1. Professors using the website to input grades.

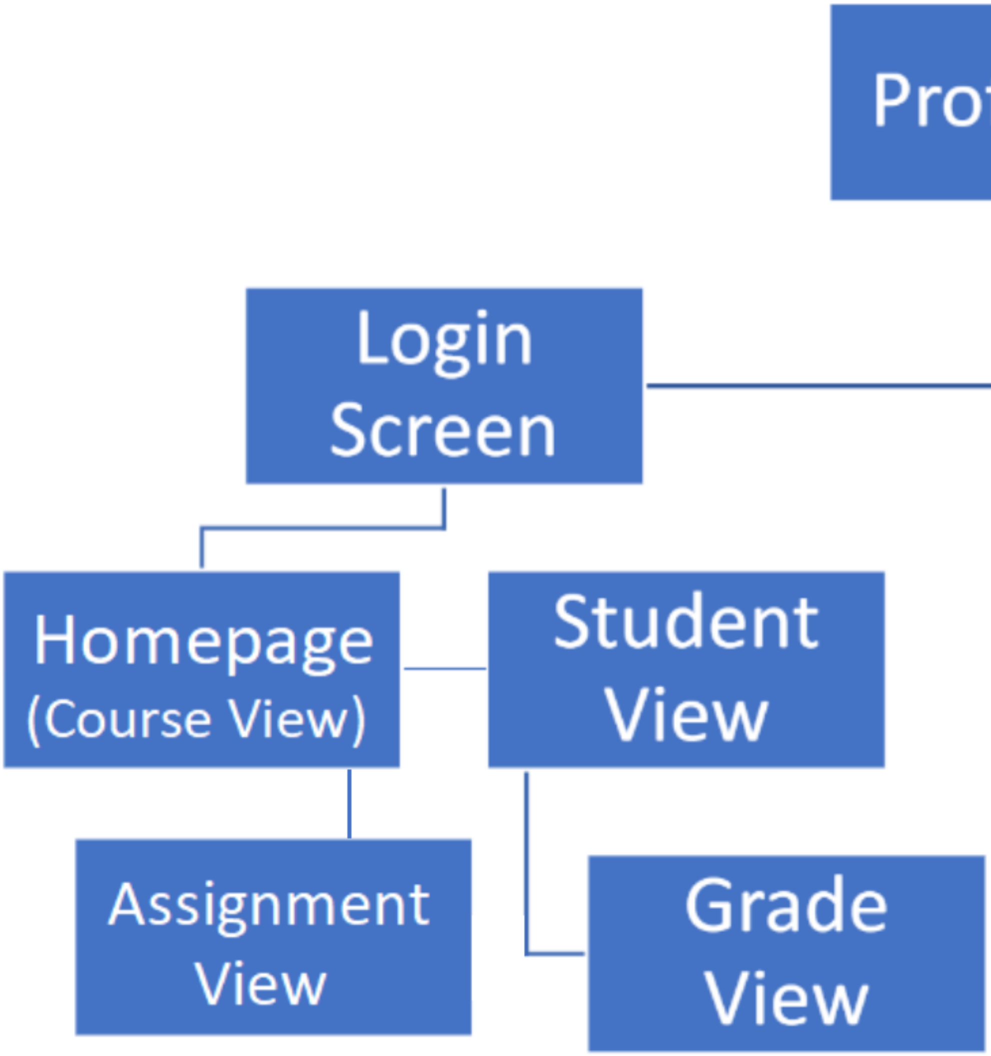
Introduction:

Professors will be utilizing this gradebook web app to have an overview of their courses, the students within, and the students' grades.

Overview:

The purpose of this web app is to allow professors to more easily and efficiently input and keep track of students' grades. Professors will be met with a login screen, like the students, and after authenticating will have several relevant views. The side navigation bar and homepage will allow the professor to see a list of courses of which they instruct and manage. After clicking on a course, they'll be taken to the student list where they can then input and manage grades for new or existing assignments.

