Project Research Document – UEFA Ranking System

X00090749, Daniel Fulham

October 1st, 2014

Contents

[Detailed Discussion 2](#_Toc400110170)

[Existing Applications 3](#_Toc400110171)

[5 Year Ranking of the UEFA(Android) 4](#_Toc400110172)

[UEFA Ranking (Windows phone) 6](#_Toc400110173)

[Platform, Technologies and Libraries 7](#_Toc400110174)

[Android 7](#_Toc400110175)

[Web API 2 8](#_Toc400110176)

[SQLite 8](#_Toc400110177)

[Azure Mobile Services 9](#_Toc400110178)

[C# 9](#_Toc400110179)

[Windows Phone 10](#_Toc400110180)

[IOS 11](#_Toc400110181)

[Potential Risks 12](#_Toc400110182)

[Bibliography 13](#_Toc400110183)

# Detailed Discussion

The proposed project is the development of an Android Application that will function as a UEFA Ranking System utility.

The Application will be developed to do the following:

Web API – Pull down League Tables from an official Uefa affiliated Site. This will ensure up to date data for the Database and the Application to then correctly process the correct information for accurate Rankings.

The Web API will also pull down the UEFA Ranking for each team that are required. The data will be obtained through GET requests within the API. There are several API services for obtaining football information such as OPTA, ESPN, livescoreboard, xmlsoccer that are free services and will be used to retrieve live data for the application to use all necessary data required for its functionality.

There will be many permutations implemented to ensure the ranking is calculated and allocated correctly. This is vital to the system functioning correctly and therefore a very high level of attention to detail whilst programming will need to be performed on a constant basis.

The Application will offer the ability for the User to request a real time update on the current rankings. This will be implemented in the system through the Web API and Database as the latest information will need to be taken from the appropriate League and UEFA ranking page and then processed for an up to date rank.

A week by week ‘main’ update on the latest ranking and standings will be done. The application will have the ability to compare the rankings of that week with previous weeks. The capability of ‘predicting’ final rankings will be implemented by comparing changes in ranking standings over a series of weeks. The user will have the ability to perform these comparisons and see the predictions for themselves.

There will also be a section that will display all the teams in one large ranking system. Each team will be ranked as per their current Uefa Ranking. This will be quite dynamic because teams will change in rank on a weekly basis. A search function will be implemented and be made available for the user to use for easy navigation to find the current ranking position of their desired team.

When the user opens the Application initially, an option will be included to allow them to choose their own team and this will show them their team’s current progress. The Ranking for Qualifying Rounds will be quite broad and easy navigation will be essential for good accessibility for Users trying to track their own team’s progress.

A Country Ranking section will show the Ranking of Countries in Europe. There are several permutations once again that will be required for correct calculation. Country ranking has a direct impact by results of European matches from teams in their Country.

It should be noted the UEFA ranking system is quite complex and due to the several permutations involved it can be quite difficult for anyone trying to manually predict what Qualifying round their team will be in and whether they will be seeded or unseeded in that round. There is a massive gap in the market for such an application and will be of benefit to Fans, statistic reviewers but more importantly the clubs themselves. Considering the money involved for getting through qualifying rounds in Europe is so substantial for the board of a club to know in advance they will be seeded could have a major influence on their club budget.

For the actual Qualifying rounds teams are allocated to one of four rounds of qualifying. This is set by the country ranking. For example Countries ranked 49-54 in the Country rankings will have the club that wins their league go in to the first round of Qualifying.

Those ranked 17-48 will have the club that wins their league enter round 2. This continues until all clubs have been allocated a Round. Once again there are exceptions which will effect what round a club will go. These complex permutations will need to be implemented correctly otherwise the ranking will not be correct.

The Ranking of a team is Calculated over Five Years. The application will correctly calculate the latest ranking for each team based on the Round they get knocked out in. Further complexity will occur for teams in the Group stages as results there will affect the team rank. The application will need to be implemented to support this and give a correct ranking for that team.

