Pontificia Universidad Javeriana de Bogota
&
École nationale supérieure d'Électrotechnique, d'Électronique, d'Informatique, d'Hydraulique et des Télécommunications

Computer Science Department - Internship

AROUND: AUGMENTED REALITY MULTI-PLATFORM TOOL IN THE VALUE CHAIN OF THALES ALenia SPACE.

Intern: Jorge Enrique GUTIERREZ ZULUAGA

28/08/2017
ARound: Augmented Reality multi-platform tool in the value chain of Thales Alenia Space.

Rédigé par
Jorge Enrique Gutierrez Zuluaga

Responsabilité
+ signature manuelle si pas de système de workflow

Vérifié par
Amandine PAILLOUX

Approuvé par
Amandine PAILLOUX

La preuve des validations est conservée par le système de gestion documentaire.
## DESCRIPTION DES EVOLUTIONS

<table>
<thead>
<tr>
<th>ISSUE</th>
<th>DATE</th>
<th>§ DESCRIPTION DES EVOLUTIONS</th>
<th>REDACTEUR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

First and foremost, I would like to express my very great appreciation to the IRD service for contributing to one of the best working experiences I ever passed, thank Olivier P. for being an excellent manager and giving me the opportunity to work on an interesting project such as this, thank Jonathan S. for being a good team leader and showing his trust in my profile and my background since the beginning, and I extend my thanks to the rest of the team for welcoming me so warmly, it was always a pleasure working and sharing time with such lovely and engaging people.

Finally, I wish to thank my tutor Amandine P. who has advised and supported me not only in the working context but also in other professional circumstances. She always shared her knowledge and experience whenever I needed and invited me to seek further alternatives in order to achieve my goals. I have to admit the fact that I am not fully bilingual in French, is always a barrier, and she constantly made the effort to break it with her kindness and her charisma.

Merci à tous.
SUMMARY

The Augmented Reality is one of the emergent technologies in the last 10 years attached to human senses; all kind of devices were created to exploit this advance, and nowadays we count with a new generation of Augmented Reality tools that optimize and improve that experience. It is clear this technology enhances the human’s perception of the real world by displaying features that cannot be directly detected by their own senses, and this could be used for education, training, and assistance purposes.

For Thales Alenia Space (TAS), the innovation is one of company strategies, is for that reason they invest some resource in Augmented Reality research. Following this, the Innovation, Research and Development service (IRD) of Thales Alenia Space in Toulouse has the idea of developing a multiplatform application which allows connecting any device each other, comprising the new AR device generation, to work simultaneously on a downloadable 3D model and displaying it in a real context by using Augmented Reality. So, this project idea was assigned as an internship project and it will be reported in this document.

First, the document introduces the enterprise context, its main activities and how it is organized and structured. Then, it explains the project and the internship framework, their objective and the procedure executed to achieve the project goals following some foundation of agile software development.

Second, the technologies and application’s components used in this project are subsequently detailed in order to clarify the principle of some tools performed and why they were selected for the application development. The Augmented Reality used is based on the visual feature tracking and it is performed by Vuforia plugin, the communication and synchronization between each device are handled by Unity Networking API and Unity engine allows build cross-platform applications including Hololens device.

Finally, the results and the conclusion for the follow-up of this project are shown. Arguably, the results were positive; the goal of developing an application which meets the requirements was achieved and it is remarkable the potential of this application. For now, the application could be used in conference and meetings where talk about a 3D CAD model is expected, furthermore, the application could be potentially exploited during conception process if IRD continues to work on its development.

Key words: Augmented Reality, Cross-platform.
5.3. **ASSETBUNDLES** ........................................................................................................... 31

5.4. **HOLOTOOLKIT FOR UNITY** ....................................................................................... 32
    5.4.1. Features Used ............................................................................................................ 32
    5.4.2. Input ......................................................................................................................... 32
    5.4.3. UI ................................................................................................................................ 33
    5.4.4. Utilities and Build .................................................................................................... 33

5.5. **VUFORIA SDK FOR UNITY** ..................................................................................... 33
    5.5.1. Image Target .............................................................................................................. 33
    5.5.2. Extended Tracking .................................................................................................... 35
    5.5.3. Extern Components ................................................................................................. 36

6. **ARCHITECTURE** ........................................................................................................... 38
    6.1. **PACKAGE ARCHITECTURE** .................................................................................. 38
    6.2. **SEQUENCE DIAGRAM** .......................................................................................... 39
        6.2.1. Create a Room ....................................................................................................... 39
        6.2.2. Join a Room .......................................................................................................... 40
        6.2.3. Download a 3D CAD Model ............................................................................... 40
        6.2.4. Paint Model Component ...................................................................................... 42

7. **RESULTS** .................................................................................................................... 43

8. **CONCLUSIONS** .......................................................................................................... 47

9. **BIBLIOGRAPHY** .......................................................................................................... 48

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Titre</th>
</tr>
</thead>
<tbody>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>AR</td>
<td>Augmented Reality</td>
</tr>
<tr>
<td>BE</td>
<td>Bureau d’études</td>
</tr>
<tr>
<td>CAD</td>
<td>Computer-Aided Design</td>
</tr>
<tr>
<td>DSI</td>
<td>Direction de system d’informatique</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>IRD</td>
<td>Innovation, Research and Development</td>
</tr>
<tr>
<td>HLAPI</td>
<td>High Level Layer Application Programming Interface</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>IMU</td>
<td>Inertial Measurement Unit</td>
</tr>
<tr>
<td>MRTK</td>
<td>Mixed Reality Tool Kit</td>
</tr>
<tr>
<td>RPC</td>
<td>Remote Procedure Call</td>
</tr>
<tr>
<td>SDK</td>
<td>Software Development Kit</td>
</tr>
<tr>
<td>TAS</td>
<td>Thales Alenia Space</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>UWP</td>
<td>Universal Windows Platform</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>VR</td>
<td>Virtual Reality</td>
</tr>
</tbody>
</table>

