William Nguyen-Luu

Nostalgia Portal

Motivation

I started this project with an interest in the idea of nostalgia and how different generations experienced it through screens, media, and everyday aesthetics. Instead of building a normal website, I wanted to create a “nostalgia machine” that reacts to users and places them inside a themed environment based on the feelings they associate with their past. I was interested in the way people from different age ranges would get a feeling of nostalgia from completely different things: some might think of iPads and YouTube, others of VHS tapes or black-and-white TV. The project dives into: what does nostalgia *look* and *feel* like when translated into an explorable digital space? And how can that be shared with many people at once, with their input stored, counted, and visualized over time?

This project is also motivated by the idea of combining emotional experience with networked data. Nostalgia is mostly a personal matter, but in this project it becomes something that is aggregated, counted, and visualized as a global distribution of “eras.” Each user would fill out a quiz which then reflects how they relate to technology, memories, media, and places. Their answers are not just used to generate a result for them individually, but are also put into a shared database. This makes nostalgia both individual and collective: users get their own era environment, but they also see how their responses compare to everyone else who has visited the portal.

Intentions and Concept

In terms of concept, the project is designed as a **Nostalgia Portal**: an 80s retro-inspired home screen that leads to three main parts: the quiz, the statistics page, and the era-specific screens. The quiz is meant to feel simple and easy to use. Each question is written around a nostalgic theme (toys, technology, music, comfort foods, places), and each answer is mapped to one of four eras: **0–17**, **18–30**, **31–50**, and **51+**. The intention of this website is not to guess the user’s real age, but to place them in the era that matches how they feel the most nostalgia.

Each era corresponds to a unique environment inspired by old screensavers and early 3D computer graphics. Since this project will be done on a computer screen, I figured why not based the concept around screensavers. The 0–17 era places the user inside a maze-like space with era-appropriate textures, movement, and game-like navigation inspired by the old school Doom games. The 18–30 era uses a morphing 3D cube with scanlines and CRT glow inspired by late 90s screens and DVD visuals. The 31–50 era becomes a starfield with slowly drifting astronauts and other space-related objects, mixing old space aesthetics with floating low resolution imagery. The 51+ era uses a starry space with rotating geometry that feels more abstract and contemplative. The intention is that each screen evokes a different flavor of

nostalgia, but they are also united by the idea of a looping “screensaver” world that you can explore, not just watch.

Implementation, Technologies, and References

The project is implemented as a **Flask application** with a **MongoDB Atlas** database serving as a memory storage for users. Flask handles routes such as `/`home, `/quiz`, `/submitQuiz`, `/stats`, and the four era screens. The quiz questions are stored as a Python list of dictionaries, where each question contains its text and a list of choices. Each choice stores three things: the era value (like `"18-30"`), an associated weight, and the label shown to the user. When a user submits the quiz, the `/submitQuiz` route loops over the questions, checks which era was chosen for each one, and adds the corresponding weight. The era with the highest score becomes the final result. That result is stored in the `nostalgiaQuiz` collection in MongoDB.

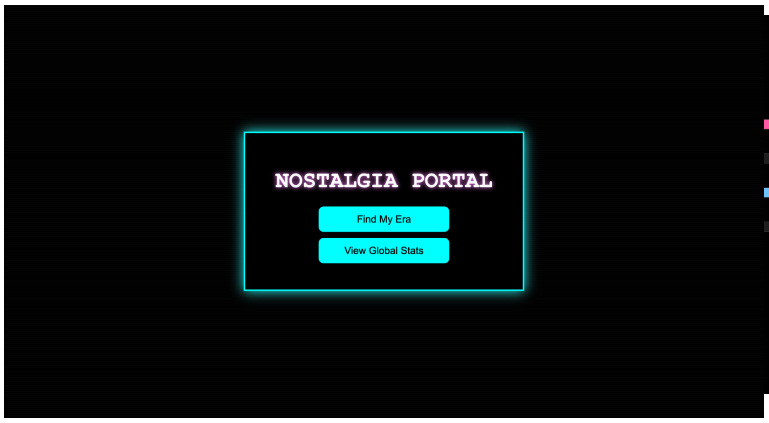
On the front-end, the project uses HTML, CSS, and JavaScript, along with **Three.js** for the 3D screens. Each era screen has its own HTML file that takes up a full-window `<canvas>` and uses Three.js to create different visuals: for example, a morphing mesh for the 18–30 era, a starfield built with `THREE.Points` and sprites for the 31–50 era, or a maze/tunnel built with geometries, textures, and a moving camera. Interactivity is event-driven: mouse moves, mouse down/up, and keyboard events (WASD, arrow keys) control camera movement, dragging of sprites, or rotation of geometry. For audio, I use simple HTML5 `Audio` objects to loop background tracks that fit each era and to create ambient sound to enhance the user experience.

