

3D ARCHITECTURAL VISUALIZATION USING UDK AS AN INTERACTIVE TOOL

By

DUNCAN MARTEL

SUPERVISORY COMMITTEE:

DEVANE, BENJAMIN – CHAIR

BARMPOUTIS, ANGELOS – MEMBER

WILLIAMS, ANNA - CONSULTANT

A PROJECT IN LIEU OF THESIS PRESENTED TO THE COLLEGE OF FINE ARTS
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ARTS
UNIVERSITY OF FLORIDA

2013

Summary of Project in Lieu of Thesis
Presented to the College of Fine Arts of the University of Florida
in Partial Fulfillment of the Requirements for the
Degree of Master of Arts

3D ARCHITECTURAL VISUALIZATION USING UDK AS A CUSTOMIZABLE TOOL

By

Duncan Martel

May 2013

Chair: Benjamin DeVane
Major: Digital Arts and Sciences

3D visualization in the field of architecture is an emerging discipline that has had a powerful impact on how architects present their work and communicate ideas to the client. However, the potential for what 3D software can do is still being realized in the architectural design process. This project proposes an alternative approach to the current design process by presenting a system that will develop a stronger relationship between the client and the architectural designer. By drawing from aspects in other fields of 3D design, specifically video-games, the project lays the framework for a system to be used by architects to immerse their clients in a proposed architecture and encourage interaction with this virtual environment. The project uses programs such as 3Ds Max, Photoshop, and Flash, to develop content for an environment realized in the Unreal Engine using the Unreal Development Kit (UDK).

ACKNOWLEDGMENTS

I thank my mother and father for everything, my girlfriend for her unending support, and my friends in architecture for showing me a different view of the world. Without any of these people, my life would be bleak and uninspired.

TABLE OF CONTENTS

ACKNOWLEDGMENTS -----	3
INTRODUCTION	
1.1 Project Introduction -----	5
RATIONALE	
2.1 The Current Architectural Design Process -----	6
2.2 Spatial Presence in a 3D Environment -----	7
2.3 Precedents -----	9
METHOD	
3.1 Overview -----	12
3.2 Construction -----	14
3.3 Design Process -----	16
3.3.1 Scenario-based User Testing Methods -----	22
RESULTS	
4.1 Project Screen Captures -----	24
4.2 User Reflections -----	26
CONCLUSION	
5.1 Discussion and Future Direction -----	29
5.2 Summary -----	31
LIST OF REFERENCES -----	32
BIOGRAPHICAL SKETCH -----	33

Introduction

1.1 Project Introduction

The purpose of the project is to create a system that will strengthen the communication between architect and client in the architectural design process. By using the Unreal Development Kit (UDK) software as the primary tool, the project demonstrates a working system of interactive visualization and customization. The goal is to give the architect and client a mutual vocabulary of design. By making use of a virtual environment, the architect can more effectively convey spatial presence in architectural design to the client.

As a professional, the architect is assigned the task of creating a construct envisioned by the client that satisfies both the client's need and the architect's vision. Although the architect has the final say in aesthetics and the technicals of decision-making, the client is the party that is paying for the service. Being the professional providing such a service, the architect must satisfy the needs of the client. Currently, the architect has a wide variety of tools to inform the client. Sketches, floorplans, sections, renderings, and documents all convey the design to the client and work well. However, they are still abstractions and they do not give control to the client nor do they accurately place them in the space. This project presents an alternative approach to the design process. It gives the client the ability to experience a proposed architecture first-hand in a virtual space, as well as execute control over certain elements in the architecture. Essentially, this project combines the elements of video game design and gameplay with architectural visualization. The combination of these fields could re-envision the way the client-architect relationship is currently handled.