Figure 1 Thales in Europe ........................................................................................................ 11
Figure 2 - Thales Organisation in France .................................................................................. 13
Figure 3 - Gartner diagram of emerging Technologies 2017 ........................................................ 15
Figure 4 - User Cases. .................................................................................................................... 17
Figure 5 - Methodology framework ............................................................................................... 18
Figure 6 - Phases and stages of the project .................................................................................. 20
Figure 7 - Hololens Device ........................................................................................................... 24
Figure 8 - Communication Calls Scheme ....................................................................................... 26
Figure 9 - Authority Concept ......................................................................................................... 28
Figure 10 - Network Architecture .................................................................................................. 30
Figure 11 - Good Marker Requirements .......................................................................................... 34
Figure 12 - Marker used ................................................................................................................... 34
Figure 13 - Image natural features .................................................................................................. 35
Figure 14 - Environment .................................................................................................................. 36
Figure 15 - Extendable tracking feature ........................................................................................ 36
Figure 16 - Package structure ......................................................................................................... 38
Figure 17 - Create a Room sequence diagram ............................................................................... 39
Figure 18 - Join a room sequence diagram .................................................................................... 40
Figure 19 - Download 3D model sequence diagram ....................................................................... 41
Figure 20 - Paint model component sequence diagram ................................................................ 42
Figure 21 - UI Around app v1.0 Start .............................................................................................. 43
Figure 22 - UI Around app v1.0 model modified ............................................................................. 44
Figure 23 - UI Around app v2.0 Start .............................................................................................. 45
Figure 24 - UI Around app v2.0 model modified ............................................................................. 45
Figure 25 - Screenshot functionality .............................................................................................. 46
1. INTRODUCTION

Engineering formation in ENSEEIHT School has a duration of three years and in the last year, every student has to complete an internship in a professional context in order to get the required experience and develop their technical skills for a certified job. Against that background, I was recruited by Thales Alenia Space Toulouse to do my internship from April 3rd to September 29th 2017, in the context of supporting the IRD (Innovation, Research, and development) department and its Augmented Reality project by developing a prototype app.

During the internship, I had the possibility to work with new technologies applied to this sector and enforced my technical knowledge in a specific and professional area. I had also the opportunity of learning about the last software methodologies and the main domains of the Spatial Industry. I was counting on the advisory of Simone G. my academic tutor and Amandine P. my tutor from the professional context to achieve the project goals.

The principal objective of my project was present to satellite conception department a cross-platform application that allows downloading and display a 3D CAD model in order work on it in parallel collaboration among operators. My constant curiosity and my persistence were indispensable to achieve this goal. It was mandatory that I researched and tested any potential resource, tool, or application that pointed to a means of resolving the specific needs demanded.
2. **INTERNSHIP CONTEXT**

2.1. **Thales**

Thales Group is a French multinational specialized in aeronautic, space, terrain transport, security, and defense domain and is part of global leaders in these sectors.

Hereditary of dedicated branches to the military activities of Alcatel, Dassault Electronique and Thomson-CSF in 1998, it is benefitting overall turnover of 15 billion euros. World -class technology, the combined expertise of 64,000 employees and operations in 56 countries have made Thales a key player in keeping the public safe and secure, guarding vital infrastructure and protecting the national security interests of countries around the globe.

With the purpose of keeping a big influence on the market, Thales focuses a part of activities in innovation, investing 20% of their activities in research and development, the key areas are the mastery of the complex system, materials (or the technology of captors), software, decision algorithms.

Thales Group comprises six entities:

- Thales Alenia Space.
- Thales Avionics.
- Thales Defence Missions Systems
- Thales Ground Transportation Systems
- Thales Land and Air Systems
- Thales Secure Communication and Information Systems.

2.2. **Thales Alenia Space**

2.2.1. **Generalities**

Thales Alenia Space (TAS) is an entity of Thales Group, is a joint venture between Thales (67%) and Leonardo (33%), it also teams up with Telespazio to the parent companies "Space Alliance", which offer a complete range of service and solutions. TAS posted consolidated revenues of about 2.4 billion euros in 2016 and has 7.980 employees distributed on 15 industrial sites in nine countries, among the most influential, French, Italy, Spain, Belgic, Germany, and the United States. Their activity sectors are space industry, satellites manufacturing, equipment, and services for special programs of telecommunication, Earth observation, navigation, and science.

TAS has more of 40 years of experience in the conception, integration, exploitation, and delivery of innovative special systems. It proposes high tech products which answer to commercial, institutional, scientific, defense and security needs. The satellites and payloads designed by TAS are internationally recognized by offering an excellent service.
2.2.2. Activities

TAS is also at the forefront of satellites for both, civil and military sector. The enterprise provides services to special agencies from ESA, France, and Italy and defense ministries. It has also exportation activities on the Korean, Brazilian, Turkish, and Arab Emirates market and is involved in some international programs such as Koreasat-6, GALILEO, Globalstar. Their principal axes are:

- Telecommunication: TAS has a roll on the market of telecommunication satellites, it is one of the global leaders in conception area and this market represents 50% of the activities society.

- Earth observation: TAS is also Global leader of geostationary weather satellites (such as Meteosat satellite). In addition, it has developed a big expertise in climatology missions like Calipso for CNES/NASA. Thus, TAS is recognized in the field of optical payload and high resolution radar for working on meteorology, oceanography, climatology, cartography, intelligence gathering, etc.

- Science and Exploration: TAS is part of European scientific programs of Solar System exploration. Society is engaged in all ESA's exploration programs. This company contributes to ExoMars mission what studies the Mars’s environment, atmosphere and ground.
• Navigation: TAS is a European leader of navigation systems and an EGNOS (European Geostationary Navigation Overlay System) developer. This system has the mission of enhance the positioning messages delivered by GPS system. It is also an important member of Galileo’s program, with the Ground Segmentation mission.

• Security and Defense system: In French, TAS is the first supplier that provides complex spatial instruments for defense and security systems. It is also responsible of integrations. TAS offers a full line of solutions to meet any defense and security need.

2.2.3. Thales Alenia Space in France

Thales Alenia Space in France is located in Cannes and Toulouse. Its principal activities are: Complete integration of Telecommunication and Ground Navigation systems, manufacturing and integration of payloads, control centers and ground missions, production and service of ground segmentation, and space electronic equipment supplier.

In Toulouse, TAS has a big role in telecommunication payload sector and is a global leader in hyper-frequency equipment manufacture. It has exclusive capacities for complex hybrid modules design, is master in cabling, bounding and polymerization, and it is specialized in research, development, production and test of antennas.

In France TAS is divided into different entities:

• Business Line Telecommunication

• Business Line Observation and Navigation

• Meta Competence Center, divided into three departements: CCSL (Centre de Compétences Solutions Logicielles), CCEL (Centre de Compétences Electroniques) and CCPIF (Centre de Compétences Plateforme Intégration France)
The Business Line defines the client’s needs and specifications. They are intermediaries between clients and Competences Center that develop and manufacture satellite according with design specifications.

In Meta Competences Center context, CCEL produce all electronic materials that are assembled in satellites.

CCPIF works on the satellite in itself; it assembles the satellite equipment and tests it. This entity is divided among Cannes and Toulouse: In Cannes they focus on satellite platform, the common part of every satellite, handling movement, communication and its principal needs (solar panels, motors, ...
etc.). In Toulouse, the payload is assembled according with the satellite's mission (observation, telecommunication, etc.)

