TECHNICAL UNIVERSITY OF CRETE, GREECE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

MusiCityX: Real-time 3D Interactive Application for Refining Urban Landscapes through their Musical Footprint

By
Maria Algiannaki

Thesis Committee
Associate Professor Katerina Mania (ECE)
Associate Professor Georgios Karistinos (ECE)
Associate Professor Panagiotis Parthenios (ARCH)

Chania, September 2020

Abstract

Rapid urbanization has led to growth of cities altering the urban scenery and contributing to the lack of cohesion. There are many elements that can be interpreted to contribute to the cohesion of an urban environment. In this research, we focus on the cohesion that the buildings relate among them. To manage these cohesion problems, we chose to use music. Music has been combined with many arts and many sectors of science.

Merging architecture with music is the main focus of this thesis, e.g. merging an art that is based on vision with an invisible to the eye art which is relying on the auditory sense. Taking advantage of music theory, we put forward a creative experience which enables composing and designing urban environments which communicate architectural features that would not emerge by using only the commonly-used, vision-based digital designing tools.

Initially, in this thesis, a sophisticated translation method was developed that connects architecture and music. The derived translation method parallels the parameters of a building affecting its connection with the urban environment, with the musical characteristics of a chord affecting its harmonized connection with the rest of the musical piece. Every chord represents a building. An extra note of the chord appears if there is a roof. The elements added to a building (windows, doors, balconies) create the melody that decorates every chord.

Subsequently, this thesis develops a multimodal interactive tool called "MusiCityX" implemented in Unity 3D that connects music with architecture integrating the translation method devised between musical and architectural elements. The MusiCityX platform is divided into four parts. In the first part, users build the road network of the urban scenery. In the second part, they can create buildings and add elements such as windows, balconies, and doors in 3D. In the third part, the users select the urban path to be translated to music. In the fourth part, they listen to the musical piece created.

Two Digital Audio Workstations (DAWs) were created to illustrate the musical theme resulted from the translation. The user can interact with the DAWs to listen to the music and alter the musical footprint. The alteration of music leads to the real-time alteration of the architectural parameter to which each musical note we change is connected to. This procedure will finally lead to a changed, potentially more aesthetically pleasing musical piece and an altered more cohesive urban environment.

Acknowledgements

Foremost, I would like to thank my supervisor, Associate Professor Katerina Mania for her guidance, support, and engagement throughout the process of the master thesis. I am also grateful for offering me the opportunity to the exciting topic of the thesis and trusted me to create not only a platform but also a new approach to connecting architecture with music.

Furthermore, I would like to express my appreciation to Assistant Professors Panagiotis Parthenios and Georgios Karistinos for their useful comments and support for the translation method. Their knowledge of the architectural and musical field respectively was of great importance and guided me through this work.

I would also like to thank my parents, Maria and George, and all of my friends, Katerina, Stefani, Anna, Onisiforos, Antonis, Myrto and Vaso for their support. I would not be able to create this thesis without my parents supporting my willingness to learn music and to study electrical engineering or without my friends studying with me late hours to keep me company while creating this thesis. I would also like to thank the theatrical team of the Technical University of Crete that helped me evolve, trust and use my creativity and imagination.

Contents

List of Figures

Chapter 1

Introduction

1.1 Theoretical Approach

Every day, we face ambivalent urban realities, which are generating a set of new urban typologies, different from the canonical city, having important impacts on urban life. Diversification of urban realities carries new problems and fragility for which it is necessary to find adequate answers. Many of these problems are strongly connected to the lack of cohesion - the architectural process of joining things together or the matter which forms them [1], [2], [3]. There are many elements that can be interpreted to contribute to the cohesion of an urban environment, but in this research, we focus more on the cohesion that the buildings have among them. To manage these cohesion problems we chose to use music. Music has been combined with many arts and many sectors of science.

Architecture is defined as the art and practice of designing and making buildings [4]. Design is an open-ended process of problem solving, bridging science and art [5]. Designers such as architects, employ analogue and digital design tools such as 3D models, drawings, sketches, etc., in order to design the functions of contemporary cities which is not the result of a single, coherent, decision making process [6]. Each tool provides an added level of vision to the architect. The commonly used visual, 3D digital design tools for urban environments often fail to capture the aesthetic as well as functional aspects of an urban design, mainly focusing on its geometrical properties [7]. Handling contextual and site information required to support urban design process is not often achieved [8].

Music is an art that we can not see, so many artists have tried to visualise the feelings of a musical piece. On the other hand music is mathematics. Music composition is basically a mathematical exercise and that is the reason that many scientists and engineers try to use music with their respective field. In music, just like in architecture, a musical part is a composition of elements. These structures can be a final independent musical piece, but they can also appear as parts of larger forms [9]. A musician and an architect will compose a piece through combining smaller composition of smaller pieces. A musician will use notes and other elements to create a measure and an architect will use concrete, bricks and/or other elements to create a building. The musician will use many measures to create a musical piece and the architect will use many buildings to create an urban landscape. Architecture and music are also two arts that are strongly connected to mathematics.

Efforts to introduce multimodal gamified elements in the design process of urban mass housing involving many stakeholders, are again, mainly centred on visual spatial design [10], [11]. Auditory interaction in architectural design is often limited to speech recognition [12], [13]. So, merging architecture with music, merging an art that is based on vision with an invisible to the eye art and taking advantage of every part of the music theory we can relate to our purpose, it can result to a creative experience composing urban environments with different aspects that would not result only through the vision-based digital designing tools.

1.2 Brief Description

This thesis describes an interactive tool developed at the Technical University of Crete, which was made in Unity 3D [14] that connects music with architecture. This tool offers the ability to the user to create an urban landscape and not only listen to the musical footprint of this landscape but also change the music and simultaneously watch the landscape change.

The user has the ability to build roads, to create buildings and to add windows, doors and balconies to them. Afterward, the user can choose as many roads as they desire to create a route that will be translated into music. Finally, if while listening to the music the user does not find it satisfactory then they can alter it. The alteration of music leads to the alteration of the architectural parameter to which the note we change is connected to. This procedure will finally lead to a changed musical piece and an also changed more cohesive urban environment.

The music translation method proposed in this thesis is concentrated on the cohesion of urban architecture. If we see every building as a block of different size and different color, the question is how these blocks are combined and do they create a cohesive outcome? To answer this question every building is being translated into a chord where its 1st factor (lowest note) is the musical translation of color, its 3rd factor (middle note) is the translation of the height of the ground floor, and its 5th factor (top note) is the translation of the number of floors of the building. The length of the chord is the length of the building's face. But, a building affects and is being affected by, the coherence of the road it belongs to. For example, a two-floor red building may coexist in harmony on a road with buildings with intense colors, but may also disrupt the harmony of another road. To be able to keep these relations between the buildings of the same road to the resulted music, every component that is being translated into a note of a chord is translated differently according to the other buildings of this road. For example, a two-floor building will be translated differently if it is among six-floor buildings than if it was among four-floor buildings. Also, the translated music mostly follows the music theory in order to show to the user where there are coherence problems but also keep the freedom of creativity and choice an art has.

Apart from these, the elements added to every building such as windows, doors and balconies are being translated into the same notes of the chord of

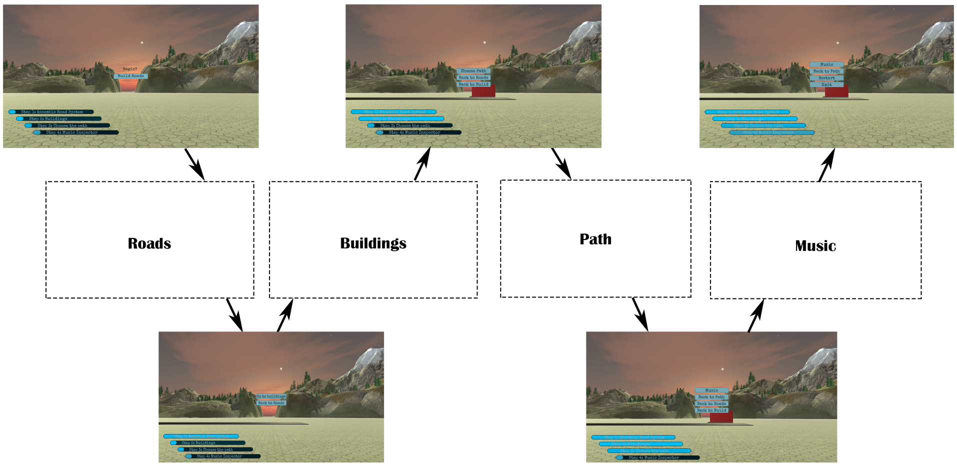


FIGURE 1.1: Subdivisions of the platform and in between menus

every building. The doors are translated into the 1st factor (lowest note), the balconies into the 3rd factor (middle note) and the windows into the 5th factor (top note) creating a melody playing on top of the chord. The length of the notes are defined by the length of the elements and the position of them are defined by the position of elements on the x-axis of the building's face, meaning if two windows are on two different floors but on the same exact position on the x-axis then their notes will be one on top of another creating a louder note.

Moreover, the program is subdivided into four parts. In the first part, the user creates the roads of the urban scenery. In the second part, they create the buildings and also add to the buildings all the elements they want (windows, balconies, doors). In the third part, the user must choose the roads that will create the path that our music translation will follow. And the final part is the music part, where the user listens to the musical piece resulted from the translation of the chosen path of the urban landscape created and they can make changes to result in a more harmonious musical and architectural composition.

On the subdivision of music, the main screen is divided into two parts and there are two cameras activated at the same time. The top part that covers 75% of the screen is used by the camera that allows the user to move and interact with their surroundings. The bottom 25% of the screen is used by the camera that shows the screen space - camera canvas that has the main Digital Audio Workstation (DAW) where the chords of the buildings of every part of the roads appear as white squares followed by their melody. The user can listen to the music translation of the path that they made through this DAW, but also they can change every building's position and length. To make other changes, the user has to click on the building they want to interact with. Then another DAW appears on the top screen which is the individual DAW of the chosen building. In this DAW the notes of the chord are shown separately and the notes of the melody are also there. The user can choose the note they want to change and buttons will appear to choose how they want to change this note. While changing the note they will notice the instant change of the corresponding architectural compound. The changes of the architectural components that can be made through this DAW are the number of floors, the height

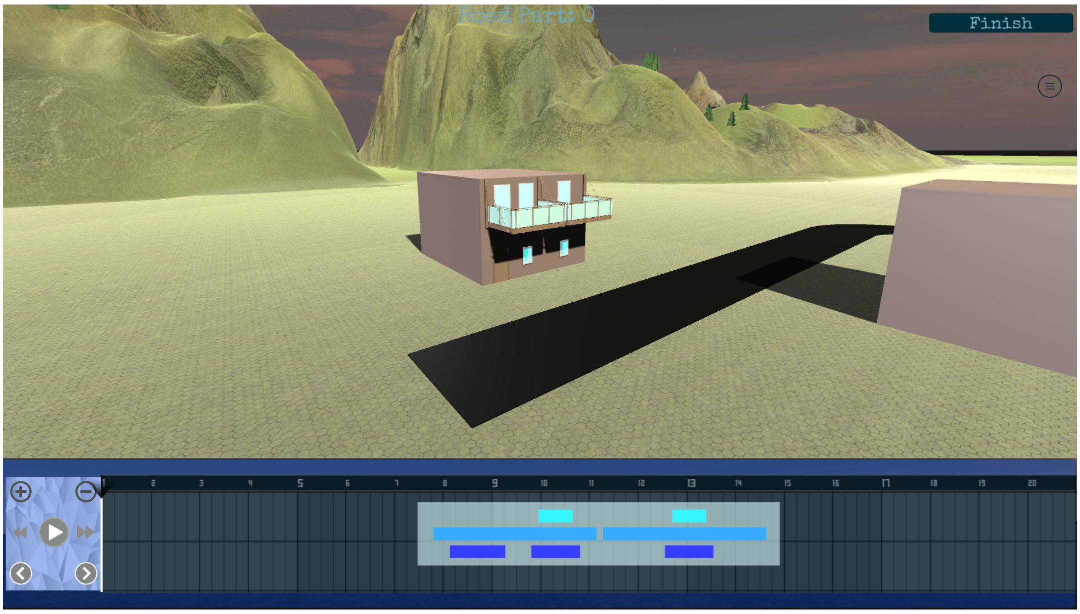


FIGURE 1.2: Music's subdivision initial layout

of the ground floor, and the color of the building, but also the placement of the elements (windows, doors, and balconies). The user can also delete or add notes and therefore elements to the building.

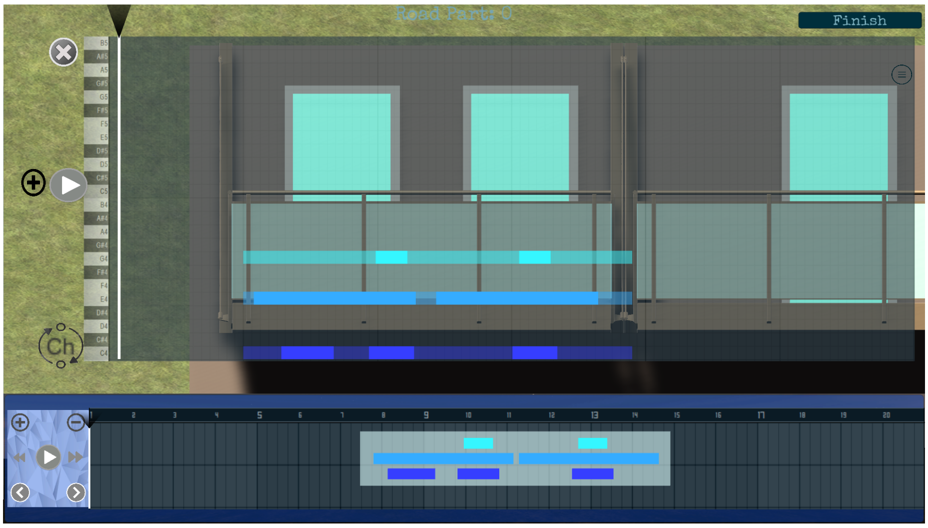


FIGURE 1.3: Initial screen of an individual DAW opened

1.3 Purpose of the Thesis

This thesis offers a tool to an urban designer that helps them to find and optimize cohesion problems in a more experiential than rational way, but also to get inspiration through using music to form urban landscapes. The tool focuses on the translation of the architectural parameters, that a passenger can observe into music, meaning the number of floors, the length of the face of the building, the height of the ground floor, and the position of windows, doors and balconies. So, this tool's purpose is not to find errors because a building can not be wrong by itself due to its height for example. This tool tries to represent the urban environment as a passenger observes it through music. The first purpose of this thesis is to make the user listen to the cohesion, listen to how every building is connected with the other, but also listen to how an impression a building offers can change depending on its environment. The second purpose is to offer the user a chance not only to improve these cohesion problems but also to give the inspiration to create a different complex of buildings through mostly listening and not watching. Architecture is an art that focuses on the sense of vision, but if you approach it through hearing, a sense that is rarely used in this context, then a new area of possibilities is created.

1.4 Structure of the Thesis

Chapter 1 introduces a brief description of the theoretical and practical approach of MusiCityX platform's functionality and purpose.

Chapter 2 analyzes the background used to develop this platform. The definitions and analysis of the architectural and music composition. It analyzes the music principles researched for this thesis, the reason why we connect music with other sectors, and the historical context of the relations between music and architecture. It focuses on the concept of composition in both music and architecture. It presents the technical tools, employed in this thesis such as the Digital Audio Workstation, the Unity engine, and all the other programs that helped the creation of this platform.

Chapter 3 presents the translation theory created to connect architecture with music. It starts with explaining the cohesion problems in urban design we manage to solve. It describes the objectives we set for this thesis but also the scope of application for the created platform. Afterward, it focuses on the theoretical approach of the translation. It explains how and why every architectural compound is being translated into music and vice versa while still maintaining the basic forms of chords defined by the music theory.

Chapter 4 describes in detail the technical implementation of the MusiCityX platform presented in this thesis. This chapter presents the final design of the platform's User Interface and the technical methodology which allowed the theoretical methodology of translation to be implemented in the system employing the technical tools described in Chapter 2. Moreover, we firstly present the general structure of the program that consists of four subdivisions, and then every subdivision is analyzed separately. For every subdivision, there are two parts. The first part explains the user interface of the subdivision and the second part focuses on the implementation techniques.

Chapter 5 describes the assessment that took place with architecture students and graduates that used the MusiCityX platform, evaluated it, and shared their opinions.

Chapter 6 provides the conclusions of this research as well as suggestions for future development.

Chapter 2

Background

2.1 Composition

The composition has many definitions. It is the nature of something's ingredients or constituents; the way in which a whole or mixture is made up or the action of putting things together; formation or construction. Another definition that applies to physics is the process of finding the resultant of several forces. Also, many definitions involve art, like the action or art of producing a creative work such as a poem or piece of music, or like the artistic arrangement of the parts of a picture [15].

There are so many definitions that apply to so many fields because everything around us is a composition of things (Fig. 2.20 by Pablo Carlos Budassi). The universe is a composition of heavy elements (0.03%), neutrinos (0.03%), stars (0.4%), gasses (4%), dark matter(22%) and dark energy(73%) [16]. To explain the composition of something we have to define every element composed of and the percentage of it. Another important parameter to define a composition is the arrangement of its elements. An important example is one of the smaller in scale known compositions, the composition of an atom. Experimental evidence indicates that, within an atom, a small nucleus, which generally contains both protons and neutrons, is surrounded by a swarm, or cloud, of electrons. The chemical characteristics of elements are intimately related to the number and arrangement of electrons in their atoms [17].

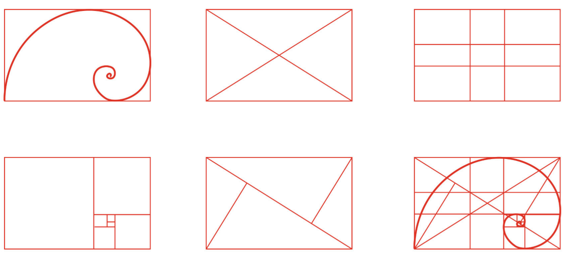


FIGURE 2.1: Composition guidelines

Composition in art follows the same definition but here, the person that creates the art is responsible for the composition. Henri Matisse said, "Composition is the art of arranging in a decorative manner the diverse elements at the painter's command to express his feelings." Composition involves arranging the visual elements with the guidance of the principles of art to promote the artist's statement. Unity is achieved only when the artist's statement is aligned with the composition. The principles are rhythm, balance, emphasis, gradation, harmony, variety, but also movement and proportion. There are a number of composition techniques used to help achieve unity in a painting, like simplification, rule of odds, rule of thirds, contrast, geometry, and symmetry or leading lines 2.1. Many artists have used the golden ratio to locate aesthetically pleasing areas to place their subjects. Although these techniques mostly have the word "rule" as their first name, they are considered guidelines, and for every "rule", there are many brilliant paintings that have broken it [18]. Also, the breaking of the rules of the composition is a guideline by itself that helps the artist to make the viewer feel uncomfortable and disoriented [19]. So, it is important to state that you can not define a piece's aesthetics by its composition, but following the composition guidelines, it can help you create something coherent.

This thesis uses the concept of composition as the key element of connecting architecture and music.

2.1.1 Architectural and urban composition

In architecture, Gaudet defines composition as "the combination of parts in a coherent whole" [2, p.158]. In architecture, the rules of composition in art mentioned earlier are applied. These rules are applied to every architectural composition level, from the sketches the architects create to present their idea to the urban designs. For example, by creating two adjacent parts different from one another, with different materials, colors, and textures, we can create a visual variety, excitement, and interest in the building. Furthermore, this can be achieved on one building or a group of buildings on a street scale. Phi, the Golden Section, has been used by humankind for centuries in architecture. The golden ratio was firstly used possibly by the Egyptians in the design of the pyramids. The Parthenon 2.2, was built in 447 to 438 BC, appears to use it in some aspects of its design to achieve beauty and balance its design. The ancient Greek Euclid ((365–300 BC) wrote of it in "Elements" as the "dividing a line in the extreme and mean ratio" [20]. Notre Dame, the Taj Mahal, the United Nations secretariat building, and Toronto's CN tower also contain the golden ratio in their designs.

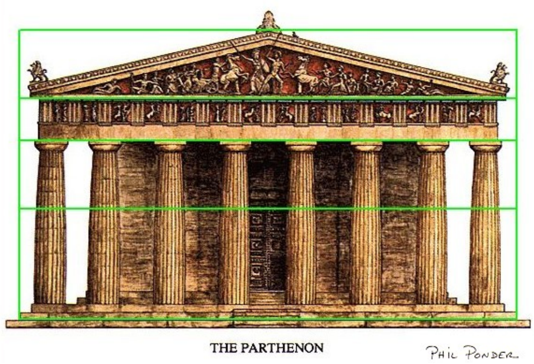


FIGURE 2.2: Parthenon and the golden ratio

An urban composition represents a form of the city where it gets a formal order so that the shape of any urban ensemble is not linked to a random phenomenon but to an intervention mastered and understood. In the matter of the city, the urban composition represents what the architectural composition represents for a building. This concept regarding the composition is common both to the architecture and to the city. The main property of the composition is that it transforms a possibly dispersed ensemble into a whole, resolving the contradictions that arise when the project's requirements and conditions are numerous. Spatial forms and urban compositions are built over time, longer than that of architectural composition [3].

In general, regarding the definition of composition, composition elements are assembled or organized to form a composed object, known as a unified object. A unified object (a composition) contains various connected elements (parts); each can be allocated to a different composition layer. From the scale point of view, the elements belong to a smaller scale, and the unified object is located on a larger scale. Thus, each composition has two main layers, the layer of the elements on a smaller scale and the layer of the unified object on a larger scale. Each composition element might be considered as a unified object of other smaller-scaled elements. Likewise, a unified object can also be a component of another larger-scale composition. The composition has chain-like composition elements and unified object, possessing a fractal essence [21]. In other words, the building is a composition of elements, and also it is an element of the composition of the urban landscape that is a part of.

2.1.2 Music composition

Aristotle knew music as one of the branches of mathematics, and Islamic philosophers like Avicenna, who has mentioned music in mathematics as part of The Book of Healing, has accepted this opinion too. Unlike mathematics, all the musical features are not absolute and unchangeable. Instead, the composer's ability and taste directly involves in it; therefore, they also call it art [22].

At its most fundamental level, the act of composition involves the ordering of pitched sounds in musical time and space. Pitch relationships are referred to as intervals; their specific occurrence in musical time is determined by rhythm, a concept that embraces all durational aspects of music. Rhythm, in turn, may or may not be regulated by a meter. In metrically organized rhythm, recurring patterns of accented and unaccented "beats" furnish a durational substructure that necessarily affects all the other composition elements, including the nature of melody, harmony, and texture. Metrical rhythm is nearly always present in dance music because its patterning is largely analogous to bodily motions and step figurations. Nevertheless, logogenic, or word-determined, music also often employs metrical patterns, corresponding as a rule to those of the poetic text [23].

The combination of form and motive is what describes the way a musical piece is composed. The term form is used in several different senses, depending on what it describes. When used in the connection of *binary*, *ternary*

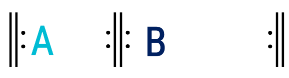


FIGURE 2.3: Binary form

or *rondo forms*, it refers chiefly to the number of parts because they are the simplest forms and they do not need more explanation. For example, the binary form describes the structure of a piece of music, which is divided into two different sections. The two sections are usually labeled A and B so that the form will be A-B 2.3.

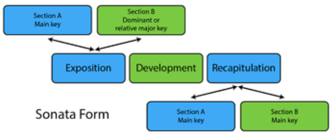


FIGURE 2.4: Sonata form

The phrase *sonata form* suggests, instead, the size of the parts and the complexity of their interrelationships. The reason why is that the sonata form is a musical structure consisting of three main sections: an exposition, a development, and a recapitulation, so their connections can vary 2.4 and it is crucial to define them to know their detailed composition.

In speaking of *minuet*, *scherzo* and other *dance forms* one has in mind the metre, tempo and rhythmic characteristics which identify the dance. A minuet is a social dance of French origin for two people and its form is usually a minuet, a trio and then a minuet 2.5. Used in aesthetic sense, form means that a piece is organized; i.e. that it consists of elements functioning like those of a living organism. Without organization music would be an amorphous mass, as unintelligible as an essay without punctuation, or as disconnected as a conversation which leaps purposely from one subject to another. The chief requirements of a comprehensible form are *logic* and *coherence* [9, p.1].

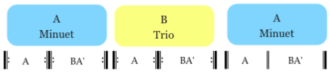


FIGURE 2.5: Minuet form

The concept of motive should produce unity, coherence, relationship, logic, comprehensibility, and fluency. The basic motive is often considered as the 'germ' of the idea [9, p.10]. The right connection of motive and form will result to a composition of a well-balanced piece.

