

Realistic 3D Modeling of Digital Networks and the Internet

Project Portfolio

NetworkCraft

Sean Anderson and Benjamin Hallman

Document History

Rev Number	Date	Modified By	Reason
1	10 Oct 2019	S. Anderson, B. Hallman	Initial Release
2	8 Dec 2019	B. Hallman	Minutes Update
3	13 Dec 2019	S. Anderson	Updated Pictures
4	10 Mar 2020	S. Anderson	Multiple Section Changes, Team Name Change, Title Change
5	5 May 2020	S. Anderson, B. Hallman	Minutes updates, added conclusion details
6	7 May 2020	S. Anderson	Added future work

Table of Contents

1 MEETING MINUTES	4
1.1 WEEKLY ACTION ITEMS.....	4
1.1.1 <i>Team Meeting #1 – Wed. September 11, 2019</i>	4
1.1.2 <i>Client Meeting #1 – Thur. October 3, 2019</i>	4
1.1.3 <i>Client Meeting #2 – Thur. November 19, 2019</i>	4
1.1.4 <i>Client Meeting #3 – Fri. February 21, 2020</i>	5
1.1.5 <i>Client Meeting #3 – Fri. April 10, 2020</i>	5
1.1.6 <i>Client Meeting #3 – Fri. February 21, 2020</i>	5
2 PROJECT LEARNING	6
2.1 WHAT OFF-THE-SHELF TECHNOLOGIES YOU CAN USE TO SOLVE THE PROBLEM?	6
2.2 WHY DID YOU THINK THE DESIGN WOULD WORK?.....	6
2.3 WHY THE DESIGN DID NOT WORK.	6
2.4 WHAT NEEDED TO BE CHANGED?	6
3 DESIGN GOALS	6
3.1 CLIENT NEED AND PROJECT GOAL (WITH TIMELINE).....	6
3.2 REVISION OF GOAL. WHY THE REVISION WAS NECESSARY?	7
3.3 NEW GOAL	7
4 SPECIFICATIONS AND CONSTRAINTS	8
4.1 PICTURE SPECIFICATION FROM CLIENT MEETING #1	8
4.2 DESIGN SPECIFICATIONS AND CONSTRAINTS.	12
5 ANALYSIS OF ALTERNATIVES	13
5.1 POSSIBLE ALTERNATIVES.....	13
6 IMPLEMENTATION PLAN	13
6.1 REPOSITORY.....	13
6.2 ILLUSTRATION OF NETWORK CREATION	13
6.3 IMPLEMENTATION	13
6.4 TESTING/EXPERIMENTAL DESIGN	13
7 ANALYSIS	13
7.1 WRITE ABOUT THE CONFIDENCE IN YOUR RESULTS.	13
7.2 LEARNING OUTCOMES	14
8 BUDGET	14
8.1 INITIAL BUDGET.....	14
9 FUTURE WORK	14
9.1 VR IMPLEMENTATION.....	14
9.2 MULTIPLE PORTS/CONNECTIONS	14
9.3 GUI UPDATES	14
9.4 NETWORK LAYERS.....	15
9.5 MAC VALIDATION CHECKING	15
9.6 FURTHERING TUTORIAL CONTENT	15

1 Meeting Minutes

1.1 Weekly action items

- Summary of progress
- Meeting review: Helpful/Not
- Attendance and participation
- Team member contributions

1.1.1 Team Meeting #1 – Wed. September 11, 2019

Overview

- Team Name: VRCraft
- Team meetings will be held as needed. Both of our schedules align pretty well already.

