ONLINE MENTORING SYSTEM

abi1562004_4435

Aadhithyan D
Abinandhini D M
Logeshwar B S
Sona B
Vimal Dharan N

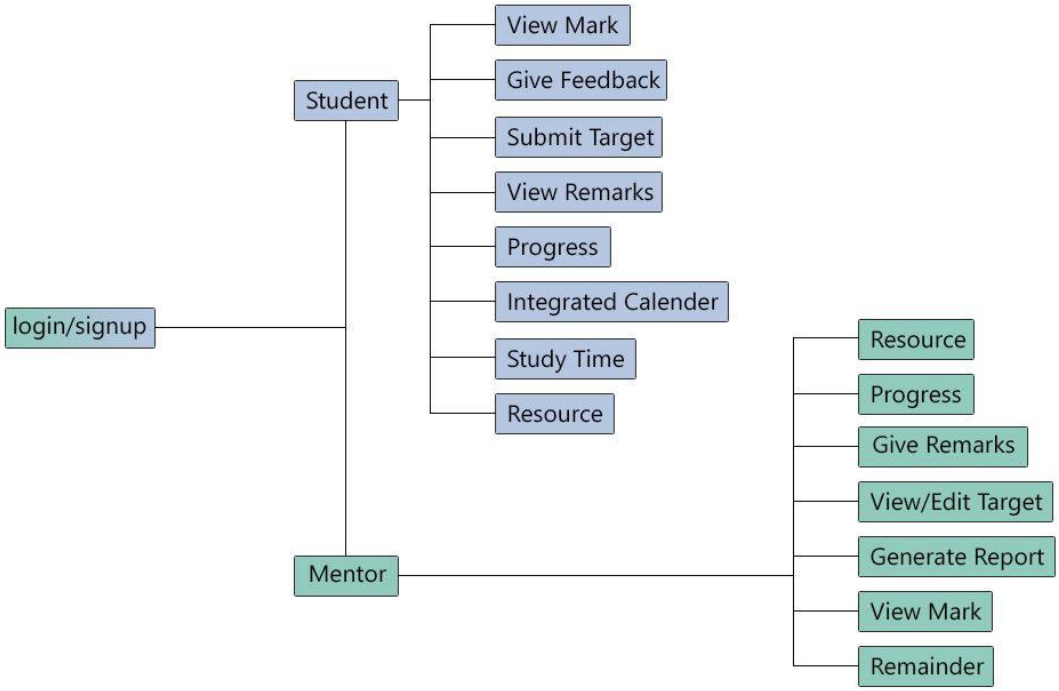
Source Code:

Abstract:
Students nowadays find it difficult to track their academics properly. Students are visiting websites for resources, tracking time, and reaching out for help online. there is no existing app or website which gives all the features.
This app gives access to both mentors and mentees where mentors can guide the students and give remarks and resources based on their performance. We've used PyQt5 for the front end, matplotlib for plotting the progress, and SQL for the back end. The information of both mentors and mentees will be stored in the database. The marks and progress will be stored in the xml file. Students can set targets and the student will be mentored to reach the target.
Mentors can remind the students to completion of tasks with an in-built calendar. Students can test their productivity by using a timer that tracks their study time. ML Algorithm is applied to track the progress of the students. Students can access resources that were being shared by mentors. ML code trains a simple linear regression model on the student's previous assignment grades and uses it to predict their grade on the next assignment. The predicted grade is then stored in the next assignment grade variable and could be displayed on the graph or in another part of the user interface.
The targeted audiences will be students and teachers

Features:

- Flexibility
- Accessibility
- Efficiency
- Increased reach
- Enhanced learning experience
- Cost effectiveness
- User friendly

Workflow:

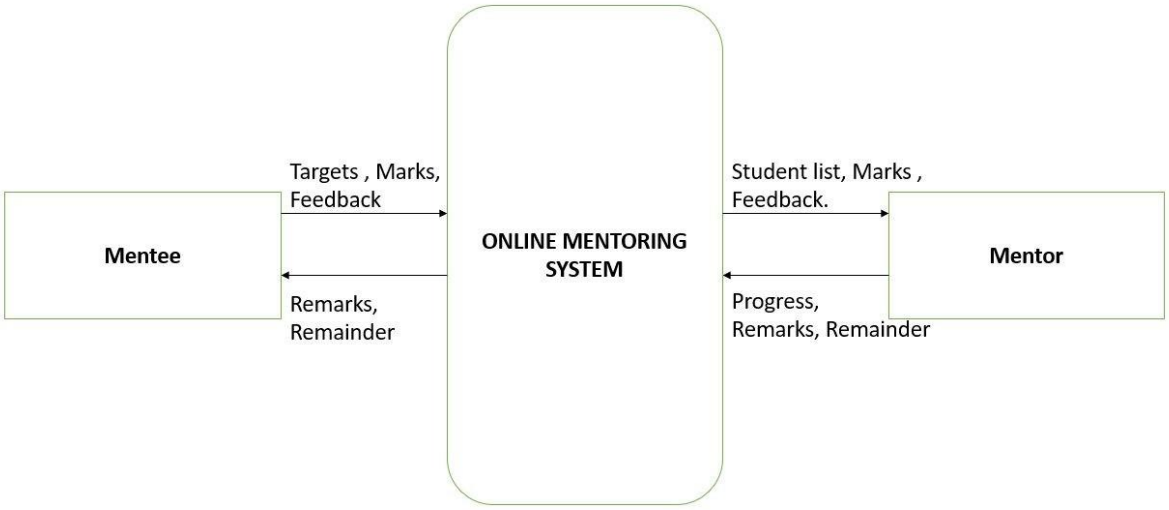


Technology Stack:

For frontend we've used PyQt5 which is advantageous over other software frameworks because it's cross-platform compatible, easy to use, integrates with Qt, enables rapid application development, and has comprehensive documentation and community support.

We've gone with SQLite3 for backend. Which is advantageous over other database management systems because it's lightweight, fast, and self-contained. It doesn't require a separate server, making it easy to use and deploy in a wide range of applications.

Architecture:



The system architecture contains an app-based Front-end for user interaction, a and back-end SQL database for managing data, and this system also contains some features like giving remarks and sharing feedback.