Rationale

2.1 The Current Architectural Design Process

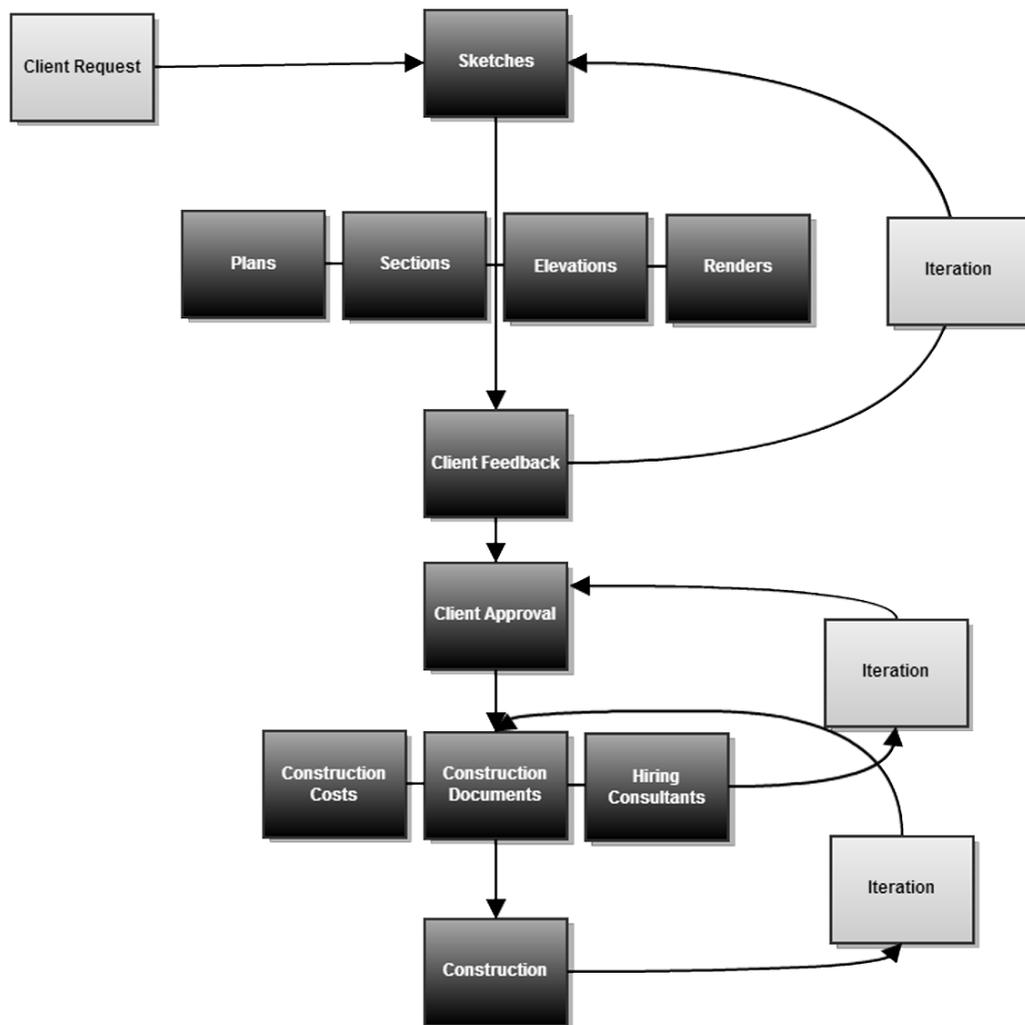


Figure 1 Architecture Process

The architect presents design ideas to the client for discussion and review. The architect analyzes the site, history, societal and environmental impacts, clients' requests, and a number of other factors. The architect develops sketches of possible solutions to show the client in the early design phase to help the client visualize what the designer has in mind. The architect then

creates several adaptations of a refined proposal, which would typically be floorplans, sections, elevations, renders, flythroughs and diagrams of the intended functionality. Bids and construction costs would be discussed at each stage along with the hiring of additional consultants and contractors. Based on client feedback, the architect would then re-design or adjust the proposed architecture. The architect would present a proposal again, and the cycle would continue until both parties achieve agreement. During construction, the architect would make site visits to monitor progress and provide additional drawings or specifications as requested by the contractor (Makstutis, 2010, p. 88-89).

2.2 Spatial Presence in a 3D Environment

In the architectural design process, it is difficult for a client to fully understand certain architectural aspects of the design that the architect is presenting to them, particularly the concept of space. Space is a nebulous principle defined in the vocabulary of modern architecture as “an apparatus we use to understand the world” (Amorim & Loureiro, 2007). The project addresses the issue of clients' understanding of space via immersion in a digital environment. Renders, floorplans, sections, and other architectural artifacts can only convey the aesthetic idea and the form to the client. While the client is able to see the building and the vision of what will occupy the site through these products, the presence of space remains elusive until a tangible construction is built. The concept is clear to the architect, but the client's perception is limited to the visuals they are given. Since space is the cornerstone of the modern architect's vocabulary, this project intends to help bridge the gap between the client's request for a particular spatial organization and the architect's design.

The rationale behind this project is to address the issue of spatial communication between architect and client during the concept presentation and feedback stages of the architectural design process. Given the power of modern technology and the recent developments in 3D visualization and architecture design, the project aims to address the issue described above using a 3D environment in two ways: first, by immersing the client in the virtual space, and second, by giving the client a way to customize and interact with the virtual environment. Immersion in a virtual space gives the user “Spatial Presence” (Wirth et al., 2007). Allowing the client to interact within the environment encourages the client to make alterations and decisions that align with their imagined view of the proposed architecture. These two things are imperative to address because it gives the client a deeper understanding of the architecture being designed and thus how the real architecture will feel. It also gives the architect insight into how the client is making decisions and therefore provides a clearer direction in future iterations of designing the proposed architecture.

By utilizing more than the current means of representation (e.g. sketches, drawings, and renderings), the project presents a method to reform the architectural process by helping the client understand and customize proposed designs. The project draws upon the execution of perspective in first-person video games and the success of “Spatial Presence” (Wirth et al., 2007) those games exhibit. The project is also putting design in the hands of the client by giving the client an interface to manipulate the architecture with. Albeit, the client has control to the extent that the architect deems necessary. The client does not supersede the architect in design control and instead is given a choice in handling the formation of the architecture as dictated by the architect. Integration of the two aforementioned mechanics will narrow the

gap between the expectations the client has and what the architect produces, by creating an interactive digital visualization system for clients.

The system the project demonstrates does not focus on the aesthetics, or polish, but rather addresses the relationships of spaces, explores the forms of participation at a deeper level between architect and client, and clarifies how the inhabitation of virtual spaces feel. Therefore, the project accomplishes “Spatial Presence” (Wirth et al., 2007) utilizing first person exploration as means of immersion to an understanding of spatial design between client and architect as well as opening a broader dialogue using the alternative model of participation (Wulz, 1986, p. 158) between the vision of the architect and the vision of the client.

2.3 Precedents

As stated, the project focuses on immersing the client in a virtual space and providing them with the opportunity for more interaction with design choices. To do so, precedents set in other fields of design must be examined. Currently computer-aided design methods in the architectural field focus on the production of static and non-manipulable three-dimensional representations. Examples of the typical tools an architect uses include drafting software (like AutoCad), 3D design software (such as 3Ds Max, Rhinoceros, Google Sketchup, Revit, and others), and post-processing tools (like Mental Ray, V-ray, and Photoshop). None of the aforementioned softwares use real-time elements. Instead, they are used to create renders. A render is a portrayal of an image or fabricated object in a virtual environment, and in the architectural process, it is an image that depicts a 3D scene—typically as real as possible. (Franklin, 2000) The programs are only used in-so-far as to create pictures that visualize the proposed architecture in a real environment. Unfortunately, in the industry this is currently as

far as it goes. Some architects may push the limit by adding 3D rendered movies, but these are all immutable forms of representation. Just like a book or cinema, the viewer is subject to the story and cannot interact. Here is the beauty of game design and why it is necessary to take a look into the field and how it can relate to architecture.

