YWCC Capstone Project Final Report

Digital Document Upload

DocuDrag.com

Team Members:
Kerim Sever
Ali Alkhateeb
Param Patel
Aaron Patel

Course: CS491
Professor: Dr. Eljabiri

4 August 2022

Table of Contents:

Chapter I: Introduction

- (Project Background, problem def, glossary of terms, iterations)

WillLouden.com is a company that is looking to transition its documents and files online on a platform where its customers, employees, and suppliers can upload documents. This project had the goal to create enterprise-level digital documents. The idea is to upload, encode, assign, and download documents to and from the database. Their goal is to allow company employees/customers to share documents in a management system that can be monitored by the CEO. They want a system that encodes documents into a database and can be able to manipulate and filter documents. Their five-point product process was Filler accounts in database, Management account roles, Class management system, Encoded upload manipulate files, and Filters. The current problem for WillLouden is their customer and employment document system is through email and hard copies but they have no actual system to transfer files. This would cost the company more time and money since the employees have to wait a while to receive the proper documents and the employees have to wait for the documents to be sent back to them. WillLouden was looking for an enterprise solution of a file management system that can replace their current dilemma of paperwork/documents. This was a great project idea and as a team, we had a few concept solutions in mind before we started.

When we met with the industry team we discussed our solution methods and they were undeceived of what method we should use first. After a few meetings and good communication, we were able to come up with a solution using a flask application. A flask application is a micro web framework that is written in python. We used this because we all had some prior experience

with Flask and we wanted to enhance our skills from our previous experiences in this project. Flask was a great way to incorporate a strong back-end and front-end system that would be able to connect to a database and hosting system and we all concluded to use Flask in our project. We pitched a few prototype layouts for all the functionality and designs we could do and we were able to amaze them with our ideas. When it came to the development however one thing that was difficult at first was a way to disperse the work and create the actual application. We had to take some time to learn and understand exactly how we should set up the application and how we can do all the connections we promised. With good communication, hard work, and an effective strategy we were able to overcome all obstacles and complete the project and exceed their expectations!

The applications and imports we used were BytesIO, Flask Render_template, request, Send_file, redirect, session, SQLAlchemy, MySQLdb, and MySQL.connector. These were all important functions that were used to help create our enterprise-level web application. BytesIO is to iterate and manipulate strings to encode the documents in the system. Render_template was used to generate an output template for our application. The request allows us to obtain data sent from a client. Send_file to transfer the content of the files uploaded/downloaded. Redirect to switch the user from the home page to a login page. The session is the time interval a user logs in and out of the server. SQLAlchemy to connect our flask application with the database in a simplified manner. MySQLdb to establish a connection to the database. Mysql connector to connect the user to the SQL database.

Our iteration methodology was a challenge to come up with because there were a lot of options to choose from such as scrum, waterfall, and other concepts. As the Project Manager, it was my responsibility to make sure everyone was on the same page and assign weekly iterations

to the team. We discovered and used a lean-lifecycle agile approach to complete the project and we broke up the work into iterations throughout the weeks. We had multiple weekly meetings discussing what needs to be completed and each team member would choose what they think fits them best and complete it within an adjustable timeframe. This was to account for any problems or issues each team member might come across throughout their iteration. We then meet with the industry team to discuss what iterations were completed and what problems occurred. We also used this opportunity to discuss potential ideas or changes they wanted us to complete to satisfy their needs and not come back and redo work for any misconceptions. When we did come across a problem we would focus on figuring out a solution and continue to complete the remaining iterations and assess our timeline accordingly. Throughout our project we used this life-cycle to set time to learn advanced flask applications, database/hosting connections, and communication/timing. Learning a flask application for an enterprise-level project was difficult and it was a risk we had to consider. It took us quite some time to learn and understand how we can use a flask application with database connection, hosting connections, upload/downloading documents, create an interactive design, add filters, create a table, and many more challenges we had to solve along the way. Since we had strong team communication, hardworking members, and the lean agile life-cycle approach we completed all these functions on time.