In music, just like in architecture and everything else, a musical part is a composition of elements, and each element can be a composition of simpler elements, but also, this musical part can be an element of a composition of a bigger part. The theme is defined as any element, motif, or a small musical piece that has given rise to some variation becomes thereby a theme. The smallest structural unit in music is the *phrase*, a kind of a musical molecule consisting of a number of integrated musical events, possessing certain completeness, and well adapting to combination with other similar units[9, p.3]. A complete musical idea or theme is customarily articulated as a *period* or a *sentence*. These structures can be a final independent musical piece, but they also can appear as parts of larger forms [9, p.20].

Music theory

Music Theory is the study of music and its organizational characteristics.We learn to associate sound with symbol (or notation) to increase our ability to perceive music at levels of increasing depth and comprehension, both experientially and analytically. Examining music involves evaluation. At any level of experience, this begins with personal taste and preference. As we learn more, our method of evaluation evolves, becoming both more fluent and more critical. Clearly, over time and study, this shapes our perceptions and our tastes, regardless of what general direction these may take.Two broad approaches are employed in the study of music:

Analysis: we learn to employ commonly accepted techniques and specialized language to describe musical organization. These techniques are a shared analytical language throughout the community of musicians. Analysis is conceptual knowledge and evaluation.

Composition: either by actively creating our own works, or (more likely for the beginner), imitating or emulating earlier composers' works. Composition is active knowledge and procedure [24, p.8].

Music theory is a bit like grammar. Languages are invented by the people who speak them, who tend to care more about what is comfortable and what makes sense than about following rules. Later, experts study the best speakers and writers in order to discover how they use the language. These language theorists then create rules that clarify grammar and spelling and point out the relationships between words. Those rules are only guidelines based on patterns discovered by the theoreticians, which is why there are usually plenty of "exceptions" to every rule. Attempts to develop a new language by first inventing the grammar and spelling never seem to result in a language that people find useful. Music theory, too, always comes along after a group of composers and performers have already developed a musical tradition. Theoreticians then study the resulting music and discover easy ways of explaining it to the audience and other composers and performers[25, p.1].

People were talking long before they invented writing. People were also making music long before anyone wrote any music down. But written music is useful, for many of the same reasons that written words are useful. Music is easier to study and share if written down [25, p.3]. And for this purpose, the music theory is needed.

Music theory is used in this thesis to translate every building into a chord that follows the "rules" of the music theory.

Sound Waves

Music is sound that is organized by people on purpose, to dance to, to tell a story, to make other people feel a certain way, or just to sound pretty or be entertaining. Music is organized on many different levels.

Sounds can be arranged into melodies, harmonies, rhythms, textures and phrases. Beats, measures, cadences, and form all help to keep the music organized and understandable. However, the most basic way that music is organized is by arranging the actual sound waves themselves so that the sounds are interesting and pleasant and go well together . To get the melodic kind of sounds more often associated with music, the sound waves must themselves be organized and regular, evenly-spaced, not random mixtures 2.6 [25, p.95].

The closer together those evenly-spaced waves are, the higher the note sounds. Musicians define this relationship of the waves as pitch of the sound, or name

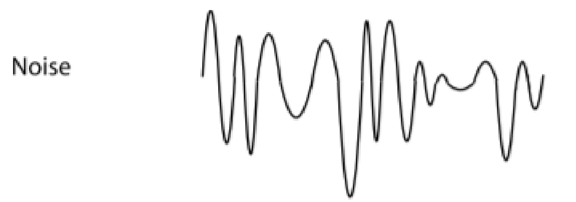


FIGURE 2.6: Noise vs. Tone

of specific notes, or tuning. Scientists and engineers, on the other hand, talk about the frequency and the wavelength of the sound.

In this thesis, every architectural element is translated into a note, a sound wave. When the note changes, the architectural element change accordingly. The arrangement of these sound waves will define the coherence of the musical piece.

Scales

A harmonic series 2.7 can have any note as it's fundamental, so there are many different harmonic series. Nevertheless, the relationship between the frequencies of a harmonic series is always the same. The second harmonic always has exactly half the wavelength (and twice the frequency) of the fundamental; the third harmonic always have precisely a third of the wavelength (and so three times the frequency) of the fundamental, and so on [25, p.111].

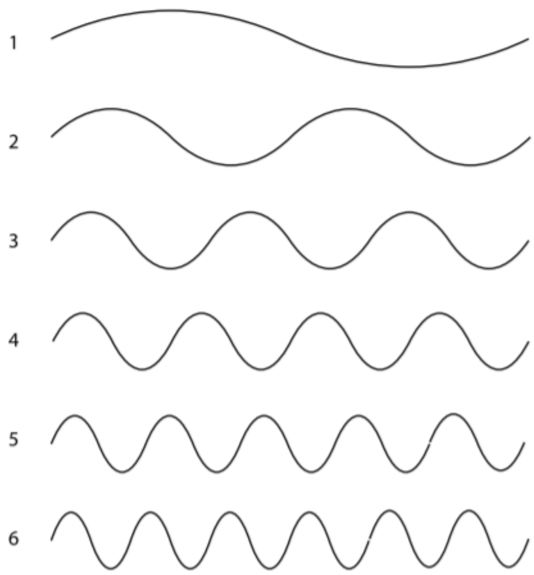


FIGURE 2.7: Harmonic Series Wavelengths and Frequencies

The scales and harmonies of most of the world's kinds of music are based on these physical facts. A scale is an ordered sequence of notes. For example:

do, re, mi, fa, sol, la, ti, do repeating this cycle. We could use this same sequence (major scale) starting from a note that was not C, but for example, G. The scale would then be: sol, la, ti, do, re, mi, fa#, sol. These scales are called the "major scales. " There are also "minor scales" and the difference is the distance between the notes 2.8 [26].

Intervals	Tone	Tone	Semitone	Tone	Tone	Tone	Semitone	
C major scale	C	D	E	F	G	A	B	C
D major scale	D	E	F#	G	A	B	C#	D
E major scale	E	F#	G#	A	B	C#	D#	E
F major scale	F	G	A	Bb	C	D	E	F
G major scale	G	A	B	C	D	E	F#	G
A major scale	A	B	C#	D	E	F#	G#	A
B major scale	B	C#	D#	E	F#	G#	A#	B

FIGURE 2.8: Tones and semitones

For the major scales, the sequence is tone, tone, semitone, tone, tone, tone, semitone. For the minor scales, the sequence will be tone, semitone, tone, tone, semitone, tone, tone. C minor scale, for example, is: do, re, re#, fa, sol, sol#, la#, do repeating the cycle.

In this thesis, the building's color defines the scale of the chord that represents this building.

Chords

Chords 2.9 in Western music, most harmony is based on chords. They are groups of notes built on major or minor triads. In traditional triadic chords, there are always at least three notes in a chord (there can be more than three), but some of the notes may be left out and only "implied" by the harmony. The notes of the chord may be played at the same time (block chords), or may overlap, or may be played separately but in a quick enough succession that they will be "heard" as a chord (arpeggiated chords) [25, p.84].

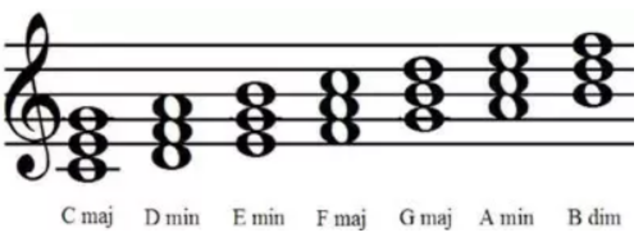


FIGURE 2.9: Chords

Every building, in this thesis, is translated into an at least triadic chord.

DAW

DAW 2.10 [27] stands for Digital Audio Workstation. It's a software program for making music: recording, editing, mixing and manipulating sound[28]. It is a relatively new platform for composing music.

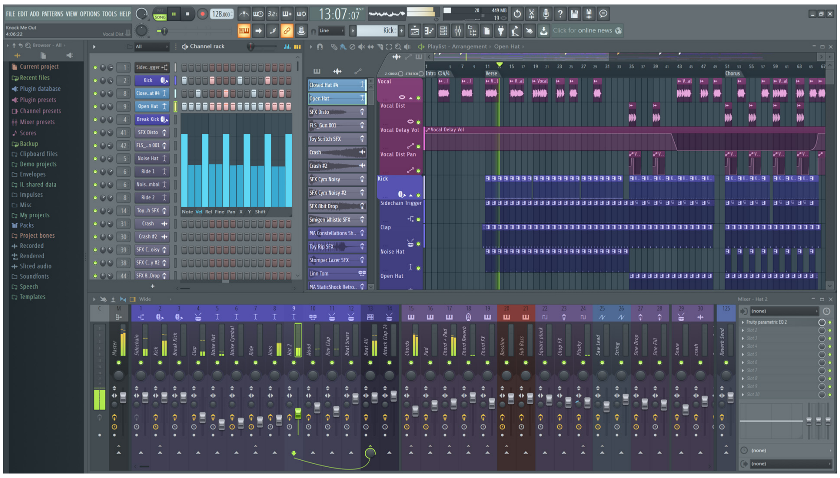


FIGURE 2.10: Digital Audio Workstation

DAW is an electronic device or application software used for recording, editing, and producing audio files. DAWs come in a wide variety of configurations, from a single software program on a laptop to an integrated stand-alone unit, all the way to a highly complex configuration of numerous components controlled by a central computer. Regardless of configuration, modern DAWs have a central interface that allows the user to alter and mix multiple recordings and tracks into a final produced piece.DAWs are used to produce and record music, songs, speech, radio, television, soundtracks, podcasts, sound effects, and nearly any other situation where complex recorded audio is needed [29].

DAw is used in the MusiCityX platform as the medium that the user can see, listen, or interact with the resulted music.

2.2 Connections of music with other sectors

Music has been combined with many arts and many sectors of science. Music, on the one hand, is an art that we can not see, so many artists have tried to visualize the feelings of a musical piece. On the other hand, music is mathematics. Music composition is basically a mathematical exercise, and that is the reason that many scientists and engineers try to use music with their respective fields.

At first, one of the oldest connections that involve music started in the 1700s and are connections between music and color 2.11. Newton believed in "a real analogy between elementary colors and the notes of the musical scale." Newton's prism experiments famously established the ROYGBIV rainbow of colors, which combine into white light—red, orange, yellow, green, blue, indigo, and violet. But for Newton, the seven colors also corresponded to the seven notes of the musical scale [30].

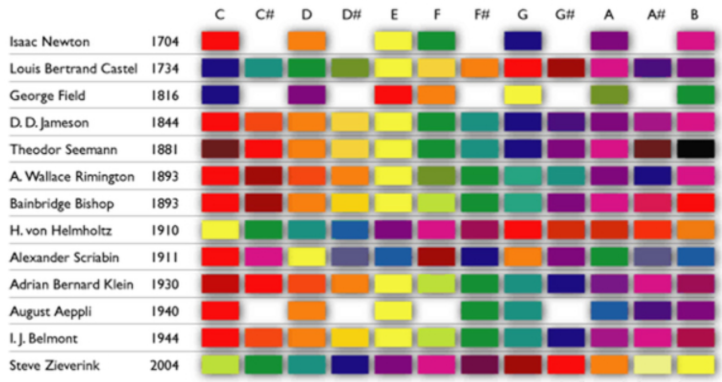


FIGURE 2.11: Connections between color and music

A very known and interesting example of a connection of music with another art is Kandinsky's work. Music - and the idea of music - appears everywhere in Kandinsky's work. The titles of his artwork were Compositions, Improvisations, and Impressions. The Impressions, although this may seem less of an obviously musical title, we know that several of them were specifically written in response to the experience of hearing particular pieces of music. He believed shades resonated with each other to produce visual "chords" and had an influence on the soul. To support his color theories, Kandinsky appealed in his manifesto to the evidence of synaesthesia, the scientific name for the condition in which the senses are confused with one another [31].

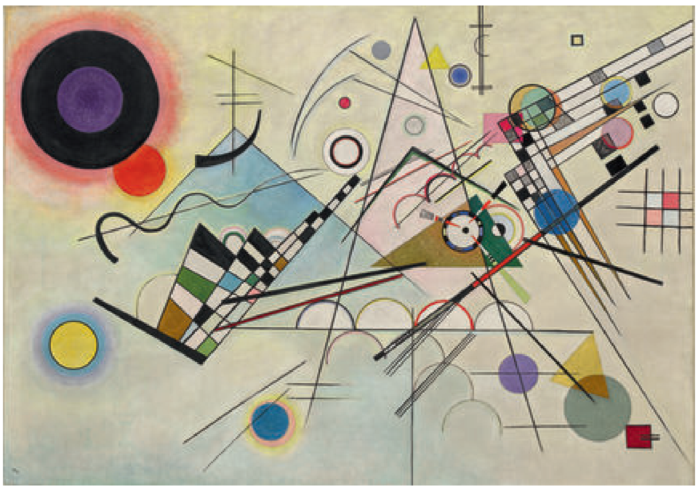


FIGURE 2.12: Composition 8 by Kandinsky

In a surprising marriage of science and art, researchers at MIT have developed a system for converting the molecular structures of proteins, the basic building blocks of all living beings, into an audible sound that resembles musical passages. Then, reversing the process, they can introduce some variations into the music and convert it back into new proteins never before seen in nature. The whole concept, Buehler explains, is to get a better handle on understanding proteins and their vast array of variations. By translating that language into a different form that humans are particularly well-attuned to, and that allows different aspects of the information to be encoded in different dimensions — pitch, volume, and duration — Buehler and his team hope to glean new insights into the relationships and differences between different families of proteins and their variations and use this as a way of exploring the many possible tweaks and modifications of their structure and function. As with music, the structure of proteins is hierarchical, with different structure levels at different scales of length or time [32], [33]. The new method translates an amino acid sequence of proteins into this sequence of percussive and rhythmic sounds 2.13.

FIGURE 2.13: Orchestra Of Amino Acids [34]

The team then used an artificial intelligence system to study the catalog of melodies produced by a wide variety of different proteins. They had the AI system introduce slight changes in the musical sequence or create entirely new sequences and then translated the sounds back into proteins that correspond to the modified or newly designed versions. With this process, they were able to create variations of existing proteins — for example, one found in spider silk, one of nature's strongest materials — thus making new proteins unlike any produced by evolution [32].

In conclusion, music is a compelling art that exists from the start of our civilization and affects our feelings and helps us evolve our interpretation of the world. The Pythagoreans were probably the first westerners at it when they declared: "The eyes are made for astronomy, the ears for harmony, and these are sister sciences" [31]. And they were right, because music is connected with everything around us.

2.3 Historical context of the relations between music and architecture

Just as the composer creates notes from his mind in a harmonious balance such that they become soothing sound waves to the hearer, so does the architect with form and space translating it to a livable experience [35].

"Music is liquid architecture; Architecture is frozen music". The first dictum, "Music is liquid architecture," is attributed by Edward Howard Griggs to Robert Browning (in his poem "Abt Vogler") and to John Ruskin [36]. "Architecture is frozen music" is attributed to Friedrich Schelling, who believed that "architecture, as the music of the plastic arts, necessarily follows the numerical relations, provided that it is music in space - however, in a sense, it is a frozen music".

Goethe in 1829 said to Johann Peter Eckermann that he found one of his writings where he mentioned architecture as "frozen music" and continued saying, "Indeed there is something important in this: the effect of architecture on us reassembles that of music [37, p.29]. All these people were philosophers, writers, poets. However, an architect was the first to express the idea of a connection between music and architecture. In fact, 100's of centuries before Goethe's existence, Roman architect and author Vitruvius, in the first of his famous "Ten Books of Architecture," expressed that the architect should have a knowledge of music in order to grasp mathematical relations. Ancient Greek Architecture was the epitome of this idea. They were, in fact, lovers of music and built their temples such that columns were placed at the same interval to resonate harmonic progressions [35].

During the sixteenth century in Italy, Andrea Palladio spent a great deal of his life re-interpreting the ancient measurements from the architectural treatise of the first-century Roman architect Vitruvius [38]. Palladio wrote:" The proportions of the voices are harmonies for the ears; those of measurements are harmonies for the eyes." [39]. Palladio based the design of his works and especially his residences on a variety of analogies derived from music. There are many searches concerning the integration of musical analogies in Palladio's architectural work. On the other hand, there are doubts about the systematic and conscious application of the musical analogies in the whole of his work[37, p.25].

Markos Novak [40] invents the term 'archimusic', describing the art and science resulting from the conflation of architecture to music. Novak's notion that architecture and music are freed from their limitations of matter and sound, respectively, is in reference to Iannis Xenakis' view of music 'inside-time' and music 'outside-time' [41]. Iannis Xenakis, who worked with Le Corbusier since 1947 as his assistant, adopted a unique approach to architecture and music [37, p.43]. At the same time, while working for Le Corbusier, Xenakis was studying harmony and counterpoint and composing. He worked long and hard, frequently far into the night, and sought guidance from a number of teachers, most of whom, however, ultimately rejected him. He then approached a French musician by the name of Olivier Messiaen. Messiaen, instead of teaching him harmony and counterpoint he advised him to take advantage of his knowledge and heritage. He said, "No, you are almost thirty, you have the good fortune of being Greek, of being an architect and having studied special mathematics. Take advantage of these things. Do them in your music." So, Xenakis started evolving his unique way of composing music through the use of mathematics. Xenakis says that "The composer uses sounds that he arranges in time to compose melodies. This arrangement in time is the basis of the quantum statics of Einstein, Fermi, and Dirac. So we can distinguish the field of quantum mechanics through the field of sounds and find knowledge" [42, p.202].

Le Corbusier assigned Xenakis to design the Phillips Pavillion. Xenakis used drawing of Le Corbusier as starting point and concluded to a three-dimensional form based on the transformation of the musical designs of *Metastaseis* 2.14 (the musical piece he was working then).

In conclusion, music and architecture have been compared and combined many times, with totally different ways. There are two forms of art that include

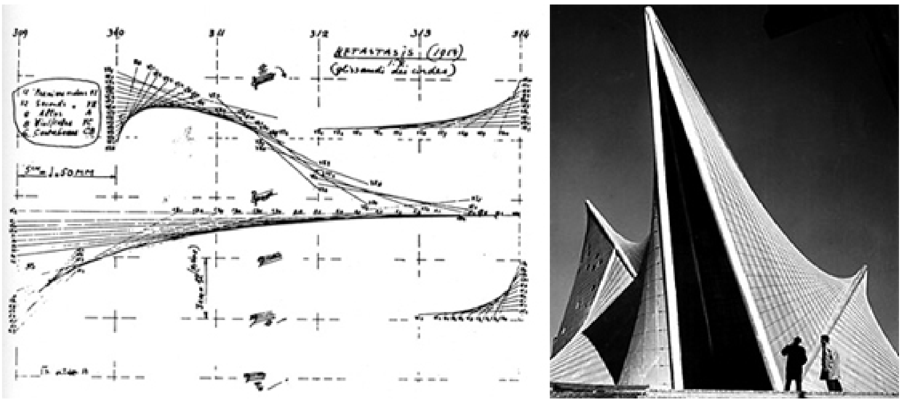


FIGURE 2.14: Metastaseis and Phillips Pavillion

mathematics and personal expression and taste, so their combinations can vary. Someone can combine them in a more abstract level like Palladio, or in a more scientific way like Xenakis. Nevertheless, even though Xenakis used mathematics to this combination of arts he also involved his personal expression. Even if it was based on mathematical or architectural forms, Xenakis's music was chaotic and strong to the ear and this was due to his experiences from war. The result is that the combination of music and architecture is also an art and every art involves a freedom to the personal expression, so there is no wrong combination of music and architecture, there are just different kind of arts created through every of these aspects of connection.

2.4 Earlier research on the subject

Another thesis on the connection of architecture and music took place on the Technical University of Crete by Fotis Giariskanis and the same committee. This thesis presented a platform 2.15 implemented in the Unity development engine that offered sonification of an Urban Virtual Environment (UVE), simulating a real-world cityscape and offering visual interpretation and interactive modification of its soundscape. The system presented in this thesis offered: First of all, the ability to view and convert an urban street to music (ready to play) based on a specific grammar of converting architectural elements to musical elements, secondly, the ability to transform this music in order to harmonize it based on musical rules and finally, the prospect of converting back the aesthetically and harmonically "corrected" musical piece to a newly refined street or urban design [43].

The Digital Media Lab from the School of Architecture, Technical University of Crete, has developed a translation method between music and architecture for compositional experiments on urban design. According to this method, different urban and architectural elements that define streetscapes are converted into musical notes based on their geometrical properties (x-y-z position, size, etc). This method was used to this thesis [44].

This earlier thesis created a tool that translated a single road and every architectural element was translated into a single note. In this thesis we started

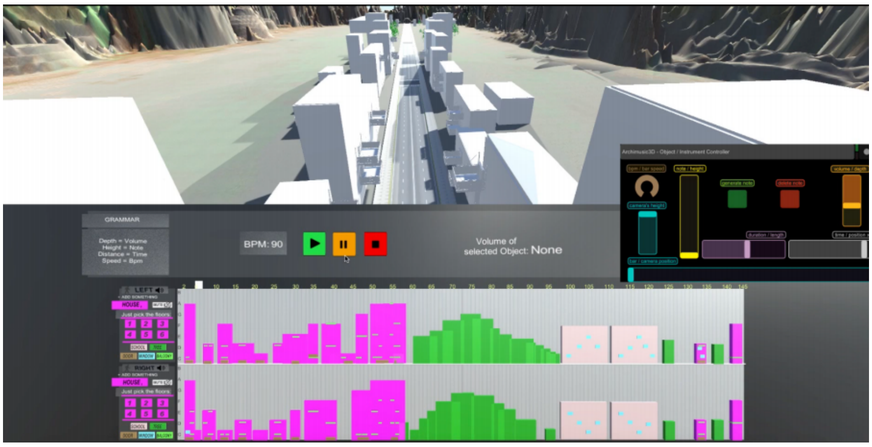


FIGURE 2.15: View of the program

from the same concept of creating a tool that translates architectural elements to music. However, everything from the translation method to the creating of the platform was anew created.

2.5 Software

In order to create this thesis's platform, the use of different programs was mandatory. Unity was the programming platform used to develop MusiCityX. FL Studio was used to produce the needed sounds and FMOD to connect these sounds to the platform. Finally, Revit was used to acquire some architectural elements.

2.5.1 Unity

Unity 2.16 is a cross-platform game engine developed by Unity Technologies to create three-dimensional, two-dimensional, virtual reality, and augmented reality games, as well as simulations and other experiences.

FIGURE 2.16: Unity platform for 3D interactive applications

The platform offers a primary scripting API in Csharp, which is the primary programming language used for the engine. It also offers visual tools for easy tasks such as importing geometry based on drag and drop functionality, mainly meant for non-technical users[14]. All the functionalities of the MusiCityX tool, creating the roads, the buildings, choosing a route, as also the visualization of the DAW are implemented into Unity through code.

2.5.2 FMOD

FMOD 2.17 is a proprietary sound effects engine and authoring tool for video games and applications developed by Firelight Technologies, that play and mix sounds of diverse formats on many operating systems.

FIGURE 2.17: FMOD sound effects engine

FMOD has been integrated into many video game engines, one of them being Unity.FMOD for Unity is a tool that builds adaptive audio that you can hear it in-game with the FMOD audio engine[45].In this thesis, FMOD is used as a tool to transfer the notes used for the music translation, into the Csharp scripts and to listen the given output.

2.5.3 FL Studio

FL Studio (until 2003 known as FruityLoops) 2.18 is a digital audio workstation (DAW) developed by the Belgian company Image-Line.It features a graphical user interface with a pattern-based music sequencer. It can also be used as a Virtual Studio Technology (VST) instrument in other audio workstation programs and also functions as a ReWire client. Apart from that it offers its own VST instruments and audio applications[46] [27].

FIGURE 2.18: FL Studio digital audio workstation

2.5.4 Revit

Autodesk Revit 2.19 is a building information modelling (BIM) software for architects, landscape architects, structural engineers, mechanical, electrical, and plumbing engineers, designers and contractors, because it includes features for architectural design and structural engineering as well as construction. The original software was developed by Charles River Software, founded in 1997, renamed Revit Technology Corporation in 2000, and acquired by Autodesk in 2002.

FIGURE 2.19: Autodesk Revit

The software allows users to design a building and structure and its components in 3D, annotate the model with 2D drafting elements, and access building information from the building model's database. Revit is 4D building information modeling capable with tools to plan and track various stages in the building's life cycle, from concept to construction and later maintenance and/or demolition[47].

FIGURE 2.20: A vertical logarithmic scale of the universe

Chapter 3

Translation

3.1 Introduction

This part of the thesis focuses on the theoretical approach of translation from architecture to music and vice versa. Firstly, there is an explanation of the problems this translation will try to find solutions to, the objectives we decided to set for this application, and how this platform with this translation can help the user. After defining these parts, then the method of translation is explained. This translation method is developed in a way that parallels the parameters of a building affecting the difficulty of its connection with the urban environment with the musical characteristics of a chord affecting its harmonized connection with the rest of the musical piece. So, every chord represents a building, more specifically, a building's number of floors, the height of the ground floor, and color. It also considers the elements added to a building like windows, doors, and balconies and translates them into the melody that decorates and beautifies every chord.