CCSL focus their software development activities for spatial applications, ground segmentation including test benches and satellite simulations or user applications related to own satellite's mission. It is worth noting that spatial system has a ground segment composed of an antenna system, a control center, and its own components for the specific mission.

The principal axes of CCSL are:

- Future Ground Segments, in charge of digital transformation for ground segment using Big Data technologies in order to analyze large amount of data provided by a satellite and a high level of performance for increase the speed up in calculations. CCSL want also protect its satellite data helped by cyber security.
- Satellite Factory of the Future: it uses the new technologies such as Machine Learning, Big Data, AR and VR to improve the manufacturing sectors.
- Software Defined Satellite: in charge of software development defined for satellites and the dynamic control of each mission.

2.2.4. IRD

The intern has made their internship in the Innovation Research and Development service (IRD) an entity of the CCSL department (Centre de Compétences Solutions Logicielles).

IRD is led by Olivier P and created in September 2016; it is a service of 15 employees distributed between TAS Toulouse and TAS Cannes. It has the distinction of be a transversal service inside CCSL department with the purpose of provides solutions in different sectors for internal and external TAS clients. The IRD members are on the lookout for the innovate technologies which will research and develop in order to propose prototypes launching new techniques and answering to client's needs for spatial applications. Its own action plan is focused around six principal axes: Big Data, Artificial Intelligence, Machine Learning, Virtual and Augmented Reality, Ergonomic and Embedded Systems.
3. **PROJECT CONTEXT**

One of the innovation strategies from IRD department is ‘Technology Push’ that consists of developing a new product derived from research and development in a new technology field. Technology Push usually does not involve market research. It tends to start with a company developing an innovative technology and applying it to a product. The company then markets the product.

In the project case, the pushed technology is Augmented Reality (AR) and is sought to seduce the potential clients to trigger the budget in order to industrialize AR application in the business. The procedure to be followed is exploring a new way to do things based on AR technology, then looking for the possible niche in different Thales Alenia Space’s sectors and finally proposing this new solution to investors and stakeholders.

![Gartner Hype Cycle for Emerging Technologies, 2017](image)

**Figure 3 - Gartner diagram of emerging Technologies 2017**
The AR is not a recent technology, it is a while AR was discovered and used, nonetheless, according to some Gartner’s studies (the world’s leading research and advisory company), this technology has already through an adaptation phase (a period around 10 years) to be mainstream approved and became a potential technology in market. And it is clearly evident that a new generation of AR/VR device had been approved, according to IDC AR/VR tracker (a research program focusing on the emerging Augmented Reality (AR) and Virtual Reality (VR) headset market), Worldwide AR and VR headset shipments are expected to see a compound annual growth rate (CAGR) of 108.3% over 2015-2020 forecast period, reaching 76.0 million units by 2020, meaning humanity is able to potentially profit from that technology.

3.1. Objectives

The project goal is to develop a prototype, in order to demonstrate the potential field of AR in spatial industry, and boost the likely investors that can release the required budget to explore and industrialize AR applications that are able to improve process in some sector of Thales Alenia Space.

3.2. User cases

The IRD team has detected two main needs of the Thales Alenia Space value chain:

The first is about how the agents who have direct contact with customers can gain the power of persuasion and make a good impression, adding value to the strategic name and the capabilities of this enterprise. The team has noticed that using an AR application in the meeting room with clients can give a plus in this area and improve the image of TAS

The second need is related to the production chain and more specifically with the design department during of the satellite design and conception phases. Throughout these phases it is important to make an exchange of information. The flow is generally 3D CAD models which are able to be displayed on screen in order to know the dimension of the components, materials, the test results on the pieces and potential changes of the model.

For the satellite designers, it is important to know the real scale of the satellite components and how it can fit with other components in a real context, e.g. in the workshop room. In addition, the conception phase is a collaborative work made by a team of many people that normally meet in the same room and talk about the model on the same screen; this fact limits the perception of the model and confines it to a single view of the object. Another difficulty is to share 3D models with collaborators all around the world due to a lack of products with this feature

Following the analysis of the needs of some satellite conception operators, the IRD team has established a hypothetical scenario by supplying the tools based on AR that will improve the information flow. This scenario take in consideration the fact of the operators could be in different rooms including their workstation and work all together simultaneously in the same model visualizing each modification from the other members, this model will be a 3D CAD object loaded by a moderator of the session.
3.3. Stakeholders

3.3.1. Innovation Cluster

There is a culture of innovation at Thales Alenia Space, and the structure that is in charge to develop and get innovative solutions to the market quickly is the Innovation Cluster. This structure is capable of rapidly assimilating and financing new ideas, and represents a breakout from the conventional approach.

Innovation Cluster supports and funds innovative, value-generating projects submitted by employees in the industrial, technical, commercial and organizational domains. The Innovation Cluster is open to all employees including IRD department, but the proposal has to comply with quality aspect such as positive trends and corporate image, and quantitative terms like the benefit for the company.

3.3.2. BE

The BE (Bureau d’Études) is the design department, its task is supporting the design and conception of satellites according to customer
3.3.3. DSI

The DSI (Direction de System d’Information), is the department of security and networking, responsible of monitoring the information flow and provides

3.4. Approach

The IRD has proposed to develop a cooperative (remote) AR application that allows users to see and manipulate a 3D CAD model in real context from different devices such as mobiles, tablets, AR glasses and computers in real time. This application enables the user to create a room (a session) for the other users, or connect to one already created, by the internet, and download a 3D CAD model from a local server in the enterprise network (intranet). At the same time, each user connected to the specific room will be able to see the loaded model placed in a common position in the meeting room thanks to a 2D marker defined by the application.

3.5. Methodology

During the project, the method set to develop the application was the agile software development, which stands out for its planning, evolutionary development, early delivery and continuous improvement, which encourages rapid and flexible responses to changes.

![Figure 5 - Methodology framework](image-url)
This employed method is based upon the popular agile method known as Scrum that is developed in unit cycles of development, called sprints(iteration). The sprint has a duration between one week and one month, for this project the sprint advance is counted as a week, each sprint starts with ‘sprint planning’, identifying the work for the sprint and making an estimated forecast for sprint goals. The sprint ends with a sprint review and a sprint retrospective that allow identifying the progress and the improvement for the next spring.