The app will be developed with different screens for certain sections. There will be a general template screen that most pages will use as a display however the specific sections such as the Qualifying Rounds/Info section/Country Ranking will vary in size and density to accommodate the level of complexity that is required for the data it needs to display to the user.

The application will be internationalized to broaden the pool of users that can use the application. An advantage of rankings are that they are universal to readers though menu sections, titles will need to be altered in language. The languages for support will be English, French, German and Spanish for initial release.

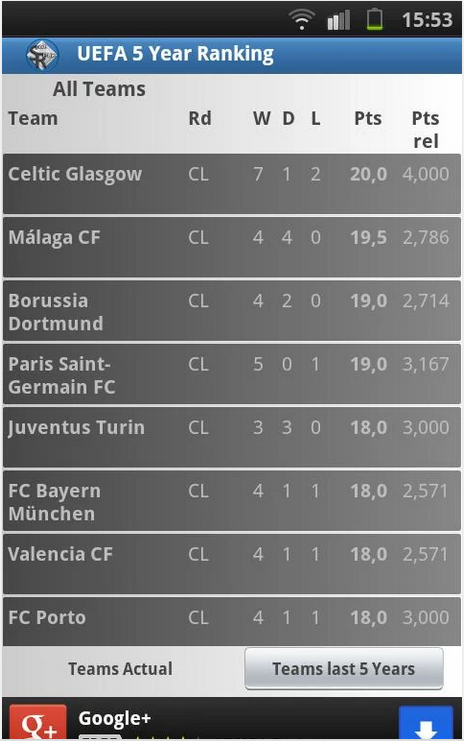
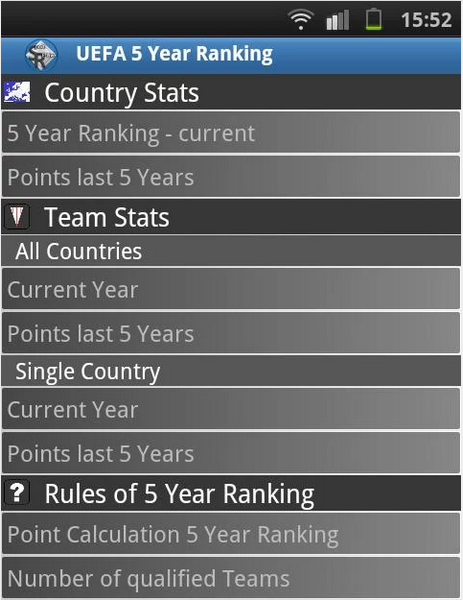
# Existing Applications

Having researched both Android and Windows Store Markets there are some Applications with some relation to the proposed Idea however they are quite limited and do not provide much functionality with regards to the Ranking System.

Features of similar Applications

|  |  |  |  |
| --- | --- | --- | --- |
|  | UEFA Ranking System(Proposed Idea) | UEFA Ranking  (Windows phone) | 5 Year Ranking of the UEFA(Android) |
| Team Ranking(Current) |  |  |  |
| Country Ranking(Current) |  |  |  |
| Team Ranking(Next Year) |  |  |  |
| Country Ranking(Next Year) |  |  |  |
| Information Section for Ranking |  |  |  |
| Search Function |  |  |  |
| Qualifying Rounds |  |  |  |
| Real Time Updates |  |  |  |
| Qualifying Predictions |  |  |  |
| Personal User View |  |  |  |
| Index and Contents section |  |  |  |

## 5 Year Ranking of the UEFA(Android)



(Wahl, 2014)

## UEFA Ranking (Windows phone)

(Patierno, 2012)