Chapter II: Project Management

- Task analysis, Roles, wbs/gantt, risk/management

Risk management was one of the most crucial steps for this project and with a good methodology and communication, we broke down and classified what levels of risk were involved. The chart below represents two of the many risk factors we considered but these were the most important out of the rest. The group coordination was important to assess because if we did not have good communication as a team our project would have suffered in timing and quality tremendously. We put in a lot of time and effort throughout the semester to ensure our final product would come out as we expected and hopefully wow our audience and industry team.

Risk	Probability	Consequence	Product	Cause	Symptom
Group Coordination	.4	9	Understanding clients feedback to create final project	Other work/ classes, family matters, time conflicts.	Lack of communication and/or group participation in meetings
Flask Application	.5	8	Run into issues of working with flask application and costing more time	Learning curve, not knowing how to work application, not enough learning sources	Not having the knowledge to apply new applications, or a very complicated process to use.

For roles, we viewed each other as equals, and everyone did their part when it came to completing the project. Param was very helpful in the development of the actual flask application itself. Param did a lot of the flask to database functionality and created a lot of the back-end and front-end. Aaron developed processes such as the filter function in the application and helped develop the back-end applications. Ali was also helpful in creating the front-end and database relationship applications. I helped create the flask blueprint application and created the upload/download feature. In terms of communication, I organized most of the meetings and did a lot of the discussion when we met up with the industry team. Param was also very expressive in team meetings and pointed out a lot of technical issues and solutions to those issues. Ali was very helpful in identifying what issues we may come across for future parts of the project. Aaron helped discuss solutions to ideas we had in mind. Altogether we have an amazing team and were able to complete this project on time with no major issues.

For our task analysis, we used the lean agile life-cycle approach and used iterations and notes to set ourselves guidelines to complete the project on time. Each member would discuss what task they wanted to complete and I would help discuss with each member how, when, and exactly what they should be completing. We were pretty flexible with timing and used that to our advantage. We started the development process as soon as we got approval from the industry team and had milestones for the project. We impressed our industry team with how much work we were able to complete due to our team collaboration!

WBS/Gantt was something that I used as a reference all the time to make sure as a team we were on the right track to complete the project on time. I would list dates and estimates of how long we would have to complete a portion of the project. I made sure to give enough time to ourselves to create the flask application and that involved both a learning phase and a

development phase. The first two weeks were mainly setting up a prototype for the project and making sure we did not have to go back and fix any mistakes because of communication. The second phase was to give us time to create a plan to learn and develop the actual code itself. The third phase was to host and connect our project with the database and industry team hosting applications. With the agile lean life-cycle approach we were able to complete everything on time and the industry team and our team were very happy with the final result of the product!

PROJECT PLAN AND GANTT CHART TEMPLATE

PROJECT NAME	PROJECT MANAGER	START DATE	END DATE
Documents upload	Kerim Sever	23-May	1-Aug

