OULU UNIVERSITY OF
APPLIED SCIENCES

Arne Caris

ESTREAM: UNITING GAMERS USING A MOBILE APPLICATION

ESTREAM: UNITING GAMERS USING A MOBILE APPLICATION

Arne Caris
Bachelor’s Thesis
Autumn 2020
Information Technology
Oulu University of Applied Sciences

ABSTRACT

Oulu University of Applied Sciences
Information Technology

Author: Arne Caris
Title of the bachelor’s thesis: Estream: Uniting Gamers Using a Mobile Application
Supervisors: Veikko Tapaninen (OUAS), Teemu Korpela (OUAS)
Term and year of completion: Autumn 2020
Number of pages: 39

This thesis is based on a personal project, Estream. The aim was to build an improved multiplatform mobile application based on the feedback received on a prototype version.

The thesis describes the development steps of a multiplatform mobile application based on competitive E-sports and the streaming of games. Firstly, an online survey is conducted to gather initial information. Secondly, a previously made prototype will be put on the Android app store to gather feedback. Lastly, both the online survey and the prototype feedback will be studied and they form the basis for the development of the improved multiplatform application.

The prototype has been distributed and feedback was gathered. With the use of this feedback the first Estream version was created. Estream is now a multiplatform mobile application ready for distribution.

At the end of reading the thesis the reader should have a perfect understanding of the development process of a multiplatform mobile application.

Keywords: E-sports, Application, Mobile

PREFACE

The work was written in Oulu, Finland during the Winter term of 2020. It is based on a personal project. This work has been supervised by Veikko Tapaninen and Teemu Korpela, both teachers at Oulu University of Applied Sciences.

Oulu, 04.12.2020
Arne Caris

CONTENTS

VOCABULARY

APK: Android package Kit. .APK is a file format used by Android systems mainly used for the installation of applications.

CLI: Command Line Interface. The Command Line Interface sends user-typed commands to the command line interpreter which then translates these commands for the system to understand and execute.

E-sports: Abbreviation of Electronic Sports. This is the art of playing computer games on a competitive level.

IDE: Integrated Development Environment. An IDE helps developers by offering enhancements for writing code.

RSS: Rich Site Summary. An RSS feed is a piece of information that has been put on the web. They provide live information and are an easy way for users to follow content. It is possible to integrate these RSS feeds into web pages or applications.

Twitch: Twitch is a streaming service mainly used by gamers. People stream themselves playing games for other people to watch live.

Xcode: Xcode is an IDE specifically made for iOS devices.

1 INTRODUCTION

E-sports is growing. And it is growing rapidly [1]. Every year more and more E-sports tournaments are being played. With so many matches being played, in so many different competitive games, it is only normal that sometimes you forget that there is a match you want to watch. This is where Estream comes into play. Estream offers a digital memory for your favourite games. Estream will even notify you when your favourite match starts, so you do not have to remember it anymore.

This thesis will describe the development process of a multiplatform mobile application called Estream. The aim was to create a mobile application running on both the Android and the IOS operating systems. The main objective was to make it as simple and as user-friendly as possible but interactive at the same time. Interactivity and a sense of progression were achieved by implementing a few gamifications.

All the steps of the development process will be explained in detail. The features of the application will also be explained. At the end of the thesis the reader should understand the development process steps and be able to take the application in use.

2 ESTREAM: THE PROTOTYPE

Prototyping is a form of testing initiated at the very beginning of development, sometimes even before development. A prototype acts purely as an idea which is distributed to potential end users who then provide feedback on the said prototype. This form of testing enables the developer to play around with many different ideas and build upon the needs of the end user. Prototyping is usually done until the application is ready for publication. This means that an application can have multiple iterations of the prototype throughout development.

Initiating prototype testing before development has a few advantages [2]. Firstly, the user feels like they are a part of the development team. Being able to have this amount of input as a user feels good and it provides a sense that the developer cares about their users. Secondly, initiating prototyping before development allows for easier changes. A finished application is way more difficult to change than one sitting at the very beginning of development. Lastly, an application tested by the users themselves gives the developer some insight into a few things, for example, whether the users will become bored of the application, whether the users will find this application, or if the application will solve a critical problem. Based on the answers to these questions, the developer can change, delete, or add certain aspects of the application at an early stage.

As it stands now, Estream has 2 prototypes.

2.1 Design

The design for the prototype was made with simplicity in mind. The aim was to build on it and not take away from it. Because of this, the prototype has a very minimal outlook of the vision that the developer has.

The bottom of every main screen contains a navigation bar, seen in figure 1. Generally, there are two different styles of navigation when looking at mobile applications: a navigation bar **or** a hamburger menu. According to a study done by Dimitrios Tsidoulos [3], a navigation bar is more likely to appease mobile application users. For that reason, a bottom navigation bar was used in the application.

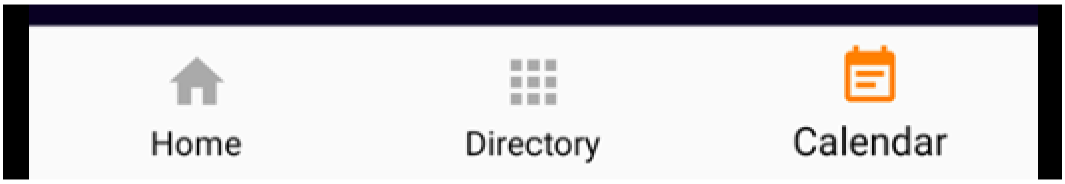


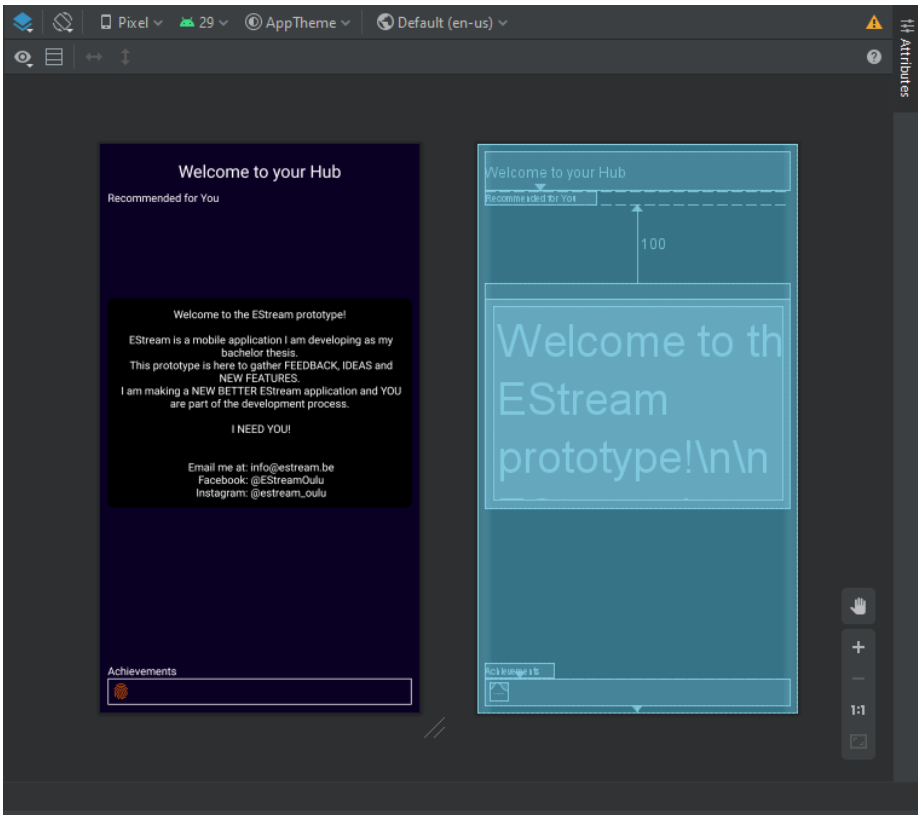
Figure 1. Bottom navigation bar with current location highlighted in orange

Simplicity in an application often brings boredom, to counteract this, the choice of colours becomes very important. The application prototype uses a monochromatic scheme of purple. A monochromatic scheme is a colour scheme that utilises only one colour and its various tints. This colour scheme brings simplicity and elegance to a mobile application and, according to Adoriasoft, is one of the trendiest colour schemes for mobile applications [4]. To contribute to that, purple is a colour often connected with royalty. Thus, adding more elegance and a feeling of prestige to the application.

2.2 Implementation

The first prototype was made with Java.

This prototype was part of a school project for Android applications. It was coded in the Android Studio IDE. Android Studio offers many advantages for programming Android applications, one of which is the design builder, which can be seen in figure 2. Android Studio also offers an easy way to build the application for release.