For this case, the project is developed by the intern who takes the planning and the goal of the current sprint, which was proposed by their tutor according to the Product Backlog and the objectives for that week, and works on it for one week in order to add new functionalities and/or improvements to the Potentially Releasable app. The next week a new iteration is executed, adding the sprint review and retrospective of the last week. To keep the stakeholders considerations in the application, each two months, at least, the developer and the validator, the intern and his tutor, respectively, have a meeting with some likely stakeholders to show the application advancement and receive feedback on it in such a way that a Backlog Refinement could be applied.

Moreover, during this internship other Agile Software Techniques such as Stand-up Meeting and Planning Poker were used with the objective of putting some values and principles of Agile Software Method into practice.

3.6. Stages of the project

To accomplish the project objectives, there are three principal tasks to be performed: implement AR module, implement Communication between different kind of devices, and to use the Hololens interface. Personally, it was the first time I did this kind of work and I had no experience with these specific tasks, this means my progress has to be submitted to a learning stage in addition to the development stage, and the testing/validation stage (see Figure 6)
In the learning stage, I had to do research on the different existing methods, techniques, and tools, test them and choose one to be implemented in the development of the application based on simplicity, effectiveness, and performance.

For AR features, the listed tools were “ARToolKit” and "Vuforia", the second was chosen because it was easier to deploy in Unity platform and take advantage of its utilities, likewise, to find the correct tool or plugin to use in communication component, an extensive search was made, analyzing tools that offer low-level and high-level solutions. The HoloToolkit proposes a Sharing plugin which only works on Universal Windows Platforms (UWP), and its socket protocol communication has an unknown encapsulation structure, whereas Unity Network API proposes a communication multi-platform able to manipulate the low and high level layer, simplifying the object management.

It is worth noting that mastering HoloToolkit to create Hololens apps counts as a learning process.

In the Development phase, once the tools have been selected, I deploy the tool utilities in order to address the requirements, stakeholders, and the project goal.

The ‘one target image’ feature of Vuforia was used to fulfill the need of AR tracking; this feature will be explained in the Application components section with more details (see “5.5 Vuforia for Unity” section).
To connect different remote devices in the same session and have a high-performance serialization of messages that handle synchronization, the high level scripting API of Unity Networking features was exploited. Moreover, the requirement of import any 3D CAD model in the application to work with, could be solved by AssetBundle utility proposed by Unity, (see 5.3AssetBundle section for more information).

The future developments will include a testing and validation phase. The project members contacted an ergonomic specialist to validate HMI and a security expert to validate the networking architecture.
4. TECHNOLOGIES

4.1. Augmented Reality

To start with, it is important to mention the principle of Augmented Reality. According to Ronald T. Azuma, the pioneer in this field, he defines AR as a variation of Virtual Environment or Virtual Reality. The difference is that the AR does not allow the user to completely immerse in a synthetic environment, enabling the users to see the real world around him. AR overlays digital elements in some parts of reality instead of replacing it completely, with the purpose of allowing users to interact with the real world and enhances the user’s perception to sense features that cannot be directly detected by their own senses.

Azuma identifies three main components in AR:
- Combination of Reality and Virtuality
- Interaction in real time.
- Registration in 3D.

It is also worth noting that 2D objects are not included in these features, in other words, 2D films are overlapped on the displays.

AR is used in many fields such as medicine, culture, education, advertising, commerce, maintenance, repair, and entertainment, all using different kinds of technology, that we are going to categorize in three classes; Display, Trackers, and Interfaces.

The Display is the mean by which the user will see the reality. Will the application display it on a video screen or a transparent surface? or above of an environmental surface as a projection? and also where will it be positioned? Will it be positioned on the eye vision like glasses? or in the hand as a handheld device? or will it even be part of the environment?

Tracker technology is essential to know the user’s position and the environment’s orientation. There are many possibilities of choices to implement tracking. They are comprehended in the environment (Indoor - Outdoor), the distance and the level of precision. Commonly the technologies used are GPS, Optical Tracking, Mechanical, Ultrasonic, Magnetic and IMU (Inertial Measurement Unit). Many of the pieces of equipment that allow the application of those techniques are available on smartphones and tablet devices, this means that this technology is accessible to the majority of the population.

The Interface is needed to help users interact with virtual objects, thus the user can interoperate with the virtual world. There are three axes in the Interface’s technology; Tangible UI, which the users of this device can touch, Haptic UI, physical feedback, and Visual UI, which uses visual tracking for gesture recognition and not necessarily physical objects.

It is worth discussing this Tracking subject, its objective is know how to position a virtual object in the real world at the real time. For this, it is primordial to know the location in the real world by detecting the coordinates in the real environment and projecting it through the camera’s perspective with its respective optical distortion. The location is a reference given by the real world as the
placeholder, it is usually easy to detect and correct with help that it is not linked to cameras, such as GPS, IMU, etc.

4.2. Unity

Unity is a game engine developed by Unity Technologies that enables to develop multi-platform applications in 2D, 3D, VR, and AR. Its interface, tools, and services allow developers to easily create a professional prototype which gives the possibility to drive their businesses.

Unity offers a powerful graphic engine and full-featured editor that enables to create a fast prototype and to deliver the application content on any media or device. Developers can expand the target market thanks to the cross-platform builder (on PCs, consoles, the web, mobile devices, home entertainment systems, embedded systems, or head-mounted displays). Tools and resources include the Unity Asset Store, Unity Cloud Build, Unity Analytics, Unity Ads, Unity Everyplay, and Unity Certification. Unity Technologies serves millions of registered developers, including large publishers, indie studios, students and hobbyists around the globe.

For this project, it will be helpful to use the Unity Networking library with a High-Level scripting API which provides developers access to commands with the most common requirements for multi-user application without needing to worry about the “lower level” implementation detail.

Another helpful service that is offered by Unity is an Internet Service to support the application throughout production and release, including Matchmaking service, a Relay server, List available matches, joint matches, etc.

4.3. Vuforia

Vuforia is a Software Development Kit (SDK) to make Augmented Reality renowned by being the best-in-class computer tool, ensuring a robust and reliable experience in varied environments. Vuforia supports the latest mobile devices such as phones, tablets, eyewear over Android, IOS and UWP platforms. Vuforia Technologies is a strategic partner of Unity, joining forces to make development of AR applications easier.

The Vuforia Computer Vision technology can recognize and track planar images (Image targets) and 3D objects, in real-time. This capability enables developers to position and orient virtual objects (in this case a 3D satellites, models or components), in relation with the real world viewed through the camera's display. Thus, Vuforia SDK supports a wide variety of 3D and 2D targets. For now it is able to recognize real objects, images with sufficient detail that are optimized with the help of the Vuforia Target Manager page (which analyzes and improves the images), User-Defined Images given by a photo from user camera, Cylinders such as bottles, cups and mugs, Text (with a support for English word recognition from a standard database), Box and VuMarks, which is a new generation of markes that provides a simple method for encoding data such URL or product serial number.

