LAMK
Lahden ammattikorkeakoulu
Lahti University of Applied Sciences

AUGMENTED REALITY FOR COMPANY ADVERTISING

Augmented Reality Application for Levypyörä

LAHTI UNIVERSITY OF APPLIED SCIENCES
Double Degree
Spring 2019
Cayetano Bañón Rubio

Abstract

Author(s)	Type of publication	Published
Bañón Rubio, Cayetano	Bachelor's thesis	Spring 2019
	Number of pages	
	44	

Title of publication

Augmented reality for company advertising

Augmented Reality Application for Levypyörä

Name of Degree

Double degree between EPSA and LAMK

Abstract

The company Levypyörä wanted to make an application in AR for mobile devices, in order to differentiate itself from the competition and show its strengths.

To prepare to the case, the tools for developing AR were examined, and the SDKs available in the market with support for 2D recognition were studied. After this, the development process was started. First, a study of the brand was made, and then the environment for developing (Unity) and the SDK for developing the AR (Vuforia) are chosen. Later, the interactions were added to the application.

The final result of the application is satisfactory for the company. The project is for private use of the company, although, the initial purpose was to release the application for platforms such Google Play and Apple Store.

Keywords

Levypyörä, Augmented Reality, Unity, 2D recognition, Vuforia, LeanTouch, Load Screens.

CONTENTS

1 INTRODUCTION 1
2 AUGMENTED REALITY 2
3 OPTIONS FOR DEVELOPING AN APPLICATION FOR MOBILE DEVICES 4
4 OPTIONS FOR DEVELOPING THE AUGMENTED REALITY PART 7
4.1 Information about Augmented Reality developments 7
4.2 Comparison of different SDKs 7
4.2.1 EasyAR 8
4.2.2 Wikitude 9
4.2.3 ARkit 10
4.2.4 ARCore 11
4.2.5 Maxst 12
4.2.6 Vuforia 15
5 DEVELOPING THE APP 18
5.1 Levypyörä 18
5.2 Restrictions in the project and the goals 19
5.3 Brand Study 19
5.4 Plugin to use 21
5.5 Base of the application 22
5.5.1 Mock-ups 22
5.5.2 Unity as the engine for developing 23
5.5.3 Starting the development 24
5.6 Add Augmented Reality Technology to our application 29
5.6.1 Vuforia as the chosen SDK for Augmented Reality 29
5.6.2 Adding the Company Targets to the Vuforia Data Base 31
5.6.3 3D objects in AR 32
5.7 Interactions In AR 33
5.8 Load Screens 34
5.9 Problems and new requirements 35
5.10 Ending the application 36
6 CONCLUSIONS 38
LIST OF REFERENCES 39
IMAGES LIST OF REFERENCES 41

1 INTRODUCTION

The purpose of this Thesis is the explanation about what tools are used to create an Augmented Reality application for the company Levypyörä. Levypyörä is a company that specializes in manufacturing of wheels for heavy machinery, and in the design and development of steel structures. They want an application using cutting edge technology. The application must show their strength, their quality and the possibility of customizing the wheel using Augmented Reality. The purpose of this application is using it, in fairs and exhibitions, as complement that they offer to the customer and as differentiation over the competitors.

Creating one application that meets these requirements is a hard work. For this, an analysis of the different options for developing application for mobile devices and about the different options available for generating Augmented Reality will be exposed in the Thesis. An application with Augmented Reality necessarily works with 3D objects, such as the wheels ring. These will be provided by the company during the project.

One of the goals of this application is releasing the application in Google Play, Apple Store or in the company's own web. The development will be carried out focusing on this goal, looking the next steps.

In the first part of the thesis the Augmented Reality is explained, as well as the options for developing in Android and iOS and a comparison of the different SDKs. In the second part of the thesis the goals and restrictions of the project and development process are exposed.

The whole project will be realized in three months. For this reason, within the document, time will be one of the important constrains. Seeing how this and other variables took a lot of importance and limit the project features.

2 AUGMENTED REALITY

The necessary technologies for developing with the Android operating system (OS) will be studied in this Thesis, as well as how to integrate Augmented Reality and the new functionalities it provides.

Before going in the depth into any of the chapters it is important for familiarize with Augmented Reality. It is an improved version of reality where elements or environments of the physical world are complemented with images or superimposed elements generated by computer over the view of the user of the real world.

Image 1 Pokémon Go (muropaketti.com, 2019)

Even though AR is relatively new, it is used more and more every day, from entertainment, like the well-known app created by Niantic and Nintendo “Pokémon Go” (Image 1), to its use in the business world. Creating applications that help employees, for example by interactive tutorial with AR, more efficient than conventional tutorials since the employee can watch the system working and virtual objects can explain or perform actions that the user would do (Image 2).

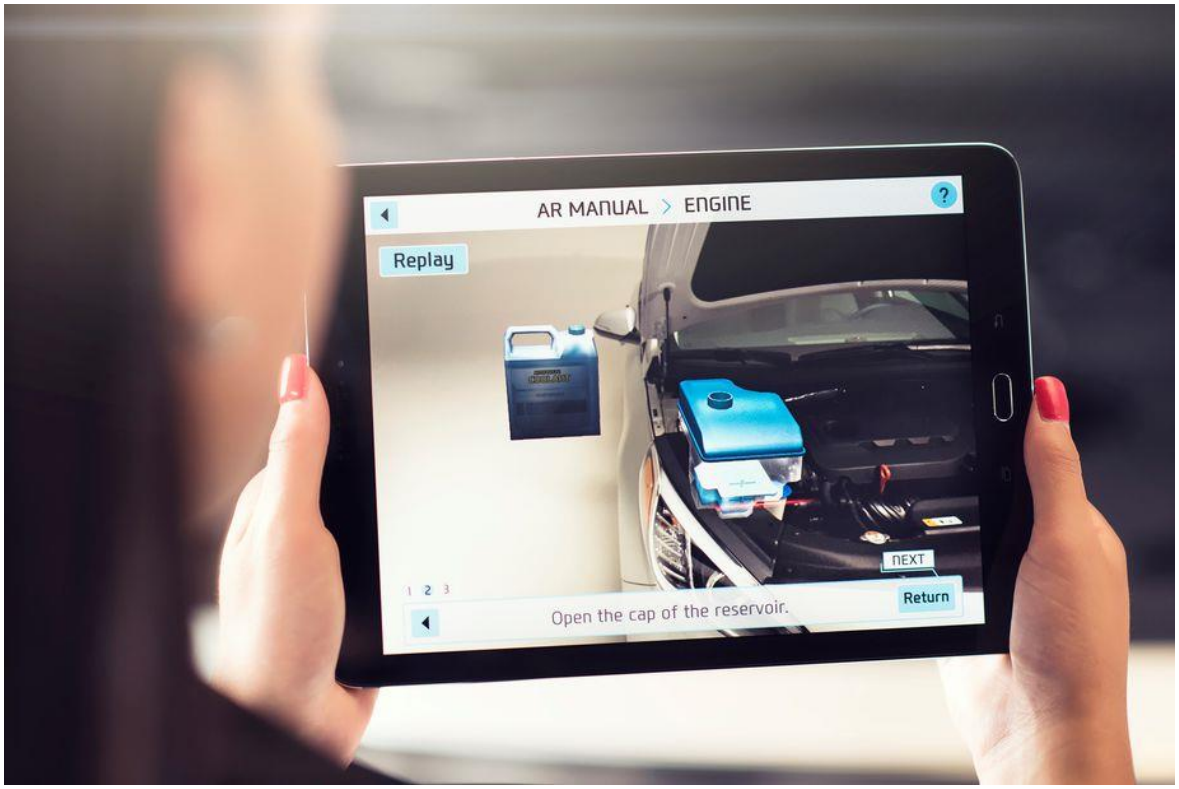


Image 2 Augmented Reality Tutorial

Nowadays, with the possibilities for developing AR easier than before, this technology is making its way into other fields like advertising and marketing, because it allows the customer to get a closer view of the product.

Augmented reality allows visualizing advertised products in real environments like our home. One of the apps known for this is IKEA's AR app in which you can visualize how their products look in your home (Image3).

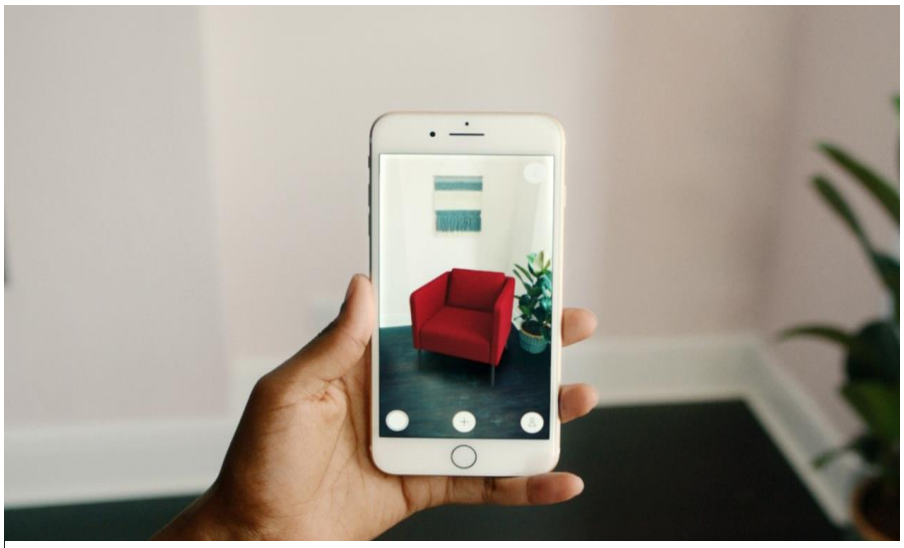


Image 3 Ikea Augmented Reality application

3 OPTIONS FOR DEVELOPING AN APPLICATION FOR MOBILE DEVICES

In order to develop applications for mobile devices a software development kit (SDK) is needed. It helps to understand and communicate with the OS.

There are many SDKs available for developing Android applications, because it is the most widely used operating system. Some of these SDKs are distributed by Google Inc., but you can find other distributed for third parties, because nowadays Android is an open-source operating system. Android Native or Flutter can be one of the example, for developing Android applications (Image 4). (Flutter 2019.)

Image 4 Android and Flutter Logo (2019)

Not

All the SDKs have the same characteristics and not all have a similar programming language. Each SDK has its own characteristics, and some are better than others, depending on what you use it for. For example, developing an application using Native Android gives the best performance, but development in Native Android is tedious and the development time is longer than when using other SDKs, which causes that on my occasions some other SDK is chosen.

For developing an app, it is always recommended to use an SDK. However, there are ways for increasing the abstraction and making the developer´s job easier while using the SDK. Android Studio is an official integrated development environment, which means that Google Inc. supports this development platform and keeps it updated and with the maximum compatibility with the SDKs. We can use this environment, for example, for speeding up the development time because Android Studio shows errors in the program execution. It notifies errors in the code, as we can see in Image 5.

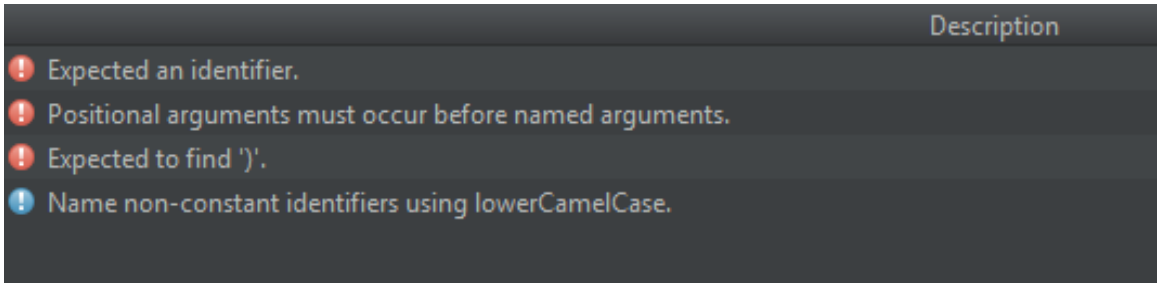


Image 5 Console Notify a failure (Android Studio running a Flutter app, 2019)

Like it was told before, there are other development environments that have not been developed by Google Inc. One of the most widely and best known is Xamarin and its environment Xamarin Studio. Xamarin is environment that offers to the developer some templates for streamlining the process of development (Image 6). (Xamarin 2019.)