Welcome to your Hub

Recommended for You

Welcome to the EStream prototype!

EStream is a mobile application I am developing as my bachelor thesis.
This prototype is here to gather FEEDBACK, IDEAS and NEW FEATURES.
I am making a NEW BETTER EStream application and YOU are part of the development process.

I NEED YOU!

Email me at: info@estream.be
Facebook: @EStreamOulu
Instagram: @estream_oulu

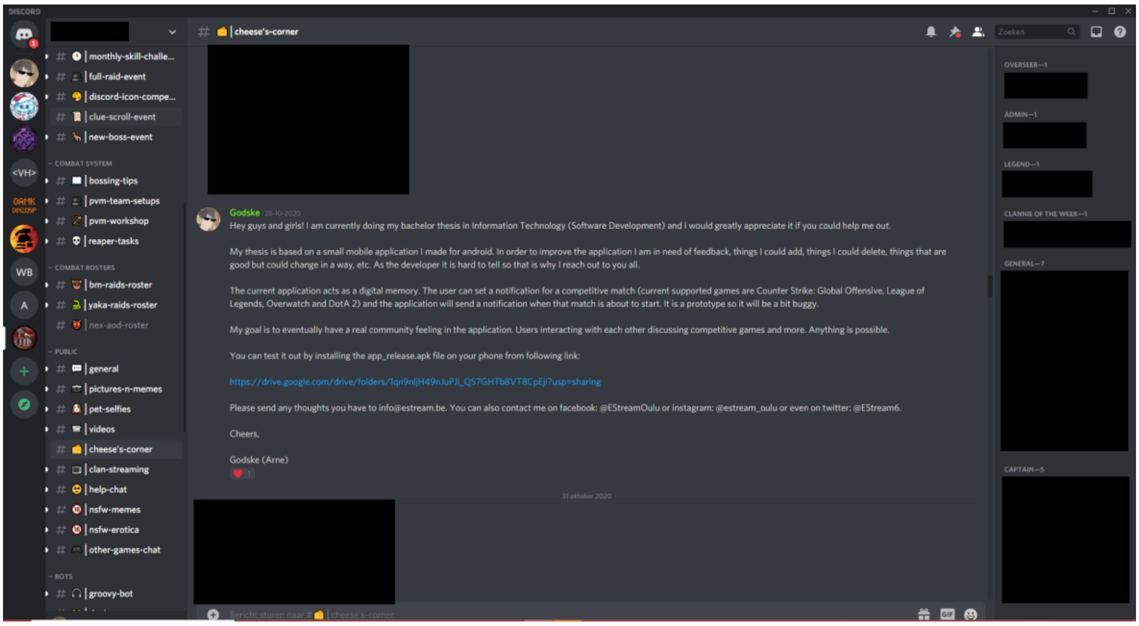
Achievements

Figure 2: The Android Studio design builder

Later, as preparation for multiplatform development, this prototype was also made with JavaScript using the React Native framework. React Native allows for the creation of both Android and iOS applications with the same code base. JavaScript and React are also more widely used for web development which made the choice of remaking the prototype more valid. React Native does not offer the same advantages that Android Studio offers but there is plenty of documentation on the web. And since React is more widely used, there are solutions for nearly every issue available online.

2.3 Distribution

Distribution is a very important factor for an application, and even a prototype, to succeed. A couple of platforms were used to guarantee high visibility for the prototype. First, a communication platform called Discord was utilised. An example of how this was done can be seen in figure 3. Discord is taking the online communication scene by storm. Discord is a platform mainly intended to provide easy communication for anyone alike but it is mainly known with gamers, which is the perfect user group for this type of application. Users can make an account on Discord and chat with their best friends or join a special chatroom, or a server, created for only a certain group of people. These Discord servers are perfect places to distribute information within a certain gaming scene. Therefore, joining free-to-access Discord servers which revolve around gaming would be a good option to distribute the prototype.



[redacted]

Godske 28-10-2020

Hey guys and girls! I am currently doing my bachelor thesis in Information Technology (Software Development) and I would greatly appreciate it if you could help me out.

My thesis is based on a small mobile application I made for android. In order to improve the application I am in need of feedback, things I could add, things I could delete, things that are good but could change in a way, etc. As the developer it is hard to tell so that is why I reach out to you all.

The current application acts as a digital memory. The user can set a notification for a competitive match (current supported games are Counter Strike: Global Offensive, League of Legends, Overwatch and DotA 2) and the application will send a notification when that match is about to start. It is a prototype so it will be a bit buggy.

My goal is to eventually have a real community feeling in the application. Users interacting with each other discussing competitive games and more. Anything is possible.

You can test it out by installing the app_release.apk file on your phone from following link:

https://drive.google.com/drive/folders/1qn9nljH49nJuPJi_QS7GHTb8VT8CpEji?usp=sharing

Please send any thoughts you have to info@estream.be. You can also contact me on facebook: @EStreamOulu or instagram: @estream_oulu or even on twitter: @EStream6.

Cheers,

Godske (Arne)

[redacted]

[redacted]

Figure 3: Example use of Discord as a distribution service (Godske is the author's nickname on the Discord server)

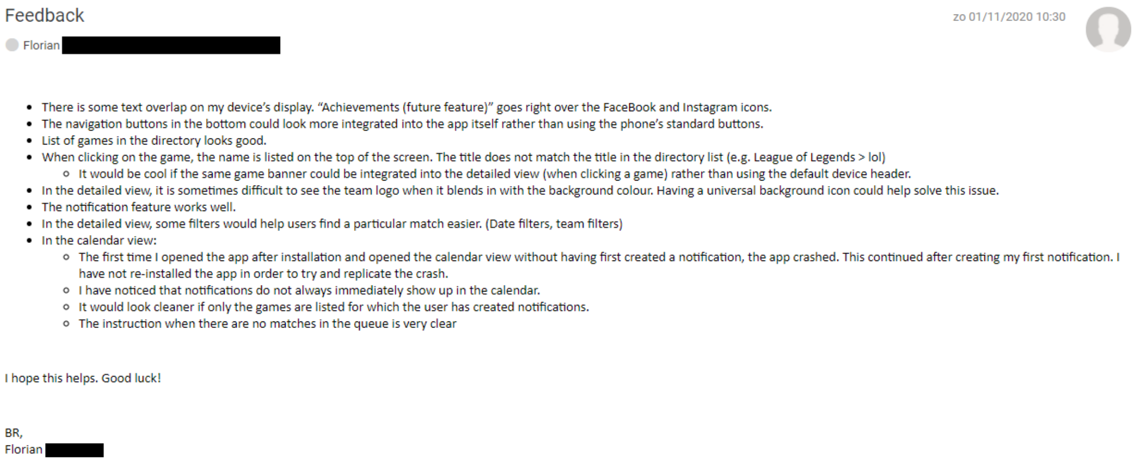
But it does not stop there.

It is important to do plenty of research as to where and when an application could flourish the most. This is not limited to one single platform. This is a mobile application, therefore there is no better place than to publish it on the Android app store. However, publishing an application on any app store comes with its own issues; for example, how to guarantee that people see the application, how to make people interested in downloading the application, and most of all, how to keep interested users testing it out and sending feedback. The first question has a solution: Facebook. Even though Facebook has been declining in popularity, it is still one of the most used social media platforms.

Facebook offers their users the possibility to advertise their products in many ways. One of them is placing advertisements between posts. According to Ali Parmelee, Facebook Ads Lead and Paid Media Specialist, it is one of the most effective ways to advertise if it is done correctly [5].

3 FEEDBACK

After distributing the prototype, it is time to analyse the received feedback. This feedback is crucial in the development process as it allows the developer to adjust the application to the needs and wants of the user. The feedback was gathered as free text, as can be seen in example figure 4, and all feedback was sent to the central Estream e-mail account.



Feedback

Florian [redacted]

zo 01/11/2020 10:30

- There is some text overlap on my device's display. "Achievements (future feature)" goes right over the FaceBook and Instagram icons.
- The navigation buttons in the bottom could look more integrated into the app itself rather than using the phone's standard buttons.
- List of games in the directory looks good.
- When clicking on the game, the name is listed on the top of the screen. The title does not match the title in the directory list (e.g. League of Legends > lol)
  - It would be cool if the same game banner could be integrated into the detailed view (when clicking a game) rather than using the default device header.
- In the detailed view, it is sometimes difficult to see the team logo when it blends in with the background colour. Having a universal background icon could help solve this issue.
- The notification feature works well.
- In the detailed view, some filters would help users find a particular match easier. (Date filters, team filters)
- In the calendar view:
  - The first time I opened the app after installation and opened the calendar view without having first created a notification, the app crashed. This continued after creating my first notification. I have not re-installed the app in order to try and replicate the crash.
  - I have noticed that notifications do not always immediately show up in the calendar.
  - It would look cleaner if only the games are listed for which the user has created notifications.
  - The instruction when there are no matches in the queue is very clear

I hope this helps. Good luck!