To-do

- Finish contract - BH & SA
- Transfer logbook to OneNote and share it - BH & SA
- Make a minutes document - BH
- Make a repository on GitHub for future work - BH
- Email out agenda - SA

1.1.2 Client Meeting #1 – Thur. October 3, 2019

Goals

- Create computer network with switches
- Wire cables
- Computers talk to each other
- Build layers 1, 2, and 3
- Build a network virtually
- Show MAC address
- Implement in VR (Look into)

Moving Forward

- GitHub repo with Wiki
- Private for now. Possibly public later
- Schedule up to us
- Start with networks and move onto VR later
- Experiment with formspec for GUI

Create Objects

- “Computer” Device - Name and MAC address
- Wire - Receives on one end and sends on the other using ABM (4 method calls)
- Switch

1.1.3 Client Meeting #2 – Thur. November 19, 2019

Overview

- Meeting began with Sean and Ben discussing our first snapshot day with Dr. Conte de Leon and showed him what we accomplished.
- Client agreed that it would be best for us to continue building from the Computer Minetest mod that Sean had found online.
- Once we finished giving an overview of what we had completed thus far Dr. Conte de Leon began to explain to our team his expectations moving forward.

This expectations, goals, and possible ideas consisted of the following:

- Creating a wiretap for TCP dumps
- Creating a switch block with the following functionalities:
 - MAC address on computer and packet
 - Switch looks at MAC address and does not change it (Routers do)
 - If there is no MAC address send to everyone.

Our goals moving forward include the following:

- Investigate how to store data, hash table. Depending on the position
- Hooking up wires to a switch
- Send message in GUI using the following:
 - Global array for that mod then the index to the array is the position of the nodes. There can only be one node in XYZ. Use `getid()` which grabs the first node Z and multiplies X gives a unique number based on position and forms a "hash" table. Each computer can have its own address.

1.1.4 Client Meeting #3 – Fri. February 21, 2020

Overview

- Meeting began with Sean and Ben mostly discussing the issues we had encountered to date.
- Dr. Conte de Leon, was able to assist Sean through the issue of sending messages through the cables. The issue we encountered of messages traversing both ways was able to be solved with the assistance of our client within the two hours we met.
- Once we finished giving an overview of what we had completed thus far Dr. Conte de Leon began to explain to our team his expectations moving forward.
- Ben gave a detailed breakdown on his work regarding the network switch.

1.1.5 Client Meeting #3 – Fri. April 10, 2020

Overview

- Met over Zoom.
- Sean and Ben broke down everything being accomplished to date.
- Dr. Conte de Leon seemed very impressed and wanted us to focus more on the educational aspect of our modification.
- Sean and Ben decided that Sean would continue working on enhancing the code of our modification and that Ben would create a student help guide for the mod.

1.1.6 Client Meeting #3 – Fri. February 21, 2020

Overview

- Met over Zoom.
- Sean showed Dr. Conte de Leon a breakdown of the Expo poster which the client approved.
- Sean proceeded to breakdown what he thought would be the best presentation possible for our group to give at the Expo. Both Ben and the client added their thoughts to how to best proceed.
- Ben explained how the help guide worked and how the students would be able to view it in-game.

2 Project Learning

2.1 What off-the-shelf technologies you can use to solve the problem?

Pre-existing mods and software for Minecraft give us an idea platform to build from. There are also open-source mods publicly available for Minetest to help with implementation of computers and can be expanded upon.

2.2 Why did you think the design would work?

Minetest is a free and open source 3D voxel game that enables the creation of complex virtual worlds. Mods enable modifying Minetest. There are 100s of free of mods, i.e.: Animals, Armor, Mechanics. There are step-by-step open source tutorials on how to create mods. However, current mods do not model computer networks and their control in a way that is realistic enough to enable quality research and education in TCP/IP-based networks and cybersecurity using 3D virtual worlds. Because of the capability to create mods with the open-source platform, we believe that we will be able to create this new setup to integrate networks into the game.

2.3 Why the design did not work.

Originally, we believed that we would be able to implement Virtual Reality (VR) to coincide with our modification. Unfortunately, due to complexity and time constraints, we were unable to implement it. We thought there were other potentially open-source modifications to the game to help us implement the functionality, but the support for it seems to be extremely limited. As a result, we decided to not continue forward with its integration and chose to focus more on the basic networking setup.

2.4 What needed to be changed?

Remove the idea of VR implementation.

3 Design Goals

3.1 Client need and project goal (with timeline).

“Enable the creation and usage of realistic and accurate 3D and virtual reality models of modern computer networks for research and instruction.”

