



KTH Byggvetenskap

Civil and Architectural Engineering
Kungliga Tekniska Högskolan



**Dipartimento di Ingegneria Strutturale,
Edile e Geotecnica**

Ingegneria Edile
Politecnico di Torino

Data Management and Virtual Reality Applications of BIM models

Datahantering och Virtuellt Verklighet Applikationer av BIM modeller

Applicazioni di Gestione Dati e Realtà Virtuale per modelli BIM

Masters Thesis in Building Engineering

November 2018

Supervisor

Kjartan Gudmundsson

Supervisor

Anna Osello

Francesca Maria Ugliotti

Candidate

Riccardo Levante

Abstract

Nowadays new digital technologies play an important role in the AEC sector, which are going through a significantly changing period. In this process BIM represent the main character, revolutionizing the traditional way of working in the main operative phases: Design, Construction and Maintenance. However, having all these aspects under a unique methodology can brings several problems without a strong framework to regulate the whole procedure. The huge quantity of information, as well as the high number of involved people, are the main weak points, that can become the main strong points when Facility Management is involved in the process. In this scenario this work will analyse a real project located in Genova (Italy), seeing as a complex structure as a hospital can be regulated through the BIM methodology.

The thesis will focus on two fundamental aspects of the new construction methodology: Data Management and Virtual Reality. The first part aims to analyse correlation between BIM and its information, proposing operative solution through the BIM software Revit and the visual programming toll Dynamo. The second part will show advantage integrating virtual reality technologies in the AEC sector, showing how realize an interactive virtual simulation using the game engine software Unity.

Through this work the importance of new technologies for the future of Construction Industry have been debated, showing how important is the sharing information involving always more people into the process, particularly for complex structures.

Contents

1	Introduction	5
1.1	Scenario	5
1.2	Work Methodology	5
2	The BIM methodology	7
2.1	BIM Definition	7
2.2	BIM and VR for AEC industry	9
2.2.1	Design Phase	9
2.2.2	Construction Phase	10
2.2.3	Maintenance Phase	10
2.3	BIM for Healthcare	10
3	The Galliera Hospital.....	12
3.1	The New Project.....	13
3.2	BIM for the Galliera.....	15
4	Data Management	17
4.1	Data Standardisation	17
4.1.1	IFC and COBie	18
4.1.2	Space Management and Data Population	19
4.1.3	Data requirement system for a new project	19
4.2	Work Methodology	21
4.2.1	Identification of target and data.....	21
4.3	Dynamo for Data Management.....	22
4.3.1	Dynamo for Export data – Application 1	23
4.3.2	Dynamo for Export data – Application 2	28
4.3.3	Dynamo for Export/Import data – Application 3	31
4.4	Results	37
5	Virtual Reality	39
5.1	VR Technologies in AEC industry	39
5.2	Work Methodology	41
5.2.1	From Revit to Unity.....	41
5.2.2	Unity Interface.....	43
5.3	VR Not-Immersive Application	45
5.3.1	Main Scene Development.....	45
5.3.2	3D Model Scenes Development	48
5.3.3	Navigation Scene Development	51
5.3.4	Standalone Game Creation	55
5.4	VR Immersive Application	57
5.4.1	Re-elaboration for Immersive Simulation	57
5.4.2	3D Model Immersive Scene	59
5.4.3	Participatory Project Design Scene	60
5.5	Results	63

6 Other Application –.....	64
6.1 Mixed Reality.....	64
6.2 Holograms	68
7 Conclusion – Hold conclusion to review.....	70
Reference.....	71

List of Figures

Figure 1: BIM along the construction value chain (Word Economic Forum, 2016).....	8
Figure 2: Aerial photo (Galliera, 2018)	12
Figure 3: New project (Galliera, 2018)	14
Figure 4: BIM standardization platform (Poljanšek, 2017).....	18
Figure 5: Relationship between elements of information management (BSI, 2014)	20
Figure 6: Anatomy of a Node (Dynamo)	22
Figure 7: Dynamo script - Application 1	23
Figure 8: Dynamo script – Application 1.1	23
Figure 9: Dynamo script – Application 1.2	24
Figure 10: Dynamo script – Application 1.3	25
Figure 11: Dynamo script - Application 1.4.....	25
Figure 12: Dynamo script - Application 1.5.....	26
Figure 13: Dynamo script – Application 1.6	27
Figure 14: Dynamo Player (Dynamo)	28
Figure 15: Dynamo script – Application 2	29
Figure 16: Dynamo script – Application 2.1	29
Figure 17: Dynamo script – Application 2.2	30
Figure 18: Dynamo script – Application 2.3	31
Figure 19: Dynamo script – Application 3.1	32
Figure 20: Dynamo script – Application 3.1.1	32
Figure 21: Dynamo script – Application 3.1.2	33
Figure 22: Dynamo script – Application 3.1.3	33
Figure 23: Dynamo script – Application 3.2	34
Figure 24: Dynamo script – Application 3.2.1	34
Figure 25: Dynamo script – Application 3.2.2	35
Figure 26: Dynamo script – Application 3.2.3	35
Figure 27: Dynamo script – Application 3.2.4	36
Figure 28: Application 3 – Example.....	37
Figure 29: Oculus Rift headset (left) HTC Vive (Right).....	39
Figure 30: Example of VR use with BIM model	40
Figure 31: Revit – Unity interoperability process.	42
Figure 32: Export/Import “.fbx” format file.	42

Figure 33: Export process from 3DS Max.....	43
Figure 34: Unity structure (Boeykens, 2013)	44
Figure 35: First Workflow Development.	45
Figure 36: Unity-Button Creation.....	47
Figure 37: Unity-Main Scene Interface.	48
Figure 38: InfraWorks - Create Model.	49
Figure 39: Unity – Urban Context Scene.	50
Figure 40: Unity – Level Scene.	51
Figure 41: Unity – Scene Creation.	52
Figure 42: Unity – Player Creation.....	53
Figure 43: Unity – Interactive Object Creation.	53
Figure 44: Unity – Examples Navigation Scene.	55
Figure 45: Unity – Standalone File Creation.....	56
Figure 46: Second Workflow Development.	57
Figure 47: Unity – Immersive Scene Creation.	58
Figure 48: Unity - Immersive Main Scene.	59
Figure 49: Immersive 3D Model Scene.....	59
Figure 50: Unity – Data Interaction.....	60
Figure 51: Unity – Player Configuration.	61
Figure 52: Unity – Interactive Menu.	62
Figure 53: Unity – Immersive Participatory Design Scene	62
Figure 54: Microsoft HoloLens application (Microsoft, 2018).....	64
Figure 55: HoloLens Headset.	65
Figure 56: HoloLens Configuration.	65
Figure 57: Hologram Creation.....	66
Figure 58: HoloLens Standalone Creation.	67
Figure 59: Microsoft HoloLens application.	67
Figure 60: Olobox Design.	68
Figure 61: Hologram Video Creation.	69
Figure 62: Olobox Application.....	69

1 Introduction

1.1 Scenario

The AEC (Architecture, Engineering and construction) industry is going through a period of strong development. Many recent technologies start to be essential to ensure a successful construction process and new engineering methods and tools provide to make possible developing of smart cities. In this scenario, however, the facility delivery process remains fragmented, and it depends on paper-based modes of communication. Errors and omissions in paper documents often cause unanticipated field costs, delays, and eventual lawsuits between the various parties in a project team (C. T. Eastman 2011). These problems damage the whole construction workflow, reducing the quality of the result. Often, in fact, without a correct information methodology and with 2D-based communication process, it is common that only on the last phase important analyses are done (cost-estimate, Energy-use analysis, structural detail, etc.), where it is already too late.

Building Information Modeling (BIM) marks a turning point for the whole AEC industry, improving collaboration, efficiency and productivity. This method represents one of the most important revolutions in the construction sector, focusing the attention on the information aspect of each project and contributing to improve well-designed Facility Management (FM) systems. Data management, in fact, becomes the main character in the lifecycle of a building, from the first draw until the maintenance plan, and in this context FM systems cover an important role. BIM and FM aspire to solve all interoperability problems that always have characterized the building sector.

After the information, another important point of BIM is the method based on 3D model, which brings new possibilities of architecture visualization that no one could image. Virtual reality (VR) technology, in fact, provides to cover the previous gap of architectural communication, making possible to see the result before it is completed through a realistic and immersive simulation of the project.

These aspects start to be necessary to cover the growing complexity of health care design, where to manage a huge number of data and documentation, for all involved people, represents a big challenge. New design technologies are essential in healthcare projects, to ensure success both in design and construction phase but particularly in the operation and maintenance phase. This because for healthcare buildings, differently for other structures, non-professional people play an important role (patients, visitors, medical staff etc.) in the whole building lifecycle, making essential to have a single database easily accessible for all stakeholders.

1.2 Work Methodology

Starting from this background, this research aims to explain the role of BIM in the healthcare field, focusing on the importance to make easier sharing information and architecture visualization. The reasons of this choice come from the difficult to manage complex buildings, as our study case, where several actors are involved in the process and where to have a clear procedure is essential for the result. Several possible issues will be analyzed that can characterize a BIM process and how they can be solved through a methodology based on data management and virtual reality.

The report starts talking of the general BIM definitions to move then on the specific relationship between BIM and VR and BIM for healthcare industry. One of the main part developed for this project will concern the important aspect of data management in a digital FM workflow based on BIM, starting from a theoretical focus on the standard regulation for this sector. The applications developed for this part see as main character the software of visual programming Dynamo, used to propose interactive method to make easier sharing data between professional and non-professional people. The second part will be dedicated to present some examples of VR uses for AEC industry, explaining the workflow to create a virtual environment through the gaming software Unity. Applications will be characterized by an immersive and not-immersive solution.

In all this process our study case will have a leading role. It is represented from a new hospital project that will be built in the city of Genova (Italy) by the public customer Ente Ospedaliero Ospedali Galliera (EOOG).

2 The BIM methodology

How said BIM stands for Building Information Modelling and/or “Management/Model” according to the *Making BIM a global success*, Manifesto published by the European Construction Industry Federation (FIEC), where it also specified that BIM is a process driven working and management method based on collaboration and sharing data throughout the entire construction value chain, including clients and final users. More generally it consists in a practical database-centered working with the aim to use and analyze huge quantities of data and information; representing the base of the overall digitalization of the construction sector and “Industry 4.0”. In the last twenty years BIM has completely revolutionized the whole AEC industries, setting significant changes not only in the lifecycle of a building but also in the roles and relationships among a project team.

2.1 BIM Definition

Thinking at the beginning of digital drawings, older CAD (Computer Aided Design) systems were not so different from the hand-drawing. They, in fact, were general file based on line-types and layer identification to produce plotted drawings, that it basically the same to draw with a pencil, but faster. In the future CAD systems have been improved, first with the introduction of 3D modelling and then with the ability to include additional information on these files, but when the attention shifted from drawings and 3D images to data itself we can see a turning point. BIM methodology have been introduced placing at the center the importance to create and collect information, producing building models through interactive tools able to support multiple different views of the data contained within a drawing set, including 2D and 3D.

The NBIMS Initiative (National Building Information Modelling Standard) categorizes the Building Information Model (BIM) three ways:

- As a Product;
- As some IT-enabled, open standards-based deliverable, and a collaborative Process;
- As a facility lifecycle management requirement.

These categories support the creation of the industry information value chain, defining BIM as fundamentally different way of creating, using, and sharing building lifecycle data (C. T. Eastman 2011).

For the first time in the history of construction, data becomes the key for each phase of the building lifecycle (Figure 1) amassing large volumes of high-integrity information.

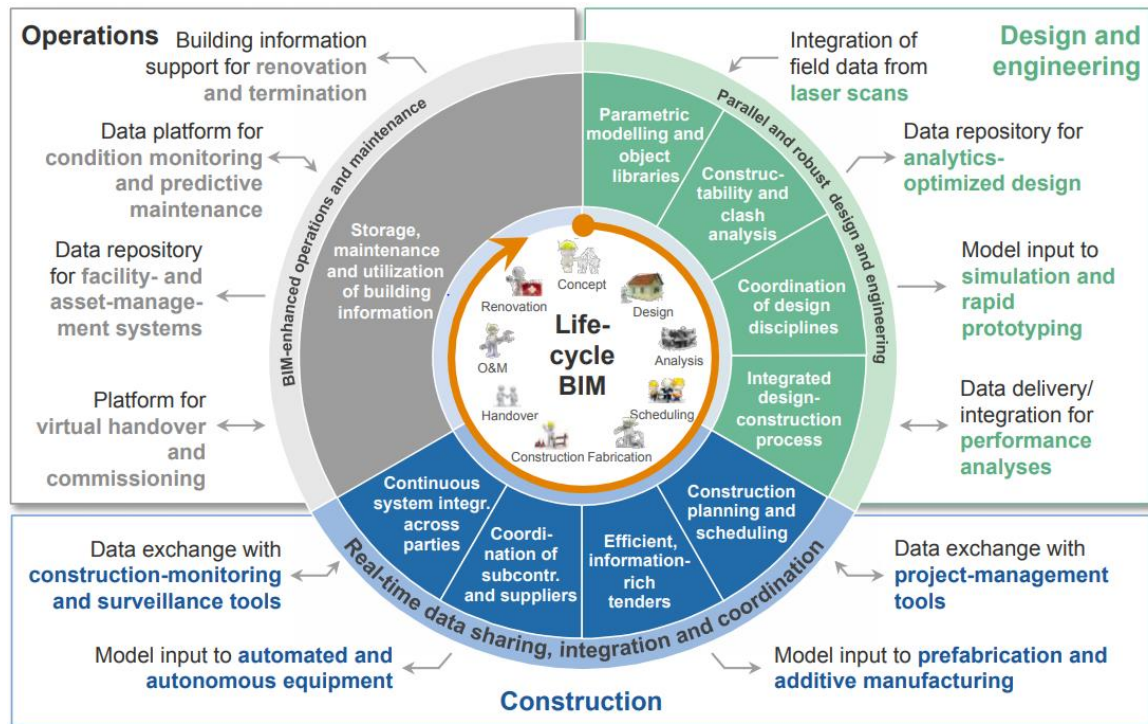


Figure 1: BIM along the construction value chain (Word Economic Forum 2016)

How explained above BIM has revolutionized not only the design phase, but the whole system, bringing many changes in construction, commissioning, operations, renovation and even demolition phase. Plus, the algorithm generates new sectors and new method changing the bases itself of the AEC industries. For example, new methods of simulation and virtual reality help to identify interdependencies and clashes during the design and engineering stages and enable a virtual experience of the building even in the early design stage (Poljanšek 2017).

Other two concepts are essential to understand the real definition of BIM and its differentiation from traditional 3D project; they are “parametric object” and “interoperability”.

With the first one it is not indicate an object with fixed properties and geometry, but an object composed by parameters and rules that determines the geometry as well as some non-geometric properties and features. The parameters and rules can be expressions that relate to other objects, allowing the objects to automatically update according to user control or changing contexts (C. T. Eastman 2011). These objects are also based on “different levels of aggregation”, which means that each element is related with its component, so it can be managed interacting with any of this level. Technologies that allow users to produce building models that consist of parametric objects are considered BIM authoring tools.

With the second aspect we mean the possibility to communicate and exchange data between software, making easier and faster the workflow. Interoperability bring to be necessary a standardization of data model, allowing to create a method to translate information from one schema language to another. In this scenario two main products are important: IFC (Industries Foundation Classes) for planning, design, construction and management and CIS/2 (CIM steel Integration Standard Version 2) for structural steel engineering and fabrication.

2.2 BIM and VR for AEC industry

Another purpose because BIM methodology was born is to improve the gap of communication and understanding between different actors that we can find in each project, to obtain a better workflow of the design process and a better quality of work. There are different meanings because problems of communication are so common in this sector, but the most common problem is that usually information is not presented in such a way that people can understand it. In fact, one important lack in the BIM life cycle is to allow to non-technical parties to cooperate in the design process.

In this context, real-time visualization and VR has been shown to offer an efficient communication platform. It offers a natural medium for building design providing three-dimensional visualization that can be manipulated in real-time and can be used collaboratively to explore design options and simulate different stages of the construction process (Bouchlaghem 2005). The use of this technology is always more accessible since the Building Information Model have assumed a principal position in the AEC field, indeed the required 3D data can be extracted from the architect's own design environment, instead of creating it from scratch using 2D-plans, elevations and sketches as a reference (M. R. Johansson 2014).

Virtual Reality can be described as a realistic and immersive simulation of a three dimensional environmental generated by a computer in which a person can naturally interact with the context by movement of the body (Milgram 1995). The aim of this technology is to replay the reality, as many as possible precise, and access to this interface via a computer (monitor, keyboard, mouse) or with other device that can create an immersive experience.

There are numerous possibilities of application for VR technologies and the number increases when we talk of big project, where there is a high number of people involved both in realization phase and in management phase. This is the case of our project where the study case is a hospital and consequently the actors aren't only engineers and architects that design the project, but also the end users of the building like patients, visitors, medical staff and people assigned maintenance. Respect to other projects this needed to be understood from several categories of people, all with difference knowledges and with difference point of view respect the building. In this scenario VR technology can help people involved in the project and not only for the Design process but also for the realization and management phase.

2.2.1 Design Phase

Nowadays this is the environment where it is possible find more applications and solutions for the VR technology in the building sector. Respect a 3D model which gives a limited experience of the finished building, VR gives an immediate understanding of spaces and features creating a realistic and immersive 3D simulation. This allow designer to explore and analyze the project in a 1:1 scale with the ability of walking inside or around the structure, making much easier to evaluate shape and design. In fact, for a correct hospital setting is important that end-users can move safety and quickly, and that physicians and staff can access the appropriate equipment easily. Using virtual reality, you have the possibility to evaluate the space disposition for each part of the structure and making this decision early you can save significant amount of time in the realization phase. In the end other advantages are represented as a communication tool, seeing as it makes more available to share information and ideas. This is useful for professional people (architect, engineer, etc.) and others to understand better each features of the project and to make easier and sliding the process.

2.2.2 Construction Phase

Some interesting applications about VR technologies concern also their use during building construction phase to provide technical information to the AEC sector operators. The most obvious benefit of virtual reality for construction companies is that it can save time and money by allowing construction professionals to test any number of features before the construction or installation phase starts. Each worker can analyze, with a suitable device, what is the correct position of a generic element or where a specific system must be installed, making easier obtain information and decreasing the possibility of errors. However also in case of problems during the construction phase virtual reality can be useful for the process evaluating the different proposed solutions directly on site.

Another Interesting application adopted by some construction company is to use VR technology for construction site safety. It proved useful for construction site safety training and to learn functions of operating heavy machinery, also explaining better the different site hazards for professionals and visitors with the opportunity to recreate the same location of building site.

2.2.3 Maintenance Phase

Virtual Reality is only in the initial state related with AEC industry but for sure Facility Management and maintenance aspect is where VR has more potential. In a building is not always possible access and analyze all the elements, usually systems are located inside wall, below the ground, behind a ceiling or in some troublesome place where it is dangerous to access. This needs to spend time locating the source of the issue before being able to implement a solution. With a virtual environment the worker can be helped to understand both the location of mechanical systems and how this equipment interacts with each other. Important is also the opportunity that virtual reality gives to analyze the buildings systems without be present on site. With this solution is possible to recreate the environment wherever you are and plus simulations can be run to see how a new equipment may impact existing operations, increasing in this way the velocity of the process and the safety for the intervention.

2.3 BIM for Healthcare

The healthcare industry is one of the largest and fastest-growing industries with a constant development to keep up with needs of patients. Healthcare construction projects have become increasingly complex and challenging from the facilities management perspective and appropriate healthcare design is essential for ensuring health and promoting wellness (Mohammadpour 2017). To achieve a great healthcare design is essential that design team involves in the construction process also stakeholder and medical staff (physicians, nurse practitioners, physicians' assistants). However, in a traditional design approaches with a 2D drawings and documents used to explain the healthcare design, it becomes difficult for the medical staff and stakeholders to understand the concept and content of those 2D drawings because they lack engineering knowledge and experience (Okada 2017).

In this scenario BIM represent a perfect solution because based on a 3D model technology and on a single database used along all the design and construction process. This technology related with healthcare and hospital needs during all the phases (design, construction and management) ensure to save time and money avoiding possible errors and delays which are very common in complicate projects like this. There are many advantages to adopt this technology in the healthcare industries, first of all it supports the design process easily and it makes possible to create a clear database with all the information. Then it can transform the design idea into

physicalizing and visualization and can be used for knowledge integration, performance analysis, cost estimation, 4D (time schedule) simulation and crash detection. Especially it exchanges information of the specific BIM model among different system, software and organization in favor to integrate information and to improve the communication efficiency (Collinge 2012) (Cooney 2012).

There are many previous researches using BIM methodology for healthcare projects. One of this sees as case study the Careggi Hospital in Florence, which is part of the STREAMER project with the aim to reduce the energy consumption and carbon dioxide pollutions for big healthcare structures and in this particular case developing a semantic approach to design starting from the combination of BIM and GIS systems (Iadanza E. 2015). Important is also the project for the New Karolinska Hospital which represents one of the biggest projects in healthcare in the County of Stockholm. The use of BIM was contractually mandatory, and they adopted this method for the entire lifecycle, creating a single data platform on which designers, contractors and eventually facility managers could collaborate. They started from the design specifications and performance requirements until every object have been digitally stored in the model, with its key attributes and exact location and with a final objects list of about one million items (University of Waterloo 2017). Another important example is the work of Manning and Messner which have verified BIM adoption during the concept and design phases of two healthcare projects, to provide visualization, time save, quantity take-offs, and decision making (Manning 2008). Also, the work of Khanzode Fisher and Reed represent a role model, about the case study using BIM/VDC (virtual Design and Construction) tool and process to coordinate the MEP systems of healthcare project in Northern California (Khanzode 2008).

These examples and the above argumentations explain how important BIM for healthcare industry could be. Following this road, we will talk of our case study, showing the structure and features of a real BIM project.

3 The Galliera Hospital

Everyone knows how it is important the role of a hospital in the city urban body and what is its importance under the social and economic aspects. The case of Galliera Hospital and the choice to build a new structure is particularly important since it is located to the center of Genova, one of the cities with a more complex urban environment in Italy. These aspects make difficult the whole building process, but they focus the attention on the methodology choice to follow preventing each possible problem and reducing the environmental impact.

The historical part of the Galliera Hospital (Figure 2) was built around the 1877 and 1888, located in the district of Carignano and near the Tyrrhenian coast, in an area with an irregular altitude profile and highly urbanized. The nineteenth-century building is composed by:

- A long construction with a length of 260 meters and a curvilinear profile, with two rectangular building situated at the extremes;
- Seven pavilions are inserted on the reverse side of the principal structure and each of them is about 50 meters long.



Figure 2: Aerial photo (Galliera 2018)

In the last years the hospital was main character to develop the project INDICATE (Indicator-based Interactive Decision Support and Information Exchange platform for Smart Cities), which was promoted by the European Union with the aim to develop a useful interactive software for all the people involved in the design and development phases of Smart Cities and a Green Hospital. The software to develop had to be founded on a cloud platform allowing

dynamic evaluations of the interaction between buildings, power grid, renewable energy systems and ICT (Information and Communication Technology) resources. The principal goal, as the official website declares, are:

- Plan using an energy-based decision support tool that can use dynamic simulation and considers buildings and occupants, the urban environment and energy distribution grids.
- Reduce energy consumption and CO₂ emissions by indicating the impact of energy conservation measures via dynamic simulation modelling.
- Integrate new technologies and services to manage supply and demand via dynamic simulation modelling, GIS and 3D urban modelling.
- Optimize existing systems to enable local balancing through demand response and tariff analysis, via dynamic simulation modelling of the interactions between buildings and their occupants, installed systems and energy distribution grids in the urban environment.

The subjects involved in the project have made available competences in several sectors like: dynamic modelling and simulation; geographic information system (GIS); software for 3D modelling (Cad3D-BIM); demand side management; urban sustainability indicator; integrate solutions to minimize energy and fossil fuel consumption.

In this context Galliera Hospital is the contribution to the demonstration of the INDICATE project and tools. It is, in fact, in the heart of the area chosen to develop the project, including about 60 hectares with many public buildings such as police headquarters, schools and youth centers which will be included in the study due to the relationship of the hospital with its neighbors. The INDICATE project focuses attentions on the historical part mapping and on designing of the new part through BIM software. All the information from the BIM model and from the GIS software will be collected in the Virtual City Model (La Franca 2016).

3.1 The New Project

The New Hospital project is born as reply to structural inadequacies of the existing structure and with the aim to increase the healthcare efficiency in a city with a constant progression like Genova. The nineteenth-century building presents several problems on the distributive aspects, due also to the complex shape, and many difficulties to adapt it on the new healthcare standards. The better choice to provide to these problems was to build a new hospital, next to the older one (Figure 3Figure 3), adapt to carry out all the requests and utilizing the technologies at the forefront in all the sectors.

After a period of standstill, the project of the whole New Hospital is moving on and it has been divided into two lots:

- The first one (Lot 1) for the construction of the new structure in a highly urbanized area, after some existing building are removed;
- The second one (Lot 2) for the restoration and repair of the current historical hospital halls, currently used for hospital activities that will be converted to host logical and organizational activities.