BR,
Florian [redacted]

Figure 4: Example of received feedback

3.1 Deletions

According to the feedback received from the testers, there are no necessary deletions. All features currently in the second prototype had a purpose and seemed to be well received. This means that the idea is solid and Estream can already achieve its goal at a bare minimum.

3.2 Adjustments

The feedback from the testers suggested a couple of changes.
Firstly, it was suggested to have a lighter background or to add a subtle pattern to the background. This was suggested because the background looked a bit stale. The lighter color was requested because some of the team logos were not visible but it is difficult to find a color that satisfies every team logo, unless it is

plain white.
Another suggested change is for the notifications. The testers proposed having an option to turn push notifications on or off.
The last suggested change is for the way the calendar has been integrated. The testers suggested integrating a Google or Outlook calendar, which sounds like a solid idea. The calendar has always been a struggle, to the point where the second prototype calendar is not even a real calendar. Google or Outlook integration could simplify the development of the calendar and make it look better for users.

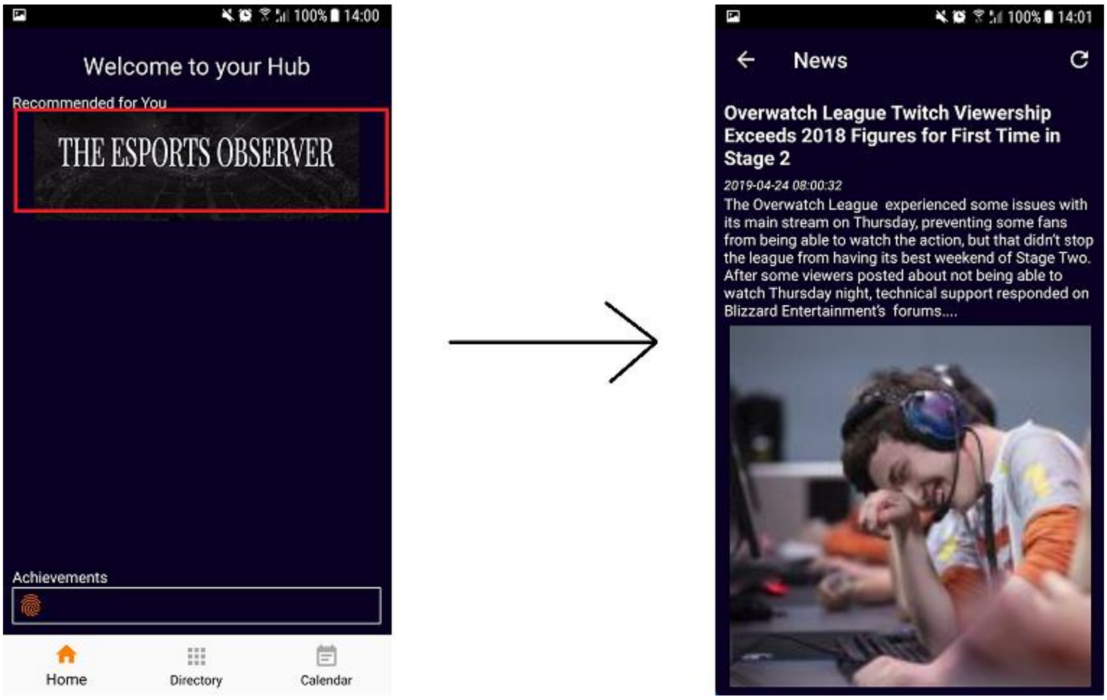
3.3 Additions

The suggestions received for additions were as follows:

- The ability to make user accounts
- Achievements and badges
- In-App currency
- A live news feed
- Multiple directory improvements

A first recommended addition to the application is user accounts. User accounts form the base from where the other features follow, such as Twitch integration. Having the possibility to watch the matches live on mobile devices will keep the users more entertained within the application. User accounts will also make it possible to talk with other viewers watching the same match.
A second possible addition that goes in line with the user accounts is achievements and their respective badges. An achievement system adds a type of gamification to the application. Badges, earned by completing achievements, can be put in front of the user's account name to show their commitment. An example achievement could be: ”Watch the team X play 50 times”. Fans of team X can then show their support to that team by equipping the earned badge for

this achievement and other users will be able to see that this user supports the team X. Another possible gamification could be to add in-app currency. In-app currency sounds like a good, and fun, addition. However, in-app currency comes paired with some risks. Firstly, in-app currency needs to be balanced really well. Secondly, it has to stay inside the app. It is not rare for online, in-game, currency to become a means of real world selling and buying. Lastly, what should be offered in return for accumulating that currency. Currently, Estream is very straightforward and there are no extra features which could be unlocked by spending in-app currency. This currency would purely be for boasting. It is a well-known strategy to introduce a kind of in-app currency and, only after a while, tie rewards to accumulating it. This could be done. A third recommended addition is a news section. The first prototype used to have a news section which was implemented as an RSS feed. This prototype included a news feed which updated every minute to find the latest news articles from The E-sports Observer [6], as seen in figure 5. However, this was scrapped in the second prototype. It would definitely improve app engagement if the news feed would be brought back, but in a better implementation.



Overwatch League Twitch Viewership Exceeds 2018 Figures for First Time in Stage 2

2019-04-24 08:00:32

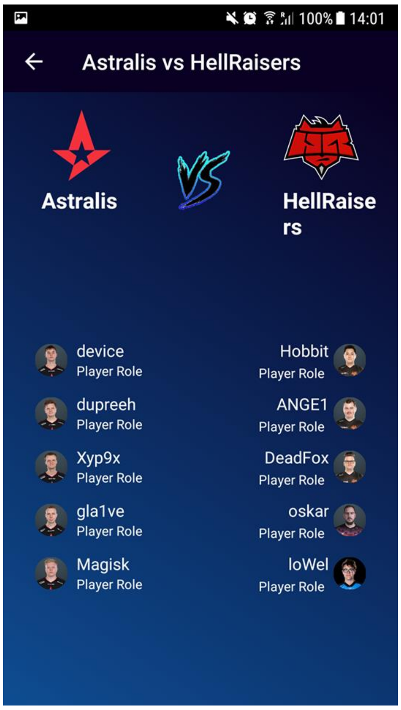
The Overwatch League experienced some issues with its main stream on Thursday, preventing some fans from being able to watch the action, but that didn't stop the league from having its best weekend of Stage Two. After some viewers posted about not being able to watch Thursday night, technical support responded on Blizzard Entertainment's forums....

Figure 5: The E-sports Observer news feed from the first prototype

The last recommended additions all have to do with the games and the matches. The first requested addition is an addition for far in the future; an options button. This button would allow the user to modify the look of the app. The layout of the notifications, the layout of the app and the possibility to hide games. In the current state, the app only offers four games which makes hiding games unnecessary. But once the app grows, and gathers a bigger following, more games would have to be offered to the users and then this is definitely an option.

The second request is to add a search function for leagues and teams. This would be a good addition. Inside the whole list of matches for a game it can be troublesome to find the matches your favorite teams are playing. The same goes for leagues. Not everyone follows every league of a game. It is actually very common for E-sports fans living in Europe to only follow the European E-sports leagues, respectively Americans are more likely to only follow the American E-sports leagues. A search function for teams and leagues would highly enhance the user experience. Another suggestion, which follows on the search function, is a separate view for the favorited games, leagues and teams of the user. The user would be able to add teams or leagues to their ”favorites” and only those teams or leagues would show in that specific view.

The last suggested addition is an info view about matches and teams. This was already implemented in the first prototype (see figure 6) and can be reimplemented in the first release version.



Astralis vs HellRaisers

Astralis

HellRaise
rs

device Player Role	Hobbit Player Role
dupreeh Player Role	ANGE1 Player Role
Xyp9x Player Role	DeadFox Player Role
gla1ve Player Role	oskar Player Role
Magisk Player Role	loWel Player Role

Figure 6: The match and teams details view from the first prototype

4 ESTREAM: VERSION 1.0

After distributing the prototypes and gathering feedback on them, the next step in the development process is preparing the first version.

