Toward “food 4 u” 3D: 3D Princeton Exploration with Customizable Avatars

Aaron Buck, Ari Braun, & Benjamin Chan
Professor: Felix Heide

1. Abstract:

We have embarked on an experimental and immersive project which situates users in the environment of Princeton. Our implementation integrates interactive map navigation and waypoint finding game with the ability to select and modify customizable meshes in the immersive setting of a 3D map user interface. We accomplished the customizability by adopting the cognitively familiar user interface of a GUI for selecting options that allow the user to creatively design and determine animations for the way that they are represented in the map as a mesh-based avatar. This avatar creation is complemented by the ability to witness animations of the avatar as it interacts with the 3D map. The 3D map also contains familiar landmarks in the Princeton area and could be navigated to the ends of the Earth.

2. Introduction

- **2.2 Goal**

  2.2.1 What did we try to do?
  We tried to implement a mesh-based avatar selection suite built on top of a 3D map-based user interface with animation and navigation throughout a 3D map of Princeton. Our goal was to provide the option of customizing one’s virtual avatar to give the user several options and enable their creativity to improve their in-map experiences. We tried to create an element of gamification and touring through a thought-provoking 3D rendering of the map of Princeton and the surrounding geographies by adding markers ad preselected locations. The goal of this gamification is to inspire the user to consider the curated navigation that we will guide them through as they revisit familiar sites in a virtual world and provide a readily accessible alternative to waiting for a campus tour or viewing only video tours of Princeton University. We also demonstrated a proof-of-concept for future full-scale integration with the pre-existing TigerApp food 4 u user interface; however, we note that this full-scale integration would likely take a couple of months to accomplish given the paradigm shift of operating on a 3D map rather than food 4 u’s current 2D implementation.

  2.2.2 Who would benefit?
  The Princeton University community will ultimately benefit from the proof-of-concept work that we are implementing into this project. Ultimately, we still intend to have our integration with the 3D map and the avatar selection and creation enhancing the experience of Princeton University’s community by complementing the existing 2D map and markers in the pre-existing TigerApp food 4 u. For the purposes of completing this final project, we hope to be able to offer a publicly available and engaging experience for

people who have had their mobility hampered by the COVID-19 Pandemic or would simply like to explore the campus of Princeton remotely. Additionally, recent studies have shown that playing video games can effectively help relieve depression and what better way than by exploring a virtual world with the ability to express one's creativity?[1] Our approach of gamification and directing a guided tour within the interactive suite that we have created will also provide a natural venue for newly accepted students considering Princeton University to explore the campus in a 3D way before they may be able to physically visit. This will likely permit better planning as well as pique their curiosity about exploring Princeton.

- **2.3 Previous Work**

2.3.1 What related work have other people done?

Previous work toward creating 3D map based UI has been implemented by Google with their creation of the 3D map which integrates with three.js for producing a geographically-based 3D mapping of the world. Many game developers, such as the companies Roblox and Bethesda Game Studios, have implemented vastly customizable avatar creation suites for RPG's and various other games. Additionally, the ability to customize one's digital representation has been baked into the business plans of some crypto-based companies seeking to enter the NFT market and permit the purchase and sales of personalized customizable digital assets that are globally unique. Another form of work carried out by similar projects is the virtualized campus tour platform Adora, founded by Princeton University students and recently acquired by fullmeasure, an education company.[2]

2.3.2 When do previous approaches fail/succeed?

Previous approaches succeed when the user-interaction is paired with a dynamic gameplay environment or there is a bit of novelty in the generation of the unique digital representation of the avatar or customizable digital asset, such as is possible with blockchain technologies. Previous approaches have also succeeded when they were addressing the acute needs presented by the COVID-19 Pandemic and provided a substitute for previously available tours. Previous approaches fail when there is disappointing gameplay, an irrelevant virtual world, a lackluster virtual world, or too few customization options for users.