3.2 Cohesion problems in urban design

Every day, we face ambivalent urban realities, which are generating a set of new urban typologies, different from the canonical city, having essential impacts on urban life. Diversification of urban realities carries new problems and fragility for which it is necessary to find adequate answers. Many of these problems are strongly connected to the lack of cohesion. Cohesion can be semantically understood as "the action or effect of meeting or joining things together or the matter which forms them"[1]. Using this terminology, there are many elements that can be interpreted to contribute to the cohesion of an urban environment, but in this research, we focus more on the cohesion that the buildings have among them. If we see every building as a block of different size and different color, the question is how these blocks are combined, and do they create a cohesive outcome?

Answering this question, there are many problems to this type of cohesion because of the uneven urban development, as we see in Figure 3.1 where on the right there are one floor abandoned buildings, and on the left, there are newer buildings with 3 or 4 floors. Furthermore, when a city is growing, older buildings are still in place, the urban designer undertakes the difficult task to create an urban landscape where the newer buildings cover the needs of the residents and coexist with the already existing ones. This task is not easily achieved and can affect the cohesion of a road in a significant way.

FIGURE 3.1: Example of uneven urban development

Another massive problem to the cohesion is the cohesion between adjacent roads, which belong to different neighborhoods. To achieve cohesion, one of the goals is diversity – concerning the capacity of a territory to accommodate different profiles of users (social classes, age groups, cultures, etc.)[1]. But, when diversity is not the case, the neighborhoods develop differently depending on the average family income in this neighborhood, and when these neighborhoods are nearby, the cohesion of the area is non-existent. For example, in the USA fifty years ago, the difference between a wealthy neighborhood and a low-income neighborhood was smaller than it is today [48]. As this difference augments, so does the cohesion problems.

Finally, another problem that often appears to modern urban architecture is the placement of malls or other non-residential buildings without taking into notice the urban environment around them. These buildings often create a problem of cohesion when they don't blend into their environment, like, for example, the building in Figure 3.2.

FIGURE 3.2: Cohesion problem in Pireus

3.3 Goals and Objectives of this translation method

This approach wants to achieve a connection between architecture and music that can give the user the ability to listen to the cohesion problems. These cohesion problems are not always apparent, except in extreme cases, and the way every architect wants to handle them depends on their taste and perception. So, this thesis goal is not to offer a tool that solely defines what is right and what is wrong, but to give a chance to the user to find different musical solutions for the same path. To do that, the translation method must follow the music theory to create the translated chords to assure that no chord is wrong by itself. Simultaneously, the translation must have the ability to show the difference between a small cohesion problem to an extreme one. So, sometimes the music generated must be strongly disharmonious to represent these extreme cases, like, for example, a road full of two-floors and one-floor buildings with one nine floor building between them. Also, another objective of this translation method is to offer a chance of creativity to the designers through its range of possible harmonious compositions.

Consequently, the objective of this translation is to be able to show cohesion problems while maintaining the subjectivity that the urban architectural design as an art offers.

3.4 Scope of application

This tool can be the medium for many different applications. First of all, the user can study the cohesion of one single road through its musical footprint and navigate through different answers to this problem while changing the music.

Furthermore, another application that this tool can be used is to help the designer to find the best parameters of a corner building to fit in cohesively to both roads that are part of. Every corner building translates into two chords. Every chord translates one side of the building. Every side of the building belongs to a different road, so their resulted chords while differ. At the same time, while changing one chord, the other changes, respectively. So, you can use this tool to find the best dimensions of a building that creates cohesive musical pieces on both roads. In addition, this tool can be used to add windows, balconies, and doors to a building in a more refined way, look for interesting combinations while listening to the music to not overload the piece. Apart from all these, this tool can be used to just help the designer find some inspiration while experimenting with different sequences of chords, thus buildings. Also, sometimes just like on a painting, if you want the viewer to feel uncomfortable and disoriented, you can break the rules of composition to achieve that [19]. On an architectural level, a designer may want to add a building that is very different from its surroundings and "disrupts" the cohesion purposely and this tool can also help to define the main characteristics of this building through trying to achieve a disharmonious piece.

Finally, using a combination of all these, the user can study and change a musical translation of a path a passenger may take, and this will offer them the ability to interact with the urban scenery in a more experiential way like the passenger sees it.

3.5 Methodology

In architecture, as well as in music theory, the harmony of the composition does not depend on rules that define which type of chord or which type of building is right to create, but on the connection of them. For instance, a two-floor building has nothing wrong, but on a street full of eight-floors and seven-floor buildings, something will seem off. Therefore, the translation method is created in a way that if there is the same building on two distinct roads, it will be translated differently depending on the other buildings of each road.

1		2		3	4		5		6		7+
8	9-	9	9+		11	5-		5+	13		
do	do#	re	mib	mi	fa	fa#	sol	lab	la	sib	si
do#	re	mib	mi	fa	fa#	sol	lab	la	sib	si	do
re	mib	mi	fa	fa#	sol	lab	la	sib	si	do	do#
mib	mi	fa	fa#	sol	lab	la	sib	si	do	do#	re
mi	fa	fa#	sol	lab	la	sib	si	do	do#	re	mib
fa	fa#	sol	lab	la	sib	si	do	do#	re	mib	mi
fa#	sol	lab	la	sib	si	do	do#	re	mib	mi	fa
sol	lab	la	sib	si	do	do#	re	mib	mi	fa	fa#
lab	la	sib	si	do	do#	re	mib	mi	fa	fa#	sol
la	sib	si	do	do#	re	mib	mi	fa	fa#	sol	lab
sib	si	do	do#	re	mib	mi	fa	fa#	sol	lab	la
si	do	do#	re	mib	mi	fa	fa#	sol	lab	la	sib

FIGURE 3.3: Scales

The translation method contains the translation of the buildings and their added elements. The translation of the buildings consists of chords that describe the building as a whole. There is also a melody of notes for every building that represent the added elements, meaning windows, doors, and balconies, all played by a piano. Also, if there is a roof, an extra note same as the 1st note of the chord is added, which is played by a violin.

1	2	3	4	5	6	7	8	9	10	11	12	No.
1				5			8					major
1			4				8					minor
1				5				9				augmented
1			4			7						diminished
1		3					8					2nd
1					6		8					4th
1							8					5th
1				5			8			11		7th
1				5			8				12	major 7th
1			4				8			11		minor 7th
1				5				9		11		augmented 7th
1				5			8		10			major 6th
1			4				8		10			minor 6th

FIGURE 3.4: Combination of notes to form a chord

3.5.1 Chord Building

To start with the chords, every chord has to define the geometrical properties of a building that affects the urban environment and the view of a passenger, but also its color. Furthermore, to define what properties to translate, we have to consider which of those properties are incorporated into the composition of a street. For example, the amount that the building expands in depth does not affect the cohesion of the architectural elements of a street because it can not be seen from the street. However, the amount the building expands lengthwise of the road, along with the number of floors, are two pieces of data that affect their surroundings. In addition, the height of the ground floor also affects the experience of the observer and, therefore, the urban design. In conclusion, the building will be translated through four elements, color, the height of the ground floor, number of floors, and width. To be able to occur the optimal matching of the architectural elements to chords, we have to notice the characteristics of the chords and how they progress (ex. from a major to a minor).

As we see in the Figure 3.4, for every note there is a variety of chords. A basic chord embodies three factors. The 1^{st} factor of the chord is the root note and defines only the scale. The 3^{rd} usually decides whether the chord is major or minor and sometimes suspended (ex. for C_{major}: 1^{st},3^{rd},5^{th}, C_{minor}: (1^{st},b3^{rd},5^{th}), C_{sus2}: (1^{st},2^{nd},5^{th}) C_{sus4}: (1^{st}, 4^{th}, 5^{th})). Also, if the 5^{th} factor is a sharp 5^{th} note or a flat 5^{th} note it can make a chord augmented or diminished accordingly. After the 5^{th} there can be more more factors (for example the 9^{th} chord). Thereupon, we have four music elements, three factors and the length of the chord.

Color

Although every research that dealt with the relation of color and music paired notes with colors, no one did it for architectural purposes. If we wanted to translate a painting into music or vise versa, we would also pair colors with notes. Nevertheless, in architecture, the color is not one of the main elements like the note is in music. To find the relationship that is needed, we go back to the chord table. According to this, after the chord's root note ,is defined, the other notes affect how the chord will sound. So, the root note does not impact a single chord, but it affects the harmony of a piece that includes many different notes. The corresponding concept on urban architecture is the color because it does not impact a single building neither to its utility nor to its aesthetics, but the change of the color in every building affects the whole image of a street. With this decision for the translation, we pair a note to color but not a defining note, just the scale of the chord that is played. To carry out this translation, the HSV model was used. Hue is the color portion of the model, expressed as a number from 0 to 360 degrees. Saturation describes the amount of gray in a particular color, from 0 to 1.Value works in conjunction with saturation and describes the brightness or intensity of the color, from 0 to 1 [49].

Red hue falls between 0 and 60 degrees, yellow falls between 61 and 120 degrees, green falls between 121 and 180 degrees, cyan falls between 181 and 240 degrees, blue falls between 241 and 300 degrees, and magenta falls between 301 and 360 degrees. As we see in Figure 3.5 [50], the higher the saturation, the stronger the color, and also as the value drops, the darker the color becomes.

So firstly, for every road, the buildings are divided into two categories named "up" and "down" based on color, where every category is a different octave. It is important to note that the lower octave used is the 4^{th} octave of a piano, and the higher is the 5^{th}. So, the lower octave is the middle, with its lower pitch note being the middle C. This means that the "down" category represents not the lowest but the most common. The way the division will work is based on how the human perception of color works. The average arousal ratings of a human are higher for highly saturated chromatic colors than for colors with medium or low saturation[51]. Therefore if the saturation of the color of the building is higher than 0.625 ($S > 0.625$), then the building belongs to the "up" category, which as the name suggests is the higher octave, or if the saturation is equal or lower than 0.625 ($S \leq 0.625$) then the building's category is the lower octave. This way, the intenser colors are translated into higher frequencies, which are more noticeable from the human ear, and the softer colors are translated into the middle frequencies.

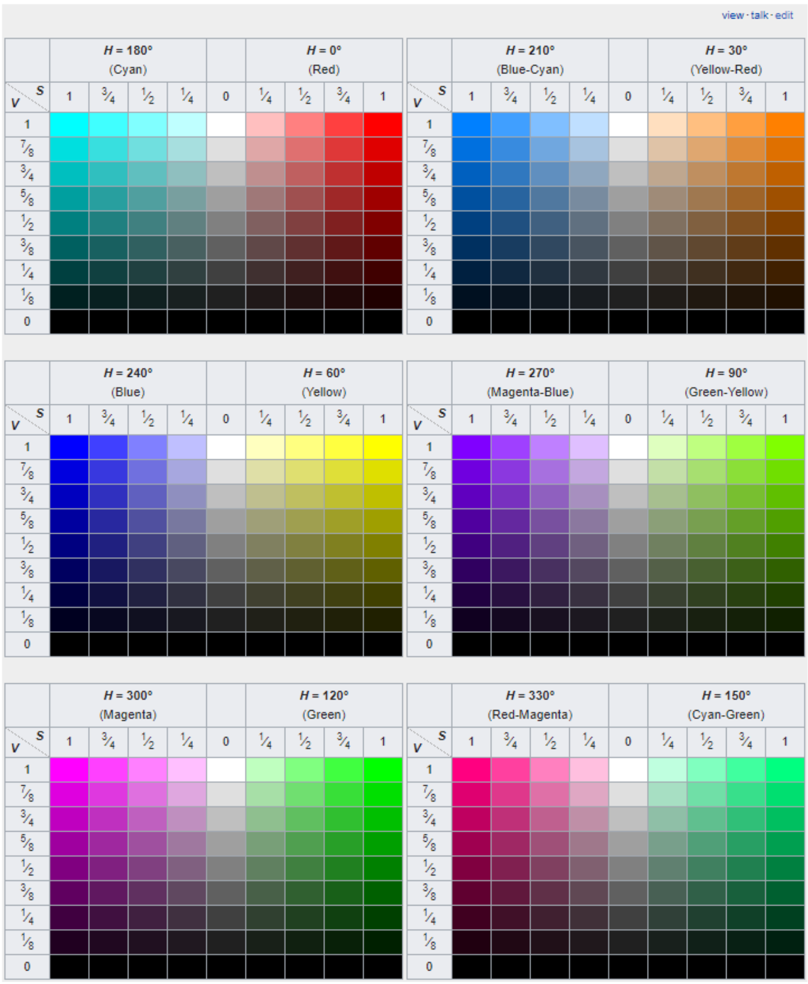


FIGURE 3.5: HSV colors

The first building of every category is taking the first place and also the lowest frequency of this category which is always the note C. Then, the next building that falls into the same category runs through a comparing algorithm to check if it deserves a place of one of the buildings that are already in this category or if it deserves the last place, which is the highest frequency until now. The first comparison is the value. The higher the value, the lower the frequency because a darker color, like a higher note, makes more impact. So, if the value of a new building is greater than an old one is ($V_{\text{new}} > V_{\text{old}}$) then it takes its place, and all the other buildings on the list move one note higher. If the values are equal ($V_{\text{new}} = V_{\text{old}}$) then we compare the saturation of the colors. Using the same concept as the octaves, the softer the color is, the lower the translated frequency must be. So, if the saturation of the new color is lower than an older one ($S_{\text{new}} < S_{\text{old}}$), it takes its frequency. However, if the saturation of the colors was equal, then we move into the final comparison, which is the hue. Suppose the hue of the new color is lower than an older one ($H_{\text{new}} < H_{\text{old}}$), then it takes its place. The hue alone does not impact how bold the color is; for that reason, the arrangement of the hues occurred from the concept that lower hue colors like yellow-red tints are more common in the urban environment and more subtle than magenta or blue. Finally, if the hues are also equal ($H_{\text{new}} = H_{\text{old}}$), then the new color is the same as the color that is being compared to, and logically it translates into the same note.

Height of ground floor

The height of the ground floor (HGF) is another parameter that affects the urban design. It changes the spectator's view considerably because the ground floor is at eye level, but it usually does not change that much as the total height of the building.

Moreover, we consider every floor's height to be 3 meters, which is the most used height in everyday buildings, and the height of the ground floor can diverse from 2.5 meters to 4 meters. This means that the user has four choices concerning the ground floor's height (2.5 meters, 3 meters, 3.5 meters, 4 meters). Four choices are enough to cover the needs of the HGF but are not enough to cover the needs of the total height of the building or else the number of the floors since there are many buildings with more than four floors. The same restriction can be found in a chord. As we see in Figure 3.4 the 3$^{\text{rd}}$ factor of the chord can vary between four positions.

Conversely, the 5$^{\text{th}}$ factor can change between four positions, but there can also be added extra notes on top of the chord to give more options to the corresponding architectural element, which will be translated through that. In conclusion, the 3$^{\text{rd}}$ factor gives the needed options for the translation of the height of the ground floor, but also it defines if the chord is major minor or suspended, so it offers the same level of impact to the chord as the height of the ground floor to the view of a building. Thereat, we choose the 3$^{\text{rd}}$ factor to translate this element of the building.

In order to achieve this translation, we firstly calculate the average height of the ground floors for every road. The HGF has four options available (S_{HGF} = {1,2,3,4}, $HGF_{\text{option}} \; \epsilon \; S_{\text{HGF}}$).The translation function for every option is:

$$Difference = HGF_{\text{average}} - HGF_{\text{option}}$$

$$PositionOfNote = 5 - Difference$$

There is the number 5 to this function because the average height of a road will be translated into the 5$^{\text{th}}$ note of the scale that is defined by the color. For example, if a building has the average height of its road and the 1$^{\text{st}}$ factor of the chord is C, then the 3$^{\text{rd}}$ factor is E, so the chord will be major (or augmented depending on the 5$^{\text{th}}$ factor) (Figure 3.4). For every other possible HGF, the translated note will vary depending on the difference between the HGF of this building and the road's average HGF. For example, the average HGF of a road is 3.5 meters ($HGF_{\text{average}} = 3.5m$). If the building has ($HGF = 3m$) then the translated note will be the 4th note of the scale that the color defines (with C as 1st factor the 3rd factor will be Eb) and the chord will be minor (or diminished depending on the 5th factor) (Figure 3.4). If the building has ($HGF = 2.5m$) then the translated note will be the 3rd note (with C as 1st factor the 3rd factor will be D) and the chord will be 2nd (Figure 3.4). Finally, if ($HGF = 4m$) then the translated note will be the 6th note (with C as 1st factor the 3rd factor will be F).

It is important to clarify that this type of translation does not aim to prove that an HGF is right or wrong, and that is why there is no choice of HGF that is not functional (ex. $HGF = 1.5m$). The translation tries to create the same relations of the buildings in music they already have on an architectural level. This means that the average HGF is translated into a major chord not to prove that it is right but because it is common, and it can connect more easily with the other chords. In a road where every building has ($HGF = 4m$) a building of ($HGF = 2.5m$) will hardly fit in. But if there is a road where there are buildings of every category, they can connect to ensure every HGF seems well implemented. This connection is the same connection that we want to achieve through music, and for that, every possible translated chord exists and follows the laws of music, but the user must connect them so that the outcome is harmonic.

Number of floors

The number of floors (NF) is a very important parameter of a building that, on the one hand, affects the urban design, but also is the geometric parameter with the widest range of options. The 5th factor of the chord is the only option that can cover the translation of NF as it is the only factor that can change between four positions, but there can also be added extra notes on top of the chord to give a wider range of translation. In MusiCityX we offer 10 options to the user for the number of floors that vary from 1 to 10 floors (S_{NF} = {1,2,3,4,5,6,7,8,9,10}). This does not cover all the possible floors, but it applies to most of the residential buildings in an urban environment. The way the translation works is similar to the translation of the height of the ground floor. This means that we first calculate the average number of floors for every road. Then, we calculate the Difference between the NF of the building and the average NF:

$$Difference = NF_{\text{average}} - NF_{\text{option}}$$

The buildings that have NF equal to the average NF or 1 floor higher or lower ($-1 \leq Difference \leq 1$) will be translated through the function:

$$PositionOfNote = 8 - Difference$$

The number 8 to this function is the position that the building with the average NF of the road will be translated into, meaning the 8th note of the scale defined by the color. A building with the average NF of the road means that there are many buildings in this road with the same NF, so it is common, or it has many buildings higher and lower, and this is the middle ground will be the connection between them. That is why the 8th note is chosen to represent the average, because it is the top factor of most of the types of chords, so it is the most common. The chords with this top factor are easier to combine with other chords because a minor chord is easier to use than a diminished one.

The buildings that have NF more than 1 floor higher from the average NF ($NF_{\text{option}} > NF_{\text{average}} + 1$) will be translated with the help of adding extra notes on the top of the chord.

$$PositionOfNote = 9$$

$$if(Difference == 2) => PosExtra[1] = 11$$

$$if(Difference == 3) => PosExtra[1] = 12$$

$$if(Difference == 4) => \begin{cases} PosExtra[1] = 12 \\ PosExtra[2] = 14 \end{cases}$$

$$if(Difference == 5) => \begin{cases} PosExtra[1] = 12 \\ PosExtra[2] = 15 \end{cases}$$

$$if(Difference == 6) => \begin{cases} PosExtra[1] = 12 \\ PosExtra[2] = 15 \\ PosExtra[3] = 17 \end{cases}$$

$$if(Difference == 7) => \begin{cases} PosExtra[1] = 12 \\ PosExtra[2] = 15 \\ PosExtra[3] = 18 \end{cases}$$

$$if(Difference == 8) => \begin{cases} PosExtra[1] = 12 \\ PosExtra[2] = 15 \\ PosExtra[3] = 18 \\ PosExtra[4] = 20 \end{cases}$$

$$if(Difference == 9) => \begin{cases} PosExtra[1] = 12 \\ PosExtra[2] = 15 \\ PosExtra[3] = 18 \\ PosExtra[4] = 21 \end{cases}$$

As shown to the function above, the chord's top factor will be the 9th note of the scale, and then depending on the Difference value, 1 to 4 extra notes will be added on top of the chord. This way, even if the average NF is 1 floor and the building we want to translate is 10 floors, the translation would be possible.

The buildings that have NF more than 1 floor lower from the average NF ($NF_{\text{option}} < NF_{\text{average}} - 1$) can not be translated with just the chord because this would mean that the top chord would go as low as the Difference value indicates, and this could lead to the top factor being the middle factor of the note destroying the structure of the chord. Therefore these buildings will be translated with the help of adding an extra note before the chord.

$$PositionOfNote = 7$$

$$PosExtraNoteBefore = 9 - |Difference|$$

As we see above, the top factor's position will stay to the 7th note, and the extra note will start from also 7th and will diminish depending on the Difference. However, this function will be applied if the Difference value is lower than -1. This means the absolute of the Difference will be 2 or more, and that is why the number 9(7+2) exists to the position of the extra note, to remove this Difference.

Length of the face of the building

The length of the building is easily translated into the duration of the chord. The more length of the side of the road the building covers, the more time the chord sounds.

3.5.2 Windows, doors and balconies

While the chord is playing, the notes that represent windows, doors and balconies will start playing depending on their placement at the building and their length. The doors will be represented by the note of the 1st factor of the chord, the balconies by the 3rd factor and the windows by the 5th factor. The floor that every element is placed does not change the resulted translation, every floor is treated equally. If for example there is the same window in all 3 floors of the building, then 3 notes will be positioned to the exact same place which will lead the user to listen one note but louder.

3.5.3 Inversion of Color

The translation of the buildings is only half of the work to achieve the goals of this project. When the translation is done, and the user listens to the results, they must have the ability to change the notes accordingly so that the auditory results are satisfying, and these changes must be translated back into the urban landscape. The inversion that is happening includes all the translation algorithms in reverse, meaning that the program calculates the new state of the component translated through this note and applies it in every change of a note.

This inversion includes the inversion of color, which in contrast with all the other translated architectural elements, is more arbitrary, and it does not involve only one value, but three (H,S,V). Consequently, we need a method that the inversion follows the rules of the first translation. As a result, when we change the note of the color, the function starts searching all the other buildings of the road to see if there is one of them translated into this new note. If the search was successful, then the building will adopt the color of this building. For instance, if a building had the note E as the translated note of its blue color and we changed it into D, and there is a building with also the note D, and it is orange, then the building will become orange and have D as the represented note. If there is no such building, the function searches if there is a building that is translated into a note where the new note is between this note and the old one. For example, if a building had the note E as the translated note of its black color and we changed it into D, and there is a building with the note C as the translated note of its white color since this relation C<D<E is valid then the building will become gray and have D as the represented note. If nothing from above was true, then the translated note of the color is either the highest or the lowest in the road. In this case, the color's shade is becoming lighter or darker, depending on if we increase or decrease the note's position, respectively.

3.5.4 Examples of the proposed translation

Before creating the MusiCityX platform, some examples of the translation were created in order to be able to calibrate the translation method. The examples helped us change some parts of the translation in order to result in the final method.

Firstly, the translation method proposed to translate the building into a chord, but in this case, the chord contained four notes. The extra note that is not contained in the final translation was the same as the base note but was played on the F key. The translation of the building's dimensions was close to the final translation, but they contained the laws considering the percentage of land cover and the structural factor, and also the 3rd factor translated the area covered by the building in square meters and the 5th factor the height in meters. The translation of the roof was through the translation of the top line's movement that defined the roof, and the way we did this was through the counterpoint. The translation of the color was not defined yet. The translation of the balconies, doors, and windows was the same. This example of translation is shown in the Figure !3.6.

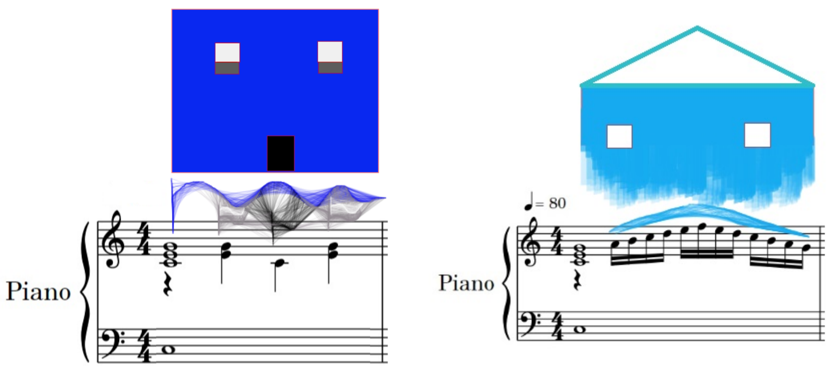


FIGURE 3.6: Theoretical example No.1

Afterward, there was another more detailed example 3.7 of an existent road in Chania. The extra note mentioned earlier was considered unnecessary. The roof's translation was very distracting as all the other notes could not be heard, so it changed to its final version. The color's translation was not precisely defined yet. To create this example, the urban scenery was depicted to Revit, and the musical translation was created through FL Studio.

