

Virtuagram: A social media application for sharing 3D content

Justin Elias, Tyler Ewald, Lexi Wright
Group 12
Fall 2019, CSCI 491
Montana State University

November 26, 2019

Hosted At:
<https://dgram-8baf0.web.app>

GitHub Repo:
https://github.com/stryder03/csci491_group12

1 Objective

Internet technology, especially the introduction of social media platforms such as Instagram, connect artists with the public in ways never before possible. Unfortunately, the influence of social media platforms has not been all positive for the art world. Artists have had to conform their art to meet the specific limitations imposed on them by the platforms in order to reach their new audiences. These limitations are especially seen in the realm of three dimensional artwork such as sculptures and ceramics. In fact, we are now seeing 3D art that is being created specifically for viewing in a 2D space.

One of the main issues has been the inability to portray what is supposed to be viewed as an object that should be viewed from multiple angles, or even physically handled, in a two dimensional browser window. Sizes of sculptures are being decreased so that their detail can still be seen in the small square boxes they are displayed. Even shape and texture is being altered to give an appearance of depth in 2D, even though it makes the art look distorted when viewed in reality.

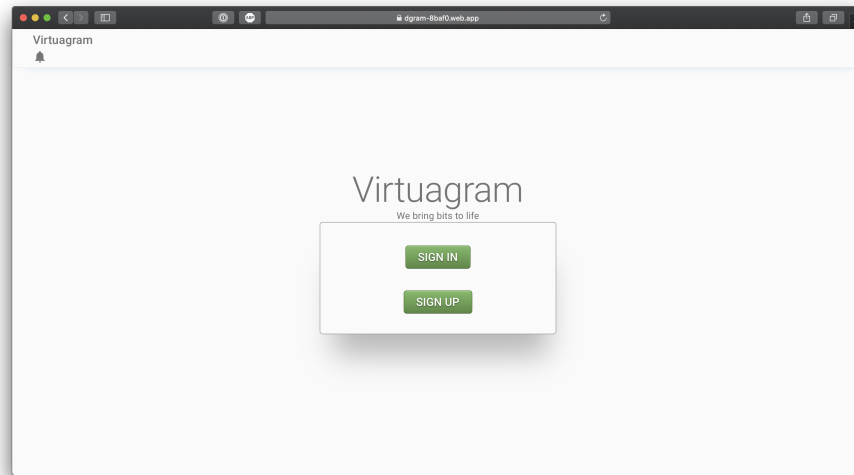
While frameworks and libraries have existed that allow representing 3D objects in a browser window have existed for almost a decade, there have been multiple barriers keeping social media platforms from implementing those solutions. Chiefly among those barriers was the increased data costs of sending the large files necessary.

With the introduction of 5G service imminent, that barrier may fall away. The time has come when an artists will be able to display their art as it was meant to be seen. A social media platform which is built around the WebGL platform is capable of showing 3D renders of an art piece that a viewer can view from any angle. Such a platform will also allow users to interact with objects, rotating them in the space, placing them in different lighting, or in different backdrops, and probably much more than what we can envision now.

Instead of limiting how an artist creates and displays their work, this platform enables artists to freely design and display their passion in ways that are unique, exciting, and new.

2 Summary

Virtuagram is a social media platform that allows a user to create, and consume posts containing 3d object files. The layout is built to be minimalistic, centering a user's attention onto the content of those they follow. Posts are featured in a card style with an infinity scroll timeline that loads content from selected creators the user chooses to follow.