Vuforia has other interesting features which make world tracking improve the AR experience: Smart Terrain is a new capability for smartphones and tablets, that reconstructs a terrain in real-time, creating a 3D geometric map of the environment, allowing developers to interact with the physical
world. Extended Tracking, is a capability that continuously visualizes the virtual object, even when the target is out of the visual field or camera view, it is delivered thanks to mapping the environment.

4.4. Microsoft Hololens

Hololens is a pair of mixed reality smart glasses developed by Microsoft, it is part of a device generation which is placed in the eye vision and uses the Visual UI as an interface. It has an inner headband to fit on the user’s head and equally distribute its weight for comfort. The hardware is located in the front area including the sensor, the camera and the visor that projects the hologram applications.

These glasses feature an IMU, four "environment understanding" sensors, a depth camera with 120°x120° angle of view, a 4-megapixel video camera, a four-microphone array and an ambient light sensor. It also features an internal rechargeable battery with 2-3 weeks of autonomy, or 2 weeks of standby time with Wi-fi and a Bluetooth wireless connectivity.

Concerning the Visual UI, Hololens, through the use of the HPU (sensor systems), uses practical and natural interface commands such as gazes, gesture, and voice (the last one is not included as Visual UI). These allow users to use their gaze as a mouse pointer and click on it with tap gesture (like an imaginary pinch on air).

2D and 3D Hololens applications can be developed in Microsoft Visual Studio IDE however, in this project, Unity engine was chosen in collaboration with Vuforia SDK to create a 3D AR app to achieve homogenization in the code and determinate the identical coordinates in the real environment with other device that are in the same room with the support of Vuforia Target Image.
5. APPLICATION COMPONENTS

5.1. Unity Networking

To understand the networking behavior in this application, it is necessary to know some concepts of the High Level Scripting API, Internet Service and the Network components in Unity.

Firstly, the High Level Scripting API, also known as HLAPI, allows developers access commands which cover most of the common requirements for multi-user games without the lower level and some implementation details. It is built on the top of the lower transport real-time communication layer, simplifying the common task for multi-player games and allowing one of the participants to be a client and the server at the same time, in order to not require a dedicated server process.

5.1.1. Server and Host

Generally, the multiplayer application works with a dedicated server and multiple clients, Unity networking system provides the possibility of giving a client the role of the server alongside the role of a Local Client, instead of a Remote Client, so server and client share the same scene and communication is made through direct function calls.

The above networking system aims to simplify prototyping and coding so developers can think about one client, instead of Remote Clients and Local Client separately.

5.1.2. Communication

When a client wants to modify the state of an object the networking system proposes managing a set of function to handle this synchronization during run-time. There are three type of functions, the first is named State Synchronization that is done from the server to the clients, in the code, SyncVars custom attribute is used to define the member variables of a NetworkBehaviour scripts that are synchronized (a NetworkBehavior script is like aMonoBehaviour script but with some additional properties for networking such as work with NetworkIdentity, if you don't know about MonoBehaviour script, it is advisable to refer to Unity Documentation for more information). When an object is spawned, or a new player joins a game in progress, they are sent the latest state of all SyncVars on networked objects that are visible to them.
The next two types of functions are known as Remote Actions or Remote Procedure Calls (RPC), one of them are Commands, which are called from the client and run on the server; the others are ClientRpc Calls, which are called on the server and run on clients.

In the application, a Command is called by a client process from a client object and sent to a client object on the server process, e.g. if a client wants to change the color of a satellite component and updates their modification on the server scene, they have to do a Command on one of their objects (the objects of which this client has their authority) and this Command will be executed on the same object that resides on the server. For security, these Commands can only be sent from this client object, so it cannot control the objects of other clients.

In the previous example, it is also necessary updates the modification on the rest of the clients, even for the client who has done the Command. For this case, ClientRpc allows sent a call from the object on the server to be run on clients. These calls can be sent from any server object with a NetworkIdentity that has been spawned. Since the server has authority, then there are no security issues with server objects being able to send these calls.

---

**Figure 8 - Communication Calls Scheme**

The diagram illustrates the communication calls between the server and client objects. The server side includes functions such as Object Spawn, State Updates, ClientRPC Calls, and Object Destroy, while the client side includes Connect, Ready, Commands, and Disconnect. The calls are indicated with arrows, showing the flow of data and control between the two processes.
5.1.3. Instantiate and Spawn

Unity uses a special network system to manage the synchronization of the Game Objects in the current scene. The Instantiate function creates a new Game Object but it must also be "spawned" to be active on the network, this is named "Spawning" and this function can only be called on the server to create the object on the clients connected.

This enables the object to add a set of network components to be handled by the server and the clients, like a unique Id used to route message and identify them. In this way, the current state of the object is sent to the clients and if another client joins the game later, the object will also be spawned on that client, avoiding issues such as objects being spawned at a wrong initial location, then popping to their correct position when a state update packet arrives.

The actual flow of operations that takes place for spawning is:

- Prefab with NetworkIdentity component is registered as spawnable
- GameObject is instantiated from the prefab on the server
- Game code sets initial values on the instance
- NetworkServer.Spawn() is called with the instance
- The state of the SyncVars on the instance on the server are collected by calling OnSerialize() on NetworkBehaviour components
- A network message of type MsgType.ObjectSpawn is sent to connected clients that includes the SyncVar data
- OnStartServer() is called on the instance on the server, and isServer is set to true
- Clients receive the ObjectSpawn message and create a new instance from the registered prefab
- The SyncVar data is applied to the new instance on the client by calling OnDeserialize() on NetworkBehaviour components
- OnStartClient() is called on the instance on each client, and isClient is set to true
- As gameplay progresses, changes to SyncVar values are automatically synchronized to clients. This continues until game ends.
- NetworkServer.Destroy() is called on the instance on the server
- A network message of type MsgType ObjectDestroy is sent to clients
- OnNetworkDestroy() is called on the instance on clients, then the instance is destroyed.

5.1.4. Authority

To avoid clients invoking commands on another client's object or even on server objects, Unity developers have set a configuration of Authority System, where the object is associated with each respective client, and their commands are routed to that object. So there is the property 'isLocalPlayer' in the object that is set to 'true' when the association is made with a client's local connection. This can be used to filter input processing, to handle client side commands that should only be present for one client.

