

Currency Trading News Alert System  
Interim Report

Word Count: (1547)

Higher Diploma in Science in Computing

(Software Development)

Gordon Leary (10352606)

10352606@mydbsi.ie

09/07/2018

# Contents

[Contents 1](#_Toc518898543)

[1. Introduction 2](#_Toc518898544)

[Acknowledgements 2](#_Toc518898545)

[Aim 2](#_Toc518898546)

[Approach 2](#_Toc518898547)

[2. Background 2](#_Toc518898548)

[3. Specification & Design 3](#_Toc518898549)

[Tools & technologies used 3](#_Toc518898550)

[Project Scope and Objectives 3](#_Toc518898551)

[Design Approach 3](#_Toc518898552)

[Data 4](#_Toc518898553)

[4. Project Testing and Evaluation 4](#_Toc518898554)

[5. Demonstration of Progress 5](#_Toc518898555)

[Status in terms of proposed goals and project plan 5](#_Toc518898556)

[Milestones Achieved 5](#_Toc518898557)

[Revised Milestones to be achieved – ‘to do’ list 6](#_Toc518898558)

[6. Future Work 7](#_Toc518898559)

[7. References / Bibliography 8](#_Toc518898560)

[8. Appendices 9](#_Toc518898561)

[Appendix 1 – Design specification 9](#_Toc518898562)

[Activities and Menus 9](#_Toc518898563)

[Schema for Currency Alert System Database 10](#_Toc518898564)

[Class Diagram 10](#_Toc518898565)

[Appendix 2 – Sample XML code 11](#_Toc518898566)

# Introduction

## Acknowledgements

I would like to take this opportunity to thank my supervisor Harnik Dhoot for his ongoing support during the development of this project. He has been very helpful, insightful and generous with his time and I have learnt a great deal from working with him.

## Aim

To develop a mobile phone application that will alert a trader in the currency market of upcoming fundamental news announcements. The user will be able to download an xml file of all these news release times from a 3rd party broker website and filter these by currency and market impact, save selected events to a watch list, and be alerted by alarm and notification on their android device when such an announcement is imminent. The user will also have the facility to set ‘personal’ alerts.

## Approach

I am using an ‘MoSCoW’, Agile type approach to development due to its fixed deadline. Aiming to keep the scope of the project small and tight, working towards my goals and objectives, expanding on these, time permitting, from a list of options in the future work section.

I am doing regular backups and commits of all code and documents to GitHub. I am also doing a lot of research on Xamarin development, utilising various tutorials including those available on Microsoft’s MSDN channel.

# Background

Currency traders take positions in the market with a view to making a profit by buying a currency pair at a given price and then later selling it at a higher price, the reverse, ‘shorting’, is also possible.

Each week economic data is announced at specific, scheduled times. Most dealing brokerages publicise the times of these news events at the beginning of each week, and then the results as they are released.

These news announcements have the potential to move the market significantly against the position held by traders, causing larger than expected losses. Many shorter-term traders stay out of the market completely at these times, deliberately avoiding any possible negative volatility.

Although there are many tools to automate the trading process, many short-term traders who open and close multiples trades in the same day, often do so manually, keeping note of any news announcements and exiting their trades beforehand.

# Specification & Design

All specification & design diagram are in appendix 1.

## Tools & technologies used

* Xamarin.Android
* Visual Studio 2017 (for development & unit testing)
* C#
* LINQ to XML
* SQLite-PCL
* GitHub
* www.Draw.IO & www.dbdesigner.net(for diagrams)

## 

## Project Scope and Objectives

Minimum Viable Product:

* To download a list of all available scheduled news events for the current week from a third-party website (sample xml file included in appendix 2).
* To be able to sort this list by both currency and expected impact on the market.
* To let the user, select items of interest from the above and store them in a watch list.
* To set alarms and notifications on the user’s mobile device based on their watch list, to alert them of an imminent news announcement release.
* User can add personal alerts to the watch list to be alerted at times not in the XML schedule.
* User can select amount of time to be notified before the news announcement, e.g. 10 or 15 minutes before the news release.

## Design Approach

I developed a basic prototype which has helped to clarify design decisions and gain insight into how a potential user might use the application.

I have also developed several small ‘proof of concept’ apps, which I have then integrated into the main project, including;

* + Phone ringing and vibrating.
  + Alarm manager – to wake up an activity at a future point in time.
  + XML to LINQ – to consume and filter data from XML file.
  + Toolbars – to use multiple toolbars on same activity using ‘appcompat’ for backward compatibility.

## Data

I plan to use a persistent data layer using SQLite-PCL, schema outlined below. The application will load existing data upon start-up. Any additions to the user’s alert list will be written immediately to the database, so as in the event of an unexpected shut down data will already be saved, e.g. device runs out of battery power. URL for downloading xml data file will also be stored in the database, thus providing a location for multiple URLs if required in future development.

As data is only relevant for the current week, all data will be purged, including database, when the user downloads the next xml file of weekly news announcements, published every Sunday night.