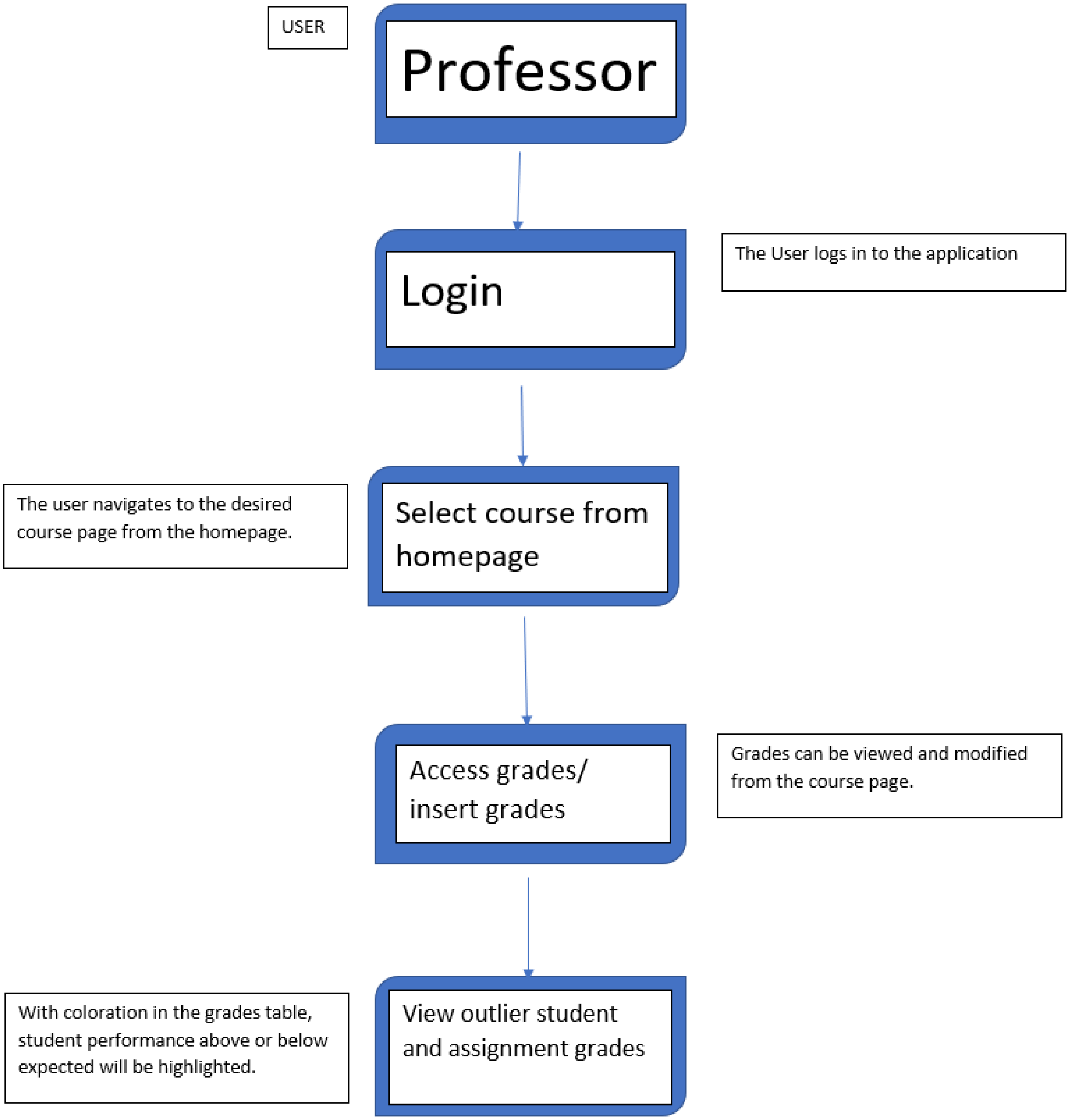
Diagram:



2. Professors identifying outliers

Once on the program, the user can go into their grades to check for a possible outlier to their expected grades and keep a closer track of them to see if there is any way that they can help improve the class, and/or those struggling students' performances.

The User will first have to log on to the program. They then will simply need to go to the grades and input any grades that they have not put in yet with an expected average for the assignment. The program will then flag assignments that are problematic according to that expected average and the average of the students and the class. The User then can filter the problematic assignments and see what exactly people are struggling with so that they can work on planning to try and help the difficulties the class is facing.


USER
Professor
Login
Select course from homepage
Access grades/ insert grades
View outlier student and assignment grades

The User logs in to the application

The user navigates to the desired course page from the homepage.