2.4 Approach

2.4.1 What approach did we try?

We tried to implement a GUI-interactive avatar creation suite on top of an explorable world that reflects the actual disposition of buildings and roads on Earth. We think that the methods that we used to create the interactivity of the map permit users to feel like they are meaningfully interacting with the world and able to explore the campus of Princeton University and then venture out. Additionally, we feel that the gamification aspect of the map, with having markers that will indicate pre-plotted destinations for

---

[1] Joni Sweet, "Video Games Could Hold Untapped Potential in Treatment of Mental Illness," Verywell Mind, July 5, 2021, https://www.verywellmind.com/video-games-could-treat-mental-illness-study-shows-5190213.

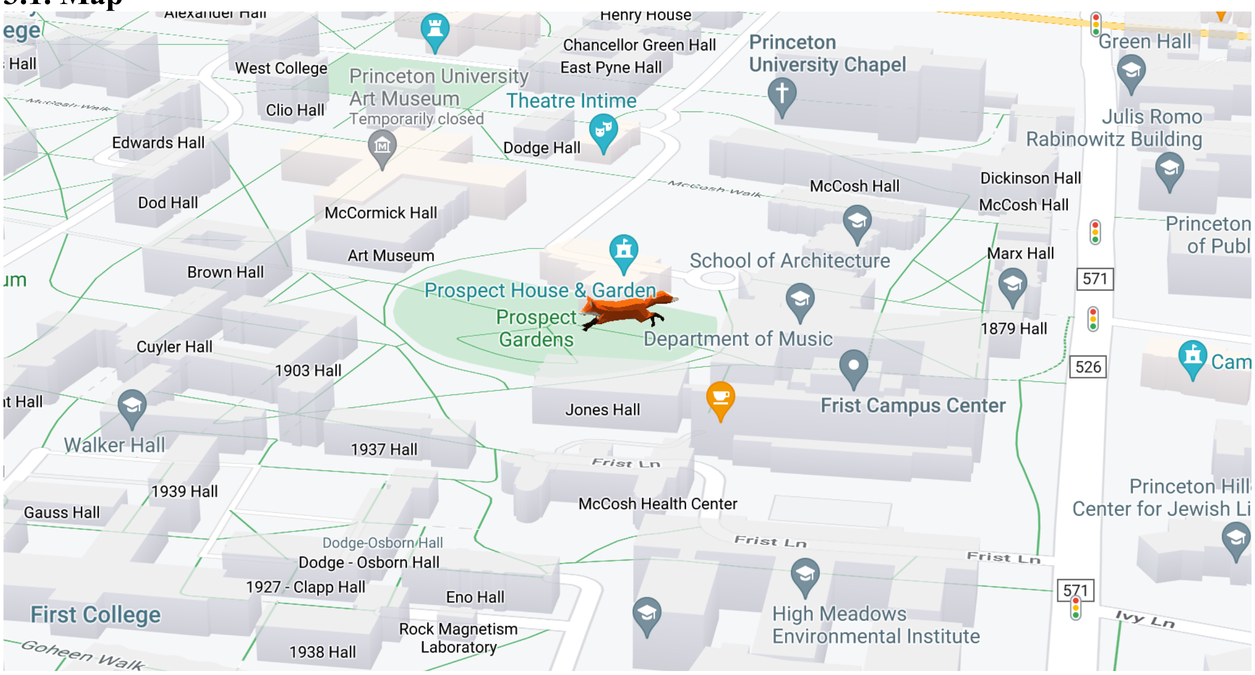
[2] "Adora — Full Measure Education," fullmeasure.io, accessed May 4, 2022, https://fullmeasure.io/full-measure-adora.

users, will add to the exploratory full of the experience. We think that our approach was suitable for this exploratory project. However, we feel that the use of so many new tools provided a very steep learning curve and some further experience with the tools would likely improve our ability to creatively imagine more extensions.

3. Methodology

To execute our approach we needed to implement interactivity for the 3D map, an interactive GUI, the creation of customizable meshes, map interaction for avatars as well as sensible animations, and importation for meshes from third-parties. Additionally, we implemented custom gamification experiences into the map to encourage users to explore and find markers.

