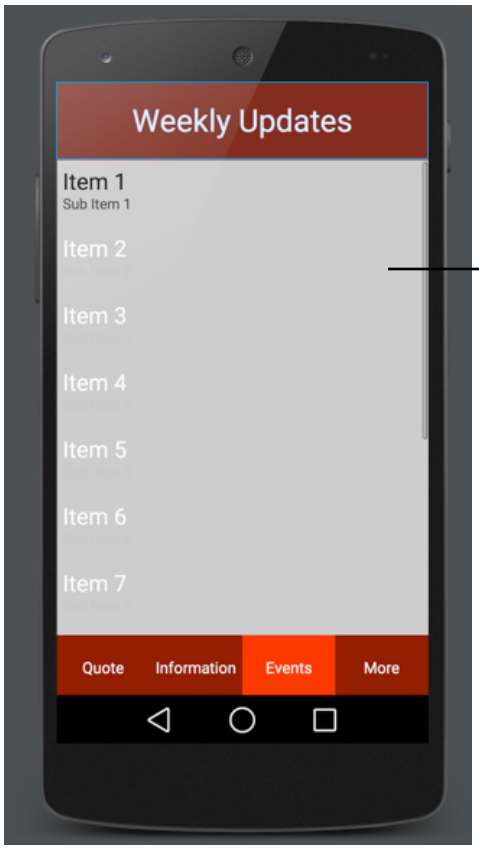
Before After

SQL commands were enabled so that users can add records to the database. Before, the users were only able to view/delete records in the database.



For less experienced users that are not familiar with SQL, a visual GUI with textboxes was enabled on phpMyAdmin.

Test plan #2, test item #10 – XML ListView adapter designed for custom news formatting



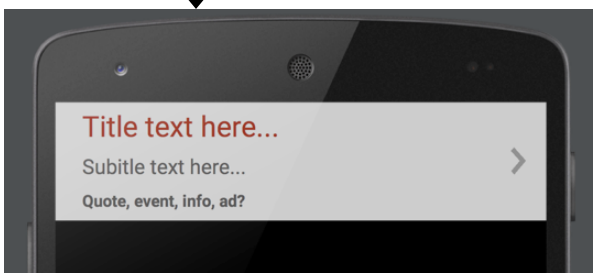
Default android theme/design when displaying items in a list. Not much design.

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/ap
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/row_select">

    <TextView
        android:id="@+id/rowTitle"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Title text here..."
        android:textColor="@color/rowTitle"
        android:textSize="21sp"
        android:paddingLeft="20dp"
        android:paddingStart="20dp"
        android:paddingRight="20dp"
        android:paddingEnd="20dp"
        android:paddingTop="6dp"
        android:paddingBottom="8dp"/>