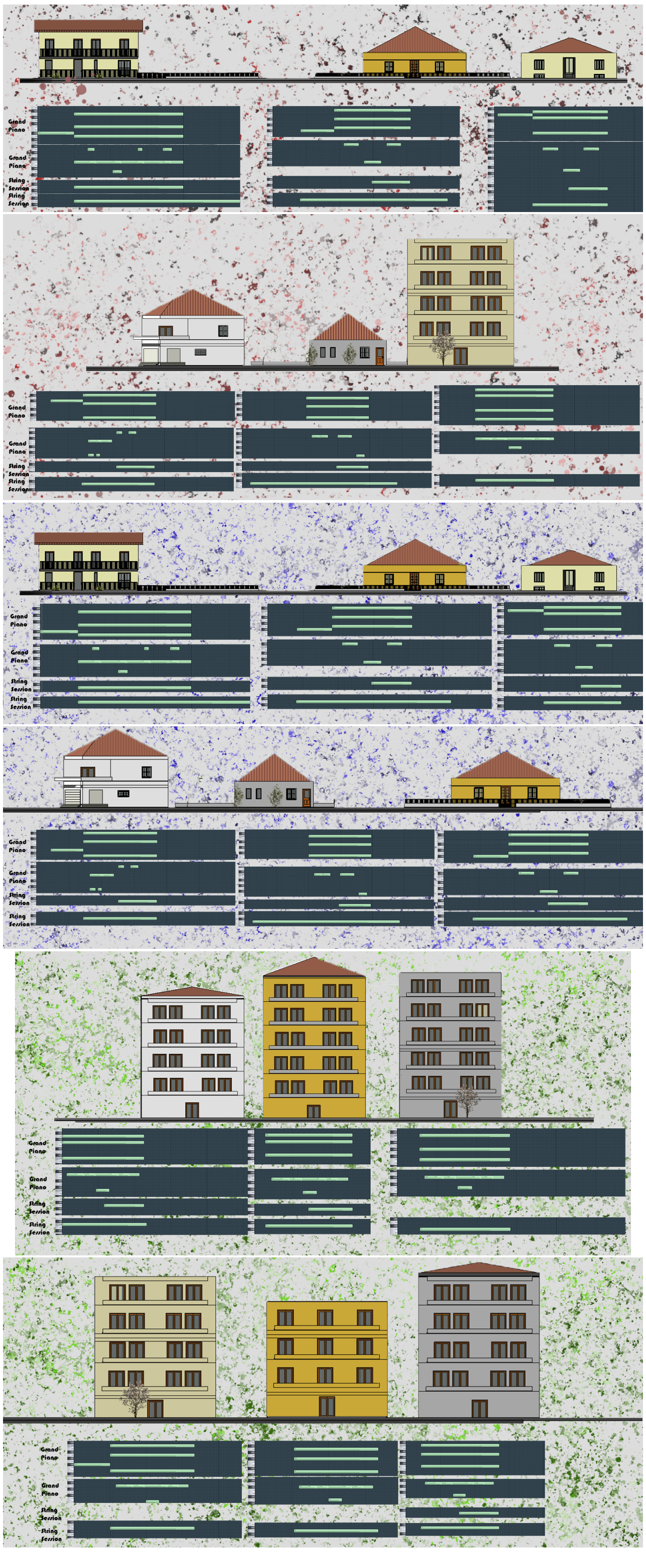


FIGURE 3.7: Theoretical example No.2

We decided that the laws will not be used on the final translation because they do not represent coherency, and also they complicate the procedure without a good cause.

The final example 3.8 was another road in Chania, where Professor P. Parthenios noted the urban landscape's inconsistencies that must be heard on the musical translation. While creating this example, there was a remark that the ground floor's height was frequently changing and played an essential role in the coherence of the road, so we translated the road with two different approaches on the 3rd factor. In the first one, the 3rd factor was connected with the building's depth and in the second one, the 3rd factor changed into the height of the ground floor. The 5th factor changed into the number of floors instead of the height counted in meters. The color translation had also two propositions that were shown in two different examples.

To create this example we used satellite views from Google Earth and Map Ruler, but we also visited the road to measure every building's parameters. Following this, we recreated the road in Photoshop as shown in the Figure 3.8 where we marked the parameters. Afterword, the musical piece was created in FL Studio, by applying the translation method proposed to the building's of this road. It is important to state, that only one side of this road was translated consisted of 27 buildings. The final result was a video were the road was translated four times to show the different approaches on the 1^{st} (color) and on the 3^{rd} (height of ground floor) factor. These examples were viewed by the professors of this thesis committee and the final form of the translation method was acquired. When the final form was decided the inverted translation was applied, meaning we altered the music theme to sound more harmonious and we changed the buildings accordingly in order to see whether the urban landscape became more coherent 3.9.

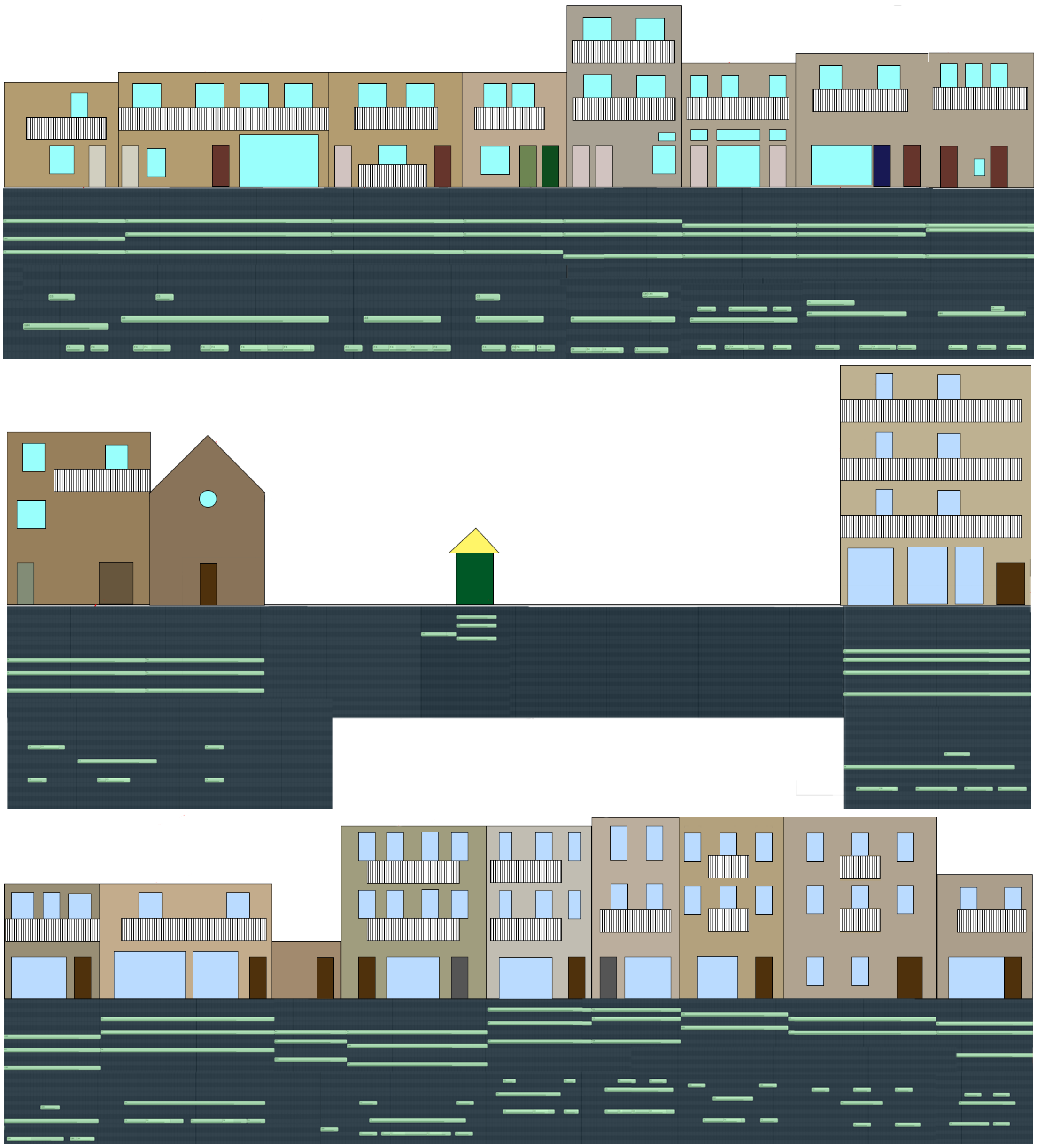


FIGURE 3.8: Theoretical example No.3 before the alterations

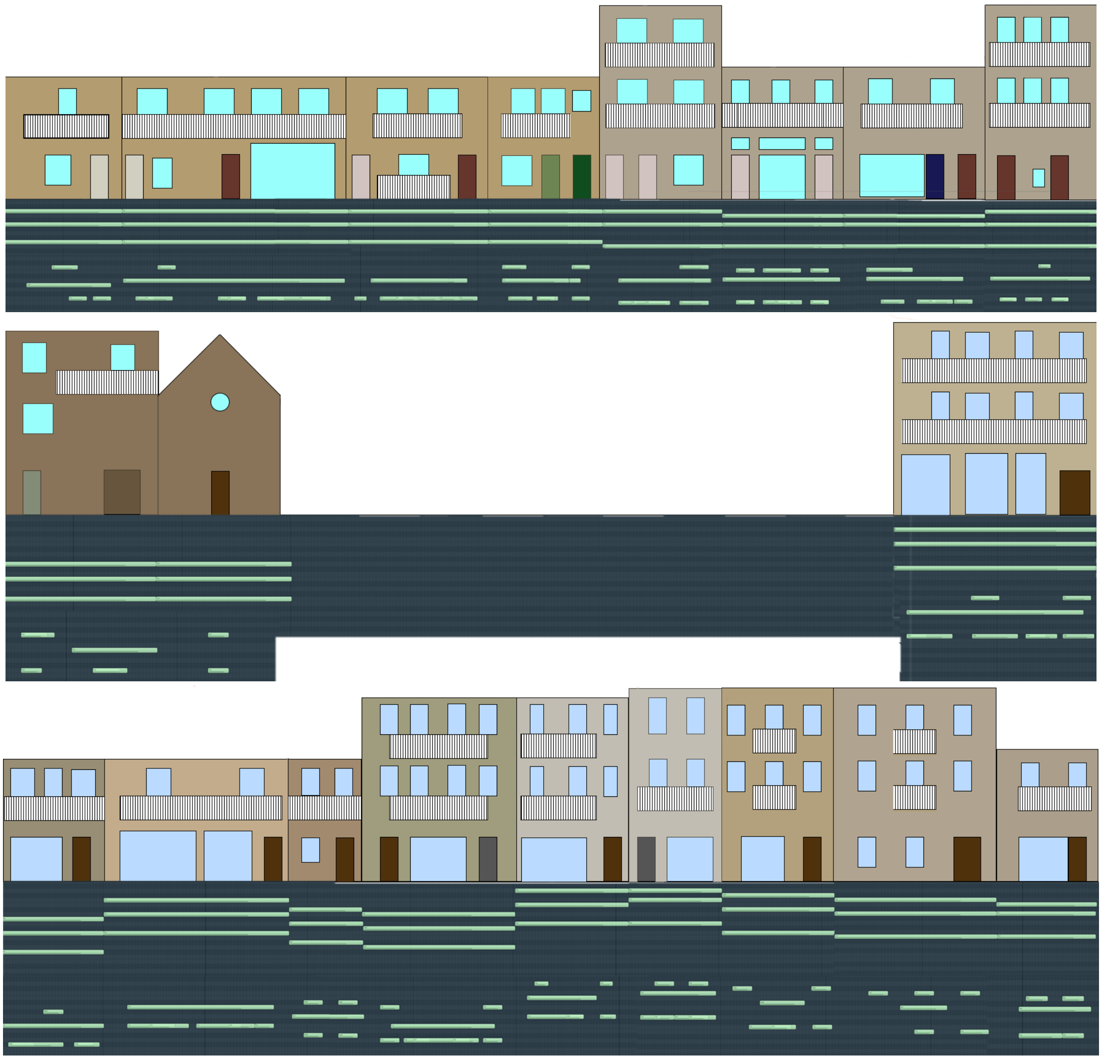


FIGURE 3.9: Theoretical example No.3 after the alterations

Chapter 4

Implementation

4.1 Introduction

In this part there is a detailed view of this thesis's approach on creating a platform that implements the theoretical framework we built. This platform must offer the user the ability to create an urban landscape in order to depict it through music. Then we have to transfer the translation theory mentioned on the 3 into this platform in order to create the musical footprint of the urban scenery created by the user. Finally, the platform must offer the tools needed to alter this musical footprint aiming to improve the coherence of a route of this landscape according to the tastes and the needs of the user.

4.2 Program subdivisions

In order to achieve the connection between architecture and music, the user has to form the urban landscape. For this purpose, the user must have the means for creating roads and buildings, adding windows, doors, balconies, and balcony doors, and choosing a route to translate into music. The program is subdivided into four parts 4.1.

In the first part, the user creates the roads of the urban scenery. They can create a single road or a road network.

In the second part, they create the buildings and add to the buildings all the elements they want (windows, balconies, doors). Moreover, they can first choose the building's dimensions, the number of floors, the color, and choose whether the building will have a roof. Afterward, they can place the building near a road or near a corner between two roads. If they want to add elements to the building, they can click on the building, and they can choose the floor where the element will be placed. Then they can choose an element (a window, a door, a balcony, or a balcony door), and the element appears to the left edge of the building's chosen floor. The user, then, can move the element left and right to choose its final position and then build it. This procedure can repeat until the user is satisfied with the result.

In the third part, the user must choose the roads that will create the path that our music translation will follow. White cubical buttons will appear on top of every road part, and the user must choose them in the correct order they want to translate them.

The final part is the music part, where the user listens to the musical piece resulted from the translation of the chosen path of the urban landscape created, and they can make changes to result in a more harmonious musical and architectural composition. In this part, there is a main DAW on the bottom of the screen that shows one side of one road part each time. This DAW can be used to listen to the musical translation of the chosen route, to move a building, and to change the length of the building's face. Apart from that, if the user clicks on a building, another DAW appears that contains the chord of this building as distinctive notes and the notes of the building's elements (windows, doors, balconies). Using this DAW, the user can change the building's color, height of the ground floor, and number of floors. They can also move, delete or add an element to every floor they desire.

In addition, whenever the user is finished with one of the subdivisions, only if they added at least one road or building the platform lets him continue to the next subdivision. If the user achieved this goal, a bar on the screen's left side is filling up, as shown in the figure below. Dividing the program into four different subdivisions was a way to make a platform as friendly and as easy to understand to the user, where every subdivision is devoted to a specific task, but also to guide the user through the platform. All the UI and cameras' changes through the subdivisions are controlled by a script called *TheCameraControl*. Also, almost every button's function is part of a script called *Button1*.

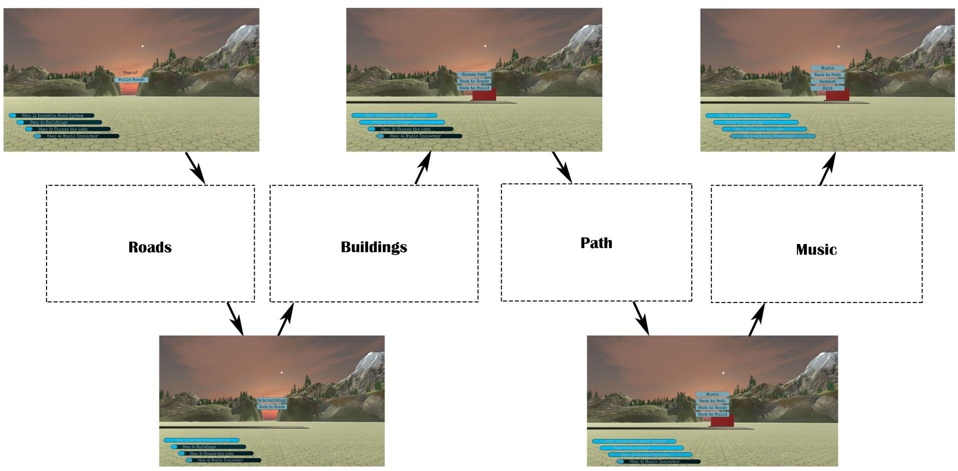


FIGURE 4.1: Subdivisions of the platform and in between menus

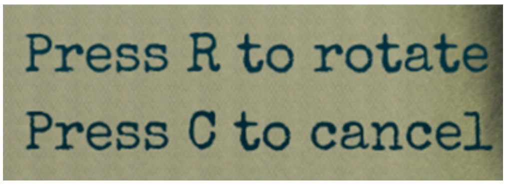
4.3 Road subdivision

This is the first subdivision of the program. Here the user is able to create a network of roads or just build a single road.

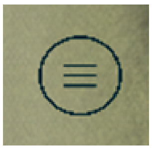
4.3.1 UI and view of the subdivision

At the initial screen of this subdivision there are three buttons on the screen. The button first button is "New Road" 4.2a that creates a road's preview, the second button is "Finish" 4.2d that closes this subdivision and shows the menu and the last button is "Help" 4.2c. Also, on the initial screen there is a text that shows the rotate and cancel commands 4.2b. When the user presses the help button then an image is appearing that shows what the user can do 4.3b. Also, when there is at least one road placed a button called "ERASE ALL" 4.3a appears that if pressed erases all the placed roads.

(A) New Road



(B) Rotate and Cancel

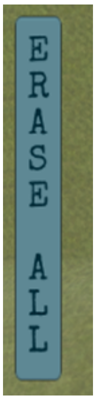


(C) Help

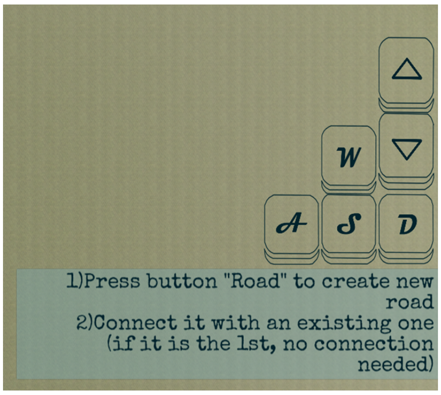
(D) Finish

FIGURE 4.2: User interface at the start of the road's subdivision

When the user presses the "New Road" button a road is attached to the cursor. If the road can be placed where the cursor is pointing then its color turns blue, if not then its color turns red. Before placing the road, the user can press "R" to rotate the road 90 degrees. If the user presses left click and the road placement is correct (the color of the road is blue) then the road is placed 4.4. Otherwise the road deletes itself. Also if the user presses the key "C" the road's preview is being deleted if one exists, so the placing of the road is being cancelled. The first road can be placed anywhere, but every other road must be connected to another already placed road to have a correct placement.



(A) Erase all



(B) Help image for roads

FIGURE 4.3: Rest of the UI of road's subdivision

FIGURE 4.4: Preview of the road

4.3.2 Components and implementation method

The way this subdivision works is through the scripts (called *BuildManager* and *Preview*) that use prefabs and collision detection. When the user places a road, a prefab of a road is instantiating through the *BuildManager* script. The roads are made through changing simple cubes. There are two types of prefabs depending on the rotation the user wants the road to be placed. Every prefab has six empty objects with sphere colliders as children that work as snapping points between the roads. Every one of these objects has a different tag 4.5. They also have two empty objects with colliders that cover the whole length on the left and the right side of the road that will be used to collide with the buildings added on the next part of the program.

When the user presses the "New Road" button the road that is attached to the cursor is a different instantiated prefab, which works as a preview of the road that is going to be built. There are also two types of prefabs depending on the rotation. Every time the user presses "R" the prefab showing changes. These prefabs have box colliders on them. This way, they trigger a collision when the user moves them near an already placed road. When a trigger is happening and the user presses left click to place the road a function of the Preview script is called through *BuildManager* that uses the tag of the collider of the already placed road to determine how the road must be placed, if extra road parts must be added but also to change the tags that need to as will be explained.

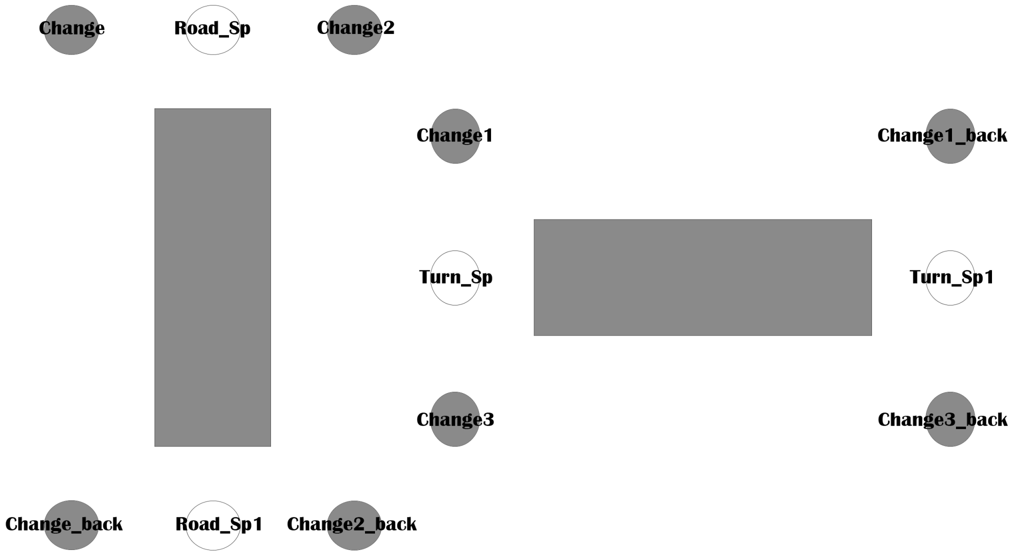


FIGURE 4.5: Prefab roads with snap points and tags

If the road preview has the same rotation as the road that is going to be connected with then the triggered child (snap point) is on the center axis of the road (the white circles on 4.5). In this case, the new road is placed and the child that was triggered changes tag name into "Useless". If the road preview has a different rotation, then depending on the name of the tag the new road is placed, with an extra road part to create the corner needed. But, sometimes more than one extra road parts need to be placed, depending on the already connected roads. This extra road part is made using the ProBuilder feature [52].

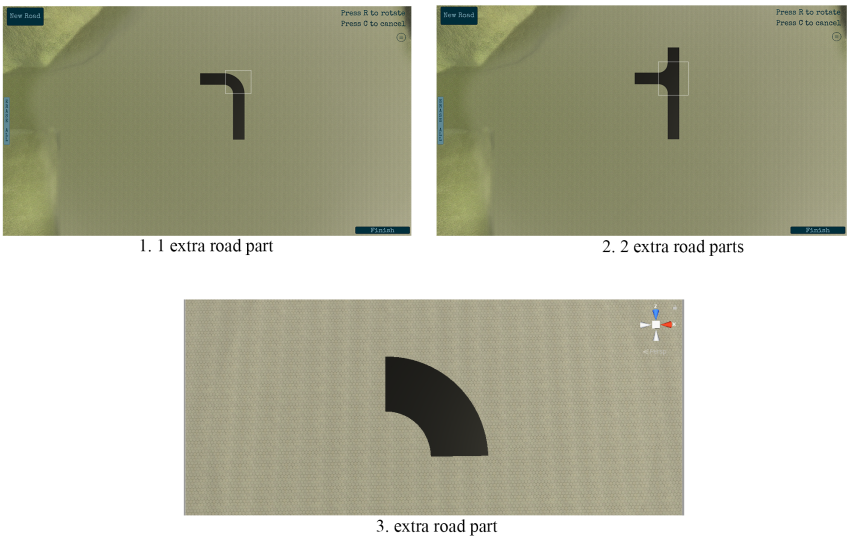


FIGURE 4.6: Extra road part examples

As we see on the example 4.6 if we connect a road to the same snapping point, but the already placed road is already connected with another road on the same side, the results change. That is why the snap points that are next to each other 4.5 (ex. Change, Road_Sp, Change2) affect each other. For example, if "Road_Sp" turns into "Useless" then "Change" and "Change2" will also change to different tag names. Finally, whenever a new road is placed then its name will be "roadN" where N = number of the roads placed. If the new road is the sequence of the "roadL" that is connected to then its name will be "roadL_N". This naming procedure will help us to find which road parts we will consider parts of the same road on the music translation.

4.4 Building subdivision

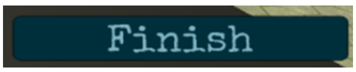
Generally, this subdivision is serving the purpose of creating buildings and adding windows, doors, balconies and balcony doors to them.

4.4.1 UI and view of the subdivision

At the initial screen of this subdivision 4.8 there are three buttons on the screen. The first button is the building button 4.7a, then there is the "Help" button 4.7b and the other one is the "Finish" button 4.7c. If the user presses the help button then an image appears that guides the user on how to use this subsection of the program 4.9. If the user presses "Finish" this subdivision ends and the menu is showing.