Firstly, video games that simulate 3D environments use game engines. Game engines are different from the software mentioned earlier for a variety of reasons. The most notable reasons are the use of real-time elements, the ability to program interactions to be played by the user, and the ability to package the software for distribution to the client. In regards to packaging the software, a person can play a developer's game as long as the player has the minimum requirements specified to play the game (such as a computer with a powerful enough graphics card or a console like the Xbox 360). The player does not need the software that created the game to play it, nor do they need the cumbersome amount of technical skills to operate the program just to play the game. Using a game engine software kit to develop a system to be used in the architecture design process adds a great amount of potential in the accessibility to demonstrate architect's designs. The 3D architecture of a game engine can be packaged for universal distribution (any computer meeting basic requirements) and a client can download the package to "play" the simulation.

The Unreal Development Kit (UDK) was chosen for this project because it is a powerful and robust tool. UDK is a software kit that combines a multitude of assets from other digital programs to create a game or system using the Unreal Engine. In the game industry, many of the same tools currently used by architects, are used to create assets (3D models, animations, sounds, effects, etc.) to be used in a game engine. Organizing the assets, adding game-play

elements, and programming interactions are all done through the game software kit. The culmination of these assets creates an experience for a person to play.

One of the most important features in UDK is the real-time rendering environment. Real-time rendering “is the most highly interactive area of computer graphics” which, “happens at a rapid enough rate that the viewer does not see individual images, but rather becomes immersed in a dynamic process” (Akenine-Möller, Haines, and Hoffman, 2008). This is a notable aspect of video game technologies and highlights the reasons why UDK was selected to demonstrate this project. The level of interaction one can achieve using a game engine’s ability to process real-time rendering is powerful and immersive. Current architecture practices that use 3D design do not create interaction. As stated earlier, most manners of representation of a design are through the mediums of print, video, renders, drawings, or physical models. The quality of immersion and interaction is non-existent in comparison to the level of video games. Even though many of the tools the architect uses in digital design are used in the game industry, the architectural field lacks the equivalent of the game engine; an engine that can package and share a simulation, immerse a client in an interactive environment, and give the client a real-time sense of space.

Method

3.1 Overview

To accomplish creating a system that architects can use to present their design, the project uses a variety of digital tools. Many of these design tools are familiar to artists and designers in the 3D field. An enormous amount of potential can be created through a wide variety of 3D software and development kits. The decision to use certain programs, described in detail later, was based on personal familiarity as well as what the software is capable of or designed for. Much like how a craftsman uses tools to construct a house, a digital designer must pick the appropriate software to accomplish certain tasks. Ability to do so comes from the proper use of the software. Quality and success comes from the experience of the designer and his or her comfortability with their designated tools.

Of course, tools do not entirely make the project. Well thought-out decisions inform the direction of each step and ensure a path to completion. The workflow for this project is as follows:

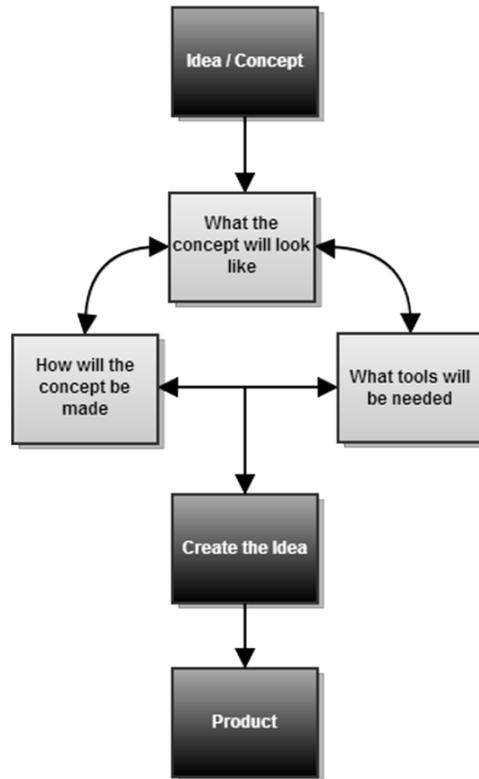


Figure II Design Process

An idea was formed about what type of system is being presented to address an issue. Questions such as, what does the idea accomplish, how does the idea work, and what is needed to make it work are asked. The concept was tested through sketches and writings, to develop a clearer goal. Software tools and resources for accomplishing the task were gathered and extensive learning on many of the functionalities in the programs were conducted (tutorials on modeling meshes, collision boxes, uv-wrapping, ui creation, scene creation, etc.). Work on developing the system then begins through asset creation. Assets, are fabricated 3D models and components that were used to set up an environment and interact with. From asset creation comes the intergration of those assets into one playable environment. Finally, after everything is set, an executable build is packaged to distribute to the client.

To understand the process outlined above in an easier fashion, imagine one is given the task to create a dining set. The type of dining set—how it will look and serve people dining—is the concept. Doodles and mock measurements explaining how big the dining set will look and the imagined style of it are the sketches and writing. The fashioning of chairs, table, and utensils are the creation of assets. Arranging them all to look coherent in a room is the integration of assets to form a cohesive, interactive environment. Publishing the dining set in a magazine for consumers to purchase or replicate is the executable package.