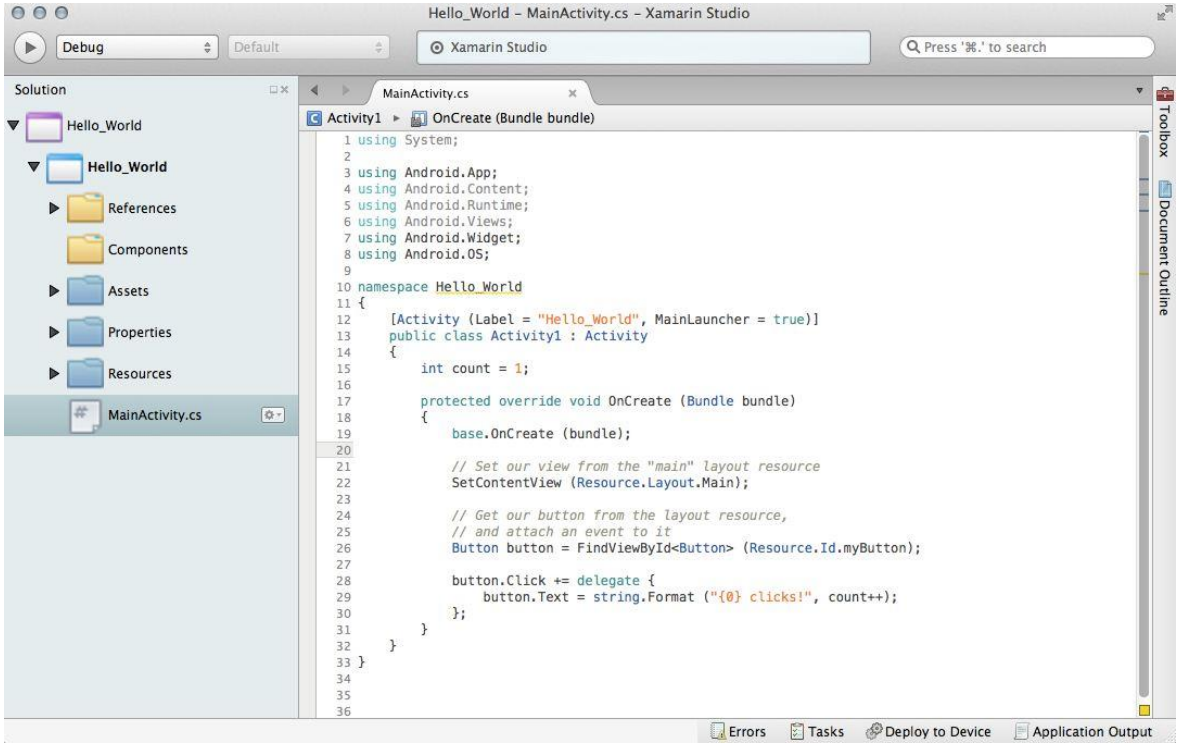


Image 6 Code of Hello World create in Xamarin Studio (2019)

With all the SDKs and development environments mentioned above, and with others that were not mentioned, an Android application for mobile devices in 2D and 3D can be developed. Nevertheless, developing 3D applications in a 2D environments is complicated because the environment is created specifically for 2D applications. For this reason, you can also developing on other platforms like graphic engine, even though it is not created specifically for creating applications outside the videogames world. Using a 3D engine makes things easier for the developer's work with objects in 3D, for example, to create the interactions between them. It also has a 2D part for developing that can be used. These

advantages make 3D engines a common choice when developing applications with 3D content.

The common engines for developing applications with 3D content are the same that are the most popular nowadays, for developing videogames. They are Unreal and Unity Engine (Image 7).

Image 7 Unity and Unreal Logos (eliassoftware.com, 2019)

Easy interfaces like hubs in a videogame, can be developed with these graphic engines, because they were created for this purpose. Test in real time without compilation, for testing and checking the applications is one the characteristic that Unity give, for example (Unity 2019a). These characteristic are useful tool for the developer, since with them you can decrease the testing and checking time of the application. These engines also brings the possibility of switch the developing platform and testing for other platform. Windows, iOS, Android, e.g. are some of the platforms available for releasing and developing with these engines.

However, these engines have problems too, the applications with them are slower and consume more resource than the applications developed using a specific SDK for app developing. Being able to compile for so many platforms, make not well optimized for the development of mobile application. They are a good example that to cover a lot of leads to not being able to very efficient.

Choosing a good tool for developing is not an insignificant or easy task. Many variables have to be taken into account. Satisfying the requirements, compatibility with the Augmented Reality SDK and compatibility with extra tool have to be considered.

4 OPTIONS FOR DEVELOPING THE AUGMENTED REALITY PART

4.1 Information about Augmented Reality developments

A specific SDK is needed for developing the Augmented Reality part. Developing for visualizing computer object in the real world, using targets, is facilitated by the SDK.

Nowadays, for Augmented Reality project many SDKs can be used, which different characteristic and improvement. For this reason, knowing more about them and which are closer to the requirements is needed. Integration with the environment, 2D Target recognition and useful tools are the application requirements in the case.

While in 2010 still existed SDK for Augmented Reality with free source-code. These were disappearing progressively, either by license changes or when purchased by another company. Big companies as Google Inc., Apple Inc. or Facebook was purchase a few of the companies that developed Augmented Reality SDKs for creating his own. Other companies, instead, changed his license and you have to pay if you want to use it for advertising or if you don’t want to see their watermarks as EasyAR did.

4.2 Comparison of different SDKs

For the aforementioned, it is necessary to know more about what SDKs are available nowadays, which are the most important and known. A short introduction to the SDK, the different recognition system and the license from each of them will be explained in the next chapters.

4.2.1 EasyAR

EasyAR is a SDK develop by VisionStar Technology (Company based in Shanghai), this SDK became known for being free in its firsts years.

EasyAR can be used with different programming languages as C++, Java, Swift and Objective-C (lasts are for iOS), and other environment as Unity. Depending of the license, different recognition software are available. For example, 3D recognition is only available in the pro license, however 2D recognition is in the free license and in the pro license.

The first release was at the middle-end of 2012, however, until the year 2018 it did not have a complete English API, the documents about this SDK and how to use it was written Chinese. This fact complicated start to develop AR applications with it. In Unity, the SDK integration is doing using unitypackage. The objects are distributed in prefabs which facilitates its use (Image 8). There are examples in its web page. It can be used for studying how to develop a base example with this technology. (EasyAR 2018b.)

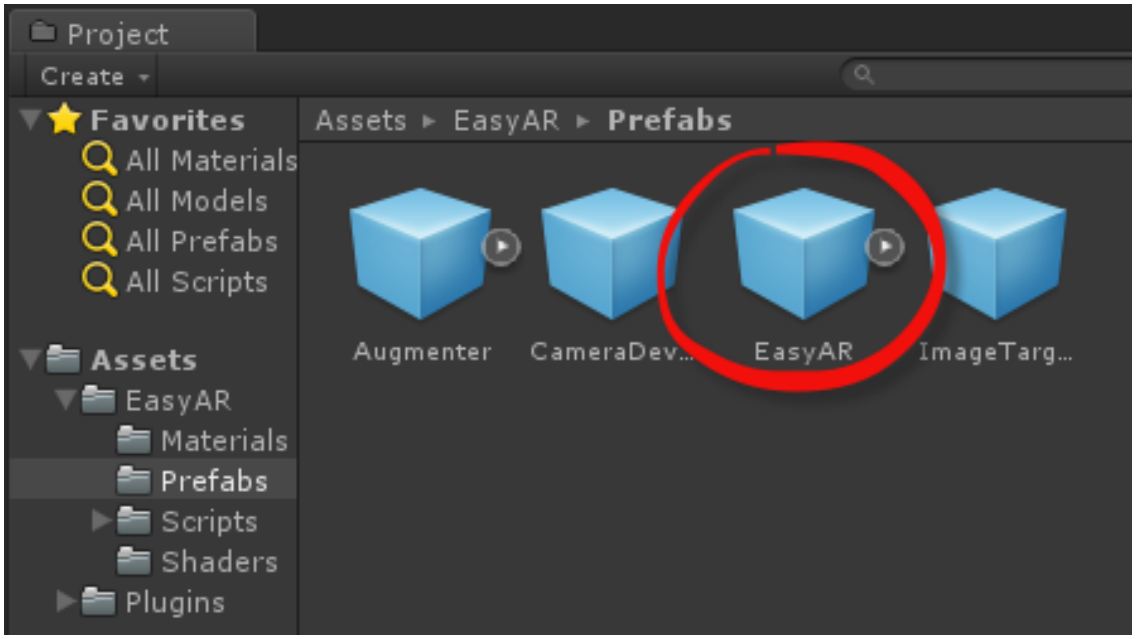


Image 8 EasyAR prefabs in the Unity Editor (EasyAR.com, 2019)

Although having the examples, starting with this SDK is, in fact, complicate. The information of the API is not enough for starting. As the English API is pretty new, getting information by blog or similar is elaborate and the used of the examples are complex, because it is necessary to fix some problem before to start. It is a hard work when you don't know how work the SDK yet.

Nowadays EasyAR is not free anymore. However, it is one of the cheapest SDK for marketing and advertising purpose without watermark. The closer license to the requirements is the standard license. It is $499.00 and the pay is unique. (EasyAR 2018a.)

4.2.2 Wikitude

Wikitude was founded in 2008, its initial purpose was the Augmented Reality location. Along the years its initial purposed was changing until the year 2012. In this year the company released its SDK for the development of AR applications. With this first SDK, it could use targets recognition, tracking and geolocation.

Nowadays, this SDK is the most expanded, talking about different platforms. It is available for Android, iOS, Windows, Unity, Cordova, Xamarin, Titanium, React Native and other, it can see in Image 9. Also, Wikitude has the option for developing with new technologies as the known Smart Glasses brands in the market. (Wikitude 2018b.)

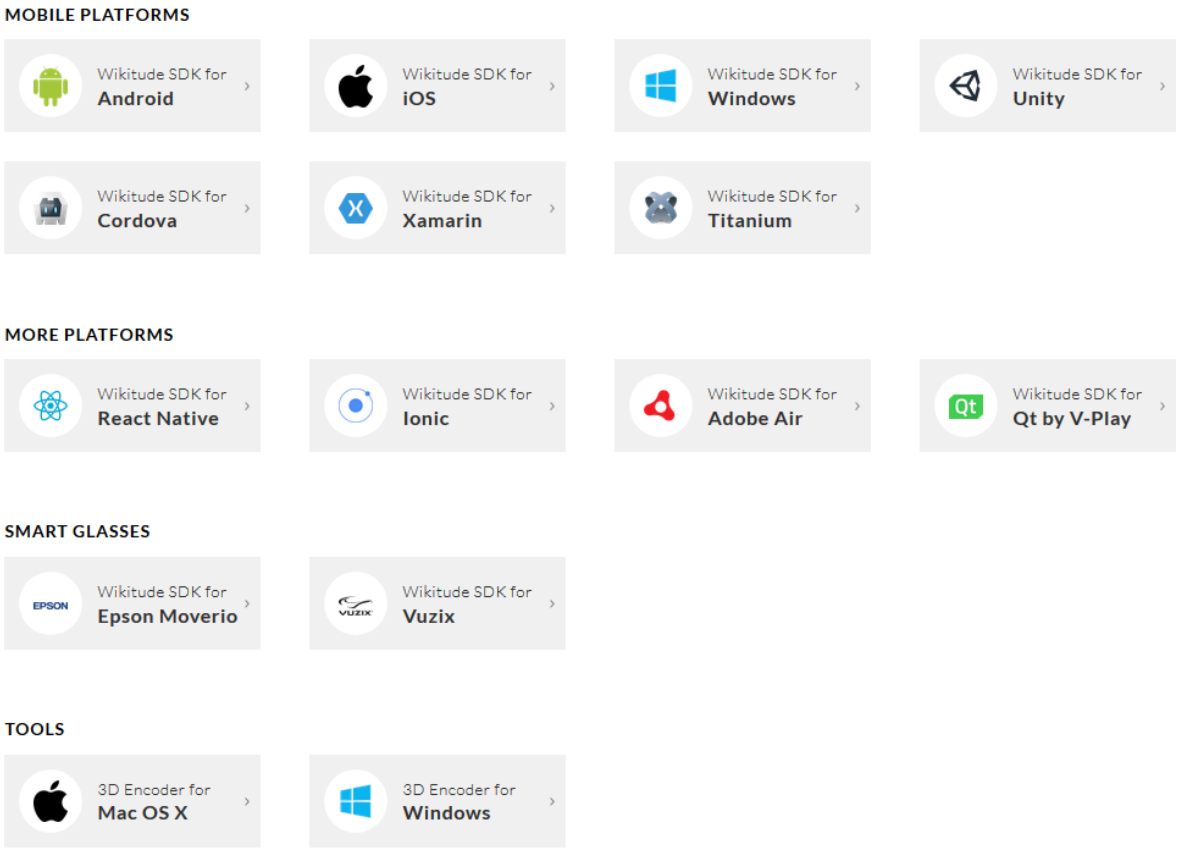


Image 9 all the platforms available for development in Wikitude (2019)

An extensive documentation and in detail for all the platforms available is had by Wikitude. Also, it has a free forum in the Wikitude web page, where the people can join and ask question. The community or the own Wikitude workers usually answer soon. In this

SDK, the help to the developers happens only in the forum, as the only help regardless the type of license. (Wikitude 2018a.)