(A) Building

(B) Help



(C) Finish

FIGURE 4.7: User interface at the start of the building's subdivision

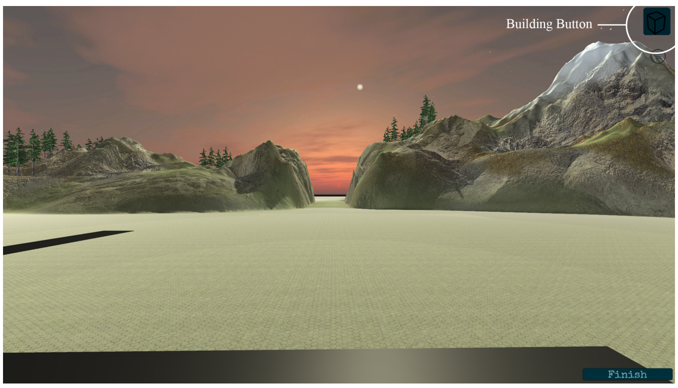


FIGURE 4.8: Initial screen of building's subdivision

When pressing the building button 4.8 a window appears were the user add the variables of the building to be created and it is called "Building Control Panel" 4.10. These variables consist of the length and the width of the building, the number of floors which vary from 1 to 10 floors, the height of the ground floor which can vary from 2.5m to 4.5m while every other floor is 3m tall, the color and the existence or not of a pyramid hip roof. The small window that let the user choose a color is an added tool called "ColorPicker" 4.11. When they choose the color from the ColorPicker window then they press the check button and the window closes. To be able to see the color without opening the window an image is added to the building window that adopts the chosen color.

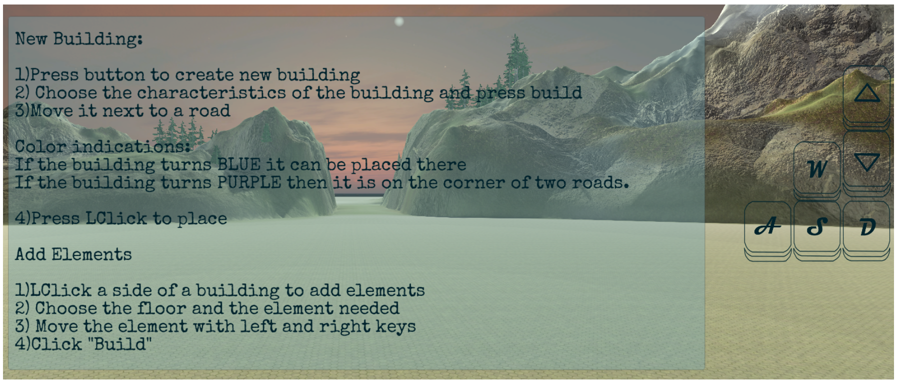


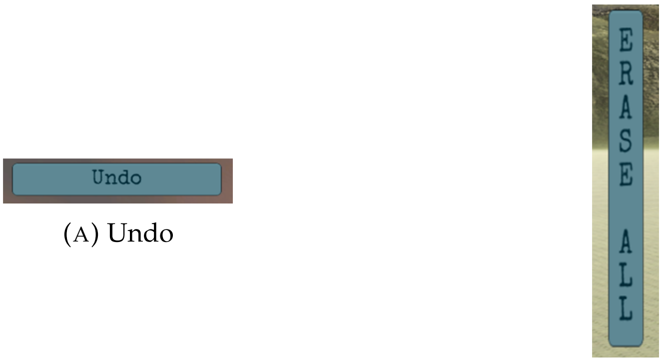
FIGURE 4.9: Help image for building's subdivision

FIGURE 4.10: Building Control Panel

Then the instantiated building is attached to the cursor. It can move left and right following the cursor but it can move to all the other directions only by following the movement of the camera. To place a building into the scenery, you have to move it near a road. When it is near a road the building turns light blue 4.19. If the user moves the building near a corner between two roads the building's color turns into purple 4.20. While the user moves the instantiated building but has not placed it a text is appearing on the screen 4.15. This text shows that if the user presses the key "C" the instantiated building will be deleted. When there is at least one building already placed, two extra buttons appear, the button "Undo" and the button "Erase All". The button "Undo" erases the last placed building and it can erase up to twenty buildings. The button "Erase All" deletes all the placed buildings.

FIGURE 4.11: ColorPicker

FIGURE 4.12: Cancel



(A) Undo

(B) Erase All

FIGURE 4.13: User interface at the start of the building's subdivision

After placing a building you can click on a side of the building and the UI for adding elements to the building is activated, and will change the camera's height to face the 1st floor (ground floor) and movement to be able to turn around of the building and zoom in and out. The UI for adding elements consists of a button for adding a window 4.16a, a door 4.16b, a balcony 4.16c, a balcony door 4.16c, a dropdown element where you can choose the floor where the element is added to 4.16e and also a "Back" button 4.16f to return on creating buildings. When pressing a button to add an element the element appears on the chosen floor and the "Build it" button 4.17b appears. The element moves left or right using the keyboard arrows and when the position is satisfactory to the user they can press "Build it" to place it. When at least one element is placed the "Undo" button 4.17a appears which if pressed deletes the last element placed.

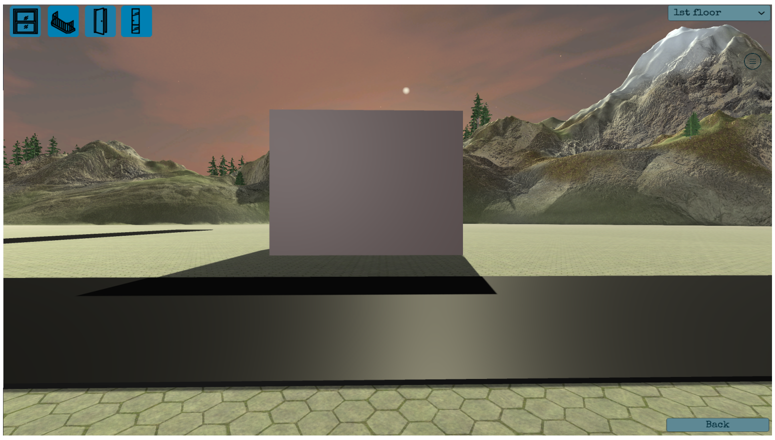


FIGURE 4.14: Screen with UI for adding elements

FIGURE 4.15: Cancel

(A) Window

(B) Door

(C) Balcony

(D) Balcony Door

(E) Drop down for floors

(F) Back

FIGURE 4.16: Starter UI for adding elements

(A) Undo

(B) Build it

FIGURE 4.17: Extra UI for adding elements

4.4.2 Components and implementation method

Firstly, when the user presses the "Building" button the UI of the Building Control Panel activates through the button1 script.When the user inserts all the parameters and presses "Build" then through functions of the script called *apartment_build* according to the number of floors given, it chooses to instantiate one of the 10 prefabs of buildings 4.18. Every one of these prefabs is a 1m by 1m building with different number of floors and it has a roof as a child. These buildings are made out of cubes that have a box collider on them and the roof is made of a pyramid shape. They also have invisible thin plates with colliders to every side of them. These plates represent the sides of the building and they contain all the needed 2D UI that will be needed for the subdivision of music and in that 2D UI there is a script attached called *PlayEach* and will be explained there. When the building is instantiating the corresponding function is applying the chosen parameters to it.

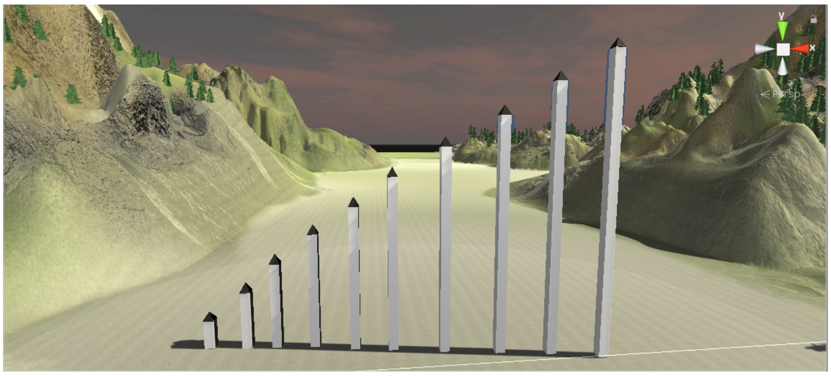


FIGURE 4.18: Prefabs of buildings of every floor

Then the instantiated building is attached to the cursor. The user can move the building to choose where to place it. When it is near a road a collision occurs between the main collider of the building and one of the colliders that cover the left or right side of the road, depending on which side the building is closer to. This collision changes the building's color through a simple script called *Floor_script* into light blue 4.19. When this occurs, if the user presses left click the building is placed as a child of the road is attached to. It also changes name to "floorN" where N = number of buildings placed. Another possibility is to place the road on a corner between two roads. If the user moves the building to this position two collisions happen simultaneously, one for each road and the building's color turns into purple 4.20. If the user choose to place it there a function creates an empty object called "roadA+roadB", where roadA and roadB are the roads the building is cornered between, and the building becomes its child and changes also name to "floorN" where N = number of buildings placed.

After placing a building you can click on a side of the building activating a collision with one of the thin invisible cubes, this collision will activate the UI for adding elements to the building, and will change the camera's height and movement through *Button1* script. The UI for adding elements appear 4.14. The elements used are extracted from Revit [47], but their color is changed and also extra window-like elements are added to make them more realistic 4.21.

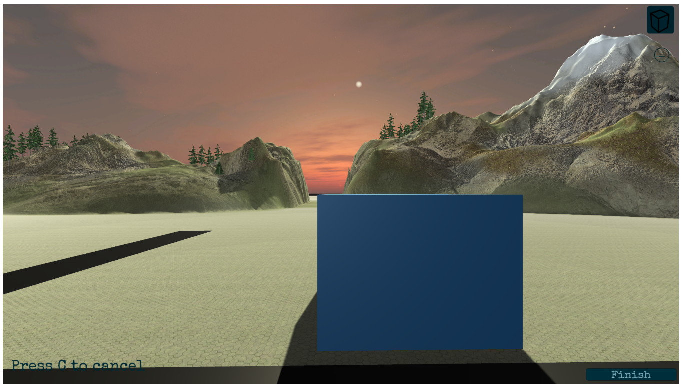


FIGURE 4.19: Collision of a building with one road

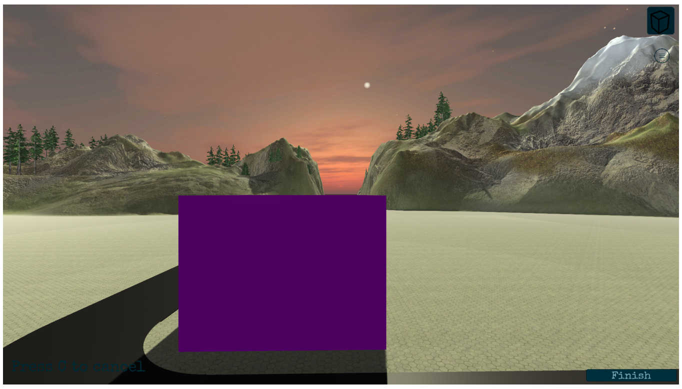


FIGURE 4.20: Corner collision of a building with two roads

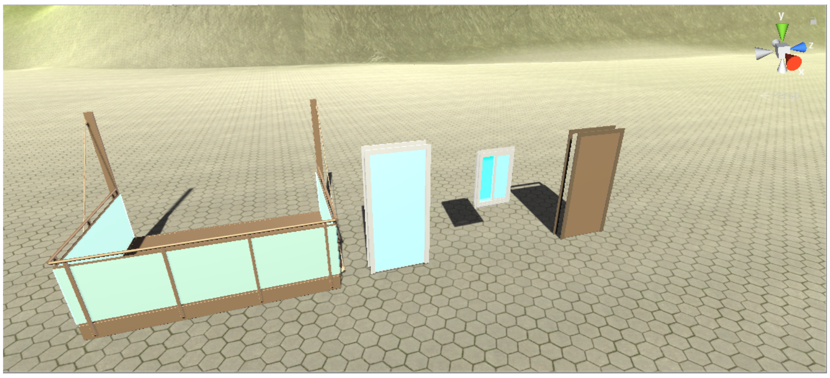


FIGURE 4.21: Wind, door, balcony and balcony door

When one of the buttons are pressed a function calculates the distance that this element must have from the center of the building to be layered to its surface. It also determines the height that this element will appear. To achieve that it takes into account the floor chosen, but also the total height of the building to find the ground floor's height. If for example the height of the ground floor (1^{st} floor) is 2.5 meters and we want to add a door to the 2^{nd} floor in comparison with a building of a 4m ground floor height, the door will be placed 1.5 meter below on the former than the latter. When an element appears on the building a button "Build it" appears 4.22. Doors can be placed only to the ground floor 4.23. When an element is placed it becomes a child of the side is placed to.

When the user wants to stop adding elements they can press the button "Back" to return the camera to its last position.All this UI and all the functions needed to add these elements are controlled by a script called *dbwind*.

FIGURE 4.22: "Build it" button

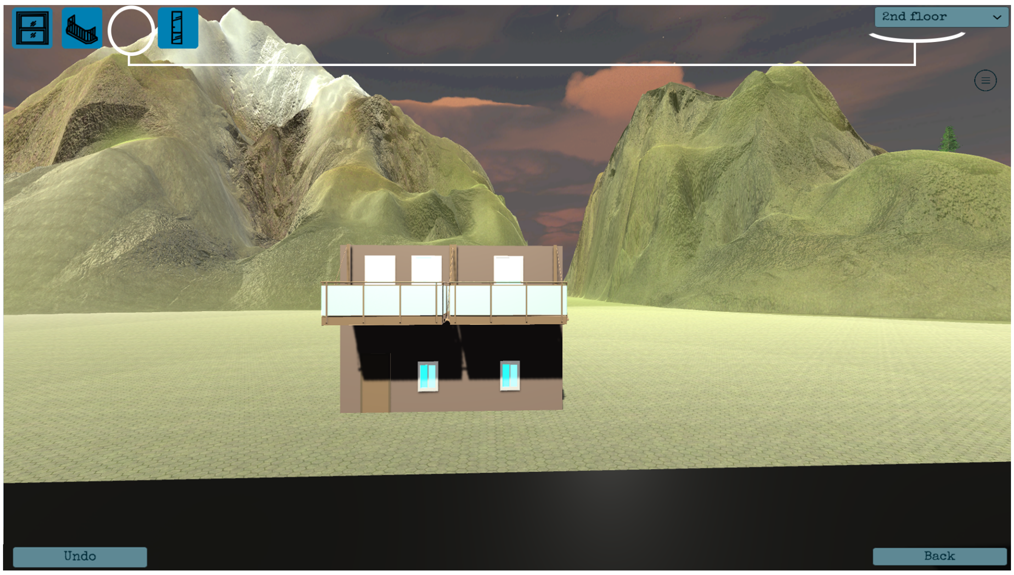


FIGURE 4.23: Example of adding elements

4.5 Choice of path subsection

This is the penultimate part of the platform. In this subdivision the user must choose the path that the MusiCityX tool will translate.

4.5.1 UI and view of the subdivision

On the initial screen 4.24 the user can see the roads they created with a white box to the center of every road part. There are two buttons showing on the path's subdivision 4.25, the help button 4.25a and the "Finish" button 4.25b. If the user presses the help button then an image appears that guides the user on how to use the path's subsection. To be able to choose the road parts that will form the path, the user must click on their boxes on the correct order they want to translate them and then press finish. When the boxes are being pressed their color changes into light blue. The "Finish" button closes this subdivision and shows the menu.

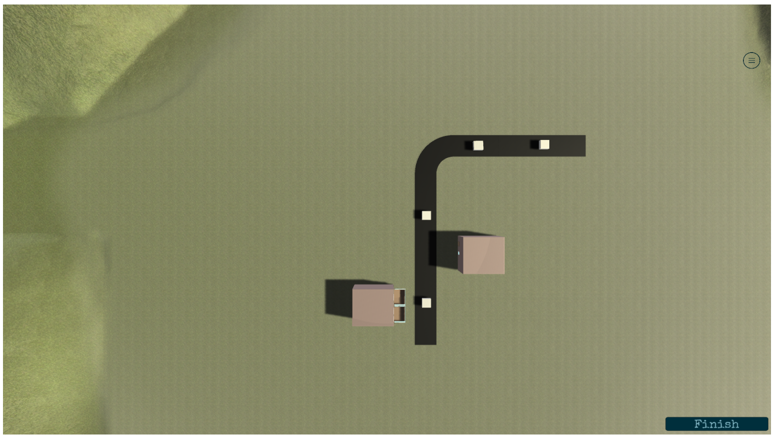
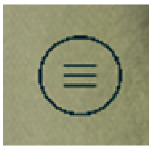


FIGURE 4.24: Initial screen of path's subdivision



(A) Help

(B) Finish

FIGURE 4.25: UI of the path's subdivision

4.5.2 Components and implementation method

In this subdivision, a function activates every road's child white cube 4.24. These cubes work as 3D buttons. When the user presses one cube it turns blue and a function adds it into the list of the roads to be translated 4.26. Whenever a road is inserted into this list two empty objects are being created as children of the DAW canvas appearing to the bottom of the screen in music mode and they represent the left and the right side of the road, so they adopt the name of the road plus the name "LEFT" or "RIGHT" at the end accordingly. They will be the platform where the chords of the buildings will appear for every side of the road. Once one empty object is created a function searches the children of the road to find if there are buildings on the side of the road this empty object represents. If the function finds a building then it creates a white 2D square that has the same length as the length of the face of this building and the same distance from the middle of the empty object as the distance of the building from the middle of the road and it becomes a child of this no longer empty object representing the road. This white square is the representation of the chord that the building will be translated into and has a script attached to it called *AudioChord*. That script contains all the notes and functions needed to play the chord, but also it contains functions that find all the details of the building that this chord represents.

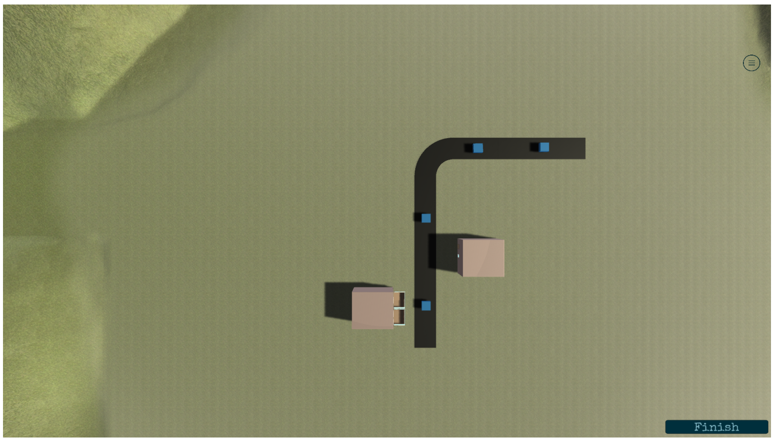


FIGURE 4.26: Example of a chosen path

The order the user chooses the roads defines the direction of the path and the order the empty objects are placed into the list that in music's subdivision when the player presses the button this will be the list that will define what part of the road will play and when.The music produced from the left side of the road will be played first and then the right one. So, every time the user chooses a road part that has the same first name for example road1_1 and road1_2, which means that they are parts of the same road then the order the created empty objects will be placed into the list is: road1_1LEFT, road1_2LEFT, road1_1RIGHT, road1_2RIGHT. If the user chooses a road part that has a different rotation than the last the script will search if there is any buildings on the corner. If for example the previous chosen road was the road1 and the new chosen one is the road2 then the function will search a game object named road1+road2. If there is no corner building then the list's form will be: road1LEFT, road1RIGHT, road2LEFT, road2RIGHT. If there is a corner road, then two new empty objects are created, one for each side of the building that is facing a road. The placement of these objects on the list depends on the direction of the path and the placement of the building. As we see on the figure 4.27 on the top left example the corner building's sides will be one after the other, but this does not happen to the bottom left example. Also, the top right and bottom right examples have the exact opposite order for the sides of the corner buildings.

Last but not least, while the user chooses the path, according to the direction of the path another list is created which defines the position of the needle that will pass through the notes and activate them for every road.

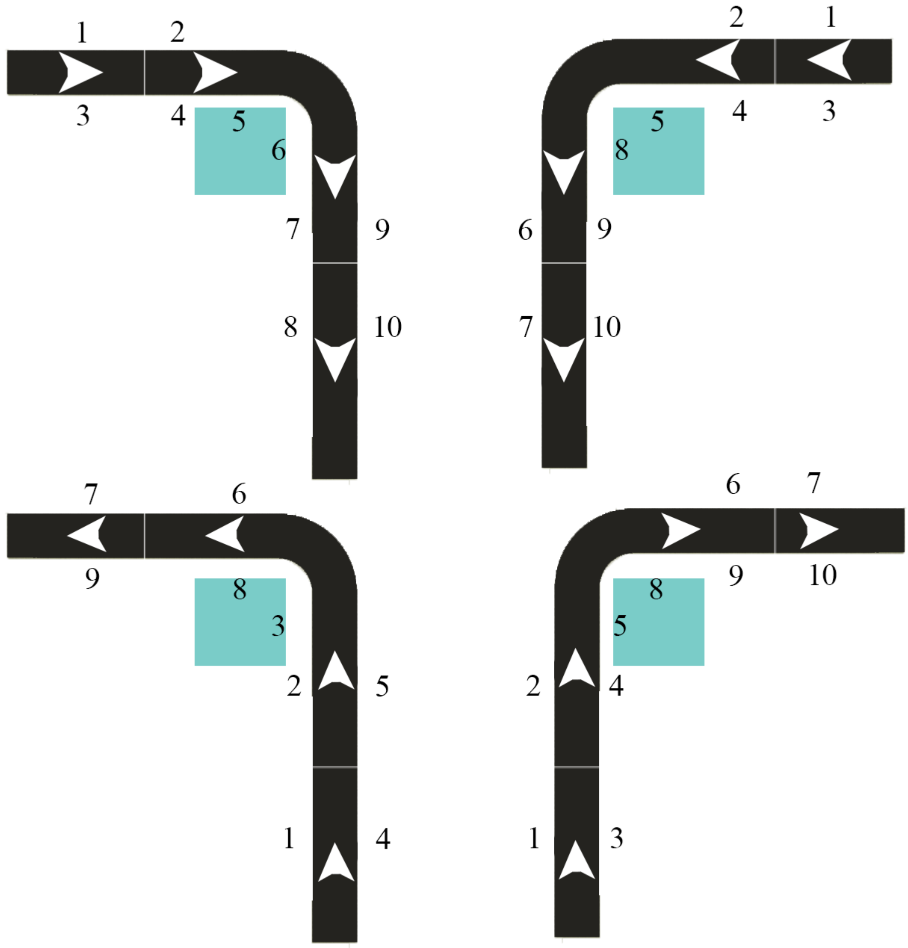


FIGURE 4.27: Examples of different orders of corners

4.6 Music subdivision

This is the last and most important subdivision of the platform. In this subdivision the user can firstly listen the music translation of the urban landscape they created. Apart from that they can change the musical theme and watch the simultaneous changes of the urban scenery.

4.6.1 UI and view of the subdivision

The main screen is divided in two parts and there are two cameras activated on the same time. The top part that covers 75% of the screen shows the surroundings where the user can move and the bottom 25% shows the main DAW (Digital Audio Workstation) 4.28. The user can move using the keys W, A, S, D to the x axis, the up and down arrow keys to move to the y axis and the camera's rotation is following the mouse movement. There are two buttons showing on the top screen, the help button 4.29a and the "Finish" button 4.29b. The help button, if pressed, an image appears that guides the user on how to use the music's subsection 4.30. The "Finish" button, if pressed, closes this subdivision and shows the menu.