3.2 Construction

The project uses a multitude of design programs. 3DS Max 2013, a piece of professional design software, was used to create 3D models. It was primarily used for creation of 3D assets such as context pieces (the buildings on site) and the main programmatic pieces (the intervening architecture). Adobe Photoshop CS5.1 is a 2D editing program and was used to create textures for the meshes (3D objects) as well as any image adjustments needed (like presentations etc). Adobe Flash, an animation and user interface creation software, was used for making selectable onscreen buttons through Scaleform environment. It was used to create a user interface, in tandem with the Actionscript 3.0 scripting language, that the user could interact with in the simulation. The Unreal Development Kit (version 2012, November Beta) (UDK), a game engine software kit as was described earlier, was used to bring multiple assets together and be placed in a scene. The largest portion of the project work was done in the UDK environment. In UDK, Kismet, a visual-scripting editor, was used to set up a system of mechanics (rules for interaction). Kismet is much like a programming tool and, in the project, it provides the framework for how the user interacts with the environment. UDK was used in the

project to combine all of the assets into one cohesive simulation that allows the user to experience a virtual environment. Finally, to make the project accessible so that a user does not need the entire development kit to run the simulation, a build was made that allows one to execute and play the simulation on any PC. The combination of assets, scene organization, scripted interface, and design methodology created a simulation that supports the project objective.

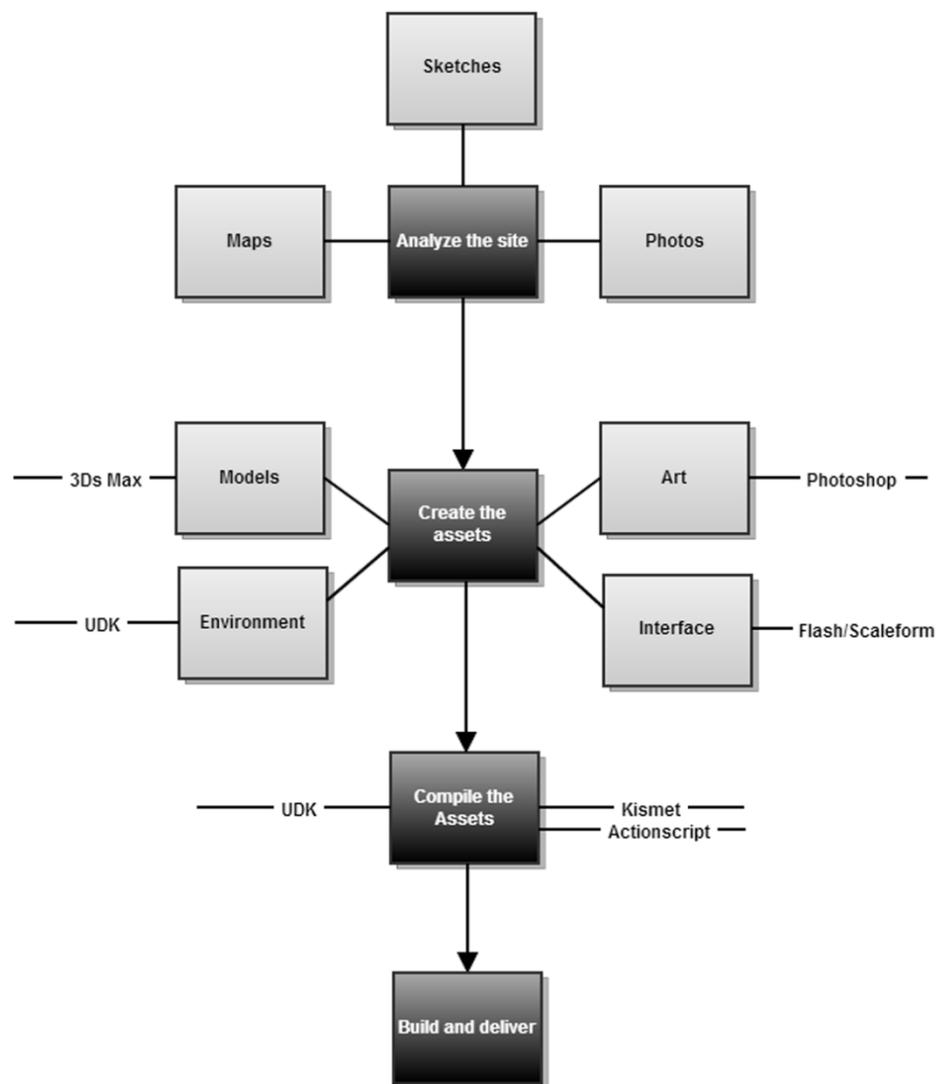


Figure III Project Work-flow

3.3 Design Process

The first stage of the project development was to decide on where the virtual environment will be and how it will look. To do so, ideas were explored such as building a residential house or even creating a completely fabricated environment with no foundation in the real world. Both of these choices were dismissed in place of the final option which was to re-create parts of downtown Gainesville, FL. A real and local site was decided on for several reasons which is discussed in further depth in the Design Narrative section.

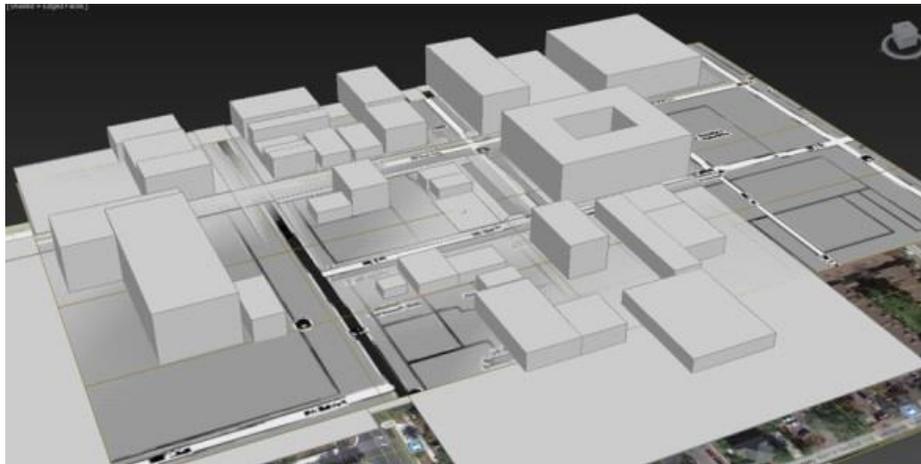


Figure IV: Blocking out the Context and Site

After deciding on the location, preliminary sketches and diagrams were created. Diagrams are useful tools for the architect that help analyze a site and inform the architect's decision on what the architecture should start to resemble. Creating diagrams in this project helped realize the forms that would relate to the context.