Figure 3: New project (Galliera 2018)

The area destined for the new construction is located on the south of the existing building with a total surface of about 26.000 m². The new building will have a global surface around 54.000 m² and structured in 7 levels, one basement, three floors partially underground and three floors aboveground. The Two big entrances, present in the opposite side of the buildings, will improve the distributive and safety aspects, obtaining one way for emergency and one way for patients and visitors. All the floors are designed to make easier the building management for all sectors; they are divided in the following way:

- *Level -1*: destined to parking service with a useful surface of about 5.580 m²;
- *Level +0*: In this level is presented the main entrance for the emergency rooms for the first aid with a surface of 2.320 m² and other activities as the diagnostic area (2.060 m²) and a zone for general laboratories (2.090 m²);
- *Level +1*: here the main sectors are the intensive care ward (1.936 m²), operating rooms (1.756 m²), operation radiology (629 m²) and surgery ward (1.134 m²);
- *Level +2*: here the main sectors are the radiotherapy ward (1058 m²), the outpatient clinic (1.848 m²), psychiatric convalescence ward (1.007 m²) and the dental center (332 m²);
- *Level +3*: In this level we find the other one main entrance with all the reception activities, then we find also the physical therapy area (1377 m²) and the day hospital zone (1084 m²);

- *Level +4 & Level 5*: In these two are present all the rooms allocated for the convalescence of patients with a total surface of 6916 m², plus the birthing rooms (1507 m²) and day surgery area (683 m²).

Other important objectives for the new realization is represented by using innovative system for the energy consume, respecting the Green Guide for Healthcare and adopting building design standards of LEED (Leadership in Energy and Environmental Design).

However, the realization does not consist of just a “simple” new hospital building construction, but of a much more complex operation, that will be developed in phases and which requires that the existing hospital functionality is kept working, before, during and after the new building construction phase. This aspect involves the complex organization of the so called “transition phase”, including logistics management issues related to the demolition and construction phase (Spallarossa 2016).

3.2 BIM for the Galliera

In this scenario, seeing the complexity of the work, the hospital administration with the technical office identify as necessary tool, to ensure a high quality of the whole process, Building Information Modelling. Both Lot 1 and Lot 2 will be developed through BIM methodology, though with different procedures connected to the several aims, and some specific bid documents are regulating the use of building information modelling.

This tender notice represents an important step for the Italian public projects because it is the first example, in the healthcare field, with BIM requests in all the realization processes. Only during the last periods, in fact, the Italian government has updated the Contracts Code (D.Lgs. n.50/2016) to the BIM methods and tools, entering into force from 1st January 2019 for contract above 100 million of euros, with a progressive adjustment until contract below 1 million of euros from the 1st January 2025.

How the project managers indicate in the bid documents, for the realization of the structure, a BIM model will be developed to cover all phases of the processes and making a common data environment to manage all documents project. This digital model will be based on the “**object oriented**” technology, defining all features of objects and creating a database where each user can get geometrical and technical information. This database will be keep update during all design phase ad it will be of support during the construction and maintenance phase (E. O. Galliera 2017). Summarizing several objectives and BIM use that Galliera Hospital wants to obtain, the most important are:

- Visual checking, useful from the beginning of the design phase to control Lot 1 and Lot 2 interaction and manage the whole process;
- Improve quality and information exchange between different involved parties (designers, contractors, stakeholders, etc.);
- Generate easily and immediately comprehensible documentation, useful to introduce and show design also to non-technical operators;

- Improve design delivery and reduce design errors with clash detection and model checking;
- Ensure correspondence of the result to initial goals;
- Generate the hospital database, in order to follow the lifecycle of the building, useful for maintenance management and Facility Management;
- Use the model for hygienic-functional planning.

The contract also establishes to the Employer (EOOG) the availability of the continuously update model in order to allow to look at it, compare the different design and construction options and to check that the project is developed according to the LOD (Level of Detail) requirements (Spallarossa 2016).

Between the most important technical documentation has an important spot the **BIM PLAN**, which represents an operative instrument to develop by the contractor. It is a method of operation with the aim to involve all professionals in the BIM process, developing the necessary technology to have a faster and more economical project. On the BIM PLAN the several roles and tasks will be specify, explaining also software used during each phase and how set the information management. The **EIR** (Employer Information Requirements) regulates the Common Data Environment, identifying minimum features of this system to allow an easy access for the all documents and to simplify the sharing method. Another one is the **AIR** (Asset Information Requirements) that, with the previous one, defines the minimum informative contents for the new hospital. The AIR provides guidelines for technical data request on design and construction phase, particularly it is divided in:

- AIR-00 that contain the guide to use the document;
- AIR-01 that define documents to give to the Customer during the whole lifecycle of the building;
- AIR-02 that split the asset (New Hospital) in 40 category and for each one it describes the level of detail for geometrical elements (LOD) and for information (LOI – Level of Information);
- AIR-03 that specifies the level of accuracy related to the UNICLASS 2015.

Other specific indication about these documents will be given in the chapter 4.1.2.

4 Data Management

As noted in chapter 2, one of the most powerful aspects of the BIM methodology is the integration and storage of all data useful during the whole construction process. It is essential today, for design and construction companies, to outline a great data management setting because it represents the key to make difference from a mediocre project to an excellent project. Advantages for owners and managers are apparent: spaces can be managed; the model can help quickly populate FM database saving labor time, money and assets can be effectively managed; the model can be used for building performance simulations and commissioning; operation simulation tools may be able to use data from the model; and as-built information can later be used for retrofits (Eastman 2008). In fact, the positive aspects are not limited at the design and construction phase but at the entire lifecycle of the building, indeed data stored in the model can be also used for maintenance and repair, management of energy, visualization and modification of general components. All this shows how important data management is to have one coordinated system with integrated BIM and FM.

In the following chapters the role of data information in BIM process will be debated, focusing on standardization both in general and for our study case. Then the attention will be moved on the main and necessary information for a hospital, identifying people targets and what kind of information they need. The last part concerns the methodology developed for the study case, aiming to create a method to make easier analyzing and sharing information, connected with the visualization application of the Model in the chapter 5. Revit software, Dynamo and Dynamo Player have been used to develop this part.

4.1 Data Standardisation

In the construction environment, as other production industries, standards and guidelines represent an essential aspect to make projects safer and reduce failure. They represent important principles to make the world equivalent across culture, time and geography ensuring process and wellness in our society.

Looking at BIM as an end to end delivery methodology strong similarities with the ICT domain are evident and it is possible learn from their experience. Standards related to ICT are usually divided into three parts (Figure 4): **Concepts**, **Data Model** and **Process**. Common concepts and classifications are necessary for everyone to speak the same language. Neutral formats for data models required for systems and players to exchange information clearly. Finally, a uniform process for information delivery and a common working methodology is necessary. Around these 3 divisions we can arrange BIM standardization themes (Poljanšek 2017).

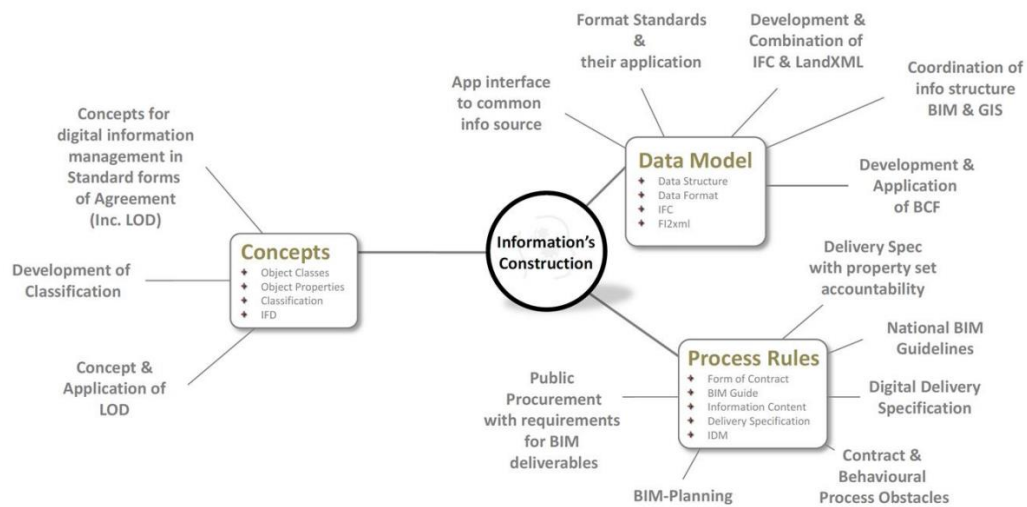


Figure 4: BIM standardization platform (Poljanšek 2017).

Following this classification several organizations (not only related with the relevant ISO and CEN technical committees) have developed BIM standardization. A lot of liaisons are present in the structural field but only those among the most important are presented here:

- CEN TC442 BIM: Standardization in the field of structured semantic life-cycle information for the built environment;
- CEN TC287 GIS: standardization in the field of digital geographic information for Europe;
- ISO/TC211 GIS: Standardization in the field of digital geographic information;
- ISO/TC59/SC13 BIM: Organization of information about construction works;
- ISO/TC184/SC4 STEP: Standards that describe and manage industrial product data throughout the life of the product,
- Open Geospatial Consortium: International not for profit organization committed to making quality open standards for the global geospatial community;
- buildingSMART: International organization which aims to improve the exchange of information between software applications used in the construction industry;
- EU BIM Task Group: Its aim is to bring together national efforts into a common and aligned European approach to develop a world-class digital construction sector.

4.1.1 IFC and COBie

It is important to focus the attention on an important aspect where BIM standardization covers an essential role, it is the exchange of information between software used in the construction industry. In this context Industries Foundation Classes (IFC) represents one of the most ambitious programmers for standardization, which is under ongoing development since its first

release in 1997. IFC is an open source data model standard developed by the buildingSMART organization with the purpose to allow the sharing of information throughout the lifecycle of any built environment asset, between all the participants, regardless of which software application they are using (buildingSMART 2018). Another important standardization is the Construction Operation Building information exchange (COBie) that has been introduced as a reference for the transfer of 3D BIM information into spreadsheet format. The National BIM Library (NBS) in the UK, one of the most famous organization for the standardization of BIM object, recommends COBie as standard, saying that advantage of COBie standardization is that information collected at various milestone dates can be compared in various ways. However sometimes this format turns out to be extremely complicated and could contain too much information to be included in a model (Ugliotti 2017).

4.1.2 Space Management and Data Population

Crating a BIM model and storing all the information related with the building objects, it is not enough to have an efficient and functionally system for facility management activities. It is essential setting an intelligent building management process to organize information, ensuring Space management is one of this information, often underestimated during the several phases, present in the construction and design process. An accurate knowledge of the space, with the combination of the spatial and geometric information, provided by the 3D model with management data, gives the possibility to facility and building manager to improve processes (Ugliotti 2017). To know the size, location, use and contents are the keys to build a great workflow with an efficient organization; indeed they represent basis for other consecutive aspects like equipment, lighting fixtures, plants position, etc. Related to this context, it is important to note that some BIM software gives the possibility to make easier it. Into Revit software, for example, **Schedules** allow a great optimization of the several processes presented in the design and maintenance phase ensuring a complete control on the digital model and its data. They represent another model view organized in tabular form that contain data from the whole model, extracting all the element's properties and updating them each time there is a change in the model.

On the other hand, an efficient population of data objects and consequently an efficient use of FM database represent another important aspect in the information management. A correct data management, in fact, provide to make easier editing of all essential documents for the owners and other parts involves in the process. While in the past the population of the data was only a final step, now it become an integrate phase already from the beginning of the design process. This make possible to create records which contain a virtual description of the building and associated data: 3D model, 2D drawings, assets (e.g., furniture, furnishings, and equipment), maintenance manuals, specifications, and other information that is usually requested at closeout (Kensek 2015).

On these concepts are based the future application in chapter 4.2. Alternative methods will be proposed for space management data export and data object population.

4.1.3 Data requirement system for a new project

How specified in the chapter 3.2, in a generic BIM project the base for data regulation is represented by two documents incorporate into the tender documentation: Employer information Requirements (ERI) and Asset information Requirements (AIR). Unfortunately, in this moment the procedure of selection for the new project is still open, so majority part of

documentation is reserved. For these reasons some of treated arguments in the chapter will be based on the standard structure of tender documentation for new BIM project

The AIR defines the information required to develop a correct Asset Information Model (AIM), which is necessary to provide graphical and non-graphical data and information as well as documents and metadata. In this project, where there is a strong attention to the asset management activities, AIR should be based on the Organizational Information Requirement (OIR) document, which establishes and categories the needs of data and information for different organization involved.

The EIR, based on the Plain Language Questions (PLQ), is directly connected with the AIR, as matter of fact it defines what the employer wants to get out of the models both for the development of the project and for the operation of the completed built asset. Plus, EIR sets out the requirements for the Project Information Model (PIM), which is developed progressively during the design and construction phase and that represents an important tool to plan the engineering and design information process.

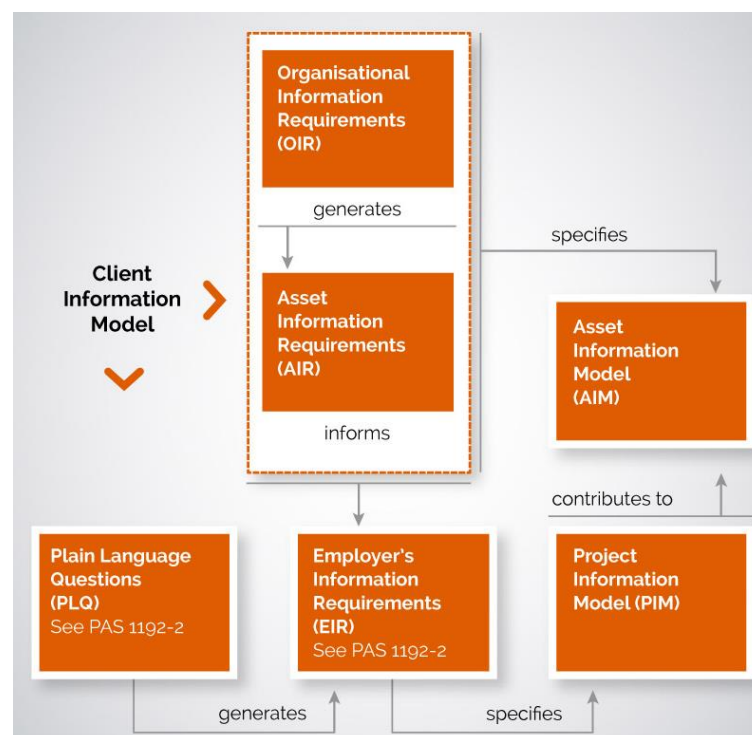


Figure 5: Relationship between elements of information management (BSI 2014)

Based on this methodology the customer, **Ente Ospedaliero Ospedali Galliera (EOOG)**, has decided to divide the whole project in 40 categories based on construction technology. This partition permits to organize and simplify the workflow, to better understand customer requests. For each category is specified the value of LOD (Level of Detail) and LOI (Level of Information) in all construction phases (Definitive, Executive and Realization phase), plus one more level related to the phase of delivery and handover of the project. Of course, the requirement, related to LOD and LOI, increases during the project development, and it is evident from this estimate the importance of the management aspect for the customer. The

higher level, in fact, is set up for the last phase when the model will be transmitted and use for management and maintenance of the building by the technical office of the hospital.

The AIR is divided in three parts and also the AIR-02 contains indication for IFC and COBie classifications of data stored into the model. In each category is indicated the IFC classification to use for the relative elements and if the category requires the COBie format, which fulfil an important aspect into the whole project. From the beginning, in fact, COBie knowledges were represented as fundamental request for the information management in the tender evaluation because the building model will be based on this type of data for Facility Management aims.

In according with the objectives of EIR documents, it is also interesting the choice to use pre-set parameters for the model development (using a file of shared parameters), integrating them in the entire process. For EOOG to set up additional parameters into the digital model is a fundamental aspect to have a correct parameters development along the whole process, ensuring that the digital model will follow the guidelines necessary for future management procedure.

4.2 Work Methodology

How said before, Schedules are a great tool in Revit, which makes easier accessing and sharing information especially for people who are not expert in using the software. However, for complex project where with a large amount of data, exporting and individuating a specific information range it is not always an easy process. It is possible find on the market several plug-in or applications which allow to make easier the export/import process from/into the model, but usually they don't give freedom of changes, keeping model and schedules settings. As a matter of fact, schedules are separated for different category (Rooms, Walls, Windows, etc.), this make more complicate to combine information or to select only a particular parameter for the different categories without to analyze a big number of data. In the first two applications of the next chapters, a process will be developed to allow inexperienced and professional people to select and combine different parameters from different category. While the third application will analyze a method for family data population. To develop all applications, Revit, Dynamo and Dynamo player have been used.

4.2.1 Identification of target and data

The first step, to develop a digital methodology for data management, was to understand which kind data analyze, starting from the target of people that will use them. In our case the building is represented by a hospital, for this we have defined three different groups of users:

- *Visitors and Patients*: they represent the majority part of people present in a hospital and at the same time who needs more information to understand the complexity of the structure. For this group is essential ensuring an ease access for all building information and a clear language of communication. Principally the kind of data to use for visitor and patience are spatial information related with the hospital distribution of sectors, department, etc. Knowing the distributive aspects of the building is an important point for general user to understand how to move inside the building and to know all the safety aspects necessary for a public building.
- *Medical staff*: also for this group spatial information are essential, however unlike the previous one they are experienced about the building distribution. For this the main information essential for medical staff are focused on the room destination and room

services. They need to know all the features of each room, starting from the functionality of the local until the medical equipment parameters.

- *Technical staff*: this group represent all people responsible for the building maintenance and management. It is easy to understand that in this case the information range cannot be limited to certain aspects, but it covers the whole building project. Also, the kind of information language is different from the previous two groups, requiring more technical specification. Data, essential for this group, are of space management, object information, structural properties, plant disposition, and parameters of this type.

4.3 Dynamo for Data Management

Dynamo is a visual programming tool which can be downloaded as free and it gives users the possibility to script using several textual programming languages through visually script code blocks. This allows both non-programmers and programmers people to manipulate element, data and geometry in Revit by coupling script nodes which are previously programmed to do a specific task. The language used to write the codes is Python and each node has a simple or a complex script depending on the operation that it must conduct. Most nodes in dynamo are composed of five parts, how see in Figure 6:

1. Name of the node;
2. The main body of the Node. Right-clicking here presents options at the level of the whole Node;
3. The in and out ports. The receptors for Wires that supply the input data to the Node as well as the results of the Node's action;
4. Lacing Icon. Indicates the Lacing option specified for matching list inputs;
5. Default Value. Right-click on an input Port, some Nodes have default values that can be used or not used.

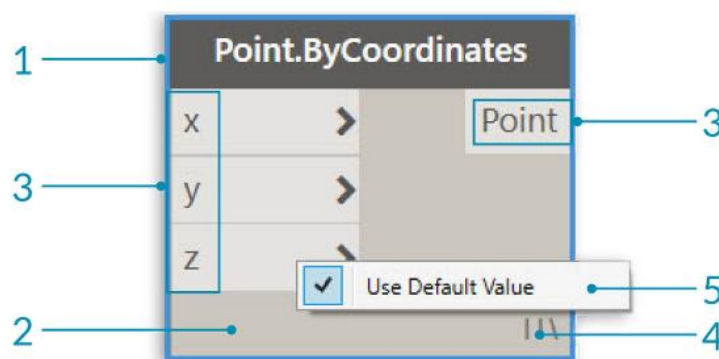


Figure 6: Anatomy of a Node (Dynamo)

4.3.1 Dynamo for Export data – Application 1

In the first application Dynamo has been used to export information and parameters related to space management application. Dynamo don't put limit on what you can develop and there are infinity ways to interact with Revit, for this part a script (and so a process) will be developed to extract Revit parameters from different Revit Categories combining them in the same file. The main idea is to allow non-expert people on Revit software to have the ability to select a project section and to choose the information that they want to analyze, and at the same time specialist can use this method to classify and to organize information making easier sharing process. The script explained below permits to extract parameters, chosen by the user, from all Revit Category presented into the model selecting first the floor to analyze.

To understand easier the process realization, the whole script has been divided in different groups with a different task for each group.

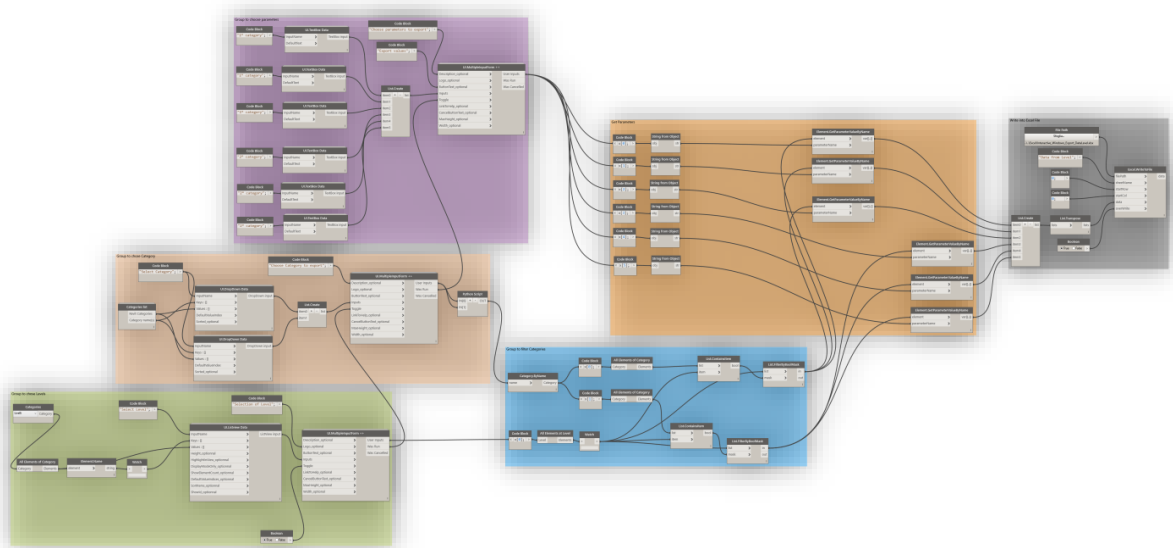


Figure 7: Dynamo script - Application 1

- **Green group:** it allows to produce an interactive window with the list of Revit model levels, giving the ability to select the floor to analyze.

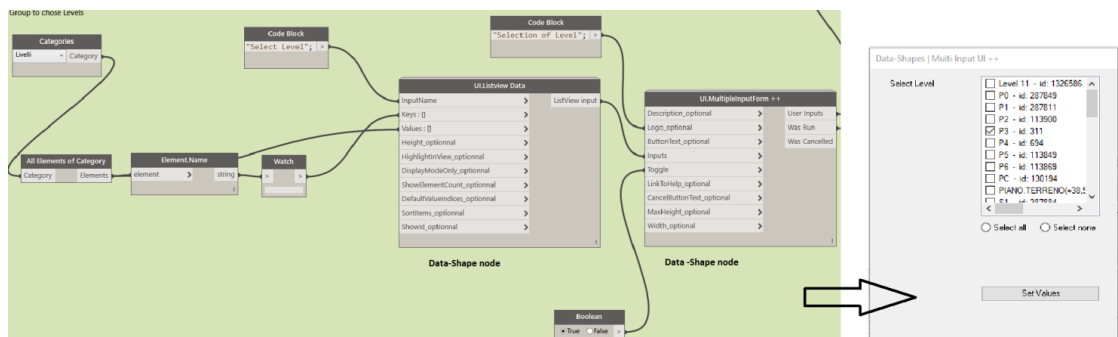


Figure 8: Dynamo script – Application 1.1

The core of this group is represented by the node *UI.MultipleInputForm++*, which allows to create an interactive widows starting from several inputs. The aim is to create a tool

to choose the level of my building for this we have used *UI.ListView Data* node, which allow creating a window with a list of all input elements. In our case the inputs are all the levels present into the Revit model. Important is the “Boolean” node to ensure that the node works. To allow that the process continues running we have connected the output “was run” with the following group.

- **Pink group:** it produces an interactive window with two boxes, where user can choose the Revit Categories from the model library.

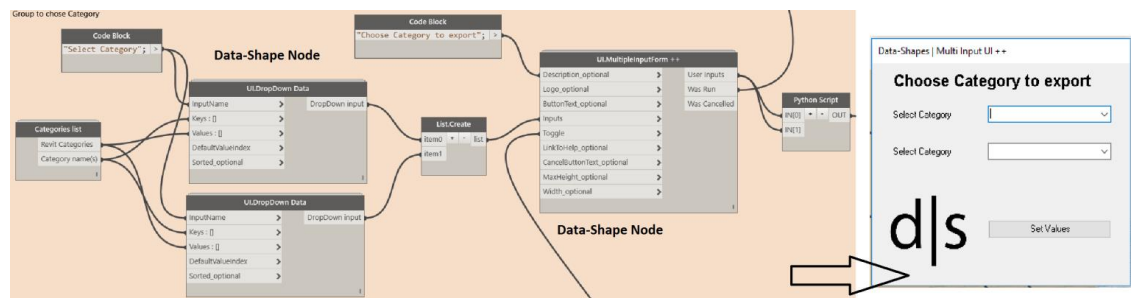


Figure 9: Dynamo script – Application 1.2

This group is based on the same node of the previous one. With the interactive window the user can select the Revit Category to analyze from the previous selected level. The node *UI.DropDown Data* gives the ability to create a task bar to write a string. For this example, only two task bars are set up but there is not limits of number to insert. With more selection input is necessary use the node *List.Create* to keep split the values. In this case a Python script node has been necessary to translate the input categories from a string to a Revit element. Here the python script:

```

import clr
clr.AddReference("RevitAPI")
import Autodesk
from Autodesk.Revit.DB import *

Cat = UnwrapElement(IN[0])
names = []
for c in cat:
    names.append( c.Name )
OUT = names
    
```

To allow that the process continues running we have connected the output “was run” with the following group.

