



503073 – WEB PROGRAMMING & APPLICATIONS

FINAL PROJECT – SEMESTER II/2024-2025

Last updated on 02/01/2025

Mandatory Compliance with Red-Highlighted Content

For content highlighted in red, students must adhere strictly to the descriptions provided. Any deviation, modification, or addition of features will result in the submission being disqualified from evaluation.

I. Overview

Develop a note management application that enables users to create and organize their notes. The notes can encompass diverse content formats, including text, images, and attached files. Students must strictly adhere to the functionalities outlined in this assignment. Introducing unrelated features that deviate from the specified requirements may indicate that the student did not independently complete the project. While students are permitted to utilize libraries and frameworks to aid in development—such as Bootstrap or React for the front end and Laravel for the back end - **no additional features beyond those described should be implemented.**

II. Detailed Description

2.1 Account Management

- To access the service, users must possess an account and log in. **If a user attempts to access the website without logging in, they will be immediately redirected to the login interface.** Upon successful login, users will be directed to their personalized homepage, where they can view their notes.
- When registering for a new account, users are required to provide only the following information: **email address, display name, and password** (entered twice for confirmation). The website will not store passwords directly but will instead retain their **bcrypt-hashed** equivalents. **Upon successful registration, users will be automatically logged in without requiring them to log in again.** Additionally, an email containing an activation link will be sent to the user's registered email address.
- Before the account is activated, users will have access to all functionalities, with the only distinction being a prominent notification on the website indicating that the account is unverified and advising them to check their email to complete the activation process. Once the user clicks the activation link in the email, the account will be activated, and the notification will disappear.
- The User Preferences screen allows users to adjust specific settings, such as the font size of their notes, note colors, or toggling between light and dark themes.
- The Password Reset feature enables users to recover their password via email. Users must either click on the link provided in the email or input an OTP sent to their email before being redirected to the new password entry screen. Following a password reset, users are required to log in manually.

2.2 Simple note management

- **By default, notes are displayed in a grid view layout.** However, users have the flexibility to switch to a list view format if they prefer. This ensures that the interface caters to different user preferences and enhances accessibility.

- When creating a new note, users are required to input **only the title and content fields; no additional fields or attributes are permitted**. The same interface must be used for both adding and editing notes, meaning it is not allowed to develop separate screens for these two functionalities. Additionally, note content must be saved automatically, eliminating the need for users to click a “save” button. This auto-save feature ensures a seamless user experience and reduces the risk of data loss.
- For deleting notes, a confirmation dialog must always be displayed to obtain explicit consent from the user before proceeding with the deletion. This adds a layer of safety and prevents accidental loss of data. Notes can also support image attachments, with the option to include one or multiple images. Students have the discretion to implement the attachment feature in any way that fits the system’s design and usability requirements.
- Notes should be sorted by default based on either their creation time or last modified time, with the most recent notes appearing at the top of the list. However, users can “pin” specific notes to the top, ensuring these are always displayed first. If multiple notes are pinned, they should be arranged according to the time they were pinned, maintaining a clear hierarchy.
- The platform must incorporate an efficient search functionality, enabling users to search for notes by keywords. The search should scan both the title and content of notes for matches. Instead of providing a “Search” button, the system should feature a live search mechanism, which triggers as soon as users start typing into the search box. A slight delay (e.g., 300ms) can be implemented to improve performance and responsiveness.
- Label management is another essential feature. Users should be able to view a list of all labels, add new labels, rename existing ones, and delete labels as needed. A note can either be label-free or associated with one or multiple labels. Furthermore, users must have the ability to filter notes by labels, displaying only those associated with the selected labels. It is important to note that if a label is deleted, it should not affect the notes it was linked to. Similarly, if a label is renamed, all notes associated with it must reflect the updated label name automatically, ensuring consistency across the system.

2.3 Advanced note management

- To enhance security, the website should offer a mechanism to lock notes with a password. For a standard note, when this feature is activated, users must set a password specific to that note. Each note has its own unique password, unrelated to others. Once password protection is enabled, a dialog will prompt users to enter the password before allowing any action, such as viewing, editing, or deleting the note. Users also have the option to change a note's password or disable password protection entirely if they no longer wish to use this feature.
- The ability to share notes is an advanced feature, allowing users to share their notes with others through their registered email accounts. When sharing, the note owner can specify sharing permissions such as read-only access or edit rights. They can also select one or multiple recipients. Furthermore, the owner retains full control to revoke access or modify sharing settings at any time.
- Recipients of shared notes should have a dedicated section where all notes shared with them are displayed. This section must clearly indicate the sharing status of each note, specifying whether it is read-only or editable. Additionally, it should provide information about who shared the note, along with the timestamp of when the sharing occurred. For notes shared with edit permissions, real-time collaboration should be implemented using WebSocket technology. This will enable multiple users to edit the note simultaneously while observing each other's changes in real time.
- For special notes, such as those that are shared, pinned, or password-protected, the system must include a recognizable icon for easy identification. This icon should be displayed in both list view and grid view layouts, allowing users to quickly identify the note's status without needing to open it or navigate to its details. For instance, users should immediately recognize if a note is password-protected or shared just by glancing at the icon.