3.1. Map



For implementing the 3D map, we used the fresh release of the 3D OpenGL google maps API which rendered a geographically rich environment for exploration on the map. An alternative to this approach would have been to custom design meshes for every building on Princeton's campus and the surrounding area and plot them on a drawn map of campus.

3.1.1 Initialization

The map was initialized in the way that Google prescribes for setting up their 3D overlay. We made customizations to this overlay setting experience and set it to be on Princeton's campus just below the University Store, where tour attendees would likely find themselves beginning the tour. We also initialized a starting point-centric mesh that acts as the avatar for the user.

3.1.2 Navigation

For implementing navigation for the 3D map, we found that using event listeners needed to be augmented with low-level manipulation of the GLOverlay object in order to present the effect of map navigation, carefully recalibrating the camera with every motion. This approach provided sufficient operability for the user to direct their avatar and move throughout the map upon keydown for up or down arrows. We also implemented the ability to rotate the position of the character and rotate the camera view with keydown of the left and right arrow keys. We additionally had to integrate the ability of the camera to follow the user and promote the immersive sense of the experience while being able to freely traverse the vast virtual Earth.

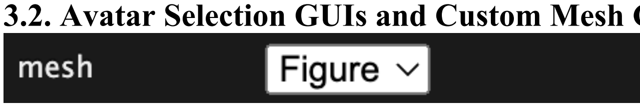
3.1.3 Avatar Animation Activation

If the selected avatar contains an animation, then this animation is activated as the keydown or keyup events are triggered. Getting this animation to run as smoothly as possible presented another challenge and we are exploring the use of TWEEN.js to give a more satisfying framerate. For meshes with multiple animations, we use the keyup as a signal to perform a stationary animation and the keydown to initiate the animation intended to represent motion. This animated motion opened the door to collisions between our selected avatar and the 3D map mesh objects. The provided google map 3D overlay presented challenges in how best to create a type of interaction with the elements that collided with their 3D outlines as physical boundaries. We do not intend to implement the ability to be repelled by the 3D building outlines when they collide with the avatar because this would prove to be too restrictive to the avatar's ability to navigate through narrow spaces between buildings.

3.1.4 Markers and Prescribed Tour + Gamification

Upon experimentally traveling around the map, using our arrow-key traversal technique, we realized that there was so much to see. We then thought to add markers as waypoints to encourage the navigation and exploration of the map, we found that the map provided a vast world of new and interesting areas which we plan to highlight with markers that can be discovered by users when interacting with the map. As for the markers on the map, we aimed to create a directed tour experience for our wandering avatars. These markers will highlight notable locations and provide a sense of accomplishment for the users discovering our campus treasures.

3.2. Avatar Selection GUIs and Custom Mesh Creation



For implementing the avatar selection GUI, we used Dat.gui and implemented the capability to select the meshes from a drop-down and then spawned a GUI that is relevant to only that mesh as it is displayed. This GUI contains options for the customizable features of the mesh, which for the anthropomorphic Figure include the selection of arm, leg, and head color as well as the torso.

3.2.1 Anthropomorphic Figure:

The anthropomorphic figure permits the selection of cylindrical, blocky, spherical, and other shapes for various body parts within the GUI-based avatar creation suite. This proved a technical challenge to implement as we found ourselves managing the creation of each body part.

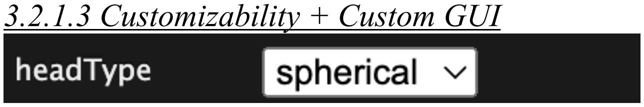
3.2.1.1 Initialization

createBody(), createHead() createArms() createEyes(), and createLegs() all initialize 3D three.js objects for the torso, head, arms, the eyes, and legs respectively by selecting appropriate shapes from three.js objects and initializing them. Additionally, each of these functions puts precise positioning and scaling into effect to present a recognizable anthropomorphic form.