- **Violet group:** it produces the last interactive window with two text boxes, allowing to insert the name of parameters that the user want to export from previous categories.

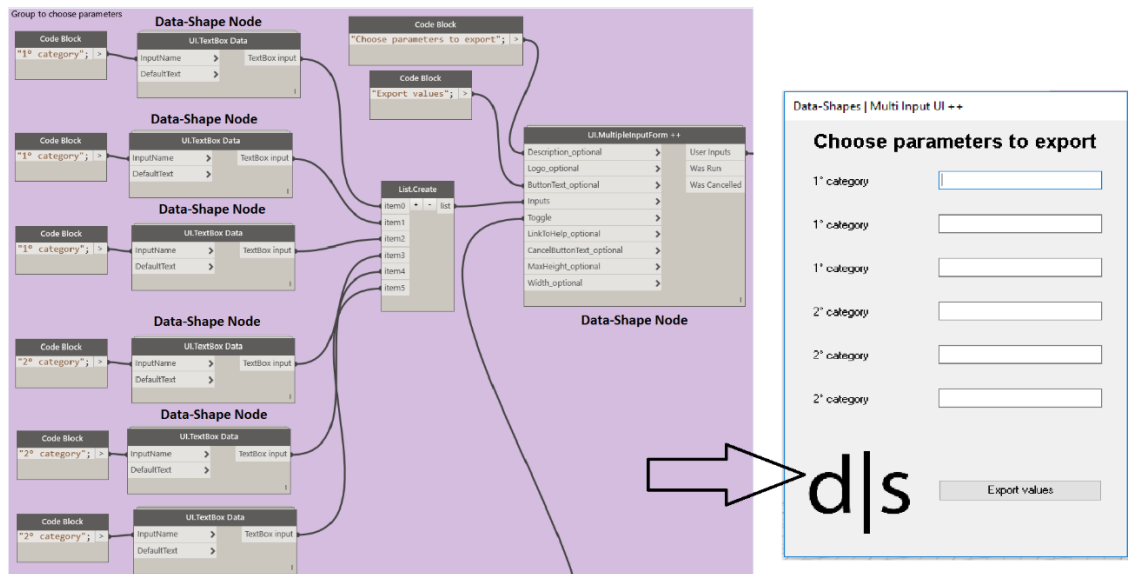


Figure 10: Dynamo script – Application 1.3

Also, in this case we have used the node *UI.MultipleInputForm++*, to create the last interactive window, where the user can write the name of the parameters to export. To do this we have used the node *UI.TextBox Data*, which creates a box where typing the input parameter. In this example only 3 parameters for each chosen category in the previous group has been set up, but there is not limits of number to insert.

- **Blue group:** it allows to filter the chosen categories in the pink group from all the elements contained in the level selected in the green group.

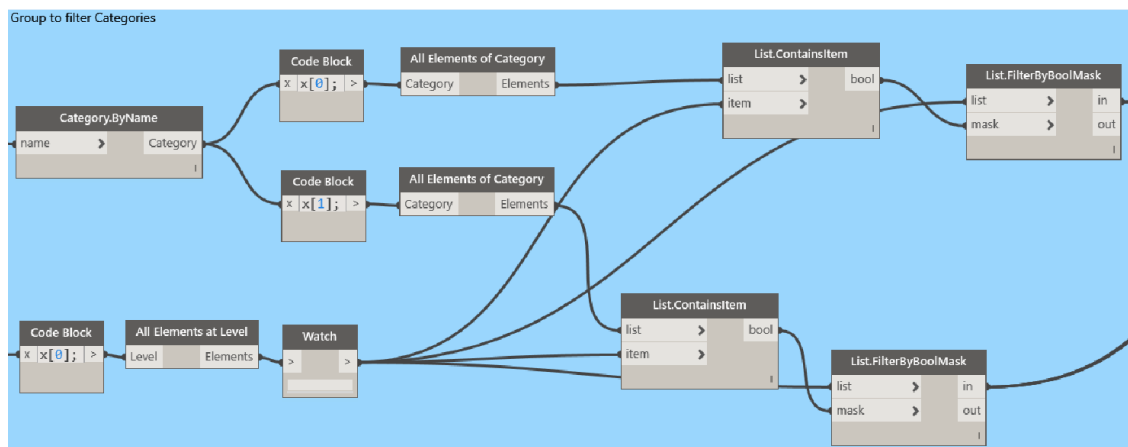


Figure 11: Dynamo script - Application 1.4

In the green group, in fact, the level of our interest has been selected but from it only the information about the categories selected in the pink group must be extracted that are present in the current floor. This is possible using the nodes *List.ContainsItem* and *List.FilterByBoolMask*. The first one needs to compare all elements of the level with all elements of categories, producing a boolean value (true/false) for the elements that have been matched and for those have not. After this the second node read the boolean value

for each element making a list with the “true” elements if you select the “in” output or with the “false” element if you selected the “out” output.

- **Orange group:** it takes the names of values from the violet group allowing to get current parameters from the related categories.

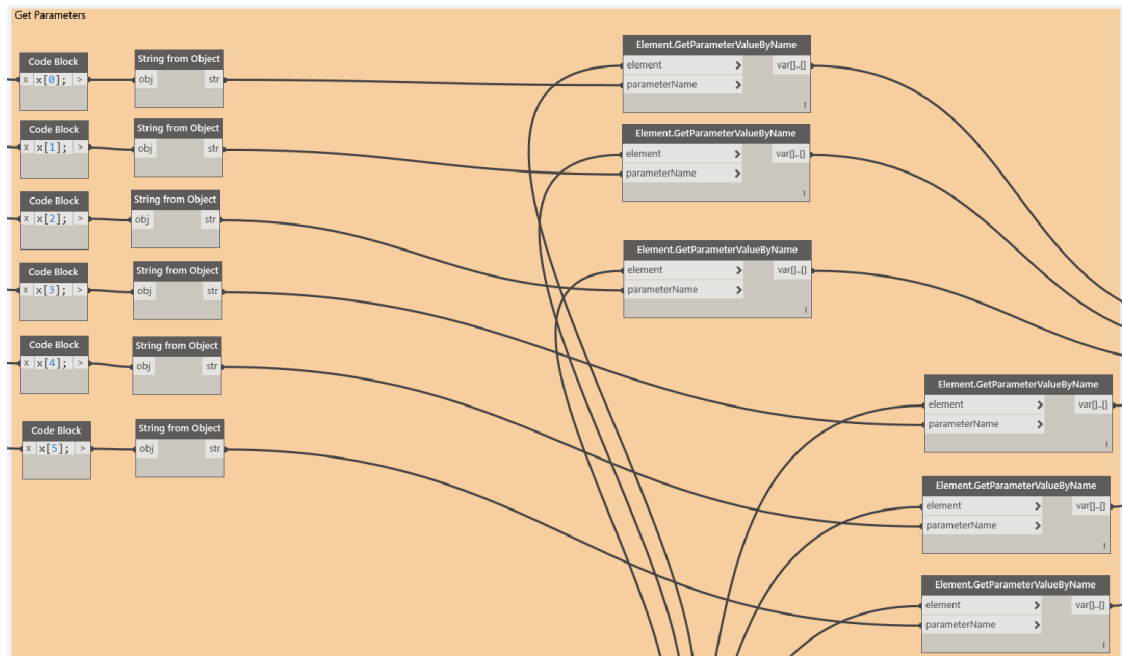


Figure 12: Dynamo script - Application 1.5

This Group connects the previous groups with the name of the six parameters from the violet group. The first part provide to split, from the violet group, the name of each parameter through a simple *code block* selection node and then they have been converted in strings. After this the node *Element.GetParameterValueByName* has been used to extract the value of the parameter from the Revit model and to write it on an excel file.

- **Grey group:** it allows to write previous obtained values into an Excel file, connecting the file path.

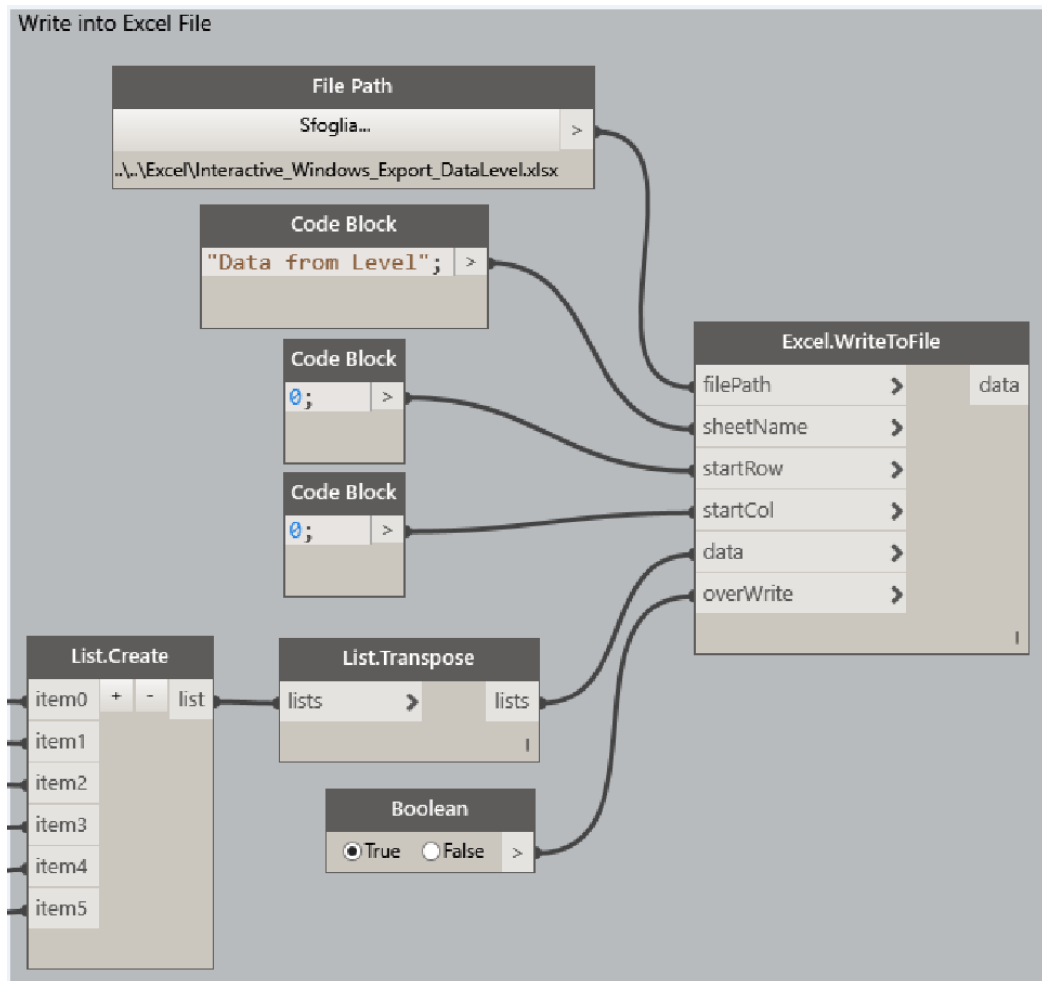


Figure 13: Dynamo script – Application 1.6

First, a list has been create using the node *List.Create* to put together the previous values and then using the node *List.Transpose* rows and colons have been inverted. After this through the node *Excel.WriteToFile* has been used to export values, with all inputs showed in the picture above.

To improve and make easier the exportation process, the script finds its perfect application combined with the Dynamo Players plug-in which provides to execute Dynamo scripts without to use Dynamo software. Opening Dynamo Player, user can select the script directory, to view current script status and to launch the script how showed in the in Figure 14.

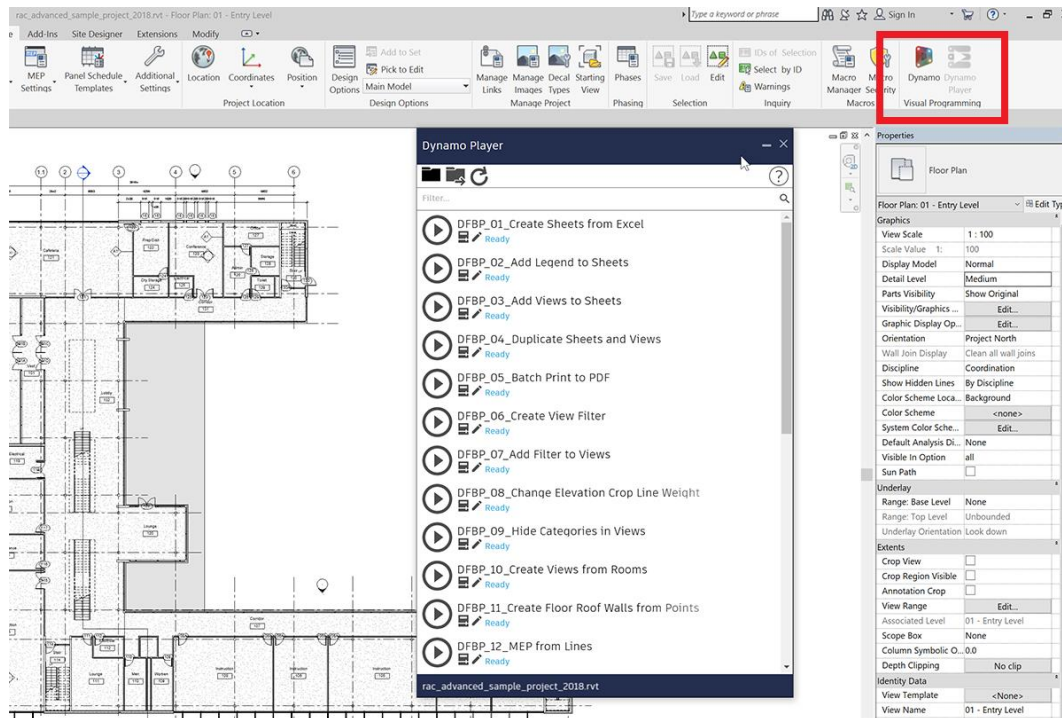


Figure 14: Dynamo Player (Dynamo)

Using this process, a general user needs only to open the Revit file and to launch the script. This can avoid interacting with Revit Schedules setting each parameter and exporting different files for each category. Of course, this cannot substitute the benefit to use Schedules, but in huge project it can make faster selection of a range of information. For this example, the script is set up to analyze only two Categories and three parameters for each category (six in total), but it can be enhancing without objective limits adding other nodes of the same type and ensuring a solid structure to the script. In fact, another positive aspect it also represented from the ability to improve and personalize the process at will changing the Dynamo script, which is not always possible with external plug-in and tools.

4.3.2 Dynamo for Export data – Application 2

This second application is always related with the exporting process from Revit, but while the previous one analyzed the whole Level this one will focus the attention only on a single Room. The aim of this part is to develop a script and so a process to extract all elements contained in a single room, chosen from the user, combining all information in a single file. The main purpose of this application is to give an easy way for non-expert user to access at data located into a room, making easy the process of maintenance and control, and for expert and specialist people to have an easy method to select and classify information.

In the following picture (Figure 15) represents the whole structure of the script composed from the different nodes and groups.

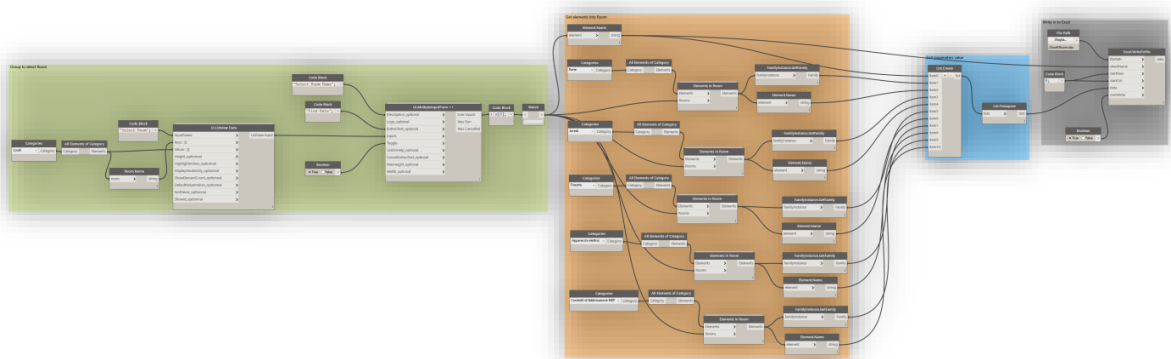


Figure 15: Dynamo script – Application 2

- **Green group:** it allows producing an interactive window with a list of all rooms present into the whole Revit model, giving the ability to select one of these.

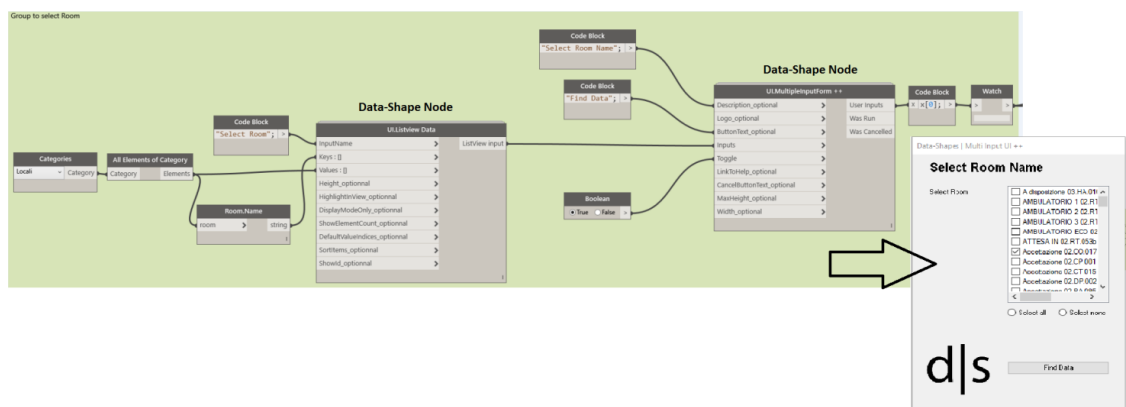


Figure 16: Dynamo script – Application 2.1

This Group is basically based on the same methodology used for the Green group in the Appendix 1. The main difference is one input value, where this time they are set up on the name of the rooms that there are in the whole project.

- **Orange group:** it takes the names of the room selected in the previous group, verifying which elements of the pre-set categories are present into the room.

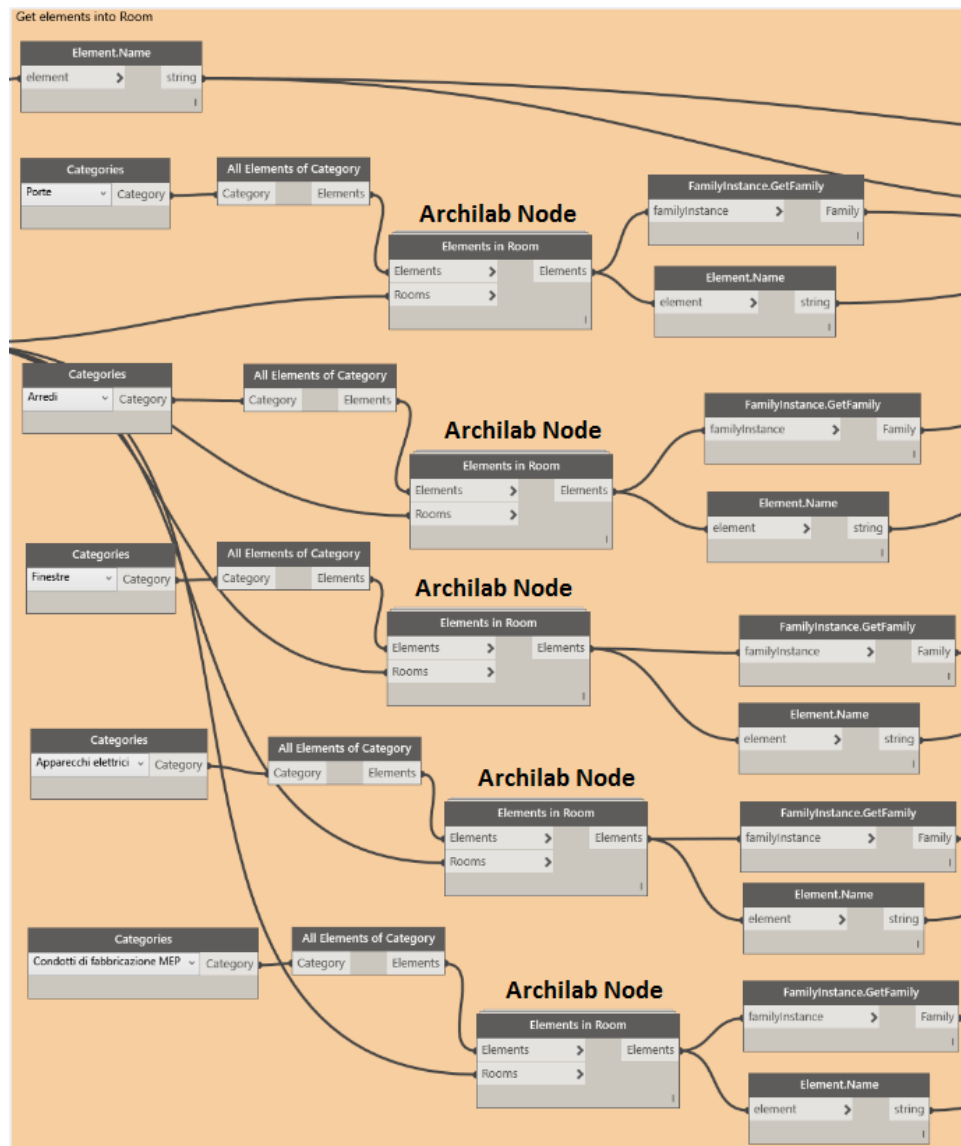


Figure 17: Dynamo script – Application 2.2

To identify the elements that are present inside the chosen room, several families have been pre-set. In this way will check only between these families using the nodes *Category* and *All Element of category*. In this example only 5 categories of element have been analysed but there is not limit of input number. Then the *Elements in Room* node has been used to check if there is that kind of element inside the room. In the positive case we get the Family Name and the Family Type of the element.

- **Blue and Grey Groups:** the first one is needed to sort the several values in a correct list to be overwrite in an excel file. The other one to write values on an Excel. Both are based on the same methodology of the previous application.

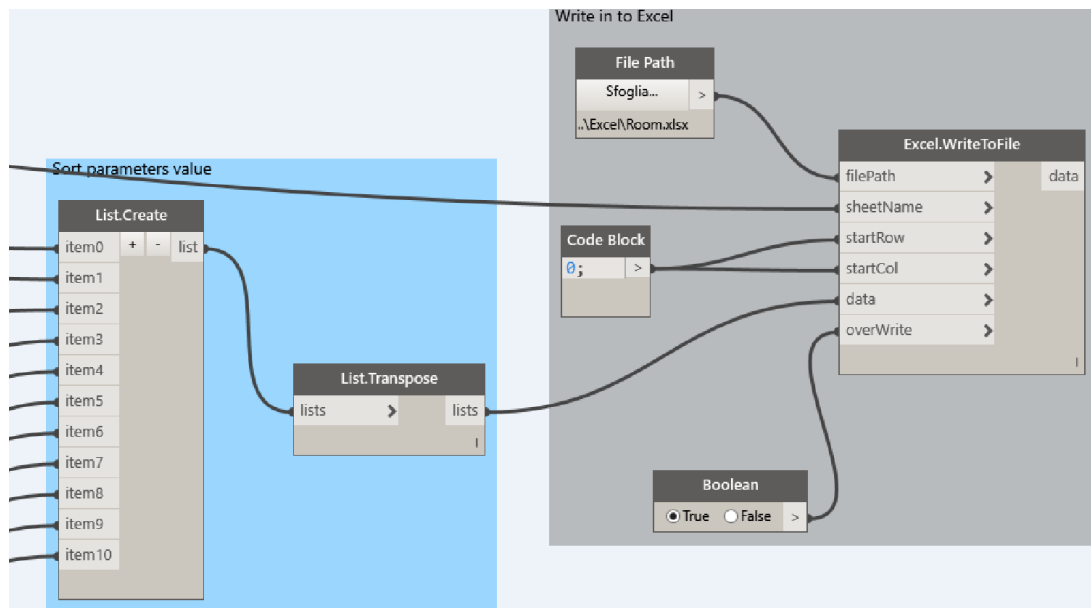


Figure 18: Dynamo script – Application 2.3

As the previous one this script finds the perfect application integrated with the Dynamo Player use. How mentioned above with this tool the users have the ability to see the family located into a room obtaining the Family Name and the Family Type of each element. In this example the Category of Families are pre-set, but there isn't limitation on the number of Families to pre-set. For this example, five Families are set to check if they are located into the room: Doors, Windows, Furniture, Electrical Equipment and MEP Fabrication Containment. Also, here the script can be improved and adapted for each necessity, allowing professional and non-professional to export values for a specific room, useful both to combine data in a simple way and for maintenance aims.

