# What is/are your name(s)? What assignment group are you?

Daniel Vo

# Which framework did you select (Fitbit, Alexa, A-Frame)?

A-Frame

# Q1: How, if at all, does this framework support package and library management?

Although A-Frame does not have a direct package manager such as PIP or NPM, A-Frame does allow integration with JavaScript libraries and frameworks seamlessly. Users can use this to extend A-Frame’s functionality and also integrate other web technologies. For example, in this project, we used external libraries such as the *spacetime* library to display the time in different time zones.

# Q2: How, if at all, does this framework support principles for code separation, like Model-View-Controller? Would the separation principles effectively support creation of a larger application? Why or why not?

A-Frame does not use the traditional Model-View-Controller (MVC) pattern, instead, A-Frame uses an ECS (Entity-Component-System) architecture. ECS follows the composition over inheritance and hierarchy principle. This allows for a better separation of concern among the components of the scene. For example, in our project, the components of the scene live within the clock.js file, while the HTML scene structure remains in a separate file.

The ECS architecture effectively supports the creation of a larger application for a multitude of reasons. For example, ECS architecture allows for greater flexibility when defining new objects by combining old and new reusable parts, provides a proven architecture for VR and 3D development, and allows for extending new features upon itself ([A-Frame Documentation](https://aframe.io/docs/1.7.0/introduction/entity-component-system.html)). These features help when a project scales in complexity and size as it keeps development easy to manage and simple to continue building on.

# 

# Q3: In lecture, we discussed a few design recommendations for each respective device. How, if at all, does the framework support the recommendations for that device? What recommendations are left to the developer to decide how or whether to implement?

A-Frame is designed for WebVR and WebXR experiences, which means it supports interactive 3D and VR apps that run on desktop and mobile browsers, as well as, VR headsets. It provides built-in support for spatial interactions (e.g., gaze-based and hand-tracked interactions).

However, certain design recommendations and considerations are left to the developer to decide. More specifically, the developers can decide different ways on how they want to optimize their app for performance, ensuring that their app is accessible to all users, and implementing a responsive UI design among different device types.

# Q4: What appear to be benefits to using web technologies for development on this device? Conversely, what advantages might a native framework have over the framework you used?

Some benefits for using A-Frame for development include the support for cross-platform compatibility, ease of use, and web-based distribution. A-Frame’s applications can run on multiple platforms such as desktops, mobile devices, and VR headsets all together without any extra requirements. Then, A-Frame’s use of HTML-like syntax makes it easy and intuitive to get started with 3D and virtual reality development. And finally, with A-frame, users don’t need to install anything else to run the app, users can gain access straight from within their web browser.

However, there are still other advantages that a native framework such as Unity has over A-frame. This includes: performance, access to hardware features, and advanced interactivity. Other native frameworks are typically more optimized to handle and render complex 3D scenes and physics. Additionally, native apps might have more access to different hardware features such as the ability to integrate themselves with different device sensors and APIs. Finally, native frameworks can provide more control over input handling and physics–having an increased and more advanced interactivity capabilities.

# Q5: What did you find easy and challenging about development in this framework?

Something I liked and found easy using A-Frame was how simple it was to set up a scene and build custom components. It’s very plug-and-play-esque and intuitive to use, especially after reading the documentation. One thing I found challenging during development was the overall 3D development environment. I had never worked with a 3D environment before, so it took me a while to understand how and where to place certain components.