3.2.1.2 Animation

This was accomplished using the gsoc animation library and provided the bouncing feature when the animate() function is called within BoxFigure.js. This offsetting of the position as well as the turning of the arms as if they were attached at the shoulders provides a recognizable improvement to the turning of the positioning of the arms through other means. We aim to produce a leg animation as well to signify walking or running in an observable way.

3.2.1.3 Customizability + Custom GUI



This was implemented with much retention of state for the initialized form as well as the custom spawning of a GUI that can be modified by the user to select actions that modify the appearance of the anthropomorphic form, such as the head shape. Selectable options for this mesh are supported through independently indicated code paths for each of the various geometries suitable for the anthropomorphic mesh's body parts. This permits the user to select the desired shapes and modify their avatar. This proved to be exceptionally challenging as managing the rendering, creation, destruction, and selection of various attributes each required considerable time to implement. Initially, the target was to provide robust customizations including a variety of customizable animations for the anthropomorphic mesh; however, the low-level manipulation of the mesh proved to be a significant impediment to progress.

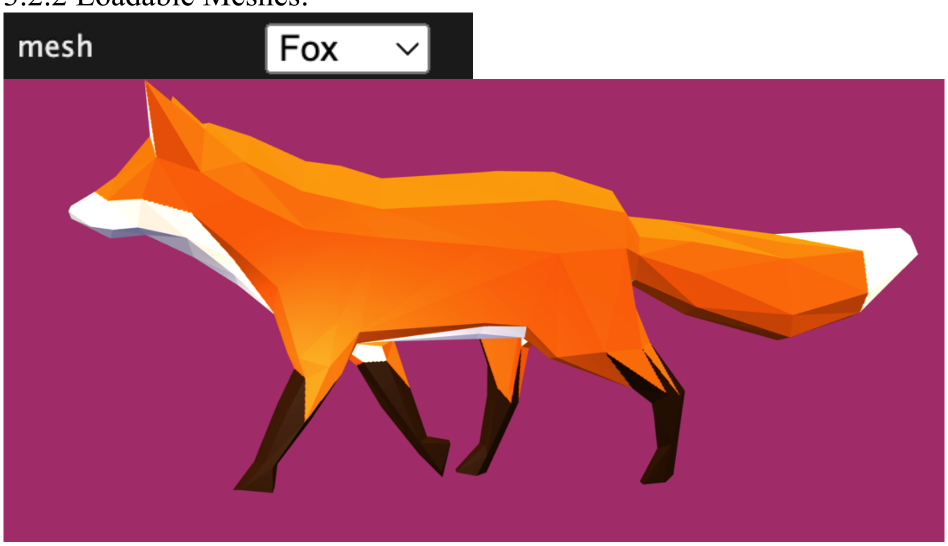
3.2.1.4 Exporting

One key feature that we implemented, as one of our stretch goals is the ability to export the GLTF format of the selected mesh with the export() function and activate it through the custom-spawned GUI for the anthropomorphic mesh. Additionally, to implement this we were able to use FireFox's built-in web downloader to export the file from the browser and download it on the user's local computer for future use.

3.2.1.4 Destruction

The deallocation of created resources for manipulating this anthropomorphic mesh requires the same low-level control and attention that it took to create the objects that make up its body parts. Additionally, this is then removed from the scene and the custom GUI that was spawned for the mesh must be deallocated as well.

3.2.2 Loadable Meshes:



3.2.2.1 Loading

We incorporated the ability to load existent meshes from external sources. To demonstrate proof of concept, we used a fox downloaded from an open-source github page managed by Khronos Group. The mesh was downloaded directly from the site using a GLTF loader with a url. In the constructor, three properties of the file were saved to the fox.state: the model itself, the 'gltf.scene' property, and the animation track, if it existed. An AnimationMixer rooted in 'gltf.scene' was also created and stored in the state– this is later used to control the playback of animation clips.