4.1 Design

An important part of creating an application is to consider what the application is going to look like. The design is the first thing a user will notice when opening the application. LogoDesignValley says:

> Basically, our mind is programmed to get attracted to colors and ambiance it perceives. [7]

So, to have an audience that sticks it is important to get this right. To make the right decision it is vital to look at the LogoDesignValley color scheme, as seen in figure 7 below.

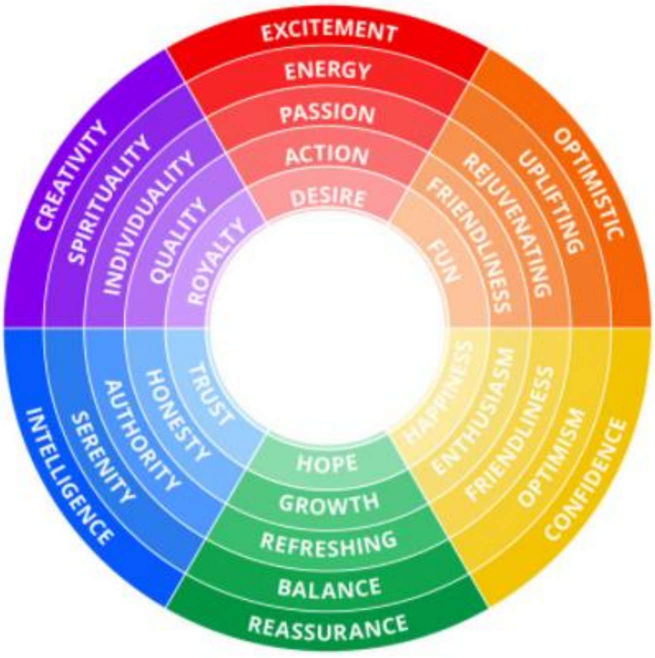


Figure 7: The LogoDesignValley color scheme

Estream will utilise the purple color scheme. Purple was chosen because it gives a sense of calm and it is a color often connected with E-sports and games. Different shades of purple are being used to give the application a more lively look, as can be seen in figure 8. It is a simple and straightforward design that calms the user and is easy to follow.

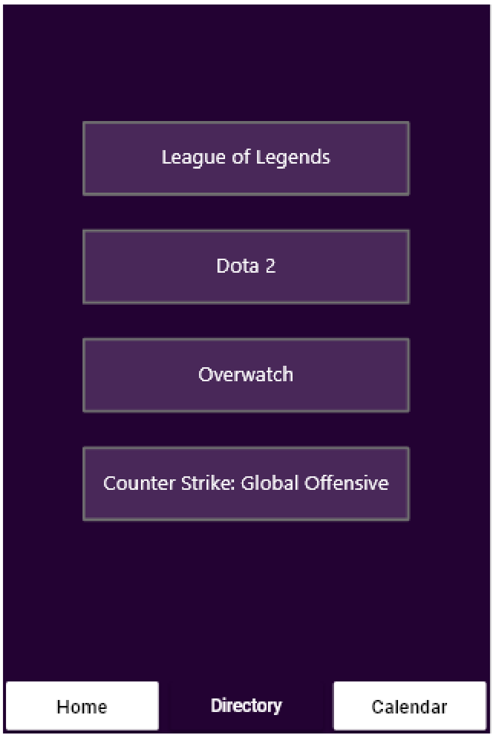


Figure 8: The directory screen showing of some purple shades

4.1.1 Adobe XD

Adobe XD was used for creating the designs. Adobe XD is an easy to understand designing software available on Mac, Windows and even Android and iOS. The main reason Adobe XD was used is because of the prototyping feature. This feature allows the user to run through their designs as if it was an actual application. This helps the designing process massively because it gives the developer a better view on how everything connects with each other and how it looks together.

Adobe XD also gives the user a lot of freedom. It is possible to create anything in any way the user wants. An example usage of Adobe XD can be seen below in figure 9.

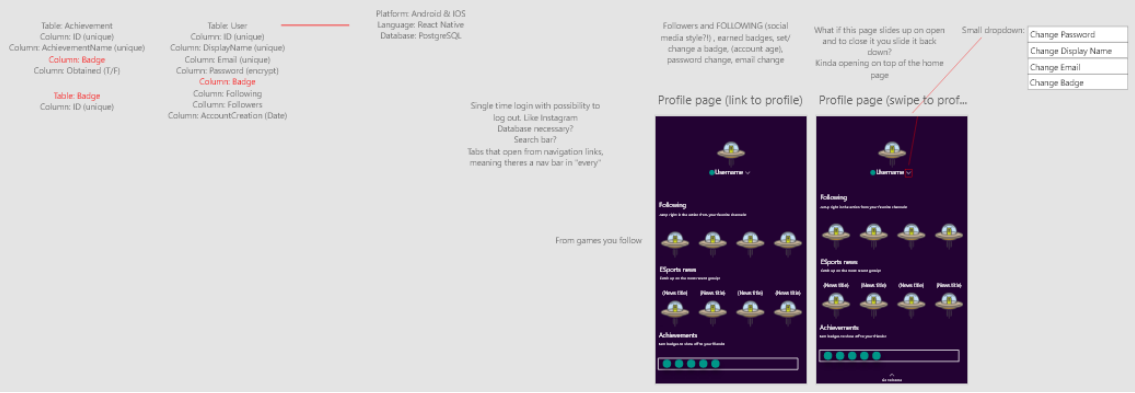


Table: Achievement
Column: ID (unique)
Column: AchievementName (unique)
Column: Badge
Column: Obtained (T/F)

Table: Badge
Column: ID (unique)

Table: User
Column: ID (unique)
Column: DisplayName (unique)
Column: Email (unique)
Column: Password (encrypt)
Column: Badge
Column: Following
Column: Followers
Column: AccountCreation (Date)

Platform: Android & IOS
Language: React Native
Database: PostgreSQL

Single time login with possibility to log out. Like Instagram
Database necessary?
Search bar?
Tabs that open from navigation links, meaning theres a nav bar in "every"

From games you follow

Followers and FOLLOWING (social media style?!) , earned badges, set/ change a badge, (account age), password change, email change

What if this page slides up on open and to close it you slide it back down?
Kinda opening on top of the home page

Small dropdown:

Figure 9: A part of the Estream design

The figure shown above displays the fluidity of Adobe XD. On the left side there is brainstorming text about the back-end possibilties, while on the right there are two possible profile screens designed fully, with every detail and images imported by the user.

As previously mentioned, it is also possible to run a preview of the prototype. This looks like Figure 10 below. The first step is to declare a home screen for the prototype, in this case the screen on the left of Figure 6. The next step is to create the links inside the prototype. An example of a link can be seen in figure 11. When this prototype is being run, it is possible to press the ”directory” button in the bottom navigation bar to open the directory screen, seen on the right side of figure 10.

