Sofia University “St. Kliment Ohridski”

Faculty of Mathematics and Informatics

Course Project Report

DESIGN AND INTEGRATION OF SOFTWARE SYSTEMS

UniNotes

Team:

Anastasia Yakimovska, 855362
Ivana Donchevska, 855364

Major: Software Engineering
Course: IV

Sofia, January 2024

Table of contents

Introduction 4

Goal of the document 4

Overview of the system 4

Definitions and acronyms 5

Application design 7

Functional Requirements 7

Non-functional Requirements 8

System Architecture 8

UML Diagram 9

Program Development 10

Used Technologies 10

Database Oraganization 10

Functionality List 12

Class Diagram 13

Sequence Diagram 14

System Deployment 15

Software and/or System Testing with suitable test cases 16

Test Strategy 16

Testing Scenarios 16

Dashboard Test 16

Notes Test 16

Board test 16

Application Walkthrough 18

User Registration and Login 18

User Registration 18

User Login 20

Board and Note Management 22

Create a New Board 22

Edit Board 26

Add Note to a Board 27

Edit Note 29

Clear Board 31

Upvote Note 32

Board Navigation 33

Importing and Exporting Notes 34

Import Notes to a Board 34

Export Notes from a Board 36

User Profile Management 37

Modify or Delete User Profile 37

Conclusion 41

Introduction

Goal

This documentation aims to provide a comprehensive understanding of UniNotes, a sophisticated system designed for efficient note management and collaboration among students within specific academic specialties. The primary goal is to offer detailed insights into the system's architecture, functionalities, and deployment, ensuring that both end-users and developers gain a profound understanding of UniNotes.

Overview

In an era where effective collaboration and timely information sharing are paramount, UniNotes stands out as a user-friendly platform. Here, students can effortlessly upload, download, and share notes with their peers, fostering a community-driven approach to learning. Beyond traditional note-sharing, UniNotes allows users to create news notes, facilitating the exchange of important updates and information within the student community. Students can actively participate by upvoting notes, ensuring that valuable content receives the attention it deserves.

The UniNotes system consists of these modules:

1. **Authentication and User Management Module:** The Authentication Module verifies user identities during login by validating credentials against stored data. In UniNotes, it ensures secure access, permitting actions like uploading, downloading notes, and collaborative features for authenticated users. The User Management Module oversees the user lifecycle, including registration, profile creation, modifications to profile in UniNotes. It centrally manages user accounts, recording and maintaining accurate data, and often involves database interactions, utilizing Laravel's Eloquent ORM for efficient data management.
2. **Note and File Management Module:** The Note and File Management Module is a critical component of the UniNotes system responsible for overseeing the creation, storage, retrieval, and sharing of notes among users. It is designed to facilitate seamless collaboration by providing a platform for students to upload, download, upvote and share notes within their academic specialties.
3. **Security and Database Integration Module:** It serves as a dual-purpose component, addressing security concerns to protect the system and facilitating smooth interactions with the database to ensure data integrity and efficient data management in the UniNotes system.
4. **Presentation and Frontend Module:** The components in the Presentation and Frontend Module ensure a well-organized and responsive user interface, effective

URL routing, seamless communication between the frontend and backend, and the enforcement of additional processing steps through middleware.

Definitions and acronyms

- **Laravel** - a web application framework that uses HTTP principles
- **React** - A JavaScript library employed for building dynamic and interactive user interfaces within the frontend.
- **User interface** - an interface in which the elements provided to the user for control are implemented in the form of graphic images (menus, buttons, lists, etc.).
- **distributed software architecture** - an architecture in which each individual module is a separate independent unit
- **application (app)** - software designed to help the user perform a certain task
- **module** - logically separated software unit
- **web application** - an application that users access over a network such as the Internet
- **user** - a person who uses a computer or network service
- **database** - a collection of information that is organized so that it can be easily accessed, managed, and updated
- **server** - a running instance of a software system that can accept requests from a client and return appropriate responses
- **client** - a part of a computer or software system that accesses a service provided by a server
- **Inertia -** A technology facilitating seamless integration between the Laravel backend and the React frontend.
- **SQLite -** A self-contained, serverless, and zero-configuration relational database management system used as the primary database.
- **Apache -** The web server software utilized for hosting on macOS, delivering web content to users.
- **CSV -** Comma-Separated Values, a file format used for importing and exporting notes, providing a simple tabular structure.
- **ORM -** Object-Relational Mapping, a programming technique used in Laravel, simplifying the interaction between the application and the database.
- **HTTP -** Hypertext Transfer Protocol, the foundation for data communication on the World Wide Web, utilized for client-server interactions.