Just like in the previous SDK, Wikitude can be integrated with Unity adding to one project using unitypackage. Also, it is organize on prefabs that doing easy to understand how to work with it. Thanks to the organization and the API starting to develop with Wikitude is relatively easy. Even though you don't have a lot of tutorials of outsiders, how they decided to organize and created the SDK are as the developer will be expect and understand. Also have a course on the Wikitude page for beginners.

The optimum 3D recognition, the documentation and the possibility of multitargets recognition do Wikitude one of the most powerful and complete SDK. These reasons also make it the most expensive SDK for developing AR. Different payments methods are had by the licenses. One of them are the annual payment for 2490€, that have the same advantages that the unique payment for 1990€. Wikitude has also a free license for developer with watermark, with this license 2D and 3D recognition can be used. Wikitude has the StartUp license free to use for companies with less of two years and that all the shareholders are natural persons. (Wikitude 2018c.)

The only requirement is this project is the 2D recognition. In the future if the project to require, using 3D recognition or other type or recognition, could be possible, buying other license.

4.2.3 ARkit

ARkit is the SDK developed by Apple Inc. for the development of Augmented Reality for the own brand devices.

ARkit is designed for developing using Swift and Objective-C that are the programming language of the brand, but also it can work with other environments as Unity. (ARkit 2018a.)

2D and 3D recognition can be used for developing with ARkit. Focusing first in its brand, ARkit has one of the most quality SDK for AR development. Apple try be the reference in the AR world. For this reason, a good Smart Terrain Augmented Reality was got in the first years. So with this and the 3D recognition, making to ARkit as a good choose for developing AR.

For starting the development is mandatory to have the brand devices, because the testing part is one of the most important parts in a project. However, the Apple devices are expensive devices and not everyone has one. And these are one of the beginning problems. Other of the problem for starting with ARkit is how to obtain the development license for Apple that is a tedious work, if you do not have apple devices. (ARkit 2018b.)

4.2.4 ARCore

ARCore is an SDK developed by Google Inc., the official release was the first of March of 2018. As the SDK is supported by Google quickly rose popularity. This SDK was created for compete versus the SDK previously mentioned (4.2.3). But in the opposite way that its competitor with it is possible to develop for Android and for iOS easily. (ARCore 2018a.)

For developing Augmented Reality, it is possible to do using its own developing environment as Android Studio using Native Android or other platforms like Unity and Unreal. It is also possible to develop in the iOS environment. (ARCore 2018b.)

ARCore has a big community from which is possible to learn easily. The previous advantage and the good descriptions in the API help to the beginner, doing possible to learn the basic of the technology very quickly. This SDK was created for developing applications in Unity, for this reason has the most detail documentation that other platforms or environment don’t have it. As the previous SDKs, it is available using packages, and also it has remote repository. The remote repository is a good option for the developer that want to watch the changes in the software and how these changes can affect to their software.

As the others, it uses prefabs for working with the environment, as we could watch in the picture number 10. (ARCore 2018c.)

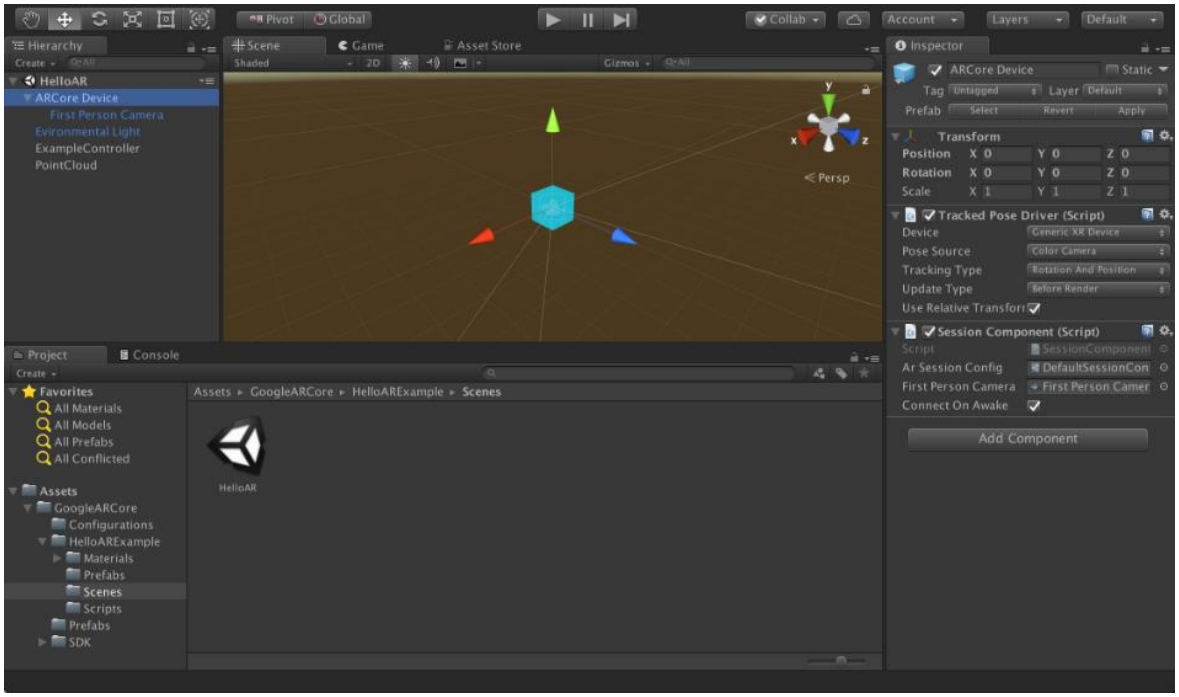


Image 10 Unity Editor (2019)

ARCore has a lot of recognition tools, nevertheless for developing with the more advances tools it is necessary to have special mobile phones. This happen, for example, with the smart terrain recognition.

Use ARCore as development tool nowadays is totally free, however also probably that Google Inc. change it in the future. For example, in the past all the functionalities of Maps was free. However, when the community improve the utilities, Google change the license and for use all the utilities is necessary to pay.

4.2.5 Maxst

Maxst was created in 2010, then, in 2011 the first AR application was released. However, until the year 2012 the SDK for public use was released but only in Korea. The list countries that can use the SDK grew along the years. (Maxst 2018a.)

Maxst can be used in the most common environments for developing, like Unity and Android Studio, also the iOS environment as happens to all its competitors. However, the documentation of Maxst is worse than, for example, Wikitude and ARCore. (Maxst 2018b.)

Even though Maxst is in all the countries since 2016 like Augmented Reality's SDK. Corporations as HYUNDAI, KIA, LG and SKTelecom endorse it and have applications using it (Kim, M 2017). Nevertheless, finding good materials for starting to develop is complicated. These reasons, making Maxst a hard SDK for beginners.

Image 11 AR Guides using Maxst (Maxst/#/en/casestudies, 2018)

After the deep study of the SDK, it is specialized in AR Guide's development (Maxst 2018d). This technology is usually use for helping the user with some task as mechanisms repair (Image 11). Which it is not the purpose of this application.

There are a lot of different license type available. As the others, this SDK has a free license with watermark for developing. Advertising and marketing purpose are not allowed with this type of license. For advertising purpose, it is necessary to buy other type of license. In the table1 it is possible to see the different license and what it includes. The number of downloads and the number of targets are one of the restriction depending of the license. (Maxst 2018c.)

Table 1 Data of different license of Maxst (maxst.com/pricing)

	Free	Pro-One Time fee	Pro-Subscrip-tion	Enterprise
Price	0 usd	499 usd	599 usd/year	Contact with Maxst
License type	Free(Water-mark)	One-time pay-ment	Yearly sub-scription	Custom
Eligibility	Non-commer-cial use only	App with less than 100K downloads	App with less than 100K downloads, than require massive target Images and that require continuous up-date of AR functions	App with over 100k down-loads. Private in-house app for internal use only Pre-installed app on a de-vice
Functions	Instant Tracker Visual SLAM Object Tracker Image Tracker Marker Tracker QR Code Tracker QR/Barcode Reader	Instant Tracker Visual SLAM Object Tracker Image Tracker Marker Tracker QR Code Tracker QR/Barcode Reader	Instant Tracker Visual SLAM Object Tracker Image Tracker Marker Tracker QR Code Tracker QR/Barcode Reader	Instant Tracker Visual SLAM Object Tracker Image Tracker Marker Tracker QR Code Tracker QR/Barcode Reader
Platform	Unity 3D Android iOS	Unity 3D Android iOS	Unity 3D Android iOS	Unity 3D Android iOS Windows macOS Smart glasses
Support	Forum	SDK Updates NOT Included Forum	SDK Updates NOT Included Forum	SDK Updates NOT Included Forum Email Conference call Code Review (optional)

Increasing the number of developer that work with Maxst is one of the brand objective. For these reasons, discounts in the Pro-One time fee license is possible, as it seen in Image 12.

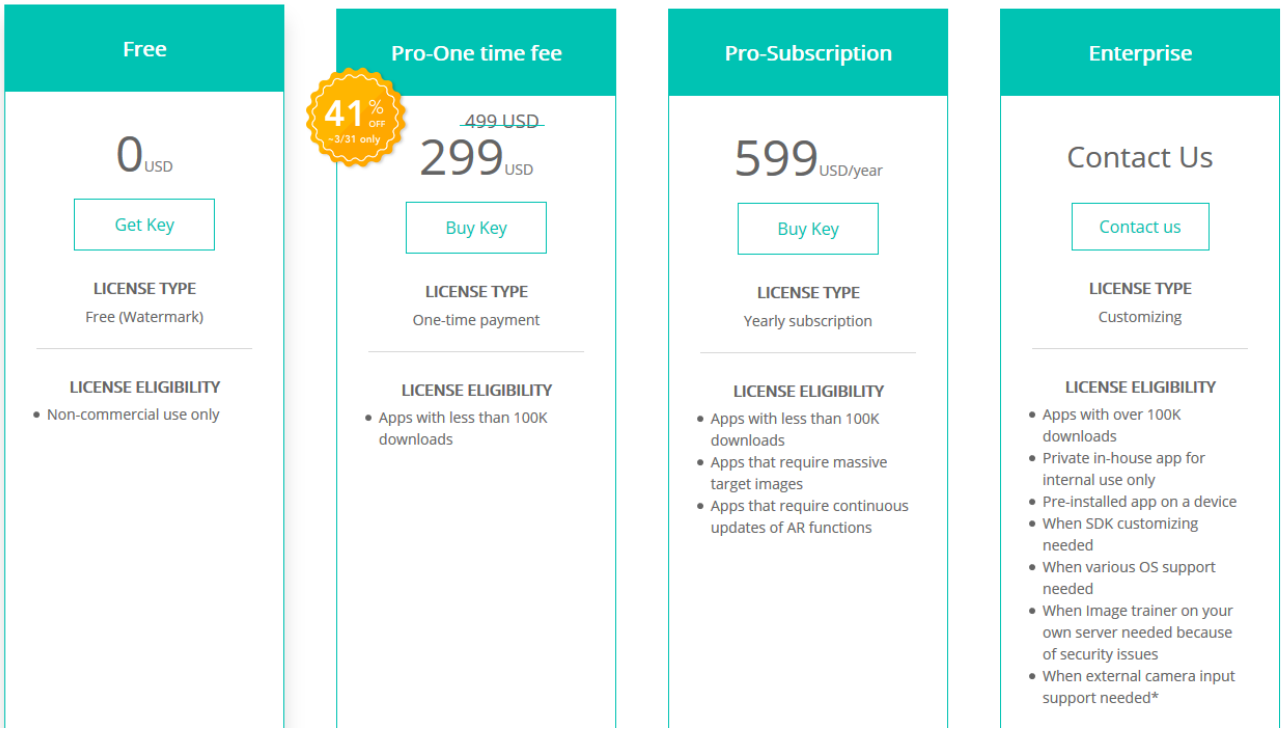
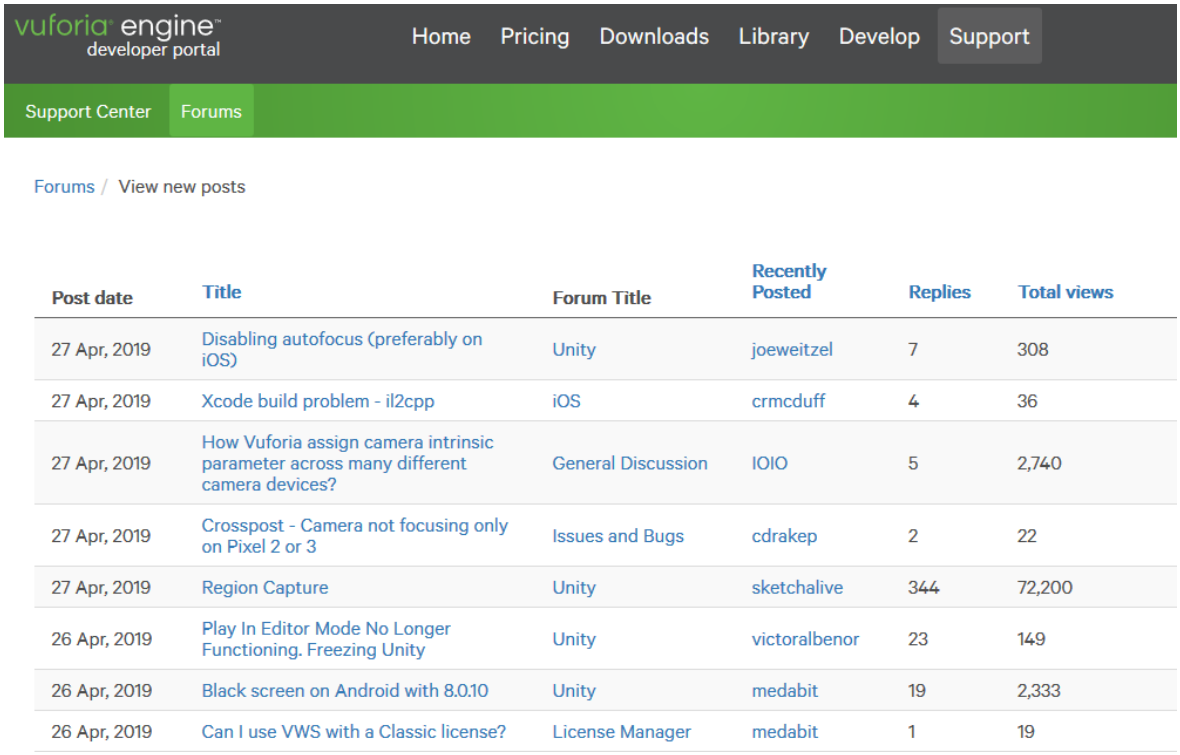