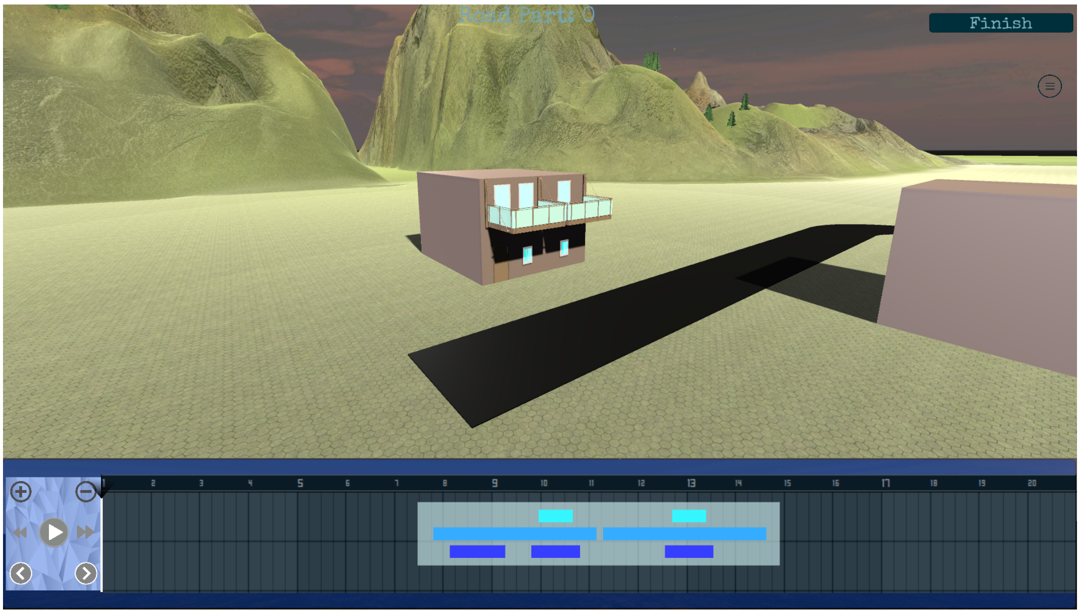
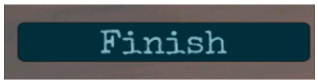


FIGURE 4.28: Initial screen of the music section

(A) Help



(B) Finish

FIGURE 4.29: UI of the music's subdivision

FIGURE 4.30: Help image for music's subdivision

The main DAW has a control panel 4.31 on the left side and on the rest DAW one side of one road part is showing each time. For example if road1 and road2 are the only roads translated then the DAW will first show the road1LEFT with every building on this side of the road. The DAW has a play/pause button 4.32 which activates or deactivates the music needle. When the music needle is activated it moves from one side of the DAW to the other depending the direction of the path chosen and triggers the chords and the notes to play when passing through them. There are also two buttons that offers the user the ability to move through the translated parts 4.33. For example if the road1LEFT is the current side showing then by pressing the right button the road1RIGHT will be activated and the last one will deactivate. At the bottom of the DAW control panel there are two arrows 4.34. If the user choose a white 2D square, which is the chord of the building and presses one of the arrows then the chord and in consequence the building moves left or right respectively. at the top of the control panel there are two other buttons that by choosing a chord and pressing them the chord and the length of the face of the building will grow or shrink accordingly 4.35.

FIGURE 4.31: Control panel of the main DAW

FIGURE 4.32: Play/pause button

FIGURE 4.33: The buttons that change the road part playing

FIGURE 4.34: Buttons that move the chord and the building

FIGURE 4.35: Buttons that change the size of the chord and building

If the user presses on the face of a building then its individual DAW appears. This DAW includes the notes of the chord separately and also the notes of the windows, balconies and doors. The first time this DAW opens the user can see both the notes of the chord and the notes of the other elements 4.36. By pressing the change button 4.38 then the view can alter between viewing both, viewing only the chord 4.37 or only the melody 4.39 created by the windows, the doors and the balconies.

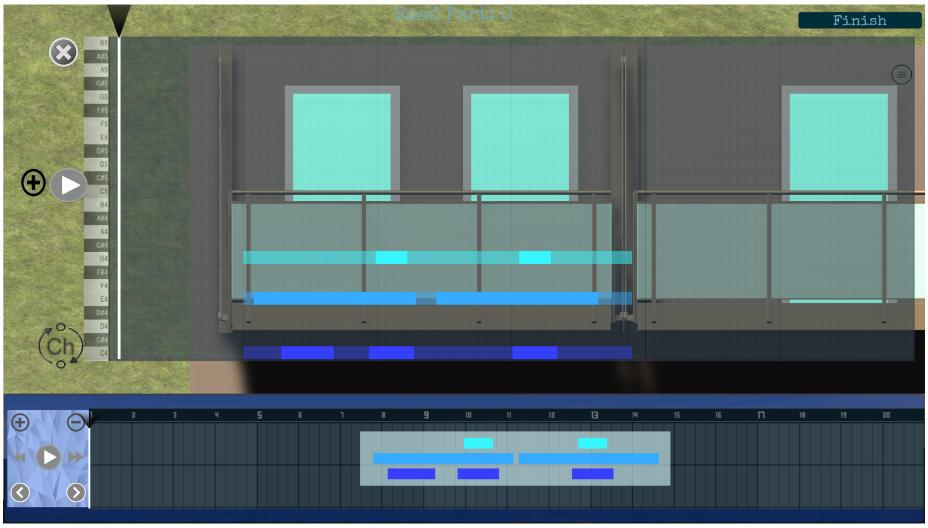


FIGURE 4.36: Initial screen of an individual DAW opened

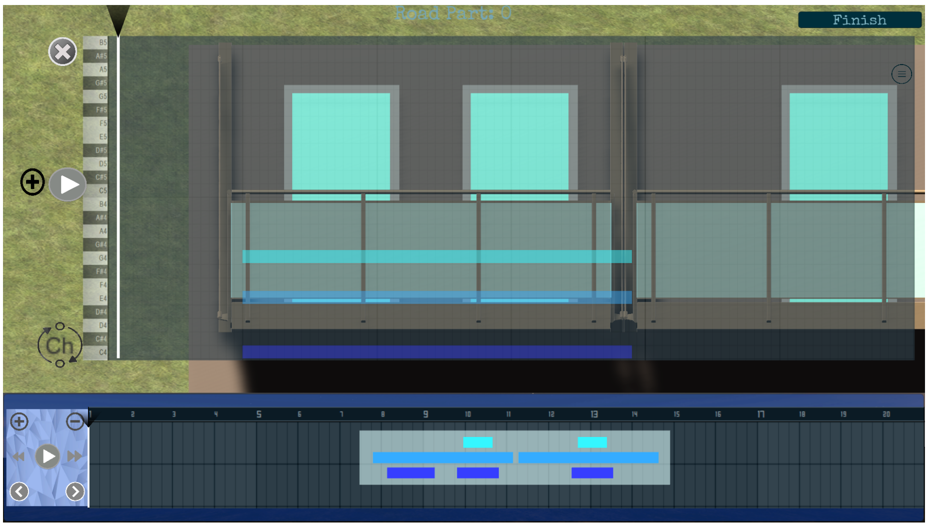


FIGURE 4.37: Individual DAW showing only the chord

Apart from the change button the buttons that are showing on the initial screen of the individual DAW are the close button 4.40a, the add button 4.40c and the play button 4.40b. If the user presses the close button the individual DAW closes and the camera returns to the position it had before. If the user presses the play button the notes that are showing will play. For example, if only the notes of the chord are activated then only the chord will play. In the chords the top note which is the lightest blue represents the number of floors, the middle note represents the height of the ground floor and the lower note and darkest blue represents the color of the building. On the melody, the top notes represent the windows, the middle notes represent the balconies and the lowest notes represent the doors and balcony doors. The melody notes share the same color as the chord notes. So, when both, chord and melody notes, are activated the chord notes are more faded and the melody notes are more opaque to be able to see both.

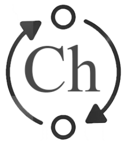


FIGURE 4.38: Change button

FIGURE 4.39: Individual DAW showing only the melody

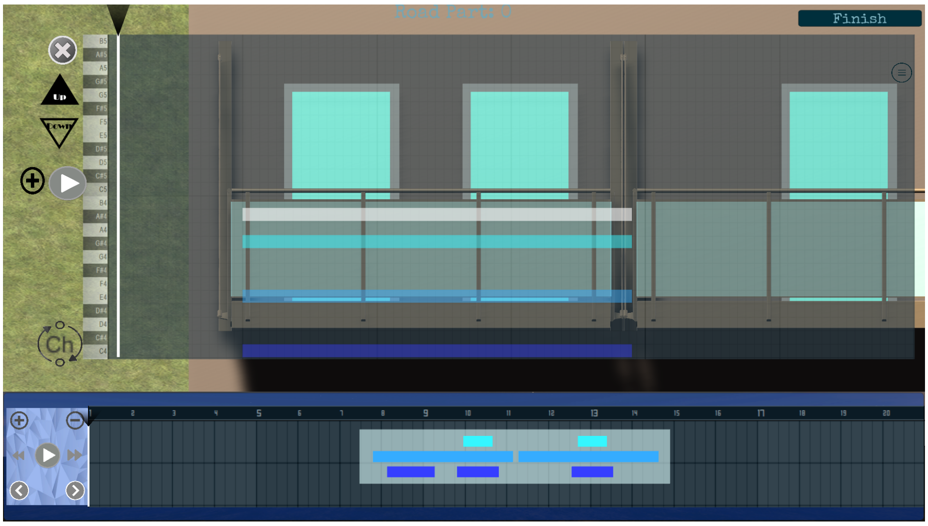
(A) Close

(B) Play

(C) Add new element

FIGURE 4.40: UI of the path's subdivision

To change a note the user has to click which note they want to alter. If the user presses a note from the chord then the up and down buttons appear 4.41. With these buttons the user can change the chosen note to make it higher or lower. If the user presses a note of the melody then buttons to move the note and the element left and right appear and also a button to delete it from the building 4.43. When the user changes a note they can see the change of the urban landscape happening simultaneously. Also, when a note is changing on the individual DAW, the corresponding note of the main DAW is changing too.



(A) View when a note of the chord is chosen

(B) Up button

(C) Down button

FIGURE 4.41: Up and Down buttons for the chord notes

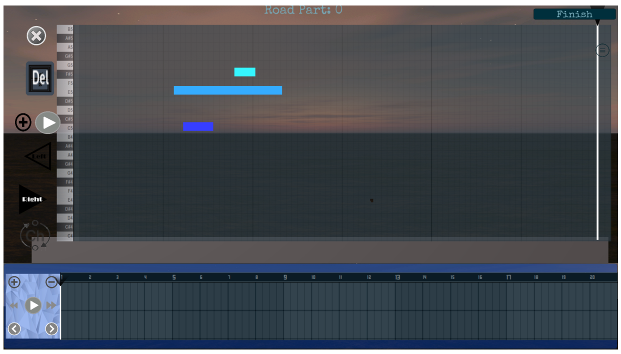


FIGURE 4.42: View when a note of the melody is chosen

The user has the ability to add a new element to the building by pressing the add button 4.40c. To do that the user can press the plus button and a new window appears 4.44. This window asks the user to choose in which floor the new element will be built. When the user chooses the floor and presses the check button, then the windows closes and new buttons appear on the left of the screen with all the element options that the user can choose 4.45.

(A) Up button

(B) Down button

(C) Delete the chosen element

FIGURE 4.43: Left, Right and Delete buttons for the melody notes

FIGURE 4.44: Window for choosing a floor when the user wants to add an element

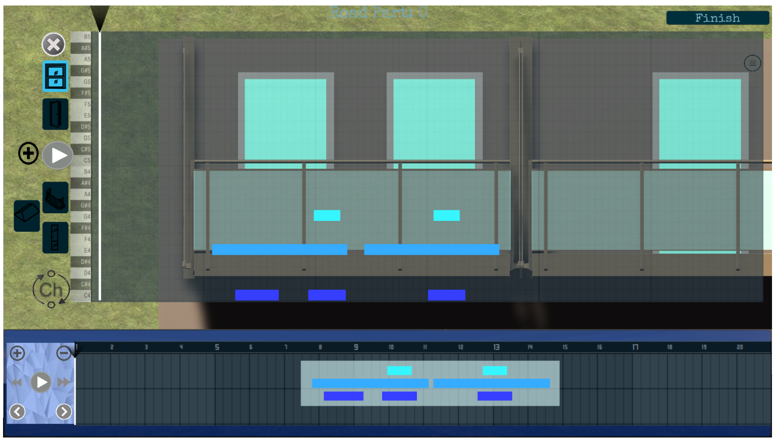


FIGURE 4.45: Buttons to choose which element to add

4.6.2 Components and implementation method

When the user press the button "Music" 4.46 to enter this subdivision the function of the translation is activated.

This function firstly counts how many different roads exist, meaning if there are two roads named road1_1 and road1_2 then they are considered two as two different parts of the same road, and their main number(ex. in road1_2 the main number is 1) is stored into a list called "roadc". Then a for-loop initiates looping as many times (x times) as the numbers that the roadc list contains (for(x = 0; x < roadc.Count; x++)). Every time in the for-loop, there is a search through the DAW canvas mentioned on the previous subdivision to find how many buildings are in the road parts with the same main number as the roadc[x]. This number is stored in the variable called "countit". This variable will be our size guideline to create all the arrays needed for the translation. It is important to state here that every translation of every building will create two identical results in two different DAWs because there is the main DAW that will have all the chords and melodies created by all the buildings, but also every building has its DAW with only its chord and melody shown in a more detailed way. So we need arrays to connect the building with the main DAW's chord that represents it, but also to manage to translate the color, the height of the ground floor, and the height of the building (number of floors) according to all the other buildings of the same road.

FIGURE 4.46: Menu before music subsection

The main arrays created are an array for the buildings of every road part of the road (called "realbuild"), an array for the corresponding chords of these building on the main DAW (called "buildingch"), and an array for every color of every building (called "buildingcolor"). After creating these empty arrays, we search again through the DAW canvas, and we fill them with every building's information of every road part of the road we are in, using the *AudioChord* script mentioned previously. While this is happening, the road's average number of floors and the average ground floor height are being calculated. Now we concentrate on color.

At first, we split the colors of all the road's buildings that we have stored into the "buildingcolor" array into two categories according to their saturation. If their saturation is higher than 0.625, they will belong to the "up" category, which means they will be translated into the higher octave. If their saturation is lower or equal to 0.625, they will belong to the "down" category, and they will be translated into the lower octave. Then for both of the categories, we follow the translation method and arrange the colors in order. Then, the chord's bottom note and the notes of the doors and balcony doors are created to both DAWs.

After that, the translation of the height of the ground floor follows using the average height according to the translation method. The 3rd factor of the chord and the notes of the balconies are created to both DAWs.

Finally, the translation of the number of floors is happening by using the average number calculated earlier. When finished, the 5th factor of the chord and the notes of the windows are added to both DAWs.

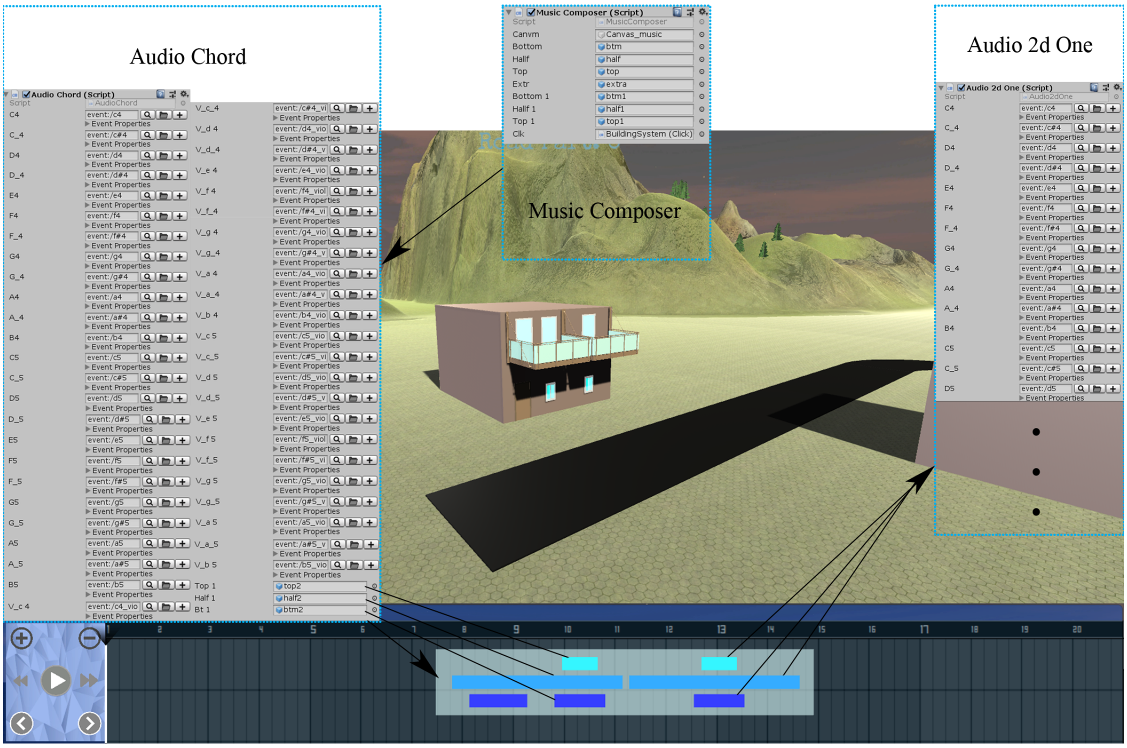


FIGURE 4.47: Music translation on the main DAW

The translation of the color results in a number representing the note and the translation of the other two factors results in a number that shows the distance that the note will have from the note of the color. To create the translated notes through these numbers to the main DAW, we call the corresponding function for every part of the chord or every kind of element (windows, doors, balconies), and all these functions are located to the *AudioChord* script. Also, the element's translated note has a script attached to it called *Audio 2d One*, which will define its note and trigger it through a 2D trigger 4.47. To create the translated notes using these numbers for the individual DAW, a world space canvas, of every building we call similar functions situated to the *MusicComposer*, the same script that the translation is happening. Within these functions, the chords' displays are created, placed to the right position, and inserted as children to the DAW of the side of the building translated. To be able to listen to the notes that these displays represent, every display has a script on it called Audio One, which has all the notes and the functions needed to trigger and stop the note from playing. These notes have a different script than the notes of the main DAW because they will be activated through a 3D trigger 4.48.

When the initial screen of the music subdivision is launching then the translation is finished. The main screen is divided in two parts and there are two cameras activated on the same time. The top part that covers 75% of the screen is used by the camera that allows the user to move and interact with their surroundings. The bottom 25% of the screen is used by the camera that shows the screen space - camera canvas that has the main DAW 4.28.

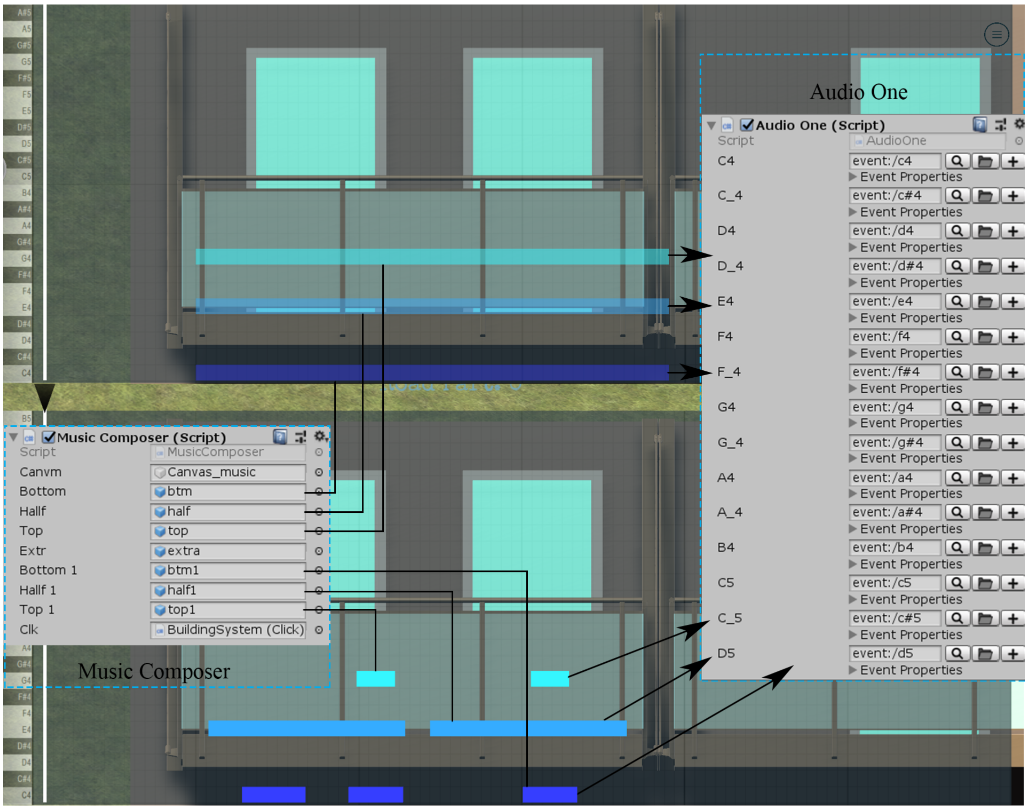


FIGURE 4.48: Music translation on the individual DAW

The main DAW has a control panel 4.31 on the left side and on the rest part of the DAW one side of one part of the road is showing each time. The control panel has the buttons mentioned earlier. These buttons are controlled through the button1 script.

If the user presses on the face of a building then its individual DAW appears. Everything that the user can change through this DAW and every button pressed is happening trough the *PlayEach* script which is attached to every individual DAW.

When the user presses a note a function is choosing which buttons to show. If the user presses a note from the chord then the up and down buttons appear 4.41. On the other hand, if the user presses a note of the melody then the left and right buttons appear and also a button to delete the element that this note represent. 4.43. To manage all these situations, every note has a tag that defines what it represents. There are three tags for the chord, a tag for the number of floors, a tag for the height of the ground floor and another tag for the color. And there are four tags for the melody, a tag for the window, a tag for the door, a tag for the balcony and a tag for the balcony door. Then, depending on which button the user will press a function starts running.

In that function there are different sub-functions depending on which note the user has chose. For example, if the user wants to make the note of the color higher then all the notes of the chord and the melody must go higher according to the translation method 3 and also the color of the building must change. On the other hand if the user choose to make the top note of the chord higher then only the notes of the windows must go higher and also the number of floors of the building must change. So, every situation has a different outcome. Except from all that whenever a note is changing on the individual DAW the corresponding note of the main DAW must change too. This also happens through the functions of the *Playeach* script 4.49. Another option the user has is to add a new element to the building. When the user chooses a new element to be added, the choice is being processed on the *Playeach* script where functions by following similar steps to those in the *MusicComposer* script where the notes where created create the note needed to both DAWs, but also create the element and add it as child to the selected side of the building.

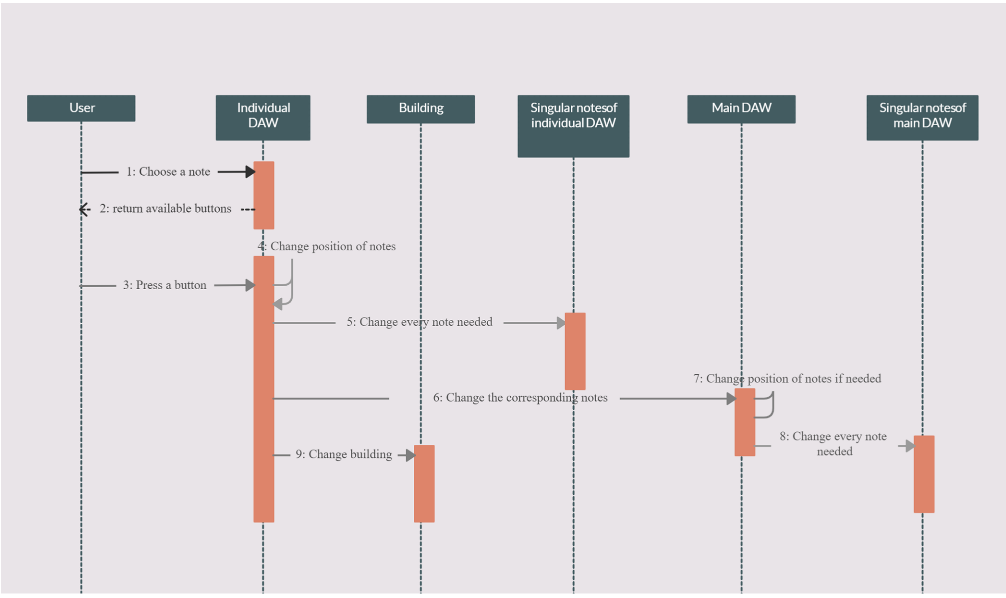


FIGURE 4.49: Flow diagram of the changes on the individual DAW

Chapter 5

Evaluation

5.1 Introduction

In order to understand in what degree MusiCityX platform achieves our objectives from an architect's perspective and generally how the user's experience is, we contacted undergraduate students of every level and graduates of the School of Architecture of the Technological Educational Institute of Crete to try the platform and evaluate it.

5.2 General procedure

The evaluation was conducted through an online meeting and a questionnaire. Every participant was first watching the demo video created to show what this platform can achieve on the online meeting. Next, the participants were asked to complete two tasks.

The first and easiest task was to make a road and a single building. The participants could create the building, they could freely choose the dimensions, the color, and if they wanted a roof, and then also they could add as many windows, doors, and balconies as they liked. The only parameter that was not totally free to choose was the number of floors that had to be between one and three floors. When they reached the music level, they were guided to try every change on the notes and the chord, and simultaneously an explanation was given considering the theoretical approach of this translation. They were guided through the platform when needed to help them get familiarized with the program.

The second task was to create a road and add many buildings between one floor and three floors and one building that was higher than seven floors. The color had to be soft, they could choose whatever color they wanted, but they had to use the same color for every building. The concept behind this task was, on the one hand, to see how the participants reacted to the platform now that they used it once before because there was no guidance to this task, but also to understand how every building's music translation depends on the other buildings of this road and how the disturbance of coherence is translated. When the participants reached the music level and listened to the musical footprint of this road, they were asked whether they could listen to the problem of the music's coherence. They could also listen that a building with the same number of floors created earlier translated differently to this example. Through this example, the participants could understand that a problem in coherence was translated through the whole musical theme of the road and not only through one building's chord. We discussed these results, and then the participants were asked to try changes on the buildings that could result in a more harmonic musical piece and, therefore, a more coherent urban scenery.