    <TextView
        android:id="@+id/rowSubtitle"
        android:layout_below="@+id/rowTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Subitle text here..."
        android:textColor="@color/rowSubtitle"
        android:textSize="15sp"
        android:paddingLeft="20dp"
        android:paddingStart="20dp"
        android:paddingBottom="9dp"
        android:paddingRight="40dp"
        android:paddingEnd="40dp"
        android:maxLines="1"
        android:ellipsize="end"/>
```

Custom XML class was created to define a list item element in an Android ListView. The new list theme is displayed below:



Final corrected test plan

All of the errors in the previous test plans were fixed. The application was tested again.

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
Enter screen works as planned	1. Click app icon launcher 2. Click "Enter" button for new screen	White box testing of launching the application and clicking Enter button	1. Application should launch to title page "Ontario Weekly Al-Akhbar" 2. Enter button should advance to news list screen	- app launches with the logo - enter button switches	- flawless	Launching/ Enter screen
Internet connection dialog works	1. Launch application process without internet connection, and delete any files in SD. 2. Launch application process without internet, keeping existing files. 3. Launch application process with any type of Internet connection (WiFi/ 3G)	White box testing of internet connection for application	1. If no internet connection and no database file, network settings dialog should appear 2. If no internet connection and database file exists, move to next screen 3. If internet connection, download new database file and move to next screen	- network dialog appears if no file or internet - offline connection works - downloaded file saved through internet connection	- works perfect	Launching/ Enter screen
Tabbed screens work as planned	1. Click all of the tabs for the different news feed categories 2. Check if all of the buttons work on EACH different tab 3. Click back button to see if process switches or remains the same	White box testing of the tab-buttons and their screens in the application	1. All of the buttons should take the users to the corresponding news category. 2. The tabs should respond to the user click in a fast manor. 3. Content displayed should switch, different list of information 4. Current page's tab should be in bright red (aid the user)	- each button goes to correct page - tabs show categorized info - current tab is bright red	- 100% effective	Quote, Information, Events, and More screens

Test 3

Tested by: Katherine Lu

Feb 17/15

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
ListView elements work as planned	1. Click the listed items in the "Information", "Events", and "More" screens 2. Scroll through the content of the new screen opened 3. Click the different list elements in the different tabs 4. Scroll through items to make sure all items are accessible.	White box testing of ListView elements	1. The corresponding list item should expand its details to another screen containing the body. 2. The relating title should be carried forward to the next screen 3. No duplicate list items in multiple tabs, everything should be categorized	- every news activity has a clickable list item - info on list is carried to new screen with more info - unique entries	- works great	Information, Events, and More screens
Expanded news screen should display all details	1. Click a news item, it should open up a new screen with all the details 2. Click the back button	White box testing of ListView/expanded screen buttons.	1. A new screen should be launched containing all of the details 2. When the back button is pressed, the user should continue with the last activity (last tab screen)	- back button takes user to exact state	- flawless	Expanded news screen
Database entries should be user friendly	1. Enter data into the phpMyAdmin text boxes 2. Enter data through SQL queries (if you can) 3. Enter different types of information for each field 4. Enter a different tag into the system (anything other than 'i', 'q', 'e')	Black box testing of database entries, and SQL statements	1. All of the data should construct a unique record into the database 2. Errors should occur if the data entered is out of bounds, or is not entered correctly. 3. If everything is entered right, there should not be any formatting errors in the application 4. If the news items are tagged with something other than 'i', 'e', or 'q', it will not appear on any of the tabbed news lists. 5. Through SQL, the user should be able to create a whole database, tables, fields, and enter records.	- all variables and fields, construct one record - data validation, error is given if data does not match - only option is i, e, q to categorize ~~into~~ news - SQL option	- designed well	GoDaddy phpMyAdmin database

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref In Product
HTML tags should format the text	1. Place HTML tags into the database when entering the news details into phpMyAdmin 2. Use different HTML tags for different styles 3. Design the title, subtitle, and body	Black box testing of the database entries, and HTML tags.	1. Not all of the tags will work. Only about 50% percent are parsed through Java. 2. Important information should be bolded, underlined, or italicized. 3. Every news item should have unique formatting based on the right HTML tags	- tags transform content into what the user wants - title, subtitle, category bold - HTML tags format info individually	- great	GoDaddy phpMyAdmin database, expanded news screen
Hyperlinks all lead to the right location	1. Click the hyperlinks underlined in blue 2. Browse through the link that opens in the default internet browser 3. Click device return button	White box testing of all the hyperlinks leading to external pages in the application.	1. When the user clicks the link, they should be taken to their default browser 2. The appropriate content should load up 3. When the return button on the device is pressed, the user should be taken back to the application	- hyperlinks work, take user to right page - return button takes user back	- perfect	Expanded news screen, internet browser
Special characters should be parsed by XML/Java	1. Enter any special characters with their unicode numbers into the database 2. Enter any Java special modifiers into the database	Black box testing of the database entries, and XML/Java special character parsing	1. The unicode number should translate to the special character to display when reading the content (like bullet points)	- bullet points appear through unicode	- works well	Expanded news screen