Image 12 Example of the discounts available in some special occasions (Maxst.com/pricing, March of 2019)

4.2.6 Vuforia

Vuforia was released in 2011 as SDK for Augmented Reality projects. Until one year later, Vuforia was not used and known for the developers. The number of developer that used was increased before the integration with Unity because it had a free use in all the purpose, but the real boom was after the Unity integration. After the bought of PTC Inc. the licenses changed and nowadays is not free for marketing or advertising purpose (Vuforia 2018a). However, it is continued use by the developer for its complete integration in the new versions of Unity (Image 14) and for the improvement that Vuforia did in the 2D recognition. Nowadays is the most use for the Unity developers.

Actually, work using Unity and Vuforia is simple and handy. Thanks to the big community of developers that work with it. The active forum and the possibility of approach with the different tutorials that exist is doing Vuforia one of the easiest tools for the beginners (Image 13).



Post date	Title	Forum Title	Recently Posted	Replies	Total views
27 Apr, 2019	Disabling autofocus (preferably on iOS)	Unity	joeweitzel	7	308
27 Apr, 2019	Xcode build problem - il2cpp	iOS	crmcduff	4	36
27 Apr, 2019	How Vuforia assign camera intrinsic parameter across many different camera devices?	General Discussion	IOIO	5	2,740
27 Apr, 2019	Crosspost - Camera not focusing only on Pixel 2 or 3	Issues and Bugs	cdrakep	2	22
27 Apr, 2019	Region Capture	Unity	sketchalive	344	72,200
26 Apr, 2019	Play In Editor Mode No Longer Functioning. Freezing Unity	Unity	victoralbenor	23	149
26 Apr, 2019	Black screen on Android with 8.0.10	Unity	medabit	19	2,333
26 Apr, 2019	Can I use VWS with a Classic license?	License Manager	medabit	1	19

Image 13 Vuforia Forum(Vuforia.com/forum, 28 of April 2019)

As is said in the previous paragraph, nowadays Vuforia has a complete integration with Unity. So much so, that Vuforia can now be accessed in the same way that 3D objects are accessed (Image 16).

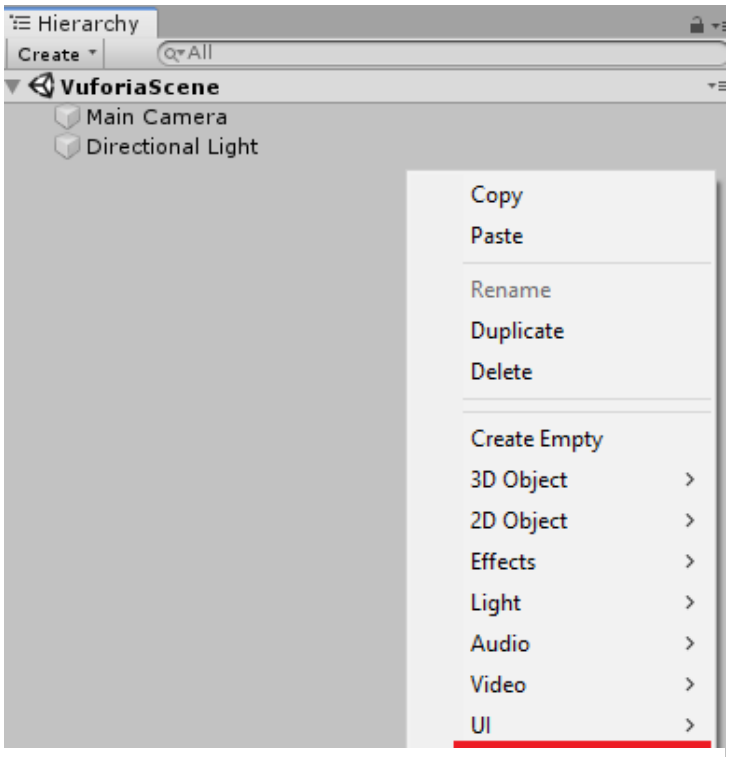


Image 14 Unity Editor screenshot (2019)

The complete integration with Unity give to the developer a handy interface and good error and warnings notification and, of course, this integration is supported by PTC Inc.as it seen in Image 15 (Vuforia 2018c). This is another reason why it is so popular between the developers.

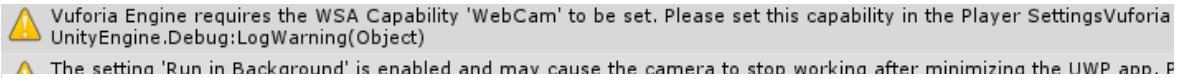


Image 15 Unity console with Vuforia warnings (2019)

Just like in the previous SDKs mentioned, Vuforia has a different type of license depending on the use that the developer can do with it. More or less the price is similar to the competitors. Having different type of license: free license with watermark, 499 usd for one-time license or 99 usd per month, if you want more utilities as cloud data base. Vuforia only has 3D recognition in pro license. (Vuforia 2018b.)

5 DEVELOPING THE APP

5.1 Levypyörä

Levypyörä is a company that specializes in the manufacture of wheels for heavy machinery, and in the design and development of steel structures (Image 16).

Image 16 Levypyörä Logo (orfer.fi, 2019)

With over 60 years of history, they continue working to always be with the latest technology, not only in the field of steel. For this reason, observing the boom of new technologies such as Virtual Reality and Augmented Reality, the aim to improve their marketing and be closer the customer. With this, they are showing that are still a young company adapted to the latest technologies.

In the application developed in this thesis, they want to show to the world their strengths, their quality and the possibility of personalizing the wheel, adapting to your needs even concerning the appearance. All the points that the competitors cannot.

An AR application can be used for many purposes. In this case, the aim is to develop and application for commercial and advertising purposes for Levypyörä, in order to stand out from the competitors.

The company wants an application that is easy to understand and interact with, in which they can show their improvement and their aesthetic options, such as the possibility of choosing the ring wheel colour. They thought that the best option for showing all the requirements previously mentioned, is to develop the application for mobile devices, like smartphones or tablets. Looking for the greatest number of devices that can use and test the application, the obvious to option is to develop for Google OS (Android) devices. However, the company wanted to include other devices, such as Apple devices, for export reasons.

5.2 Restrictions in the project and the goals

Like in the vast majority of the project there are restrictions that must be taken into account. In this case, one of the restrictions was the time for developing the app. The company wanted to present the app in the one of the fairs that they have in the end of February. The first meeting was in November. Between November and February the tasks were: to do a brand study, choose the technology for developing the Android part (chapter 3), choose one SDK for developing the Augmented Reality part (chapter 4) and develop a complete useful application. The second restriction is the license cost for used the SDKs, because a large investment in not justified until it is known if the technology will meet expectations.

The software prototyping technique was used to establish the goals, which were:

1. Do a brand study for the first week of November.
2. Study the technologies for developing an Android application with AR in two weeks.
3. Finish one demo for the end of December and have a meeting to get feedback.
4. Rebuild the application considering the feedback given.
5. Improve the application until the estimated presentation day.

The second goal was not simple, because as you can see in chapter 3 and 4, nowadays, there are a lot of different SDKs for developing Android applications and there so also a lot of different SDKs for Augmented Reality purposes. It is important because it has o some SDK that have the possibility for developing in the estimated time and with the chance for releasing an iOS version.

After introducing the goals and what the company expected from the application, in the next chapters will explain how the different parts of the application were built, what SDK was chosen, how to add the AR part in an application and the last improvements.

5.3 Brand Study

For developing an application, it is necessary to know how the company usually shows to the customer. For this reason, the brand the company uses on its website was studied (https://levypyora.fi/en/).

They usually use different grey, black, white and red colour tonalities, as we can see in the pictures 17 and they also use the same colours for their specific typography (Picture 18).

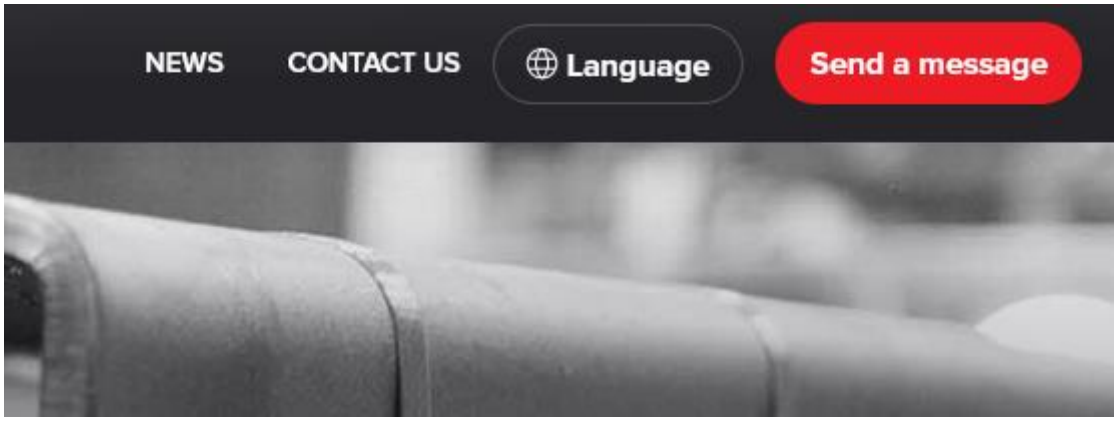


Image 17 Colours of the brand (levypyora.fi)

They usually use rounded buttons with the colours of the brand for the interactive buttons. Also for dividing the different parts to the web page they use squares with different colours, depending the context. (Picture 17)

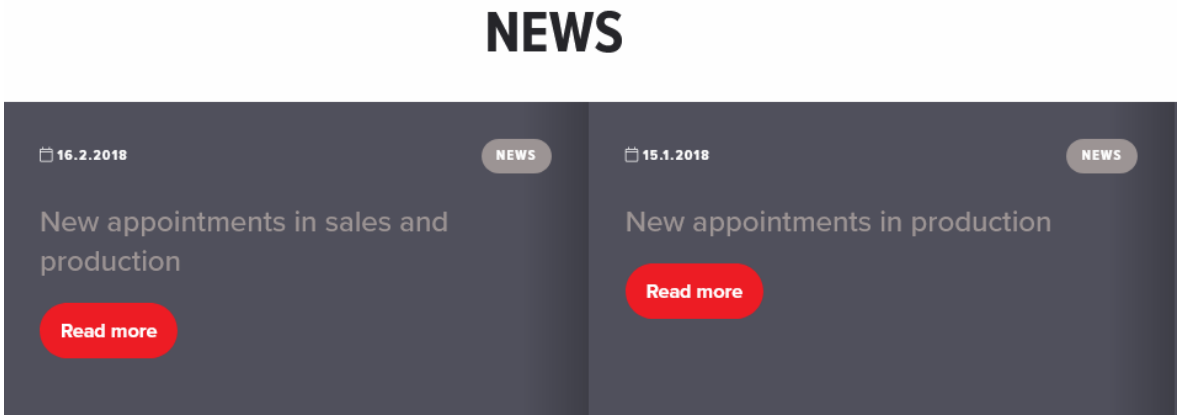


Image 18 Example of typography and how to divide content by using squares. (levypyora.fi/en/)

After studding the company colours, the typography and the way that they usually implement their web projects, it was now possible to start the development of the app being more faithful to the brand.