OVERALL PROGRESS
100%

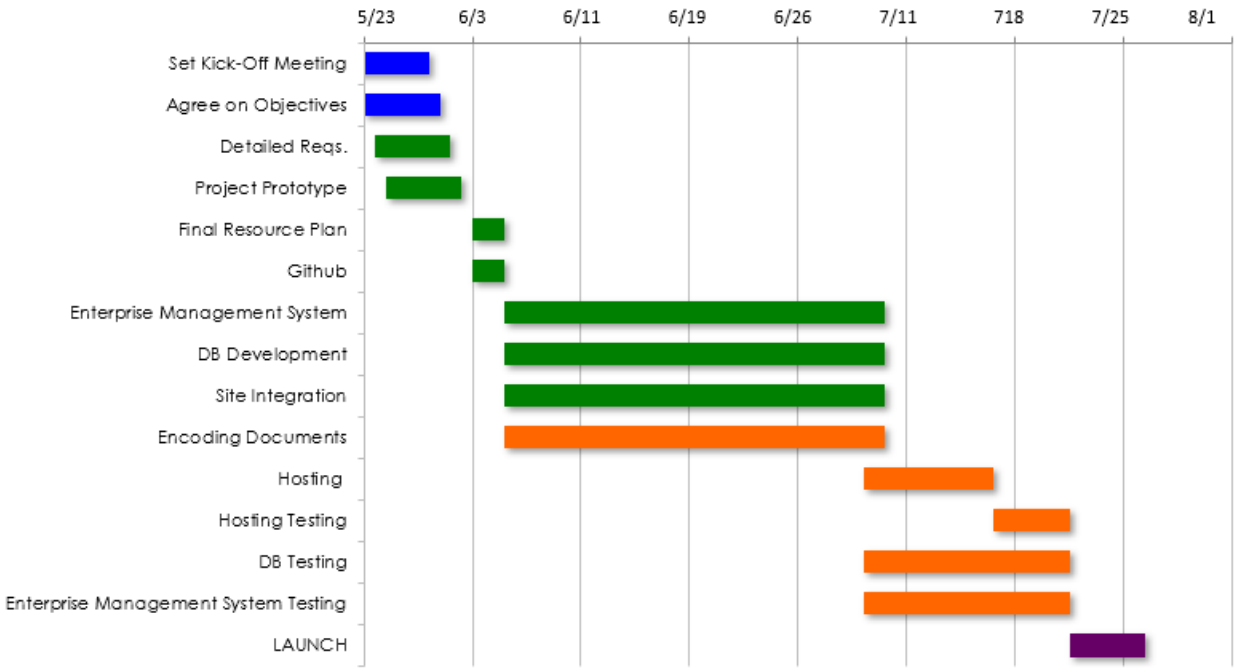
TASKS	RESPONSIBLE	START	END	DAYS	STATUS
Set Kick-Off Meeting	Alex P. and Kerim Sever	5/23	5/30	7	Complete
Agree on Objectives	Industry Team and NJIT Team	5/24	5/31	7	Complete
Detailed Reqs.	Kerim and Alex	5/25	6/1	7	Complete
Project Prototype	Both Teams	5/26	6/2	7	Complete
Final Resource Plan	Kerim Sever	6/3	6/6	3	Complete
Github	NJIT Team and Alex P.	6/3	6/6	3	Complete
Enterprise Management System	Aaron and NJIT Team	6/6	7/11	35	Complete
DB Development	NJIT Team	6/6	7/11	35	Complete
Site Integration	Alex P. And NJIT Team	6/6	7/11	35	Complete
Encoding Documents	Param and NJIT Team	6/6	7/11	35	Complete
Hosting	Alex P.	7/11	7/25	14	Complete
Hosting Testing	Ali and NJIT Team	7/18	7/25	7	Complete
DB Testing	Ali and NJIT Team	7/18	7/25	7	Complete
Enterprise Management System Testing	NJIT Team	7/18	7/25	7	Complete
LAUNCH		7/25	8/1	7	Complete

PROJECT DELIVERABLE

High Level Enterprise Management System for Digitual Documents Upload/Download

SCOPE STATEMENT

This SoW aims to align the Client's website presence with the retail sales growth and business objectives
PM will provide complete website discovery, comprehensive user experience, creative redesign and development of the Digitual Documents Upload/Download.
NJIT Team will work with WillLouden.com to expand existing website functionality to include:
- Enterprise Management System: Ability for visitors to purchase baby products directly on the website.