# Platform, Technologies and Libraries

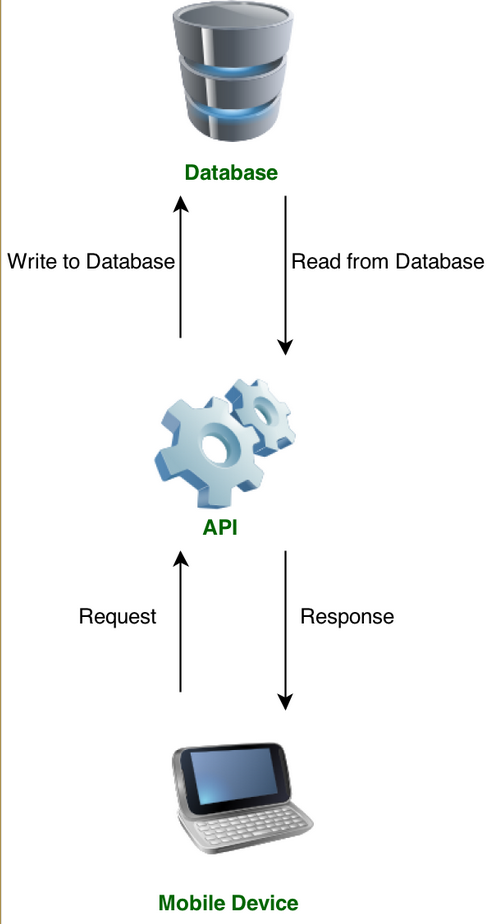
## Android

Android is a Mobile specific Operating System which has grown to become the leading Operating system on the market today. The Android market is currently the largest on the mobile marker. There are one billion android users currently with IOS users hitting half that figure(470 million). Android is a popular choice among users and the development of systems through Android is a preferred choice by many. (Hay Newman, 2014)

An actual android Application is a mobile software application that is developed for use on Android devices powered by Google’s Android platform. The Apps can be developed by anyone and also placed on the Google Play Store to be made available for over one billions devices to download. Apps that are developed can come free of charge to the user or given a premium price to which both the user(70%) and Google(30%) will gain proceeds from.

(Stroud, 2014)

An android application can be broken down in to the Application itself, the API and the database. When the application is functioning it can operate by requesting data from the API to which the API writes/reads from the Database, the API then executes a response back to the application.



(Johnson, 2013)

## Web API 2

Web API is a framework for building web APIs on top of the .NET framework. It has the ability to work with mobile clients such as Android and Windows Phones. Web API 2 has the ability to be called from a Windows 8 phone and Android. On the foundation of things HTTP methods are the minimum required by the developer to perform a GET, POST, PUT and DELETE operation. Web API 2 can also support interactive applications that will allow that application to retrieve any information and data from the Web API Application.

Using Android for the Application for development calling from Web API is quite straight forward. In order to function correcting an operation of simply calling the API and then parsing the results is the basics to what is required. This idea will rely on requiring live data from the internet for essential functions and Web API 2 is very strong to being the catalyst in that regard.

(Wasson, 2014)

## SQLite

SQLite is an Open source database that can support an Android application. It is incredibly efficient considering it only requires approximately 250Kbyte at runtime. It is embedded within every android device providing full compatibility. The database is automatically managed by the Android platform, the developer is only required to define SQL statements for creating and updating the database.

(Vogel, 2014)

## SQL Azure

SQL Azure is the database supplied when using the Azure Mobile Services and other services provided. It allows relational queries against stored data, search, data analysis and data synchronization capabilities. Data transfer is done through an XML-based format.

Krishnaswamy, Jayaram (2010).

## Azure Mobile Services

Azure Mobile Services has the ability to provide a cloud-based backend service to Applications of various operating systems. It provides compatibility not only to Windows phones but also to Android and IOS along with various others. Android Developer tools are required in order to pursue implementing the service. The backend can be implemented through both JavaScript and .Net. The service is offered for free with limitations and the Developer will need to be registered with Microsoft Azure. Azure services tutorials on Microsoft’s site shows the web service is efficient and straight forward in terms of being implemented and working correctly with a Mobile application.