5.4 Plugin to use

Developing all the interaction possible is a hard work, and for this reason there are different plugins that can be used for improving and optimise the development time. This project used three different plugins for speeding up the development time.

DoTween

DoTween is one of the plugins. It is used for creating animation using code and not using the animation window that Unity has. With DoTween it is possible to soft some animation and create animation without a start point, using the object position as the start point. That means that if the start point of the object is modified, it is updated at the same time.

The DoTween plugin was only used in this project in the Gallery View to create the slider as we can see in Image 31, so that you do not need a start position like in the normal Unity animation. With DoTween you get a soft result, which it is friendly and comfortable.

LeanTouch

LeanTouch is another of the plugin that was used in the project. Lean Touch is useful for screen clicks, because it alone takes care of all the control of the touchpad, and the developer only has to control what objects are or are not selectable. Lean Touch uses raycast and collider for detecting what the user is touching. Developing the functionality on your own it is possible, but nevertheless it is a hard work and time was limited. For these reasons this plugin was used.

LeanTouch can be used for different purposes, but in this project, it was used only for the Gallery Scene, with DoTween for creating the slider, and of course in the AR Scene for controlling what the user clicks and also for rotation of the wheel.

TextMesh Pro

TextMesh Pro is a plugin created by the Unity Company. It was created because the normal text of the UI interface on some occasions was a bit sparse, and with it you cannot create a complex text interface. In the past it was not integrated in the Unity Editor, but nowadays it is totally integrated, and you can use it, in the same way that you use a normal text, but with more suitable tools.

TextMesh Pro is recommended to be used for all the texts that exist in one application, and in the project, it was used for all the text, especially in the Gallery View, which is almost all text.

After choosing which plugin to use and deciding to use LeanTouch for the AR interactions, it was time to start developing it.

5.5 Base of the application

5.5.1 Mock-ups

The base of the application is the User Interface (UI), and how the different views are interrelated. In the UI part, the company wanted an easy and understandable main view, so a gallery view was added, presenting the wheels in a conventional way, with a picture of the wheel and a descriptive text.

In the AR´s interface, for my experience an interface that overwrites the camera view, it is preferred to use the simplest UI as possible, in order to have the view clean for the AR.

After starting to think how to organize the UI, it is necessary to think whether to develop the application in landscape, portrait or both. In these cases, it was thought that it would be better if the application is developed only in landscape, because then you only need to think in one direction, and to interact with a big 3D object in AR, it is better to have more available space. This makes the development faster.

For these reasons, two different mock-ups were made:

- This mock-up is serious and static with well-defined company colour, for the main and the gallery view. Less is more (Image 20 right)
- Unlike the other mock-up, the idea was to have something new and different that do not expect when you click on the application. There are 3D objects in the background that represents one of the rooms that already exists in the factory, trying to merge the company, the customer and the factory. (Image 20 left)

Looking at the different mock-ups (Image 19), the first option is better for this project, because it is more serious, as the brand wants. The other reason for choosing the first option is that in the Gallery view the text is more readable without the wall and the machine in the background. This could be better user experience.

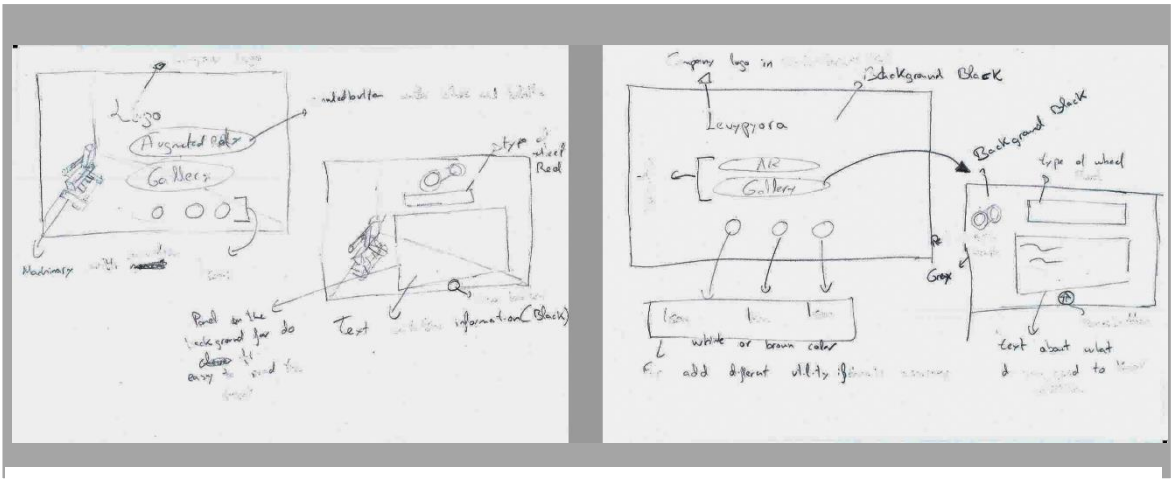


Image 19 Mock-up two and one (2019)

5.5.2 Unity as the engine for developing

Existing a clear idea about how we want to build the interface, it was time for choosing a platform for developing the application. Since time is one of the restrictions at the project, the chosen platform must satisfy the requirements, time, the ability for release for iOS easily, if it desired, and be available for using AR.

As mentioned in the chapter 3, there are a lot of SDKs, environments and engines for developing Android applications. However, it is necessary to focus on only one of them. For choosing the best option available, three of them were analysed: Unity, Flutter and Android Studio. (Table 2).

Table 2 Analyse of 3 Android environments for developing

	Unity	Flutter	Android Studio
Easy to learn	4	4	2
Integration with AR	5	0	5
Previous Experience	5	4	3
Ability for release for iOS	4	4	4

Work with 3D Objects	5	0	0
Possibility of test in the moment	5	5	4
Result	28/30	17/30	18/30

After the comparison, because of the integration with AR, the possibility for working with 3D objects and my previous use experience, Unity was chosen as the environment for developing the application.

5.5.3 Starting the development

Best practices

Unity has many different versions, and some of them do not support the upgrades or downgrades version very well. For that reason, it is important to choose a stable version and work with it and try not to change versions. In the Levypyörä application, version 2018.3.11f1 was used, because it is a stable version and it was one of the newest versions when the project started.

It is necessary to have a good organization folder in the project, because all the plugins that will be used are going to write their folders in the Assets folder. The plugins folders are variable folders because they can be necessary or not at the end of the project. Usually, a folder with statics assets which are included by the programmer is created in the assets folder, as seen in Image 20.

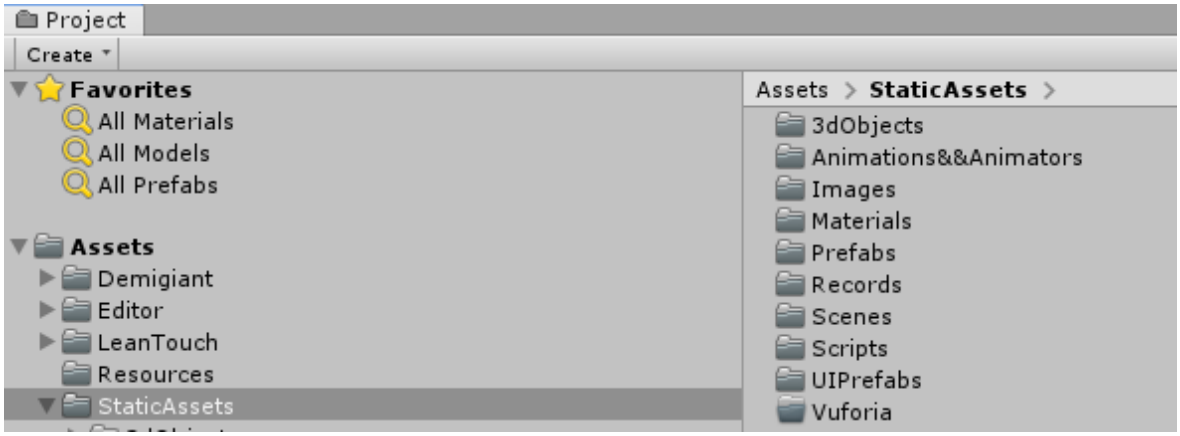


Image 20 Good practice folder organisation (2019)

By default, when a Unity project is created, it is always created for PC, Mac and Standalone platforms. Changing to the platform that will be used for the application is one of the best practices. It's possible to do it in "File/Built setting…/Platform", where some basic settings can be modified. (Image 21)

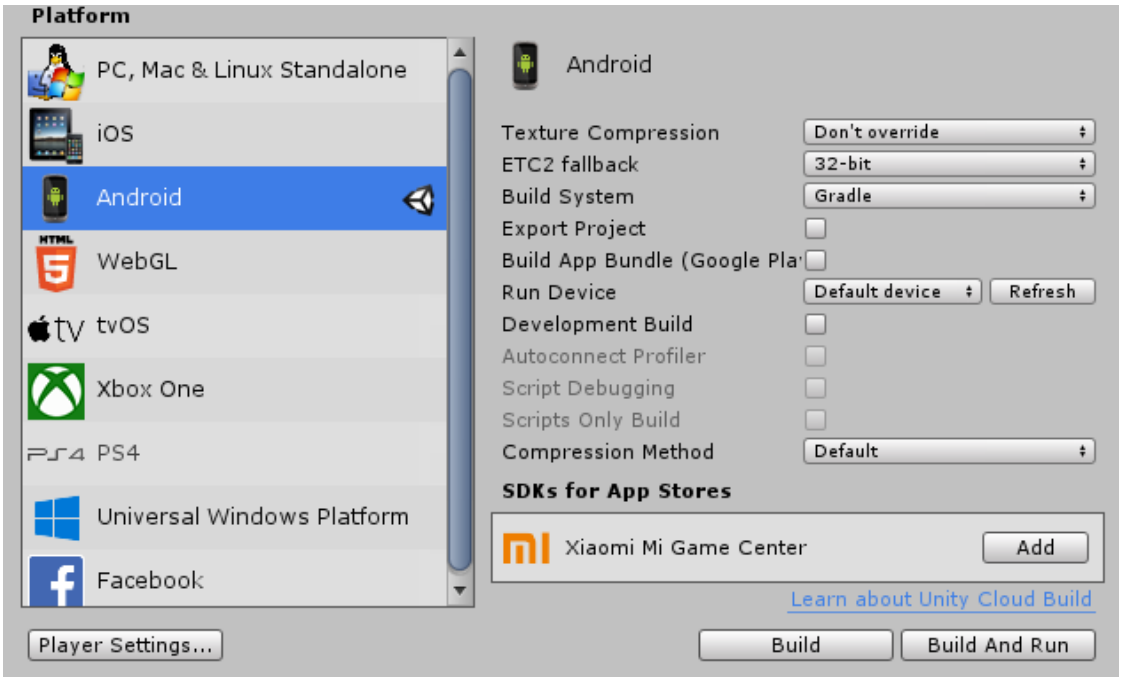


Image 21 Change the platform in a Unity project (2019)

Main view

Now it was time for starting the development of the app. The first view that it was mandatory to build was Index view or Main view. It has to be as close as possible to the solution presented in sub-chapter 5.4.1.

