T.A. Marryshow Community College

Computer and Electronics Department

Student Name	Ulric Aird & Trevon Friday
Project Title	Content Delivery Platform
Due Date	19th January, 2025

Table of Contents

Executive Summary

Ztube is a content-driven platform designed to combine video streaming and blogging functionalities in one space. The primary goal of the project is to develop a responsive web-based platform where users can create, share, and consume diverse content, fostering community engagement through discussions. The platform addresses the growing need for creators to share content freely, without the restrictive algorithms often found on other platforms, and aims to build a supportive user community.

Introduction

Problem Statement

Existing content platforms prioritize algorithms over user engagement which makes it difficult for smaller creators to gain visibility and recognition. Users often struggle to discover various content tailored to their interests. Ztube addresses this by creating a fair, community-driven space that prioritizes authentic engagement, content discovery, and transparent sharing experiences without an unwanted algorithm.

Target Audience

Content Creators:

- Aspiring creators looking for fair visibility and engagement opportunities.
- Established creators seeking a platform that values transparency and community connection.

Content Consumers:

- Individuals seeking fresh, authentic, and personalized content experiences.
- Users are tired of algorithm-dominated platforms and looking for diverse content curation.

Community Enthusiasts: People eager to participate in discussions, upvote content, and influence the platform's ecosystem.

Goals and Objectives

Goals

- To provide users with tools for content creation, including video uploads and blogs.
- To offer a space where users can engage in discussions and interact with creators.
- To build a recommendation system that enhances content discoverability.

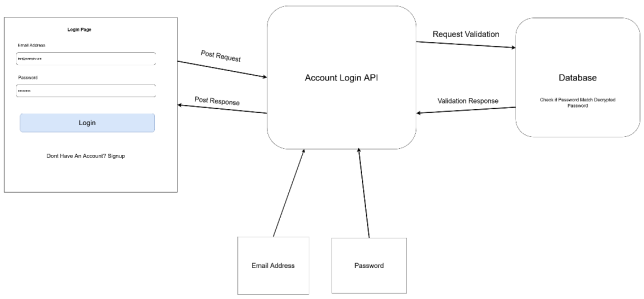
Objectives

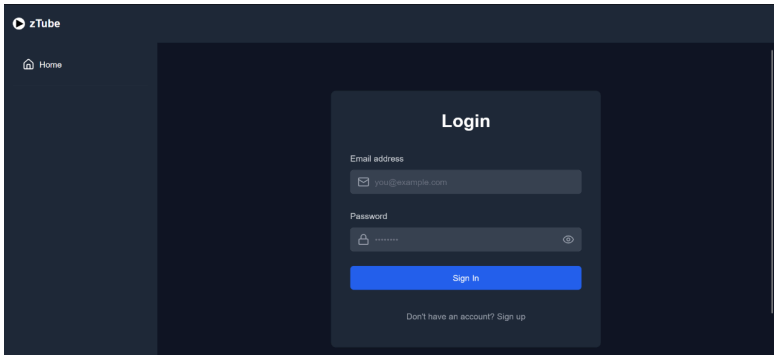
- Develop features that prioritize community involvement, such as voting systems, creator spotlights, and direct audience interaction tools.
- Implement a fair content discovery algorithm that balances creator effort and audience preferences.
- Create a robust monetization model with clear policies, ensuring creators earn a fair share.
- Regularly engage users through campaigns, challenges, and live events to foster a vibrant ecosystem.

System Architecture

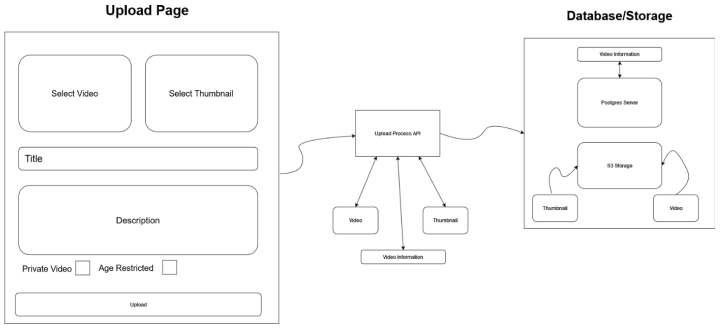
zTube was created using the NextJS React Framework using Supabase as its backend. Other technologies used were FFmpeg for video compression, next-pwa, a plugin for converting your Next.js App into a Progressive Web App (PWA), and ShadCN for UI components.

