

Description of Scenario

My client is the psychology teacher in my school. For the subject Psychology, students often need to quote studies to support their claims in their exams or research paper. The categorization for the paper are very specific to IB as well, such as the nine designated topics, the three different approaches. There is also a very specific set of code studies that are considered valid for the examinations. Students can benefit greatly from a search engine that can filter the studies for them as they would not need to manually go through all the studies themselves to find the relevant one. The current search engine such as Google Scholar and Jstor are not suitable for the specific scenario as they contain a very large dataset, many of which are not suitable for the IB Psychology. They also do not have the ability to search for specific IB Psychology topics and approaches. They also do not give the teacher the control to guide students through tagging documents.

Rational

I judged that using a web app would be most suitable for the Scenario. More specifically, using Django + Python + Semantic UI as the development frameworks, SQL, AWS S3 as the database and storage, and deployed on Heroku. A web app is chosen over a conventional app for several reasons. First, the database will be updated periodically by the teacher, so it would be convenient if all the students and teacher share the same database in order to eliminate the duplication of files. Second, a web app would load faster as it does not require download from the user. Third, it works cross platform as it would only require a web browser from the user to run well, which exist on almost all platforms such as Mac, Windows, Ubuntu Desktop and smartphones and tablets. It significantly widens the compatibility of the program without requiring the programmer to make a program for every platform. While cross platforms tools such as electron does exist, those are still unnecessary for the purpose of this project as the project would not need to be run offline. In addition, the city of the client has well built internet infrastructure, so a slow or unavailable internet is not taken into account. Lastly, making this a web app significantly reduces the maintenance cost as an update can be pushed out instantly for everyone if there is a feature update.

The reason to choose Python and Django as the back end framework is predominately due to the convenience. Python is chosen over other programming language such as Java and C++ because of the faster development speed in trade of the slower runtime speed (Barot). The program does not need to have excellent performance as there would not be too much traffic. A web development framework would be very beneficial as that eliminates the need to write a web app from scratch. Plain HTML, CSS and Javascript (i.e. a static website) is not suitable for this project as the database will be dynamically built by the client. For the exact reason, Django is chosen over other Python web development frameworks such as Flask because it has a very well built interface for SQL databases that allow the user to interface with Python instead of SQL syntax (Singh).

I choose Semantic UI as the front end development framework because it speeds up the development time as well. Semantic UI is particularly useful as it has a very large library of beautiful elements and modules for developers to incorporate into their design.

Success Criteria

1. Functionality

1. Teacher can upload paper and add tags
2. The search function would return relevant result
3. The teacher can add students to add studies to the website's database
4. Students who are given the permission can upload paper and add tags
5. The user should be able to download the papers in the server.
6. The user should be able to see the thumbnail and successfully download the PDF.
7. The storage should be large enough to store all the PDFs in the database.

2. Security

1. Only editors can modify the database.
2. Only the admins can modify manage user permissions.
3. The site is encrypted in transit.
4. Plain text password is never stored in the database.

3. Accessibility

1. The UI would still work when it's zoomed in or viewed on a smaller device like a mobile screen.