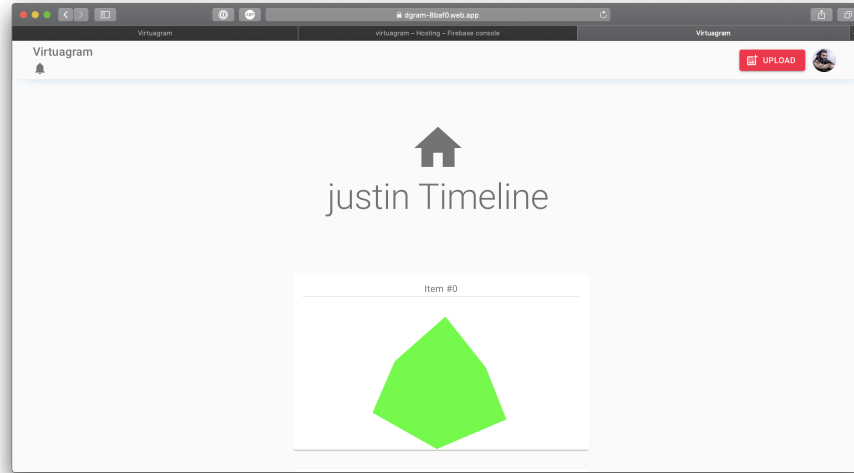
2.1 Technologies Used

React

The front-end of the website was created using the React^[7] javascript library that runs off of NodeJS. For our application, React worked very well. While there is a steep learning curve, especially regarding how higher order properties are passed to lower level states, the effort is worthwhile. A properly built component ends up being a fully self contained structure, including properties to customize its own css styles. By going this route, we were able to quickly iterate placing UI objects in different hierarchies by only changing a few lines of javascript. React also has a very large user base, which means that there are prefabbed components available for many tasks.

2.1.1 ThreeJS

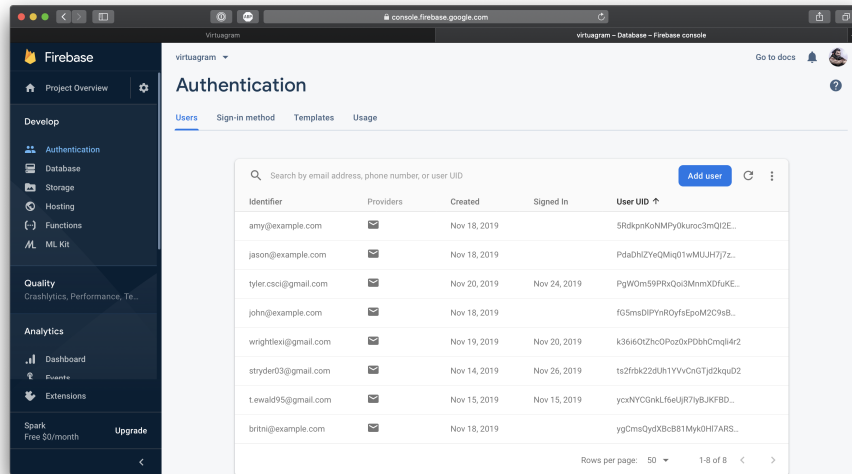
Threejs[9] is a JavaScript wrapper for WebGL. It allows a canvas to be created which can display 3D objects with generated light sources and camera positioning.



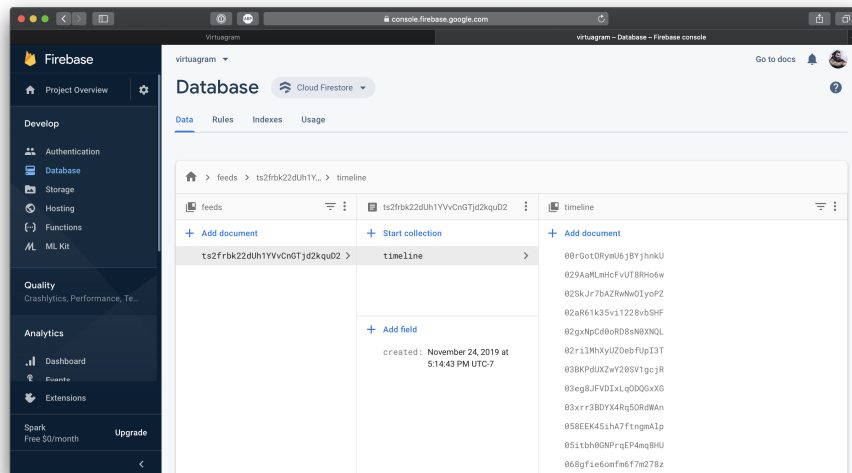
2.1.2 Firebase

Google's Firebase[4] is an all-in-one infrastructure management system for web and mobile apps. We choose this route because we felt that it would allow us to focus much more on the JavaScript design while handling many of our back-end integration concerns such as databases, storage, and authentication.