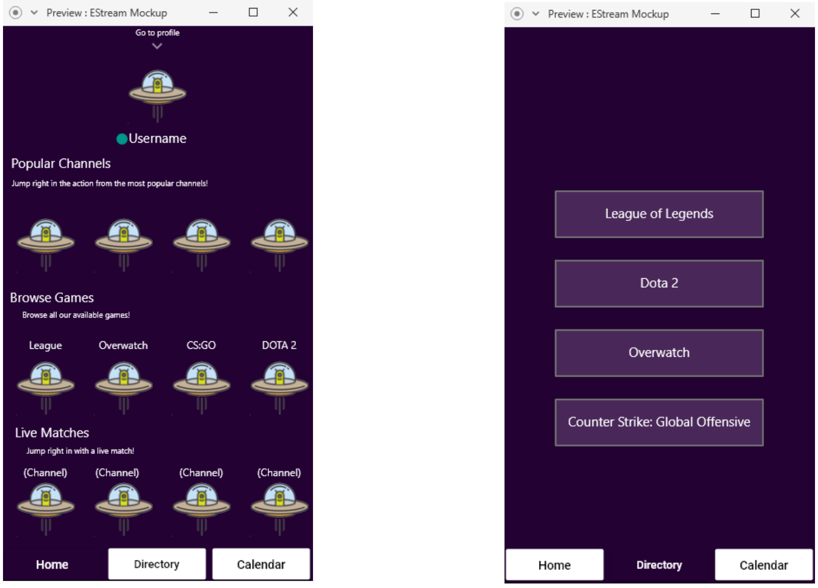


Figure 10: Two prototype screens opened in the Adobe XD previewer

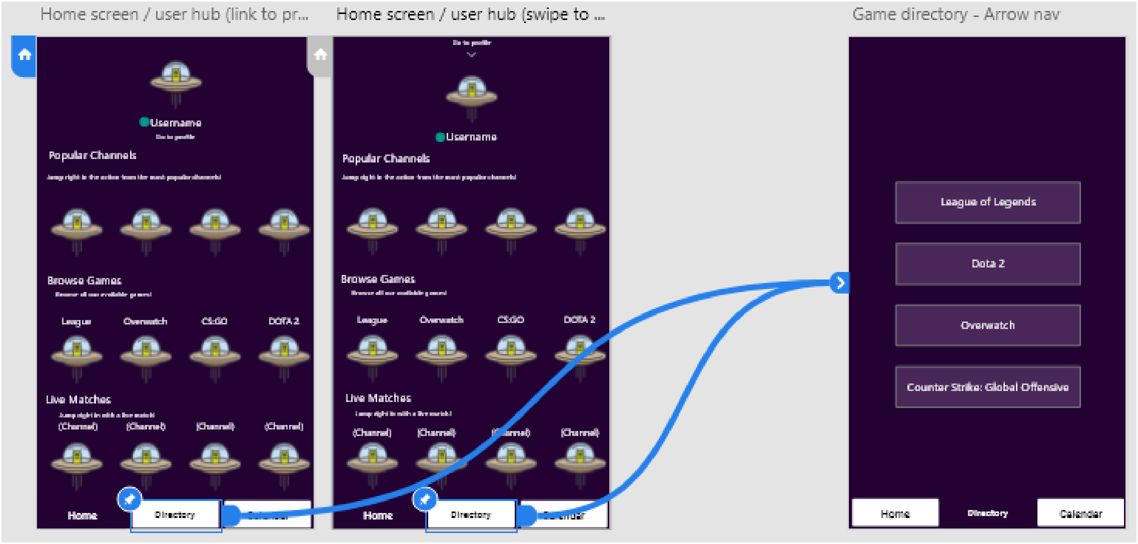

Home screen / user hub (link to pr...
Home screen / user hub (swipe to ...
Game directory - Arrow nav
Username
Popular Channels
Browse Games
Live Matches
Home
Directory
Calendar
League of Legends
Dota 2
Overwatch
Counter Strike: Global Offensive

Figure 11: Example of linking screens in Adobe XD

4.1.2 Copyrights

Estream is pulling information from the Internet. This information has been put on the Internet by other individuals or companies. This means the team logos, the team names or even the RSS feed for the news section. These are all elements that have not been created by the developer of Estream.

Copyrights are a very fragile element that is often forgotten while eagerly designing an application. The designer must find out whether it is even allowed to use the images that are being used. The answer is simple: nobody knows, except for the original owner of the information. According to Nathan Orr [8], internet copyrights do exist but are a very grey area. There is no international standard in regard to the legal use of online information. Because of this, it is difficult to defend oneself against the dangers of a copyright strike. A copyright strike can be anywhere in between the owner of the information asking the user to remove the information from their application and the owner of the information challenging the user legally.

The easiest, and most friendly, way to go about using online information from others is to include a copyright statement. Estream does this by adding the following statement under every screen which shows pulled information.

```
Estream does not take credit for any of the team names and logos
presented on this page. All claim goes to the original owners.
```

4.2 implementation

This chapter will give a detailed explanation about the implementation of the application. This chapter explains how the application was made, why it was made in this way and what it has to offer to the user.

4.2.1 Cross Platform

Cross Platform compatibility is the main reason React native was chosen for this project. Being able to publish an application on both Android and iOS significantly increases the reach of the application. It is also just more for users if they can talk about the app with their friends even though they have different mobile operating systems.

4.2.1.1 Android

According to GlobalStats [9], Android has 72.92% of the worldwide mobile operating market share. Because of this, it is crucial to make the application compatible with Android. React Native makes this easy. All that is necessary is to get a signed APK of the application.

There are a few steps to making a signed APK.

Firstly, an upload key must be generated. This must be done via the Keytool that

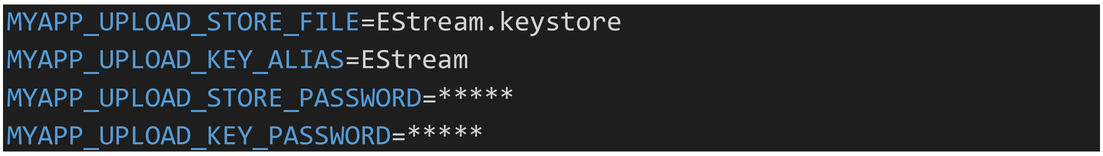
```
C:\Program Files\Java\jdkx.x.x_x\bin
```

came with the Java installation. To find the Keytool it is necessary to navigate to the following path in the command prompt:

Once there, the upload key can be generated with a single command:

```
C:\Program Files\Java\jdk1.8.0_211\bin>keytool -genkey -v -keystore
EStream.keystore -alias EStream -keyalg RSA -keysize 2048 -validity
10000
```

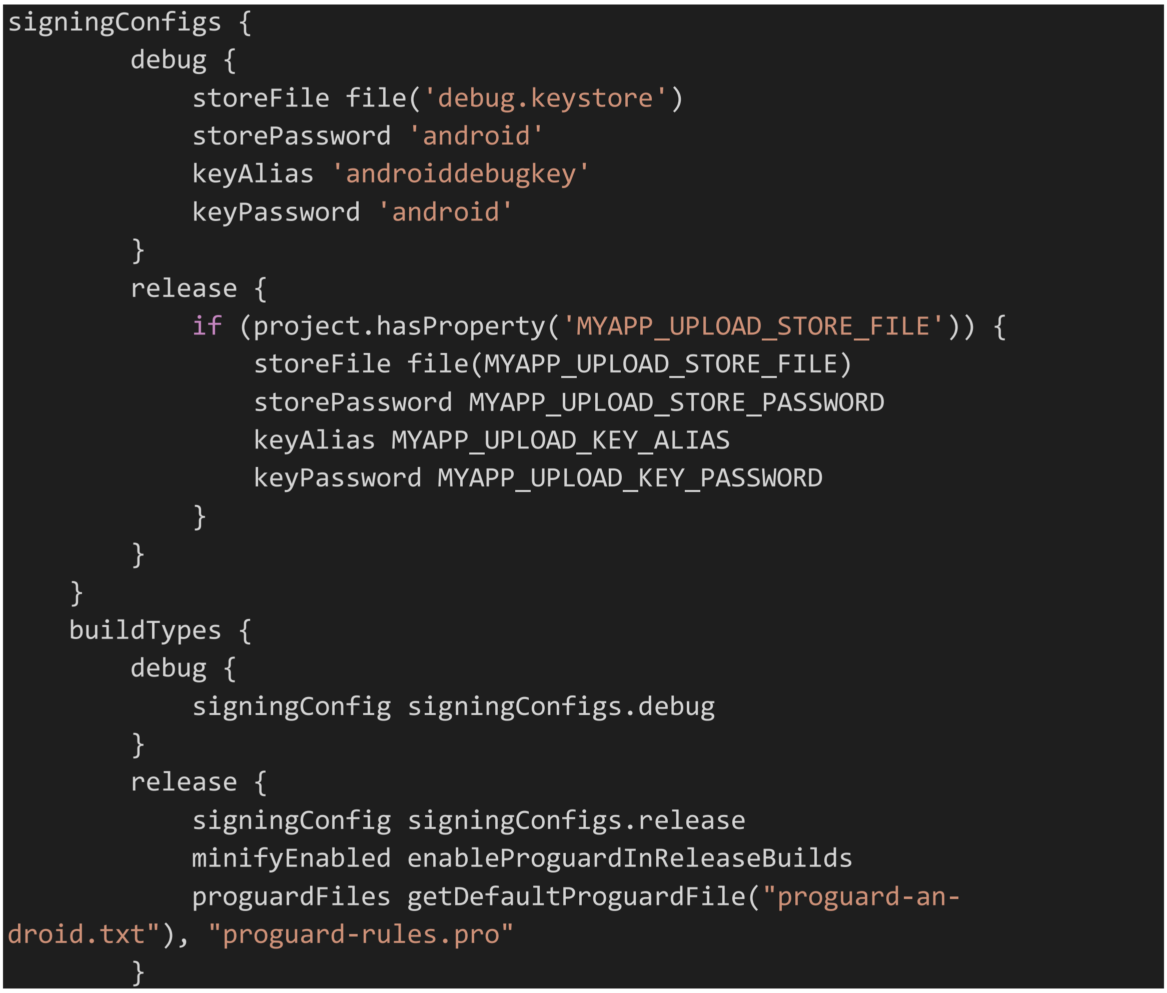
This will prompt a couple of questions to supply information to the Keytool. With this information an EStream.keystore file is generated. Then the generated EStream.keystore file is placed in the android/app folder. This way it can be reached by the gradle.properties file in the Android folder. And for the gradle.properties file to call upon the EStream.keystore file, the following couple of lines have to be added to the gradle.properties file:



```
MYAPP_UPLOAD_STORE_FILE=EStream.keystore
MYAPP_UPLOAD_KEY_ALIAS=EStream
MYAPP_UPLOAD_STORE_PASSWORD=*****
MYAPP_UPLOAD_KEY_PASSWORD=*****
```