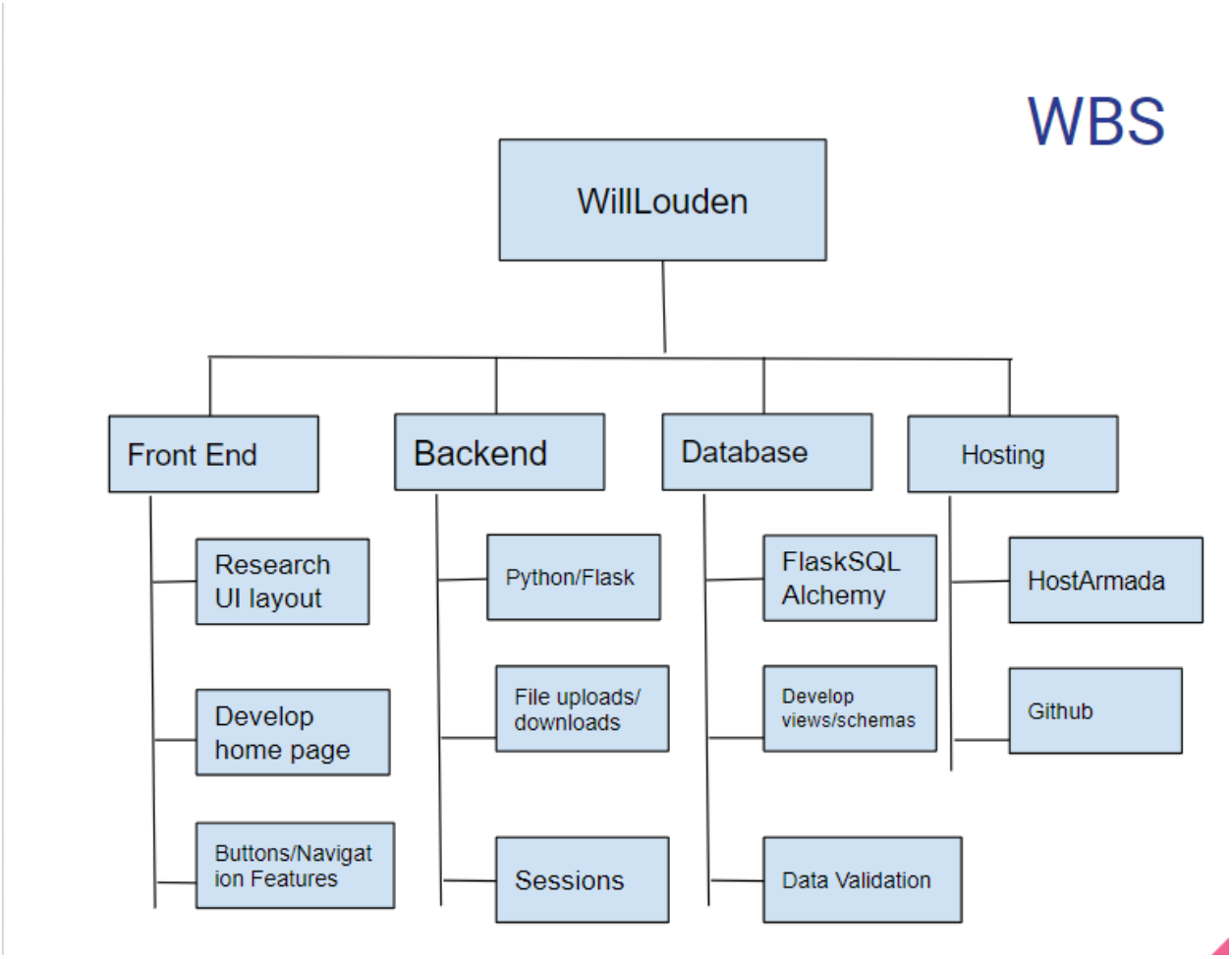


Chapter III: Define

Stakeholders are identified as everyone who has some type of involvement in the project or is affected by it in some way. The stakeholders, in this case, can be categorized into three groups. This first stakeholder group is the project sponsors. WillLouden representatives are looking for a team that can build an enterprise-level web application where they can digitally store their documents and allow the employees of their company to view/upload files into their database. The second stakeholder group is the project team members and our project manager. The project manager assigned for this project was Kerim Sever. He was mainly responsible for working directly with project sponsors to make sure that we are correctly building the required application. The other group team members were Ali Alkhateb, Param, and Aaron. Everyone was involved in the development phase of the project where we were building the web application making sure that it meets the standards that sponsors have laid out. The third group of stakeholders is the NJIT Capstone executive team and the class professor, Osama Eljaibiri. This group serves as the judge for our project where they mainly focus on making sure that we are building an application that the sponsor can use. Another possible group of stakeholders is employees at WillLouden that will potentially use the application to store digital documents in their database.

