EZ-Drive

What is EZ-Drive?

EZ-Drive is a web application designed to function as a hub for uploading files to multiple cloud services. These cloud services being Google Drive, OneDrive, and Dropbox. Users of this application are able to create an account and connect each of their cloud service accounts through the power of OAuth 2.0 Authorization. This provides a fast and secure way of linking our user's accounts.

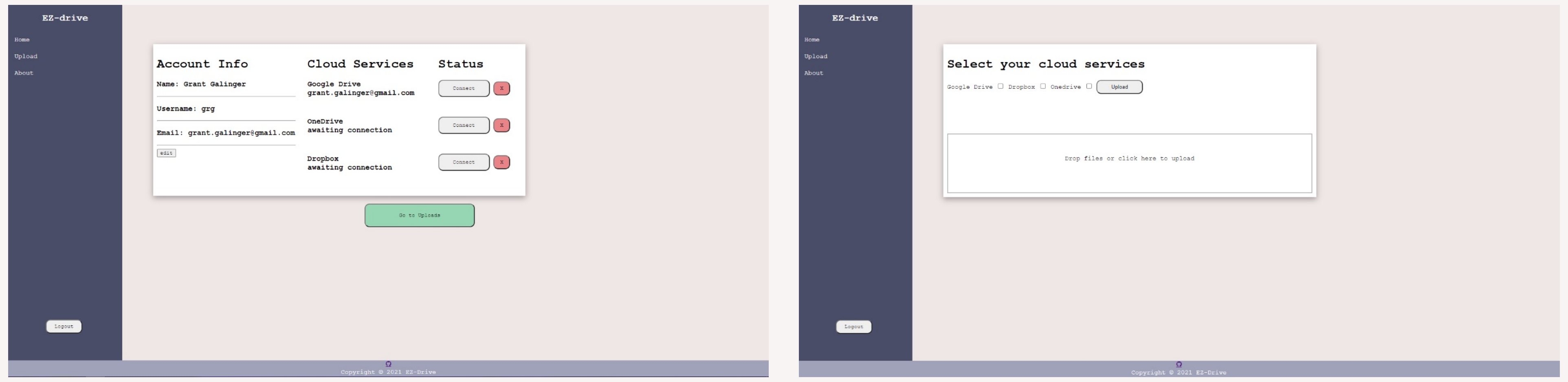
Our goals for this project were to provide an easy solution for backing up user files, to combine multiple APIs into one web application, and to create well designed, aesthetically pleasing web interfaces.

Approach & Design Decisions

EZ-Drive was designed to support an easy and seamless user experience. From creating an account and logging in, to connecting your cloud service accounts, we wanted each step in the process to feel intuitive. We opted for a sleek and simple design when it came to our web pages. We felt that this approach would best complement our design goals.

At its core, the system design follows a full-stack web application methodology and is split into four parts. The frontend was built using HTML, CSS, and Javascript. The backend runs on an Apache server and was written in php. The database being used is phpMyAdmin, which hold all of our user and authorization data. Lastly, cloud service resource data is retrieved through their respective API libraries, and user authentication is certified through the OAuth 2.0 Authorization protocols.

Figure 1. Application Screenshots



EZ-drive

Home

Upload

About

Account Info

Name: Grant Galinger

Username: grg

Email: grant.galinger@gmail.com

edit

Cloud Services	Status
Google Drive grant.galinger@gmail.com	Connect X
OneDrive awaiting connection	Connect X
Dropbox awaiting connection	Connect X