Task Description
Completed Team Contract - Sept. 12, 2019
Budget Estimates - Sept. 19, 2019
Approval of Requirements - Sept. 26, 2019
Initial Mod Creation - Oct. 3, 2019
Client Meeting #1 - Oct. 3, 2019
Logbook and Portfolio Initial - Oct. 10, 2019
Workflow Layout + Further Mod Development - Oct. 14, 2019
Snapshot Day #1 - Oct. 15, 2019
Design Validation Plan - Oct. 24, 2019
Project Value Proposition - Oct. 31, 2019
Logbook and Portfolio Update - Nov. 14, 2019
Concept Design Review - Nov. 22, 2019
Snapshot Day #2 - Dec. 6, 2019
Project Portfolio, Wikipage, Logbook First Semester Finalized - Dec. 13,
Project roughly 50% completed - Dec. 31, 2019
Design Expo Registration - Jan. 28, 2020
Engineering Release Review and Logbook draft - Feb. 21, 2020
Complete Prototype build - Apr. 5, 2020
Logbook - Apr. 17, 2020
UI Design EXPO - May 1, 2020
Final Report / Drawings - May 5, 2020

3.2 Revision of goal. Why the revision was necessary?

Our original goal didn't include a timeline on specifically VR implementation, and as such, does not need to be revised.

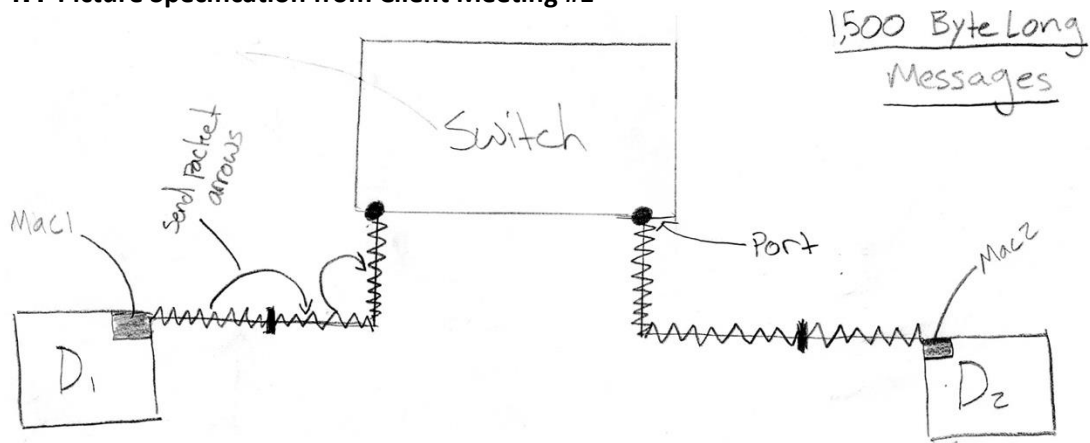
3.3 New goal

Our new goal and correlating objectives are as follows:

- Create the first prototype of the game-based environment using Minetest
- Allow computers to communicate with one another if configured properly
- Add corresponding instructional content to assist learner
- Nicely commented code, allow for future improvements