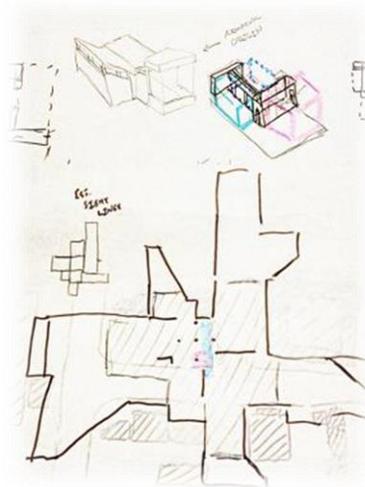


Figure V: Diagram Sketch of the Site

A context is the immediate information around the intervening architecture which includes the landscape and surrounding buildings. The diagrams informed the decision to make an armature piece with three adjacent interchangeable zones. Typically, an “armature” is a design term to describe a core part of a structure that holds, grasps, or informs spatial qualities and physical pieces that relate to it. At it's simplest definition, an armature could be considered a grid. In the project, the armature is a C-shaped structure that serves as the entrance to the construct as well as the primary means of egress through the structure. Rooms in the armature are placed to fulfill the most basic needs of any typical public building (such as bathrooms and a mechanical room). In this project, the armature is a permanent piece in the architecture and informs how the other spaces communicate.

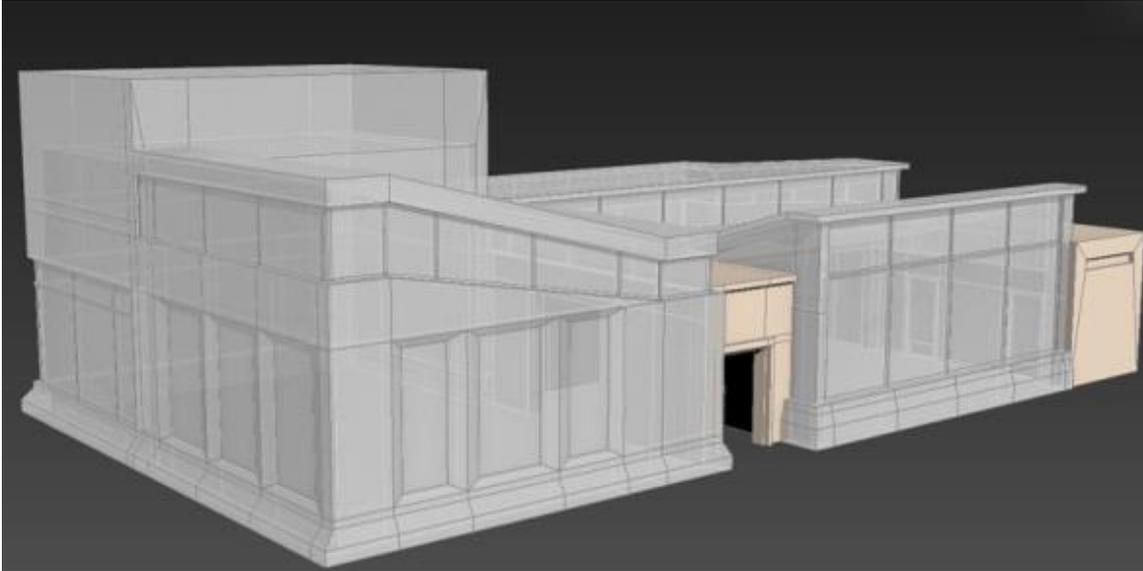


Figure VI: Armature (tan) Holding 3 Zones

Three zones were created to serve as dynamic, interchangeable spaces where the client could customize the design. Unlike a typical architecture design, these zones were created purposefully to show the client and highlight multiple instances of rooms or spaces that could be placed in each respective zone. Each zone has three variations of room and each room has two variations of type. Creating three zones illustrates variety in selection. When starting the simulation (“playing” the environment for the first time), there is nothing in these zones until the user chooses to place something. This is atypical for the current architectural design process because in the typical process, one version is presented to the client before iterating on it at a later time. The client does not get the option to change things instantly. However, in this project, the option of selecting from and interchanging multiple instances is presented.

Assets were created to populate the virtual environment. A rough layout of the site using volumetric shapes were built using a scale of 1:1 in 3Ds Max (1 unit in 3Ds Max is equal to 1 foot in real life). By using Google Maps and measuring out parts of the downtown Gainesville

site, an estimate of the size and scale of the site was created. From there, context pieces (the surrounding buildings and streets) were constructed. The surrounding buildings, such as the Hippodrome, Courthouse, and law offices were detailed to represent the physical buildings that exist. Construction of the variable architecture pieces were created as well. Once the assets were completed, they were imported into the UDK software.