The first part of collecting requirements is correctly identifying the problem at hand. In the first few weeks of the project development, no actual development was made and we were in the phase of discovery and research, continuously collecting requirements. We conducted several team interviews with our sponsors to correctly identify the main problem that we are trying to solve. The main objective was to build a web application that allows all users to upload

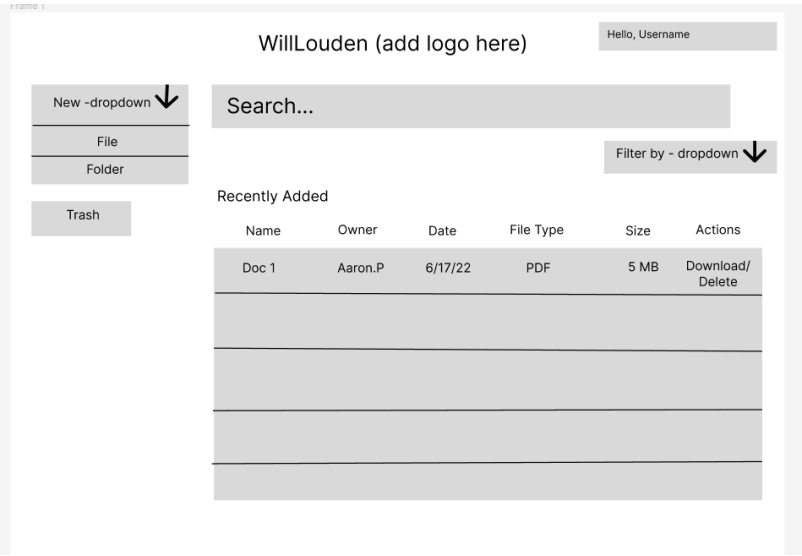
documents, search for documents, and send alerts to users using a type of web-based framework as the backend application. This would serve to be our main project scope. As part of defining the project scope, the deliverables will be a bug-free, fully functioning web application that allows users to upload files as different users with an intuitive user design. It will also store the uploaded files in a database for the persistence of user data. The deliverables will also include a full report on the entire project identifying all of the different aspects that went into the development of the project, explaining the implementation details and the main business aspect. The project scope exclusions are excluding the login feature in the application. This means that users will not be able to log in with their current email addresses but will rather have to use premade accounts that the system administrator will control. A future feature that can be implemented is OAuth. This is where the website will allow the user to log in with one click using their google email address. The constraints of the project are that we will approximately have 11 weeks to complete all of the phases of the project. The next step was to hold focus group meetings with our internal project team to brainstorm techniques to be able to break down the problem into actionable tasks that can be developed. This includes identifying the application frameworks to be used, the data requirements, database requirements, and all of the tools/frameworks to be used in our web application. The main goal of trying to identify all the requirements into actionable tasks is to develop a Work Breakdown Structure. Creating a WBS involves the decomposition of each actionable task into a smaller one, breaking it down into its smallest component.



As part of the verification process of the scope, we held weekly meetings with our sponsors showcasing the current features that we are working on. The sponsor would give their review and whether our project was meeting the requirements and it was what they had in mind. In this phase, the sponsors would give their recommendations and their requested changes. This would cycle back to our internal meetings where we would meet to plan the recommended corrective actions and split the work to complete newly updated requirements. The parts of the project that would change are the smaller implementation details such as the main layout, colors, button navigation and orientation, and the way the site is structured.

Chapter IV: Design

A simple yet understandable user interface is crucial for web development so that users may navigate the site with ease. Similarly, having a good layout is also crucial since it gives the website a nice, polished appearance. In order to start the design process, First, we created a design blueprint on Figma to use as a starting point for building the layout for DocuDrag. The layout of the new website would be discussed between our team and the project manager Kerim Sever. It was crucial to get the fundamental design perfect so that only the minimum of items would be displayed. We went with a very basic design that had a button for adding new files, a button for removing old files, a drop-down menu for filtering, a search bar, and a container for all the recent file uploads.



Initial Figma Layout

Alex was our project coordinator who worked at Willouden. We would hold weekly meetings with Alex and Will Louden, the company's CEO. During one of our first meetings, we