4 Specifications and Constraints

4.1 Picture Specification from Client Meeting #1



Custom Functions

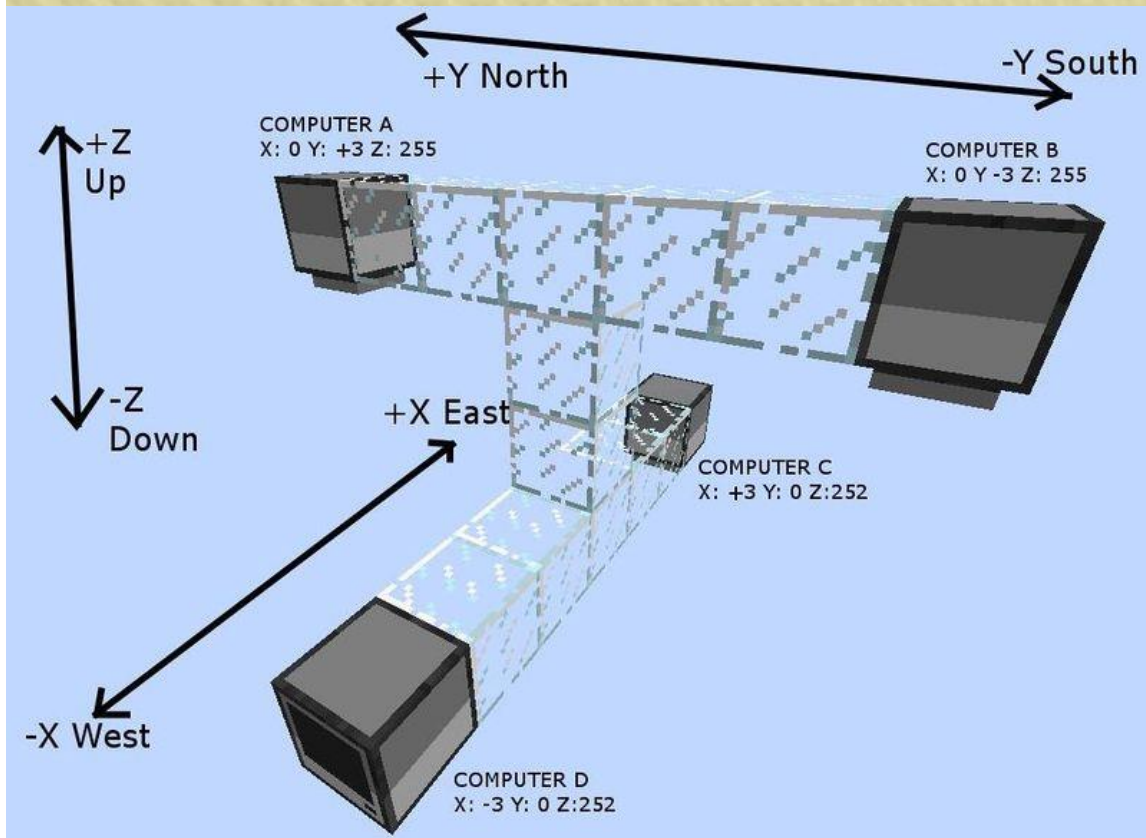