# Project Testing and Evaluation

As outlined in my objectives, my initial priorities have been to develop a prototype to decide upon the user interface, and to develop several, small proof of concept applications to learn and get a greater understanding of the technologies I’ll be implementing in the project. With these milestones achieved I will now integrate testing into my development as follows;

* Manual testing for UI elements (collect data & manually enter into Excel spreadsheet).
* Unit tests for non-UI code, e.g. filtering of xml data in the view all currency events section.

I would prefer to be using test driven development, TDD, but would need to implement the system using an MVC or MVVC type pattern to achieve this, beyond the current scope of this project, but this has been included as a possible consideration for future work.

# Demonstration of Progress

* Log of all work done to date (accompanying text file).
* Main Application (Visual Studio – zip file)
* Proof of Concept Apps (Visual Studio – zip file)

## Status in terms of proposed goals and project plan

I feel that I am making timely progress in the development of my application. Milestones 1 to 5 detailed below have been achieved. Following an Agile-type methodology, producing a prototype on which to base further development, has helped to clarify my approach, especially the layout and design of the app.

I have completed an extra ‘Xamarin.Android’ course on Udemey.com and worked through several tutorials on Microsoft’s developer network, which has given me more options in terms of screen real-estate management, menus, toolbars etc.

* Using ‘AppCompat’ toolbar, for backwards compatibility with older mobile devices, also allowing me to have two toolbars on each activity screen, a global navigation menu, and a context driven one.
* Alert notification manager for alerts based on Item-Click events in a list of data.
* Changing button-state (on-Click color using xml).

## Milestones Achieved

1. Small proof of concept app – to demonstrate a working example of calling an alarm or notification at a pre-scheduled time, e.g. setting mobile phone to ring, vibrate and display an on-screen message at a set time.
2. Console app to demonstrate retrieval and manipulation of the 3rd party XML news event list, e.g. convert XML data to list & sort by currency, date or impact.
3. A non-functional mock-up / prototype to;
   1. display design and flow of app from one screen to another,
   2. use of menus, graphics and theme templates,
   3. hard-coded sample data.
4. Prototype produced with basic functionality.
5. Wire-up Date-Picker & Time-Picker ‘proof of concept’ app into Personal Alerts.

## Revised Milestones to be achieved – ‘to do’ list

* List of Personal alerts (watch-list):

Display WatchList

Add Personal Alert

Edit / Delete Personal Alert

* Download xml file from ForexFactory.com, i.e. replace current hard coded sample data.
* Display all data (formatted)
* Sorting data:
* By currency – e.g. Yen, Dollar, CAD, GBP (need currency icons)
* By impact – high, medium, low
* By both, in any combination – checkbox menu
* Add onItemClick() event to currency announcements – to add alert to an item (add it to watch list)
* SQL-PCL:
  + - Research 🡪 Proof of concept app 🡪Integrate into app.
    - On app start, in SpashActivity -Load data from SQL-PCL database.
    - On app close, write out, save, data to SQL-PCL database.
* Theme & UI polishing / refinement
  + - Test on different emulated phones (set up other emulators on my development machine).

# Future Work

Below are listed items for possible future development which I believe would be valuable additions to the application time permitting.

Listed in order of preference;

* Localization.
  + Converting the app to another language, e.g. French, would mean that all strings and text are in separate xml files, ensuring better design and separation of concerns, making it easier to translate the application to other languages and possibly gaining more potential users in other territories.
* Utilizing the ‘ReCycler’ view:
  + Not required yet, due to the small amounts of data in each list, but this would enhance the performance of the app if the size of data was to grow.
* Automatically add new data sources and verify their content:
  + The app relies on one data source, ForexFactory.com for its XML data (news announcement schedule). While this is a reliable and well-established resource some redundancy would be desirable. A web service could be developed and deployed to Azure. The app could call an API to enable downloading from multiple sources and to verify that their content is current.
* Import, export .csv file of ‘watch-list’ events.
* Landscape and portrait modes for different devices.
* Integrate application with Google Calendar.
* Tracker facility for the user to be able to keep a record of their market trades & forward reports to a selected email.
* Convert application to a Model, View, Controller (MVC), or Model View, View Model (MVVP) type structure, to have greater separation of concerns and to enable more comprehensive testing of the application.

# References / Bibliography

Deitel, Deital, 2014. *Visual C# 2012 How to Program* 5th ed.  H. Marcia et al., eds., Harlow: Pearson Education.

Leibowitz, M., 2015. *Xamarin Mobile Development for Android Cookbook* 1st ed. A. Banerjee et al, eds. Birmingham: Packt Publishing Ltd.

Albahari, Albahari, 2016. *C# 6.0 Pocket Reference* 1st ed. B MacDonald et al., eds., California: O’Reilly Media Inc.

Sommerville, I., 2015. *Software Engineering* 10th ed. H. Marcia et al., eds., Harlow: Pearson Education.

Murphy, Mark L., 2018. *The Busy Coder's Guide to Android Development* 8.10th ed. Commons Ware

Xamarin University. Available at: <https://university.xamarin.com> [Accessed May 2018].