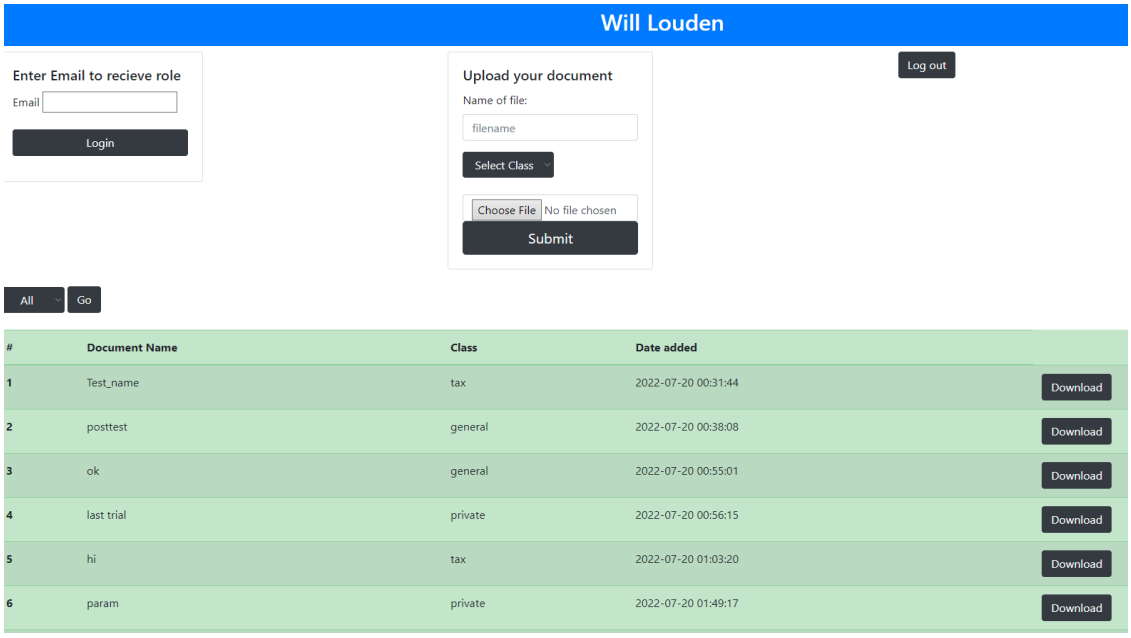
showed Alex and Will our Figma layout. They appeared to like the design we came up with because it was straightforward and user-friendly, which was the main goal when developing this website. Will did propose, however, that we alter our design and include additional elements to make the file uploading process more streamlined and user-friendly. Will recommended that we incorporate encoding. We would essentially need to add a class field and a role functionality as a result of this. By using each user's email address as a primary key, the role functionality essentially assigns a role to each user who accesses the website. Once a person logins into the website, they can upload files by various classes. Each user would have access to particular classes of files depending on their position within the firm.

We promptly began working on the front end and created an HTML page with our layout after discussing all the necessary adjustments that needed to be made to the initial layout. In the beginning, Ali Alkhateeb and I created a basic functional HTML page with all of the buttons and the dropdown menu using HTML and CSS. This gave us a good understanding of the layout of the website. This also allowed us to concentrate and spend more time on the Python application and database management system, which can be among the most challenging aspects of building a website. We, as a team, agreed upon using Flask to create the website. Since all of the team members had experience with Flask through previous projects, we chose to use it. Param Patel and I were given the assignment to construct the Flask application.

We continued trying to improve the Flask application's functionality. Meanwhile, our backend team, which was made up of Param Patel and Kerim Sever, did a fantastic job in quickly setting up the database. Our team used PhpMyAdmin, which was also provided by Will-Louden, to create the back-end database. The team used MySQL to store user information and build role tables for the database. The database connection for the Flask application was the responsibility

of the front-end team, which included Ali Alkhateeb and me. The database connection for our Flask application, which is occasionally the trickiest component of building a website, worked well for us. Login requests were successfully sent to the backend from the front end. This gave database users access to their accounts so they could upload files to the website that would be saved in the database.