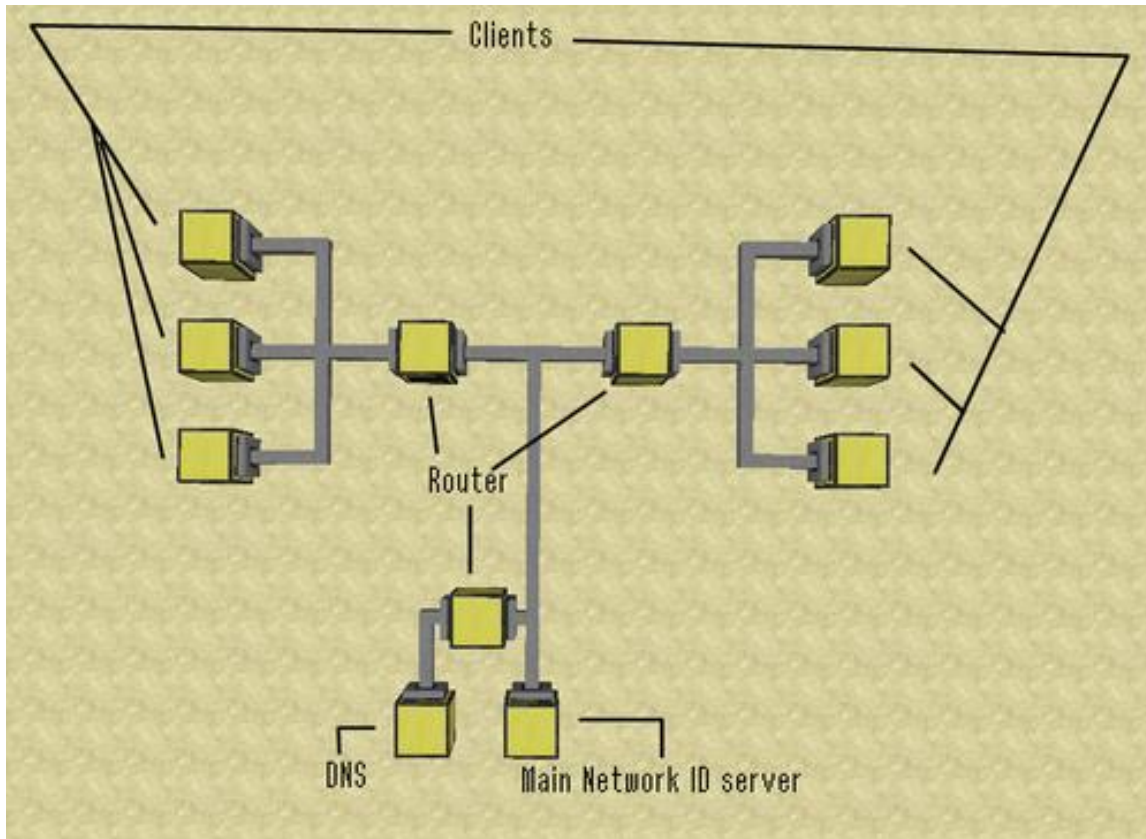
sendxdir()	recxdir()
sendydir()	recydir()
sendzdir()	reczdir()