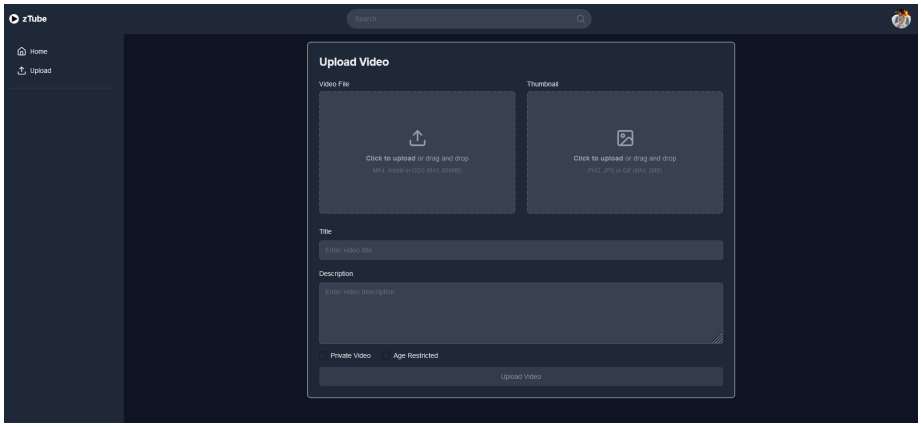
Authentcation Data Process



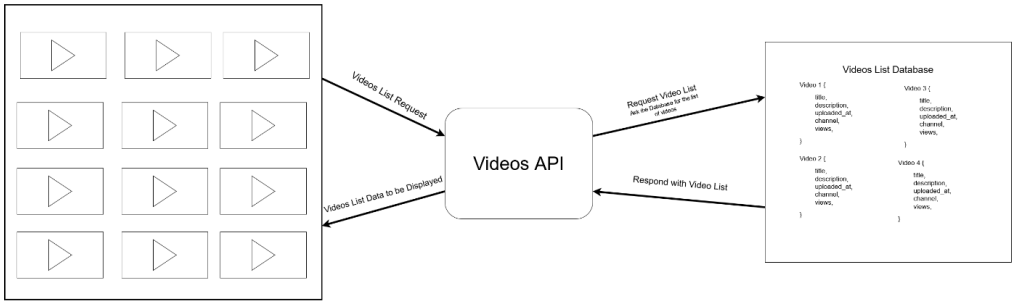


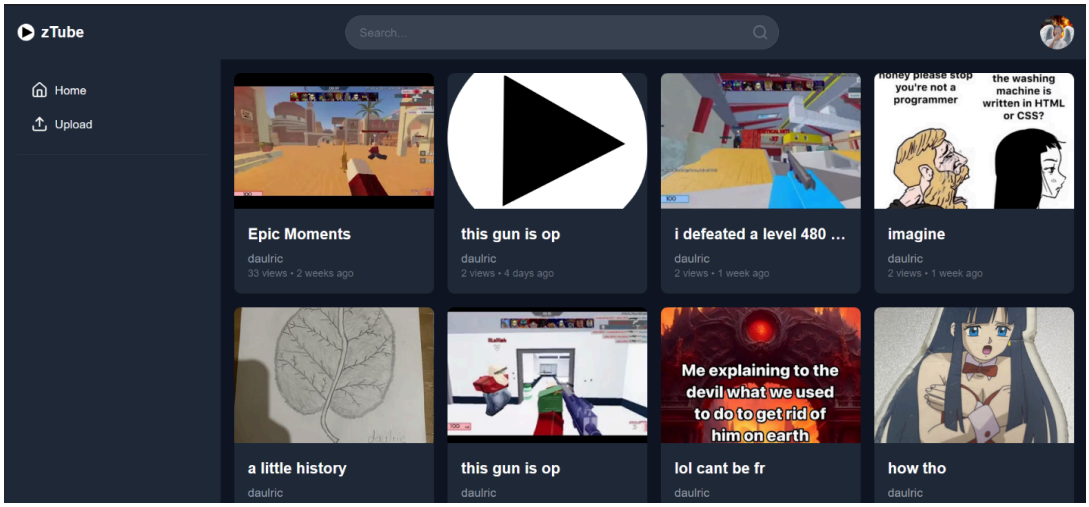
Upload Data Process





Video Display Process





Features

A content-driven platform offers various features designed to engage users, foster community interaction, and facilitate content creation and consumption. User management is a core component, including registration and login options via email and password. Users can customize profiles with pictures, bios, and social media links. User roles, such as admins, moderators, and contributors, ensure effective content moderation and management.

The platform provides robust content creation and management tools, including a rich text editor supporting multimedia like images and videos.

For content consumption, the platform offers personalized feeds powered by algorithms and advanced search functionality with filters. Community features play a vital role, enabling likes, comments, and threaded post replies.

HCI Principles

The UI/UX of zTube follows established Human-Computer Interaction Principles:

- **Usability**: The interface is innate and easy to navigate, making sure that there is an optimal user experience.
- **Consistency**: Design elements follow a universal style which improves accessibility and stability.
- **Feedback and Responsiveness**: Feedback was provided for user action while fostering engagement.
- **Minimal Cognitive Load**: Layouts and interactions are made clear, which reduces confusion and improves usability.

Testing and Results

The responsiveness of mobile and other devices is very good. The website can operate on any browser, such as Safari, Chrome, and other various Chromium browsers, Firefox, and other Gecko Engine Browsers. All interactive elements within the app perform as they are supposed to.

Challenges and Solutions

One major challenge while creating and deploying this app was Cloudflare's cache control. I know that Cloudflare has the best cache control system in the world, but due to the nature of the app, it had to be disabled. Also, another challenge we faced was creating the APIs, because we didn't want the API to break our backend infrastructure. We solve this by carefully constructing the API, with all checks and error handling to mitigate these issues. Also, we made sure to protect user assets such as videos and images by using the Supabase Signed URL feature.