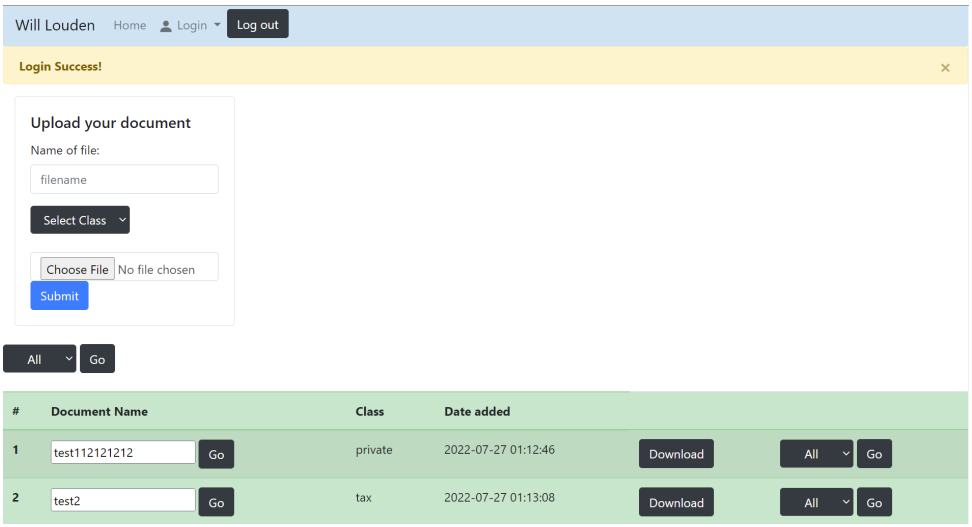
Soon after building a basic functional website with the necessary functionality, we decided to shift our focus to the front end. Param Patel and I, both felt that the web site's fundamental HTML and CSS appearance needed improvement. Hence, we decided to incorporate Bootstrap which would give the website a much nicer look. Since we were already introducing filters that would separate the files based on their classes, we also decided to remove the search bar, which was of no value by this point.



Main Landing Page

Soon after, we gave Alex and Will a walkthrough of the website and demonstrated all of its features during one of our weekly meetings with them. Additionally, we sought suggestions for improvements that would help Will Louden make the most of this website in the future. Will proposed making a few adjustments to the website that would give a specific user role more authority. We currently had three user roles in our database: manager, user, and CA. The Manager had the most authority out of the three roles because he could see files from all different classes. Will proposed that we give the manager authority to rename the files and change the class of files. This initially appeared to be a highly challenging task because our backend team would need to expand the database's functionality in order to alter the classes and file names. However, Param Patel did a fantastic job and devised a method to alter classes and file names. Given that time was running short at this point, this assignment seemed very challenging.

We were able to finish all of the work that our front end and back end teams were required to do by the end of the week. With the help of Bootstrap, we also added a navigation bar and flash messages to the website to enhance its visual appeal. This Capstone project taught me a lot, and I will surely use that knowledge in the future.



Final Landing Page

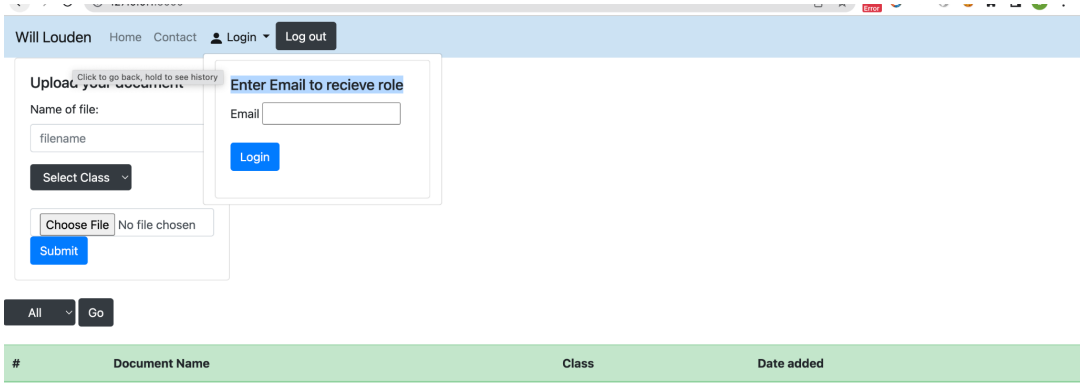
Chapter VI: Evaluation and Conclusion

Solution Testing: Collaboration is key to attaining our deadline. It was challenging to test the functionality and connectivity with the database while simultaneously constructing the backend and frontend. Several computer languages were used to develop the website. The website's front end is made up of HTML, bootstrap, and JavaScript, while the back end is made up of Python (flask) and MySql. While team members work on it individually, the website is tested on a local server. After checking that there were no problems, it was deployed to the hostarmada server. The website makes use of the PhpMyAdmin database to test the backend queries. The webpage is up to date on the server and accessible to users.