4.3.3 Dynamo for Export/Import data – Application 3

The last application aspires to find an easy method for data family population. Differently from the previous two, this is divided in two Dynamo scripts because one has the aim to export values from Revit to Excel while the second one is necessary to re-import values from Excel to Revit. Using this process, the user can select one or more elements from the Revit model exporting their data in Excel where everyone can set values for several family-parameters and then import these changes into the Model.

The following picture (Figure 19) shows the whole first script (export data) composed from the different nodes and groups.

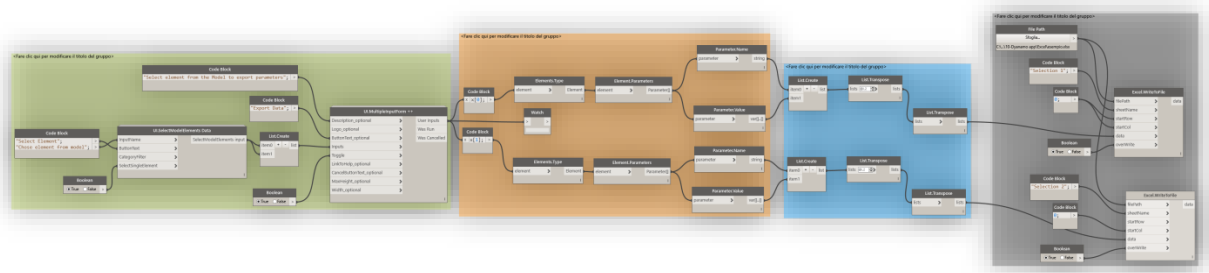


Figure 19: Dynamo script – Application 3.1

- **Green group:** it allows producing an interactive window which gives the ability to select one or more element from the whole Model selecting them from a generic Revit view.

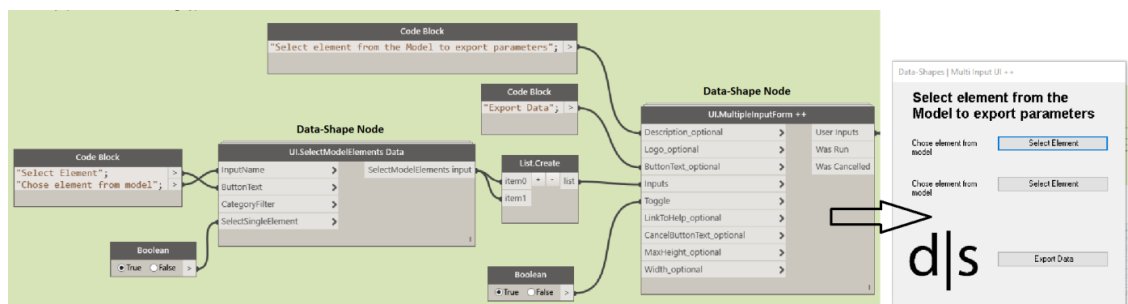


Figure 20: Dynamo script – Application 3.1.1

For this Group the same methodology has been used for the Green group in the previous two applications. The main difference is on input values, *UI.SelectModelElements Data* node has been used to create an interactive window, to allow choosing by a simple click elements to analyse from a model view. Creating a list, with the *List.Create* node, after the previous node it is possible set the number of element to select, only two for this example.

- **Orange group:** it takes input values from the previous group obtaining parameters values and parameter name starting from the type of family selected in the Model.

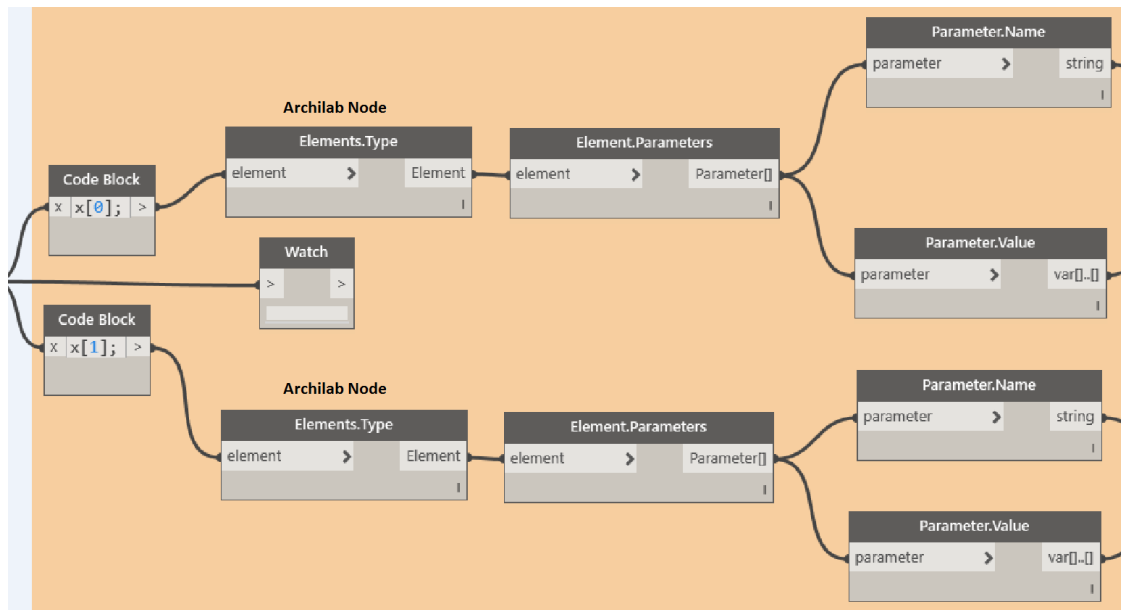


Figure 21: Dynamo script – Application 3.1.2

This group starts taking the outputs from the previous one, dividing them through a *code block* node. Then *Element.Type* node has been used to obtain the element Family Type and from this, and using the *Element.Parameters* node, it is possible to get the all Type parameters of current element. Then it has been necessary split the parameter between the parameter name and the parameter value, to can collect the two values in a different excel cell.

- **Blue and Grey groups:** They have the same function of the blue and grey groups in the previous application.

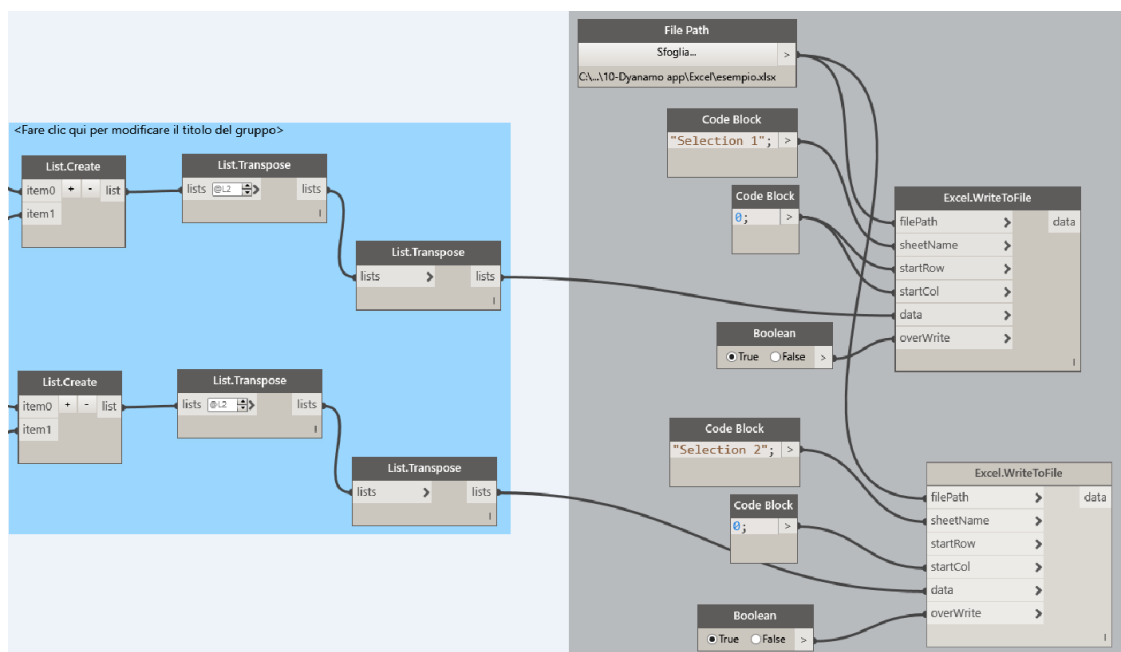


Figure 22: Dynamo script – Application 3.1.3

In this case are present two *Excel.WriteToFile* node (both are connected at the same excel file) one for each selection in the first group.

The following picture (Figure 23) shows the whole first script (import data) composed from the different nodes and groups.

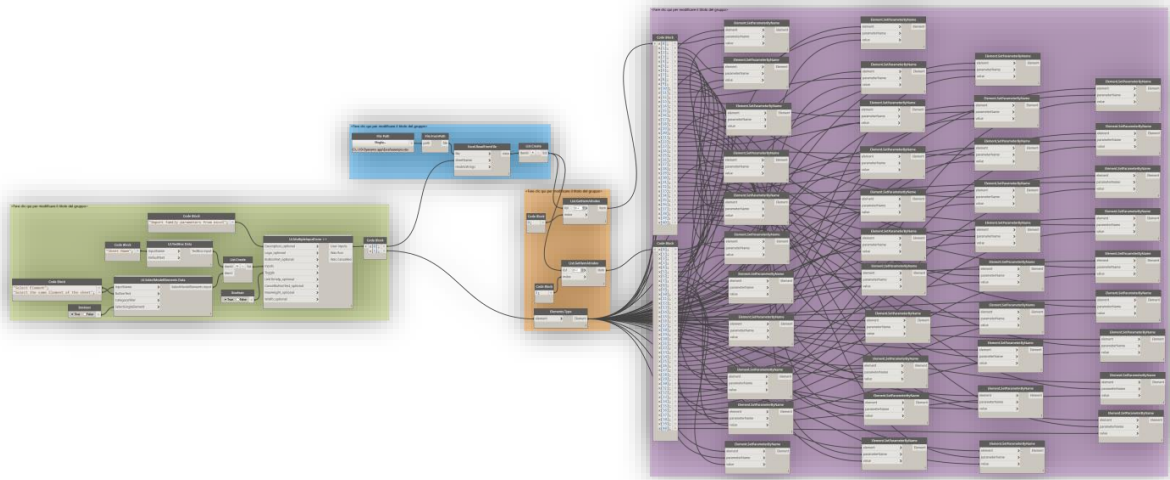


Figure 23: Dynamo script – Application 3.2

- **Green group:** it allows to produce an interactive window with the aim to select, from the model, the element (only one) where import the new values and it is also necessary to type the name of the excel sheet.

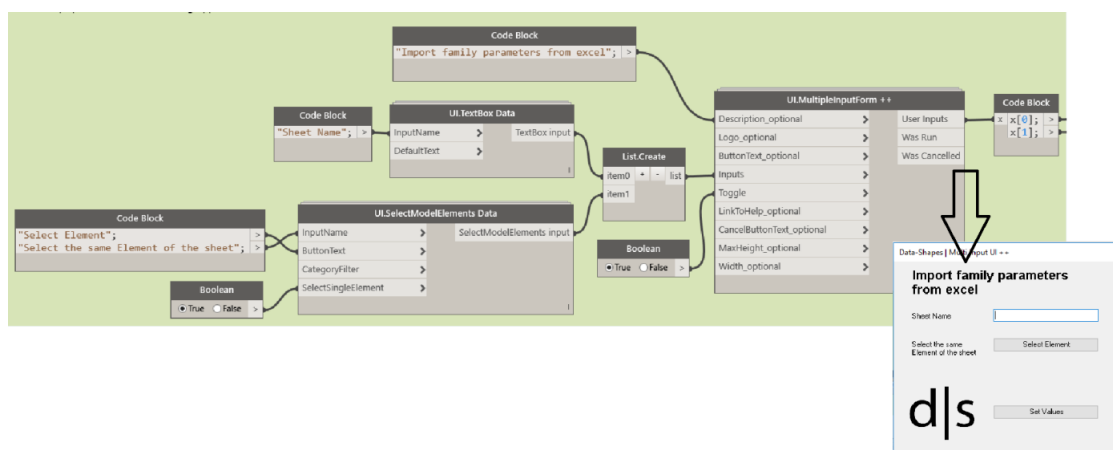


Figure 24: Dynamo script – Application 3.2.1

The main difference of this group is always on input values, where this time the nodes *UI.Textbox Data* and *UI.SelectModelElements Data* have been used. These two nodes insert in the interactive window a typing box (the first one) to write the name of the sheet to analyse, and a selection button to have the ability to click from the digital model the element connected with the sheet values.

- **Blue group:** it allows Dynamo to read data and create a list from an excel sheet (previously typed) connecting the file path.

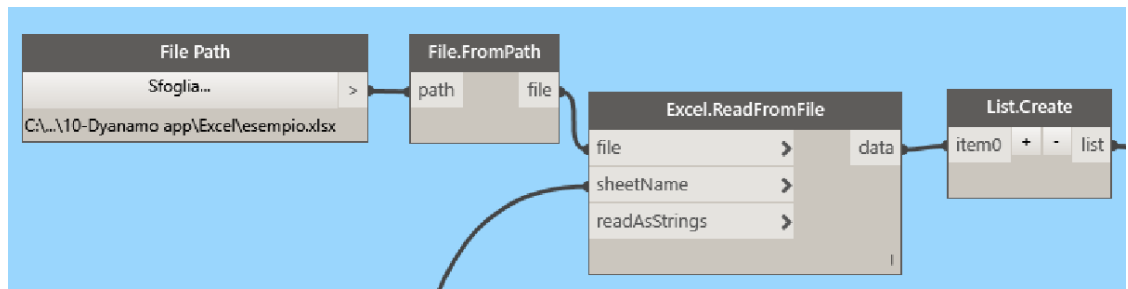


Figure 25: Dynamo script – Application 3.2.2

The script can read the values of the previously selected sheet, connecting at the *Excel.ReadFromFile* node the file path of excel file and the creating a list.

- **Orange group:** it takes input values from the previous group and splitting the list of parameters obtained from the excel file in two different list: one for parameters name and one for parameters values.

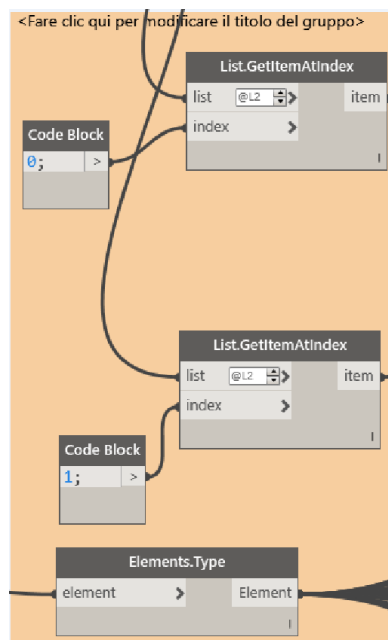


Figure 26: Dynamo script – Appllication 3.2.3

The aim of this group is to organize the data to be imported into the Revit model. In the Orange group of the previous application, in fact, family values have been split in name parameter and value parameter. For this the *code block* node with the index 0 has been used to obtain the name and with the index 1 to obtain the value. Then the *List.GetItemAtIndex* node gives the ability to create two separated list, one for name and one for value. In Addition to use the *Elements.Type* node has been necessary to connect element selected in the first group, having the type of element necessary for the next group.

- **Violet group:** it gets values from the previous two lists sorting each parameter for the respective index and using each couple of same indexes to set the new values into the Model.

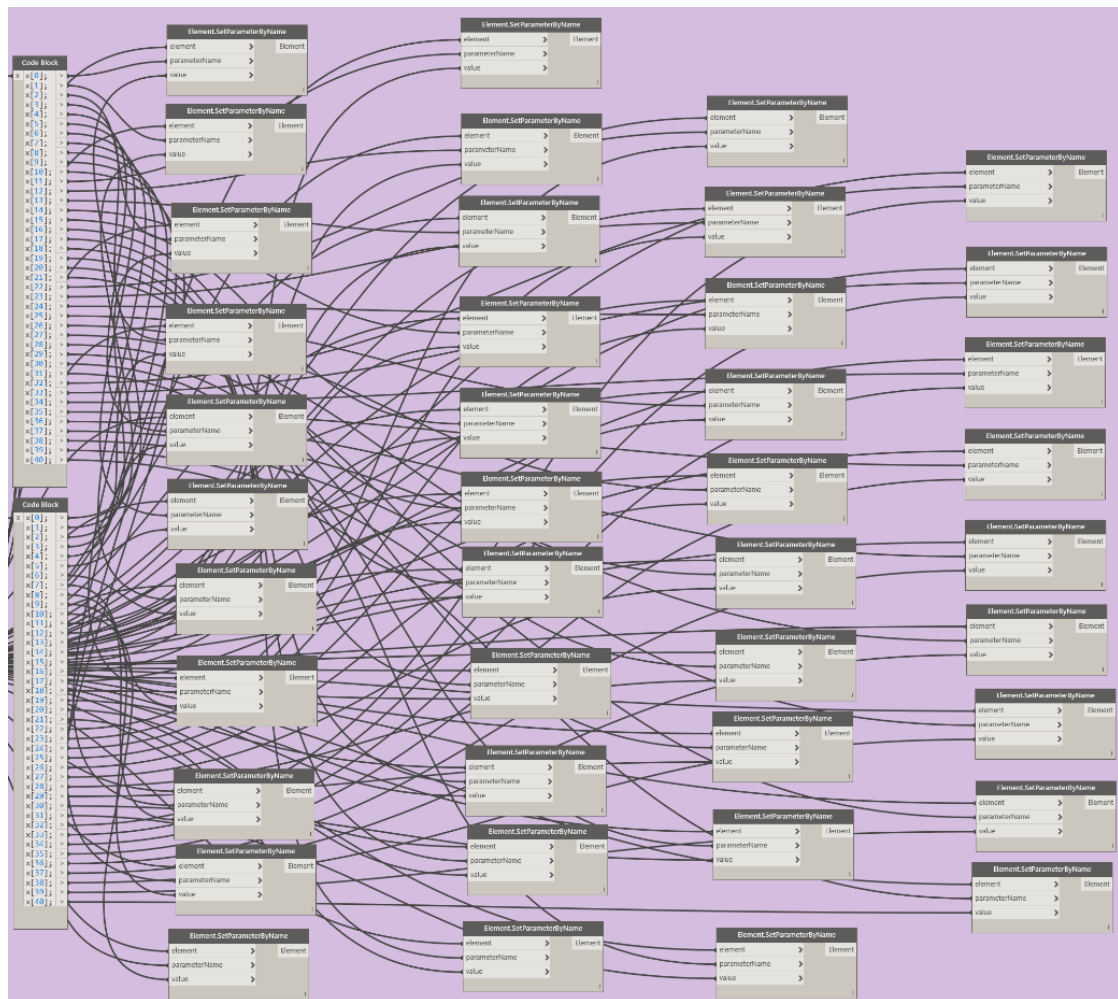


Figure 27: Dynamo script – Application 3.2.4

This group is born from the inability to find an iterative method able to analyse and combine multiple parameters to set the new values into the model. The aim of this group, in fact, is to insert in the Revit model the new values that come from the excel sheet. To do this *Element.SetParameterByName* node has been used, which has the limit to set up only one parameter for node. For this reason, with the *code block* nodes on the left each value and name have been separated from their list. Proceeding in this way to the index parameter name at “x[0]” in the first *code block* node, it will correspond to the index parameter value at “x[0]” in the second *code block* node. Using together these values, plus the element type from the previous group, the new values have been imported in Revit model. The number of 40 items, in both the code block, has been chosen to be sure that all the parameters of a family can be analysed. Some family, in fact, can have less of 40 parameters and in this case, Dynamo will show the error message for the node without an input, but the remaining parameters will be set up correctly.

Also, for this case using Dynamo Player, the process will result faster and easier to use. How explained these two scripts should help the process of data population for elements located into the model, making easier the compiling process. With the script 3.1 each user can export Type Parameters (and no Instance Parameters) from the model, just selecting one or more element from a Revit view, into an excel file with separated sheets for each selection where one column is for the name of the parameter and one column is for the value. Of course, the process can be

adapted also for different kind of parameters but for simplicity we have chosen Type Parameters because Instance Parameters, for example, are not modifiable in majority part since depending by the geometry and features of the Family.

For this example the number of element to select is limited at two, but it can be augmented using the same script structure. After to have filled the excel file, inserting new values, they can be re-imported into the Model through the second script (script 3.2). For this part, in order that the process can works properly, is important to type exactly the name of the excel sheet and to select the related element with the values of the sheet. In the Figure 28 is showed an example of this process applying it for a floor family, the image on the right is the empty Identity Data list before to apply the scripts and the image on the left is the list after the process.

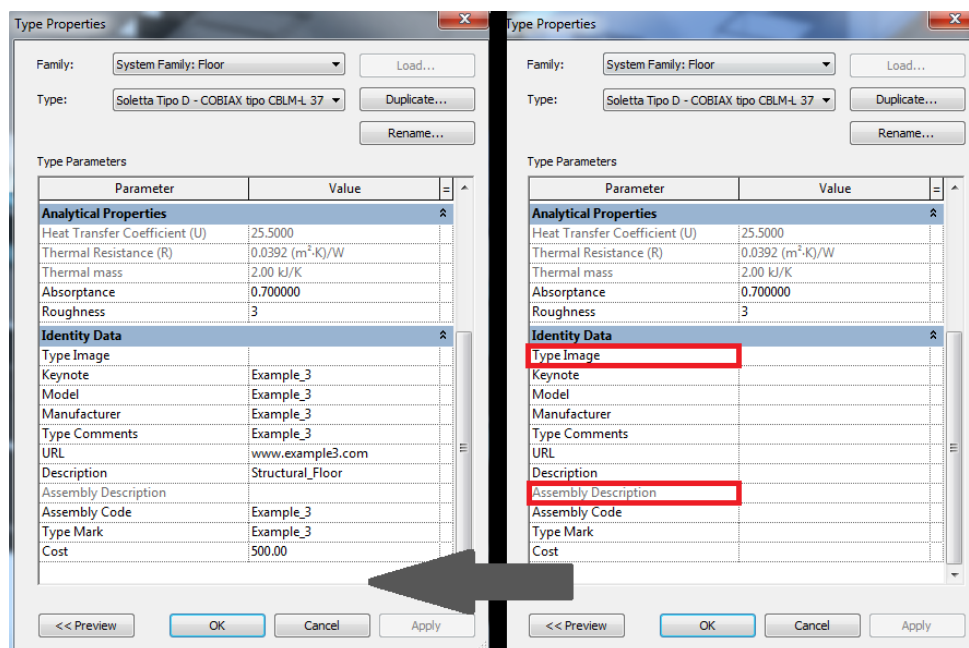


Figure 28: Application 3 – Example.

The process is limited by the nature of some parameters. Grey parameters showed in the Figure 28 are values which cannot be altered using excel because they are defined within the family. Other as “Type Image” parameter need different information format, in this case is a graphic format, that importation process don’t allow to set.

It is clear that Revit gives the ability to type values directly from the model, but this means to type for, each different element, each value and of course the user must have knowledges of the software. On the other hand, using the workflow presented in this chapter it is possible to export the parameters of all families at one time and for project with hundreds of different elements, it is a great advantage, plus everyone can fulfil the task simply using excel.

4.4 Results

With this chapter the attention has been focused on the importance of data in a correct Facility Management context, related on our study case and with our target of users. These work processes, in fact, aspire to improve the sharing information for all figures involved in the process, professional and non-professional. The main idea is to use these methods as base for

future developments, showing the possibilities to set up an interactive data management method using Dynamo software. Starting from this workflow the possible applications are unlimited, especially for complex structure like a hospital where is present a huge number of data, so not restricting the Dynamo use at space management application. In the last years, in fact, it has become an essential tool for innumerable aspects with two essential strong points: to allow interacting in a complete way with Revit and the freedom of application limited only by user knowledges.

5 Virtual Reality

How said in the previous pages the advantages of BIM methodology are many, from the easier storage of all data into the model until the easy way to cover each phase of the construction process. BIM represent also a revolution in this sector for the new way to design and think the project, based on a 3D environment. This aspect obviously helps professional people to have a clearer idea about the project in each construction phase, but having software based on 3D data a lot of new technologies approach to AEC industry. In this context new visualization methods become main characters on the utilizing process of 3D data, and one of the method is based on the Virtual Reality (VR) technology.

In the next pages, the concept of VR will be introduced, seeing how it can be useful for construction sector and presenting an example based on our study case.

5.1 VR Technologies in AEC industry

The VR (Virtual Reality) technology is a computer simulation to create an environment like the real life, where the user can interact with this environment through some input like a movement of a hand or a vocal command; this means that the computer can recognize the input and to create an answer in real time. This technology is based on a real-time rendering which (differently of the traditional 3D rendering where 3D-data are converted to pixels in a 2D image) is a process that running continuously. To give to the user the impression to travel around in a virtual word, computer must generate a sufficient number of frames per second to prevent the user from experiencing motion sickness (Hettinger 1992).