Go to Uploads

Logout

Copyright © 2021 EZ-Drive

EZ-drive

Home

Upload

About

Select your cloud services

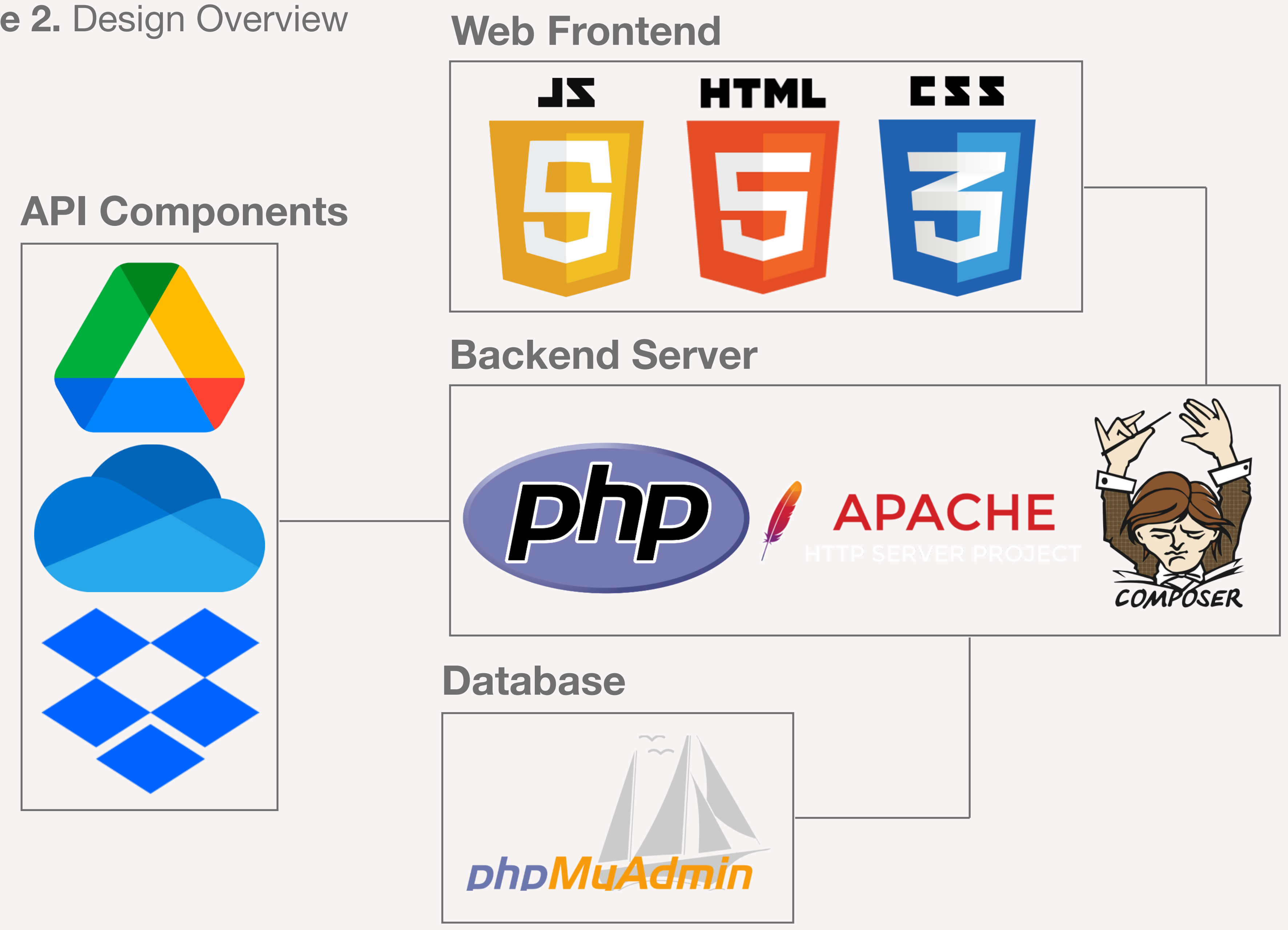
Google Drive □ Dropbox □ Onedrive □ Upload

Drop files or click here to upload

Logout

Copyright © 2021 EZ-Drive

Figure 2. Design Overview



Web Frontend

API Components

Backend Server

Database

Challenges Faced

Configuring Cloud Service APIs: In order to setup the API components for each of the cloud services, we had to rely upon the documentation provided for their developers. Each component was build individually at first, and then added to the main project.

OAuth 2.0 Authorization: In order to access user resources from each cloud service, this authorization protocol was used. It required understanding of client credentials, authorization requests, redirect URIs, and several other components.

Utilizing Composer: Composer allows us to import external php libraries that house functionality for the cloud service APIs. Initially, we were facing issues when importing all three of our required libraries together, which is needed for our project.

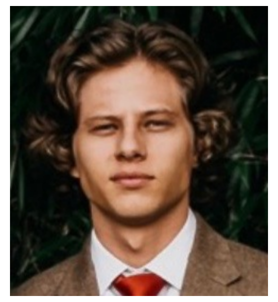
Accomplishments

Implementation of Model: Using Google Drive, OneDrive, and Dropbox API components, we created a file uploading system for users to backup and save files to multiple cloud service accounts at once.

Full Stack Web Application: With a running database, multiple functional API components, and a frontend to display it all, we created a full stack web application.

Experienced Gained: Each of us used a technology we were unfamiliar with. The three cloud service APIs were not used by any of us before this project, and utilizing the php components, Composer and phpMyAdmin, was new to most of us as well.

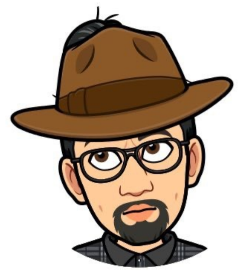
Grant Galinger
Computer Science



Kyle Spraggins
Computer Science

Josh Phillips
Computer Science

Prof. Fred Annexstein
Project Advisor



University of CINCINNATI