

# Software Requirement Specification (SRS)

for Evalio: A Note-Taking Platform for mentors

---

## Outline

<u>Overview</u>	2
<u>Development Team</u>	2
<u>Our Sponsor</u>	2
<u>Definitions</u>	2
<u>Introduction</u>	3
<u>Purpose</u>	3
<u>Users</u>	3
<u>Functional Requirements</u>	3
<u>Use Cases</u>	3
<u>Use Case Details</u>	5
<u>Functional Requirements in terms of Features / Functionality</u>	15
<u>Non-Functional Requirements</u>	16
<u>User Interface</u>	16
<u>Key/Common UI Elements</u>	16
<u>Search, Filter, and Sort</u>	17
<u>Other Key Requirements</u>	17
<u>Performance Requirements</u>	17
<u>Technical Constraints</u>	18
<u>Design Constraints</u>	18
<u>External Interface</u>	18
<u>Security / Privacy Requirements</u>	19

## Overview

This document provides a comprehensive Software Requirements Specification (SRS) for Evalio: A Note-Taking Application for Mentors, a web-based platform designed to help mentors efficiently manage and organize their notes, tasks, and interactions with students. The document outlines the system's purpose, functional, non-functional, and performance requirements, constraints, and design considerations.

## Development Team

Ananya Sachdev	( <a href="mailto:ananya22069@iiitd.ac.in">ananya22069@iiitd.ac.in</a> )
Ananya Garg	( <a href="mailto:ananya22068@iiitd.ac.in">ananya22068@iiitd.ac.in</a> )
Nishchaya Roy	( <a href="mailto:nischaya22333@iiitd.ac.in">nischaya22333@iiitd.ac.in</a> )

## Our Sponsor

TalentSprint by NSE(National Stock Exchange), an EdTech firm headquartered in Hyderabad. It organizes coding boot camps and executive programs, and its hybrid digital learning platforms use onsite and online learning experiences. It partners with academic institutions and corporations to develop and deliver certificate programs in Artificial Intelligence & Machine Learning, Blockchain, FinTech, and Emerging Tech.

Email: [we.support@talentsprint.com](mailto:we.support@talentsprint.com)

## Definitions

**Mentor:** A system user responsible for guiding, tracking, and managing student progress. They are also responsible for taking sessions for students.

**Session:** A session refers to a scheduled or recorded interaction between a mentor and a student or group of students. Sessions can include one-on-one meetings, group discussions, office hours, or mentoring workshops.

**Cohort:** A group of students enrolled in a specific program, course, or mentorship period. Cohorts help mentors organize students based on shared attributes such as start date, academic year, or program.

## Introduction

### Purpose

The primary purpose of this project is to build a secure and interactive platform for evaluators and mentors of TalentSprint to make and manage notes, feedback, and student reviews. The platform would also have dedicated pages for all students, mentors, cohorts, and courses. Additionally, there would be a side panel containing a calendar, a to-do list, and an upcoming meetings section. The platform would also incorporate analytical data and insights that are accessible to users.

### Users

The primary users of this web application would be the mentors and instructors organizing the sessions and learning programs at TalentSprint.

## Functional Requirements

### Use Cases

Use Case	Description	Priority	Precondition
For Mentor			
Sign Up	To sign up on the platform	High	None
Log In (essential)	To log in to the platform	High	Signed up
Search	Search for a mentor, student, or cohort to view their stats	High	Logged In
Home/Logo button	To go back to the home page	High	Logged In
Sort Notes	Sort the notes by date, student name, cohort	Medium	Logged In

Filter Notes	Filter the notes by cohort, visibility, tags, pinned/unpinned	Medium	Logged In
View Notes Analytics	View analytics like number of notes taken, category used most, etc.	Low	Logged In
Add new note	To add a new note	High	Logged In
Select a tag for the note	Select a tag for a note to categorize it as progress, behavioral, academic, or other	Medium	Logged In, New note added
Add note content	To write the content of a note	High	Logged In, New note added
Flag note	To flag the note as public or private	Medium	Logged In, New note is added
Assign note	To assign the note to a student, cohort, junior mentor	Medium	Logged In, New note is added
Give access to people for a note	To give access (view, edit, comment) to people for a note	Medium	Logged In, New note is added
Edit an existing note	To edit a previously existing note, its visibility, content, etc.	High	Logged In, Note Exists
Delete an existing note	To delete a previously existing note	High	Logged In, Note Exists
View Meetings	To view scheduled meetings and events	High	Logged In
View to-do list	To view the tasks scheduled in the to-do list	High	Logged In