Figure VII: Model of the Gainesville Courthouse

In UDK, the context assets were placed in the 3D environment to emulate the real site. The armature was positioned on the intervening location in the site as well. The architecture assets (variable pieces) were placed into zones around the armature. Once all of the modeled assets were placed, the scene was divided up into “levels.” A level is a virtual area where built environments are played (UDN, 2012). In the project, each interactive piece of architecture is made in a sub-level. A sub-level is loaded or unloaded into the permanent level (the scene with the context and armature). To clarify, the armature piece has three zones that make up the architecture. In each zone, there are three large scale variations (rooms). In each room, there

To allow a user to interact with the architecture, an interface was created using Flash and Actionscript 3.0 via Scaleform. Flash is an animation program and provides the visual of an interface. Actionscript is a scripting language that communicates inputs (such as mouse or keyboard) to the visuals in conjunction with Scaleform. An interface is a visual system, typically overlaid on the screen (like a video game) that reacts to user inputs. In the project, a user interface allows the client to easily load or unload architecture pieces as well as change cameras and close out or open the user interface display. The interface was created to easily allow the client to interact. This is a key part in the project for it gives partial control to the client over the architecture through simple actions. By creating an understandable way to interact and manipulate the architecture, the user interface allows clients to feel more immersed and knowledgeable about design decisions. Clients can customize the architecture in real-time, giving them immediate feedback.

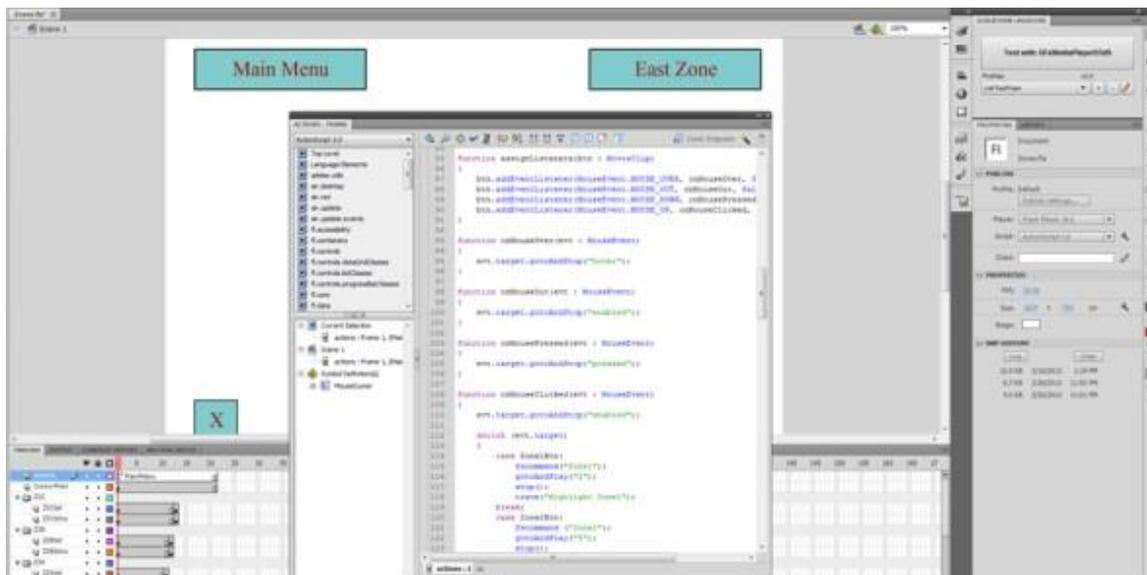


Figure IX : Actionscript Code in Flash Environment

Finally, a build is made that compiles all of the scripting, levels, assets, and content that went into the UDK environment. The build is an executable (runable) file that allows most anyone with a Windows computer to test and play the simulation. This gives clients an easy way to see their architecture design and even the chance to play with it without the architect being physically present.

3.3.1 Scenario-based User Testing Methods

A narrative was developed for participants testing the project. As nice as it would be, the project cannot use real clients due to several limiting factors like building permits, personal freedom to iterate components in the development of the thesis, and level of work necessary to succeed as a professional tool. Because this project cannot be realized in its fullest form currently, a narrative is needed in place of real world application. In architectural design projects, a narrative is more commonly applied to “intervention“ projects. An intervention is a proposed architectural construct that is developed on a site based on certain specifications (instructor set, site-determined, solve a problem, etc.). The intervention can be theoretical or actual (it will or will not be built) and in the case of this project, it was constructed on a parking lot in downtown Gainesville, Fl, on the corner of SE 1st Street and SE 2nd Ave. The program for the project is identified as a gallery and the narrative for the user testing the project is as an art curator. The role of art curator and intervening art gallery was picked for several reasons.

Downtown Gainesville has yearly civic events called “art-walks“ that are very integrated within the community and city. The creation of an architecture that could show off the art of vendors, or serve otherwise during the art-walks, would be endearing to locals testing this project. As a person involved in the community, it would matter what decisions are made when

interacting with the project. For other users who are not involved with the community, the opinion of aesthetics becomes a driving motivation in the interactions in the project. Deciding on what piece goes where is important to the user because it gives them an abstract choice. Interaction between the user and the project becomes meaningful and immersive and can provide valuable results.

A pool of six users was selected to include people familiar with the location as well as those who are and are not educated in the architecture pedagogy. Having a diverse selection of users—participants novice in architecture practices as well as those familiar with architecture—provide insight into both perspectives of thought—the client (unfamiliar user) and the architect (professional developer). Users were also asked of their familiarity with 3D navigation in games to assess how comfortable they were (in regards to control and movement in a virtual space) when participating in the project.