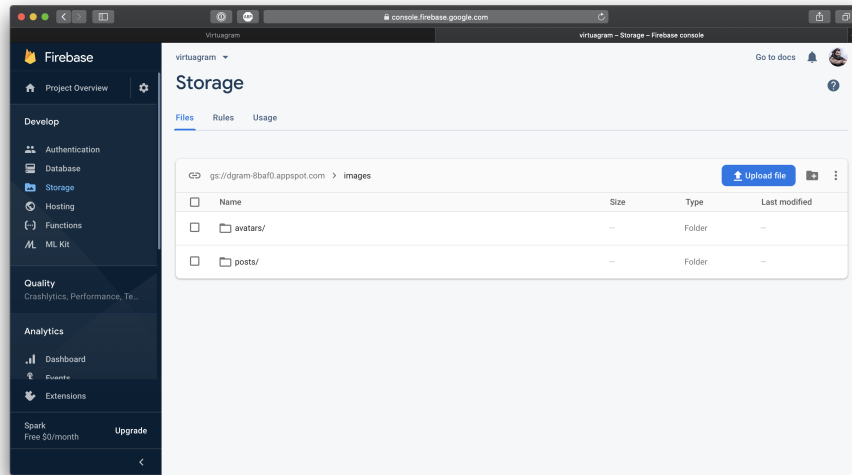
We chose to use Firebase's Authentication service with a email/password system,



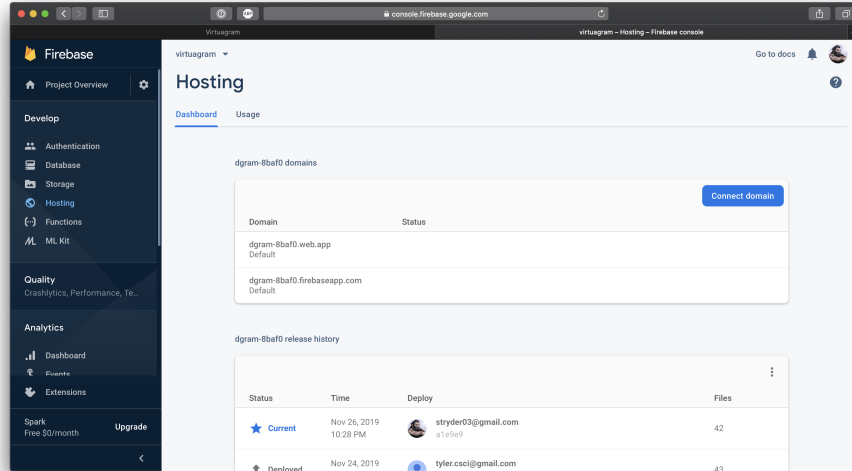
Firestore as a NoSQL database,



Storage for file hosting,



and Hosting as a free option to deploy our site.



In the end, we found out that Firebase was not as simple to setup as we had hoped. Although, after finding a templated tutorial^[1] for the authentication system, we were able to make quicker progress. Our key takeaway was that while Firebase is simple to setup, and the documentation is easy to navigate, the security rules are difficult to get correct. Most of our troubles stemmed from not understanding when the security rules applied recursively, and when they

did not.

2.1.3 Materials-UI

Materials-UI[6] is a styling package for React. The package supplies predefined React components for many of the most commonly used css styling functions. With these components, iterating through a visual design process by modifying the javascript files is simple and easy.

3 Member Contributions

3.1 Justin Elias

My work focused more on the back-end technology. I Researched what to use as a database, how to create a secure authentication system, and what type of server to build off of.

For Firebase, I setup the authentication system, Firestore and Storage, writing the security rules for each. Initially, the most difficult part was learning the syntax and API functions needed to tie the app into the Firebase system. In the end, I found the React+MaterialUI+Firebase[1] template on the MaterialUI website[6]. By using the authentication adapter from that template, I was able to connect Firebase authentication, and then extend that to create a adapters for post creation, and feed generation

I also helped work on the front end of the stack by building the post dialog component that allowed users to upload content to the storage server, the timeline generator component, and the post-card component with embedded threejs canvas.