When finished, the participants were asked to express their opinion on the MusiCityX project. Their responses were recorded or written down. Finally, the questionnaire was sent to them to answer it after ending the call.

5.3 Demo

In order to offer to the participants of the evaluation a chance to watch a demonstration of MusiCityX's abilities, a video with many examples of usage was created.

The first example created was one building with three floors (counting also the ground floor) with many added elements on it. While creating everything needed for this building the steps (subdivisions) that the user must take were explained. This exampled is focused to demonstrate the MusiCityX tool's user interface. So, there is a demonstration of the changes that can be made through the individual Digital Audio Workstation (DAW). Then there is a small demonstration on the abilities of the UI of the main DAW which is considered a part of the first example.

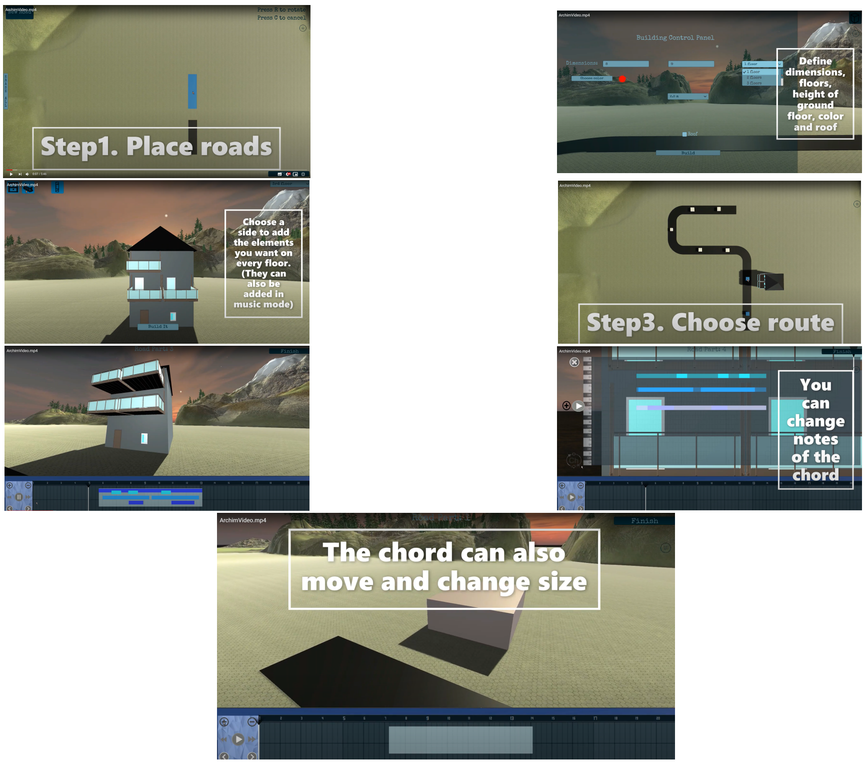


FIGURE 5.1: Example No.1

Subsequently, the notes of the chord, the notes of the melody and what they represent are explained. Afterwords, there are some examples that focus on each note of the chord separately. The first one of these examples involves buildings with the same parameters except the color.Moreover there are four buildings without added elements, two buildings on the left side of the road that have a soft tone of red and blue respectively and two buildings on the right side that have the same hues but are more intense. This example shows how the colors are divided in two categories and translated into two different octaves according to the intensity of the color, but also how the colors of every category are placed in an order. The resulted music is two chords on the left side of the road that belong to the lower octave and the first is higher than the second and on the right side of the road there are the same chords but an octave (eight notes) higher.

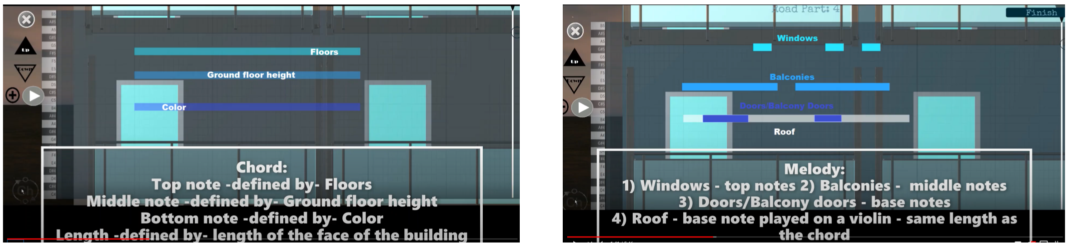


FIGURE 5.2: Explanation of notes

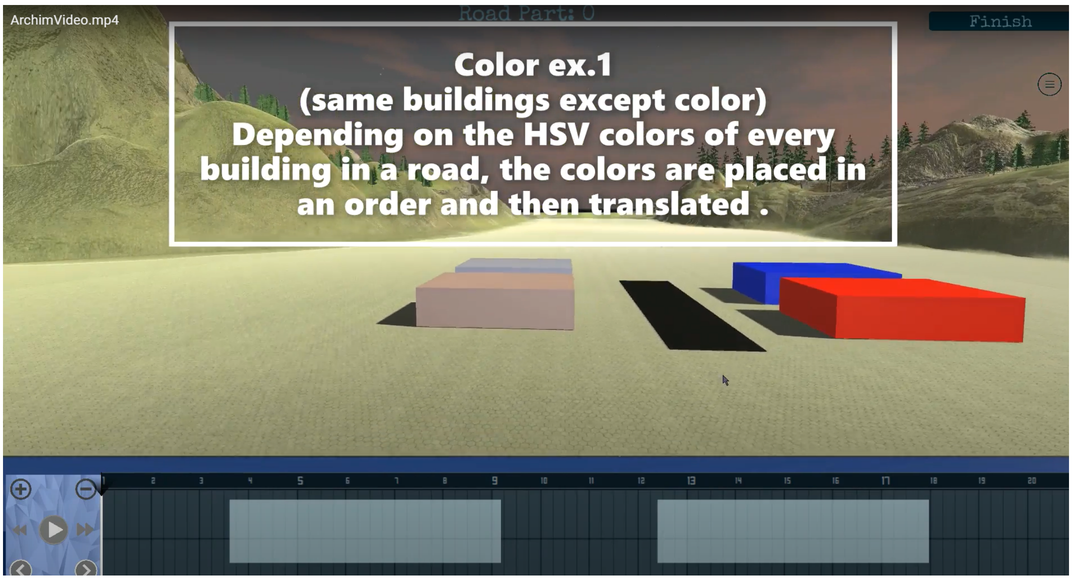


FIGURE 5.3: Example No.2

The next example involves four buildings without added elements with every parameter except from the height of the ground floor the same. Every building has a different height of ground floor (HGF). The heights used are 2.5m, 3m. 3.5m and 4m and they are all the possible heights that MusiCityX tool offers. This example shows how the change of the height of ground floor affects the same chord. The music translation of this example resulted into four chords where every one's middle factor is one note higher that the previous.

After explaining the other notes of the chord, now we need to explain how the note of the number of floors affects the chord. The number of floors affects greatly the chord and it is the parameter that is much more affected from the

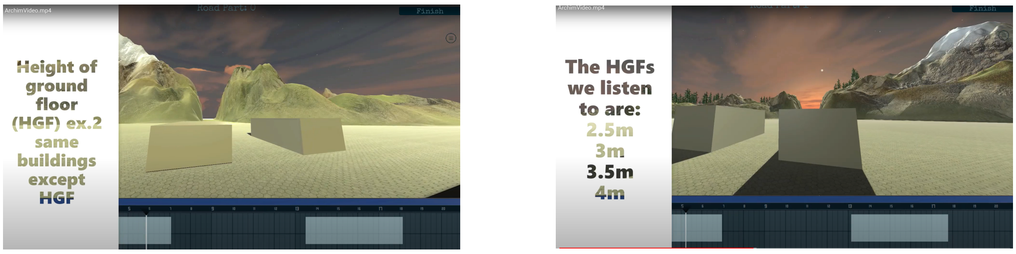
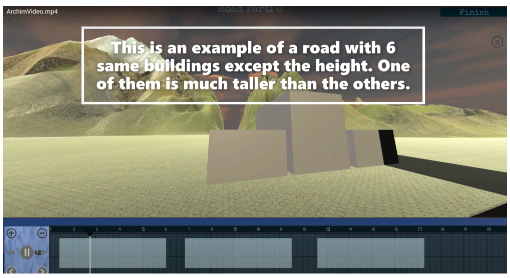


FIGURE 5.4: Example No.3

corresponding average of every road compared to the height of the ground floor. Also, the number of floors can vary from one to ten in comparison with the height of the ground floor that has only four possible conditions. All things considered, more than one examples are needed to explain this parameter's effect to the final music theme. Firstly, we show an example of three buildings without any added elements that have one floor difference each. The music result is a chord that escalates in terms of the top factor, because every chord's top factor is one note higher. The second example that focuses on the number of floors contains six buildings where one of them is much taller than the others. This examples indicates the way MusiCityX tool presents a coherence problem. In this translation, all the buildings of the road sound distorted, because all the buildings affect and are being affected by the coherence of the road. Additionally, the extra notes that are added to the chord if the number of floors is much lower or higher than the average is presented.

(A) Example No.4



(B) Example No.5 left side

(C) Example No.5 right side

FIGURE 5.5: Examples for the number of floors

The next example shows how the order of choosing the road parts changes the direction we move through the route and eventually how this affects the order the chords and the notes of the melodies are playing. To be able to show

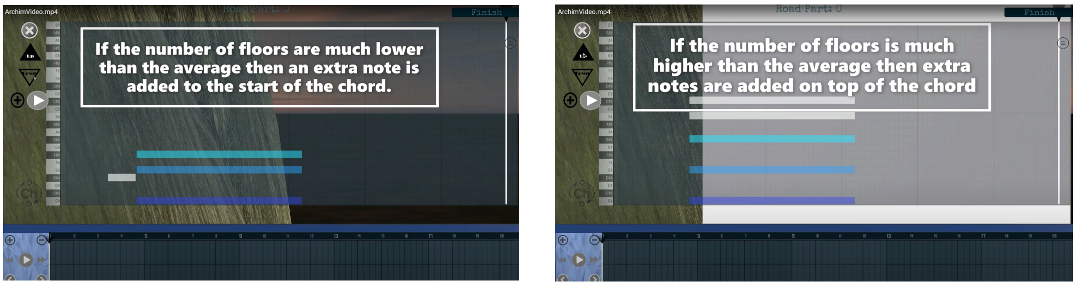


FIGURE 5.6: Demonstrations of extra notes

this, two roads that create a corner are placed and one building with a door and a window is placed on one of the roads. Then we choose a path two times and every time we choose the roads with a different order. The music themes differentiate as the window's note is playing first on the first example and the door's note is playing first on the other and so the melody is changing.

FIGURE 5.7: Example No.6

This part of the demo explains the three main functions that MusiCityX has to offer. These functions can be combined to achieve the refinement of every route the user will create.

The first of these functions is the ability to change the notes, the length and the position of the chords to achieve more coherent buildings in a road. To display this ability, there is an example of three buildings without any added elements very different from each other. The procedure of improving their initial disharmonious music theme that represented the absence of coherence is presented as well as the resulted improved music theme and refined scenery. This also proves that the coherence is not necessarily the result of many identical buildings but the balance between diversity and uniformity.

The second function is the ability to delete, add or change the position of the notes corresponding to the elements of the building (windows, balconies, doors, balcony doors) to refine the exterior of a building. The example created was

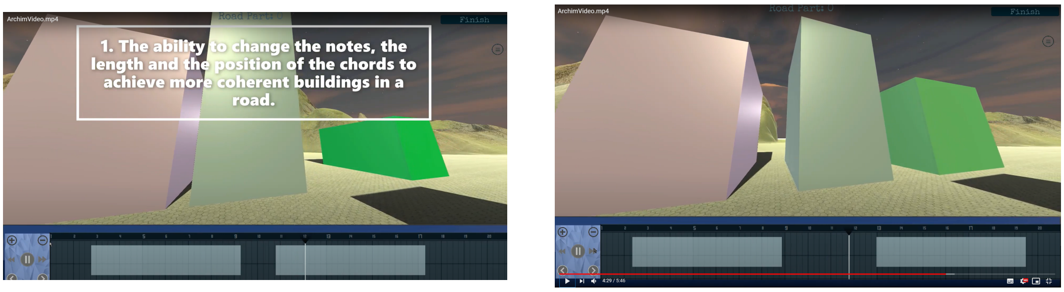


FIGURE 5.8: Example No.7

one building with four floors (counting also the ground floor) with many added elements on it. The melody created by the translation of the building's elements was chaotic without any rhythm, but the chord was a simple major chord. Our translation method is changing depending on the parameters of the other buildings on its street and the average they create (except from the color and the length of the face of the building) and since there aren't other buildings the average is defined by this building so its translation is the most average chord. This exampled is focused to the elements of every building, the windows, doors, balconies and balcony doors and how these can be manipulated to create a rhythmic melody.

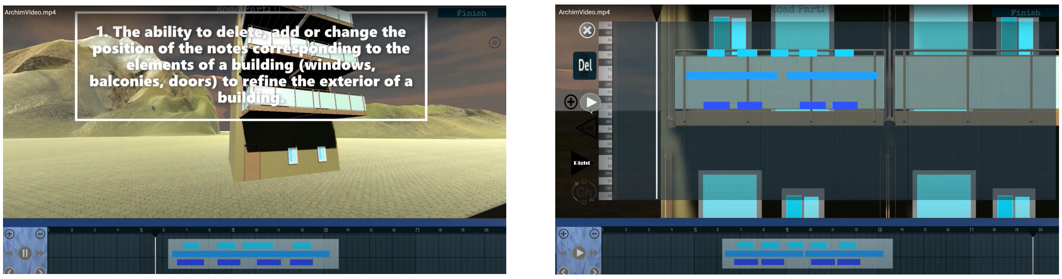


FIGURE 5.9: Example No.8

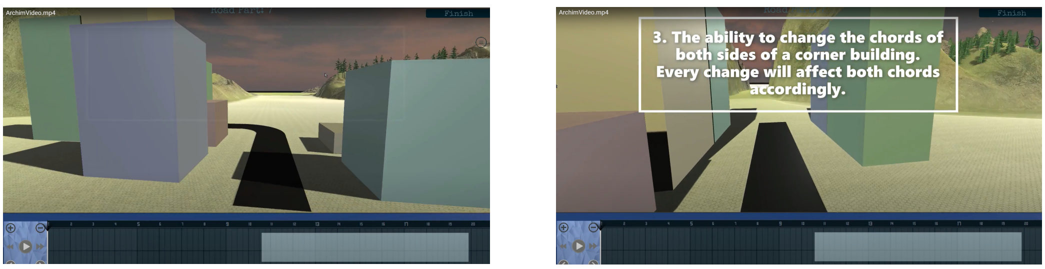


FIGURE 5.10: Example No.9 Roads Before

Lastly, the third function is the ability to change the chords of both sides of a corner building and every change will affect both sides accordingly. In this example there are two roads that create a corner. One road has shorter buildings than the other and there is a building placed on the corner of these two roads. The example is showing how the user can change the chords of both of the sides of this corner building in order to make it more coherent for both of the roads it is a part of.

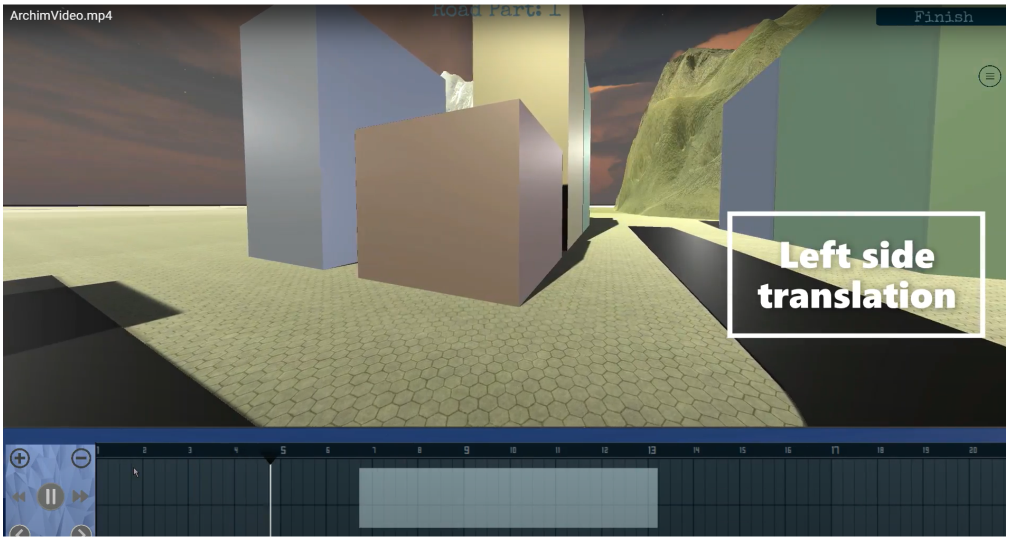


FIGURE 5.11: Example No.9 Corner building before

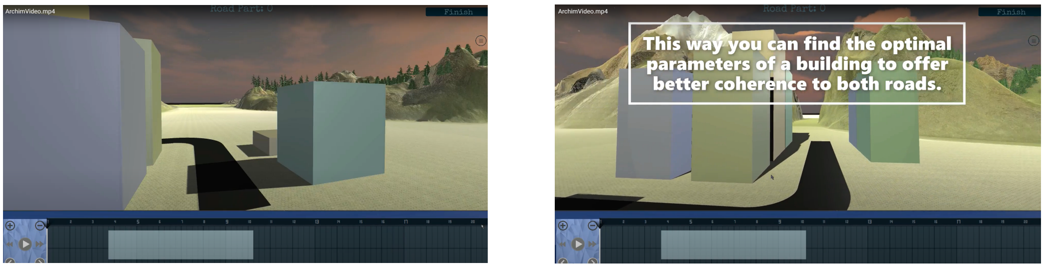


FIGURE 5.12: Example No.9 Roads After

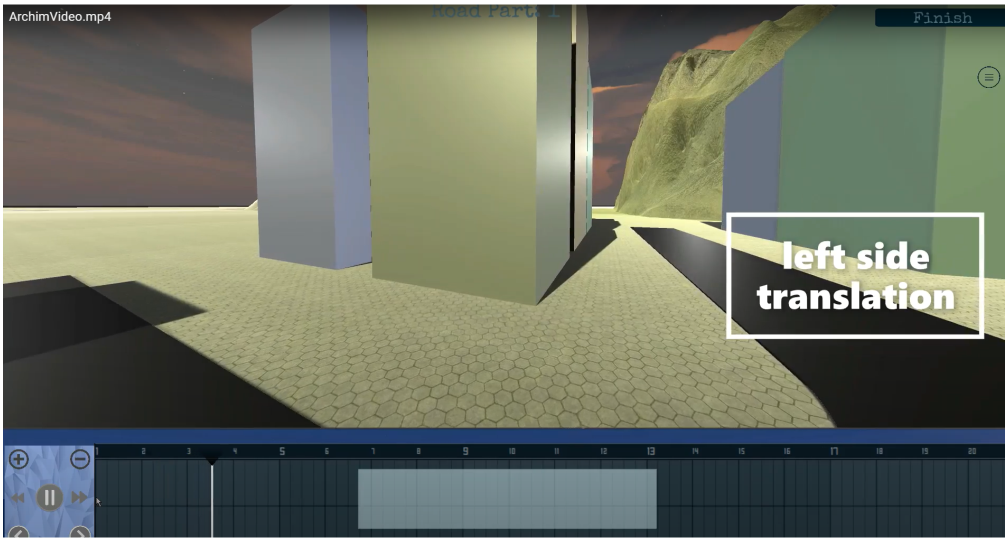


FIGURE 5.13: Example No.9 Corner building after

5.4 Questionnaire

The questionnaire was divided into three sections. The first section consisted of two questions that aimed to define the education and knowledge of the participants in architecture and music. The next section had questions of the

NASA Task Load Index (NASA-TLX) assessment tool. The final part consisted of two questions for assessment of the layout of the platform and of the total project.

On the first section the first question was "What is your level of study in architecture?" with the results shown in Fig. 5.14 . The next question was "At what level do you think you know music?" with the results shown in Fig. 5.15 .

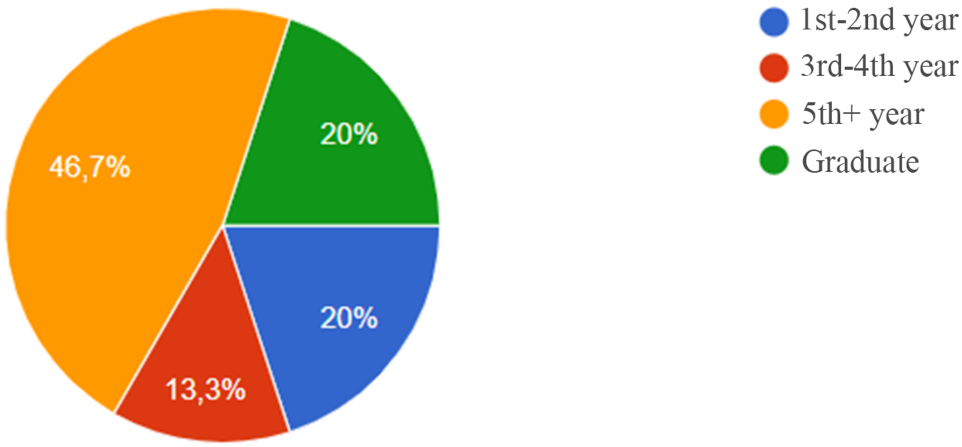


FIGURE 5.14: Level of study in architecture of the participants

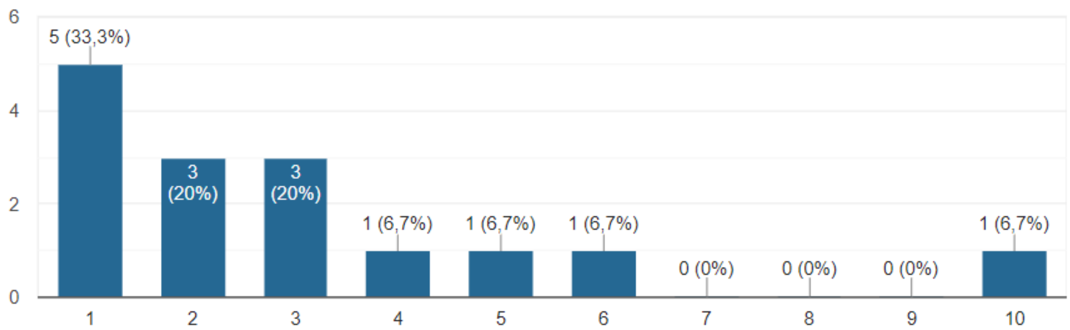


FIGURE 5.15: Level of music knowledge of the participants

Afterwords the participants answered four questions of the NASA Task Load Index (NASA-TLX) assessment tool. The first question was "(Mental demand) How mentally demanding was the tasks?" 5.16. The second question was "(Performance) How successful were you in accomplishing what you were asked to do?" 5.17. The next question was "(Effort) How hard did you have to work to accomplish your level of performance?" 5.18. And the final question "(Frustration) How insecure, discouraged, irritated, stressed,and annoyed were you?" 5.19. The answers are shown below.

On the final section, the first question was "How would you rate the difficulty of this platform in terms of the layout?" 5.20 and the next question "To what extent do you think this program achieves its purpose?" 5.21. The answers are shown below.

