# Worklog for Currency Trader News Alert App - Gordon Leary (10352606)

27/May/2018 Setting phone to vibrate on / off

2 hrs:

28/May/2018 Getting alarm/ringtones to play on / off

1hr: (small sample program Worklog for Currency Trader News Alert App - Gordon Leary

(10352606)

27/May/2018 Setting phone to vibrate on / off

2 hrs:

28/May/2018 Getting alarm/ringtones to play on / off

1hr: (small sample program with buttons(click events) to control vibrate & play alarms

(on/off)

29/May/2018 Research Alert Notifications

4 hrs https://docs.microsoft.com/en-us/xamarin/android/app-

fundamentals/notifications/local-notifications

NotificationsCOMPAT (compatibility version - included in AppCompat V4)

Broadcast / Receivers / Alarm Manager.....'this' to 'context' issues

30/May/2018 XML & LINQ research

2hrs (Deitel, Deital, 2014. Visual C# 2012 How to Program 5th ed. H. Marcia et al., eds.,

Harlow: Pearson Education.)

31/May/2018 Meeting with Dhoot

31/May/2018 XML & LINQ research

2hrs

1/June/2018 XML & LINQ research

2hrs

2/June/2018 Proof of concept app console app for XML & LINQ

2hrs

3/June/2018 Proof of concept app console app for XML & LINQ

2hrs

4/June/2018 Udemy Course (by Grant Klimaytys) - UI

2hrs (https://www.udemy.com/xamarin-android-a-master-guide-to-mobile-app-

development/learn/v4/overview)

5/June/2018 Udemy Course - Adapters

2hrs

6/June/2018 Udemy Course - Adapters

2hrs

7/June/2018 Udemy Course – scrolling lists

2hrs

8/June/2018 Udemy Course - Assets / localisation & language conversion / graphics sizing for

2hrs different screen densities

9/June/2018 Udemy Course - Activities & Activity lifecycle

2hrs

12/June/2018 Udemy Course - Animations - changing button state / xml transitions / property

2hrs animations /

13/June/2018 Udemy Course - Android file system / app data folder / writing to external storage /

2hrs threading on UI & non UI threads (Task.Factory.StartNew)

15/June/2018 Udemy Course - Alerts / Notifications / Toasts

2hrs

16/June/2018 Udemy Course - Themes / Material design / appComat

2hrs

17/June/2018 Udemy Course - Action Bar & ToolBar

2hrs Set up GITHub for project & backup - https://github.com/gordIRL/AndroidCollegeProject

18/June/2018 Began work on actual prototype project.

2hrs Added splash screen using appCompat, wouldn't run until MainActivity had an

ammCompat theme activity associated with it.

Emulator wouldn't run, running short on memory, had to do a factory reset from within

VisualStudio (android emulator manager)

Emailed update to Dhoot.

19/June/2018 Toolbars / 2 toolbars in one activity / appCompat Toolbar (4hrs)

5 1/2 hrs Research: Themes / Material Theme / appCompat themes (1 1/5 hrs)

20/June/2018 Added 2 toolbars to App / set up various appCompat styling themes / added 'change

5 hrs button state' via xml simple list view added between toolbar at top of screen and 2nd toolbar at bottom of screen (3 hrs)

Tried to add Proof of concept xml to linq code to app. Didn't work.

Needed to add a reference: System.Xml.Linq. (1hr !!!!)

Added xml data file to Assets folder / to read file used:

XDocument xmlFile = XDocument.Load(Assets.Open("ff\_calendar\_thisweek.xml"));

(30 mins)

several loops to add LINQ sorted data to a listView (30 mins)

21/June/2018 3 x Linq to XML test options added to DataAccess Class & wired into menu options on

4 1/2 hrs toolBar (lower) on MainActivity.

Workaround for NotifyDataSetChanged not working.

22/June/2018 Context(?) menu (Alert using builder) added to ItemClick –

4 hrs uses array in Resources/Values/Strings - calls next activity & passes data across(string)

via a method contained in the 2nd/called activity. (2hrs)

Wiring up toolbar(top) (30 mins)

Added Set Person Alarm activity - from ProofOfConcept App (1hr)

DatePicker & Time Picker research on MSDN (30mins)

23/June/2018 DatePicker & TimePicker MSDN tutorials & code experimentation

4 hrs - proof of concept app.