The result of the real-time rendering can be displayed in a wide variety of ways. The most common solution for a great virtual experience is wearing a head-mounted display (HMD), which allows the user to involve more sense (touch, eyesight, hearing, etc.) with different levels of immersion. Nowadays many different devices are present on the market; two of the most common devices for an immersive VR experience are Oculus Rift and HTC Vive (Figure 29).



Figure 29: Oculus Rift headset (left) HTC Vive (Right).

Oculus Rift use two 1200x1080 pixel Pentile OLED displays per eye, for a total resolution of 2160x1200 pixels and 110° field of view. The final version includes two Sensor used to track the position of the user's head as well as other VR devices and two Oculus Touch controllers. These are devices specially designed and engineered to transplant hands movement into the virtual environment. With these controllers, the external infrared tracking sensors allows the

user to manipulate objects into the virtual environment with extraordinary precision (Oculus 2018).

HTC Vive uses a dual AMOLED 3.6" diagonal screen, with a resolution of 1080 x 1200 pixels per eye (2160 x 1200 pixels combined) and a field of view of 110°. The system uses "room scale" tracking technology, allowing user to create a play area where he can physically walk around and interact with objects. Also this device has two controllers and two Vive Base Stations that through timed infrared pulses allow to create a 360 degree virtual space up to 15x15 foot radius.

A lot of other devices are used for virtual experience; many do not need to use a computer but only a common smartphone as screen to elaborate the virtual experience. This makes simpler and cheaper interaction with these types of devices and for this also more popular. Other interesting alternatives, but less ordinary, are the CAVE and the Powerwall. The first uses LCD shutter glasses to have a stereoscopic vision with a method that allows to render an image for each eye sequentially and this switching process is synchronized with the graphics card and performed at a high frequency; the second one is a semi-immersive and it is basically a small cinema screen allowing more people to view the VR simulation simultaneously with shutter glasses or polarized glasses (Johansson 2016).

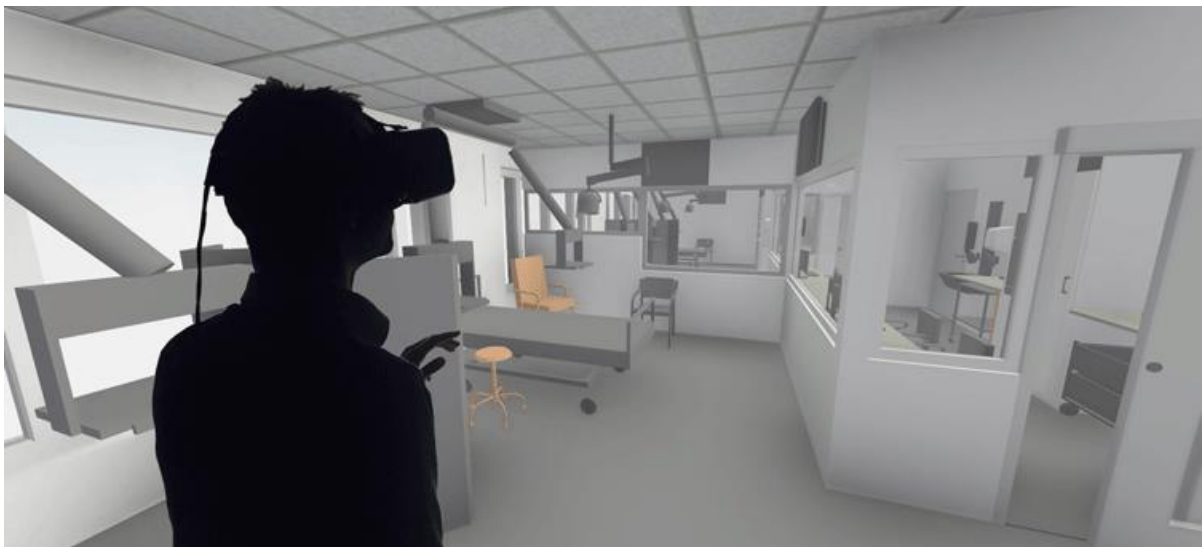


Figure 30: Example of VR use with BIM model

Behind this word it is possible finding a lot of software used to create three-dimensional simulations. The most used and powerful in this moment are software based on game engine environment, generally used to develop video-games and between these, big exponents (related with AEC applications) are: **Unity, Unreal, Stingray and Cryengine**. They allow users to have freedom of development the virtual environment, limited only by software knowledge and programming language. Not being designed for AEC applications, sometime to use game-engine software can represent not an easy method. The main problems, in fact, are correlated with the interoperability aspects between these kinds of software with common design software, causing information losses during transfer process.

To simplify the process of virtual environment creation, using previous software, several tools are now available into the market, which can be used as plug-in for design software. The advantage in this case is represented by the ease method to get the virtual simulation of your project, launching it directly from your project browser and without interaction with other complex software. The most common software, of this type, are **Enscape, IrisVR, Autodesk LIVE and Fuzor**. On the other hand, they present several limitations to personalize and develop your virtual environment respect the previous game engine software. These tools don't allow, in fact, interacting with the creation process limiting the result on a tree-dimensional environment where the user can only navigate.

In the following pages a workflow example will be showed of how develop a virtual environment using Unity software, providing a basis method useful to understand which potentialities it has related with AEC industry. Two application will be developed, an immersive and not immersive Virtual Reality application, using only some parts of the Revit model of the Galliera Hospital.

5.2 Work Methodology

The successive chapters will analyze the whole process to obtain two virtual reality application from our BIM project. One is a not-immersive application, creating a **Design Interface** with the aim to promote the project between professional and non-professional people. This solution has been developed to satisfy the request, by the technical office for the project, to involve more people on the future realization and to better explain features and shapes of project. The second one is an immersive application where the previous application is re-elaborated to be used through Oculus Rift headset, adding a new part of **Participatory Project Design**. This solution was designed for the healthcare industry, where the medical staff can be involved into the design process of hospital spaces. Both solutions are developed through the game engine software Unity, starting from the geometry of Revit model.

5.2.1 From Revit to Unity

The first step to develop a virtual environment starting from a BIM model, and in this case from a Revit model, is to select the portion of model that we want to analyze simplifying the model. All unnecessary information can be hided, as annotation elements or linked file, keeping the essential building geometry to create the environment. To start the process of transfer in fact, Revit file has been exported as ".fbx format file", which ensure to have a good level of interoperability between modelling software, keeping all necessary 3D data. However this format do not permit to preserve other information as facility management data stored into the model.

Intermediate software must be used between Revit and Unity to allow that all elements in the model can have the correct standardization to ensure a good quality of the final product. Unity, in fact, cannot read accurately standards of Revit material library, making problematic a direct transfer. For all this reason the 3D Studio Max software has been used (Figure 31).

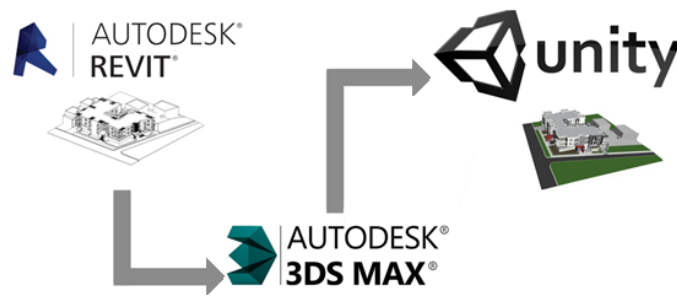


Figure 31: Revit – Unity interoperability process.

The process starts exporting the Revit model in “.fbx” format, from the Revit menu through *Revit menu > Export > FBX*. It is important that the model is on a 3D view. After opening 3DS Max is necessary go on the general menu section to import the previous extracted file clicking on *Import > Link FBX*. At this point, it is important set up parameters using the window that automatically appear. In this case the *Combine by Revit Material* option has been chosen, including in one group all elements based on the same material (Figure 32).

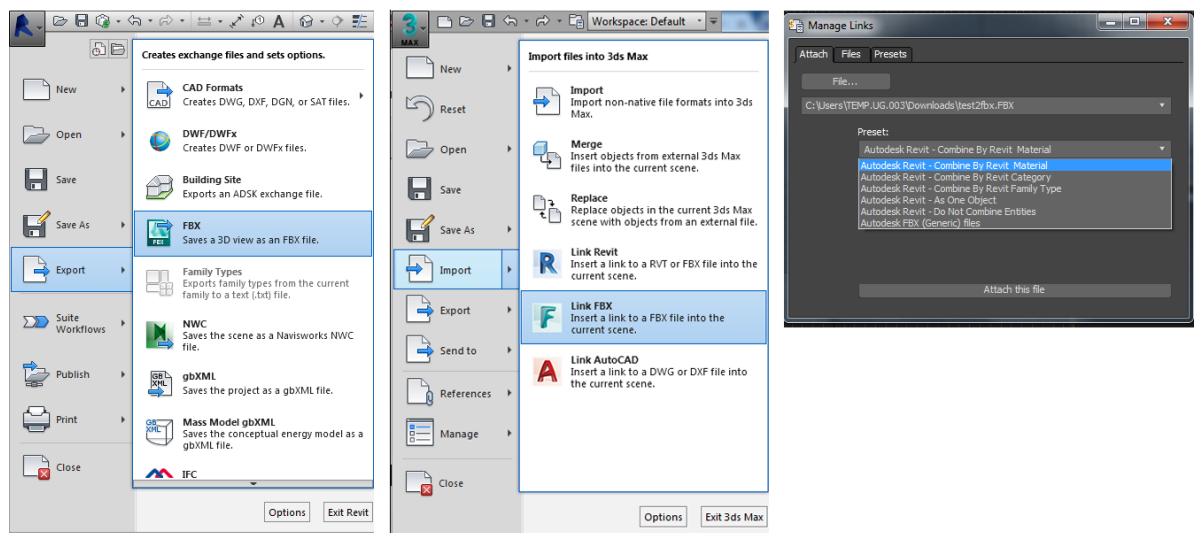


Figure 32: Export/Import “.fbx” format file.

Sometimes could be necessary import the model in 3DS Max using other options depending on the final aims, through the previous method it is also possible to combine elements for family type or also as one object. The choice is correlate with the complexity of the model, for small part could be useful also the standard import method of 3DS Max by clicking on general menu section and then in *Import > Import*. This method allows keeping the same division of elements present in the source Revit file.

After this procedure, 3DS Max allows to manage all elements and materials present into the model converting them to standard materials. Now it is possible to choose different way to continue, depending on the outcome and quality to reach. Using the software, in fact, you can change the render quality of each material and change each propriety. In this example the model

materials have been simply converted selecting *Rendering>Scene converter* and then exported it from the 3DS Max menu, how showed in Figure 33. Anyway when a “.fbx file” is imported in 3DS Max, the software provide to group elements using its standardization, so another alternative could be simply import and export the file. The best graphic solution would be to change each material present in the model using the 3DS standard materials located in the *Rendering Tab > Material Editor*, but this method will not be debate in this document.

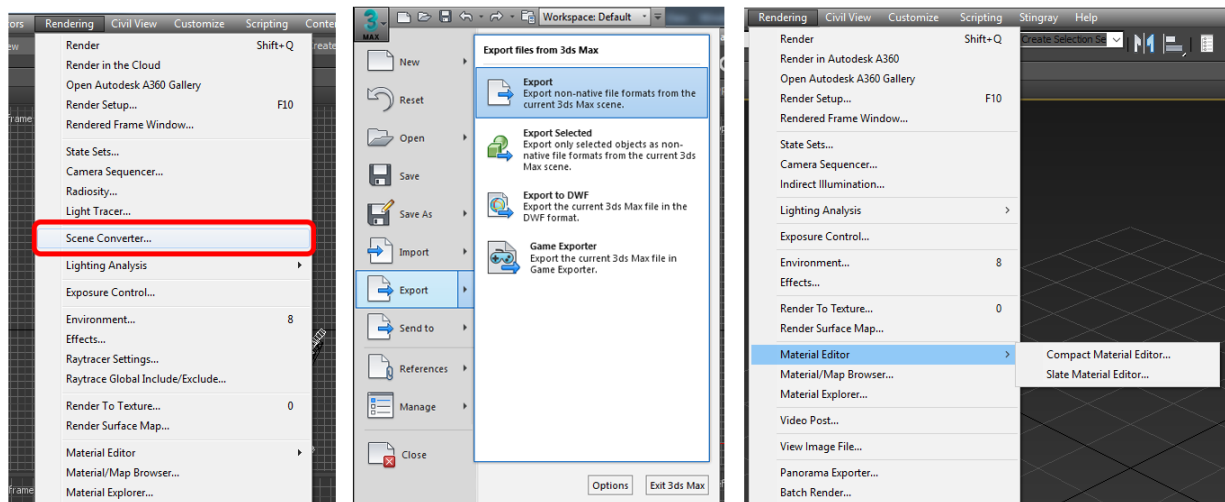


Figure 33: Export process from 3DS Max

Now the model is ready to be imported in Unity. The following chapter will analyze the software structure to better understand the procedure to create a virtual environment.

5.2.2 Unity Interface

As already mentioned Unity is a cross-platform game engine used to develop three dimensional and two-dimensional video games and simulation for computer, console and mobile device through geometrical modelling and coding. Unity is structured to facilitate the files management, the whole project, in fact, is stored in a single folder, where there are a wide variety of folders and files, which are mostly regulated in background by Unity. In particular, the Assets folder is where all files are stored, such as model, script, textures and scene files (Boeykens 2013). Before to start with the description of the whole workflow the structure of Unity must be explained, recapping the main parts (Figure 34).

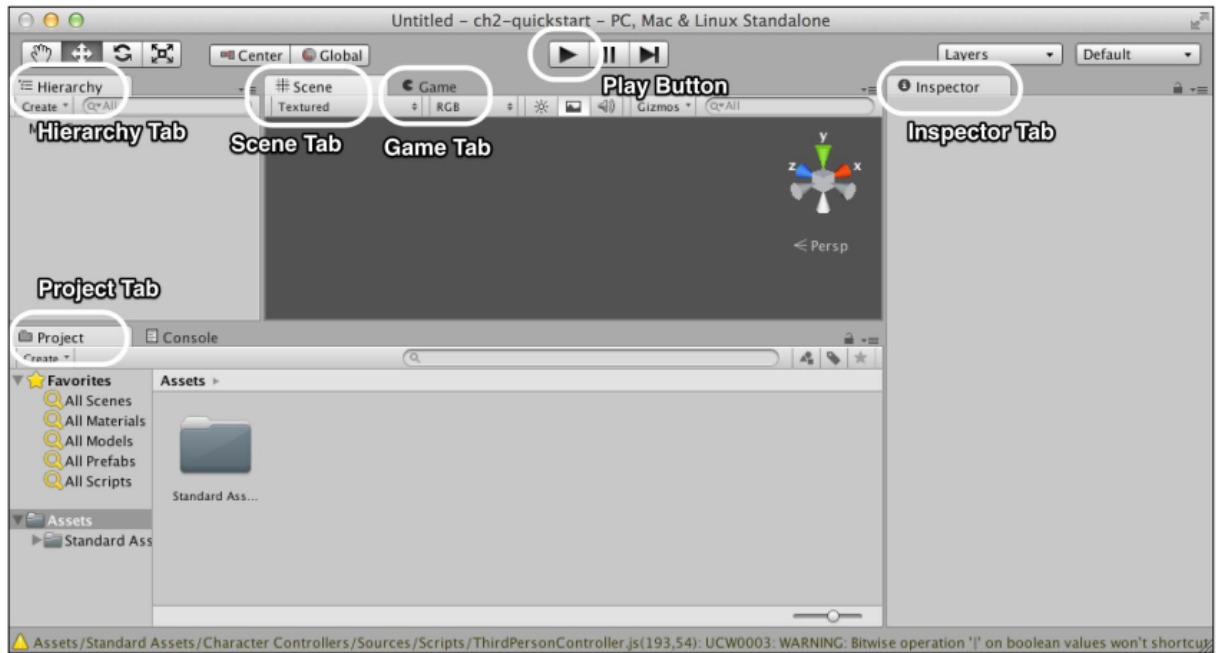


Figure 34: Unity structure (Boeykens 2013)

A Unity project is structured by **Scenes**, one or more, that will compose the whole game as is common for virtual game. Each scene is developed separately from others and this process take place within the main Unity interface, which is divided in the following parts:

- The main part is represented from the **Scene Tab** where the user can manage the geometry of the model in a 3D or 2D view. It is the work place where each element can be visualized and modified.
- The **Hierarchy Tab** displays elements of the current view that can be also used a main menu to select each element present into the scene. From this tab is possible to insert object inside the scene using the *Create* button, it allows to import essential element as 3D object, Light, Sound or UI object that are present by default in Unity.
- **Inspector Tab** represent an essential part allowing seeing and managing all the properties of any selected object.
- **Project Tab** manages all the folders present in the project and from here, the user can access at each element and to load news objects.
- **Game Tab** represent a preview of how the scene will look when you the game will be used, and this tab permit to launch a simulation pressing the **Play** button.

All these definitions will be essential for the next chapters, then it is important to know that, how other game engine software, a scripting language regulates Unity. Each user can develop the game through three different coding: C#, JavaScript and Boo.

5.3 VR Not-Immersive Application

Before to start setting the application it was essential to think how to structure and divide the application and relative scenes (Figure 35). The start idea to develop this application is to have three main scenes:

- The main scene, that will be the principle menu where the users can choose the different actions;
- A scene where the user can explore the whole 3D model of the new building.
- A scene of navigation, where the user can explore one of the main entrance of the new project.

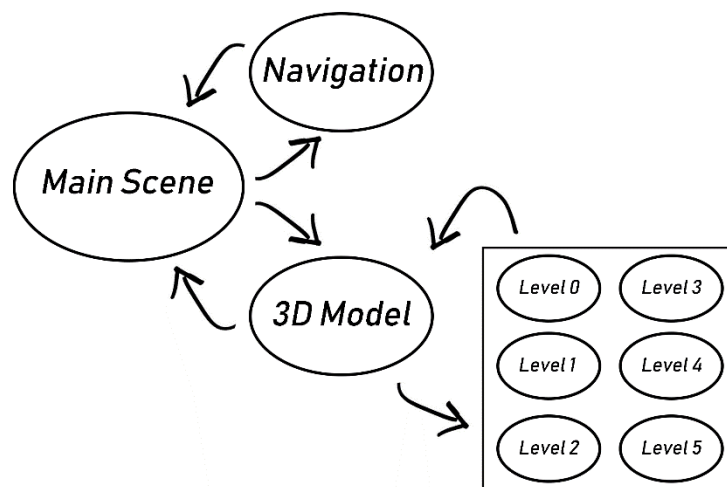


Figure 35: First Workflow Development.

Using this structure, it has been possible to understand which kind of information will be inserted in the new application and how develop the connection between each scene.

5.3.1 Main Scene Development

The main scene will represent the connection between other scenes and to obtain this, the first step is to create an interactive menu where the user, through several button, can chose different options.

By default on a new Unity scene, we find a “Main Camera” and a “Directional Light” object in the Hierarchy Tab. The first element is a static camera that will represent the user view when the game is running, while the second element is the light that we need to see inside the scene. Unity gives the ability to modify the previous elements, but in this case it is unnecessary and we can keep these default objects. Afterwards interactive buttons have to be inserted in the scene, a new object can be inserted in the Hierarchy Tab by clicking on *Create > UI¹ > Button*. This action will bring up other two elements with the button, *Canvas* and *Event System*, which will regulate the correct use of *Canvas elements*. All UI elements must be stored under the *Canvas* object to be visible on the scene. Unity will insert automatically the new *Canvas*

¹ UI Objects are dynamically elements that allow the player to interact with object or other.

element in the point with coordinates zero for X, Y and Z, this position cannot be changed with the current setting. However the element stored inside *Canvas* can be moved using Inspector Tab options and this allows to set the view (sometimes could be useful change the position of the *Main Camera*). After finding the correct position the *Button* have to be set up and in the Inspector Tab it is represented the element that make up the *Button*. In addition with the position and dimension setting a Script Image and a Script Button is located inside the object, through this default script² the button aspect may be changed loading a new image, changing normal and highlighted color and other option. The most important part is the *On Click* option in the Script Button, which allow to add a function when it is clicked.

To create a function, perfect suitable with the aim of the application, the coding language C# has been used. The buttons to insert will have the following features:

- “Navigazione” button, to be teleported in the scene of the virtual navigation;
- “Modello 3D” button, to be teleported in the scene to discover the new project;
- “Esci” button, to end the application.

The first step to do this is to create the C# code by *Asset menu > Create > C# Script*, now a new C# file will appear on the Standard Assets folder and with a double click it can be compile with MonoDevelop software or similar.

- Script to change Scene:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class ButtonManager : MonoBehaviour {

    public void NewGameBtn (string newgame)
    {
        SceneManager.LoadScene(newgame);
    }
}
```

- Script to end the application:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class Quit : MonoBehaviour {

    public void doquit()
    {
        Application.Quit();
    }
}
```

² Default scripts are provided by Unity to interact with objects without to use coding.

The previous script must be loaded in the scene to use them in the application. One way to do this is to create an empty object going in the Hierarchy Tab, *Create > Create Empty* and adding in the Inspector Tab of the element one of the script. After this the two objects will be renamed in “Button Manager” and “Quit”, respectively with one of previous script. Now the function can be loaded, dragging the object from the Hierarchy Tab into the small box of the *On Click* option of button. In this way the button will contain the script. For the “Button Manager” script, the function is assigned selecting *No Function > Button Manager > NewGameBtn* (Figure 36). In the text line that will appear the name of the scene where to be teleported has to be typed. In order to allow Unity the correct access to each scene, it is necessary clicking on *File > Build Setting* and adding in the “Scenes In Build Tab” all the scenes to manage. For the “Quit” script, it is only necessary load the object in the *On Click* option of the respectively button and click in *No Function > Quit > doquit*.

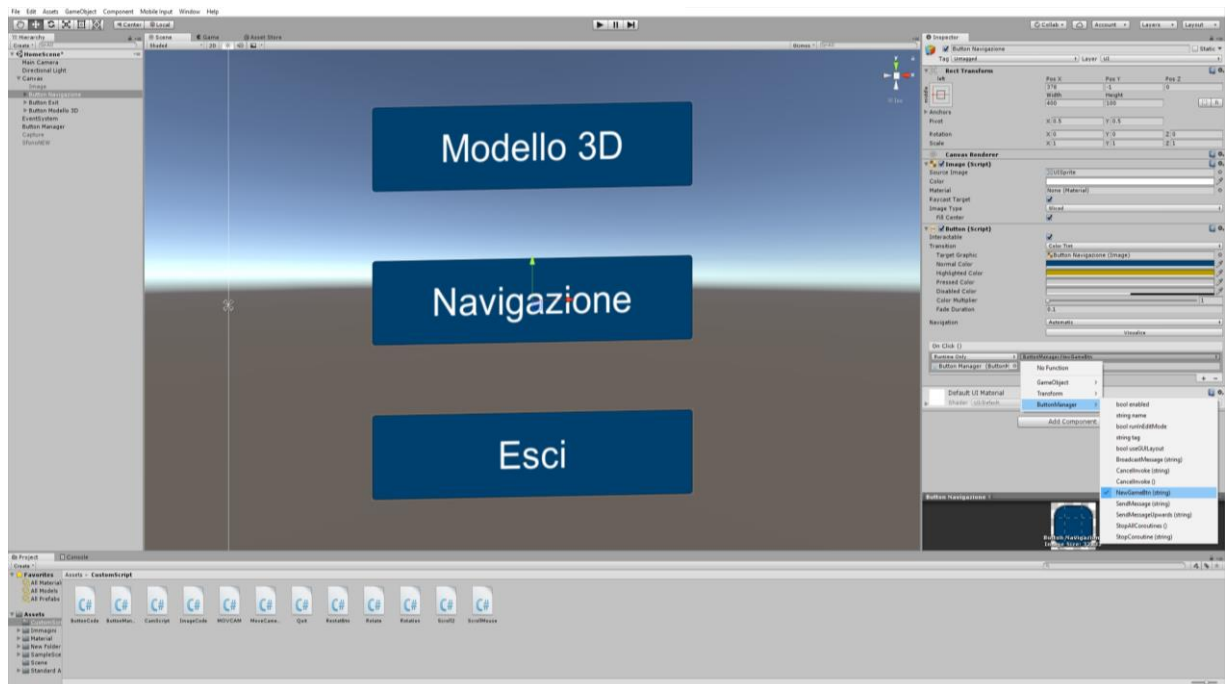


Figure 36: Unity-Button Creation.

At this point the last step to set up correctly the Main Scene is to insert a background image, this is possible adding at the Canvas element in the Hierarchy Tab by *Create > UI > Image*. To add a custom image, first the image have to be dragged from the computer into the Asset folder, then it is necessary modify its setting in *Inspector Tab > Texture Type > Sprite (2D and UI)*. Then, the custom image can be loaded going in the Inspector Tab of the *Canvas > Image > Image (Script) > Image Source*. The last passage is to set correctly position and dimension of the image, it is possible to obtain a preview of the result just opening the Game Tab and seeing how the application will appear. At this point the Main Scene is ready to be used and connected with the other scenes (Figure 37).



Figure 37: Unity-Main Scene Interface.