Figure 12: Example of the lines to add to the gradle.properties file

The first line contains the name that was given in the keytool command. The second line contains the alias provided in that same line. The third and fourth lines contain the passwords that were provided to the prompted questions.
The next step is to provide those values to the build.gradle file, as seen in figure 13.



```
signingConfigs {
        debug {
            storeFile file('debug.keystore')
            storePassword 'android'
            keyAlias 'androiddebugkey'
            keyPassword 'android'
        }
        release {
            if (project.hasProperty('MYAPP_UPLOAD_STORE_FILE')) {
                storeFile file(MYAPP_UPLOAD_STORE_FILE)
                storePassword MYAPP_UPLOAD_STORE_PASSWORD
                keyAlias MYAPP_UPLOAD_KEY_ALIAS
                keyPassword MYAPP_UPLOAD_KEY_PASSWORD
            }
        }
    }
    buildTypes {
        debug {
            signingConfig signingConfigs.debug
        }
        release {
            signingConfig signingConfigs.release
            minifyEnabled enableProguardInReleaseBuilds
            proguardFiles getDefaultProguardFile("proguard-android.txt"), "proguard-rules.pro"
        }
```

Figure 13: Example of the lines added to the build.grade file

This file can be found in the android/app directory.

The last step is to navigate to the Android folder in the command prompt. Once there, the following command must be executed:

```
gradlew assembleRelease
```

Gradlew assembleRelease will build the APK with the keystore file created in the previous steps. The signed APK will then be visible at

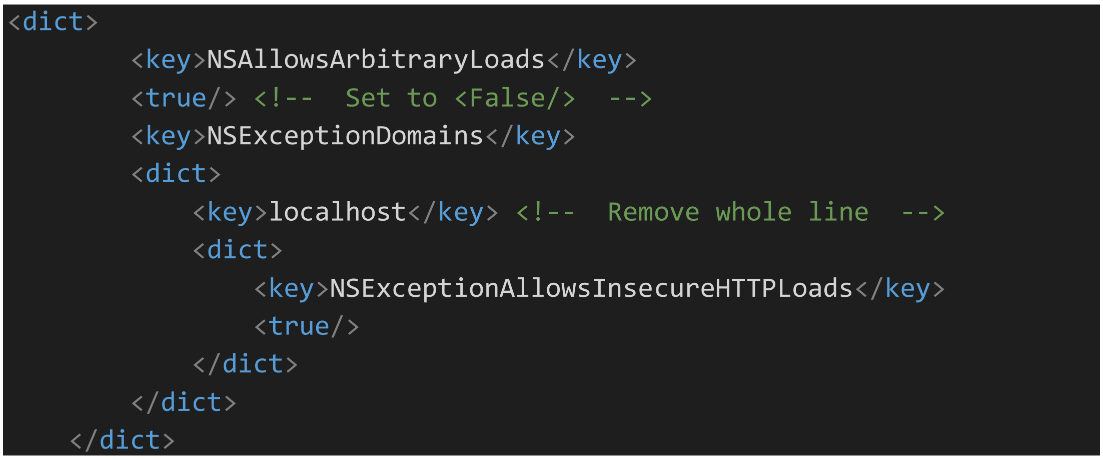
```
android/app/build/outputs/apk/app-release.apk
```

and can be distributed and installed on Android devices.

4.2.1.2 IOS

GlobalStats also shows that iOS makes up for 26.53% of the global mobile operating market, making Android and iOS the vast majority with 99.45%. IOS is then also the second, and last, operating system Estream will be compatible with. Building the iOS version of the app for release is a whole lot easier than for Android, at least that is if the developer has access to a MacOS operating system and an iOS device. All configurations for the iOS release can be made simply by opening the project in Xcode. Xcode is an IDE specifically made for MacOS containing tools to develop software for Apple products.

Once inside Xcode, it is important to enable App Transport Security (ATS). This is a security feature that blocks all HTTP traffic for the application. It is possible to do this by making the following changes to the Info.plist file in the /iOS folder (figure 14):

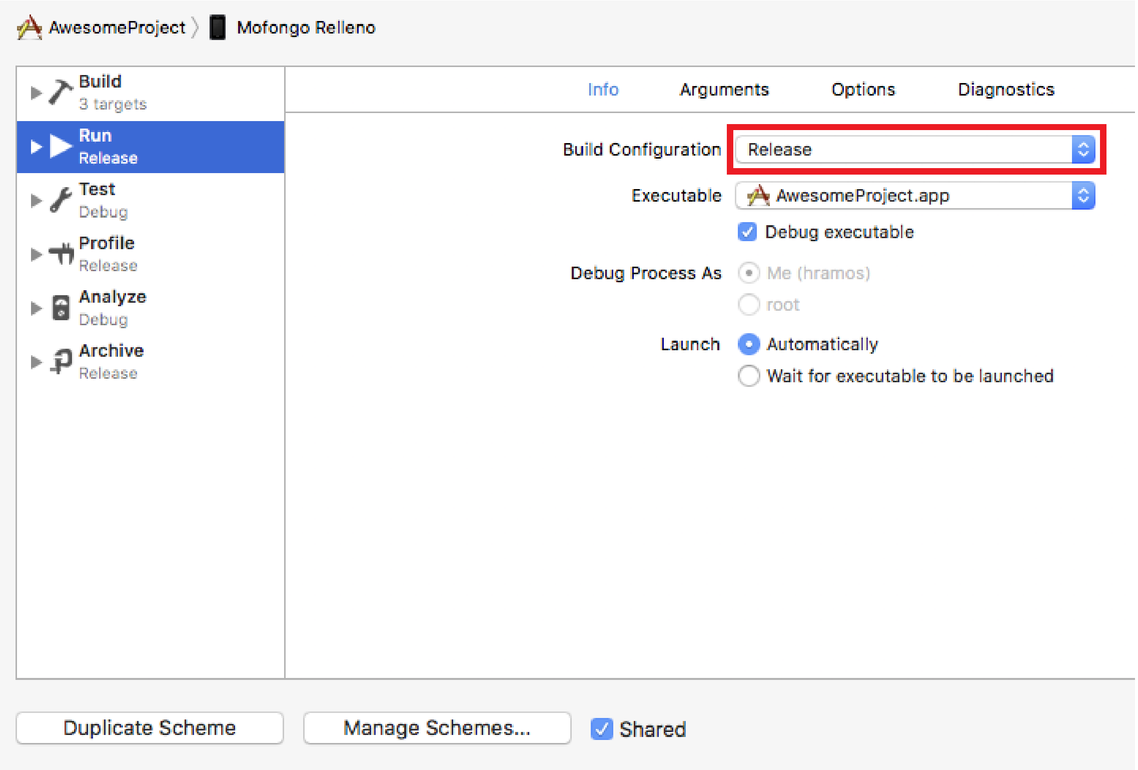


```
<dict>
        <key>NSAllowsArbitraryLoads</key>
        <true/> <!--  Set to <False/>  -->
        <key>NSExceptionDomains</key>
        <dict>
            <key>localhost</key> <!--  Remove whole line  -->
            <dict>
                <key>NSExceptionAllowsInsecureHTTPLoads</key>
                <true/>
            </dict>
        </dict>
    </dict>
```

Figure 14: Example of Info.plist changes

It is also possible to enable ATS by accessing the info pane in Xcode, then navigating to the target properties and editing the app ATS entry. The author does not own a MacOS device, consequently, also no Xcode, hence both ways are explained.

The second step is to configure the build configuration. This is achieved by navigating to Product, then to Scheme and Edit Scheme. On the left side of the panel there is a tab called Run. The user must open this tab and select Release under Build Configuration, as can be seen in figure 15.



Build Configuration: Release

Executable: AwesomeProject.app

☑ Debug executable

Debug Process As: ● Me (hramos) ○ root

Launch: ● Automatically ○ Wait for executable to be launched

☑ Shared

Figure 15: Example of build configuration Release setting

Lastly, the app must be built. Inside Xcode, this can be done by navigating to Product. Inside the Product panel, the designer needs to press “Build” and the application will be built for release.

If not using Xcode, the app can be built by executing the following line in the React Native CLI.

```
 npx react-native run-ios --configuration Release
```

After this, the app will be available for iOS distribution.