2.4 Additional Requirements

UI and UX Considerations

- User Interface (UI) and User Experience (UX) are critical factors in creating an engaging website. The UI focuses on aesthetic elements, such as whether the design is visually appealing, while UX pertains to how users feel while interacting with the site—whether it is

convenient or frustrating, fast or slow. Websites with average UI and UX will not be awarded points in this category. To receive 0.25 points, the website must demonstrate good UI and UX, while exceptional performance in these aspects will earn 0.5 points.

Responsive Design

- Responsive design is a separate evaluation criterion due to its importance in both UI and UX. A responsive website must adapt seamlessly to different screen sizes, optimizing the use of available space across devices. At a minimum, the website must function well on three types of devices: smartphones, tablets, and desktop monitors. Good responsiveness ensures that users have a consistent and enjoyable experience regardless of the device they use.

Offline Capabilities

- Offline capabilities, a hallmark feature of Progressive Web Apps (PWAs), should be implemented using JavaScript. This feature allows the website to remain accessible and display note content even when the user is offline or temporarily loses their internet connection. To implement this effectively, developers need to explore PWA principles and integrate a local database (using JavaScript) with the online database (e.g., MySQL). This combination ensures a seamless user experience by enabling users to view notes offline and synchronizing the data when an internet connection is restored.

Online Deployment

- Online deployment involves hosting the website and all related services, such as databases, on a public platform, making it accessible to anyone via a domain name. To earn 0.5 points for this requirement, you must deploy your website and database on an online service, ensuring the system runs without errors and remains operational during the instructor's grading period.
- If online deployment is not feasible, an alternative is to use docker-compose. In this case, you must follow the instructor's guidelines and provided project template to develop your application. Submitting the project in the correct format will allow the instructor to replicate and test it on their own system.

2.5 Notes

The descriptions provided above are not exhaustive or definitive. Students must rely on their experience and conduct additional research on similar services to ensure optimal implementation. A feature is considered fully completed only if it functions as described, includes error handling, avoids critical security vulnerabilities, and adheres to industry best practices commonly employed by other services. Below are examples illustrating approaches that distinguish between subpar and better implementations:

Sharing Notes

- Subpar Approach: Allowing users to share notes simply by entering an email address. If the email exists in the system, the recipient sees the shared note when they log in.
- Better Approach: Implementing a mechanism to validate the entered email, ensuring it belongs to a registered user. Additionally, the recipient should receive an email notification about the shared note. At a minimum, a notification should be prominently displayed in the recipient's account upon their next login, making it easy to notice and access the shared note.

Password-Protected Notes

- Subpar Approach: Creating a password by allowing users to enter the desired password once. Similarly, when changing the password, the user enters the new password directly. Disabling password protection requires only selecting an option from the menu.
- Better Approach: Enhancing security by requiring users to enter the password twice when creating or updating it, ensuring there are no typing errors. When changing the password, the user must first provide the current password and then input the new password twice. Disabling password protection should require the user to re-enter the current password as a final confirmation before deactivating it.

Shared Notes Details

- Subpar Approach: Simply allowing recipients of shared notes to view the content without any additional details. The note owner has no visibility into who the note has been shared with.
- Better Approach: Providing a detailed overview of shared notes for the note owner. When the owner clicks on any shared note, they should see its sharing status and a list of all recipients. This list should include each recipient's email address and their

permissions (e.g., read-only or editable). The owner should also have the ability to modify or revoke sharing permissions from this interface.

III. CODING GUIDELINE

- If you are unable to deploy your website publicly and choose not to use Docker Compose, your submission will still be evaluated, but you must strictly adhere to specific guidelines to ensure the instructor can run your project successfully. Always use relative URLs instead of hardcoding fixed ports and hostnames; for example, replace `` with `` to ensure flexibility across different configurations. Additionally, avoid placing your web project in subdirectories; rather than serving it at `http://localhost:8080/final_project`, ensure it is accessible directly at `http://localhost:8080/`. For those using XAMPP, this means storing your project files in the `htdocs` directory instead of subfolders like `htdocs/final_project`. While the port 8080 is used as an example, you may use any port, but it must not be hardcoded in the source code.
- You are permitted to use any libraries or frameworks, such as React for the frontend or Laravel for the backend, provided you either deploy the project online or set it up using Docker Compose to facilitate access for the instructor. If neither option is feasible, include clear and detailed instructions in a `readme.txt` file to guide the instructor on running the project locally. Adhering to these guidelines ensures your project remains accessible and reviewable.

IV. RUBRIK

ID	CRITERIA	LEVEL	1	2	3
		POINT	0 PT	25-75% POINT	FULL POINT

	Account management	2.0			
1	User registration	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
2	Account activation	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
3	User login and logout	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
4	Password reset	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
5	View profile and avatar	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
6	Edit profile and avatar	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
7	Change password	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
8	User preferences	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
	Simple note management	4.0			
9	Display notes in listview	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
10	Display notes in gridview	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.