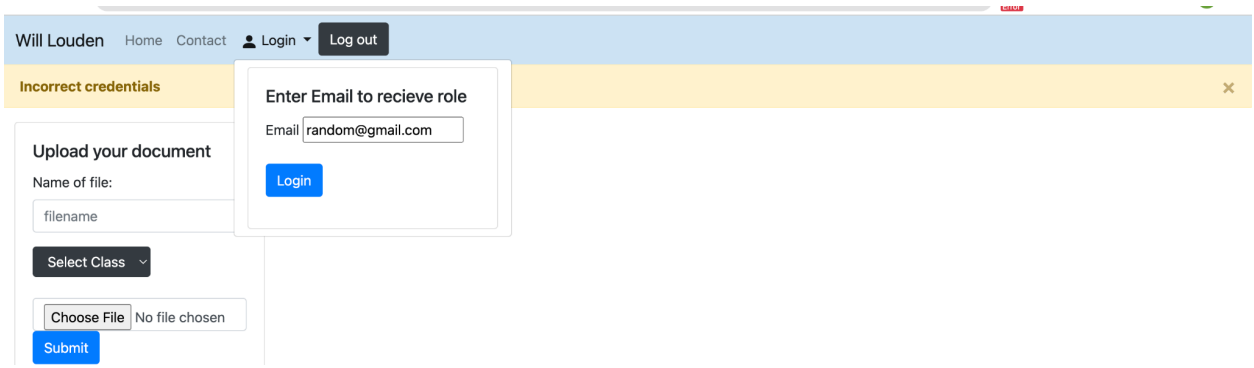
Verification: The project manager assigned specific duties to each team member to complete. After the task was accomplished, team members were instructed to publish the code to Github. After that, the project manager and sponsors acknowledged that it satisfied all of their criteria. Furthermore, the team evaluated the project with several web browsers such as Safari and Google Chrome to ensure that it is also accessible with other web browsers. As a result, the quality assurance testing was a success.

Validation:

Email login: Several methods are used to determine whether or not a user is registered and what role has been allocated to that email address. If the email address in the login field is left blank, the website will prevent the user from posting or viewing any papers marked "Enter Email to Receive Role."



If the email address supplied is not registered, the error "Incorrect credentials" will appear. After the user enters the right email address, the backend will assign the user, manager, or CA role depending on the database.



Team Conclusion: The ultimate goal of this program is to provide students with hands-on experience in the technology industry. This course taught us about the interview process, the importance of deadlines, teamwork, and engaging with project managers and sponsors, among other things. This schooling equipped us for real-world professions. This course helps students who lack internship or project experience to incorporate into their resumes. Project managers learn how to interact effectively with customers and bosses, as well as assign duties to team members as clearly as possible, so developers know what to expect.

The frontend team has learned how to create a visually pleasing website using HTML, JavaScript, and Bootstrap. The backend team learned flask and followed several tutorials to provide the groundwork for the website. The expertise gained via several hours of research and tutorial access to be educated in the areas that the website would require has been a tough, yet rewarding process. Team members learned the importance of deadlines and how to work together to find the best solution for the assigned task. We also looked into the development cycle. . It was difficult at first since we didn't know anything about each other, but as time passed, we came to comprehend each other's strengths and shortcomings. Perhaps displaying our previous work would have offered a greater picture of what we are best at, but in the end, we reached the best possible conclusion. This course's ultimate goal was not to produce a product. We also had to present and were evaluated on a number of things, such as our presentation skills, product knowledge, communication with team members and sponsors, and so on. These presentation skills will surely help everyone's profession.

WillLouden's main goal is to create a platform for businesses to store their data in the cloud. Our team successfully built a fully functional and user-friendly UI website with basic upload features, as desired by the sponsors. All of the sponsors checked and validated the website, and they all offered positive feedback.

Github Link:

https://github.com/bobcom2/DocWcom

- Branches
    - Main
    - Feat_front
    - Final-Changes
    - New-Front
    - Search_bar
    - Api_db
    - Final
    - Labels
    - Param

Website Link:

https://docwcom.gx850l.com