4.2.2 Full-Stack

While the prototype only consisted of a frontend, the first version will consist of both a frontend and a backend.

The backend will make use of Google Firebase. Firebase is a highly scalable application manager, featuring e.g. scalable databases, file storage and sign-up methods. The main feature Estream is looking for from Firebase's offerings is the scalable database.

It is never easy to predict the number of users an application is going to have. Having a database which can be increased in size dynamically is important to cut unnecessary costs.

A second interesting feature for Estream are the sign-up methods. Authentication is very important to get right when it comes to security. Setting up authentication without using trusted third-party software might cause headaches and security risks. Firebase offers this authentication for applications. Firebase also offers multiple platforms for authentication, including email, mobile phone, Facebook and Google, giving the user many options. The developer can also freely enable the authentication methods they would like to offer to the user.

Estream is a lightweight application and is, as a whole, not very complex. It is important as a developer to realize that, sometimes, you should not overcomplicate things by adding unnecessary complexity. Keeping it simple is the key feature. A scalable database and a couple of authentication methods will do for Estream, at this point in its lifetime.

Just like the second prototype, the frontend for the first version will be made with React Native. The feedback received on the second prototype will serve as a baseline for the changes to the frontend. Old features will be made better and new features will be added.

4.2.3 Features

This chapter will describe the main features the first Estream release version will have.

4.2.3.1 Home Screen

The home screen is the main screen of the application. This is the access point to the other features of the application.

Firstly, the home screen shows the profile picture and the username of the user, along with a link to visit their profile.

Secondly, there are all kinds of quick links to the main parts of the application. Data aggregation will show which channels are viewed the most and it will show them to the user. This view also contains a section which shows all the games Estream offers so that the user does not always have to go via the directory. The last view of quick links shows the matches that are currently being played. With this section the user can easily jump into a live match and start watching.

Lastly, there is a navigation section at the bottom. Generally, there are two different styles of navigation when looking at mobile applications: a navigation bar **or** a hamburger menu. According to a study done by Dimitrios Tsidoulos [3], a navigation bar is more likely to appease mobile application users. Because of this, Estream makes use of a bottom navigation bar containing links to the home screen, the directory and the calendar.

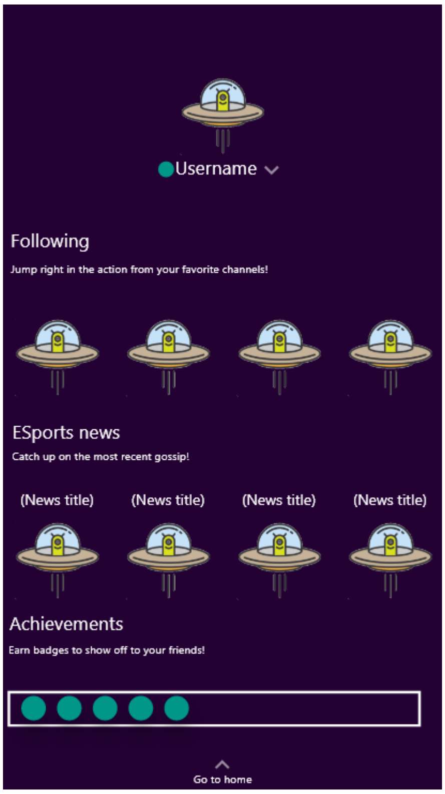
4.2.3.2 User Profile

The user profile, see figure 16, is a new addition to the Estream version one.

The first feature of the user profile is the following section. The users will be able to follow channels or teams. This section is where those channels or teams will show so that the user can easily access those profiles.

The second feature visible on the user profile page is the news section. This section will contain the most recent news articles from the E-sports world.

Lastly, the user will be able to view a list of achievements, as well as their completed achievements and earned badges. On top of the page, next to the username, the user will be able to open a dropdown menu. This dropdown menu offers account related actions, such as changing password, username or badge.



Jump right in the action from your favorite channels!

Catch up on the most recent gossip!

Earn badges to show off to your friends!

Figure 16: The user profile designed for Estream version 1

4.2.3.3 Calendar

The calendar view will contain an integrated Google calendar. This is to simplify development and so the users can plan their matches according to the rest of their schedule. The calendar will only contain the matches that the user has specifically added to it via the match directory. When the user presses a specific day, a popup view will show an overview of the items on their schedule for that day.

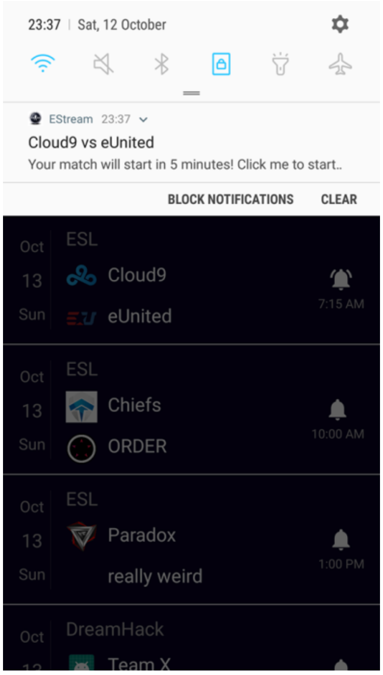
4.2.3.4 Directory

The directory is a collection of games and matches from which the user can choose. It acts as a medium to give the user several options. All directories are built on the PandaScore APIs [10]. PandaScore is the only E-sports data service that is free to use but has limited games. For the current scale of the project this is perfect.

In the game directory the user can choose which game they want to view. Pressing the game icon will bring the user to the match directory of that game. Currently, the four most popular games are supported: League of Legends, CS:GO, Overwatch and DotA 2 [11]. The game directory is designed in such a way that it makes it easy to add more games if needed.

The match directory is the collection of matches played in the game chosen by the user. It covers all the upcoming matches for up to 50 matches. All data is gathered from the PandaScore API and then formatted and designed as seen below in figure 17.

The notification symbol, seen on the right side of every row in figure 17, is the feature that makes this application unique in its kind. This is the notification icon. The notification icon is directly linked to the calendar. When the notification icon is pressed, it will send that match to the personal calendar of the user and add it as an event. Pressing the notification icon once will change its appearance to signify that this match has been added to the calendar. Pressing it a second time will change it back to the original and alert the user that it has been deleted from the calendar.



23:37 | Sat, 12 October

EStream 23:37

Cloud9 vs eUnited

Your match will start in 5 minutes! Click me to start..

Figure 17: Example of notification with the match directory on the background

4.2.3.5 Streaming

Streaming has been introduced by embedding a twitch player [12] into the app. This embedded twitch player has all the same features that the actual twitch application would have. The twitch player makes it very easy for users to watch their matches and chat about them with other viewers, without having to switch applications.

4.3 Testing

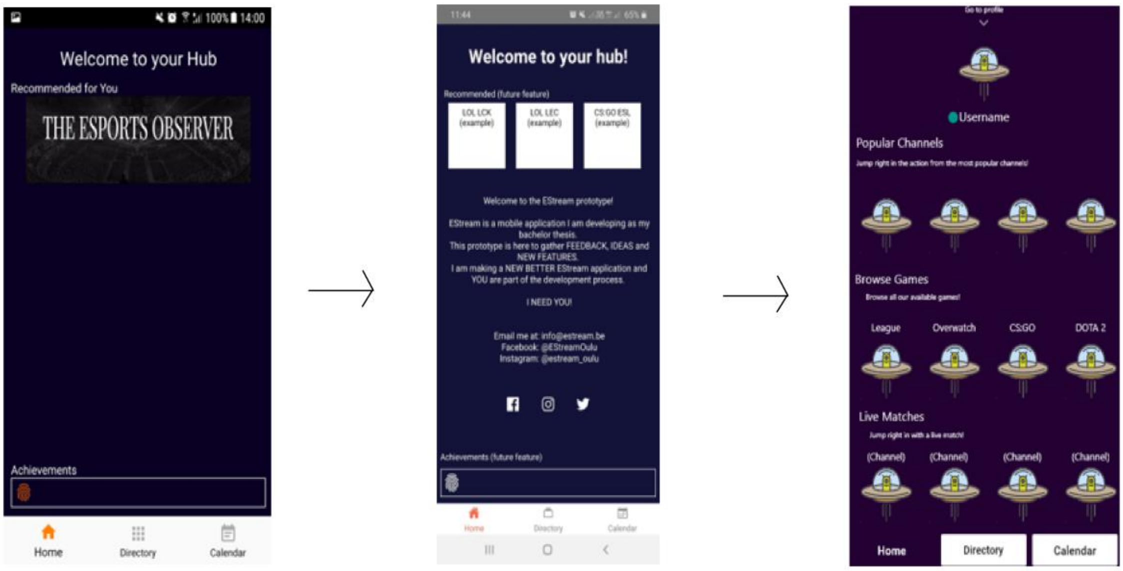
During development, a few testing methods have been used to ensure the best release possible for the app.