Xamarin Android Tutorials by Joe Rock. Available at: <https://www.youtube.com/channel/UCFH-Tn6ggt9yLWJ4V9NizZA> [Accessed May 2018].

C# Training Tutorials. Available at :<https://www.youtube.com/user/IAmTimCorey> [Accessed May 2018].

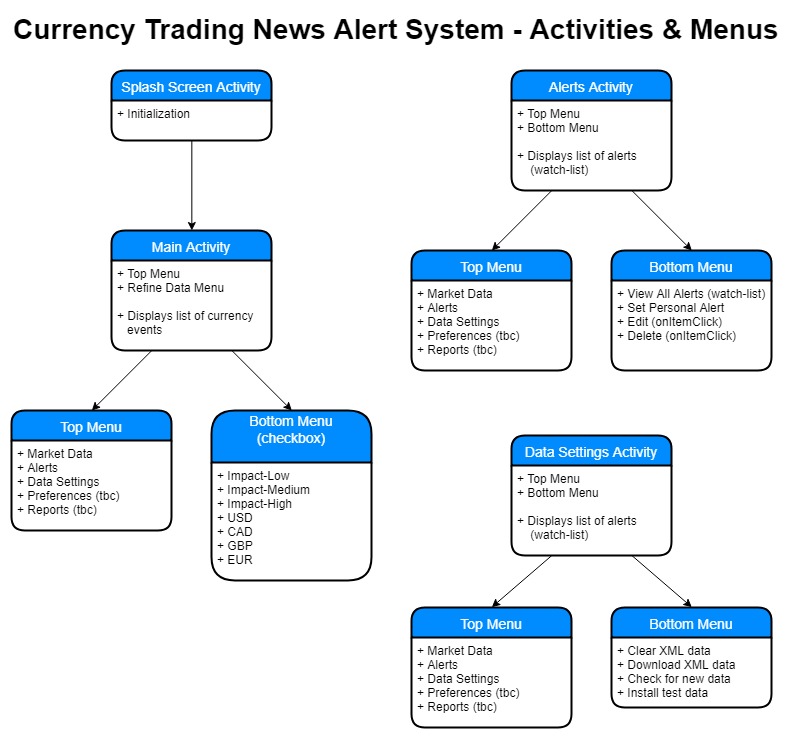
Xamarin Documentation: Available at: <https://docs.microsoft.com/en-us/xamarin/#pivot=platforms&panel=Android> [Accessed May 2018].

Xamarin Android Udemy Course by Grant Klimartys. Available at: <https://www.udemy.com/xamarin-android-a-master-guide-to-mobile-app-development/learn/v4/overview> [Accessed May-June 2018]

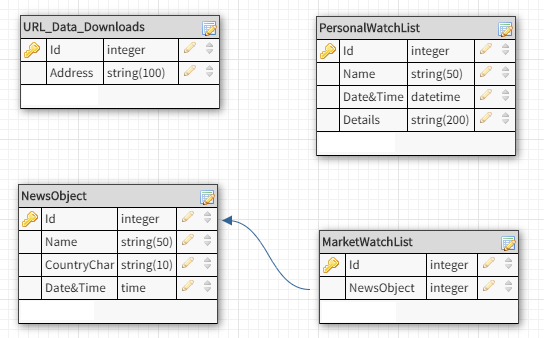
# Appendices

## Appendix 1 – Design specification

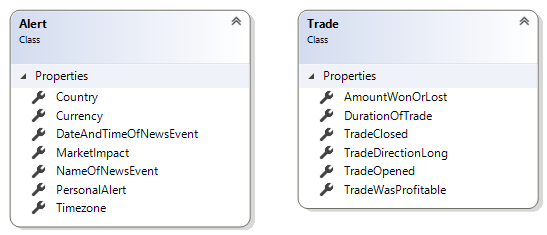
### Activities and Menus



### Schema for Currency Alert System Database



### Class Diagram



Nb. Trade class is only for use in future possible development.

## Appendix 2 – Sample XML code

Sample XML of fundamental news releases published weekly. Available at: https://www.forexfactory.com/ffcal\_week\_this.xml

<?xml version="1.0" encoding="windows-1252"?>

<weeklyevents>

<event>

<title>FPI m/m</title>

<country>NZD</country>

<date><![CDATA[04-15-2018]]></date>

<time><![CDATA[10:45pm]]></time>

<impact><![CDATA[Low]]></impact>

<forecast />

<previous><![CDATA[-0.5%]]></previous>

</event>

<event>

<title>Rightmove HPI m/m</title>

<country>GBP</country>

<date><![CDATA[04-15-2018]]></date>

<time><![CDATA[11:01pm]]></time>

<impact><![CDATA[Low]]></impact>

<forecast />

<previous><![CDATA[1.5%]]></previous>

</event>

<event>

<title>WPI m/m</title>

<country>EUR</country>

<date><![CDATA[04-16-2018]]></date>

<time><![CDATA[5:52am]]></time>

<impact><![CDATA[Low]]></impact>

<forecast><![CDATA[0.4%]]></forecast>

<previous><![CDATA[-0.3%]]></previous>

</event>

</weeklyevents>