Add a task to the to-do list	To add a task to the to-do list	High	Logged In
Delete a task from the to-do list	To delete a task in the to-do list	High	Logged In, Task Exists
View profile	To view and edit profile details	Medium	Logged In
View Sessions Taken	View the session a mentor has previously taken	Medium	Logged in, Viewing profile
View Session Stats	View the attendance for a session taken by the mentor	Low	Logged in, Viewing profile, Viewing Sessions
View Student Page	To view a student's details and notes assigned to that student	Medium	Logged In, Search
Comment on a Student's note	To add a comment to a public note assigned to a student	Medium	Logged In, Viewing Student Page
Add a note to a Student	To assign a new note to a student	High	Logged In, Viewing Student Page
View Mentor Page	To view some other mentor's details and notes given by that mentor accessible to the mentor who is viewing	Medium	Logged In, Search
Comment on a mentor's note	To add a comment to a note given by mentor that is public or with comment access	Medium	Logged In, Viewing Mentor page
View Cohort Page	To view a cohort	Low	Logged In, Search

	page and its notes		
Add a note to Cohort	To assign a new note to the cohort	Low	Logged In, Search, Viewing Cohort Page
Log Out	Log out of account	Low	Logged In
For Admin			
Approve an account	To verify and approve when a mentor signs up	High	None
Delete an account	To delete an existing mentor's account	High	The mentor account to be deleted exists

## Use Case Details

### 1. Sign Up

**1.1 Actor:** User

**1.2 Purpose:** To sign up on the platform.

**1.3 Precondition:** None.

**1.4 Event Flow:**

1.4.1. User navigates to the platform's sign-up page.

1.4.2. User provides necessary information (e.g., email, password, name).

1.4.3. User submits the sign-up form.

1.4.4. The system validates the provided information.

1.4.5. If valid, the system creates a new account and sends a confirmation email (if applicable).

1.4.6. The user is redirected to the login page or automatically logged in based on the platform's behavior.

**1.5 Special/Exceptional Requirements:** None.

### 2. Log In

**2.1 Actor:** Mentor

**2.2 Purpose:** To log in to the platform.

**2.3 Event Flow:**

2.3.1. Mentor navigates to the main landing page.

2.3.2. Mentor clicks on the login button.

2.3.3. Mentor is redirected to the Google OAuth login page.

2.3.4. After successful login, the user is redirected back to the home page, and the login button is replaced with a logout button.