Acronyms:

- **PHP -** Hypertext Preprocessor.
- **UML -** Unified Modeling Language.
- **API -** Application Programming Interface.
- **SQL -** Structured Query Language.

- **JSON -** JavaScript Object Notation.
- **SMTP -** Simple Mail Transfer Protocol (used for sending password reset emails).
- **CI/CD -** Continuous Integration/Continuous Deployment (methodologies for software development and deployment).

Application design

Requirements

Functional Requirements

ID	Requirement	Description	Priority
FR_01	Registration (User Authentication)	Enable user to securely create account, using their name, email and password.	1
FR_02	Login (User Authorization)	User can securely access the system using the correct email and password combination.	1
FR_03	User Profile Update	User can update their email, password and delete their account.	2
FR_04	Board Creation	User can create a board, where they can add notes.	1
FR_05	Board Access	User can access a board to see all notes in said board.	1
FR_06	Clear Board	User can clear board of all notes.	2
FR_07	Note Creation	User can create a new note to add to the board, as well as select the color.	1
FR_08	Note Edit	User can edit the text of the note, and its color.	1
FR_09	Note Upvote	Users can upvote a note, to have it gain more attention by other users.	2
FR_10	Group Notes By Color	Notes should be arranged so that they are grouped by color.	3
FR_11	Import/Export CSV	Users should be able to import and/or export notes to a board using csv data files.	2

FR_12	Password Reset	User should be able to reset their password.	2

Non-functional Requirements

1. Security:
    - The system should employ secure authentication mechanisms.
    - User data, including notes and personal information, should be stored securely.
    - HTTPS should be enforced for secure data transmission.
2. Performance:
    - The application should be responsive, with minimal latency in loading and interacting with notes.
    - Scalability to accommodate a growing user base should be considered.
3. Usability:
    - The user interface should be intuitive, ensuring ease of use for a diverse user base.
    - Accessibility features should be implemented for an inclusive user experience.
4. Reliability:
    - The system should be reliable, minimizing downtime and ensuring data integrity.
    - Regular backups and a robust error-handling mechanism should be in place.

System Architecture

UniNotes adopts a client/server architecture:

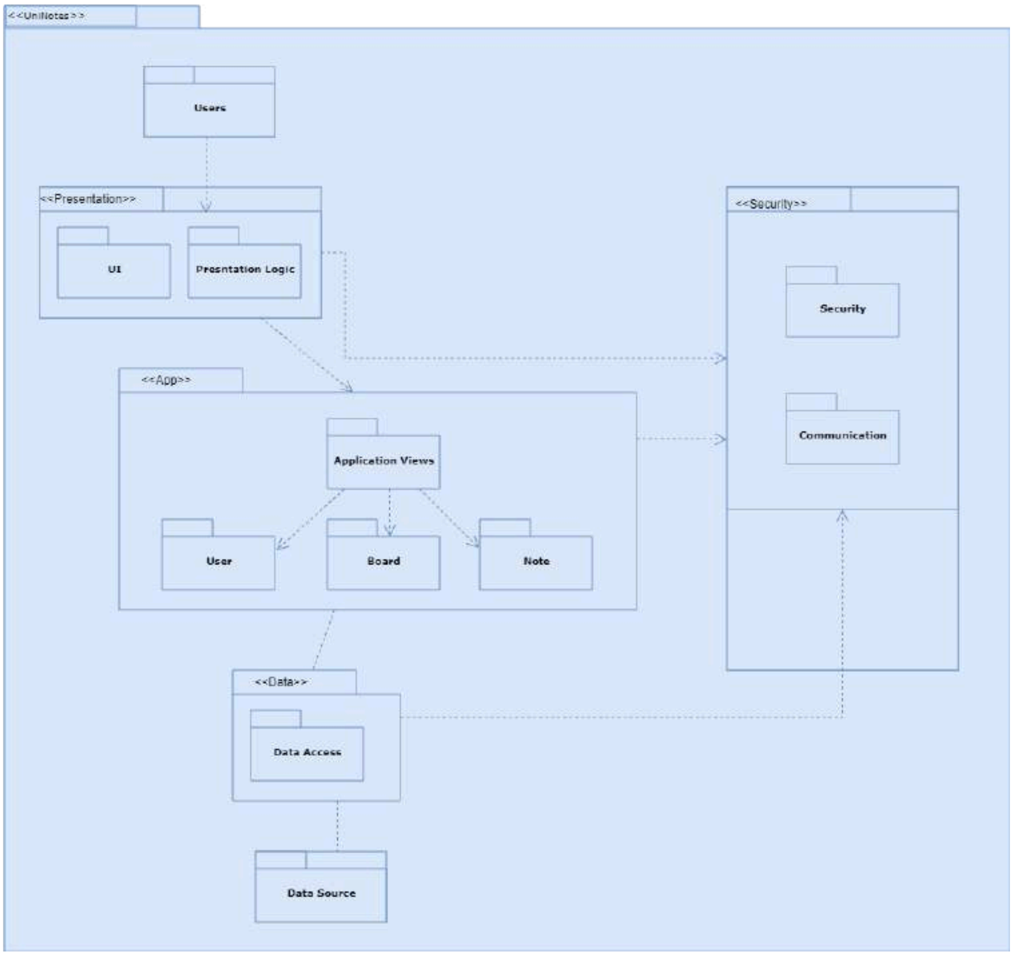
1. Client:
    - The client side includes the user interfaces where students interact with the system.
    - It is a web-based client using HTML, CSS, and JavaScript for a browser-based application.
2. Server:
    - The server side is implemented using Laravel, handling business logic, data processing, and storage.
    - The server communicates with a database, using Laravel's Eloquent ORM for data management.
3. Database:
    - The application data, including user profiles, boards, notes, and system settings, is stored in a relational database.