← Sending message functions

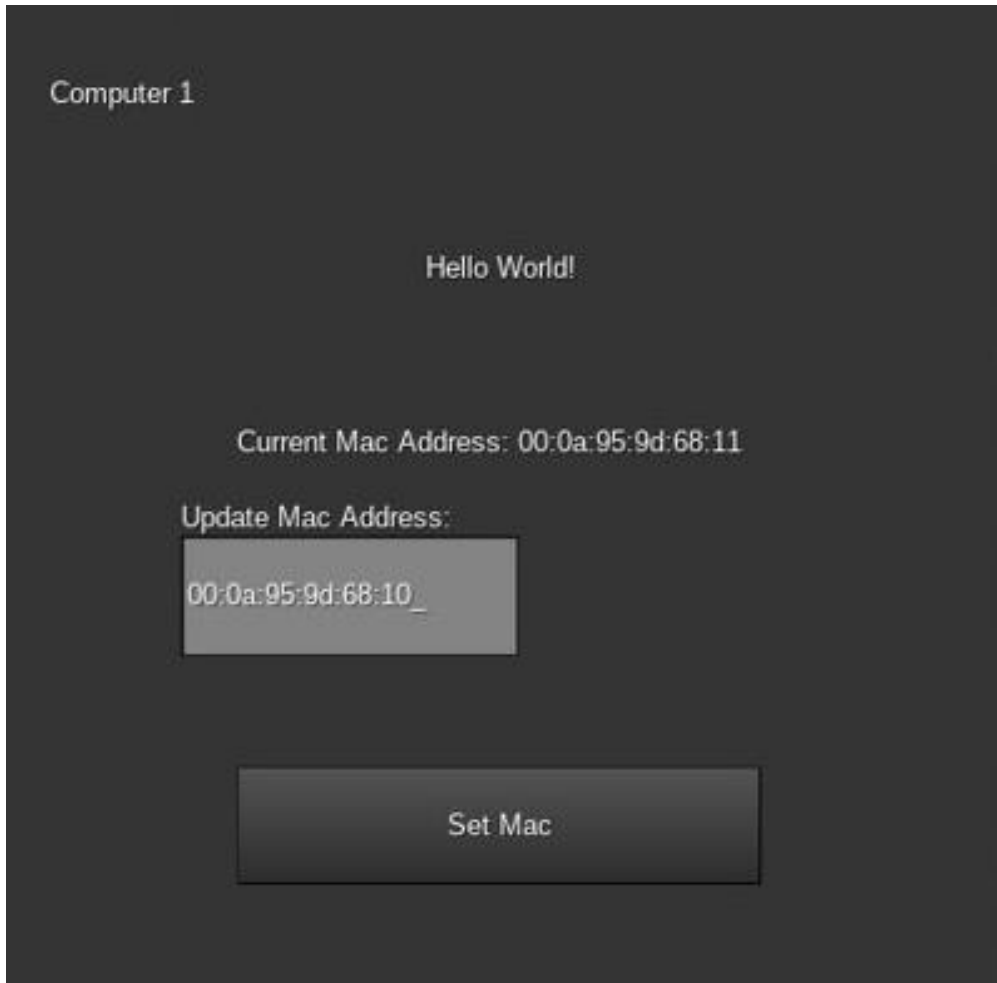
← Receiving message functions

Coordinate Key

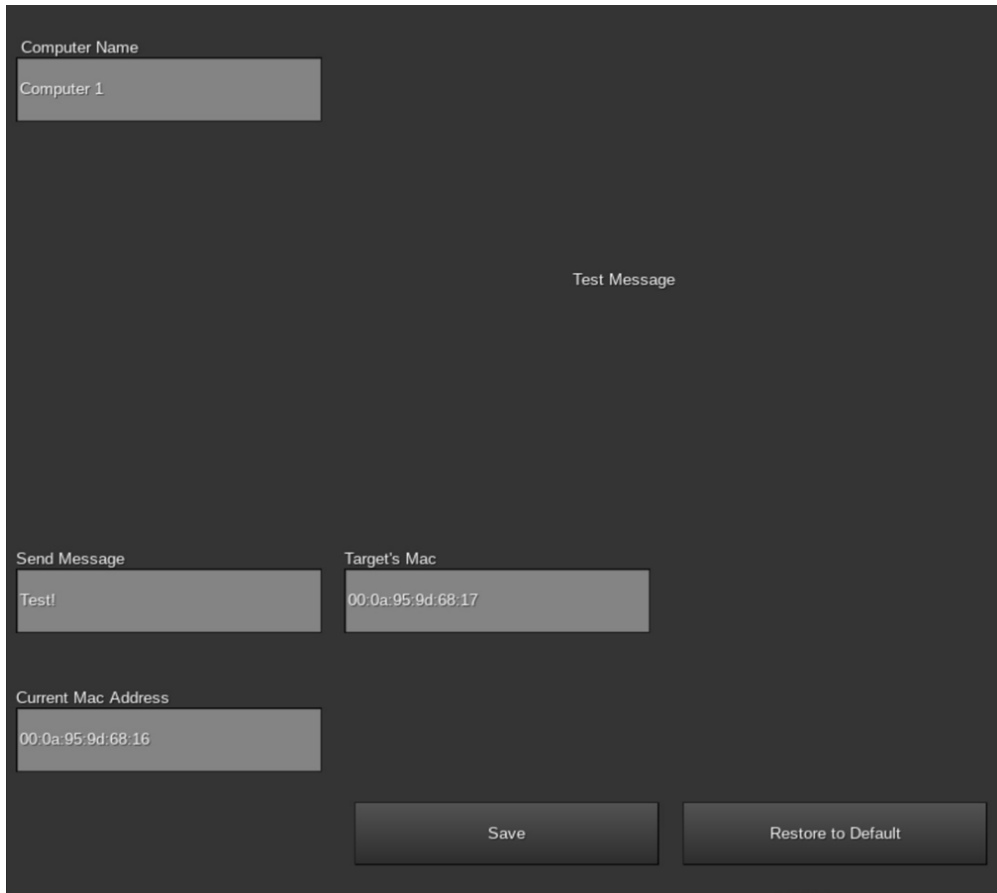
A 3D coordinate system with x, y, and z axes. The x-axis is horizontal, the y-axis is vertical, and the z-axis is diagonal.