**2.4 Special/Exceptional Requirements:** None (Google API handles incorrect logins).

### 3. Search

**3.1 Actor:** Mentor

**3.2 Purpose:** To search for a mentor, student, or cohort to view their stats.

**3.3 Precondition:** Logged in.

**3.4 Event Flow:**

3.4.1. Mentor clicks the search bar.

3.4.2. Mentor enters search criteria (mentor, student, or cohort name).

3.4.3. The system displays search results based on the entered criteria.

3.4.4. Mentor selects an item from the search results to view detailed information.

**3.5 Special/Exceptional Requirements:** None.

### 4. Home/Logo Button

**4.1 Actor:** Mentor

**4.2 Purpose:** To go back to the home page.

**4.3 Precondition:** Logged in.

**4.4 Event Flow:**

4.4.1. Mentor clicks the home/logo button in the navigation bar.

4.4.2. Mentor is redirected to the home page.

**4.5 Special/Exceptional Requirements:** None.

### 5. Sort Notes

**5.1 Actor:** Mentor

**5.2 Purpose:** To sort notes by date, student name, or cohort.

**5.3 Precondition:** The user is logged in.

**5.4 Event Flow:**

5.4.1 The user navigates to the notes section and clicks the sort icon.

5.4.2 User selects a sorting criterion.

5.4.3 The system sorts the notes accordingly.

**5.5 Special/Exceptional Requirements:** None.

## **6. Filter Notes**

**6.1 Actor:** Mentor

**6.2 Purpose:** Filter notes by cohort, visibility, tags, or pinned/unpinned status.

**6.3 Precondition:** The user is logged in.

**6.4 Event Flow:**

6.4.1 The user navigates to the notes section and clicks on the filter icon.

6.4.2 User applies filter criteria.

6.4.3 The system displays filtered results.

**6.5 Special/Exceptional Requirements:** None.

## **7. Add New Note**

**7.1 Actor:** Mentor

**7.2 Purpose:** To add a new note.

**7.3 Precondition:** Logged in.

**7.4 Event Flow:**

7.4.1. Mentor clicks the "Add Note" button.

7.4.2. A new dialog box opens.

7.4.3. Mentor selects a tag for the note (e.g., progress, behavioral).

7.4.4. Mentor writes the content of the note.

7.4.5. Mentor chooses the visibility (public/private) for the note.

7.4.6. Mentor assigns the note to a student or cohort (optional).

7.4.7. Mentor clicks "Save" to save the new note.

**7.5 Special/Exceptional Requirements:** None.

## **8. View Notes Analytics**

**8.1 Actor:** Mentor

**8.2 Purpose:** To view analytics like the number of notes taken, categories used most, etc.

**8.3 Precondition:** Logged in, User is on the home page.

**8.4 Event Flow:**



8.4.1. Mentor clicks on the "View Analytics" button.

8.4.2. The system displays an overview of the mentor's notes and usage statistics.

8.4.3. Mentor can filter the analytics by different categories (e.g., date range, note type).

**8.5 Special/Exceptional Requirements:** None.

## **9. Edit an Existing Note**

**9.1 Actor:** Mentor

**9.2 Purpose:** To modify an existing note.

**9.3 Precondition:** User is logged in, and a note exists.

**9.4 Event Flow:**

9.4.1 User navigates to the note they wish to edit.

9.4.2 User modifies the content, tag, or visibility.

9.4.3 User clicks "Save Changes."

**9.5 Special/Exceptional Requirements:** The note must exist.

## **10. Delete an Existing Note**

**10.1 Actor:** Mentor

**10.2 Purpose:** To delete a previously existing note.

**10.3 Event Flow:**

10.3.1. Mentor logs in and opens the note to be deleted.

10.3.2. Mentor clicks "Delete."

10.3.3. A confirmation prompt appears, and Mentor confirms the deletion.

10.3.4. The note is deleted from the system.

**10.5 Special/Exceptional Requirements:** The note must exist.

## **11. View Meetings**

**11.1 Actor:** Mentor

**11.2 Purpose:** To view scheduled meetings and events.

**11.3 Event Flow:**

11.3.1. Mentor logs in and navigates to the "Meetings" section.

11.3.2. Mentor views the list of upcoming meetings and events.

## **12. View To-Do List**

**12.1 Actor:** Mentor

**12.2 Purpose:** To view the tasks scheduled in the to-do list.

**12.3 Event Flow:**

12.3.1. Mentor logs in and clicks on the "To-Do List" section.

12.3.2. Mentor views a list of scheduled tasks.

## **13. Add a Task in To-Do List**

**13.1 Actor:** Mentor

**13.2 Purpose:** To add a task to the to-do list.

**13.3 Precondition:** Logged in.

**13.4 Event Flow:**

13.4.1. Mentor clicks on the "To-Do List" section.

13.4.2. Mentor clicks on the "Add Task" button.

13.4.3. Mentor enters the task description and due date.

13.4.4. Mentor clicks "Save" to add the task to the to-do list.

**13.5 Special/Exceptional Requirements:** None.

## **14. Delete a Task in To-Do List**

**14.1 Actor:** Mentor

**14.2 Purpose:** To delete a task in the to-do list.

**14.3 Event Flow:**

14.3.1. Mentor logs in and opens the "To-Do List."

14.3.2. Mentor selects the task to be deleted.

14.3.3. Mentor clicks "Delete," and the task is removed.

## **15. View Profile**

**15.1 Actor:** Mentor

**15.2 Purpose:** To view and edit their profile details.

**15.3 Precondition:** Logged in.

**15.4 Event Flow:**

15.4.1. Mentor clicks on the profile icon in the top navigation.

15.4.2. Mentor is directed to their profile page.

15.4.3. Mentor can view and edit personal information (e.g., contact, biography).

15.4.4. Mentor saves any changes.

**15.5 Special/Exceptional Requirements:** None.

## **16. View Sessions Taken**

**16.1 Actor:** Mentor

**16.2 Purpose:** To view the sessions taken by the mentor.

**16.3 Precondition:** Logged in, profile viewed.

**16.4 Event Flow:**

16.4.1. Mentor clicks on the "Sessions" tab in their profile.

16.4.2. The system displays a list of all sessions the mentor has taken, including details like date and students involved.