A local object associated with a no remote client can have a "local authority". This means that local object on its owner's client is responsible for the object, allowing local clients to execute commands on the local side only. In the event that non-local objects are not associated with a client, the authority resides on the server.
The above diagram shows the objects involved in the networking actions and the concepts of authority, visibility and hosting. The different environments are surrounded by lines, specifying the server and the clients. The Host, as has been already mentioned, is a client which can perform the role of a server in the same process and keep the communication between both with a system of functions calls between Local Client and Server. The authority can be identified on the objects icons, so when a silhouette object is filled means that process (client or server) has a local authority, otherwise, if the object is a contour means the process has no authority on it. It is also important to notice that a not-spawned object is not visible in the client’s scene.

5.1.5. **Unity Multiplayer Service**

Unity offers a growing range of complementary services to help developers make applications and engages audiences. Unity’s Multiplayer Service is the easiest way to set-up real time, networked applications for Unity, avoiding issues such as server availability, security on the server, match management, etc. So in this project the developer profit this advantages for prototyping purpose. Unity provides servers and matchmaking services ensure that users can easily find a 'Room' with another user in and interact with each other (for this application the noun match will be treated as a 'Room'). The device that runs the application only needs to be connected to the internet, to get access to matchmaking Unity servers and enter or create a 'Room'.

5.1.6. **Remotes 3D CAD models**

Following some stakeholder's requests, it was important for the application to be able to download any 3D CAD model from a server and display it in the application, knowing that this 3D CAD model has to be post-processed and compressed before being sent by the server, with the purpose of simplifying the work processor and the files size.
This means that the mentioned application have to build new 3D models (make an archival format comprehensible for Unity Engine) which have not been built before installation. For that, Unity proposes the AssetBundles system that provides a data delivery method for non-code content after installation, this subject will be explained and more detailed in the next section, the aim of this section is to address the method used to get those AssetBundles.

5.1.7. FTP implementation

To get the 3D models from the server it was proposed to use the File Transfer Protocol(FTP), it is a standard network protocol used for the transfer of computer files between a client and server (FTP server) on a computer network.

Its model architecture is based on a client-server connection and uses separate controls such as authentication protocol (user and password) and SSH File Transfer Protocol (SFTP). For reasons of the scope and time, SFTP was not implemented. Another approach to improve the security and avoid problems with NATs and Firewalls is for the NAT to alter the values of the PORT command, using an application-level gateway for this purpose.

FTP has a stateful control connection which maintains a current working directory and other flags, and each transfer requires a secondary connection through which the data are transferred.

Most common web browsers, included UnityWebRequest system, can retrieve files hosted on FTP servers. When an FTP—rather than an HTTP—URL is supplied, the accessible contents on the remote server are presented in a manner that is similar to that used for other web content.

5.1.8. UnityWebRequest

The UnityWebRequest is a modular system for composing and handling HTTP messages (requests and responses). Its primary goal is to allow Unity applications to interact with web browser backends; For this context, the UnityWebRequest system will be used to handle the FTP Request to an FTP server.

The architecture of this system provides an interface for each important operation in an HTTP transaction:

- An UploadHandler object handles transmission of data to the server
- A DownloadHandler object handles receipt, buffering and post-processing of data received from the server

A UnityWebRequest object manages the other two objects, and also handles HTTP flow control. This object is where custom headers and URLs are defined, and where error and redirect information is stored.
UnityWebRequest.AssetBundle function is used to download an AssetBundle from a remote server, streaming data into an internal buffer, which decodes and decompresses the AssetBundle’s data on a worker thread. This function takes the URL server as an argument that it will be the FTP server address and attaches a handler (DownloadHandlerAssetBundle) that is used to extract the AssetBundle once enough data has been downloaded and decoded.

5.2. Networking Architecture

The next diagram explains the network architecture used on the application. It is clear that there are two kinds of flow which were separated by two different types of message.

![Network Architecture Diagram]

Figure 10 - Network Architecture

The first flow is related to the 3D CAD models and therefore it is desired to keep this confidential flow in the company’s intranet. That is why it was proposed FTP architecture with an FTP server hosting the 3D model files and receiving FTP requests and with a configured Ports NAT that establishes a specific port for the users.

The second is dedicated to the network messages such as the client’s calls and the objects variables synchronization. So it is not necessary the confidentiality for this kind of flow and availability, reliability, safety, security, integrity and maintainability aspects are delegated to the Unity Services Servers. This allows users to connect to other users without necessarily being on the same network.

Different approaches could have been implemented, such as handling the match connection and the transfer of 3D CAD models in the same local server or avoiding the fact of having a dedicated server to handle the matches. Nevertheless, one of the project goals is considering all viable alternatives and to put forward a prototype planned over the long term.
5.3. AssetBundles

An 'Asset' is a representation of any item that can be used in the Unity project and at the time of doing a Built and generating an executable of this project, the asset will be serialized and contained in the application. An asset may come from a file created outside of Unity, such as a 3D model, an audio file, an image, or any of the other types of file that Unity supports. There are also some asset types that can be created within Unity, such as an Animator Controller or a Render Texture. In other words, an asset is a file on disk, processed into native formats for Unity and stored in the Assets folder of a Unity project.

The difference between an 'Asset' and an 'AssetBundle' is the AssetBundle like Assets, it is a Unity content but it is generated after installation. This means applications that use AssetBundle system can download and serialize new content at run-time. Thanks to the capacity of Unity to index remote files in archival format, developers can reduce shipped asset size, minimize runtime memory pressure, and selectively load content that is optimized for the end-user's device. The purpose of the project using AssetBundles tools is to provide a set of 3D CAD models on the spot, preventing the memory and processor issues.

An AssetBundle has two parts: a header and a data segment.

The header contains information about the AssetBundle, such as the AssetBundle's identifier, whether the AssetBundle is compressed or uncompressed, and a manifest.

The manifest is a lookup table keyed by an Object's name. Each entry provides a byte index that indicates where a given Object can be found within the AssetBundle's data segment.

The data segment contains the raw data generated by serializing the Assets in the AssetBundle.

Hopefully, Unity develops and maintains a reference implementation of an AssetBundle Manager (it is open source) that provides a useful starting point to integrate the AssetBundles workflow into the project.

Introduced on mobile platforms in Unity 5.3, the UnityWebRequest API provides a more flexible alternative to Unity’s WWW API. UnityWebRequest allows developers to specify exactly how Unity should handle downloaded data and allows developers to eliminate unnecessary memory usage. The simplest way to download an AssetBundle via UnityWebRequest is the UnityWebRequest.GetAssetBundle API.