The object that Unity uses for rendering 2D UIs, is the canvas object. Some options in the canvas have to be changed in the canvas scaler, if we want a responsive canvas. The first step is changing the option "constant pixel" to "Scale with Screen Size", and the second step is changing the Match to 1. (Image 22)

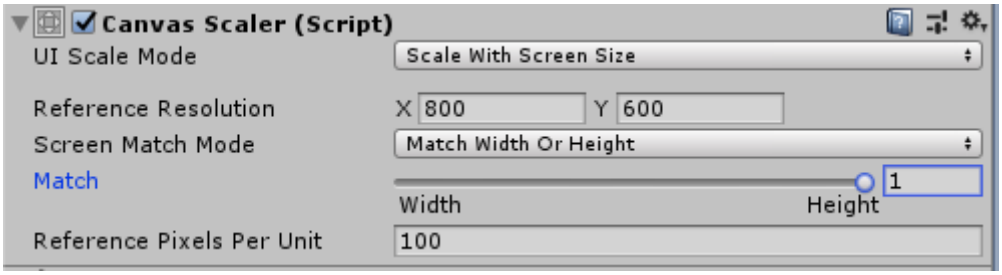


Image 22 Canvas Scaler Option (2019)

After the canvas had been configured in the correct way, the next step was to add the buttons and the images (Image 23). The black background, the rounded buttons and logo, are added as we agreed in the picture 19.

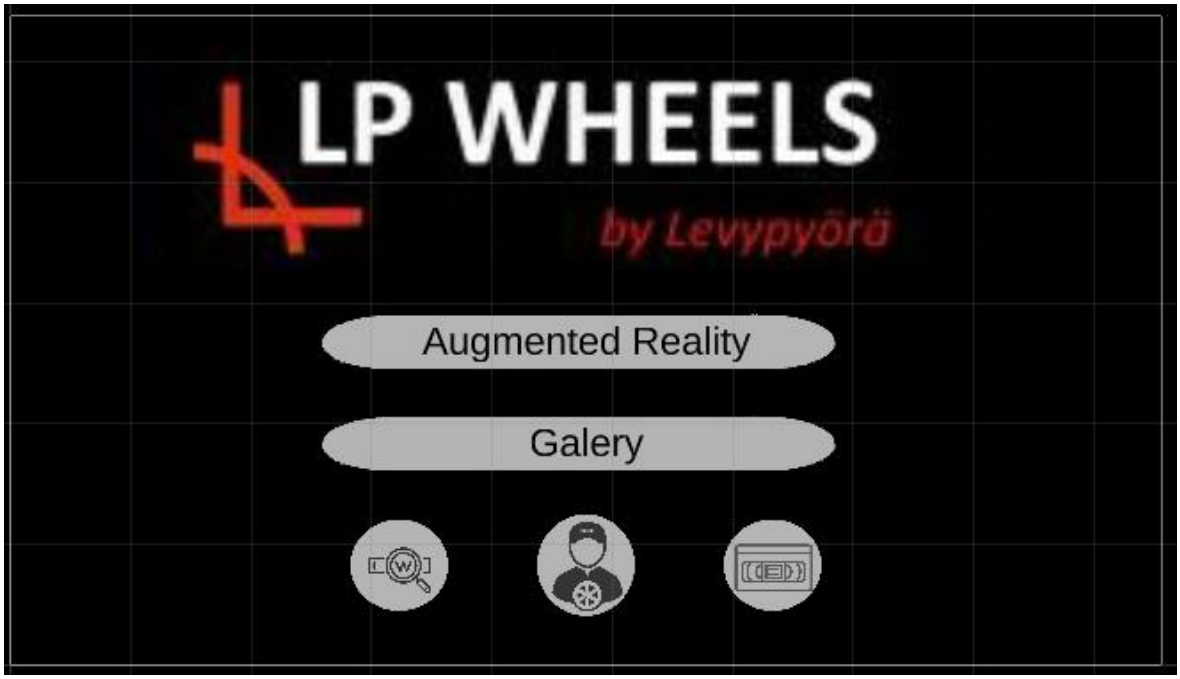


Image 23 Canvas in Unity (2019)

Then it was time to program the navigation between the different views that the application will have. The navigation between views are done in the same way that the navigation between game scenes. It is a good practice to make a general script with only the scene management, which is for using in all the necessary views. Also it's necessary to add the scenes in the "Scenes in Build".

Catalogue View

Another scene that had to be created was the Catalogue View, with a short information about the different types of wheels (Image 24). A canvas must be created with the same option as in the previous stage

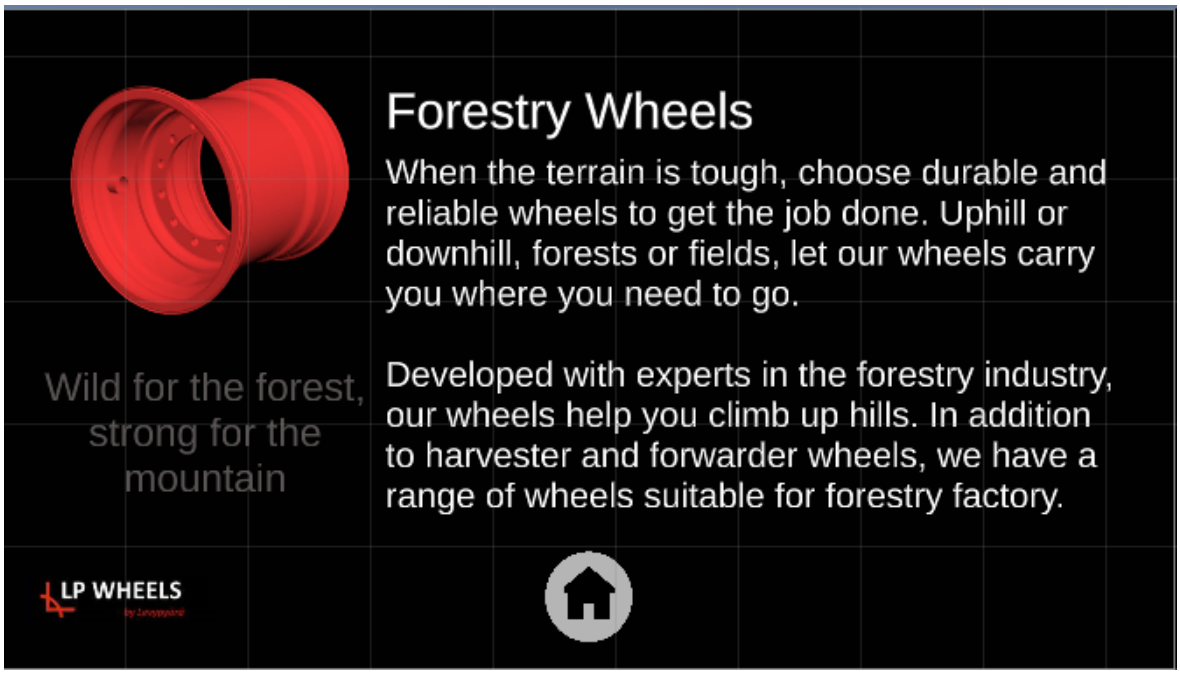


Image 24 Catalogue View

To change between the different wheels, several options where considered for the interaction: back and next buttons, a list of items or a swipe interaction like book pages.

It was agreed that the most comfortable for the user can be the last interaction, implement a swipe. This requirement with other SDK as Flutter, it is simple and it is only a piece of code, but with Unity it is not so simple. For this reason, it was implemented using a free plugin (Image 25). With the swipe add to the view, the Catalogue View would be satisfactorily finished.

```
void Update() {
    if (!firtTime)
    {
        if (panelToMove.transform.localPosition.x > actualPosition && oneTime)
        {
            oneTime = false;
            Movement(false);
        }
        else if (panelToMove.transform.localPosition.x < actualPosition && oneTime)
        {
            oneTime = false;
            Movement(true);
        }
        else if (panelToMove.transform.localPosition.x == tmpPosition)
        {
            oneTime = true;
            actualPosition = panelToMove.transform.localPosition.x;
        }
        else
        {
            tmpPosition = panelToMove.transform.localPosition.x;
        }
    }else if (panelToMove.transform.localPosition.x < actualPosition)
    {
        panelToMove.transform.DOLocalMoveX(-130, 1);
        firtTime = false;
    }

    if (panelToMove.transform.localPosition.x > 1087)
    {
        panelToMove.transform.DOLocalMoveX(1004, 0.5f);
    }
    if (panelToMove.transform.localPosition.x < -1330.038)
    {
        panelToMove.transform.DOLocalMoveX(-1244, 0.5f);
    }
        tmpPosition = panelToMove.transform.localPosition.x;
}
```

Image 25 Code for changing between wheels

Augmented Reality View

The last scene that developed was the AR scene. This view is the most important one, because this is where the company was looking for, a way to rise above its competitors. In this view, a button for changing the colours and another for returning to the Main view have to be implemented. It was agreed in the point 5.5.1 the implementation must be as clean as possible. In this moment of the development, the view only had a panel with the colours and a home button. (Image 26). After the AR implementation it will be necessary to add more buttons and functionalities.

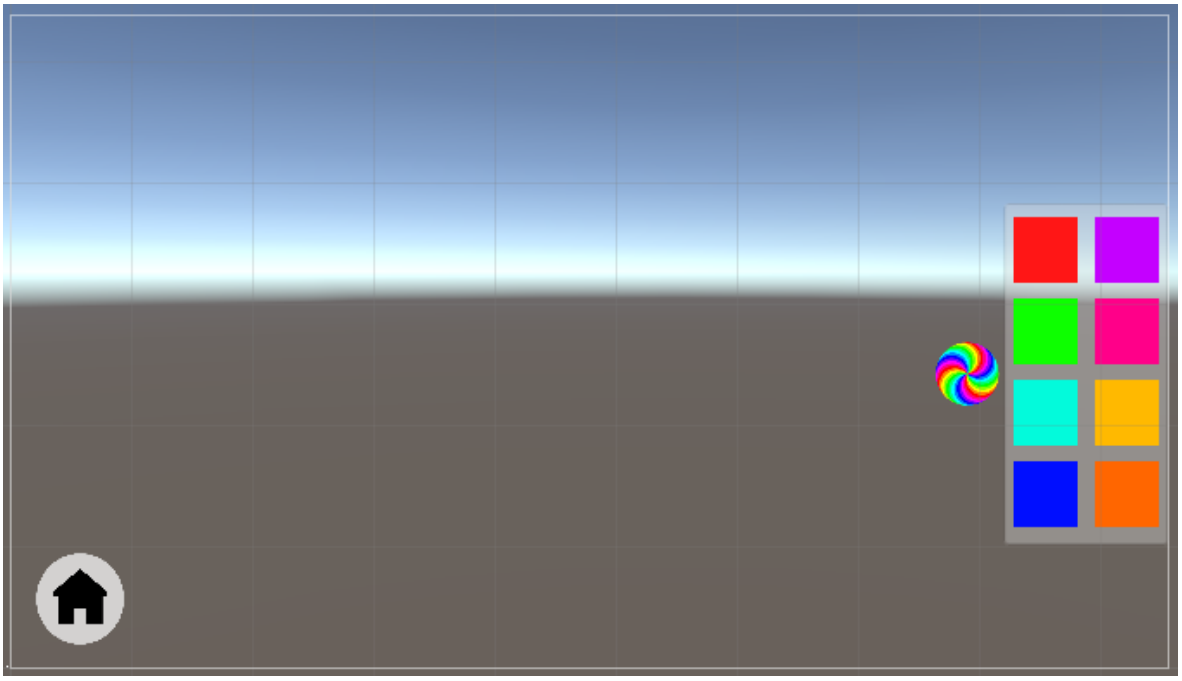


Image 26 Canvas of the AR View (2019)

This view was now finished, and the next step was to add the AR part. The AR view has implemented in the next chapter, for the complexity that it's has.

5.6 Add Augmented Reality Technology to our application

The different available SDKs, with some of their advantages and disadvantages, were exposed in the point 4. Now, one of them must be chosen for starting the development.

The project can also be considered R & D for the company, the technology has to be the most stable and cost effective. Considering that the main idea is releasing an AR application, a single pay license is preferred.

5.6.1 Vuforia as the chosen SDK for Augmented Reality

For the reasons mentioned above, a comparative table was compiled with all the SDKs mentioned in chapter 4. The following items are compared in table 3:

- Useful tools: the possibility to modify Scripts, add new plugin, objects, e.g.
- Pricing: the price of the license compared to the company requirements
- Learning: The difficulty to learn the programming language, the number of tutorials, the good quality of the API, e.g.

- Previous experience: The experience with the SDK, and prior knowledge.
- Integration with Unity
- 2D target recognition

Table 3 Comparison of the different SDK (2019)

SDK	EasyAR	Wikitude	ARkit	ARCore	Maxst	Vuforia
Useful tools	3	5	5	4	3	4
Pricing	5	1	3	5	5	5
Learning	2	3	2	3	2	5
Previous experience	3	3	0	1	2	4
Integration with Unity	3	3	3	3	3	5
2D target recognition	5	5	5	5	5	5
Result	21	20	18	21	20	26

Looking at the table, the best option is Vuforia. Vuforia has a complete integration with Unity. For this reason, it is easy to add the Vuforia SDK to the Unity editor. It is as simple as we can see in image 14.