(Azure.microsoft.com, 2014)

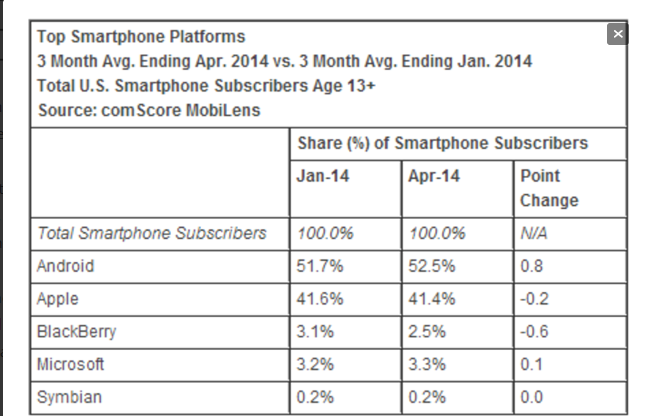
## C#

C# is a programming language that offers the functionality to build native apps for iOS, Android and Windows Phone. It can be implemented through .Net and JavaScript. Many large applications such as the Microsoft Azure Service relies on C# for development. The actual interface is friendly with the majority of mobile development still being able to be implemented through Android Applications(Android Studio/Eclipse) and then the actual C# end of things being implemented through programs such as Visual Studio. C# is a strong force on the mobile market in terms of being used for development.

(Shackles, 2012)

## Windows Phone

Windows Phone is a mobile specific operating system like Android and IOS designed and developed by Microsoft. It is an underdog in the market with Android and IOS users vastly overtaking the total users using a Windows Phone. It should be noted however it is on the rise and Microsoft have developed many things since such as the Azure Mobile Services, introducing better mobile devices and improving the Operating system itself to grow their total users in recent times.

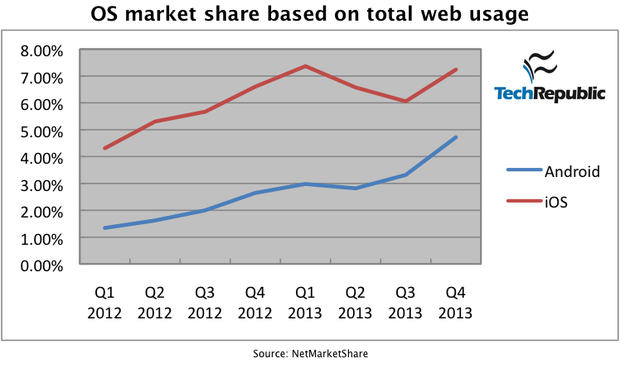


(Surer, 2014)

In terms of choosing the windows phone for Application development the actual market size compared to IOS and Android isn’t great. Windows have come a long way recently and their Azure Mobile Service is excellent however Android is where the market is, when developing an application hitting a large audience is essential.

## IOS

IOS is the Mobile Operating System created, developed and offered by Apple on their IPhones. In order to develop IOS apps an environment of Apple devices are essential. A Mac computer, Xcode and iOS SDK is required for initial development to create any Applications to work the with IOS operating system. Apple’s official site offers a lot of documentation to their Operating System and also provide tutorials to learn how to develop an App of your own. Although there are double the amount of Android devices compared to IOS devices Apple have held their own on the market and the number of users with their phones is increases every year. Half the devices being used compared to IPhone’s the amount of web usage is far exceeded by iOS users compared to Android. This is a very important factor to what platform to choose for App development because with more active iOS users there is actually a bigger market pool of active iOS users.



(Forrest, 2014)

# Potential Risks

There are definitely a number of potential risks that have been identified. The proposed idea will require the use of the Database, Web Service and the Android Application to perform the majority of Functionality so it is essential none of those elements experience any issues.

The most complex functionality will be fully Dynamic in that updates will be required from Websites in order to perform the algorithms for correct calculation of Ranking. The option for the user to execute Real Time updates could also propose many risks and again it is vital the coding is correct and the communication between all elements is correct for the Real Time update to be available at any time.

The Android Application will heavily Rely on the web service and an internet connection in general. Limited functionality being available could result in large limitations and should be addressed so a user could possibly be provided with an offline offering of data to be able to use the majority of the Apps features.