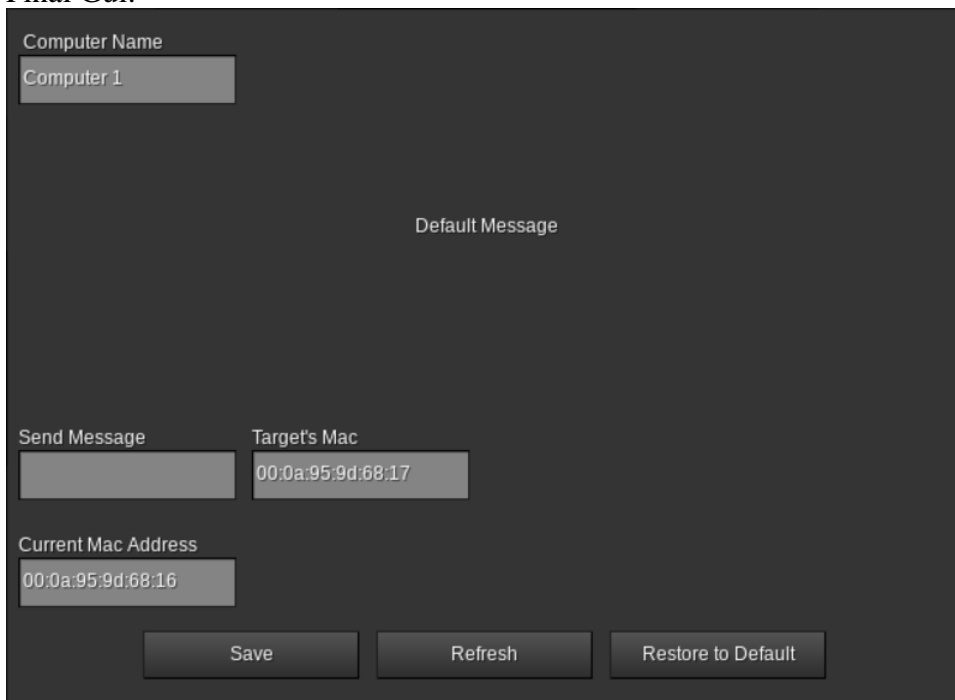
Initial GUI design:



Secondary GUI design:



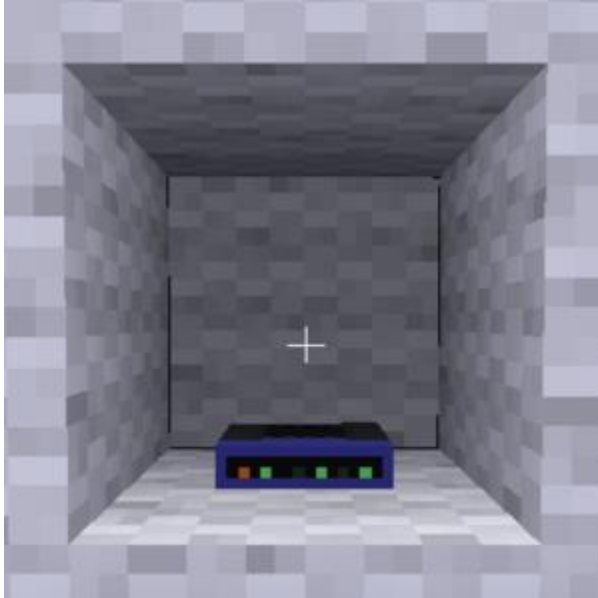
Final Gui:



Computer Design:



Switch Design:



Example setup:



Demonstration:

<http://images.shoutwiki.com/mindworks/4/49/Network.gif>

4.2 Design specifications and constraints.

Our specifications revolve around the basic implementation of a system of networks inside our game. We are designing and implementing computers, network switches, packets, MAC

addresses, cables, and similar concepts. We will also have a debugging mode to showcase what actions are occurring, if desired.

5 Analysis of Alternatives

5.1 Possible Alternatives

The game that Minetest is based upon, Minecraft, is a much further developed game. Networking structures can also be implemented in Minecraft, and Minecraft is already supported with VR. However, due to potential licensing constraints, Minecraft is not a current possibility for the University to work with.

6 Implementation Plan

6.1 Repository

A private repository has been created by our client on the popular code-sharing website GitHub. This repository will be kept private for now and access will be given on a case-by-case basis. Eventually, the code will likely be opened for public use and modification.

6.2 Illustration of network creation

Illustrations have been made at various points to help us visualize the creation of a network in the game and allow us to generate a concise order of steps to take in order to implement these networks. See above for the drawings that we had in early conceptual development stages.