4. HTTP Communication:
    - The client and server communicate using the HTTP protocol.
    - Laravel's routing system and controllers handle incoming HTTP requests and generate responses.
5. Middleware:
    - Middleware in Laravel is used for processing HTTP requests at various stages, enforcing security measures, and modifying requests and responses.

By adhering to these requirements and system architecture, UniNotes aims to provide a secure, collaborative, and user-friendly platform for students to manage and share notes within their academic specialties.

UML Diagram - Package

This diagram visually represents the major packages or components within the UniNotes system and the relationships between them. The arrows or lines between packages indicate dependencies or associations, showcasing the overall structure of the system in terms of its major functional areas.



Program Development

Used Technologies

Operating System:

- MacOS: The operating system used for development.

Web Server:

- Apache (Built-in): The built-in Apache web server on macOS used for hosting and serving the UniNotes application.

Database:

- SQLite: The relational database management system used for storing and managing structured data.

Backend:

- PHP with Laravel Framework: Laravel is used as the backend framework for developing server-side logic and APIs.

Frontend:

- React: React is employed for building the user interface and enhancing the frontend experience.

Frontend-Backend Interaction:

- Inertia: Inertia is used to connect the React frontend with the Laravel backend seamlessly.

Testing:

- Laravel Dusk: Laravel Dusk is utilized for browser testing and end-to-end testing of the application.

Database Organization

The database organization of UniNotes is intricately designed, utilizing SQLite as the relational database management system within the Laravel framework's migration system. SQLite efficiently structures and manages data, configuring tables and relationships to store essential information. This setup ensures a well-organized and responsive database schema, enabling UniNotes to seamlessly handle user data, notes, and related information, thereby supporting the core functionality of the application.

1. **'users' Table:**
    - This table stores user information.
    - Fields: id, name, email, email_verified_at, password, remember_token, timestamps.
2. **'password_resets' Table:**
    - Used for password reset information.
    - Fields: email (indexed), token, created_at.
3. **'boards' Table:**
    - Stores information related to boards in UniNotes.
    - Fields: id, title, description, notes (a JSON field to store notes associated with the board).
4. **'notes' Table:**
    - Contains information about individual notes.
    - Fields: id, text, color, votes, selected, board_id (foreign key referencing 'boards' table).

This database schema is designed to capture user details, manage password reset tokens, organize boards, and store individual notes associated with those boards. The use of foreign key relationships ensures data integrity, and the JSON field for 'notes' allows for flexibility in storing note-related information.