The AssetBundle component workflow is:

Before Installation.
- Import in the Asset folder a 3D CAD model in the Unity Editor to be processed as Asset file.
- Create an Object from the Asset and set it with the wished configuration.
- Tag the object with the name of the AssetBundle to serialize.
- Build the AssetBundles for each platform on which the AssetBundle is being Load.
- Place the AssetBundles files in the selected server repertory to be uploading.

At run time.
Use a UnityWebRequest’s DownloadHandlerAssetBundle to move the file from the server to the cache device.

5.4. HoloToolKit for Unity

HoloToolKit, also known as MixedRealityToolKit (MRTK), is a collection of scripts and components intended to accelerate the application development on Microsoft HoloLens. The MRTK-Unity uses code from the bases MRTK and makes it easier to consume in Unity.

5.4.1. Features Used

MRTK provides a set of functionalities that could be profited by HoloLens technology and make simple the application implementation. The features areas include Input, Sharing, Spatial Mapping, Spatial Understanding, Spatial Sound, UI Controls, Utilities and Build tools. In this project, the spatial and sharing tools were discarded because they were replaced by Vuforia tool and UnityNetworking system that homogenize the multi-platform code implementation.

5.4.2. Input

The input model is designed to handle the gaze, gesture, voice and motion controllers inputs. It could support various types of gazers. Each input source (hands, gestures, others) implement an IInputSource interface. The interface defines various events that the input sources can trigger. The input sources register themselves with the InputManager, whose role it is to forward input to the appropriate game objects.

In the project case, the input interfaces consumed by Game Objects, such as buttons and 3D CAD models components, are:

- **IFocusable** for focus enter and exit. The focus can be triggered by the user’s gaze or any other gaze source.
- **InputHandler** for source up and down. The source can be a hand that tapped, a clicker that was pressed, etc.
- **InputClickHandler** for source clicked. The source can be a hand that tapped, a clicker that was pressed, etc.

The input manager listens to the various events coming from the input sources, and also takes into account the gaze.

In recap, the input manager forwards the various input sources events to the appropriate game object, using the following order:

- The registered modal input handlers, in LIFO (Last-In First-Out) order of registration
- The currently focused object
- The fallback input handlers, in LIFO order of registration

The module, also have some prefabs related to input features, with the interface codes included in the GameObject.
**DefaultCursor.prefab**: 3D animated cursor that follows the user’s gaze and uses the Unity animation system to handle its various states. This cursor imitates the HoloLens Shell cursor.

**HoloLensCamera.prefab**

Unity camera that has been customized for Holographic development.

- Camera.Transform set to 0,0,0
- 'Clear Flags' changed to 'Solid Color'
- Color set to R:0, G:0, B:0, A:0 as black renders transparent in HoloLens.
- Set the recommended near clipping plane.
- Allows manual movement of the camera when in editor

**InputManager.prefab**: Input system that manages gaze and various input sources currently supported by HoloLens, such as hands and gestures.

This also includes a fake input source that allows the developer to simulate hand input when in the editor. By default, this can be done by holding Shift (left hand) or Space (right hand), moving the mouse to move the hand and use the left mouse button to tap.

### 5.4.3. UI

UI module has useful common UI controls, allows an easier construction of 2D Objects from the 3DTextPrefab and UITextPrefab which are legible and with the accurate resolution for the Camera.

### 5.4.4. Utilities and Build

There are common helpers and tools to build and deploy automation windows for Unity Editor.

### 5.5. Vuforia SDK for Unity

Vuforia SDK is another package that like HoloToolKit, it attempts to accelerate the process development providing components, code, prefabs and editor utilities to add advanced computer vision functionality to any application.

Even if the set of features offered by Vuforia SDK is vast, the current scope project has limited its use to only detect an 'Image Target' (an Image that the SDK can detect) and takes advantage of Extended Tracking capability.

#### 5.5.1. Image Target

It is worth noting, Image Targets do not need special black and white regions or codes to be recognized. Thanks to comparing the natural features of the images against a known target resource database (it will be explained later), the SDK achieves to detect and track the image. Once the Image is detected, the image will be tracked as long as it is at least partially in the camera's field of view, if the image is out of this field, the Extended tracking feature will help to continue the image tracking.
An Image Target can be created from the platform online of ‘Vuforia Target Manager’ that supports JPG and PNG images in RGB or grayscale with a size maximum of 2MB, from this images, it generates a downloadable database of the stored features. This database improves detection in run-time.

There are two phases to create an Image Target, First choosing or designing the images and then upload it to the Vuforia Target Manager for processing and evaluation. An ideal image target should have the next attributes to enable the best detection:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rich in detail</td>
<td>Street scene, group of people, collages and mixtures of items, and sport scenes</td>
</tr>
<tr>
<td>Good contrast</td>
<td>Has both bright and dark regions, is well lit, and not dull in brightness or color</td>
</tr>
<tr>
<td>No repetitive patterns</td>
<td>A grassy field, the façade of modern house with identical windows, and a checkerboard</td>
</tr>
<tr>
<td>Format</td>
<td>Must be 8- or 24-bit PNG and JPG formats; less than 2 MB in size; JPGs must be RGB or greyscale (no CMYK)</td>
</tr>
</tbody>
</table>

**Figure 11 - Good Marker Requirements**

The image below is the picture designed to be processed as Image Target that meets the above requirements. The repetitive pattern in the background is for branding purpose not for detection however, this pattern feed the natural features of the image
A Natural Feature is a sharp, spiked, chiseled detail in image that helps tracking; these natural features must be well-balanced and well-contrasted in image avoiding repetitive patterns and organic shapes. In the next image the natural feature are represented by small yellow crosses detected by the image analyzer.

![Image showing natural features detected by image analyzer](image)

**Figure 13 - Image natural features**

### 5.5.2. Extended Tracking

Extended Tracking is a feature that uses the environment to improve tracking performance even when the target is no longer in view field.

When a target has been detected Vuforia uses other information from the environment to infer the target position (such as visually tracking of environment, figure 14) and keep a degree of persistence once the target goes out of view. Vuforia maps around the target and assumes that target and environment are largely static. The Figure 15 shows how it works displaying the satellite even when the marker is out of field of view.
This mentioned capability can enhance user experience allowing the operators to point the device away from the target following the 3D CAD model and, visual and scale large 3D CAD models.

5.5.3. **Extern Components**

There are two extern components to take account in:

**The Vuforia Target Manager**, it is a web based tool that enables the developer to create and manage target databases online.

**Device Database**, its purpose is to provide the AR application with a locally accessible database of targets.

A Vuforia Developer account is required to begin working with the Target Manager.

Once the license key given by the web platform is defined in the License Manager, the Database flow is:
• Create a database
• Add targets
• Download the database and add it to the project
• Update and manage the databases and targets throughout the life of the app.
6. ARCHITECTURE

See 'Unity Networking' in 'Application Components' section to know about Network Architecture and their components.