Conclusion

The Ztube project is progressing well, with the design phase nearly complete and development fully underway. By actively incorporating user feedback and tackling challenges promptly, we are building a platform designed to meet the diverse needs of content creators and audiences alike. Moving forward, our focus will be on refining core functionalities and preparing for a smooth and successful launch, ensuring a robust and engaging user experience.

Appendix

Budget Plan

- **Hosting:** $0 for using hosting services like Vercel. $20 per month in case the application needs scaling.
- **Domain:** $12 per year for securing a unique and memorable domain name.
- **CMS and Plugins:** $25 a month for integration of tools like Next-PWA and Supabase.
- **Design Tools:** $0 for using Shadcn and Figma for UI Development.

Communication Plan

- **Meetings:** Regular discussions for design approvals and feedback sessions.
- **Content Gathering:** Collaboration with creators for initial content uploads.
- **Feedback:** Periodic user feedback to refine platform features.

Timeline

1. **Wireframes & Prototyping:** Initial sketches and designs are expected to be finished within January 2025.
2. **Content Upload & Testing:** This testing phase is expected to take place from February - March 2025.
3. **Functional Testing:** Comprehensive testing across devices and browsers is expected to take 3 weeks in March 2025.
4. **Launch:** Official platform release in April 2025.

Risk Management Plan

- **Risk Identified:** Downtime during deployment.
    - **Mitigation:** Deploy updates during low-traffic hours and maintain backup systems.
- **Risk Identified:** API instability.
    - **Mitigation:** Implement robust error handling and monitoring systems.

Testing Reports

- **Browser Compatibility:** Logs confirming functionality across major browsers.
- **Error Reports:** Detailed logs of bugs and resolutions.
- **Fixes:** Updates ensuring all features work as intended.

Technical References

- **Frameworks and Tools:**
    - NextJS
    - Supabase
    - Shadcn
    - Figma
    - Vercel
    - Cloudflare

- **Documents**

  Documents used in creating the Documentation

https://github.com/daulric/College/tree/Year-2-Semester-1/Capstone%20Project