XML custom adapter should work well	1. Enter all sorts of values into the title field in the phpMyAdmin database 2. Enter HTML tags into the field	Black box testing of the XML ListView adapter	1. No matter how long the title, there should be 3 dots at the end to indicate more information 2. No HTML modifier applied to title, so noting should happen with HTML tags 3. The theme should remain the same, nothing can be modified from the outside. 4. If the body is long, then the screen should begin to scroll	- 3 dots shown if sentence longer than screen length - scroll enabled - theme remains same: red	- great	ListView list in all tabs

Final specifications outline

Hence, all of the specifications were tested by the client, and the application was approved.

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
Specification Validation	Specification 1): Create Android application to display weekly news updates, event information, and Quote of the Week from Ismaili Institution Canada (IIC).	Java + XML code, tabbed activities/intents, SD storage, internet connection, multiple Java classes (downloader, screens, and thread) working together, onClickListener for user interaction	- Android app opens up - Home button is pressed to update device storage - Updates, events, and quotes are displayed	- Functions work correctly	- Tested by client on February 28th	Whole application
	Specification 2): Application must be user friendly	HD themes/graphics for all types of devices (LDPI, MDPI, HDPI), home page, buttons, tabbed activities/intents, ListView, Screen messages (Dialog boxes), AsyncTask threads (background process), click-to-view based list items, WiFi/3G network dialog, If statements	- User is presented with "Enter" button - Tabbed activities are presented to the user at the bottom, all information is appropriately organized - Download process in background - If no connection and no database file on the device, user asked to connect to network. - List of scrollable items, when clicked, the specific item is then viewed with detail - When back button pressed, activity starts where user left off (saves last scrolled position). - List items have a title and subtitle describing the content of that item, arrow leads to more information - Tabs that the user currently views will be bright red	- Functions work correctly	- Tested by client on February 28th	Home screen, tabbed screens

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
	Specification 3): Application should be visually appealing, with a logo.	HD graphics based on screen, Custom ListView adapters, launching icon design, consistent color scheme, Roboto (sans-serif) font for easy reading,	- Logo helps to identify and find the app quickly in the app drawer - Titles, subtitles, and body have different levels of styling - Red color scheme used for app, with white font for easy readability. - Font is Roboto (Sans-serif) - ListView items have custom designs, the default Android theme is not used. Makes the application look professional and custom-built. - No images/graphics are pixelated, all are HD	- Functions work correctly	- Tested by client on February 28th	Whole application
	Specification 4): Application should provide important contact information for IIC.	XML hard-coded string, "More Details" tab	- A special tab labelled "More" has additional details about the IIC. - Provides contact information	- Functions work correctly	- Tested by client on February 28th	More tab

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
	Specification 5): Create a server-end application to allow committee members to input information into the database and update the application.	URL parsing, active network connection, offline database in device, buffer streams when downloading through HTTP request, reading/writing device storage, parsing downloaded JSON for content, Hashmap key value pair for items in each tab, updating XML attributes to display fetched data, HTML tag parsing for styling, online GoDaddy hosted domain for JSON file, SQL based online database	- Application should connect to *"zainzulfiqar.com/thisweek.json"* and download this file containing the database content for the week. - File should be saved on local storage of device, to be used if device is offline. First time connection of device will always require internet connection to download file - File should be read and the JSON formatting should be parsed into readable Java string. - Specific parts of the JSON are parsed (sections appropriate for the type of tab the information is fit for) and displayed in the respective tab - Each list item in ListView has a unique key ID linking the list item (only containing the title and one-sentence information) to the full body information for the expanded view	- Functions work correctly	- Tested by client on February 28th	Whole app, populating ListView items

Test Item	Test Data	Part of System Tested	Expected Outcome	Actual Outcomes	Comments	Ref in Product
	Specification 6): Database entries are user friendly, and organize content appropriately.	Online secure database to read information (no one else can modify), If statements to filter out data for each tab, Loops to populate multiple items in list,	- IIC committee members can use SQL statements to enter information into the online database if they are experienced users, or they can use the GUI provided by CPANEL/phpMyAdmin to insert the information. - Each record in the database has a Key ID - Each record has a category ID, linking it to the quote, event, or information item. They are populated in the respective app tab - Information is read publicly, but can only be modified by the admin of the database - GoDaddy account. - Data from the online database is converted to JSON through SQL or GUI. - JSON file is hosted on the GoDaddy domain for app to download.	- Functions work correctly	- Tested by client on February 28th	Internet connection, online GoDaddy database

Client approval of specifications: M.K. Bhimani

Agreement of Client

I confirm that the Android application specifications were appropriately designed for the set requirements, and that the planned design of the application fully implements the discussed vision of the product.

H.K. Bhimani

Client signature

Hamida Bhimani

02/22/2015