3.2.2.2 Animation

Finally, we added functionality to allow for the playback of the animation tracks stored in the gltf file. We retrieved the 'action' property to determine which animation track to play, and then used AnimationClip.findByName() to retrieve the correct track from clips, which we then played.

One brief hiccup that we faced was that when new animations were selected, they just seemed to play along with the old ones. We quickly discovered that AnimationMixer adds new clips to an array, and then plays them all simultaneously. To prevent the cumulative effect that resulted, we needed to clear the mixer of all tracks anytime we added a new one. However, we discovered that ‘survey’ was an animation that looked good with both ‘run’ and ‘walk,’ so we decided to keep the option open of overlaying them.

3.2.2.3 Movement

One the mesh is loaded and animated, we added functionality to change in position and orientation. This was done by adding an eventListener that changed the meshes ‘position’ and ‘orientation’ properties based on arrow key presses: ‘ArrowUp’ and ‘ArrowDown’ move the fox forward and back, respectively, while ‘ArrowLeft’ and ‘ArrowRight’ rotate it in either direction.

To create a realistic looking movement, we added code that set the active animation to “Walk” while the up arrow was being held down, and reset it to “survey” when the arrow was released. This created a pairing between mesh movement and the intuitive corresponding animation.

3.2.2.4 Custom Gui

Finally, we added a custom gui that contained controllers for different aspects of the mesh. In particular, we had an option to change the animation being played, as well as the speed of the animation.

3.2.2.5 General Model

What’s neat about this feature is that it can easily be expanded to include any downloadable mesh. Adding a new mesh would use the same scaffolding, and would simply require a new URL for loading the file, as well as a new drop down in the GUI based on the animation tracks available.

4. Discussion

The approach we took is promising, given the ability to interact meaningfully with the 3D map overlay. This means that there are ample opportunities to develop experiences in the map for users, and we hope to increase the amount of gamification that users can enjoy. The two options for meshes give us good flexibility in terms of avatar creation.

For future work, we would like to add more meshes, create more compelling gamification (like a celebration when users reach a marker), and fully integrate our results from this project into the TigerApp food 4 u. We realized about halfway into implementing the project that the integration effort with the pre-existing TigerApp food 4 u would require a significant overhaul of the existing functionality within the app which would stretch beyond the timeframe of this project’s first stretch, in this semester.

We additionally believe that the importation of these meshes as well as the ability to create mesh-based NFT’s for one’s personal collection would be very interesting. Additionally, if we could facilitate multiple users interacting on the map, this would provide a compelling alternative way to potentially meet other people in the virtual world and perhaps share fun facts for Princeton tour enthusiasts.

5. Conclusions

As acknowledged earlier, we believe that we succeeded in creating a 3D map-based virtual tour experience with customizable avatars. This was the goal of our project, and we have achieved a fruitful amount of progress that will serve as an extensible baseline for future engaging tour experiences.

6. Contributions

To accomplish this project we divided the tasks among the three of us. Aaron implemented the Anthropomorphic Mesh and attached GUIs. Ari implemented the Loaded Mesh interaction and attached GUIs. Ben implemented the map and its interactive features with markers.

7. Acknowledgements

We wanted to acknowledge the helpful feedback that we received from Professor Heide, our Graduate TA Adviser, Ethan, our Undergraduate TA Joanna. We would not have had a defined direction for our project without your advice and guidance.

8 Honor Code

We pledge our honor that we have not violated the honor code in the writing of this paper or implementation of this project.

/s/ Aaron Buck
/s/ Ari Braun
/s/ Benjamin Chan

Bibliography

fullmeasure.io. "Adora — Full Measure Education." Accessed May 4, 2022. https://fullmeasure.io/full-measure-adora.

Sweet, Joni. "Video Games Could Hold Untapped Potential in Treatment of Mental Illness." Verywell Mind, July 5, 2021. https://www.verywellmind.com/video-games-could-treat-mental-illness-study-shows-5190213.