Users	Familiar with 3D navigation	Unfamiliar with 3D navigation	
Familiar with architecture principles	1	1	
Unfamiliar with architecture principles	2	2	

Results

4.1 Project Screen Captures



Figure X: Welcome Screen



Figure XI: View from User Start Position

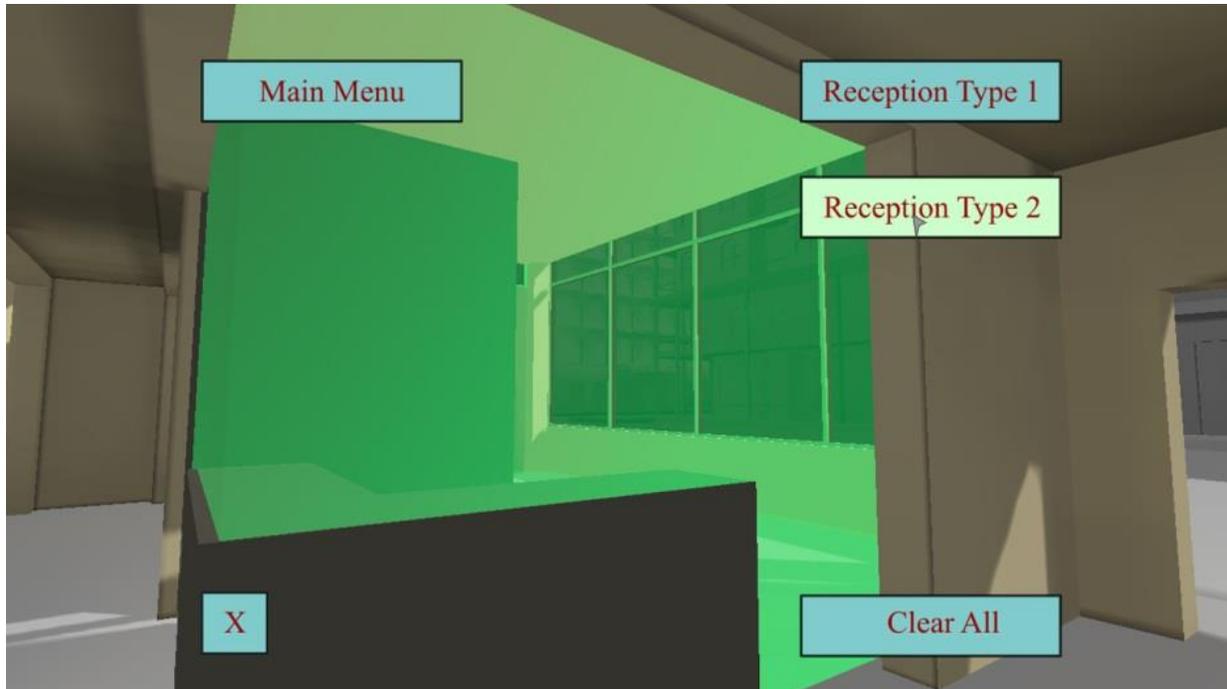


Figure XII: Interior Perspective With Active User Interface



Figure XIII: Overview from Arial Camera of One Selected Version

4.2 User Reflections

After demonstrating the project, in several user studies, I was left with some informative results—particularly about how users interacted with the demonstration and suggestions on future iterations.

A common issue was navigation in the virtual environment. Currently, the project controls are set up like a First-Person-Shooter (FPS) in computer games. W-A-S-D keys control the user's movement in the virtual environment forwards, left, back, and right respectively. The mouse controls the camera movement or view of the user. To someone who is familiar with first person video games, moving around on screen in a 3D space is second nature. Those who were not familiar with 3D navigation (especially in games), had difficulties maneuvering themselves through the demonstration. The mechanic taken from many first-person shooter games did not translate as well as hoped into the experience of the virtual environment for some users. Using the WASD keys and mouse did not provide an intuitive navigation system.

The interaction design of the menu system was another issue that arose during the user testing. Across users, there were varying opinions on the amount of control given to the client. After initial studies, the issue seemed to stem from the user interface that was constructed. Users had problems with understanding what actions affected what space—what buttons when clicked would create a certain room—and going forwards or backwards in the interface menu. To solve this issue, button titles were changed to represent the spaces more accurately. For example, "Zone 1A" was changed to "Large Windows" to better, and more simply, represent what the user was expecting (instead of ordering by hierarchy, devoid of representation). Also, menu navigation was made easier by having a way to go back to the main menu at any time.

However, during the second round of testing, it became apparent the issue lay a little deeper. Users still had a hard time changing from camera to camera and opening the menu to select their options in a fluid manner. The user interface within the program wasn't main the issue as much as there was too much control over the system. The objective of the project to make a complex system that a professional uses—simple enough for typical clients to use—was still too complex. The project was repeating the problem it was trying to solve.

Potential resolutions to these issues of spatial navigation and interaction design grew out of conversations with users. Several users in separate studies had insightful input on resolving navigation and interaction problems. To resolve the navigation issue, it was suggested to look at intuitive designs that exist outside of the game design discipline like the Google Art Project and Street View in Google Maps. Those precedents utilize a very simple and intuitive interface that many users unfamiliar with navigation in 3D space have a way of quickly understanding. Instead of using a keyboard or controller, selecting and dragging (panning or rotating using the mouse and mouse buttons) within the environment is the main way of navigating. In regards to issues of interaction design, one suggested that the software should be revised to “make it idiot-proof.” Instead of allowing users full control over changes, he suggested to give them the ability to change aspects they wish in real time—as the project is currently presented—but without the interface. Other suggestions consisted of letting the user make the changes but at the architect's discretion, by giving the client the same navigational abilities but moving the interface that exists in the project into a separate system that the architect controls. Whenever the client wants to make a change, the architect has the tools disposable to make that change, without confusing the client or needing to guide the client

through the process.

Overall, user feedback on the concept of the process was quite positive. Participants felt they had a much greater involvement when using the project in comparison to typical representations. In users who have knowledge of downtown Gainesville, there was an immediate recognition of the site. Those who knew the location felt that the project had a good representation of scale and enjoyed the experience of vicariously being there. As for problems mentioned earlier, many of them arose from lack of technical means to create an intuitive full-fledged system. These issues are redeemable in future iterations with the right software and skill.