24/June/2018 Continued from yesterday. Challenge to get the DateTime object from DialogFrament

4 hrs used in MSDN code. Kept getting a 'need object reference error'.

Initially solved by changing a textView control to 'static' which i didn't think was correct, despite it working. Eventually solved correctly(?) by adding a 2nd 'FindViewById' for the textview (1st one is in onCreate()) inside the fragment code.

The fragment code contains a static method that was the root of my scope issue.(4 hrs)

25/June/2018 Interim report, planning, brainstorming & research. (3 1/2hrs)

5 hrs Added DatePicker & TimePicker Proof of concept app to main project. (1 1/2 hrs)

26/June/2018 Localisation - button.text names and textView.text added to Strings.xml (1 hr)

Report writing & research (2hrs)

Diagram of activities & menus (draw.io website)

27/June/2018 Added ‘Data Updated date’ to Main Activity (30 mins)

Document formatting (30 mins)

7/July/2018 Interim Report

8/July/2018 Interim Report

9/July/2018 InterimReport

Sqlite – Research

10/July/2018 SQLite

2 hrs

11/July/2018 SQLite – converted data retrieval from xml to database

2 hrs

12/July/2018 SQLite – more database work – fixed LINQ query error that returned no results because

3 hrs some fields had a \n character added to the end!! (2 hrs)

Convert 2 separate strings (string containing a date & a string containing a time) into a dateTime object using

DateTime.Parse(my2CombinedStrings); (1 hr)

13/July/2018 Research – ‘Clean Code’ by Uncle Bob (1 hr)

7 ½ hrs Refactored dateTime conversion from yesterday into a separate standalone method

(30 mins).

‘Download/update XML’ option fleshed out. Put into ‘Data Access’ folder

(previously in Main Activity) (1 hr).

Alert Dialog – major issues trying to get a checkbox type list working – settled on using the alert dialog. All online reference material & tutorials were in JAVA, tried to convert some of these – limited success!! (rest of the day and evening – 5 hrs!!!)

14/July/2018 Found & did tutorial on Alert Dialog that worked!! (1 hr)

5 hrs Proof of concept app (on the above)(1hr)

Implemented proof app into main application (1 ½ hr)

Started work on LINQ query from Alert Dialog checkbox results (1 ½ hr)

15/July/2018 Alert Dialog for Market Impacts Selected (1 ½ hr)

5 ½ hrs Alert Dialog for Currencies Selected (1 ½ hr)

* Selected checkboxes stay selected when user leaves & returns to alert.

Refactoring (30 mins)

Continued Alert Dialogs & bottom toolbar & localisation (2 hrs)

16/July/2018 Fixed problem: (3hrs)

5hrs App crashed when deployed to real phone – later discovered it crashed on phone & simulator when running app for the 1st time (deleted app each time to be sure). Method was trying to access database before table was set up.

Added a create table method - which is called in Main Activity, SQLite won’t create a new table if one already exists – solved the problem.

* Need validation everywhere to check if null tables…. ?

Updated list if currencies (Inc. ‘ALL’) & Market events (Inc. ‘Holiday’)

Sample Data & Sample Data LINQ query (to demonstrate use of LINQ/ XML – i.e. everything is done straight from xml file in Assets folder) (1 hr)

Url for XML download is now stored in its own table in database & retrieved from db when needed (1hr)

17/July/2018 Shared Preferences – simple test added to Main Activity.

3 ½ hrs Then added to a separate class – needed have the Context of the calling activity to work,

Can’t use a static class or method – class constructor call where ‘this’ is the context:

new MySharedPreferencesMethods(this); (2hrs)

Date of xml update / download is now stored using Shared Preferences

(1 ½ hrs)

18/July/18 LINQ\_SortAllByUserSelection()

6hrs now sorts data by date & then by time (strings) – to be changed to sort by DateTime object (when possible) (1hr)

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

All work to date has involved getting xml download into database -> retrieving data from db -> putting it into string to display. That’s ok, but I will need to have access to objects when I am setting the alert, ie dateTime & name fields etc.