5.3.2 3D Model Scenes Development

This part of the application will represent the section to discover the geometry of the new project including space information from Revit model. The following part is divided in several section, the main scene contains the whole new building inserted in the existing urban context. Others sub-scenes instead contain a 3D model of each level with space department information and the number of beds in that level. It means that the following section will have seven scenes, one for the urban context and six for the various floor. Only the creation of two scenes will be debate as example. The starting point to create this part is represented by the creation of 3D models, which are suitable for Unity standards.

3D urban context has been created using *Autodesk InfraWorks* software which allows to extract easily an acceptable model from the existing urban context in a few passages. After opening InfarWorks it is necessary to access to *Create a new model through Model Builder*, this will bring up another window with a map where we can chose the area to extract with a maximum value of 200 km². Now clicking on *Create Model* the software will elaborate the area, creating your 3D (Figure 38). Now the model can be extracted opening the project just created going in *Setting and Utilities > Export 3D Model*, this process will generate an “fbx. file” including relative textures. Obviously the current urban context does not contain the new project, therefore it requires to be modify adding the new Hospital building. The model of the new project has been obtained through Revit how explained in the chapter 5.2.1, then both “fbx file” have been imported and correlated in 3DS Max creating an appropriate file to be imported in Unity.

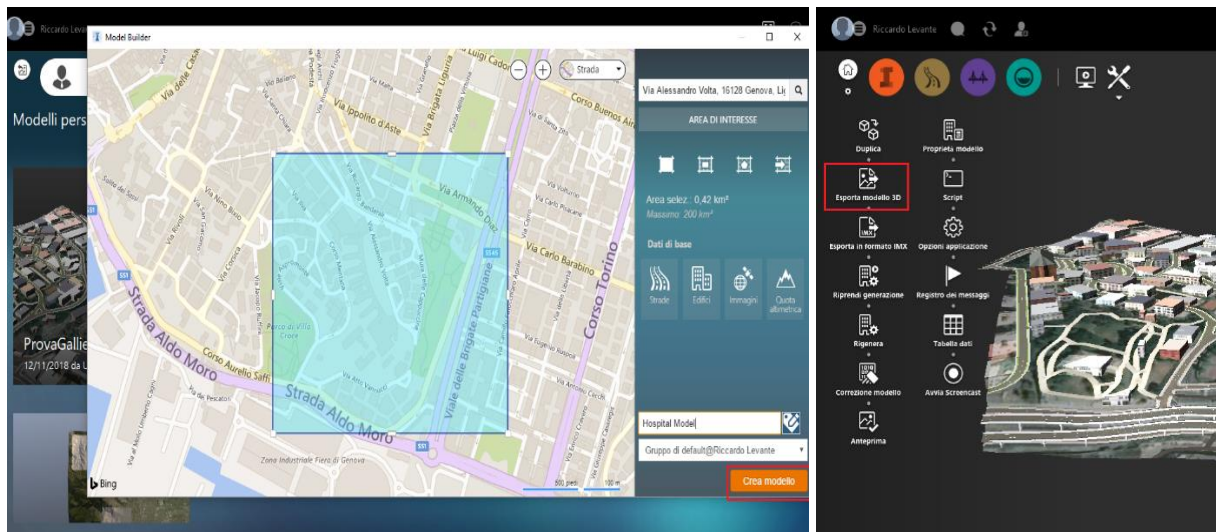


Figure 38: InfraWorks - Create Model.

How said, the next step is to import the previous “.fbx file” into the Asset folder of our project. It is essential that with the “.fbx file” also the textures folder, that 3DS Max automatically generates, is imported into the Unity project. After dragging the files from the computer into the Asset folder, the model must be moved in the Hierarchy Tab to be loaded in the current scene. Sometimes could be necessary rearrange textures and materials of the model, this is possible creating new material from the *Project Tab > Create > Material*. The new material can be modify from the Inspector Tab where is also possible adding custom texture (previously added to the Asset folder) clicking on the *Albedo* option. Using this method is possible to personalize each element of the model, adapting it to each purpose. The next step is to make movable the model to be examined using the mouse while the application is running. This is made possible using two custom scripts which must be loaded into the *Main Camera* present on the scene. One script allows the user to move and rotate the model binding the camera view to a pivot point., while the other script allows to zoom in and out using the specific zoom button of the mouse.

- Movement script:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CamScript : MonoBehaviour {

    public Transform cube;

    public float sensitivityX;
    public float sensitivityY;
    GameObject pivot;

    void Start()
    {
        pivot = new GameObject("pivot");
        pivot.transform.position = cube.position;
        transform.SetParent(pivot.transform);
    }

    void Update()
    {
```

```

        pivot.transform.eulerAngles += new Vector3(Input.GetAxis("Mouse Y") * -sensitivityX *
Time.deltaTime, Input.GetAxis("Mouse X") * sensitivityY * Time.deltaTime, 0);
    }
}

```

- Zoom script:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Scroll2 : MonoBehaviour {

    float ROTSpeed = 300f;
    void Update()
    {
        gameObject.transform.Translate(0, 0, Input.GetAxis("Mouse ScrollWheel") * ROTSpeed);
    }
}

```

Adding the first script to the *Main Camera*, it will create three task bar in the Inspector Tab under the relative section. These three option allow to assign the position of the pivot and the movement sensitivity along X and Y directions. The limitation of this script is that the user view will be always anchored to the pivot position during the entire scene reproduction. In the second script modifying the “ROTSpeed” value it is possible regulate the zooming sensitivity, also this script needs to be added to the *Main Camera*. Now the model can be moved and analysed using the mouse. To complete the scene interface, seven interactive buttons have been inserted as explained in chapter 5.3.1. They allow the user to access to each level of the new building and one button is used to go back at the Main Scene. In each button is used the changing scene script (Figure 39).

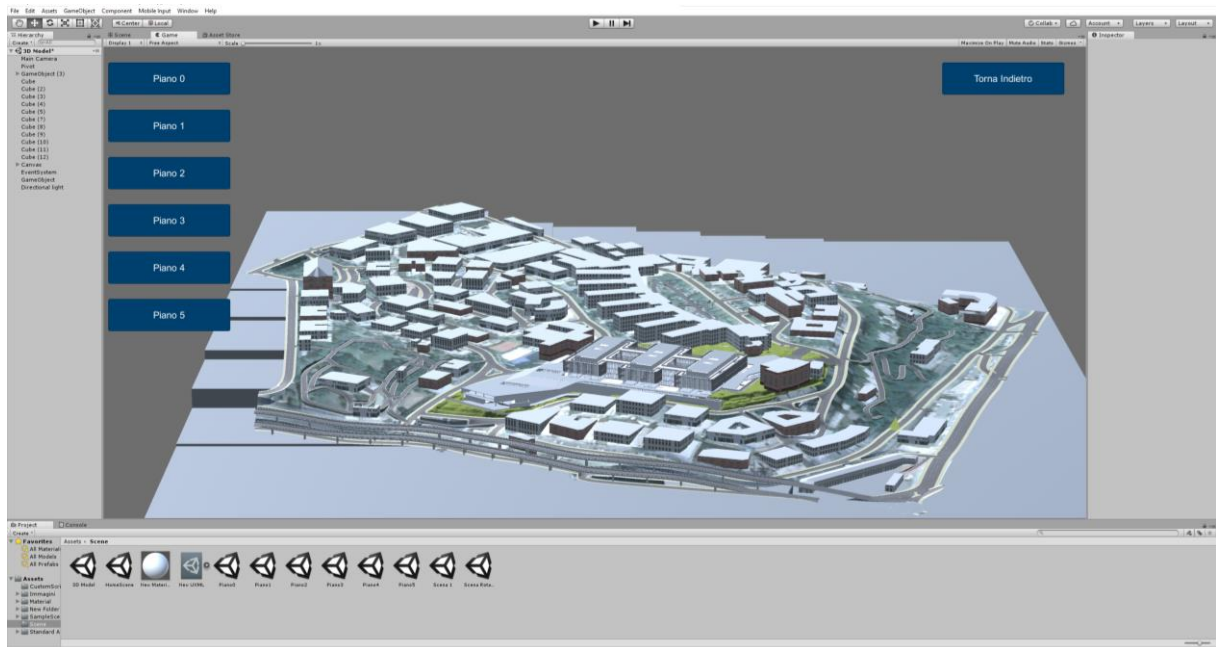


Figure 39: Unity – Urban Context Scene.

In a similar way of the previous part, each level scene has been created starting from the “.fbx file” exported from Revit. The process to import the model in Unity is the same of the previous one. The main different with the construction of the previous scene is that in this case another “.fbx file” has been exported from Revit, containing only floors elements divided by

department. This method has been necessary to highlight the floor of each department assigning them different material with different colour, based on the FM classification of the Revit model. The aim of the following scene, in fact, is to access not only to the geometry aspects of each level but also to know the designated use of each space.

Importing the two different model in the scene (level and floor), they will have the same sharing coordinates from Revit and this will connect automatically them inside the scene. Assigning the correct material to each element, the model will be ready to be analysed and this will be done by the movement and zooming scripts viewed in the previous part. Inside the scene two *Canvas Images* have been imported to add the Department Legend of colours and the number of beds present in that level. Then a button, to go back to the urban context scene, has been added always using the changing scene script (Figure 40).

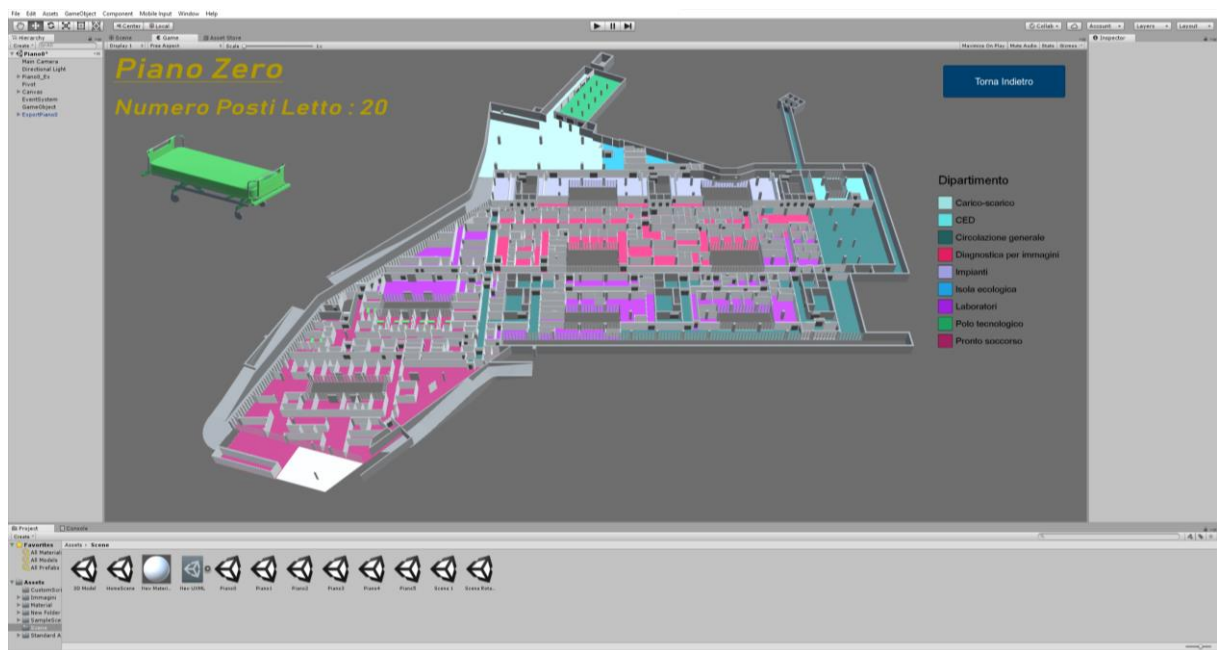


Figure 40: Unity – Level Scene.

The same procedure is repeated for each scene contained several levels. In this way the user can discover easily all departments located into the Hospital making clearer the project communication with professional and non-professional people.

5.3.3 Navigation Scene Development

The last part of the current application is represented by a virtual walkthrough where the user can discover a small part of the new project, in particular the main entrance located on the third floor of Alessandro Volta street. The aim of this part is to create a virtual environment that the user can explore interacting with object and data, getting from the Revit model, allowing to understand how people that will use the Hospital can react exploring it. The Navigation will contain several information map to understand if this kind of information is suitable for people that will use the structure. In addition with this, the player will receive some indication to reach different parts of the Hospital and it can choose to use (through specific information Totem) the information map or to activate an help path which shows the way to follow.

How in the previous chapter the procedure start importing from the Revit model the portion of model that has to be explored, using always 3Ds Max as intermediate software. Loading the “.fbx file” in the Asset folder, before to move it in the Hierarchy Tab it is essential select from the Inspector Tab the *Generate Collider* option (Figure 41-left) to ensure that all elements present in the model will have a solid geometry necessary for a correct navigation. Also this time, to obtain a better model quality new materials have been created. The *Directional Light* object present in the scene by default can be keep, while the *Main Camera* can be deleted because it is only a static camera not allowing movement inside the scene. For our aim, in fact, it is necessary load a specific “player” and this is possible downloading from the *Asset Store* the *Standard Assets Package* and importing it clicking in *Assets > Import Package > Custom Package* (Figure 41-center). In the Asset folder on the Project Tab will be loaded this new package where it is located the *First Person Controller (FPSc)* object, that will be our “player”.

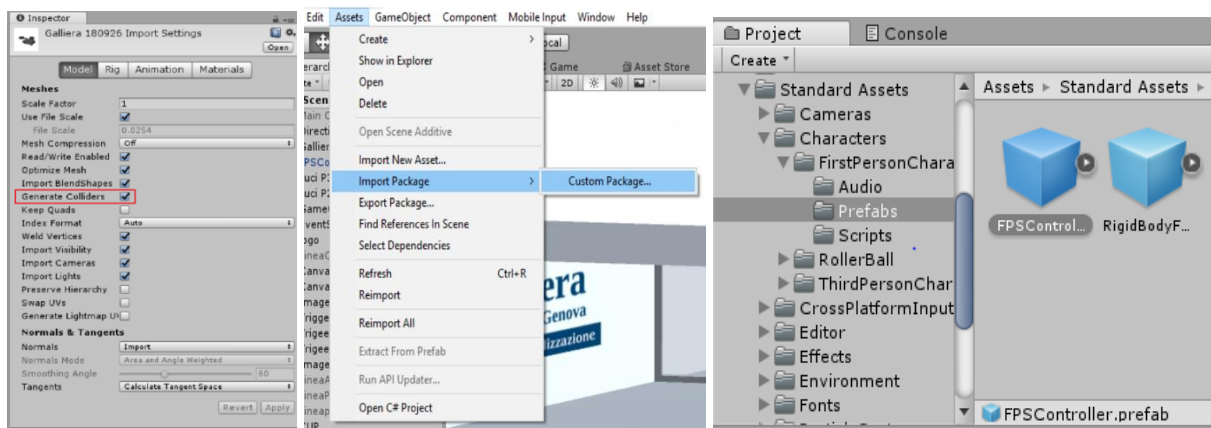


Figure 41: Unity – Scene Creation.

Dragging the *FPSc* in the Hierarchy Tab and positioning it, in a correct position, the model can be navigable after pressing the Play button (Figure 42). Features of *FPSc* can be modified using the Inspector Tab, in our case only the *Walk Speed* option and dimensions of player have been changed. To recreate a more realistic environment other element could be loaded as as trees, the terrain and the sky. All these passages have done importing the related package as *Tree Creator*, *Terrain Assets* and *SkyBoxes*, from the Standard Asset folder previously inserted.



Figure 42: Unity – Player Creation.

The next step is to improve the scene, adding interactive objects and several information which can help users during the navigation. For this aim different solutions have been found inserting informative maps, indications, buttons, help path that can be activate and deactivate.

One of the applications found for this scene is to have elements that automatically appear on the user view when it is located in a specific place. The process to obtain this starts creating a general object (in this case a cube) through *Game Object menu > 3D Object > Cube*. After located the object on chosen place, it is essential to select the option “Is Trigger” from the *Box Collider* section in the Inspector Tab (Figure 43).

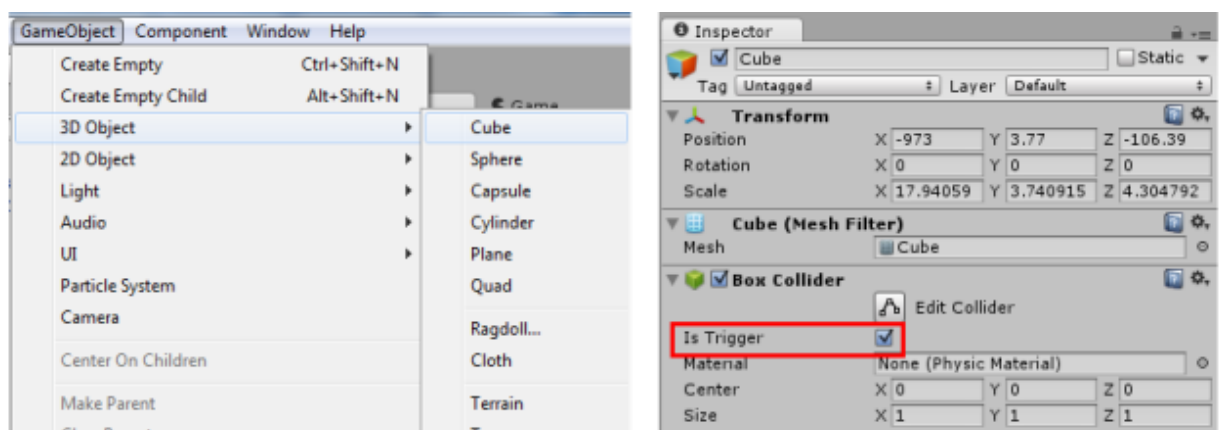


Figure 43: Unity – Interactive Object Creation.

This element will be the point of reference and to give it the ability to bringing up a picture when the player is close, a new custom script must be created:

- Image Appear script:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class ImageCode : MonoBehaviour {

    [SerializeField] private Image customImage;

    void OnTriggerEnter(Collider other)
    {
        if (other.CompareTag("Player"))
        {
            customImage.enabled = false;
        }
    }
}
```

The script allows to show a specific hided image when element with the “player tag” comes into contact with the element that contains this script and with the *Trigger* option activated. Therefore the current *FPSc* in our scene has to contain the correct tag, going on the Inspector Tab on the *Tag option > Player*. The last step is to set correctly the image and the cube. The previous script must be loaded in the cube of reference and it is also possible to make invisible the cube clicking on the *Mesh Render* option. Under the script section of the cube, in the task bar of the *Custom Image* option, the *Canvas* element that has to appear must be dragging in. The image must be present on the Hierarchy Tab and it is necessary hide the element from the Inspector Tab to visualize it only when the script will be activated. Following this kind of procedure each element can be used to appear and disappear, also several buttons have been used with this method coming into the view user only in specific place.

In the whole model also static elements, categorized as image, have been loaded as simple *Sprite Image* or *Canvas Image*. The different is that the first one is a normal image just moved into the Hierarchy Tab and adapted to the context, like it is a painting. The second one is regulated as a *UI* element, therefore it will be applied to the user camera and could have interactive features.

Some of the buttons present along the whole navigation are equal to the previous button to change scene or to end the game. In addition with these buttons two other types, with new function, have been inserted. One has been realised to restart the current scene leaving again from the starting point, also in this case a custom script has been used.

- Restart Button Script:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class RestatBtn : MonoBehaviour {

    public void RestartGame()
    {
        SceneManager.LoadScene(SceneManager.GetActiveScene().name); // loads current scene
    }
}
```

To add this function to one button, it is necessary create an empty game object to contain the script and drag this object on the *On click* option in the Inspector Tab clicking on *No Function* > *RestartBtn* > *RestartGame*. The other types of buttons are used to activate and deactivate the help paths present into the scene. Before setting the button it is necessary to insert the help path in the scene and in this case through *Create* > *3D Object* > *Cube*. After resizing and positioning each objects as required, all the elements have to be included in a unique group which has to be deactivate through the tick in the Inspector Tab. At this point two buttons must be created (one for the activation and one for deactivation) and to connect them with the object, the group previously created has to be dragged into the task bar of the *On Click* option. Now, clicking on *No function* > *GameObject* > *SetActive* the button will be connected with the element. For the activation button the tick must be clicked and the element hidden into the scene, while for the deactivation button the tick must be empty.

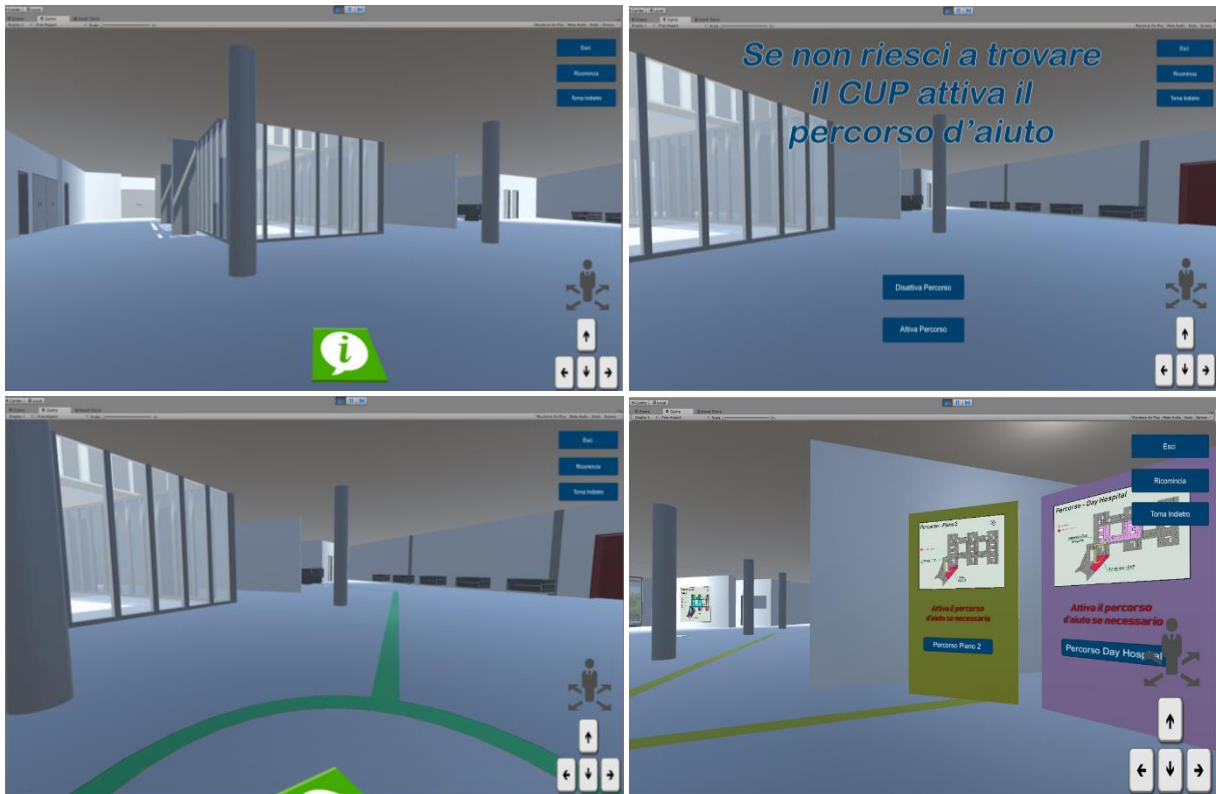


Figure 44: Unity – Examples Navigation Scene.

Using all the previous elements it is possible obtain an adequate quality of navigation and interaction with the model, allowing to different target of people to discover in a new way the new Hospital Project how showed in Figure 44 .

5.3.4 Standalone Game Creation

To conclude the application process, it is necessary create from the whole project an “.exe file” which can be shared and used from several stakeholder. Unity makes available several ways to “build a game” from your model. All these options are present into the *File menu* > *Build settings* (Figure 45).

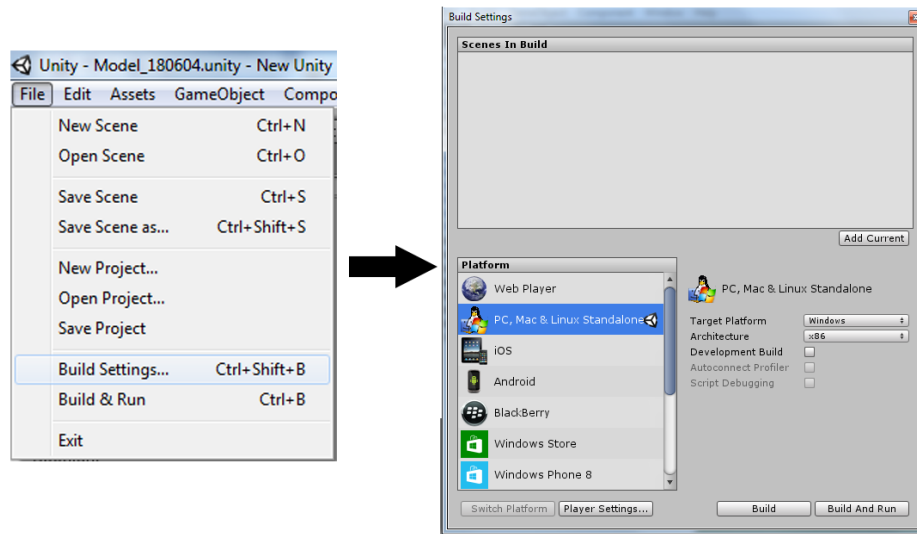


Figure 45: Unity – Standalone File Creation

From this window, in fact, user can create for example a platform for Smartphone (Android, iOS, etc.), video game console (Xbox and PlayStation) and standalone files usable from computer. In this example the *PC Standalone* option has been chosen. There is also the opportunity to use the *Web Player* option, which allows creating a platform available from web. However, this option sometimes brings problem of interoperability with some of the most common browsers as Google Chrome. Through the previous option, Unity allows creating an independent folder with the “.exe” game file and all data necessary to launch the simulation, selecting only the kind of platform and where to save the new folder. At the end of the building process, a folder with the application will be created and it can be sent and used without using Unity software just clicked on the play button.