Although the permutations are quite complex the actual programming of these should be done efficiently and correctly to avoid any bugs that could enter the frame due to the coding not being done in the best manner possible.

Upon research of the actual operation of Android Applications memory management and the management of running processes is very important for the Application to work right and for the Phone to run the Application in conjunction with shared services/resources. These will definitely be identified throughout development and any potential issues that may arise will be actioned on immediately.

Security is a massive element to Development of any Application not only on Android devices. The majority of developed Android Applications will go on the Google Play store. There have been breaches to Apps on there with Malware the main threat that these apps have been hit by. It is also important the application is developed to the latest in date features, coding and implementation as a whole. Should an app be developed with outdated resources it can cause a threat to the phone itself due to security updates not being applied to deal with outdated apps since. A compliance policy could be introduced to the Application on release.

(Violino, 2013)

# Bibliography

Wahl, C. (2014). *5 Year Ranking of the UEFA*. Germany: Sports Relive App

<https://play.google.com/store/apps/details?id=fjw.table&hl=en>

Patierno, P. (2012). *Uefa Ranking*. Italy.

<http://www.windowsphone.com/en-ie/store/app/uefa-ranking/3357abc1-84fa-48b0-a38c-1f3edc0da7b4>

Hay Newman, L. (2014). Android Users Won’t Drop Money on Just Any App. [online] Slate Magazine. Available at: <http://www.slate.com/blogs/future_tense/2014/06/26/there_are_twice_as_many_android_users_as_ios_but_ios_users_spend_double.html> [Accessed 3 Oct. 2014].

Stroud, F. (2014). *What is an Android App? Webopedia*. [online] Webopedia.com. Available at: <http://www.webopedia.com/TERM/A/android_app.html> [Accessed 3 Oct. 2014].

Johnson, L. (2013). Android SDK: Making Remote API Calls - Tuts+ Code Tutorial. [online] Code Tuts+. Available at: <http://code.tutsplus.com/tutorials/android-sdk-making-remote-api-calls--mobile-17568> [Accessed 3 Oct. 2014].

Wasson, M. (2014). Getting Started with Web API 2 (C#). [online] The Official Microsoft ASP.NET Site. Available at: <http://www.asp.net/web-api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api> [Accessed 3 Oct. 2014].

Bibliography: Vogel, L. (2014). Android SQLite database and content provider - Tutorial. [online] Vogella.com. Available at: <http://www.vogella.com/tutorials/AndroidSQLite/article.html> [Accessed 3 Oct. 2014].

Azure.microsoft.com, (2014). Get Started with Azure Mobile Services for Android apps. [online] Available at: <http://azure.microsoft.com/en-us/documentation/articles/mobile-services-dotnet-backend-android-get-started/> [Accessed 3 Oct. 2014].

Shackles, G. (2012). Cross Platform Mobile Development in .NET. 1st ed. Sebastopol, CA: O'Reilly & Associates.

Surer, S. (2014). Windows Phone grows US share by 400K new users. [online] WMPoweruser. Available at: <http://wmpoweruser.com/windows-phone-grows-us-share-by-400k-new-users/> [Accessed 3 Oct. 2014].

Forrest, C. (2014). Apple v. Google: The goliath deathmatch by the numbers in 2014. [online] TechRepublic. Available at: <http://www.techrepublic.com/article/apple-v-google-the-goliath-deathmatch-by-the-numbers-in-2014/> [Accessed 3 Oct. 2014].

Violino, B. (2013). A clear-eyed guide to Android's actual security risks. [online] InfoWorld. Available at: <http://www.infoworld.com/article/2609338/android/a-clear-eyed-guide-to-android-s-actual-security-risks.html> [Accessed 3 Oct. 2014].

[*Microsoft SQL Azure Enterprise Application Development*](http://www.amazon.com/Microsoft-Azure-Enterprise-Application-Development/dp/1849680809/ref=ntt_at_ep_dpt_2). Packt Publishing. p. 420. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [978-1-84968-080-6](http://en.wikipedia.org/wiki/Special:BookSources/978-1-84968-080-6).