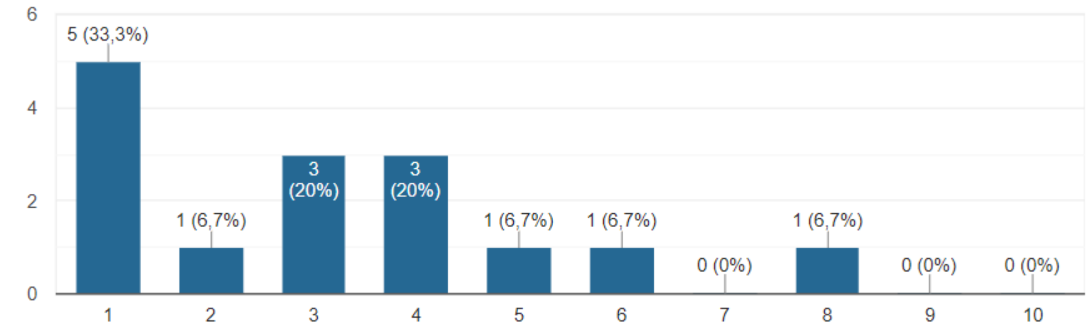


FIGURE 5.16: Mental demand

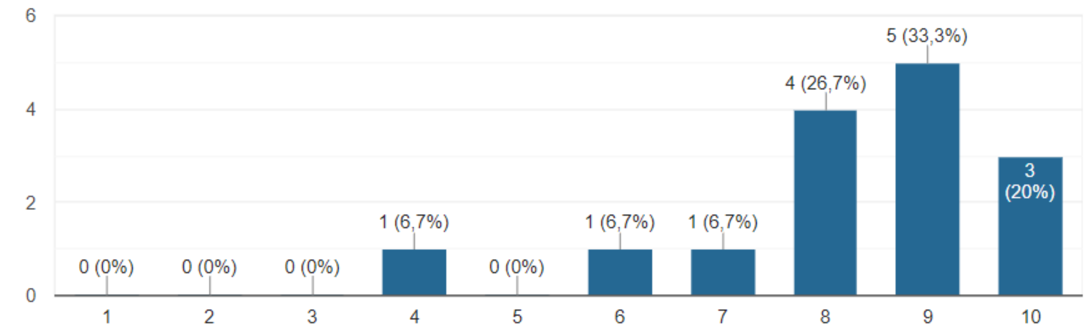


FIGURE 5.17: Performance

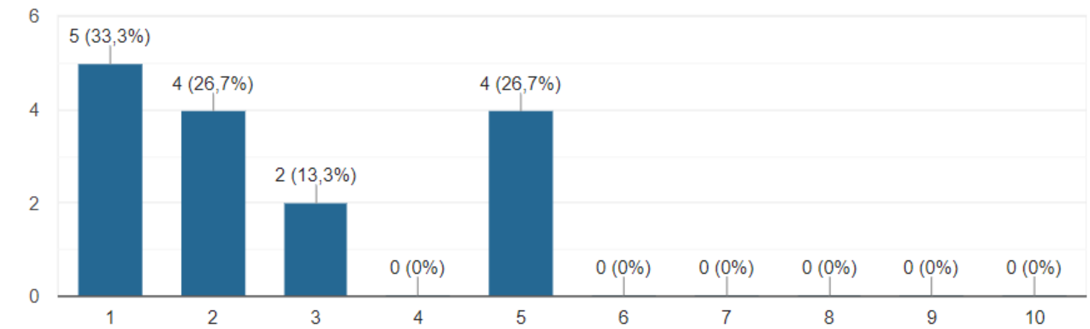


FIGURE 5.18: Effort

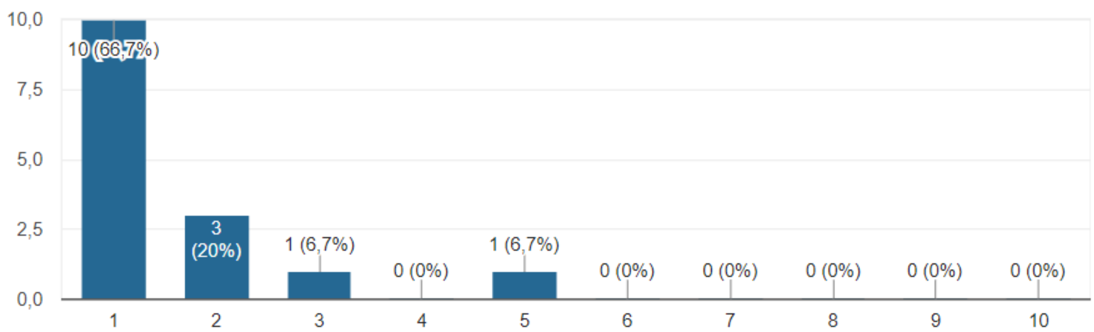


FIGURE 5.19: Frustration

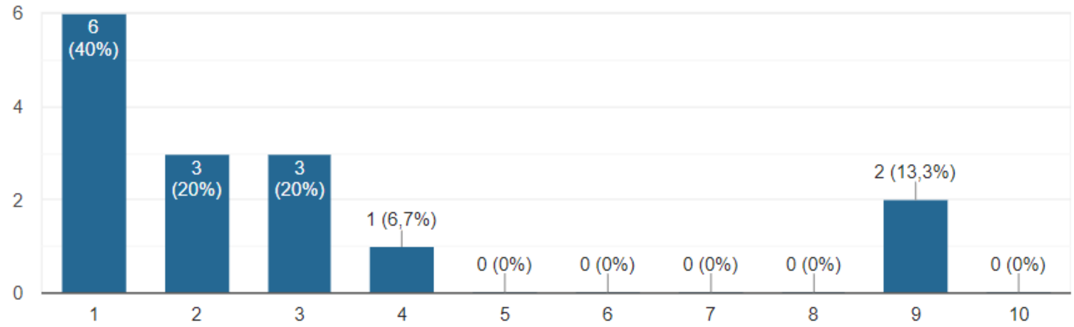


FIGURE 5.20: Rating the difficulty of the layout

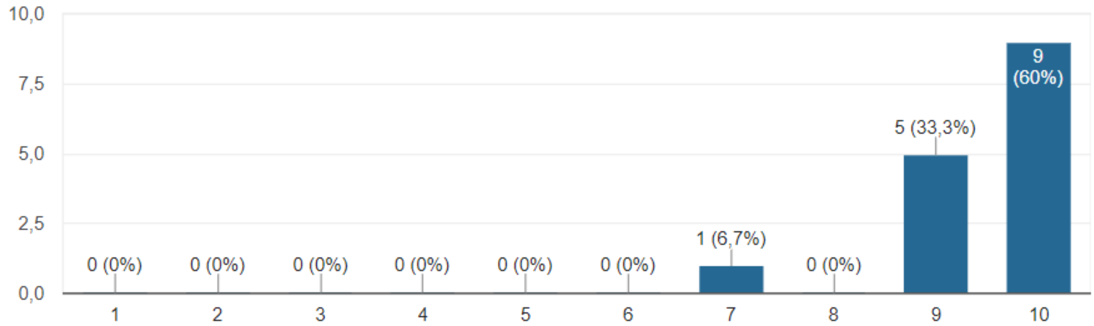


FIGURE 5.21: Level of achievement of its purpose

5.5 Opinions of the participants

The sharing of these opinions was the most valuable part of this evaluation as it offered not only feedback about the experience of the usage of this platform, but also an insight to the extend of the possibilities the continuation and development of this tool can have.

As far as the layout and user interface and experience, some of the participants thought that it reminded them Twinmotion which is a Real-time immersive 3D architectural visualization tool. All the other participants said that it is very different from every other program they used. On one hand, some of the participants found it difficult on the first example to understand how the camera moves and especially how you can move a building before you place it, but it was not a problem for the second example. On the other hand many participants said that they prefer that the layout, the buttons and the camera movement does not remind them of other architectural programs, because it has a very different scope of application which fits the different layout.Finally, the way that the menu changes every time the user has finished a step of creating the urban landscape, according to participants, guides the user and it is helpful.

The opinions concerning the way we connected architecture and music were many and different. Below, there are some of the opinions that include every opinion stated by the participants.

- "It (MusiCityX) offers an experiential way to find and explain to others the faults of an urban landscape. And it could be developed to receive more variables".

- "This program reminds me of the quote "Architecture is frozen music"".
- "This music translation could be used not only to find coherent problems... If an architect wants to create a building as a statement and wants purposely to disrupt the coherence of a road, they can use this program too, to define the parameters of this building."
- "This program reminds me of jazz music, which was developed through the sounds of the city, and the music translation also sounds like jazz."
- "It can help to urban designing or to designing residential complexes or residential complexes with a mall into them, where the mall sometimes ends up huge compared to the residential buildings of this area."
- "I like the idea that as an architect when I have some primitive designing ideas, after trying materializing them, I would take an extra step to see how harmoniously my idea works on the total frame."
- "The translation is not kept in a basic level and that softens the result, it makes it sound nice but also it let's you hear where something is wrong."
- "It is very productive to study everything that involves the principle of composition and comparing one to another. Because ostensibly through seeing their results they may seem irrelevant, but on the most basic level the way I compose something is the same for everything, either it is a theatrical piece, a painting, a statue, a building etc. If I take a result and go many steps back before this result I will find some compositional ideas that can lead me to another result that concerns a different art or something different like a meal."
- "The way that I can draw ideas to differentiate or harmonize a design through music is very interesting."
- "It (MusiCityX) is ideal to study the strictly urban landscape... If you add many buildings of an urban landscape you can see a monotony (for example the placements of elements) window, balcony, window, balcony and you can understand how a city is regulated. If you can translate extra parameters like curves you will leave from the strictly urban landscape and you will enter something else with a new type of music. "
- "I would use it again even as a tool to think something architecturally but also from a directing (assembling the scene) point of view."
- "When we design we normally use the visual stimulus, but I prefer the experiential method. When the architect designs experientially, they can make themselves experience how they feel in a building, or in this case in a neighborhood, in a set of roads or in a path, all the senses must be combined and all the senses together must end up in one result and one conclusion. So, the fact that this program can offer you apart from a visual stimulus, also an auditory stimulus, it is a plus and for me the goal would be to be able to combine all the senses... To understand the feeling of a

neighborhood you need many senses and also memories, but the most powerful senses, I personally think, are the vision and the hearing and also the most accessible ones when you visit a place."

5.6 Expert evaluation

In order to understand whether MusiCityX provides a reliable and functional platform , Fotis Giariskanis, a graduate student of the Department of Electrical and Computing Engineering of the Technical University of Crete that developed the earlier research on the subject, tried the platform and evaluated it. This evaluation was focused more on the technicalities of this project.

- The usage of one central DAW and one extra more detailed DAW for every building was a very helpful insertion for the user. Also, according to F.Giariskanis, the DAW interface was a successful simulation of an existing DAW program.
- He approved the user interface and considered it easy to use. Based on his evaluation, the platform's layout had evolved and the addition of color in the buildings was pleasant.
- In relation to the translation method, he considered that it was a step forward towards a more inclusive connection between architecture and music, meaning that it translated more architectural parameters than before and this could result to a translation that involves all the parts of an urban scenery.

5.7 Results of the evaluation

The evaluation results are made by combining the questionnaire's information and the participants' opinions. Also, there was research on each individual's answers on the questionnaire to get information about how the music knowledge and the level of study on architecture affected the results.

5.7.1 Positive

- Almost all of the participants answered they performed above average with a below average effort and mostly the least amount of frustration. This means that the MusiCityX tool was easy to handle.
- Many participants said that they liked the translation method because it offers them a translated musical theme that is not chaotic, like they expected, but sounds nice and some of them said that it reminds them of jazz music. This means that the MusiCityX managed to translate the beauty of the urban design in a harmonious way following the classical music theory.

- The way the coherence problems appears to the user, meaning that there is not one chord that sounds very wrong but the total musical theme of the road is "strange" does not force the architect to change one building, it maintains the artistic liberty of the designer which lets them choose. That is why more than one participants stated that they would use this program not only to create a coherent landscape and musical piece, but also to create deliberately a more distorted scenery and sound.

- MusiCityX manages to offer inspiration to the youngest participants who were on their second year of studies to the graduates. Both of these groups and the groups between expressed their will to use it again to get inspiration and to think the proportions of the buildings they would design. This is the ultimate goal of combining to arts. There is no way of determining if the translation method is the optimal, because there is not a correct method of combining two arts, but since this combination with this layout and user interface was a tool that people of the field of architecture would use it again to get inspired is a very positive result.

5.7.2 Neutral

- It defines parameters that concern a strictly urban landscape. On one hand that was our goal, but on the other hand it does not have an option for more complex buildings that some participants pointed which would be a good addition.

- It has a user interface that does not look like most of the architectural programs. Many of the participants said that they don't mind, some of them would prefer some changes to be closer to what they are used to, others prefer it as it is.

- The people with more knowledge on the music sector were able to understand deeper the translation method and how to alter the translated music to make it sound more harmonious. But, the people who did not know anything about music theory, they all would hear the coherence disruptions and they also were able to make changes. So, it is logical that the people who already use music to express themselves would found a better use of this program, but since the music knowledge does not exclude someone from using MusiCityX this result is neutral.

5.7.3 Negative

- The camera movement on the music section, when the music is not playing made it sometimes difficult to the participants to see the building while changing parameters from the main DAW. This happened because the rotation of the camera was defined by the movement of the computer mouse. So, when the user was moving the mouse on the bottom left side of the screen to press a button, the camera was moving too and the user had to be close to the building to be able to see the results simultaneously.

- It has only one option for every key element. Moreover it has one option for a window, one option for a door, one option for a balcony door and also one option for a balcony. Some participants would prefer to have more options to choose from.

Chapter 6

Conclusions - Future work

6.1 Conclusions

This thesis described an interactive tool called "MusiCityX" that works as a medium between music and architecture. This tool can offer the user the ability to create or alternate an urban landscape through vision, but also through hearing.

Moreover, through MusiCityX, the user can create an urban scenery and not only listen to its musical footprint but also change the music and simultaneously watch the scenery change. Architecture and music are both arts, and that is why we can not define a correct method of connecting these two fields, but they are both based on mathematics, and they have their own rules of composition, so there is a limit to the numerous correct methods of connecting them. This research tried by following these rules and defining a more targeted approach to the urban architecture to create a tool where the user can find and resolve parts of incoherence, but also experiment. MusiCityX is a platform that offers to the user the interface needed to achieve these goals in a controlled environment where the rules of both fields are followed.

The connection between these two fields for the purpose we defined was successful. The problems in the coherence of a road were translated in the musical piece, but the resulted chords were still following the rules defined by the music theory. Furthermore, this means that we managed to translate the inconsistencies while maintaining the freedom of an art such as architecture and music. We also observed, through the evaluation of the MusiCityX tool, that the user interface was friendly to the participants. The MusiCityX tool's functions offer the ability to translate and modify entirely a strictly urban envi- ronment and partially another more complex one.

Nevertheless, the connection of these two arts through their common elements, mathematics and the concept of composition, can lead us to an unknown ground of creativity. We can know try to listen to our cities or try creating our cities only through listening.

6.2 Future work

The MusiCityX system developed in the context of this thesis could be improved and further expanded, specifically, through the following suggestions:

- It would be useful to add more architectural parameters in the program and the translation method, for example curvature, to include more complicated buildings. Every building includes the parameters defined by this translation, so every building can be partially translated by MusiCityX tool. But, if there were more parameters affecting the translation, then the user could hear the difference between a building with curvature and a simple straight building. So, our platform could then affect more layers of the architectural and urban design.

- More windows, doors, balconies, balcony doors and roof types can be added to offer the user more choices when adding elements to a building.

- Elements that exist in an urban scenery can be included and translated into music, for example benches and traffic lights, to create a completed urban system.

- Saving and opening an MusiCityX scene would be a great help to impel the user to create a bigger and more complex urban network. If the user knows that their work will not be able to be saved then they would not easily create a time consuming urban scenery.

- A great addition would be saving the corresponding musical pieces created by the translation of the urban scenery in .wav, .mp3, .mp4 format, commonly used by media players.

- Another addition helpful to develop the connection between urban architecture and music would be to add parameters involving the citizens' allocation and movement, where they mostly are, and the routes they mostly walk. These issues are significant for an urban designer and it would be beneficial to include them.

References

[1] Ana Júlia Pinto and Antoni Remesar. *Urban cohesion: a guiding concept for new urban realities*. URL: https://www.researchgate.net/publication/299885966_Urban_cohesion_a_guiding_concept_for_new_urban_realities (cit. on pp. 1, 23, 24).

[2] Lucan J. *Composition, non-composition: architecture and theory in the nineteenth and twentieth centuries*. EPFL Press, Routledge, 2012 (cit. on pp. 1, 8).

[3] Amjad Almusaed and Asaad Almssad. *City Phenomenon between Urban Structure and Composition*. URL: https://www.intechopen.com/online-first/city-phenomenon-between-urban-structure-and-composition (cit. on pp. 1, 9).

[4] *Definition of architecture*. URL: https://dictionary.cambridge.org/dictionary/english/architecture (cit. on p. 1).

[5] Nigel Cross. "The nature and nurture of design ability". In: *Design Studies* 11.3 (1990), pp. 127–140. ISSN: 0142-694X. DOI: https://doi.org/10.1016/0142-694X(90)90002-T. URL: http://www.sciencedirect.com/science/article/pii/0142694X9090002T (cit. on p. 1).

[6] Panagiotis Parthenios. "From atoms, to bits, to notes – an encoding-decoding mechanism for tuning our urban eco-systems". In: Sept. 2013 (cit. on p. 1).

[7] Yujie Hong and Panagiotis Michalatos. "LumiSpace: A VR Architectural Daylighting Design System". In: SA '16. Macau: Association for Computing Machinery, 2016. ISBN: 9781450345484. DOI: 10.1145/2992138.2992140. URL: https://doi.org/10.1145/2992138.2992140 (cit. on p. 1).

[8] Christian Schneider, Anastasia Koltsova, and Gerhard Schmitt. "Components for Parametric Urban Design in Grasshopper from Street Network to Building Geometry". In: *Proceedings of the 2011 Symposium on Simulation for Architecture and Urban Design*. SimAUD '11. Boston, Massachusetts: Society for Computer Simulation International, 2011, pp. 68–75 (cit. on p. 1).

[9] Schoenberg A. *Fundamentals of Musical Composition*. Faber and Faber, 1967. URL: https://monoskop.org/images/d/da/Schoenberg_Arnold_Fundamentals_of_Musical_Composition_no_OCR.pdf (cit. on pp. 1, 10).

[10] Marc Aurel Schnabel, Tian Tian Lo, and Serdar Aydin. "Gamification and Rule Based Design Strategies in Architecture Education". In: Dec. 2014. DOI: 10.13140/2.1.5150.3689 (cit. on p. 2).

[11] Chia-Hung Tsou et al. "Immersive VR Environment for Architectural Design Education". In: *SIGGRAPH Asia 2017 Posters*. SA '17. Bangkok, Thailand: Association for Computing Machinery, 2017. ISBN: 9781450354059. DOI: 10.1145/3145690.3145726. URL: https://doi.org/10.1145/3145690.3145726 (cit. on p. 2).

[12] Lou Boves et al. "Multimodal Interaction in Architectural Design Applications". In: *User-Centered Interaction Paradigms for Universal Access in the Information Society*. Ed. by Christian Stary and Constantine Stephanidis. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 384–390. ISBN: 978-3-540-30111-0 (cit. on p. 2).

[13] Jean-Baptiste Thiebaut, Patrick G. T. Healey, and Nick Bryan-Kinns. "Drawing Electroacoustic Music". In: *ICMC*. 2008 (cit. on p. 2).

[14] *Unity (game engine)*. URL: https://en.wikipedia.org/wiki/Unity_(game_engine) (cit. on pp. 2, 19).

[15] *Dictionary*. URL: https://www.lexico.com/definition/composition (cit. on p. 7).

[16] John P. Millis. *Composition of Universe*. URL: https://www.thoughtco.com/composition-of-the-universe-3072252 (cit. on p. 7).

[17] Roger John Tayler. *Chemical element*. URL: https://www.britannica.com/science/chemical-element/The-atomic-nature-of-the-elements (cit. on p. 7).

[18] Dan Scott. *A Comprehensive Guide To Composition For Artists*. URL: https://drawpaintacademy.com/understanding-composition-for-artists/ (cit. on p. 8).

[19] McKella Sawyer. *What is Composition in Art?* URL: https://conceptartempire.com/what-is-composition/ (cit. on pp. 8, 25).

[20] Gary Meisner. *Phi and the Golden Section in Architecture*. URL: https://www.goldennumber.net/architecture/ (cit. on p. 8).

[21] Yüksel Demir Seyed Farhad Tayyebi. *Architectural Composition: A Systematic Method to Define a List of Visual Attributes*. URL: https://www.researchgate.net/publication/333684064_Architectural_Composition_A_Systematic_Method_to_Define_a_List_of_Visual_Attributes (cit. on p. 9).

[22] *Study on the Relationship between Architecture and Music*. URL: https://tinyurl.com/yythl6pl (cit. on p. 9).

[23] Peter Crossley-Holland. *Musical composition*. URL: https://www.britannica.com/art/musical-composition (cit. on p. 9).

[24] *Music Theory*. URL: https://2012books.lardbucket.org/pdfs/music-theory.pdf (cit. on p. 11).

[25] Jones R. Schmidt-Jones C. *Understanding Basic Music Theory*. CONNEXIONS (cit. on pp. 11–13).

[26] *Music scales – Definition*. URL: https://www.simplifyingtheory.com/music-scales/ (cit. on p. 13).

[27] *FL studio image line website*. URL: https://www.ableton.com/en/education/ (cit. on pp. 14, 20).

[28] *What is DAW*. URL: https://www.careersinmusic.com/what-is-a-daw/ (cit. on p. 14).

[29] *Digital Audio Workstation*. URL: https://en.wikipedia.org/wiki/Digital_audio_workstation (cit. on p. 14).

[30] *Music scales – Definition*. URL: http://theappendix.net/posts/2013/08/music-and-color-the-french-connection (cit. on p. 14).

[31] *Wassily Kandinsky: the painter of sound and vision*. URL: https://www.theguardian.com/artanddesign/2006/jun/24/art.art (cit. on pp. 15, 16).

[32] *Translating proteins into music, and back*. URL: https://news.mit.edu/2019/translating-proteins-music-0626 (cit. on p. 16).

[33] Chi-Hua Yu et al. "A Self-Consistent Sonification Method to Translate Amino Acid Sequences into Musical Compositions and Application in Protein Design Using Artificial Intelligence". In: *ACS Nano* 13.7 (2019). PMID: 31240912, pp. 7471–7482. DOI: 10.1021/acsnano.9b02180. eprint: https://doi.org/10.1021/acsnano.9b02180. URL: https://doi.org/10.1021/acsnano.9b02180 (cit. on p. 16).

[34] *Orchestra of Amino Acids*. URL: https://soundcloud.com/user-275864738/orchestra-of-amino-acids (cit. on p. 16).

[35] *ARCHITECTURE AS FROZEN MUSIC*. URL: https://archithrill.com/architecture-as-frozen-music/ (cit. on pp. 16, 17).

[36] *Music is liquid architecture; Architecture is frozen music*. URL: http://www.artandpopularculture.com/Music_is_liquid_architecture (cit. on p. 16).

[37] Mpiris T. et al. *Architectural and musical concurrences*. Pataki (cit. on p. 17).

[38] *MUSIC – HARMONY FROM APOLLO TO PALLADIO and MOZART TO JENKINS*. URL: https://www.thecultureconcept.com/music-harmony-from-apollo-to-palladio-mozart-to-jenkins (cit. on p. 17).

[39] *The harmony and lucidity in Palladio architecture is a matter of scale and not of size*. URL: https://angelsferrerballester.wordpress.com/2017/08/06/the-harmony-and-lucidity-in-palladio-architecture-is-a-matter-of-scale-and-not-of-size/ (cit. on p. 17).

[40] John Thompson et al. "The Allobrain: An interactive, stereographic, 3D audio, immersive virtual world". In: *International Journal of Human-Computer Studies* 67.11 (2009). Special issue on Sonic Interaction Design, pp. 934–946. ISSN: 1071-5819. DOI: https://doi.org/10.1016/j.ijhcs.2009.05.005. URL: http://www.sciencedirect.com/science/article/pii/S1071581909000688 (cit. on p. 17).

[41] Merrill Bradshaw. "Formalized Music: Thought and Mathematics in Composition: By Iannis Xenakis. Bloomington: Indiana University Press. 273 pp. Illustrations, bibliography, appendixes, index. Hard cover, $15". In: *Music Educators Journal* 59.8 (1973), pp. 85–88. DOI: 10.2307/3394288. eprint: https://doi.org/10.2307/3394288. URL: https://doi.org/10.2307/3394288 (cit. on p. 17).

[42] Solomos M. Iannis Xenakis. *IANNIS XENAKIS texts on music and architecture*. Psychogios (cit. on p. 17).

[43] *ARCHIMUSIC3D: Real-Time Graphical System for Reciprocal Transformations between Music and Refined Urban Architectural Design* (cit. on p. 18).

[44] *Reciprocal transformations between music and architecture as a real-time supporting mechanism in urban design. In: International Journal of Architectural Computing.* (Cit. on p. 18).

[45] *FMOD*. URL: https://en.wikipedia.org/wiki/FMOD (cit. on p. 20).

[46] *FL studio*. URL: https://en.wikipedia.org/wiki/FL_Studio (cit. on p. 20).

[47] *Revit*. URL: https://www.autodesk.com/products/revit/overview?plc=RVT&term=1-YEAR&support=ADVANCED&quantity=1 (cit. on pp. 21, 48).

[48] Matt Nowlin. *Inequality Between Neighborhoods on the Rise*. URL: https://www.savi.org/2018/09/11/the-growing-divide-between-rich-and-poor-neighborhoods/ (cit. on p. 24).

[49] Jacci Howard Bear. *HSV color*. URL: https://www.lifewire.com/what-is-hsv-in-design-1078068 (cit. on p. 28).

[50] *HSL and HSV*. URL: https://en.wikipedia.org/wiki/HSL_and_HSV (cit. on p. 28).

[51] Lisa Wilms and Daniel Oberfeld. *Color and emotion: effects of hue, saturation, and brightness*. URL: https://www.staff.uni-mainz.de/oberfeld/downloads/Wilms-Oberfeld2018_Article_ColorAndEmotionEffectsOfHueSat.pdf (cit. on p. 29).

[52] *ProBuilder*. URL: https://unity3d.com/unity/features/worldbuilding/probuilder (cit. on p. 43).