3.2 Tyler Ewald

My work focused more on the design as well as the middle-man. Technology is constantly changing. For the longest time, the LAMP stack was the “go-to” for web applications, then modern technologies became more adept with frameworks like Angular and libraries such as React. Rather than being bound by the Linux OS, Apache server, MySQL database, and PHP stack, I designed the project to use the JAMstack[5]. The JAMstack doesn’t use specific technologies, it’s a concept on developing web applications that deliver better performance, developer experience, security, and has a low cost of scaling. JAM stands for Javascript, API’s, and Markup. It’s a modern architecture that is all based on the client-side. Static sites don’t have to be static, and that’s what I wanted to show.

I was also able to get the web application hosted using Firebase. This tended to be more difficult than planned. Firebase is a great database API, but it lacks in deployment and production builds. For web hosting, really the only two Firebase commands are “serve,” which shows you what the web application will look like at deployment, and “deploy,” which deploys the web application

to the generated web address. Every time we need to make an update to the hosted web application, we have to build and serve the app first using npm, and then set up all of the Firebase hosting dependencies to deploy. Another thing I added was cache clearing. Since Firebase doesn't have any form of cache invalidation, this needed to be added to our Firebase javascript object notation so there won't be any issues with past deployments. Future involvement of the project will be further three.js implementation and front-end modifications such as incorporating Gatsby.js[3].

3.3 Lexi Wright

I designed the color scheme, and lightweight feel of the UI. I customized all of the css, modifying the template to give a very appealing visual effect. I learned a lot during this process. I have never used github or react, so I navigated through that and got a basic understanding of how to change the HTML and CSS in react which was challenging and exciting to learn. Also with the help of my group mates I can use github for future classes. My biggest resource was the internet. I watched youtube videos about react and referenced the react page.

4 CONCLUSIONS AND FUTURE WORK

Virtuagram has shown that a social media platform focused on 3D content is viable with current technology. This project could be further worked on to provide optimization, especially when rendering threejs components. Currently, once a post component is mounted, it stays in memory causing a memory leak in the browser, eventually crashing the application.

Instead, posts need to be mounted only when they are focused, and unmounted otherwise. One method that we may explore is to have a static image of the object viewable in the timeline mode, with the 3D canvas only rendering when the post is clicked on. Another method is to use a mouse enter event to cause the post to mount the 3D canvas as the user scrolls. An example of this style can be found in the gallery style display on Sketchfab[8]. Finally, we can explore unmounting any post object that is not within the current view window.

Another optimization to work towards is integrating a publish/subscribe message passing system to populate user feeds and timelines. We have the current system working by having a user read in the posts that will show from the firestore database. Firestore is capable of handling this load, but the costs will scale upwards very quickly. Firestore charges for both reads and writes, with the current rate \$0.18 per 100k reads. By offloading post read functionality to something like Apache's Kafka[2] on Amazon's AWS, cost would become less volatile.

References

- [1] Linus Långberg. *Phoqe/react-material-ui-firebase*. original-date: 2019-01-12T22:23:31Z. Nov. 14, 2019. URL: <https://github.com/Phoqe/react-material-ui-firebase> (visited on 11/14/2019).
- [2] *Apache Kafka*. Apache Kafka. URL: <https://kafka.apache.org/> (visited on 11/26/2019).
- [3] *GatsbyJS*. GatsbyJS. URL: <https://www.gatsbyjs.org> (visited on 11/27/2019).
- [4] *Google Firebase*. URL: <https://firebase.google.com> (visited on 11/25/2019).
- [5] *JAMstack — JavaScript, APIs, and Markup*. URL: <https://jamstack.org/> (visited on 11/27/2019).
- [6] *Material-UI: A popular React UI framework*. URL: <https://material-ui.com/> (visited on 11/14/2019).
- [7] *React – A JavaScript library for building user interfaces*. URL: <https://reactjs.org/> (visited on 11/25/2019).
- [8] *Sketchfab - Publish & find 3D models online*. Sketchfab. URL: <https://sketchfab.com> (visited on 11/26/2019).
- [9] *three.js – JavaScript 3D library*. URL: <https://threejs.org/> (visited on 11/26/2019).