Figure 46: Video – Not Immersive.Application.

5.4 VR Immersive Application

In the following chapter the previous application will be transformed in a solution to use through Virtual Reality Immersive headset, and in this case using the Oculus Rift device. The main structure of the previous application will be keep, but plus another scene will be inserted. The new structure will be divided how show in Figure 47. This scene has been developed to create an interactive simulation allowing non-professional people and medical staff to participate on the design phase of a specific Hospital section. The user will be able to decide how set up the environment choosing and moving physically the object inside the scene.

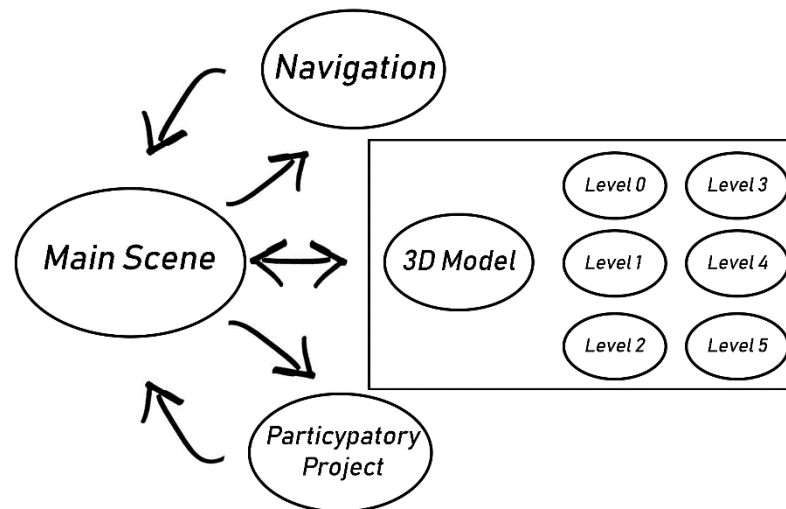


Figure 47: Second Workflow Development.

How sketched only four scenes are present in this new solution, indeed the previous scenes of the 3D model have been grouped in a unique scene. In this way the user can interact both with the urban context and with the 3D of each level.

5.4.1 Re-elaboration for Immersive Simulation

To create an immersive solution it is necessary to change approach toward the structure of each scene. This time, in fact, the application will not be reproduce using a screen but the same user will be teleported in a virtual environment. For this reasons scenes will be set up as separated rooms, where it is possible to move freely and interact with object.

The first step is to set up the software for the Virtual Reality and in this case through the Oculus Rift device. Oculus web site provides several packages to use Oculus with Unity, in this case three package must be downloaded from the developer section of the web site. They are *Oculus.Utilities Package*, *Oculus.Platform Package* and *Oculus.Avatar Package*. Before to add these elements to the project it is necessary “to unzip” the folder where files are located, then they can be added clicking on Asset > Import Package > Custom Package³. Then it is necessary click on Edit > Project setting > Player and in the Inspector Tab under the *XR Settings* section check on *Virtual Reality Supported* (Figure 48). The last package to download is provided from the *Asset Store* with the name of *VR Samples* and importing all components. Now the project is ready to be developed in VR.

³ Only the respective “.package files” have to be imported, not the whole folder.

The room for the Main Scene has been created simply loading new game objects into the scene using *Create > 3D Object > Cube* for the walls and *Create > 3D Object > Plane* for the floor. Now a player has to be present in the scene, from the Oculus folder in the Assets folder it is possible to import the *OVRPlayerController* object in the Hierarchy Tab, this will give you the ability to move around into the scene. After creating the room, interactive buttons have to be inserted to teleport the user in each scene. Respect to the previous application, preset buttons have been used. The relative buttons have been extrapolated from the demo scene “Intro”, present in the Asset folder under *VRSampleScens*. From this scene it is necessary to copy highlighted elements in the center Figure 48, moving them in the Hierarchy Tab of the Main Scene. Now the player have to be set up correctly to interact with objects, for this the *MainCamera* previously copied must be moved under *OVRPlayerController* and in substituting the *CenterEyeAnchor* (Figure 48) element. This element will represent the user view and through the precedent procedure it will have all essential script to work properly, avoiding to add them individually. The last step to conclude the player setting is to insert the object that will replace the user hands in the virtual scene. To do this *LocalAvatar* object must be dragged from the Assets folder inside the player object and exactly the *TrackingSpace* voice (Figure 48).

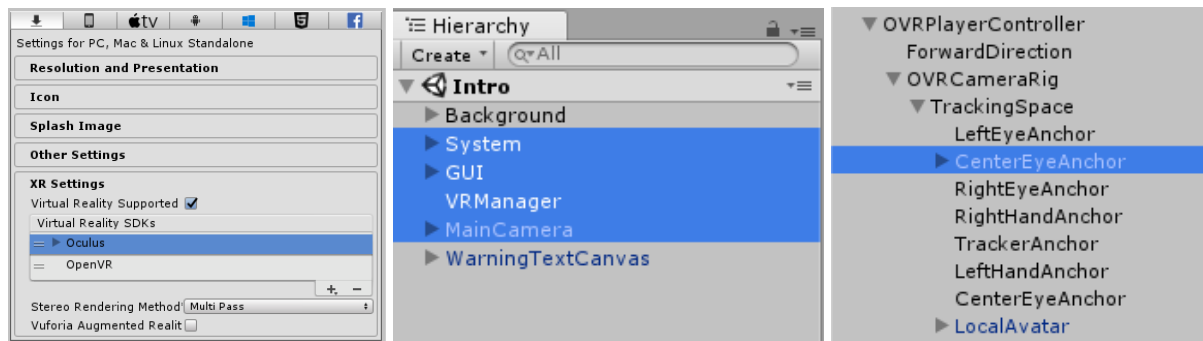


Figure 48: Unity – Immersive Scene Creation.

All essential element to set up the interactive buttons are present in the *GUI* object, this group is made up of *Canvas* elements (text and slider button) just preset to be used in VR. Previous elements can be modified for position, dimension and contents, depending on the user aims. Also in this case, the important part is to assign the function to the button. That is possible using the same procedure viewed in the Chapter 5.3.1, adding on the *Slider* element in the Hierarchy Tab, the correlated function on the *On Value Changed* option.

On important aspect, different from the previous application, concern the use of *UI* elements in a VR scene. Using a virtual headset, in fact, *UI* elements cannot be added to the screen like before. Therefore clicking on the *Canvas* element, in the Inspector Tab, under the Canvas section it is necessary change the option *Screen Space – Overlay* with the option *World Space*. This solution allow to position interactive elements in a specific point of the scene.

How mentioned above, the Main Scene has been modified for the Virtual Reality. Now the scene consists in a room with two interactive section adding to the walls. In one section it is explained how use buttons through the controllers and in the other one, it is present the main menu where the user can choose in which scene to be teleported (Figure 49).

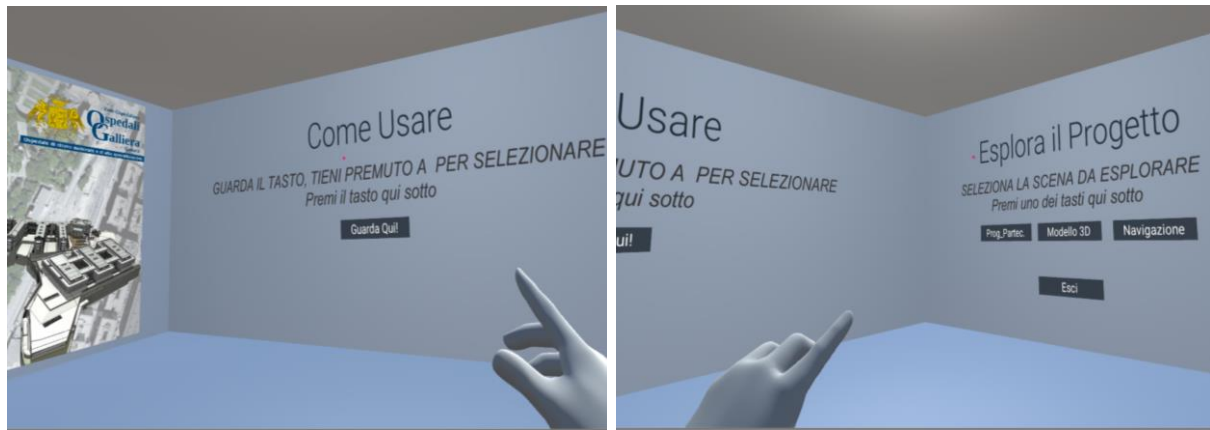


Figure 49: Unity - Immersive Main Scene.

5.4.2 3D Model Immersive Scene

For the 3D Model section in this application has been created only on scene for all building models. Two separated work stations have been created to access both to the Urban Context model and to each building level. Also in this case the design approach to the scene has been changed because to build several scenes for each building floor would have been too heavy for a VR simulation. Two table will represent the work sections and on each table, the user will find the respective model.

The current scene has been created based on the previous chapter. The same methodology was used to recreate the room and to insert the player and interactive buttons. Later the different models have been imported using the same “.fbx file” of the not-immersive application and positioned on their respective table how showed in Figure 50.

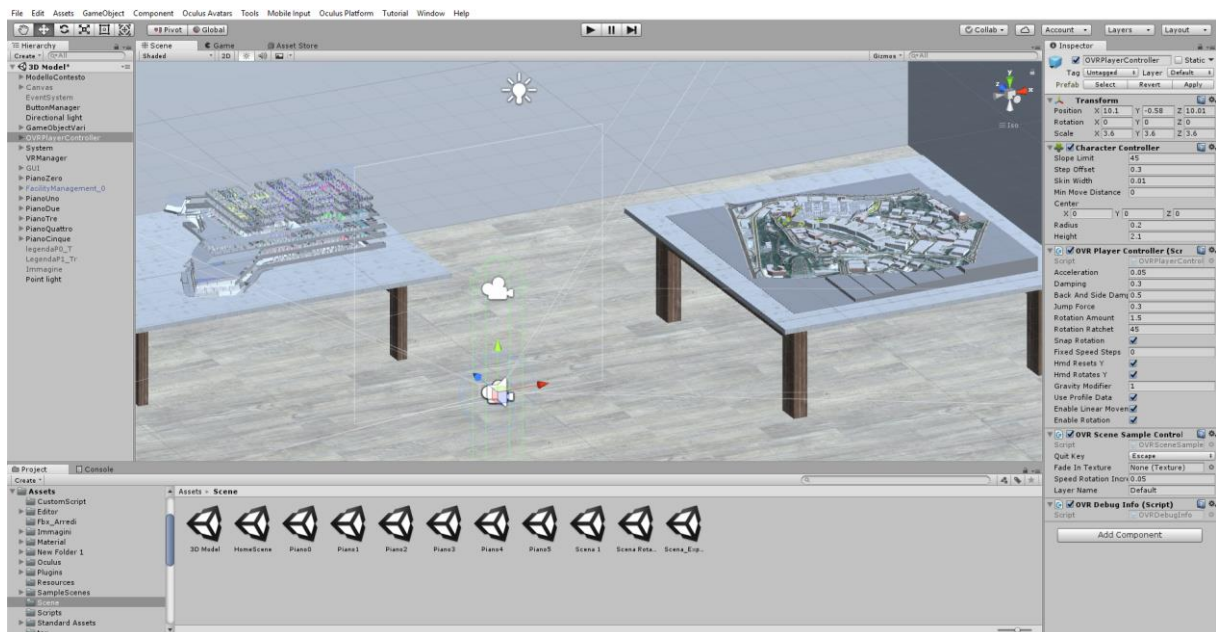


Figure 50: Immersive 3D Model Scene.

Starting from this disposition only the table with building floors will be provided of interactive buttons to activate and deactivate each floor and to have access to the FM data. The second

table, instead will be used only to consult the Urban Context model. For the interactive table two button for each level have been inserted using the same methodology of the Chapter 5.3.3 (*No function > GameObject > SetActive*). Using the previous methodology the user can click the button activating the relative floor and the FM data buttons. In fact, in the inspector Tab through the *On Value Changed* option it is possible to add more task bar clicking on the “+” symbol. This allow adding several functions in one button when it will be clicked and in this case to show more than one object. The FM information have been extrapolated from the Revit model and using 3Ds Max, volumes of each department have been created and then imported in Unity. Through the FM buttons user can activate the coloured volumes to visualize the department division of each level and to see some data concerned to them.

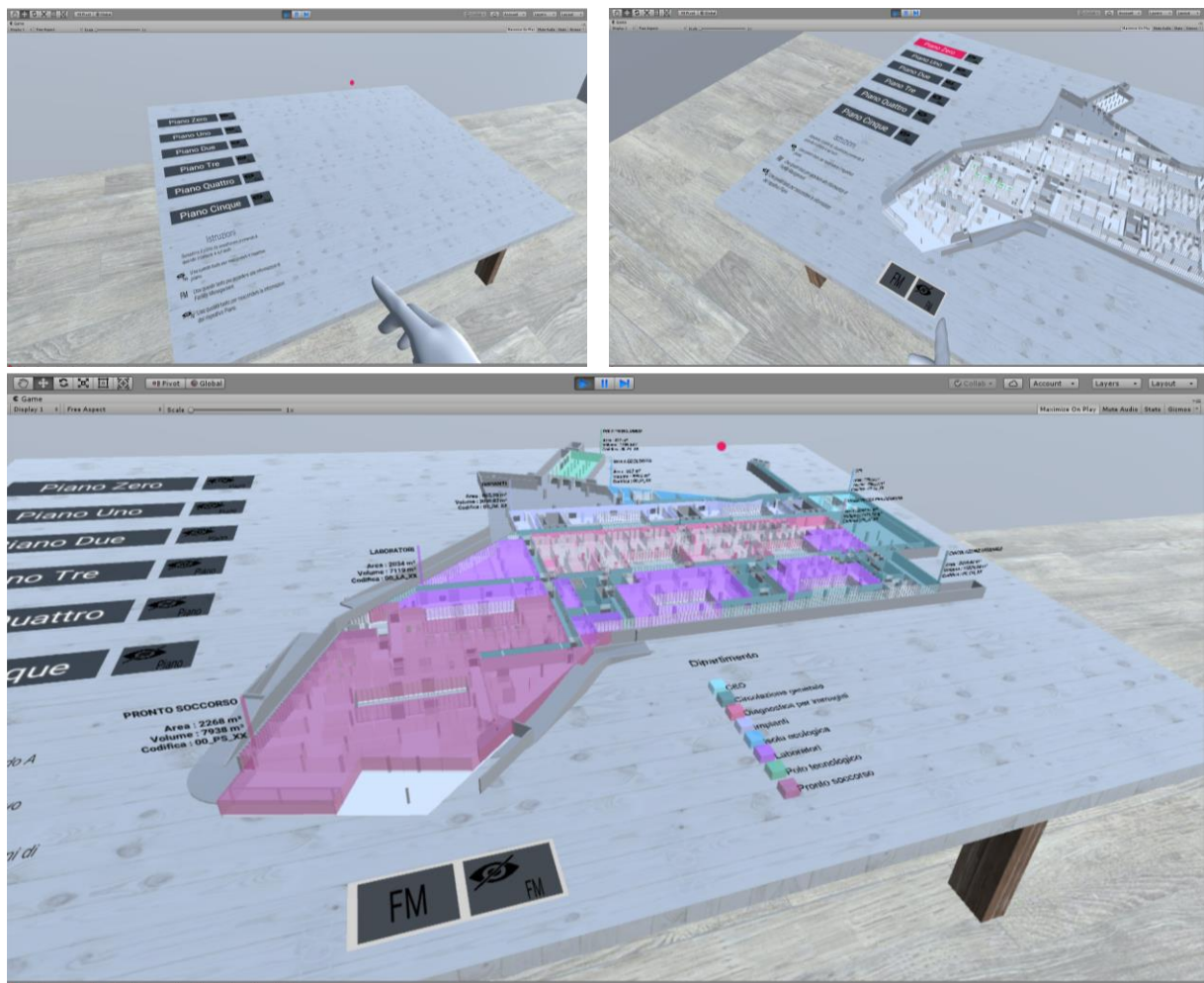


Figure 51: Unity – Data Interaction.

To have the ability to go back on the Main Scene also a Button with the script to change scene has been inserted.

5.4.3 Participatory Project Design Scene

How above, this last scene is based on the previous two. In this case the user is teleported in a small section of the new Hospital, where through specific buttons can import into the scene medical furniture. Each loaded object is provided of a data sheet, that the user can choose to

consult. The player will be also able to grab the relative element using the controllers, moving and positioning it everywhere in the scene.

The room model has been imported isolating the interested part from the whole Revit model and exporting it in “.fbx format”. The same player of the previous scene has been loaded, with all elements copied from the standard “Intro” scene. To allow the user grabbing object, it is necessary to modify some elements in the current *OVRPlayerController*. Going in the Hierarchy Tab under the voice *TrackingSpace* on the player object and clicking on the *Left/RightHandAnchor* element it is necessary add in the Inspector Tab, through *Add Component* option, *Sphere Collider*, *Rigidbody* and *OVR Grabber* script. Set *Is Trigger* option and 0.05 as *Radius* value in the first script and to check on *Is Kinematic* option for the second one. In the *OVRGrabber* script under the voice *Grab Volumes*, to press one as *Size* value and then to drag the relative *Left/RightHandAnchor* object in the *Grip Transform* and *Element 0* task bar (Figure X). Through the *Controller* option in the *OVR Grabber* script, it is possible to choose the controller button that the user must use to grab elements. Now the player has all essential features to drag element and testing the scene the current changes should be visible. However, to allow a physic object to be grabbed also this element must be implemented. Positioning the object into the scene, three components have to be added in the Inspector Tab: *Rigidbody*, *OVR Grabbable*, *Box Collider*. To have a better quality of the results each element has to be set according to the user aims, but the essential option in order that the procedure works is to drag the relative object under the voice *Grab Points* in the *OVR Grabbable* (Figure 52).

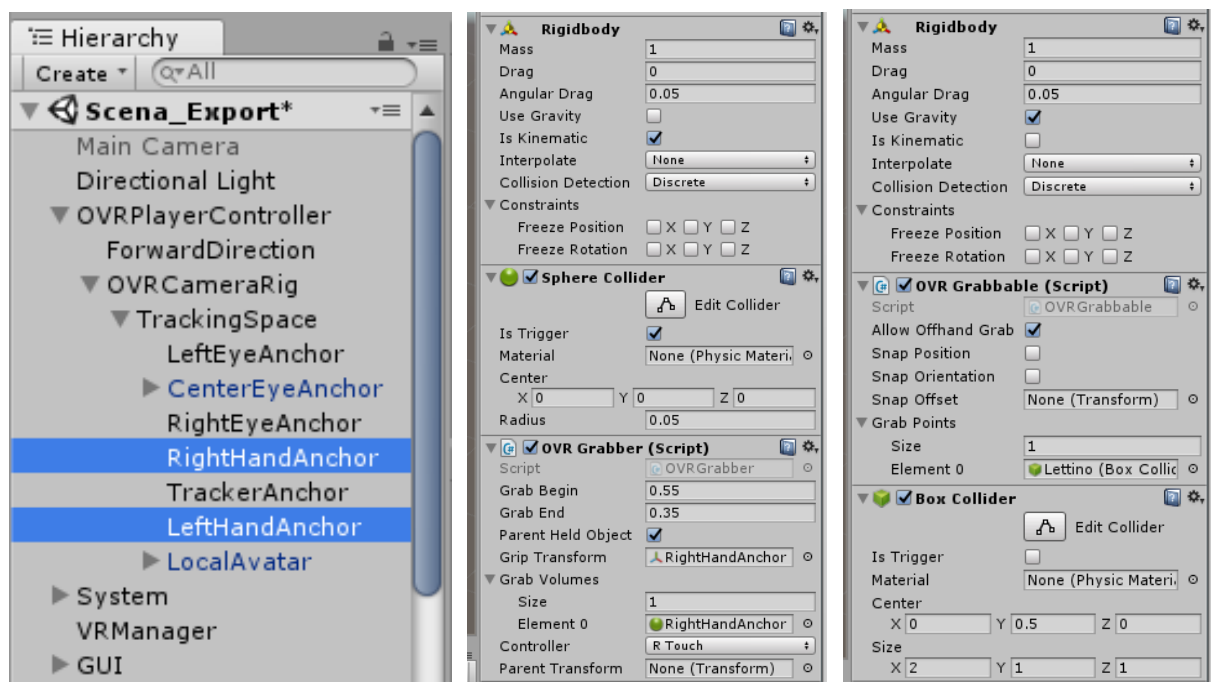


Figure 52: Unity – Player Configuration.

To select the object that the user can insert into the room, an interactive menu has been created using the same interactive buttons of the previous scenes, changing shape and contents. Using the instruction above the menu (Figure 53), the user can choose from different furnishings and for each object other three buttons are present: one to hide the object, one to show the data sheet of the object and another one to hide this last one.

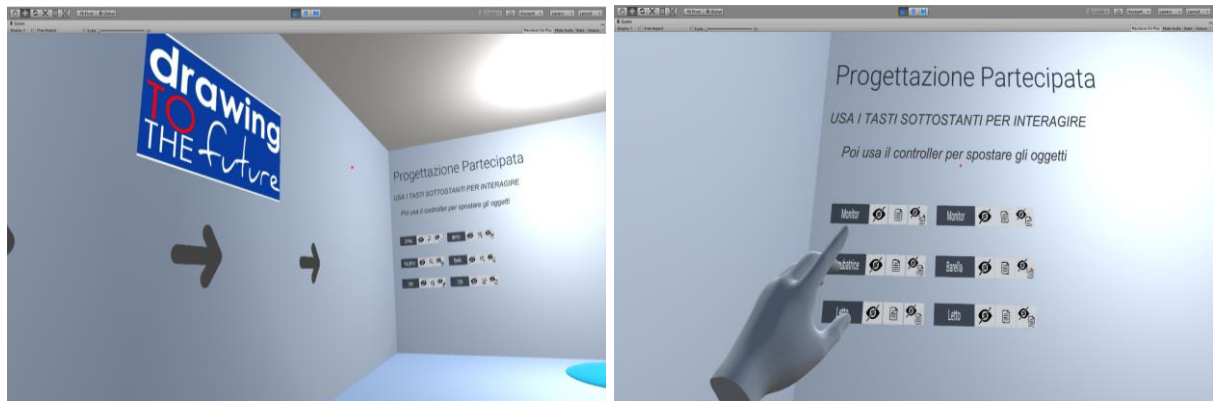


Figure 53: Unity – Interactive Menu.

The buttons use the function to activate and deactivate the object already loaded in the Hierarchy Tab, so present in the scene. In this case only few furnishings are present on the menu, this because the following scene provides an example of the Participatory Design methodology chosen for this solution (Figure 54). Obviously, the scene can be implemented adding more object or using new function. The procedure to create an executable file, to use the application without use Unity is the same of the Chapter 5.3.4.



Figure 54: Unity – Immersive Participatory Design Scene.



Figure 55: Video – Immersive Application

5.5 Results

This chapter have showed how a Virtual Simulation of a building project can be create, starting from a Revit model. It goes without says that the applications of this methodology are almost unlimited in the AEC sector, bringing benefits for all phases and sectors of a construction process. During the virtual exploration, a general user can access to several information of the building.

How saw in some part of the process there are some methodology limitations yet, interoperability problems or problematic procedures that makes the technology not easily accessible. But this represents a normal aspect for a technology that only on the last years has seen a continuous development. On the other hand, the building sector cannot ignore how important Virtual Reality is becoming for all construction and design phases, solving problems that always have been the weak point of the AEC industry.

6 Other Application –

As Virtual Reality, nowadays other visualization technologies start to be very common in the AEC industry and Augmented Reality (AR) represents one of these. AR refers to a set of technologies and devices able to enhance and improve human perception, where physical and artificial objects are mixed together in a hybrid space where the user can move without constraints (Manuri January 2016). Many applications are developed using this technologies in several fields, as medicine, manufacturing and repair, culture heritage, architecture, education and so on. In this context it is allocated the Mixed Reality (MR), which denotes the possibility of the user to interact with these virtual objects (Debandi 2018). This technologies are based on holographic representation, inserting virtual elements in the real environment giving the impression to interact physically with the object (Figure 56).