Re-did whole application – 8 methods or so – to work with news-Objects & lists of

news-Objects – for both input & output – took all day!!!

(5 hrs !!!!) (1 in am, 2 in afternoon, 2 in pm)

19/July/18 Completed yesterday’s update, did some refactoring & committed all to GitHub

4hrs New method: RefreshTxtDataLastUpdated() (1 hr)

Added ‘Custom Adpater’ experiment to TestActivity – preparation for adding to MainActivity (3hrs)

20/July/18 Fixed bug in Select / DeSelect All Currencies option – wasn’t updating display.

5 ½ hrs Finished ‘experimental’ Custom Adapter – now ready to be used in Main Activity. (1 hr)

Changed all dateTime in database & in methods from using strings to using a ‘long’ number which stores the number of ticks for a DateTime object. A date(string) and time(string) are still read from the XML file, but are immediately converted to ticks & that is what is stored in the database, after which everything is based on the tick count.

Also spend some time working out display formats for date & time. (approx. 4 ½ hrs)

21/July/18 What a rollercoaster day!!!!

5 hrs App wouldn’t run on my phone, discovered it still ran on laptop emulator.

Only difference was the language setting, English(Ireland) vs English(American). This

caused DateTime.Parse(dateAndTimeString) to fail in method

ConvertString\_s\_ToDateTimeObject.

(This takes a string & returns a DateTime object.)

Solution:

DateTime.Parse(dateAndTimeString, new CultureInfo("en-US"));

(5 hrs – including lots of refactoring)

-------

-------

27/July/18 Recycle-View & Card-View

5 ½ hrs MSDN tutorial on both, reading, research, typing in sample code.

Added sample code to project app (test activity).

Next - used Xamarin ‘recycle adapter’ boiler plate code & incorporated code from tutorials.

Changed data set to my own news-Object list – ready to implement fully into project.

28/July18 2 ½ & 1 ½ + 2 = 6 hrs

6 hrs Research on recycle adapter & NotifyDataSetChange() (this wouldn’t work…. Ages!).

My method had returned the correct data, but it seems to have gone to a different memory reference. (see website guide below).

Main Activity is now fully converted to use Recycle-View & Card-View.

--------------------------------------------------------------------

List<NewsObject> tempDisplayListOBJECT = SetUpData.TestXMLDataFromAssetsFile(xmlTestFile2);

foreach (var item in tempDisplayListOBJECT)

{

DisplayListOBJECT.Add(item);

}

mAdapter.NotifyDataSetChanged();

--------------------------------------------------------------------

Notifying the Adapter (<https://guides.codepath.com/android/using-the-recyclerview>)

Unlike ListView, there is no way to add or remove items directly through the RecyclerView adapter. You need to make changes to the data source directly and notify the adapter of any changes. Also, whenever adding or removing elements, always make changes to the **existing** list. For instance, reinitializing the list of Contacts such as the following will not affect the adapter, since it has a memory reference to the old list:

*// do not reinitialize an existing reference used by an adapter*

contacts **=** Contact**.**createContactsList**(**5**);**

Instead, you need to act directly on the existing reference:

*// add to the existing list*

contacts**.**addAll**(**Contact**.**createContactsList**(**5**));**

2/Aug/18 Graphics for currency icons – had to download individually & then convert to different

5 ½ hrs mimap sizes (mimap-hdpi/ mdpi / xhdpi / xxhdpi / xxxhdpi)

using Android Asset Studio: <https://romannurik.github.io/AndroidAssetStudio/>

lots of good icons at: https://www.flaticon.com

If for some reason the Build Action(in properties) is not set to Android Resource, then the files will be excluded from the APK, and any attempt to load or access the resources will result in a run-time error and the application will crash.

App wouldn't compile when graphics (png) started with digits - looks like they must start with a letter (character).

3/Aug/18

Graphic for splash screen – used Paint3D (windows 10) – png

– transparent – 200 width x 333 pxls

<https://www.colorhexa.com/0070bf> (for working out complementary colour schemes)