Easy Attendance - Proxy

Group 9

CS14B013 - K. Vinod Reddy
CS14B024 - Hemanth Reddy
CS14B028 - Srinidhi Prabhu
CS14B041 - D. Sai Pavan
CS14B042 - G. Satish Kumar

What is Easy Attendance and Why is it required?

Easy Attendance is a face recognition tool intended for use in classrooms.

This software is intended for use by professors to simplify the process of taking attendance of a class and check for proxies. Though there are many applications that can perform face recognition and identification tasks, there are very few which are specifically used for taking attendance in universities or schools.

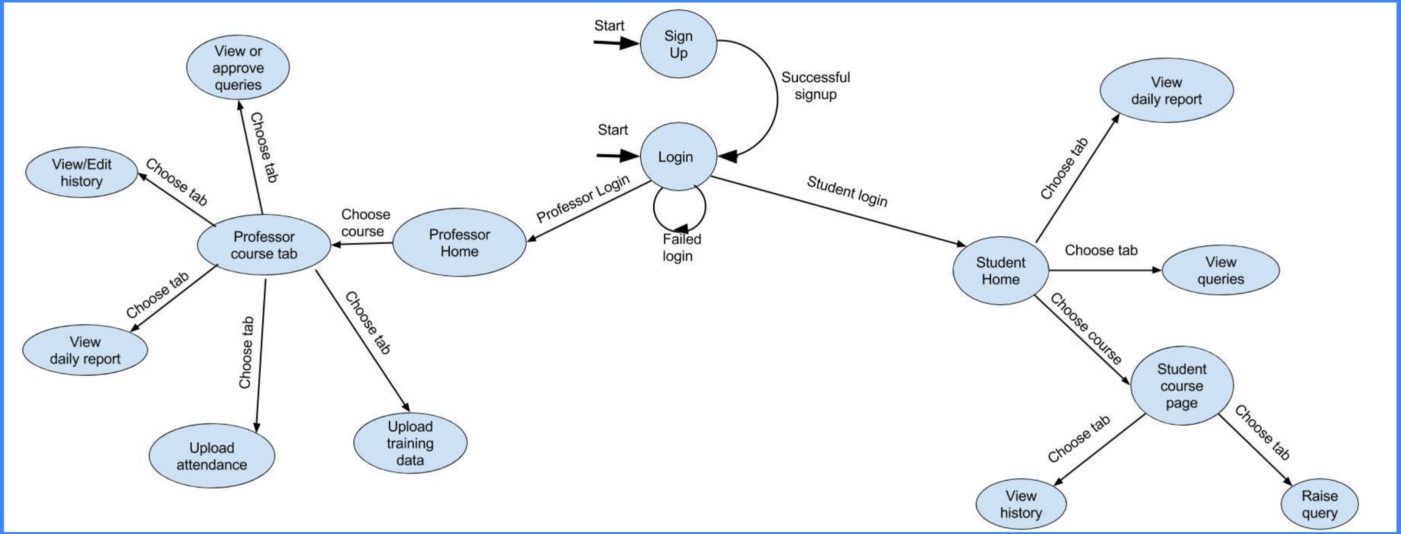
How is Attendance taken?

- The admin shall collect the pictures of individual students and use them for training.
- The underlying learning algorithm in the software gives the professor a daily attendance report once he uploads the photos of the class.
- The professor shall have an option to view history, i.e., pictures and attendance.
- Students can make a query in case their attendance was not marked, in which case the professor can override the system to approve that the student has indeed attended.

Functional Requirements

1. The software shall authenticate students and professors during sign-up and provide required access to each account.
2. In the case of professors, the account shall have the following options:
   - Upload images of the class for a particular day and get the attendance for that day.
   - Look up the images uploaded on a particular date and get details of the students who were present or absent.
   - A section for the queries raised by students, in case the student is present but marked absent. These queries shall be approved only by the professor.
3. In the case of students, the account shall have the following options:
   - Show the number of classes attended by the student and the total number of classes.
   - Warn the student, in case his attendance is less than the threshold.
   - Provide an option to raise a query, in case he is present in the photo but not marked present.

High level diagrammatic representation



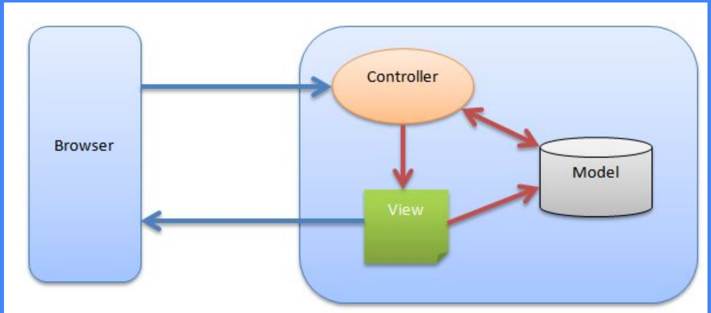
A detailed set of models can be found in the document.

System architecture

We use the model-view-controller(MVC) architecture.

- The user interacts with the HTML pages in a browser.
- Button clicks fire events, which are redirected to the corresponding URLs.
- After processing in the backend and making required changes in the databases, the new HTML page is displayed.

We have used Django framework for our project because it inherently uses the MVC pattern.



Implementation issues

Reuse:

- Use of Django - Django helps us in setting up the components, so that we can focus on the development of the project
- APIs for Face Recognition - Since the project was over a span of 3 months, coming up with our own algorithms for image recognition was difficult. So, we used Face++ APIs for Face Recognition

Configuration management:

- We used github for version control and system integration.

Host-target development:

- Since the project is a web application, the user only needs a good browser on his system.
- The HTML pages have been designed so that the layout is same across all devices.

Software Testing

We have done development testing for our software. The testing was done as follows:

- We used the Z specifications to come up with test cases.
- The test cases were written in a python file within the Django project.
- The tests were run to check if they matched the expected output.
- The test cases were written for each component(Professor module, student module and authentication module) separately.

Possible improvements in future

- The image recognition algorithm can be included with the code. This would reduce overheads of making API calls to Face++ and make the recognition faster.
- UI and UX can be improved.
  - We can support attendance to be shown for a period instead of just one day.
  - Hide queries that have been approved.
  - Pagination wherever long tables are displayed.
- Rather than the professor taking photos, we can install a camera in the class that recognizes the faces and automatically updates attendance.
- Admin GUI can be added.
- Course data, students and professors info can be directly obtained from workflow.
- LDAP / ADS can replace the existing auth module.