11	Create notes	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
12	Update notes	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
13	Delete notes	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
14	Auto-save notes	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
15	Attach images to notes	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
16	Pin notes to top	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
17	Search notes	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
18	Label management (listing, add, edit, delete)	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
19	Attach labels to notes	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
20	Filter notes based on labels	0.25	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
	Advanced note management	2.0			
21	Enable and disable password on notes	0.5	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.

22	Password protection, change password on notes	0.5	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
23	Share and receive notes	0.5	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
24	Collaboration and realtime modification	0.5	The feature is not available or it does not work at all or it is done completely wrong	The feature has been implemented but it is not working properly or there are still some serious bugs	The feature has correctly been implemented as required, with no bugs or only insignificant bugs.
	Other requirements	2.0			
25	UI and UX <small>Scored based on the teacher's perception; only above-average work earns points (a basic interface gets no points)</small>	0.5	Poor design, inconsistent UI, no feedback, no accessibility.	Basic styling, inconsistent UX, limited feedback, partial accessibility adherence.	Polished UI, intuitive UX, comprehensive feedback, fully accessible.
26	Responsive <small>Scored based on the teacher's perception; only above-average work earns points (a basic interface gets no points)</small>	0.5	No adaptation to screen sizes, misaligned elements, no media queries.	Partially responsive, limited media queries, layout issues on some devices.	Fully responsive, optimized layouts, proper scaling, smooth transitions across devices.
27	Offline Capabilities	0.5	No offline functionality, no caching, no service worker.	Limited offline access, partial caching, unreliable service workers, errors offline.	Robust offline access, effective caching, proper offline data synchronization.
28	Online deployment	0.5	Not deployed, no hosting or deployment plan.	Online but unstable, poor performance, lacks HTTPS, minimal optimization.	Fully deployed, secure (HTTPS), optimized for performance, scalable, high uptime.

V. OUTPUT REQUIREMENTS

- Required submission components include:
 - o **Rubrik.docx:** This file lists the required 28 features for the project. Teams should self-assess their completion level in this file. The instructor will provide this file at the submission time. The file will also include the public URL to the web application and any required username/password for login.
 - o The "source" folder:
 - **if you do not use docker compose:** the source folder should contain the entire source code of the web app (e.g. frontend, backend), along with relevant database files. Ensure that this source code can be run on the teacher's

computer. The project needs to be "**cleaned**" to remove unnecessary content before submission and to reduce the size of the compressed file.

- **if you use docker compose:** This folder must contain all source code for the necessary modules, the docker-compose file, and instructions for running the application on the instructor's machine (e.g., where to run npm install and docker-compose up). Ensure thorough testing is completed before submission.
- The demo video, titled "**demo.mp4**" must include the participation of **all team members** to introduce the group's product. The video should provide a brief overview of the technologies and architecture of the web application. Following this, the team **must sequentially demonstrate each feature developed, based on the evaluation form containing 28 criteria. Any criteria not demonstrated in the video will be considered as not implemented, even if the group claims otherwise in their self-assessment form.** The video must have a minimum resolution of 1080p, clear audio, and be easy to understand. If the video file is too large, the team should upload it to YouTube and include the link in the submission..
- **Readme.txt** file: Provide all necessary information for the evaluation process, such as project building and running instructions, URL + server login information (if applicable), and **usernames/passwords** for accounts with pre-loaded data for assessment. Include any relevant notes on building, running, and using the application for teachers to reproduce the project. If your team implements some optional features (which get extra points), it should be clearly stated in the readme file.
- Organize all the above contents into a folder named **id1_fullname1_id2_fullname2**, then compress this folder in ZIP format with the same name, e.g., id1_fullname1_id2_fullname2.zip. A team representative should submit this file on the online learning system as instructed by the course instructor.
- Teachers do not accept submissions via email, **only elearning is accepted**.
- Failure to submit source code, video or rubrik.docx will result in the entire team receiving **a score of 0**.
- If your team submits an unrelated project, it will not be graded, and **the entire team will receive a score of 0**. For example, if your team uses the source code from a different e-commerce site and only implements a few features related to this assignment while most features are unrelated, the project will receive 0 points.
- The Essay is entirely independent of the Final Project. Therefore, all team members must participate in both the Essay and the Final Project. The Essay will be assessed by the lab instructor, while the Final Project will be assessed by the theoretical instructor.
- Groups are prohibited from sharing code with each other, obtaining source code from the internet, and must take responsibility for protecting their group's source code. Groups with similar source code (verified by specialized software) or code found online, even if only in part, will receive **a score of 0** for all members, regardless of which group shared or received the code.

- Deductions will also apply in the following situations:
 - Late submission: 1-day late deducts 1 point. Submissions late by 1 second to less than 1 day are considered 1 day late.
 - Complex project configuration without specific instructions for instructors to compile and run the program: Deduction of 2 points.
 - Submit the entire project without performing a clean to remove unnecessary files: 0.5 point.
 - Failure to provide necessary grading information, such as missing usernames/passwords, incorrect file naming, or not submitting required content: 1.0 point.

Don't forget to contact the instructor directly or via email at maivanmanh@tdtu.edu.vn to promptly address any questions or concerns related to the project.