Grades can be viewed and modified from the course page.

With coloration in the grades table, student performance above or below expected will be highlighted.

3. Professor looking at single student performance view.

- Primary Actor: Professor
- Scope: The gradebook
- Level: User goal
- Brief: The professor logs in, selects a class, and views a student’s profile.
- Stakeholders:
    - Professor
    - Student
    - Gradebook management
- Postconditions: Student’s grades were accessed.
- Preconditions:
    - Professor has an account.
    - Professor has an active class.
    - Student is in at least one of the professor’s classes.
- Triggers:
    - Student’s grades are requested.
    - Professor needs to view the student's missing assignments, attendance history, or statistics.
- Basic flow:
    - 1. Log in
    - 2. Select student’s class
    - 3. Select student’s profile
- Extensions: No available selections would modify the flow or outcome.

4. **Professors and students sharing feedback with each other.**

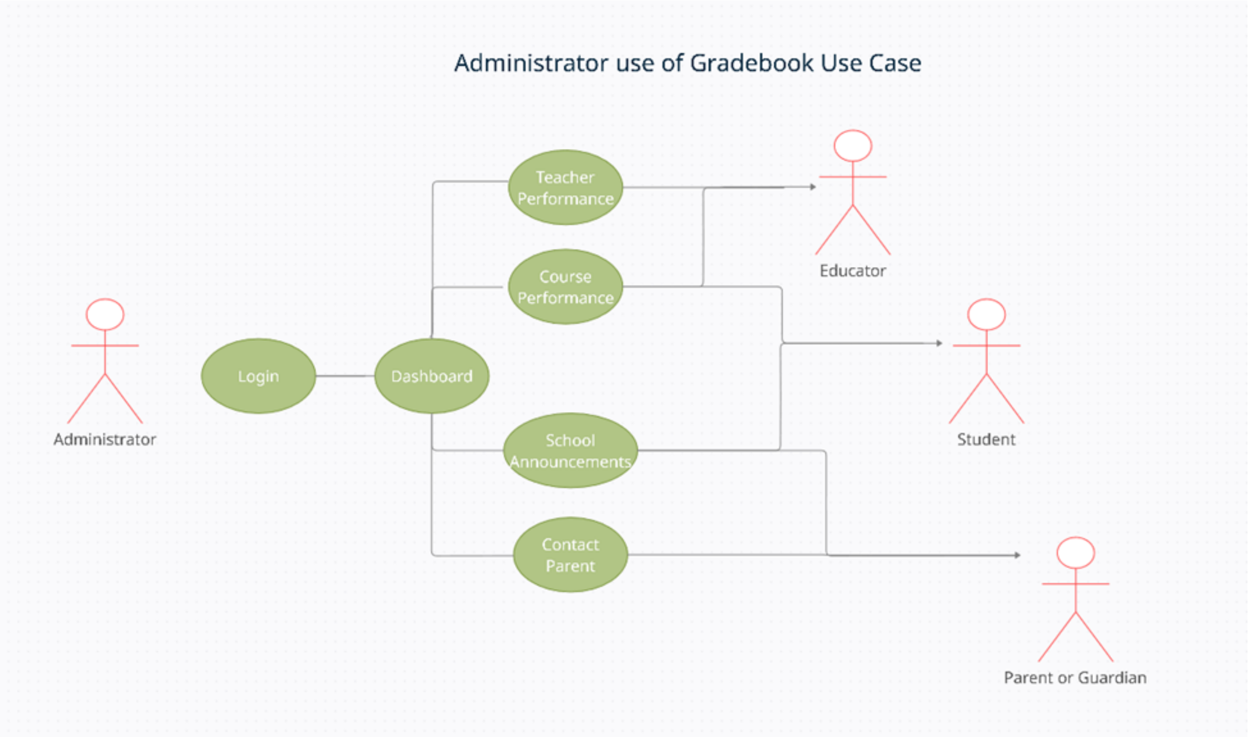
- Primary Actor: Professor, Student
- Scope: The gradebook
- Level: User goal
- Brief:
    - Professor:
        - After logging in, the professor will see any student audits on the homepage. The professor can then navigate to the course and the assignment. While grading, a message can be left for the student.
    - Student:
        - After logging in, the student will navigate to the course. On this page, professor comments are shown. The audit checkbox allows the student to dispute a grade and send the professor a notification.
- Stakeholders:
    - Professor
    - Student
    - Administration
    - Gradebook management
- Postconditions:
    - Student received a comment on a grade.
    - Teacher received a notification about a possibly incorrect grade.
- Preconditions:
    - Professor has an account.
    - Professor has an active class.
    - Student has an account.
    - Student is in at least one of the professor’s classes.
- Triggers:
    - A professor wants to leave a comment regarding why a student got a certain grade. (i.e. rubric notes)
    - A student finds a grade that he/she believes to be incorrect.
- Basic Flow:
    - Student:
        - 1. Log in
        - 2. Select course
        - 3. Check professor messages on graded assignments
        - 4. Click “Flag for Audit” to send professor a notification
    - Professor:
        - 1. Log in
        - 2. View audit notifications
        - 3. Select course
        - 4. Select student
        - 5. Write a message while adding/updating assignment grade
- Extensions: A professor leaving a comment does not require a student to respond, and a student flagging a grade does not require a teacher to change it.