Two items are used for creating the base AR in Vuforia, the Vuforia Image and the AR camera. It is necessary to import the Vuforia package the first time that it is added to the project, Unity imports it automatically. After that, a Vuforia Image must be added. A pop up window appears for adding the default data base, if you do not have one. (Image 27)

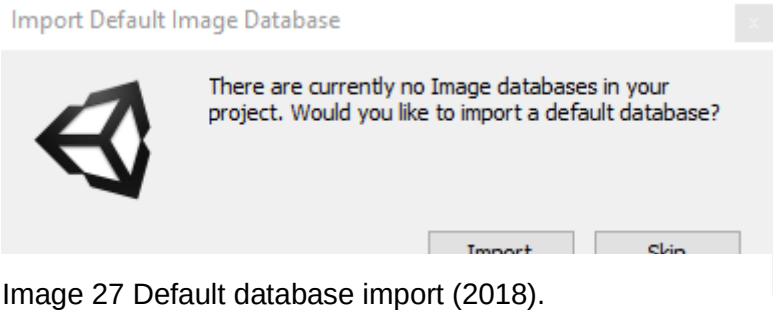


Image 27 Default database import (2018).

After finishing the previous part. Basic AR views without interaction can be created, adding one Vuforia Image and a 3D object as the child of the image. We can see the result in image 28.

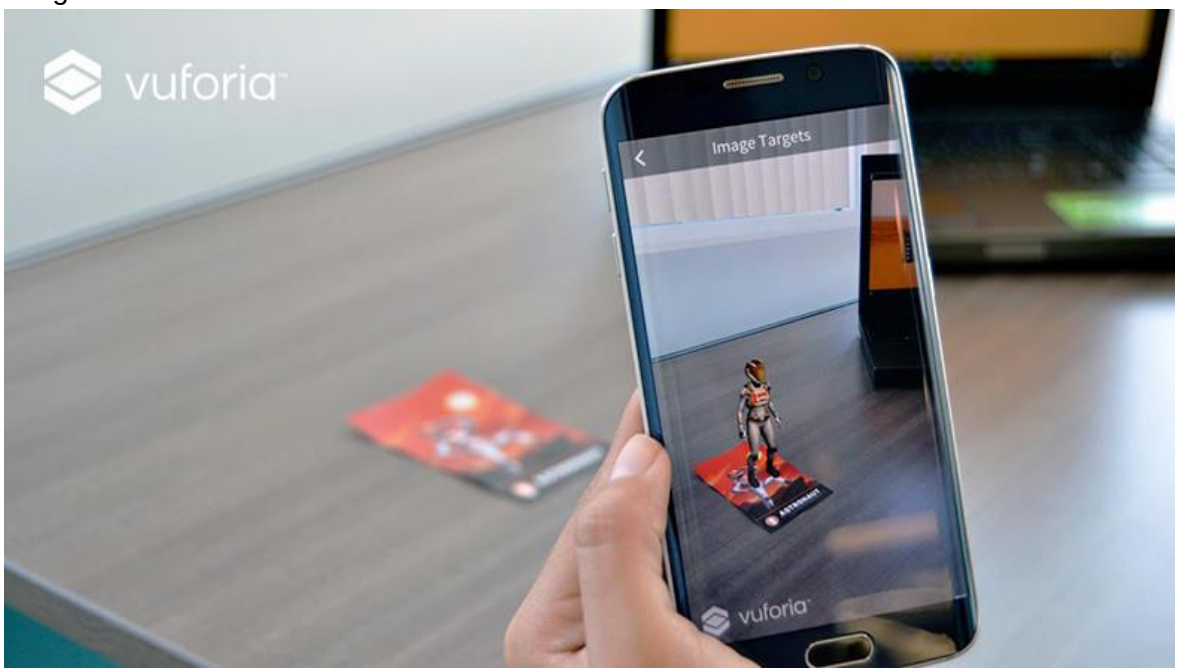


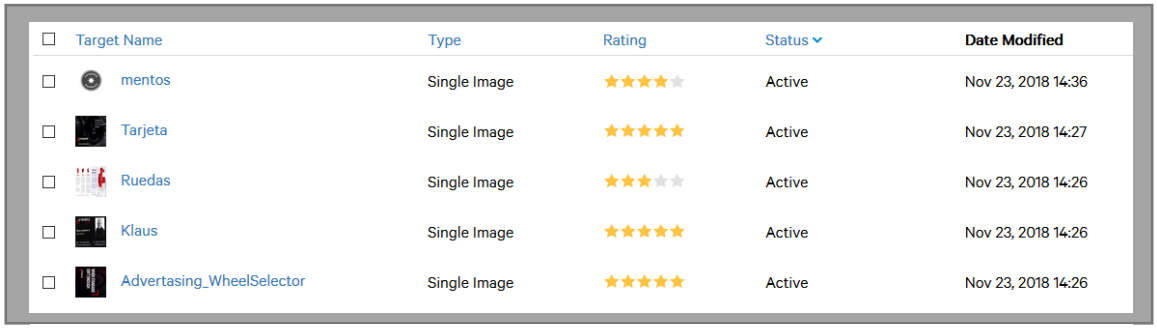
Image 28 Example of Vuforia (unity.com/packages/templates/packs/Vuforia, 2019)

Knowing that Vuforia has a free license for developing in the case, the correct option will be to choose this type of license, before starting the development. The license can be changed if we want to release the app in the future. For now, we will choose the free license in the Vuforia Developer Portal, which allows us to release the application.

Our license allows us to use a custom target, so we will not use the default data base. The targets provided by Levypyörä are the advertising pamphlet, one business card and the logo in their box of mints. Having good target recognition is essential for achieving a stable AR experience. Because we have more targets than necessary, it helps us select the one that works best with the Vuforia Engine.

5.6.2 Adding the Company Targets to the Vuforia Data Base

It is mandatory to add the targets selected to the Vuforia Development portal, if we want to have our own data base. These pictures have to be added in the following image format; jpg or png. There will be a target ranking for each of them. The ranking obtained for our target can be visualized in the next image. (Image 29)



Target Name	Type	Rating	Status	Date Modified
mentos	Single Image	★★★★☆	Active	Nov 23, 2018 14:36
Tarjeta	Single Image	★★★★★	Active	Nov 23, 2018 14:27
Ruedas	Single Image	★★★☆☆	Active	Nov 23, 2018 14:26
Klaus	Single Image	★★★★★	Active	Nov 23, 2018 14:26
Advertasing_WheelSelector	Single Image	★★★★★	Active	Nov 23, 2018 14:26

Image 29 Vuforia Ranking target (November, 2018)

There are different ways to add our data base to the project. The simplest way for do it, is downloading the data base as unitypackage and import it to the project. The data base will be imported to the project in that way. (Image 30)

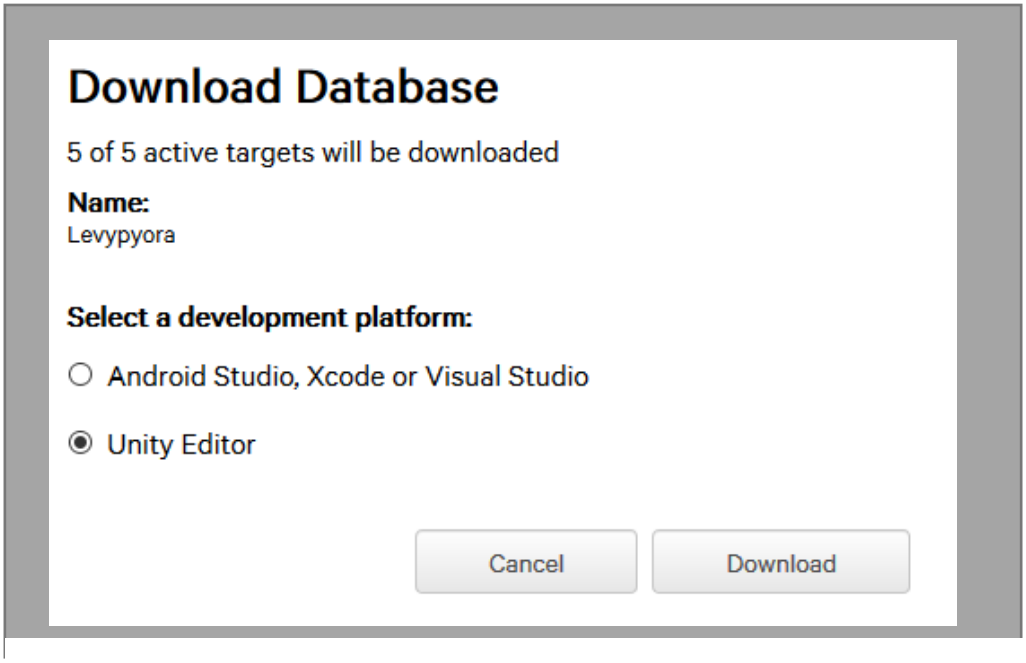


Image 30 Download data base (2018)

5.6.3 3D objects in AR

Our project can start to use AR tools after as described in subchapters 5.6.1 and 5.6.2. An interface in AR to select the type of wheel that the viewer wants to see with more detail was designed (Image 31). After the selection it will be possibly change the colour and move the ring. In order not to confuse the viewer, the selector will be hidden, when the interaction has been done, then the ring selected will be appear, doing an animation.

Image 31 AR Selector (2018)

The base of the AR Scene, together with the interface made in chapter 5.4.3, is now ready to have extra features implemented. The materials corresponding to the colours provided by the company and the development of interactions with the AR are some of the milestones achieved in the following subchapter.

5.7 Interactions In AR

The existing interaction in the AR Selector, the interaction of changing the ring colour and the interaction of rotating the ring, are developed in this chapter. These interactions were selected by the company.

LeanTouch help us to developing the interactions with the 3D objects. Two mandatory components must add for work with it, a collider and one LeanTouch script. The interaction that you can do it with objects, depends of the LeanTouch scripts that it is added to the object. The rotation of an object is controlled by the "Lean Rotate" and this is the script that we will use for the rotation of the ring. However, the selection of a ring can be controlled by the "Lean Selectable". With this two script, now is possible developing the interactions some interactions.

Eight different colours without counting the standard colour, can be selected in the application. For these reasons, 9 different materials must be created. The colours of the materials are chosen as the colours that the company has for painting the ring.

The materials and one script, change the material colour of the ring that it is visualizing, it is possible. For changing the material of the colour, it must be accessed to the component render (Image 32). The colours change as the user click to the colours available in the right panel of the UI (Image 26).

```
wheelComponents[i].GetComponent<Renderer>().material = colors[number];
```

Image 32 How to access component renderer for changing the material (2019)

Now that it can be chosen a type of wheel, visualizing and change the colour as it wish, add a new navigation button is necessary. At the moment, return to AR selector it isn't programming and if it is wanted to choose other wheel, it must return to the Main Scene. One reset button to AR Selector and script, it was developed for fixing. After these changes, the UI of the AR scene is the next one (Image 36).

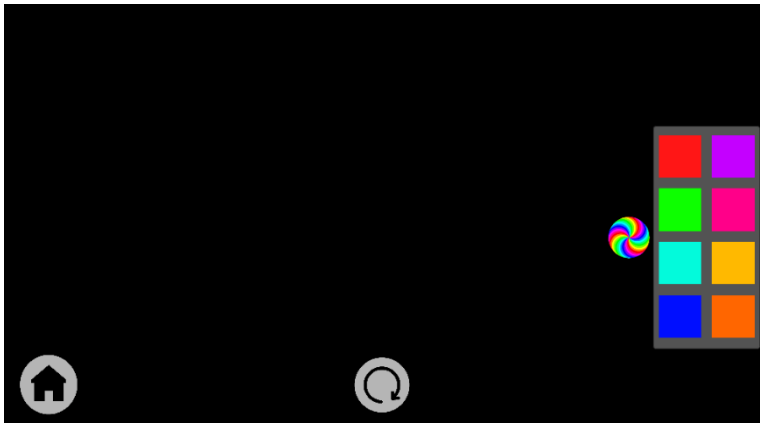


Image 33 UI in the AR Scene (2019)

Other improvements to the UI, after adding the AR interaction is that the right panel shouldn´t be available all the time. Only when a ring is selected, changing the colour must be enabled. When you cannot change the colour, the panel and the reset button fade until disappearing.

The interactions in the AR view are completed, after be completed the three milestones and with it the application as the company wanted for the first demo.

5.8 Load Screens

Although the application is almost done, for finishing it we have to add some improve, like two Load views. One, before of the Main Scene and the other when the Main Scene call to the AR Scene. This load views are necessary because if in the future the company wants to add more objects to the AR part, perhaps the app it takes a while to load the scene, and the user experience could be bad if we not add load scene.