Conclusion

5.1 Discussion and Future Direction

There are a number of directions the project can take from here to improve both the immersive quality and control. Examination of results for future implementation is elaborated on as follows:

1) Increasing the accuracy of materials and lighting representation:

A large reason the UDK software was picked is because of its real-time rendering ability and realistic environments. In the project there are currently a few ways that the software places the user in a verisimilitude representation of the real site. There is much more potential with a real time software, that in the future, could generate a more immersive experience.

In the Unreal Engine, dynamic lighting can be used to light a scene. Dynamic lighting is different from static lighting in that the program constantly updates the information about how light is cast on objects. Use of this lighting, for example, could be a light that is placed in orbit around the scene much like the sun. In the project, this could be highly beneficial to understanding certain spaces and how they look at different times of the day. If there is a light (the sun) that can be changed at any time to replicate real world conditions, it gives the client a better understanding of how a lit space will look and feel in different lighting moods. With this addition, clients can make more informed decisions based on the lighting of the architecture.

In the project, materials are simple solids. With future work, the materials on object can be changed to more accurately represent materials in structures like concrete, brick, tile, stone, plaster, etc. Adding materials to 3D objects is nothing new, but changing them immediately for the client, could be a powerful addition. UDK utilizes materials that can be triggered

(referenced or activated) through code and Kismet. If the client wishes to see a different type of material on a certain wall, they would be able to do that in real-time. All that would be necessary is an interface to execute the desired action—which is something the project proves is capable. A further addition to lighting and the look of the environment could be added in the post-processing stage. Things like depth-of-field, motion blurs, etc... could be determined by the same interface and real-time commands.

2) Improving client interaction through collaborative, real-time design:

Instead of giving the client full control, give them options and remove the interface. The architect could be nearby, updating the virtual environment as the client experiences it, without ever being interrupted with a user interface screen. The architect would be networked to the client machine so that all changes would occur immediately upon request from the client. Options the architect could choose from could be then be much more robust.

The incorporation of generative modeling could become an enormous help to creating a sequence of variables, almost endless, that the client can choose from. An example of generative modeling in 3D is the plug-in, Grasshopper. It is a design software that plugs-into the 3D modeling program, Rhinoceros and is different from typical modeling software because it bases construction on a set of constraints. Each constraint in the hierarchy of an object being made is stored so that at any point in the construction, the modeler can adjust a parameter to update the final product. This method of construction is similar to Maya's history tool and 3Ds Max's modifier toolbar, but it is much more robust. Grasshopper's way of modeling gives immediate feedback. However, it cannot be used in real-time rendering because of the process-intensive nature of storing each constraint. Instead, when the modeler is happy with a version

of the model, it is “baked” (made permanent so that the past information is erased and therefore frees up memory). Baked models are then able to be exported and used in other programs, but do not retain the information that was used to construct it. If this process could be utilized in a game engine and with easy ways to interact, the potential for an enormous amount of customization is opened up in the client and architect design process.

5.2 Summary

In conclusion, the project articulates an alternative approach to the current architectural design process. It presents a method for architectural design that allows the user to navigate and customize architectural designs in a 3D space, which helps clients develop solid understandings of space in proposed constructions. The depth of experience goes further than current forms of representation by allowing a client navigation in a 3D space as well as real-time, immediate iteration on parts of the architecture.

This project envisions a future in which architectural designs find form in dynamic, navigable and customizable three-dimensional environments, that clearly communicate the structure of the envisioned space to clients. At this point, the project is just the first step in a much more powerful system. There is a great amount of potential, as well as work, in future iterations. The biggest step is creating a system that architects can use easily and efficiently before presenting their ideas to the client in a professional setting. Currently, it is done using a multitude of resources that are all used for other purposes. A new work-flow would need to be developed to make it much less time consuming for the designer. The software would also have to be capable of real-time rendering, like UDK. Before the architect can allow the client control, they need to first have mastery of the tools themselves.

References

- Amorim L., Loureiro C. (2007), The space of architecture and a new conservation agenda. *City & Time* 2 (3): 1. [online] URL: <http://www.ct.ceci-br.org>
- Akenine-Möller, T., Haines, E., & Hoffman, N. (2008). *Real-Time Rendering, Third Edition*. Natick, MA: A K Peters, Ltd.
- Franklin, C. (2000, July 15) "How 3-D Graphics Work" [HowStuffWorks.com](http://computer.howstuffworks.com/3dgraphics.htm).
<http://computer.howstuffworks.com/3dgraphics.htm>
- Unreal Developer Network (UDN)(2012) Getting Started: Level Editing. Retrieved from:
<http://udn.epicgames.com/Three/GettingStartedLevels.html>
- Makstutis, G. (2010). *Architecture*. London: Laurence King Publishing Ltd.
- Wirth, W. et al. (2007). A process model of the formation of spatial presence experiences. *Media Psychology*, 9, (493-525)
- Wulz, F. (1986). The concept of participation. *Design Studies*, 7(3), 153-162.

Biographical Sketch

Duncan William Martel was born on October 3, 1989 in Bangor, Maine. An only child, he grew up in Hudson, Maine, and moved to Ocala, Florida, in 1997. Graduating from Forest High School in 2007, he went on to earn his Bachelor of Design with a degree in Architecture from the University of Florida in 2011. Currently, he is pursuing a Masters of Arts degree in the Digital Worlds Institute from the University of Florida. He plans to graduate in May, 2013.

Upon graduating, he plans to seek employment in the field of game design, as either a level designer or 3D modeler. His hope is to bring his experience and knowledge from architecture into the profession of game design.