**16.5 Special/Exceptional Requirements:** None.

## **17. View Session Stats**

**17.1 Actor:** Mentor

**17.2 Purpose:** To view the attendance for a session taken by the mentor.

**17.3 Precondition:** Logged in, View profile, View Sessions.

**17.4 Event Flow:**

17.4.1. Mentor navigates to the "Sessions" tab in their profile.

17.4.2. Mentor selects a specific session from the list.

17.4.3. The system displays attendance details, including student names and their attendance status.

17.4.4. Mentor can review the session attendance and make notes if needed.

**17.5 Special/Exceptional Requirements:** None.

## **18. View Student Page**

**18.1 Actor:** Mentor

**18.2 Purpose:** To view a student's details and notes assigned to that student.

**18.3 Precondition:** Logged in, Search a Student.

**18.4 Event Flow:**

18.4.1. Mentor searches for a student using the search bar.

18.4.2. The system displays a list of students matching the search criteria.

18.4.3. Mentor selects a student from the search results.

18.4.4. The system redirects the mentor to the student's page, displaying personal details, assigned notes, and other relevant information.

**18.5 Special/Exceptional Requirements:** Student must exist in the database. Else, an error message will be shown.

## **19. Comment on a Student's Note**

**19.1 Actor:** Mentor

**19.2 Purpose:** To add a comment to a public note assigned to a student.

**19.3 Precondition:** Logged in and Viewing Student Page.

**19.4 Event Flow:**

19.4.1. Mentor navigates to the student page and views the student's notes.

19.4.2. Mentor selects a note to comment on.

19.4.3. Mentor adds a comment in the provided text field.

19.4.4. Mentor clicks "Save" to post the comment.

**19.5 Special/Exceptional Requirements:** Comment visibility depends on the note's visibility (public/private).

## **20. Add a Note to Student**

**20.1 Actor:** Mentor

**20.2 Purpose:** To assign a new note to a student.

**20.3 Precondition:** Logged in and Viewing Student Page.

**20.4 Event Flow:**

20.4.1. Mentor navigates to the student page.

20.4.2. Mentor clicks the "Add Note" button.

20.4.3. The "Add Note" form appears, allowing the mentor to select a tag, write content, and assign the note to the student.

20.4.4. Mentor chooses the visibility (public/private) for the note.

20.4.5. Mentor clicks "Save" to assign the note to the student.

**20.5 Special/Exceptional Requirements:** The student must exist in the database.

## **21. View Mentor Page**

**21.1 Actor:** Mentor

**21.2 Purpose:** To view another mentor's details and notes given by that mentor.

**21.3 Precondition:** Logged in, Search.

**21.4 Event Flow:**

21.4.1. Mentor searches for another mentor using the search bar.

21.4.2. The system displays a list of mentors matching the search criteria.

21.4.3. Mentor selects a mentor from the search results.

21.4.4. The system redirects the mentor to the other mentor's page, displaying details

like their notes and assigned students.

**21.5 Special/Exceptional Requirements:** The mentor must exist in the database.

## **22. Comment on a Mentor's Note**

**22.1 Actor:** Mentor

**22.2 Purpose:** To add a comment to a note given by a mentor that is public or has comment access.

**22.3 Precondition:** Logged in and Viewing Mentor Page.

**22.4 Event Flow:**

22.4.1. Mentor navigates to another mentor's page and views their notes.

22.4.2. Mentor selects a note that is public or has comment access.

22.4.3. Mentor clicks on the "Add Comment" button associated with the note.

22.4.4. A text box appears for the mentor to type a comment.

22.4.5. Mentor enters their comment and clicks "Save" to submit it.

22.4.6. The comment is saved and displayed below the note.

**22.5 Special/Exceptional Requirements:** The note must be either public or have comment access enabled for the comment feature to be available.