6.3 Implementation

We began our initial implementation by meeting with our client and creating basic illustrations to create a starting point. From there, we began looking for open-source computers to work off that had been previously created within Minetest and began implementing our own unique GUI onto the imported computers. We continued meeting with our client to get updated illustrations and instructions and proceeded to program in wires/cables and our network switch. We learned the scripting language Lua to implement various functions to add functionality of the computers, wires, and switches. These functions allowed us to save/delete specific text within the fields of the GUIs, send text and corresponding via wires, have the switch detect which corresponding MAC is correct, and send information to another computer. We also send out debug messages to the user to showcase what is happening in real-time.

6.4 Testing/Experimental Design

With programming, bugs often require a lot of time, experimentation, and test cases to find, so adequate time will be given to find any that present themselves within our basic network implementation. We expect to be bug-free when expo/client-presentations come around.

7 Analysis

7.1 Write about the confidence in your results.

As with any early-stage software, our program presents various bugs. Specifically, sometimes sending a message from a specific direction can be a problem, as well as certain buttons, such

as the 'Restore to Default', don't always function properly. However, shown through various presentations, demos, and gifs, our project does work as intended with the core mechanics – sending a message from one computer to another via wires and a switch if the MAC addresses are correct.

7.2 Learning outcomes

From this project our team ended with multiple takeaways, some more positive than others. These learning outcomes included the importance of a strong foundation for large-scale technical projects and the need for well-articulated building blocks which can be used for future development by other groups. Specifically, a strong foundation for our project was needed as our team was unable to tackle the entire addition of implementing virtual reality. Therefore, in order for the next team to work on this project to have an easy transition a proper yet clear foundation was created. Secondly, much like the game itself, our project was created in a layered block form where each portion of the code is separated, well documented, and commented for easy use by future teams.

8 Budget

8.1 Initial Budget

Our initial budget is \$0.

Minetest is free, open-source software, and all the components needed for our mod's implementation will not cost anything.

9 Future Work

9.1 VR Implementation

As has been mentioned multiple times throughout this document, we were unable to implement virtual reality into our initial model. We were able to create a basis for the network, but we did not develop a mod to integrate VR into Minetest (which does not already exist at the time of writing). Creating this independent mod and the interlacing key mappings for the controllers/keyboard would be a huge step forwards in regards to the progression of the project.

9.2 Multiple Ports/Connections

We were unable to finish integrating multiple ports/connections into the switch to allow multiple computers to send messages simultaneously. As it stands, only one computer can send a message until the switch is broken and replaced. There is a section in our code specifically sending messages to only `Switch.Port1` and this could be expanded (as well as in other sections) to `Switch.Port2`, etc. and determine which port it should fill depending on the coordinates of the wire and switch. For example, if $(pos.x == node_pos.x + 1)$, then fill `Port1`. This could be repeated for each position (x, y, z) $(+1, -1)$ for six total ports, one correlating to each side of the switch.

9.3 GUI Updates

We would have liked to slightly improve both the computer and switch GUIs. For the computer GUI, we would like to add an actual functionality feature for the Computer Name field. As of right now, it only exists for students to mess with to make communication between students

easier. One potential functionality feature that could be added is displaying the corresponding computer name that either sent or received a message in the debug chat. The switch gui currently does not update properly and does not give any relevant information to the user. It could show which port is attempting to which Mac address.

9.4 Network Layers

Adding more layers of the OSI model would be an interesting concept and would further enhance the learning of students. For example, adding TCP and IP would likely be the next step to further improving our current model.

9.5 Mac Validation Checking

Currently, by design, we allow the user to match any mac address, whether it follows the correct format or not. For example, if the target mac of one computer is “MAC” and not “00:0a:95:9d:68:16”, as long as the mac of the other computer is also “MAC”, the message will be able to successfully send. This is to cater to a younger audience, however, a validation check could be implemented and potentially pushed to the Minetest chat if it doesn't pass it.

9.6 Furthering Tutorial Content

As any content is added, adding onto our tutorial book or GUIs will likely be extremely beneficial to the student.