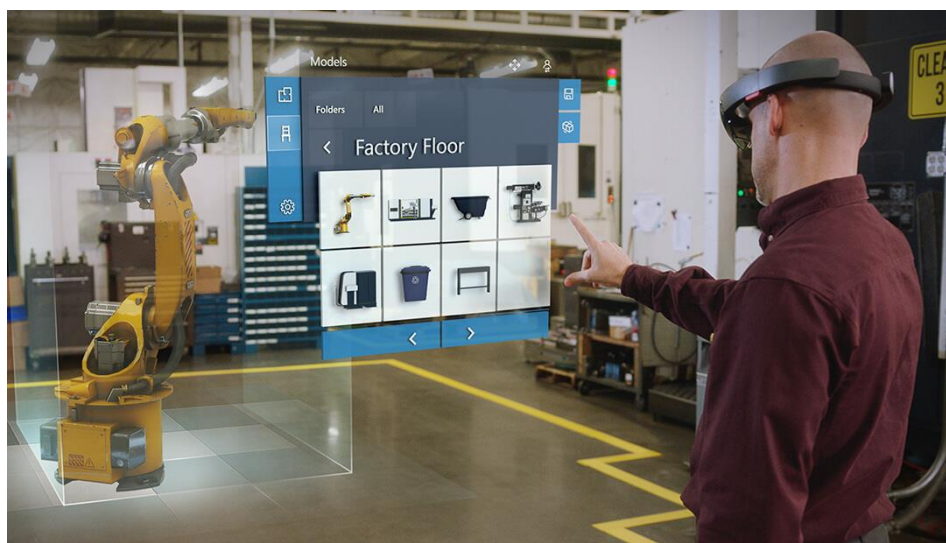


Figure 56: Microsoft HoloLens application (Microsoft 2018).

In this chapter two applications will be developed using as much devices and in particular: Microsoft HoloLens and the Olobox. The aim is to test how other augmented reality application can be used for new project as the study case.

6.1 Mixed Reality

The Microsoft HoloLens device is a Head Mounted Display (HMD) for mixed reality applications. They are composed from a pair of transparent combiner lenses where holographic elements are displayed, overlapping them to the surrounding environment. Windows 10 is the operating system installed on the devices and in addition to an Intel Cherry Trail SoC containing the CPU and GPU, HoloLens has a custom-made Microsoft Holographic Processing Unit (HPU), a coprocessor manufactured specially for the device. Using a personal hardware and software, the main advantage of this technology is the ability to use it as standalone device, because it does not require an external computer for computation, featuring also a Wi-Fi and Bluetooth wireless connectivity.



Figure 57: HoloLens Headset.

Microsoft HoloLens has been used to create a custom hologram starting from the Revit model of the current Hospital project. The aim is to create an hologram of the new Hospital with the urban context, which can be manipulate from the user; also for this part, the application has been developed using Unity software and Microsoft Visual Studio.

How saw for the Oculus Rift application in the Chapter 5.4, the first step is to set up Unity for the Mixed Reality and in this case, using Microsoft configuration. Through the web site of HoloLens on *Developer* section > *Install the tools*, it is possible download all required components and especially the *Mixed Reality Toolkit* package. To import the previous package into Unity, only some elements present in the Asset folder of the HoloToolkit have to be imported how showed in Figure 58. Dragging all components in the project Asset folder, a new section will appear on menu tab with the of *Mixed Reality Toolkit*. Clicking on it and then in *Configure*, all three option has to be set up to configure Unity: *Apply Mixed Reality Project Settings*, *Apply Mixed Reality Scene Settings* and *Apply UWP Capability Settings*⁴. The last configuration step is to click on *Edit > Project Settings > Player* and in the *Inspector* Tab under *XR Settings* check the *Virtual Reality Supported* option, verifying that *Windows Mixed Reality* is listed as supported device (Figure 58).

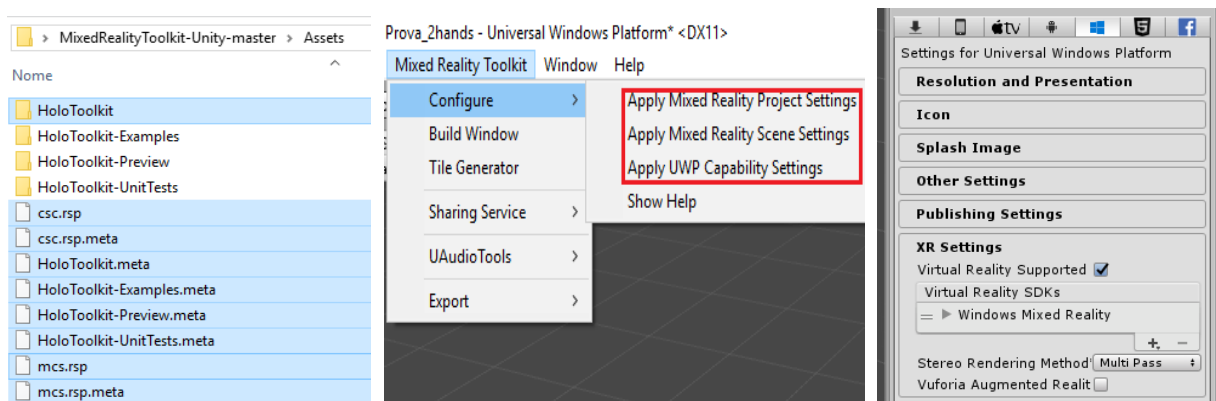


Figure 58: HoloLens Configuration.

To develop a simply application based on hologram, it is possible to use the pre-set elements allocated in the *HoloToolkit* folder. Therefore the following prefab object have been imported in the Hierarchy Tab: *MixedRealityCameraParent*, *DefaultCursor*, *InputManager*. These elements provide a base structure to develop HoloLens applications, giving the ability to implement the project according to the aims. The next step is to import the object that will be

⁴ This procedure could require a restart of the software.

showed as hologram, using the same methodology of the previous chapters and creating an “.fbx file” suitable for the Unity format. The position of this object has to be adapted with the view field of the user camera changing, if needed, the value on the Inspector Tab under the *Camera section > Field of View*. The *InputManager* object, indeed, allows the correct operation of the application for different platform, for this it is essential connect the *DefaultCursor* object with it. This is possible going on the Inspector Tab of the *InputManager* under the *Simple Single Pointer Selector* script, dragging the *DefaultCursor* from the Hierarchy Tab into the task bar of the *Cursor* option.

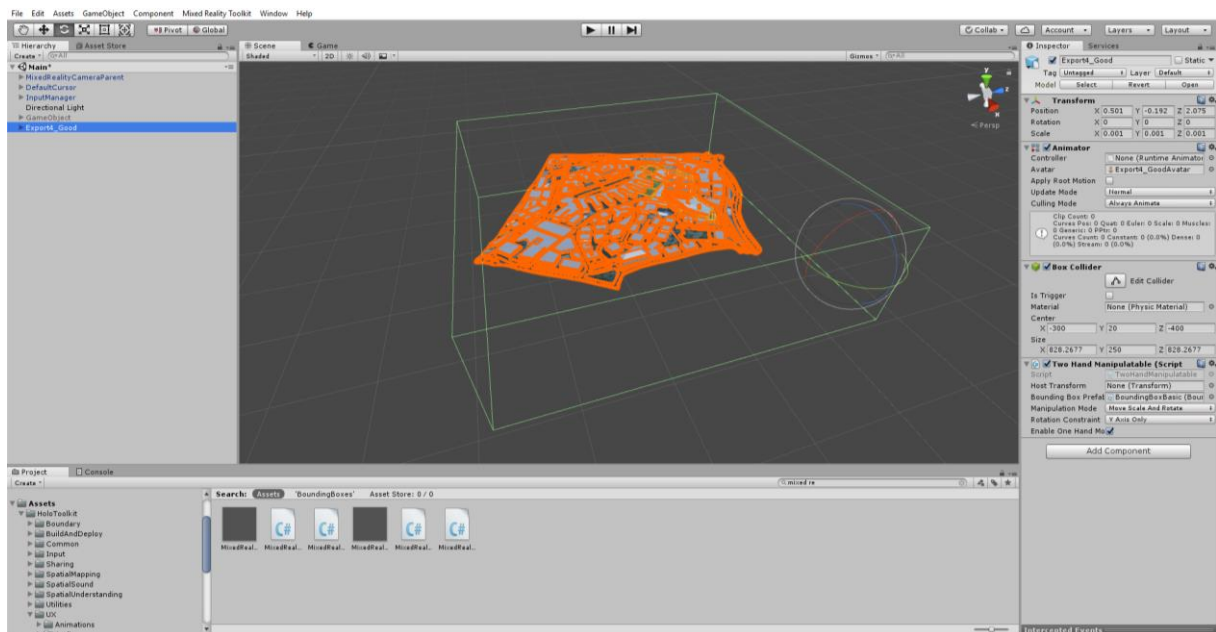


Figure 59: Hologram Creation.

Through the previous passages, the project is set up correctly for the HoloLens view but momentarily the custom object previously inserted is only a static element. To make possible to interact with the object during the visualization it is necessary add to the object two script using the Add Component button in the Inspector Tab: *Box Collider* and *Two Hand Manipulatable* script⁵. Dimensions of Box Collider have to be adapted using the *Size* and *Center* option how showed in the previous picture, it will represent the boundary box that will appear when the user will interact with the object. The second added element, indeed, allows manipulating the object using the hands. To set correctly this script on the Bounding Box Prefab option has to be inserted the prefab *BoundingBoxBasic* element, allocated on the Asset folder. Now the project is ready to be loaded on the device.

To export the project it is necessary to access on *File > Build Settings* and select the *Universal Windows Platform* option. After setting several field as in the Figure 60, clicking on the *Build* button, a new folder to contain the project has to be created. This procedure will create a new folder where it is allocated a Microsoft Visual Studio file with the name of the project and opening this file the Visual Studio interface will appear. Through this window it is possible to test the application with a Emulator⁶ or directly connected the device to the computer using the

⁵ The script is provided from the Microsoft HoloToolkit package.

⁶ The Emulator has to been downloaded separately from Microsoft portal.

USB cable. The first time that the second option is used a PIN code has to be inserted, using the main menu of the HoloLens interface on *Settings > Update > For Developers* and tapping on *Pair* a PIN code will be displayed on the screen to type it in Visual Studio.

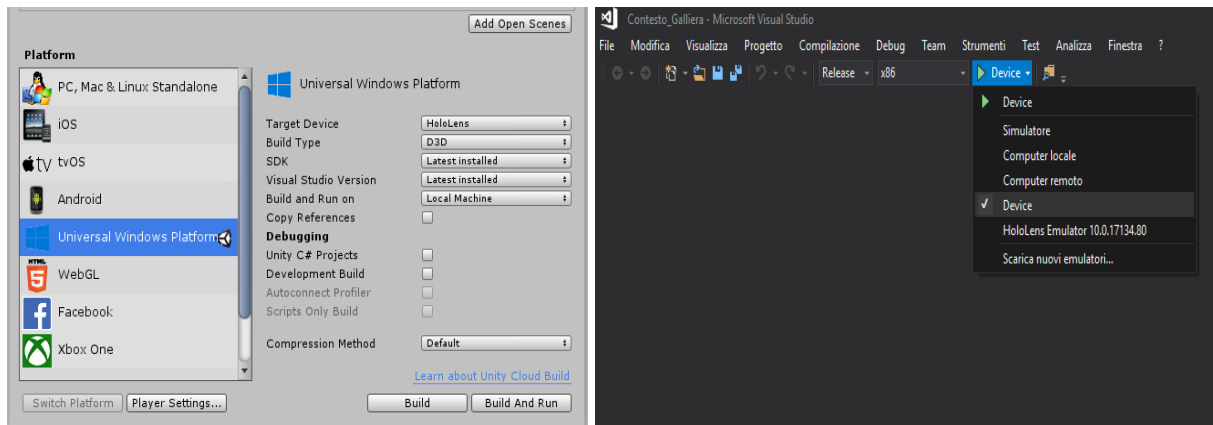


Figure 60: HoloLens Standalone Creation.

This procedure allows also loading the application on the device, saving the file on the HoloLens memory. To launch the application it is necessary to use the same settings showed on the Figure 61 and then to click on the *Device* button, after a few moments the hologram will appear automatically on the HoloLens screen.

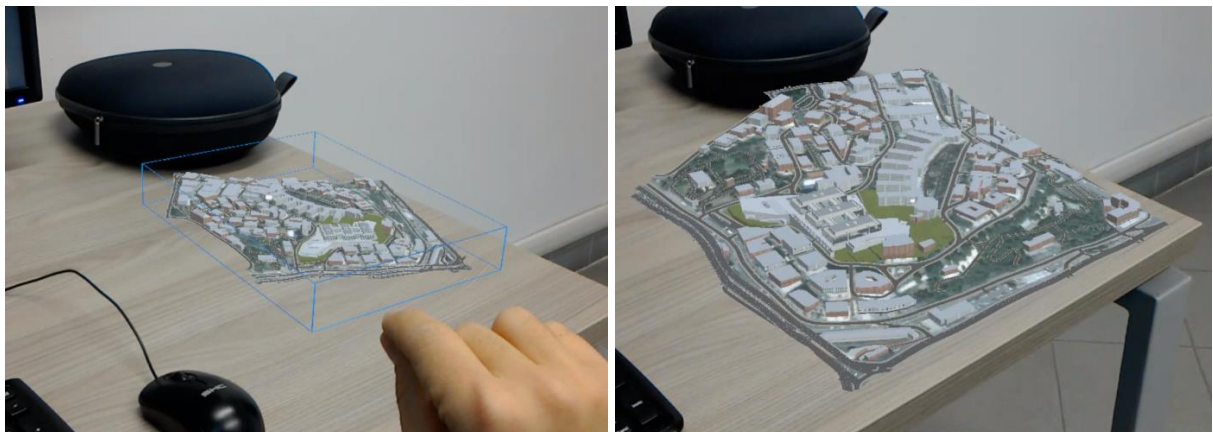


Figure 61: Microsoft HoloLens application.



Figure 62: Video – Mixed Reality Application

6.2 Holograms

In this part a will be used the 3D holographic screen Olobox, which allow projecting high quality image giving the idea of a three dimensional object. Essentially Olobox is composed from a monitor of 24 inch on the top part and from three pyramidal glasses, that redirect the screen image making a unique hologram. The device is equipped of an internal processor and a wireless LAN and Bluetooth connectivity, providing to control and manage the system through the web platform.

With this part two holograms of the New Project will be created, one with the new project inserted in the urban context and one with only the new project. During the visualization both holograms will rotate on their axis.



Figure 63: Olobox Design.

Olobox does not allow to manage directly 3D model or to create a hologram starting from an object, but it is necessary to realize an image or a video using other software before loading it on the web platform. For this reason other two intermediate software have been used: 3DS Max and Adobe Premier. The 3D model has been extrapolated from Revit using the “.fbx format”, then importing it in 3DS Max it was possible set up appropriate parameters. To obtain the rotation effect of the object, an animation have been added to the model allowing to have a rotation of 360°. After importing the model, it is necessary to insert a camera through *Create > Camera > Physical Camera* and then the light going on *Create > Lights*. After to have selecting the new *Camera* as current view and arranging the model position, it is necessary to click on *Auto Key* option and drag the task bar on the bottom from zero to 100. Now, clicking on the object and using the rotation button it is possible set a value of 360° to the zeta axis. The last step is to extract the animation of the model using the *Rendering Tab*, setting up the value according with graphic target to reach. It is possible to extract the result as a video or frame per frame, the second one allows creating a video with smaller dimension and it can be set through the *Render Setup* button. This last solution has been used for this application.

After created the folder with all frames, using Adobe Premier it is possible create a video starting from this folder clicking on *File > Import* and selecting the first frame of the list, automatically the software will import all frame creating a video. To be adequate for a hologram representation the video must be divided in three parts, positioning one sequence on the bottom, one on the left and one on the right (Figure 64). Each sequence has to be symmetrical with the screen axis to obtain a correct result.

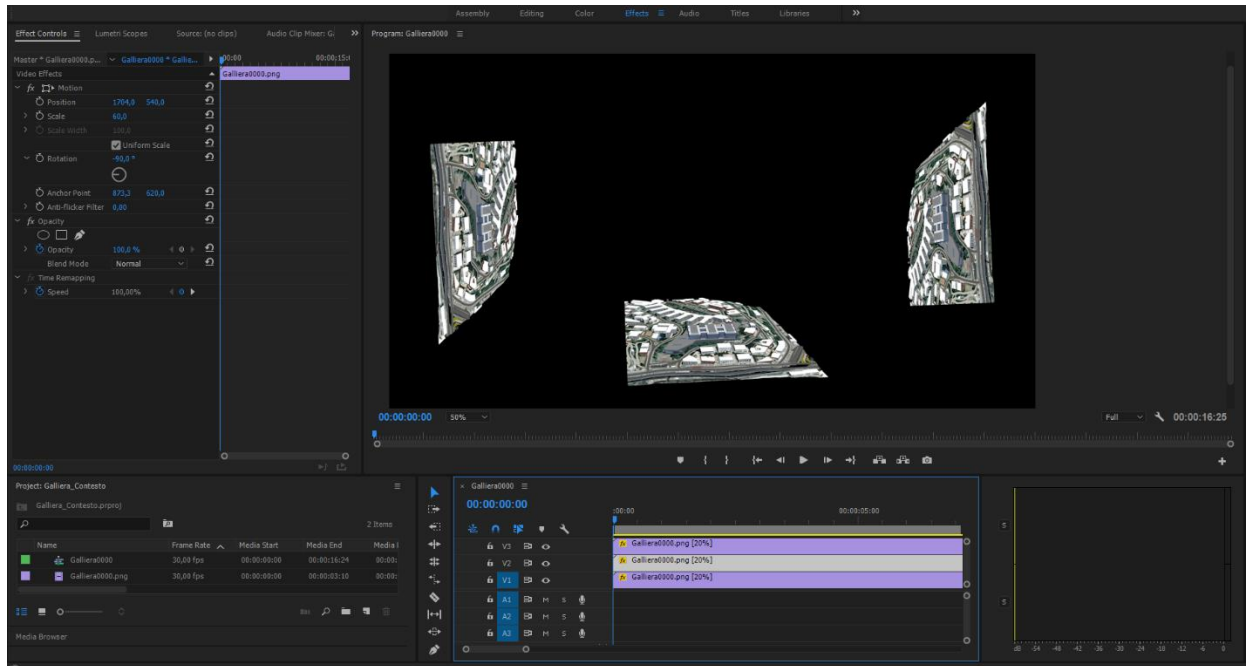


Figure 64: Hologram Video Creation.

Exporting the video, it is possible to load it on the Olobox using the web platform provided from the seller. It is possible to set the reproduction of only one hologram or to create a playlist with different object. After loading one of this two solution on the Olobox, it will be saved on the internal memory of the device, allowing continuing the presentation also without internet connection.



Figure 65: Olobox Application.

7 Conclusion

It is evident that construction industry is inside a period of strong changes, many new technologies start to be essential and new regulations govern the sector structure. Always more people are involved during each construction phase, to ensure a high efficiency during the whole building lifecycle, but at the same time complicating significantly the management project aspect. These and other problems are highlighted talking of healthcare structures, where the growing complexity of design and the huge number of data represent a difficult challenge for all stakeholders. In this framework new methodologies are necessary, and BIM is the leader, allowing to make more efficient the construction process.

However, functionally workflow is essential, particularly when we talk of data management. In this context the role of data is essential to storage and share information, but also complicated and not always accessible for everyone. Setting up a data structure, based on the ease of use and simplicity to understand, represents the key point for a great Facility Management method. Following this concept, a solution has been developed to promote information sharing, both for professional and non-professional subject, making faster data population and consultation from a Revit model. Through Dynamo applications, in fact, a possible management method has been provided which can be a base for future development, showing how powerful the computational design software is.

At the same time, to cover the communication gap between different involved actors, Virtual Reality represents a great solution. Integrating it in an efficient workflow, many problems of misunderstanding can be solved saving money and time. For this it was chosen to connect information management with real-time visualization, because both are two side of the same coin. 3D data, in fact, can bring a strong improvement in the whole sector, changing the same concept of construction that now we have. Nowadays new technologies make more available this methodology and, how showed, to create a virtual simulation becomes always easier.

This work wants to provide an overview about where the construction industry is going, displaying potentiality of new work methodologies applied to an important project example. The main purpose, in fact, was to develop a method, which could explain the deep change of the sector, showing how to integrate new technologies in a construction workflow. Following this idea, the thesis aims to produce a methodology that could be a starting point for future application based on sharing information and real-time visualization.

Reference

- Boeykens, S. *Unity for Architectural Visualization*. Birmingham, UK: Packt Publishing, 2013.
- Bouchlaghem, D., Shang, H., Whyte, J., & Ganah, A. "Visualisation in architecture, engineering and construction (AEC)." *Automation in Construction*, 2005: 287-295.
- BSI, British Standard Institute. *PAS 1192-3:2014 - Specification for information management for the operational phase of assets using building information modelling*. London: BSI Standards Limited, 2014.
- buildingSMART. *IFC Introduction*. 2018. <https://www.buildingsmart.org/about/what-is-openbim/ifc-introduction/> (accessed Aprile 29, 2018).
- Collinge, W.H. "The role of the design artefact in hospital briefing and design works." *HaCIRIC (Health and Care Infrastructure Research and Innovation)*. Cardiff, UK., 2012.
- Cooney, J.P. "The use of building information modelling (BIM) with healthcare." 2012.
- Debandi, F., Iacoviello, R., Messina, A., Montagnuolo, M. "Enhancing cultural tourism by a mixed reality application for outdoor navigation and information browsing using immersive devices." *IOP Conf. Series: Materials Science and Engineering*, 2018: 364.
- Dynamo. *The Anatomy of a Visula Program*. n.d. http://dynamoprimer.com/en/03_Anatomy-of-a-Dynamo-Definition/3-1_dynamo_nodes.html.
- Eastman, C., Teicholz, P., Sacks, R., Liston, K. *BIM Handbook : A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*. Hoboken, New Jersey, 2011.
- Eastman, C., Teicholz, P., Sacks, R., Liston, K. "Chapter 12: BIM for Owners and Facility Managers." In *In BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, and Contractors*, 93-147. New Jersey: John Wiley & Sons: Hoboken, 2008.
- FIEC. "Making BIM a global success." Brussels, 2017.
- Galliera, Ente Ospedaliero Ospedali. "Disciplinare di Gara." Genova, 2017.
- . *Il Nuovo Ospedale*. 05 2018. www.galliera.it/immagini.
- Hettinger, L., Riccio, G. *Visually induced motion sickness in virtual environments*. 1992.
- Iadanza E., Turillazzi B., Terzaghi F., Marzi L., Giuntini A., Sebastian R. "The STREAMER European Project. Case Study: Careggi Hospital in Florence." 2015.
- Johansson M, Roupe M, Viklund Tallgren M. *From BIM to VR, Integrating immersive visualizations in the current design process*. 2014.
- Johansson, M. *From BIM to VR- The design and development of BIMXplorer*. Gothenburg: Chalmers University of Technology, 2016.
- Kensek, Karen. "BIM Guidelines Inform Facilities Management Databases: A Case Study over Time." *Buildings*, 2015.

- Khanzode, A., Fisher, M., Reed, D. "Benefits and lessons learned of implementing building virtual design and construction (VDC) technologies for coordination of mechanical, electrical, and plumbing (MEP) systems on a large healthcare project." *ITcon Vol.13*, 2008: 324-342.
- La Franca, G. "Ospedali Galliera, Progettare la sostenibilità." *Tecnica Ospedaliera*, May 2016: 50-56.
- Library, NBS - National BIM. *NBS BIM object standard* . 05 2018.
<https://www.nationalbimlibrary.com/en/nbs-bim-object-standard/>.
- Manning, R., Messner, J. "Case studies in BIM implementation for programming of healthcare facilities." *ITcon Vol.13* , 2008: 446-457.
- Manuri, F., Sanna, A. "A Survey on Applications of Augmented Reality ." *ACSIJ Advances in Computer Science: an International Jurnal* , January 2016: 18-27.
- Microsoft. *Microsoft HoloLens*. 15 09 2018. <https://www.microsoft.com/en-us/hololens>.
- Milgram, P., Takemura, H., Utsumi, A., & Kishino, F. *Augmented reality: a class of displays on the reality-virtuality continuum*. 1995.
- Mohammadpour, A., Anumba, C.J., Messner J.I. "Retrofitting of healthcare facilities: case study approach." 2017.
- Oculus. *Oculus Rift*. 10 10 2018. <https://www.oculus.com/rift/>.
- Okada R.C., Simons A.E., Sattineni A. "Owner-requested changes in the design and construction of government healthcare facilities." 2017.
- Osello, A., Ugliotti, F.M. *BIM: verso Il catasto del futuro*. Turin, 2017.
- Poljanšek, Martin. *Building Information Modelling (BIM) standardization*. Joint Research Centre (JRC), 2017, 7.
- Spallarossa, E. "Bim for Italian Public Employers - Galliera Genoa Hospital Example ." *Solibri Journal*, June 2016: 40-42.
- Ugliotti, F.M. *BIM and Facility Management for smart data management and visualization*. Torino: Politecnico di Torino, 2017.
- University of Waterloo. *New Karolinska Hospital - Future of Construction*. 5 January 2017.
- World Economic Forum. "Shaping the Future of Construction: Future of Construction: A Breakthrough in Mindset and Technology." Geneva, Switzerland, 2016.