The first method of testing used is prototyping. Two prototypes were made and both versions gathered feedback to improve the next iteration of the app. The first prototype was built entirely using Java. This prototype was made as a school project and feedback was gathered from both students and teachers, purely from a presentation point of view. The app was not ditributed for use.

The second prototype was made with React Native. This is the most recent prototype and includes changes which were requested after the release of the first prototype, as well as changes the developer felt were necessary. This prototype version was distributed for use to about 400 testers. 27 of those testers sent their feedback.

During development, the developer also utilised continuous testing. Continuous testing is a method of testing in which automated tests are being executed at every stage of the development process. React Native offers a useful debugger tool for this purpose. This debugger tool will reload the application every time new changes are being saved in one of the application files. This gives the developer the possibility to test very detailed implementations.

Another used testing method is black-box testing. During development, a couple of people were appointed to test the application at certain stages. Black-box testers only know **what** the application is supposed to do, **not how** it is supposed to do it. Because of this, this testing method is mainly used to improve the user experience, the design of the application and possible changes to some features. Figure 18 shows the evolution of the home screens through the multiple stages of testing.



Welcome to the EStream prototype!

EStream is a mobile application I am developing as my bachelor thesis.
This prototype is here to gather FEEDBACK, IDEAS and NEW FEATURES.
I am making a NEW BETTER EStream application and YOU are part of the development process.

I NEED YOU!

Email me at: info@estream.be
Facebook: @EStreamOulu
Instagram: @estream_oulu

Figure 18: The evolution of home screens through application testing stages

4.4 Publication

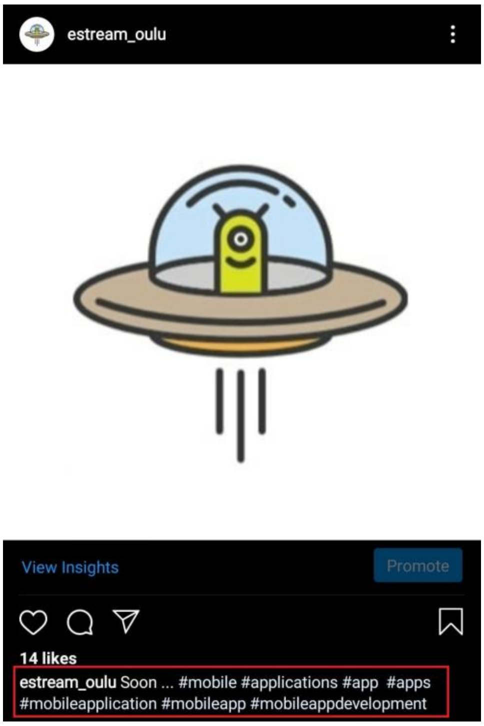
The publication process of the first version is mainly the same as of the second prototype, bar a couple of changes.

Estream will not be distributed via Discord,. The use of Discord servers was purely a means to find testers for the prototype. However, once the user base grows, it is very likely that a Discord server will be started for Estream. This offers another platform for users to discuss the application with both other users and the developer. The main publication platform used for the first version will be the app stores, both the iOS app store and the Android play store. All mentions of the application will lead the users to those app stores.

A couple of different means have been used to ensure the maximum visibility of the app. Firstly, Facebook. Facebook is a massive social platform and offers high visibility. Placing advertisements on Facebook is highly customizable, meaning that it is not necessary to go bankrupt on some advertising. Estream also has its own Facebook page where people can follow the development process and see regular updates.

Another, just as large, social platform is Instagram. Instagram is less about advertisements and more about interactivity. Visibility on Instagram comes more from utilizing ”hashtags” (see figure 19) well enough and gathering clicks on the posts. Hashtags are keywords which will categorize the post in a wide collection of posts that use the same hashtag. When people then search for that hashtag, they will see this collection of posts, and thus also the one Estream posted. Instagram also offers to promote your posts, just like Facebook, but it is less impactful here.

Quick views and likes also cause usre’s post to appear in the Instagram home view. This is a view where Instagram decides which posts a user should see. Of course, this can give Estream a massive boost in interactivity so quick views and likes are essential.



14 likes

estream_oulu Soon ... #mobile #applications #app #apps #mobileapplication #mobileapp #mobileappdevelopment

Figure 19: Example of hashtags on an Instagram post

Lastly, Estream is also active on Twitter; another social platform. Twitter is different from Facebook and Instagram in the sense that it is more about discussion. It is easier for users, or interested people, to discuss Estream related things with each other and the developer.

5 CONCLUSION

The aim of this thesis was to guide the reader through the development process of a mobile application using React Native. Many junior developers are so caught up in the implementation and the “look” of an application that they forget everything that happens around it. A good planning, a good design and a thorough research are equally important, if not **more** important.

The same goes for testing the application. As a developer, it is possible to gather massive amounts of info just from having people use the prototypes. It is important to remember that a prototype is not supposed to be a finished product. It can be difficult sometimes to show something that is not finished and contains a high number of bugs or inconsistencies.

There is a definite improvement in the current version compared to the previous two prototypes. However, development on applications like these never stops. The application is now in a state where it is deployable, but it is far from perfect. Continuous improvements and changes to accommodate the needs of the users is a must and will continue for as long as the lifetime of the application.

Looking back at this project, it was an educational experience in creating mobile applications. During this time, several new techniques were acquired which will be helpful in future projects.

REFERENCES

[1] I. MarketingHub, “The Incredible Growth of eSports in 2019 [+ 2019 eSports Stats],” Influencer MarketingHub, 28 March 2019. [Online]. Available: https://influencermarketinghub.com/growth-of-esports-in-2019-stats/. [Accessed 12 October 2019].

[2] A. Dossey, “4 benefits of mobile app prototyping,” Clearbridge Mobile, 8 Nobember 2019. [Online]. Available: https://clearbridgemobile.com/4-benefits-of-mobile-app-prototyping/. [Accessed 26 May 2020].

[3] D. Tsidoulos, “Comparison of hamburger and bottom bar menu on mobile devices for three level navigation,” 22 April 2016. [Online]. Available: http://www.diva-portal.org/smash/get/diva2:922114/FULLTEXT01.pdf. [Accessed 12 October 2019].

[4] Adoriasoft, “Mobile app design: 14 trendy colour schemes - Adoriasoft - Medium,” Adoriasoft, 1 February 2017. [Online]. Available: https://medium.com/@Adoriasoft/mobile-app-design-14-trendy-color-schemes-2669b5bb77d3. [Accessed 26 May 2020].

[5] A. Parmelee, “How Effective is Facebook Advertising? The Truth About Facebook ROI,” Impact, 28 February 2018. [Online]. Available: https://www.impactbnd.com/blog/how-effective-is-facebook-advertising-the-truth-about-facebook-roi. [Accessed 26 May 2020].

[6] T. E. Observer, “The Esports Observer - We strive to be the most reliable source for esports business information,” The Esports Observer, [Online]. Available: https://esportsobserver.com/. [Accessed 03 December 2020].

[7] LogoDesignValley, “Why mobile app design is important for applications?,” LogoDesignValley, 6 November 2017. [Online]. Available: https://www.logodesignvalley.com/blog/mobile-app-design-important-applications/. [Accessed 29 October 2020].

[8] N. Orr, “RSS Feeds: Concerns about Copyright,” Creativeminds, 28 july 2020. [Online]. Available: https://www.cminds.com/rss-considerations-copyright/. [Accessed 10 December 2020].

[9] statcounter, “Mobile Operating System Market Share Worldwide,” GlobalStats, October 2020. [Online]. Available: https://gs.statcounter.com/os-market-share/mobile/worldwide. [Accessed 3 November 2020].

[10] PandaScore, “Supported videogames,” PandaScore, 2020. [Online]. Available: https://pandascore.co/videogames/league-of-legends. [Accessed 04 December 2020].

[11] Newzoo, “Most Watched Games on Twitch & YouTube,” Newzoo, 2019. [Online]. Available: https://newzoo.com/insights/rankings/top-games-twitch-youtube/. [Accessed 12 October 2019].

[12] Twitch, “everything | Twitch Developers,” Twitch, 2020. [Online]. Available:

https://dev.twitch.tv/docs/embed/everything. [Accessed 04 December 2020].