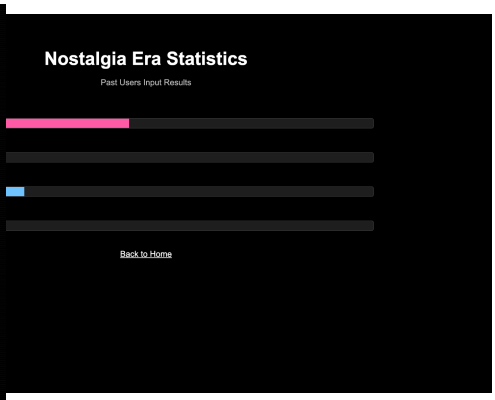
The statistics page (`/stats`) retrieves aggregated data from MongoDB using `count_documents` for each era and then passes the counts and total into a template. The stats are visualized as colored horizontal bars for users in each era. This creates a simple visualization of past users input results. I consulted the pages on moodle as well as past in-class exercises for routing and form handling. For the Three.js, meshes, sprites, and particle systems, I used OpenAI’s ChatGPT to help me understand and create the appropriate environment for each era/screen. I also use it as an assistant for debugging.

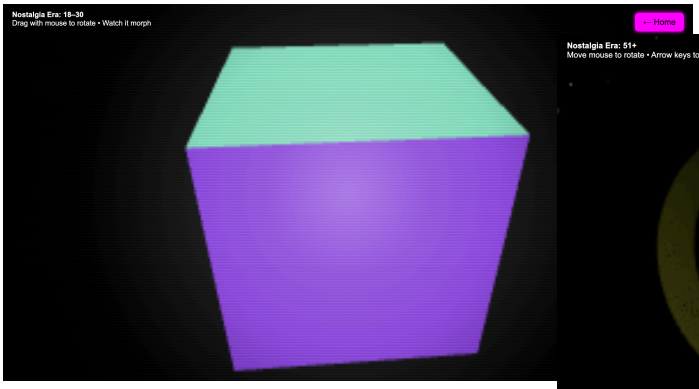
Expected Outcomes and Reflection

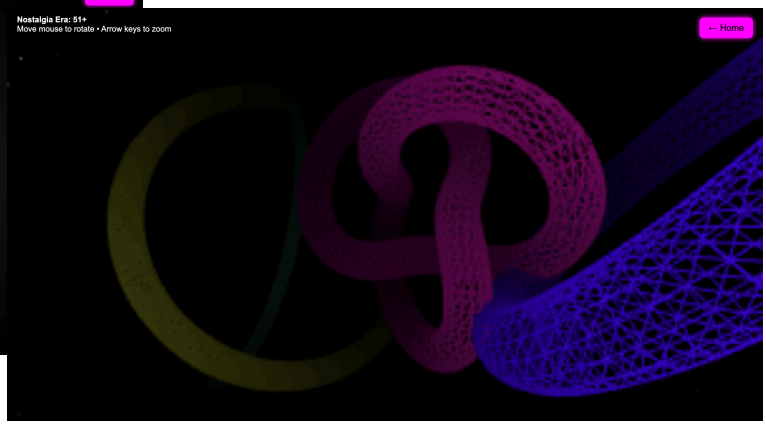
The expected outcome for my project is for users to get a feeling of being brought back into a nostalgia era that feels accurate. Even though the quiz is short, the weighting system is designed so that different questions have more impact than others which results in the appropriate era. After seeing their era, users can enter the corresponding screen and experience an interactive, ambient, screensaver-like environment that can also be interacted with. The hope is that the visuals, motion, and sound evoke subtle feelings of familiarity or

distance, depending on the user. From a networked perspective, the statistics page turns individual quiz responses into shared data that can be visualized. This transforms nostalgia into a kind of living dataset. If I had more time, I would expand the quiz with more personal questions, add more era screens.









William Nguyen-Luu