6.1. Package Architecture

This diagram describes the content in the asset folder, including the distribution of each package. Each folder has a specific purpose:

- **Editor** - Contains the scripts required to interact dynamically with project data in the Unity editor, so developer is able to build AssetBundles, configure scene and properties to Vuforia and HoloLens implementation.
- **Plugins** - Contains Java and native binaries that integrate the Vuforia AR SDK with the Unity Android or Unity iOS application
- **Vuforia** - Contains the prefabs and scripts required to bring augmented reality to your Unity application
- **Resource** - Streaming Assets / QCAR - Contains the Device Database configuration XML and DAT files downloaded from the online Target Manager.
- **Materials** – Contains the Unity material files used in the project.
- **Prefabs** – Contains the specific GameObjects used in the project.
- **Scenes** – Contains the two scenes executed in application.
• Layouts – Contains the 2D art files used in application such as JPG and PNG files to feed into UI.
• Satellite-Default – Contains the satellite model and its prefab used in case the connection with models server is unreachable.

6.2. Sequence Diagram

The following main sequence diagrams help to understand the Method Calls between the processes and objects in run-time, and how network calls is handled by NetManager class which calls to other functions on network objects. These diagrams are based on UML 2.0 specifications.

6.2.1. Create a Room

This diagram shows the sequential order of Creating a Room, that means and operator (user) could be the host by having a local client and a server simultaneously and create a match (session) where other user may connect. It is worth noting that spawn action is executed once the server is ready and ‘OnServerReady’ call is made when the client get the server connection receiving a message from server with AssetBundle information to download it.

![Figure 17 - Create a Room sequence diagram](image)
6.2.2. Join a Room

This diagram shows how clients establishes connection with host by using the NetworkManager and the NetworkMatch, once the matchMaker had found the match (Room) according to match specifications sought, if the match exists, the NetManager set the host connection to the client.

![Join a room sequence diagram](image_url)

**Figure 18 - Join a room sequence diagram**

6.2.3. Download a 3D CAD Model

In this diagram, it is important to note the DownloadModel call is only made by Host user, the method verifies if a model is already downloaded in order to destroy it, then it make a RPC call (*execute the command on each client scene*) to start a coroutine (*a routine in parallel*) to establish a connection with the server which host the 3D CAD models and another coroutine to download and instantiate an AssetBundle.
Figure 19 - Download 3D model sequence diagram
6.2.4. Paint Model Component

The purpose of this diagram is shown by the sequence of a RPC call, that means the other calls to modify the 3D CAD model components have the same procedure, such as explode model, resize model, rotate model. A Client executes an action that execute a Command Call (Cmd) received by the Server (Host) and then it makes a RPC Calls to all clients to update 3D model states, and it works for each command on the model components.

Figure 20 - Paint model component sequence diagram
7. **Results**

I have developed and delivered an executable application called ARound as AR tool prototype with the purpose of proving the potential utility in the industry of satellite conception.

The first version, allows user to do the following functions:

- Connect to a local Host (any device with ARound application in the same network)
- Being the Host to create a Room and receive other client’s connection.
- Display a 3D model (a not real satellite created for videogames environments) over the AR market.
- Change the material color of a 3D model component.
- Turn the 3D model left and right.
- Explode the 3D model (disassemble and assemble the components from the 3D model).

![Figure 21 - UI Around app v1.0 Start](image-url)
The second version was subjected to some changes after a couple user tests with some stakeholders and the feedbacks from UI and networking validation.

The functionality of the second version is:

- Connect to a Remote Host (Any device with ARound connected to Internet).
- Being the Host to create a Room and host any client’s remote connections.
- Download and display over AR market any post-treated 3D CAD models hosted in a Local Server.
- Change the material color of a 3D CAD model component.
- Turn up, down, left and right the 3D CAD model.
- Explode the 3D CAD model (disassemble and assemble the components from the 3D CAD model)
- Take a screenshot of current view and current modifications on the 3D CAD model

It is noteworthy that the arrangement of the UI was changed according to UI principals of composition avoiding user manuals to use the application, and a real TAS satellite model was added.
Figure 23 - UI Around app v2.0 Start

Figure 24 - UI Around app v2.0 model modified
Figure 25 - Screenshot functionality
8. CONCLUSION

This project has shown the great potential of implementing AR in the industry and how interesting AR tools could be in meeting rooms where discussions on CAD objects are expected. That also opened the doors to continue exploring the benefits of AR and keep working with the new generation of AR/VR devices. AR aspect was not only the remarkable feature of the application, also the attributes such as portability, security, and availability were primordial to the conception and development phases, looking to fill the stakeholder’s necessities.

In addition, the project has proved to be a valuable step to AR industrialization area in TAS and how important is not staying away from the upcoming generation of AR/VR technology. The project’s results were absolutely positive even if they were not quantifiable, the goal project was achieved, develop a cross-platform application that allows users to display and work in collaboration on a downloadable 3D CAD model, and it got the attention of some internal investors.

This work will be taken by another IRD developer in order to carry on with the labor achieved thus far and not lose that progress. There are also some features that could be re-evaluated and/or improved such as depending on an external company server and their services (in terms of reliability and budget), fixing Unity engine as develop environment for a complex and heavy application, hosting 3D CAD models in a FTP Server (in terms of security), functions (adding more actions and tool to interact on), etc.

The code documentation, the project’s set up's manual and a last demonstration of the application, are the last work to settle for the rest of my internship that they could not be registered in this document for reasons of report’s deadline.

Developing this software application has given me the opportunity to show my skills and how I found my way to confront a real problem in the industry of innovation. The fact of putting me in unexplored fields which I had never worked on, allows me to challenge myself, in order to create, master and choose the suitable means, among a wide range of new existing technics to find the best solution, one of the most common cases in professional life.
9. BIBLIOGRAPHY

- AUGMENTED AND VIRTUAL REALITY HEADSETS EXPECTED TO GROW AT A 5 YEAR CAGR OF 108.3%, ACCORDING TO IDC’S NEWLY LAUNCHED AR/VR TRACKER

- BELL, D. The sequence diagram

- CONFIGURER VOTRE COMPTE FTP DANS FILEZILLA | FILEZILLA CLIENT FTP

- GETTING STARTED

- PANETTA, K. Top Trends in the Gartner Hype Cycle for Emerging Technologies, 2017 - Smarter With Gartner

- Unity - Manual: Multiplayer and Networking

- Unity - Manual: AssetBundles

- Augmented Reality Systems

- A Survey of Augmented Reality