Criterion E

Success Criteria	Fulfilled?	Evaluation
Different signup pages for students and admins, that prevent improper sign-ins, and protect user information.	Yes	It has an intuitive GUI and layout, and performs function well, especially in ensuring password security for students. The font size could have been larger, and the space of the screen could have been used better for student sign-up.
Different log-in pages for students and admins that lead to different windows	Yes	It has a simple design that achieves the main goal, however, but navigation between different types of sign-ins was confusing.
The students will have task management system on the website where events (anything ranging from meetings, exams, homework, classes, etc.) can be added, deleted, or edited.	Yes	The add, delete, and edit buttons worked very well, allowing the product to be used dynamically. The choice of database allows the app to adapt and run on various devices.
A display of all the tasks entered by the user, ordered in the form of a schedule that places tasks in the order that is most helpful to the student	No	Tasks were arranged according to the due date and priority well but what would have enhanced the product was also scheduling by the time taken to perform each task.
A page that makes use of the deadlines and the event names inputted by the user to display these events on a calendar based on their deadlines, or submission dates.	Yes	A dynamic Calendar was created that efficiently displayed the task in a calendar for the user, for visualization of the tasks. Color coding of the task was a good extra step in helping students organize the different types of tasks they have.
A notes space beside the calendar where the user can write any general anecdotes	Yes	The simple note space had the essential qualities of the previous platforms, and it's

that they have about the week.		longevity due to the saving function made note-writing efficient. However, if color and fonts were also added, it would have benefited the students more.
A feature that allows students to receive reminder emails when they click a button to remind themselves to finish a task later.	Yes	Efficient implementation and method to remind students after closing the app of what needs to be done. HTML email could have been beautified to appeal to the students more.
An admin window that displays all the students and admins and allows the admins to remove students if they leave the group or remove illegitimate admins.	Yes	Implemented in a simple manner that satisfies the basic functionality.
The admin window will also allow admins to access each student's Calendars and give them the ability to set reminders for the students.	Yes	Implemented well, allowed admins to see the tasks and send reminders with ease. If admins don't have time, an option to send an automatic email 24 hours before the task deadline for the student would have been helpful.

Recommendations

Use of Machine Learning to create the optimal schedule.

The use of a machine learning technique like neural networks trained different types of tasks and the completion of which tasks provide the most productivity, as per the client's wishes, could be implemented. Over time, the NN model will learn the patterns of the student and know what types of tasks need to be completed first for the student to ensure that they complete all tasks in time. This information would allow for the creation of a schedule that truly benefits the students and pushes them to optimum productivity. The NN would function using TensorFlow which would allow the ML model to be coded in JavaScript itself.

Creation of a catalog for completed tasks.

Another method of motivating students would be to create a catalog on each student's app that displays the five latest tasks completed by the user. Another option for completed tasks can be

created in the task manager and when the completed button is clicked, the task is archived in the database and a log can be created with these tasks. This log would feature the students who most recently completed their tasks, and could be monitored by the admin to ensure no malpractice is allowed. A leaderboard that holds the student with the most finished tasks could also be created. 2D arrays and Linked Lists can be used to store completed tasks and arrange the students in the correct order for the leaderboard. This would provide the students a sense of community while using the app as well and drive to be placed on the leaderboard.

Word Count: 253 words