## **23. View Cohort Page**

**23.1 Actor:** Mentor

**23.2 Purpose:** To view a cohort page and view its notes.

**23.3 Precondition:** Logged in, Search.

**23.4 Event Flow:**

23.4.1. Mentor searches for a cohort using the search bar.

23.4.2. The system displays a list of cohorts matching the search criteria.

23.4.3. Mentor selects a cohort from the search results.

23.4.4. The system redirects the mentor to the cohort's page, displaying its details and associated notes.

**23.5 Special/Exceptional Requirements:** The cohort number must be valid.

## **24. Add a Note to Cohort**

**24.1 Actor:** Mentor

**24.2 Purpose:** To assign a new note to a cohort.

**24.3 Precondition:** Logged in and Viewing Cohort Page.

**24.4 Event Flow:**

24.4.1. Mentor navigates to the cohort page.

24.4.2. Mentor clicks the "Add Note" button.

24.4.3. The "Add Note" form appears, allowing the mentor to select a tag, write content, and assign the note to the cohort.

24.4.4. Mentor chooses the visibility (public/private) for the note.

24.4.5. Mentor clicks "Save" to assign the note to the cohort.

**24.5 Special/Exceptional Requirements:** None.

## **25. Log Out**

**25.1 Actor:** Mentor

**25.2 Purpose:** To log out of the account.

**25.3 Precondition:** Logged In.

**25.4 Event Flow:**

25.4.1. User clicks the "Log Out" button in the navigation menu or profile settings.

25.4.2. The system prompts the user with a confirmation message to ensure they want to log out.

25.4.3. User confirms the log-out action.

25.4.4. The system logs the user out of the platform.

25.4.5. The user is redirected to the login page or the homepage without access to personalized features.

**25.5 Special/Exceptional Requirements:** The user must be logged in.

## **26. Approve an Account**

**26.1 Actor:** Admin

**26.2 Purpose:** To verify the legitimacy and approve a mentor's account during the sign-up process.

**26.3 Precondition:** None.

**26.4 Event Flow:**

26.4.1. Admin receives a notification or checks the pending mentor sign-ups.

26.4.2. Admin selects a mentor account that requires approval.

26.4.3. The system displays the mentor's submitted details and information.

26.4.4. Admin verifies the details and approves the account.

26.4.5. The system updates the account status to "Approved."

26.4.6. The mentor is notified of the approval and can now access the platform.

**26.5 Special/Exceptional Requirements:** None.

## 27. Delete an Account

**27.1 Actor:** Admin

**27.2 Purpose:** To delete an existing mentor's account.

**27.3 Precondition:** Mentor account to be deleted does exist.

**27.4 Event Flow:**

27.4.1. Admin navigates to the mentor account management page.

27.4.2. Admin searches for the mentor account to be deleted.

27.4.3. The system displays the mentor's account details.

27.4.4. Admin clicks the "Delete Account" button.

27.4.5. A confirmation message prompts the admin to confirm the action.

27.4.6. Admin confirms the deletion.

27.4.7. The system deletes the mentor's account and associated data.

**27.5 Special/Exceptional Requirements:** The mentor account must exist for deletion to proceed.

## Functional Requirements in terms of Features / Functionality

- **Note-Taking:** Users can add, edit, and delete notes related to students.
- **Pin Notes:** Important notes can be pinned to the top for quick access.
- **Public & Private Notes:** Users can flag notes as public (visible to all mentors) or private (visible only to them). They can also be marked public for some users.
- **Tag Notes:** Notes can be tagged as academic, behavior, progress, or a custom category. These categories can be color-coded. Tags will also be suggested based on the content of the note,
- **Search & Filter:** Users can search for students by name, cohort, or note content and filter notes based on date, visibility, and tags.
- **Shortcuts:** Shortcuts (Ctrl+N) are used for new notes, and (Ctrl+S) are used to save notes.
- **To-Do List:** Mentors can manage their tasks and keep track of essential to-do items.
- **Calendar Integration:** Office hours and scheduled meetings are displayed in an integrated calendar.
- **Search:** Users can easily search for students, mentors, or cohorts by name, streamlining access to relevant information.

- **Student Page:** Each student will have a profile page showing the notes/feedback given by mentors, session attendance, and other relevant information.
- **Mentor Page:** Each mentor will have a profile page that will show notes/feedback given by them, sessions taken, and other relevant information.
- **Cohort Page:** Each cohort will have a page with a list of all the students in that cohort and any feedback given by mentors to the students of that cohort.
- **Student Insights:** Mentors can flag students for special attention, helping track defaulters and students requiring follow-ups.
- **Session Insights:** Mentors can view attendance for sessions taken by them. This will be implemented using Zoom API.