5. Students checking grades and assignments.

Students will be able to log into the website anytime using their school email address and password. This will load a dashboard view, listing all of the courses that the student is currently taking, as well as their most recently graded assignments. It will also show any assignments that are due soon or overdue, and well as the course score to-date.

If the student clicks on one of the course names, they will get a list of all assignments in that class, both graded and ungraded, along with more detailed information. They will also be able to see notes here that were left by the teacher. There will be a button to dispute a specific grade if they believe it to be incorrect, which flags it for teacher review.

6. Administrators monitoring professor performance.



In this use case, the Administrator will have various functions within the Gradebook Application. Administrators will first log into the Gradebook Application using a username and password. When login has been approved, the administrator will have access to a dashboard. This can show brief, “widget-like” information relating towards general school function.

From there, an administrator can go to a Teacher Performance tab. They will be able to select an Educator from their list of Faculty and pull up general information relating towards their performance. This could include things like list of classes teaching, individual class performance statistics, display of course schedule, and other information relating directly to the Educator.

The Administrator can then go to a Course Performance Tab. There, the administrator can view individual course performance statistics. One topic of discussion that would be important is if Administrators should be allowed to see individual student performance, but for now I think that having just statistics with no names attached might be okay. The Administrator can view information like classroom outliers, assignments with data on number of turned in assignments and number of graded assignments, etc.

Two other ideas I had for this use case involved the use of sending notifications or emails to various individuals. Administrators should be allowed to push school announcements to the Educators, Students, and Parents/Guardians. In addition, if an administrator needs to contact an individual parent for follow-up to meet, there should be a feature to do so.

Overall, I think that the most important aspects of the Administrator use case are the ability to review teacher performance as well as individual class performance.

7. Administrators managing user accounts.

- Primary Actor: Administrator
- Scope: The gradebook
- Level: User level
- Brief: The administrator logs in, navigates to the accounts page, and creates or updates an account.
- Stakeholders:
    - Administration
    - Professor
    - Student
    - Gradebook management
- Postconditions:
    - The Administrator has created an account or updated a user's account information.
- Preconditions:
    - Administrator has an account
- Triggers:
    - New account needs created.
    - Account information (email or password) needs to be changed.
- Basic Flow:
    - 1. Log in
    - 2. View accounts page
    - 3. Select create account or edit account
    - 4. Enter necessary information into fields
    - 5. Save changes
- Extensions: The course of action only changes for creation or modification after navigating to the accounts page.

8. **Administrators managing courses and registrations.**

- Primary Actor: Administrator
- Scope: The gradebook
- Level: User level
- Brief: Administrator
- Stakeholders:
    - Administration
    - Professor
    - Student
    - Gradebook management
- Postconditions:
- Preconditions:
    - Administrator has an account
    - There is at least one professor
    - There is at least one student
- Triggers:
    - A course or section needs created
    - A course or section needs modified
    - A professor needs assigned to a section
    - A student needs added to a section
- Basic Flow:
    - 1. Log in
    - 2. Navigate to the courses page
    - 3. Select create button or edit icon
    - 4. Enter necessary information into fields
    - 5. Save changes
- Extensions: The course of action is similar for both creating and modifying courses, as well as adding both professors and students to a course.