The first view is added for not appear in the Main Scene sharply. It is done one transition between your desktop and the application smoothest. In this load view is only visible the logo of the company doing a fade (Image 34). This view is not a really load view, it does not load in the background. In the future, it is can change to a load view if it necessary.

In the second load view, one animation with the company´s motto is added. For creating a real load view a coroutine with an asynchronous load scene was added. The charge of the new view and the animation are done at the same time.

```
private IEnumerator _LoadScene()
{
    AsyncOperation asyncLoad = SceneManager.LoadSceneAsync("AugmentReality_Scene");

    while (!asyncLoad.isDone)
    {
        yield return null;
    }
}
```

Image 34 Coroutine for asynchronous load scene

5.9 Problems and new requirements

With the actual state of the application, an APK for showing to the company and obtain some feedback was released. In this meeting we agree to add an AR explanation about how the mint's box works, because they have negative feedbacks about it. Open and close the box is not easy understandable for the costumer. The voice of one of their employees and an explicative animation was added to the AR Scene for fixing. A silence button for the voice was added too, giving the next result (Image 35).

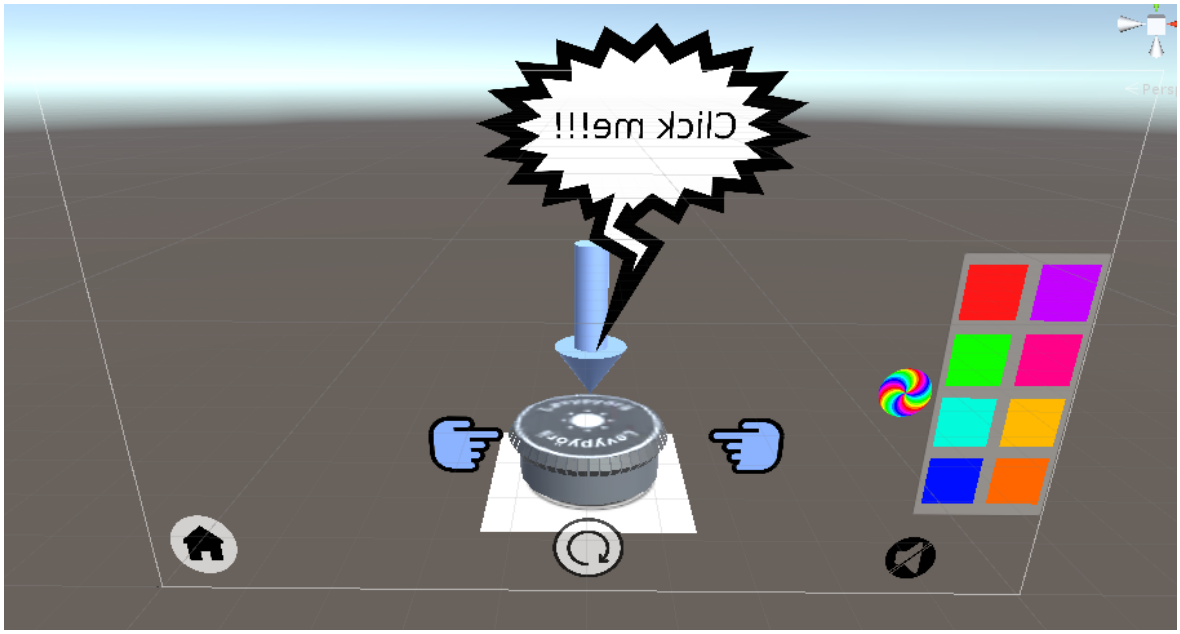


Image 35 AR Scene after add the mint box and the voice (2019)

In the same meeting too, a Vuforia problem with the animated objects, like the wheel ring after select one, was find. For fix this error, it must be overwritten one of the Vuforia scripts that control when a target is found (Image 36). This script must to be added in the target.

```
public class SpecialTrackableEventHandler : DefaultTrackableEventHandler
{
    public GameObject[] objetosClaves = new GameObject[5];

    protected override void OnTrackingFound()
    {
        var rendererComponents = GetComponentsInChildren<Renderer>(true);
        var colliderComponents = GetComponentsInChildren<Collider>(true);
        var canvasComponents = GetComponentsInChildren<Canvas>(true);

        // Enable rendering:
        foreach (var component in rendererComponents)
            component.enabled = true;

        // Enable colliders:
        foreach (var component in colliderComponents)
            component.enabled = true;

        // Enable canvas':
        foreach (var component in canvasComponents)
            component.enabled = true;
        //Active the different KeyObjects that are disavailable when we lose the Target
            for (int i = 0; i < objetosClaves.Length - 1; i++)
            {
                objetosClaves[i].SetActive(true);
            }
    }
```

Image 36 Overwrite script about what happen when a target is found. (2019)

In the Gallery view a problem was found too. Faster movement doing that the system kill the process created by the script, stopping the movement and it is denied the swipe. Creating a new script was the solution (Image 25).

5.10 Ending the application

In the last part of the project, a meeting is had for view the last result. Removing the animation of the box and only do the animation the first time that you click the box was agreed.

The first goal of releasing the app in Google Play was postponed, because the final purpose for the application is unsure. Thus, one APK with free license for personal use was built. Before building the APK, some PlayerSettings as activate Vuforia in the XR Settings and disable Android TV compatibility was modified. Also, the orientation was fixed to

Landscape Left as it was agreed in 5.4. For available the navigation between the views, the new views was added to the Scenes in Build (Image 37).

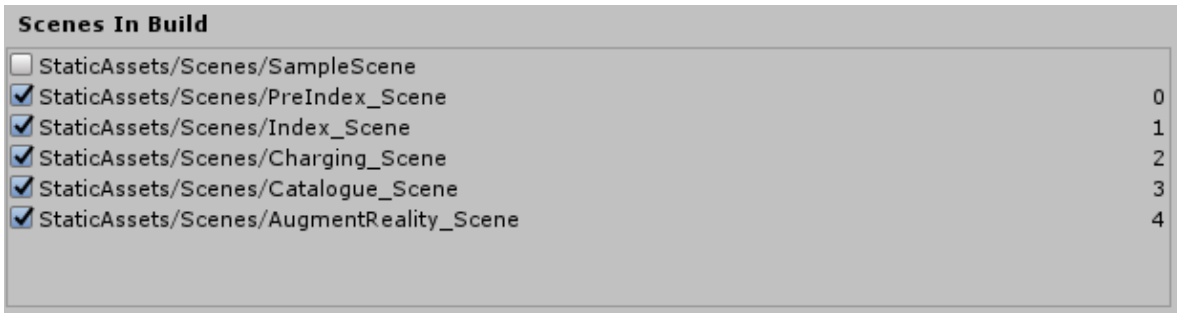


Image 37 Scenes in Build. (2019)

This APK with all the APKs before, the pictures and the resources (without the source code) was uploaded in a file storage shared by both. With this, the development process was ended until the company want to add new features.

6 CONCLUSIONS

Releasing an application in short time is complicated, because you have to hurry while doing it the best you can. The application was not released in a platform like Google Play or Apple Store that is disappointing because it was one of the first goal that the company proposed.

During the first steps of development, Levypyörä was closely involved in the project, but in the middle of the project the company started to have a lot of work and the project lose priority. Finally, it was more like a research project. This coupled with not having a clear idea about what they wanted to show exactly, caused the project to remain quite superficial. However, it is common in the IT world being that only 20% of the software projects end as expected (Anex M, A. 2019).

However, it comforts me to know all that I have learned about SDKs for development in Android, but above all to have had to use six different SDKs to know which was the most suitable for this project.

In the future the project could be resumed, add more features and do it more interactive for the user.

LIST OF REFERENCES

ARCore. 2018a. Overview [Accessed 12 December 2018]. Available at: https://developers.google.com/ar/discover/

ARCore. 2018b Platforms [Accessed 12 December 2018]. Available at: https://developers.google.com/ar/develop/

ARCore. 2018c. Community [Accessed 12 December 2018]. Available at: https://developers.google.com/ar/community/

ARkit. 2018a. ARkit [Accessed 12 December 2018]. Available at: https://developer.apple.com/arkit/

ARkit. 2018b. Develop [Accessed 12 December 2018]. Available at: https://developer.apple.com/develop/

Anex M. A. 2019 ¿Por qué fracasan los proyectos en las organizaciones? [Accessed 9 May 2019] Available at: http://www.emb.cl/gerencia/articulo.mvc?xid=1275

EasyAR. 2018a. Pricing [Accessed 5 December 2018]. Available at: https://www.easyar.com/view/sdk.html

EasyAR. 2018b. Overview [Accessed 5 December 2018]. Available at: https://www.easyar.com/view/download.html

Flutter. 2019. Flutter [Accessed 5 January 2019]. Available at: https://flutter.dev/docs

Kim, M. 2017 Leading Korean AR startup Maxst raised USD 3 M to cover more industries. Alltechasia. [Access 13 December 2018]. Available at: https://alltechasia.com/leading-korean-ar-startup-maxst-raised-usd-3-m-cover-industries/

Maxst. 2018a. AR SDK. [Accessed 13 December 2018]. Available at: http://maxst.com/#/en/arsdk

Maxst. 2018b. Case Studies [Accessed 13 December 2018]. Available at: http://maxst.com/#/en/casestudies

Maxst. 2018c. Pricing [Accessed 13 December 2018]. Available at: https://developer.maxst.com/Pricing

Maxst. 2018d: Features [Accessed 13 December 2018]. Available at: https://developer.maxst.com/Features

Unity. 2019. Unity [Accessed 5 January 2019]. Available at: https://unity3d.com/es/unity?_ga=2.56577835.1028620403.1557777511-1732380972.1542617545

Vuforia. 2018a. Overview. [Accessed 15 December 2018]. Available at: https://library.vuforia.com/content/vuforia-library/en/features/overview.html

Vuforia. 2018b. Pricing. [Accessed 15 December 2018]. Available at: https://developer.vuforia.com/vui/pricing

Vuforia. 2018c. Samples. [Accessed 15 December 2018]. Available at: https://assetstore.unity.com/packages/templates/packs/vuforia-core-samples-99026

Wikitude. 2018a. Wikitude [Accessed 10 December 2018]. Available at: https://www.wikitude.com/products/wikitude-sdk/

Wikitude. 2018b. Platforms [Accessed 10 December 2018]. Available at: https://www.wikitude.com/download/

Wikitude. 2018c: Pricing [Accessed 10 December 2018]. Available at: https://www.wikitude.com/store/

Xamarin. 2019. Xamarin [Accessed 20 January 2019] Available at:

https://visualstudio.microsoft.com/xamarin/?rr=https%3A%2F%2Fwww.google.com%2F

IMAGES LIST OF REFERENCES

Android NDK. 2019. Android NDK. [Accessed 8 February 2019] Available at: https://www.androidvenezuela.com/desarrollo/creacion-aplicaciones-android-basadas-ndk-arquitectura-intel

Arstechnica. 2019. Flutter Image [Accessed 8 February 2019] Available at: https://arstechnica.com/gadgets/2018/12/google-bridges-android-and-ios-development-with-flutter-1-0/

EasyAR. 2019. EasyAR prefabs in the Unity Editor [Accessed 5 December 2018] Available at: https://www.easyar.com/

Orfer. 2019. Levypyörä Logo. [Accessed 15 March 2019]. Available at: https://www.orfer.fi/LinkClick.aspx?fileticket=QR2PEHYJ8VY%3D&tabid=13059&language=en-US

Levypyörä. 2019. Colours of the brand. [Accessed 29 November 2018]. Available at: https://levypyora.fi/

Levypyörä. 2019. Example of typography and how to divide content by using squares. [Accessed 29 November 2018]. Available at: https://levypyora.fi/

Muropaketti. 2019. Pokémon Go [Accessed 21 January 2019] Available at: https://muropaketti.com/pelit/kaukana-kuolleesta-pokemon-go-tuotti-viime-vuonna-yli-800-miljoonaa-dollaria/

Maxst. 2019. AR Guides using Maxst [Accessed 13 December 2018]. Available at: http://maxst.com/#/en/casestudies

Maxst. 2019. Augmented Reality Tutorial [Accessed 21 January 2019] Available at: http://maxst.com/#/en/casestudies

Vuforia. 2019. Vuforia Forum [Accessed 15 December 2018]. Available at: https://developer.vuforia.com/forum

Wikitude. 2019. All the platforms available for development in Wikitude [Accessed 10 December 2018]. Available at: https://www.wikitude.com/store/

Wikitude. 2019. Example of the discounts available in some special occasions [Accessed 10 December 2018]. Available at: https://www.wikitude.com/store/

Wired. 2019. IKEA Augmented Reality [Accessed 13 February 2019] Available at: https://www.wired.com/story/ikea-place-ar-kit-augmented-reality/