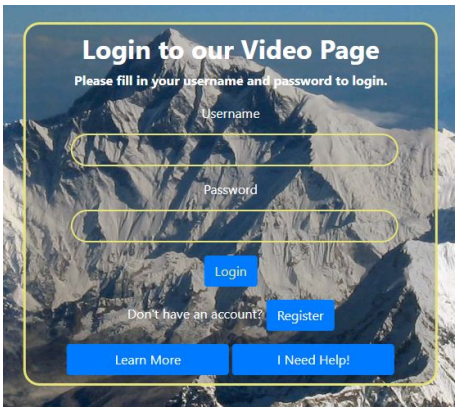
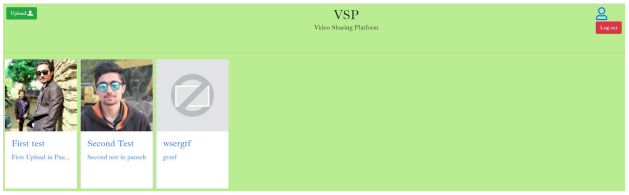
Internet Systems Programming
Kyle and Yuvraj

Final Project Report

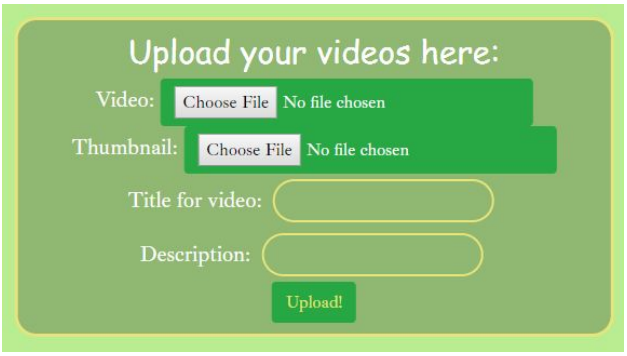
Implementation Details



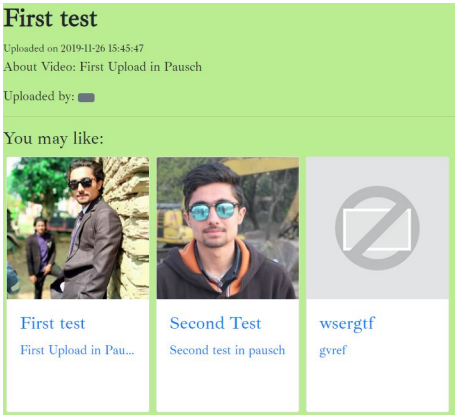
Upon visiting our site, the first page that a user will see is a login page. If a user has already made an account, they can login using their registered username and password. If they have not, they will have to go to the sign up page and make one. Registering on our site stores the user's chosen username in a database along with a hashed version of their password, so that their password is not stolen. Registration only requires choosing a username not in use and a password.



Once logged in, the user is sent to our homepage where they can view a list of currently uploaded videos, choose to sign out of their account, or go to the upload page. On the upload page, the user may choose to upload a new video from their computer, complete with a title, thumbnail and description for the video. Uploading a new video stores the video and thumbnail in storage folders on the server and the other related information in a separate database. This includes the name of the video, the storage location of the video and thumbnail, and the user provided description for the video.



Clicking on one of the listed video previews on the homepage will open the video in the user's web browser using the html video tag so that they can watch it. It will also display the video's title, description, and upload date underneath the video, along with who uploaded it and a list of other videos for the user to watch.



Contributions Of Group Members

Our original intent was to have the website be accessible through Amazon Web Services, so I (Kyle) had made an AWS account so that we may work on doing that. I set up an Elastic Beanstalk (EB) instance for us to host our website on and setup an RDS database for us to use as well. The login and homepage were written by Yuvraj and uploaded to the EB instance, but proved to have some sort of incompatibility issue with EB and how we were connecting to the database, so we uploaded it to Pausch instead and got it working there. The database for video information is still hosted on AWS, however the videos themselves are not. I set up an AWS S3 bucket so that we could host the videos on AWS, but we ended up not using it after moving the website to Pausch since we would have had to install the AWS SDK. The page for uploading new videos and viewing them was started by me, but Yuvraj ended up finishing it. I added the technical document and the online help to website, along with the popups for viewing them. Yuvraj and I both worked on the css for the website, making any changes we each liked.

Lessons Learned

Creating a website that can host user uploaded content for other users to view is not an easy task. We ran into a lot of issues that required us to change the way we were doing things.

Amazon Web Services seemed like a good service for us to host our website with all the different services that they offered that we could use. However, using Elastic Beanstalk gave us a lot of trouble for seemingly arbitrary reasons; so, after we moved the whole website to Pausch, we thought that would be easier to deal with. But, that gave us issues with allowing the user to upload their own videos since, after some testing, we figured out that Pausch is not configured to allow html forms to submit files. In the process of all this though, we learned how to setup and use some of AWS's different services and how to properly store file uploads on a server, while storing the file path in a database so it can later be fetched.

Future Work

Compared to Youtube, our website is very barebones. It's only able to store login information for users and basic info about uploaded videos. In the future, we would want to work on tying more information to both users and videos. We would want to allow users to be able to make their own profiles and tie video uploads to the user that uploaded them. We would also want to tie an email to the user account so that they can login with that instead, if they choose to, and be able to recover their account if they forget their password. Videos could also use more information about them, such as upload date, view count, or even comments made on the video. Improving the look of the site with the use of css is also something we would want to do.