## Non-Functional Requirements

- The application should allow adding new features (e.g., analytics or reporting) without major architecture changes.
- Ensure that all operations (e.g., saving notes and calendar updates) are atomic and consistent to avoid data corruption.
- The codebase should follow modular design principles, making updating or replacing components easy without affecting the entire system.
- Comprehensive documentation for code, APIs, and deployment processes to facilitate maintenance.
- The system should be intuitive and require minimal training for mentors to use effectively. Onboarding tooltips should be provided to guide new users.
- Add support for bulk actions, such as:
  - Deleting multiple notes.
  - Pinning notes.
  - Flagging or unflagging notes as private/public.

## User Interface

### Key/Common UI Elements

- **Side Panel:**
  - Display calendar, to-do list, and office hours.
  - View profile and related information upon clicking on the profile button
- **Main Screen:**
  - Show notes sorted by date(default), with an option to add, sort, and filter the notes.



- Include a search bar at the top to quickly find notes by keywords or dates.

### Search, Filter, and Sort

- Include global search functionality to search across notes, students, and events.
- Filters should allow users to narrow results by several categories. An option to sort notes should also be implemented.

### Other Key Requirements

- Maintain a consistent color scheme and typography throughout the application and use visual hierarchy to guide users.
- Provide clear error messages with actionable suggestions and use non-intrusive toast notifications for errors and success messages.
- Support common keyboard shortcuts (e.g., Ctrl+N to create a new note, Ctrl+S to save a note).
- Use color-coded labels for notes and items on the calendar and to-do list.

### Performance Requirements

Evalio is designed for a relatively small user base of approximately 30 mentors, ensuring that performance requirements remain efficient without needing large-scale optimizations. The following constraints apply:

- **Concurrent Users:** The system should support simultaneous access by all mentors without noticeable performance degradation. Given the limited user base, high concurrency optimization is not a primary concern.
- **Response Time:** Page loads, note retrieval, and updates should have a response time of under 500ms under normal usage conditions.
- **Database Performance:** PostgreSQL should efficiently handle queries related to note searches, student records, and filtering without significant delays. Indexing and caching will be implemented where necessary.
- **Server Load:** Given the expected traffic, a single instance of the backend server should be sufficient, with the potential for scaling if future demand increases.
- **Storage Requirements:** Since the platform primarily handles text-based data (notes, to-do lists, and metadata), database storage requirements will be minimal, with periodic clean-up of old or irrelevant records.
- **Scalability:** While not an immediate requirement, the system should be designed to allow easy expansion if the number of mentors or data volume grows significantly.

These constraints ensure a smooth and responsive user experience while optimizing resource usage.

## Technical Constraints

The platform will be built using the following technologies:

- **Backend:** [Phoenix](#) (Elixir-based web framework)
- **Frontend:** [Next.js](#) (React framework for server-side rendering and static site generation)
- **UI Components:** [shadcn/ui](#) (Customizable UI component library for React)
- **Styling:** [Tailwind CSS](#) (Utility-first CSS framework)
- **Database:** [PostgreSQL](#) (Reliable, Scalable, ACID-compliant)

## Design Constraints

- The tech stack for the platform should be free and open source.
- The database should support scalability to accommodate future features, such as analytics or additional mentor/student information, without requiring significant refactoring.
- The application must be compatible with already existing systems in use at TalentSprint.
- The application must adapt seamlessly to various desktop screen sizes. No support is required for mobile or tablet views.

## External Interface

- The application should be compatible with major browsers like Google Chrome, Firefox, Safari, and Microsoft Edge. Backward compatibility (i.e. older browsers) is not required.
- The application should be able to sync with external platforms like Google Calendar and Zoom via APIs. (optional)
- Provide external interaction via keyboard commands.

## Security / Privacy Requirements

- The application should be accessible only to the members of the organization. Only authorized users should be able to execute the different use cases.
- Notes flagged as "Private" must remain confidential and accessible only to the creating mentor. Data for "Public" notes should be accessible only to authenticated mentors within the same system.
- Mentor and student information should not be accessible to unauthorized users.

- Unless explicitly authorized, prevent users from downloading or exporting data (notes, profiles, etc.).