Functionality List

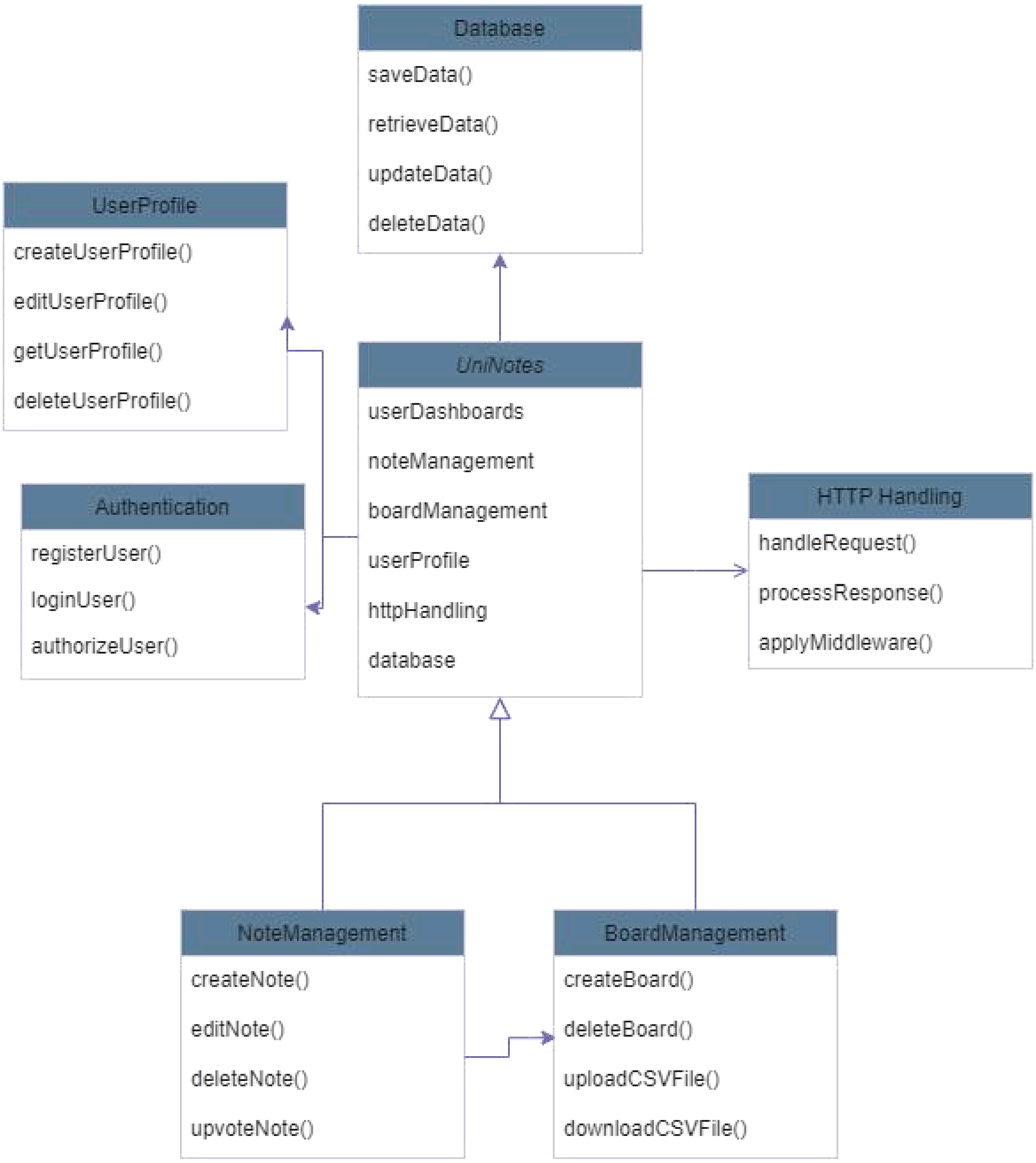
1. **User Authentication:**
    - User registration, login, and logout.
2. **Note Creation and Management:**
    - Creation of new notes with text content, color coding, and voting functionality.
    - Editing and deletion of existing notes.
    - Importing notes to a board using a CSV file.
    - Exporting notes from a board to a CSV file.
    - Management of notes within specific boards.
3. **Board Creation and Management:**
    - Creation of new boards with a title and description.
    - Adding and removing notes from boards.

- Displaying boards and associated notes on separate dashboards.

4. **User Profile Management:**
   - Creation and customization of user profiles.
   - Modification of user details.
5. **Password Reset:**
   - Sending password reset tokens to users via email.
   - Processing password resets based on valid tokens.
6. **Background Jobs:**
   - When executing a command it returns how many notes you have to do.
   - It is a custom made command that is scheduled to run on each minute and gets the number of notes, that are stored in the database.

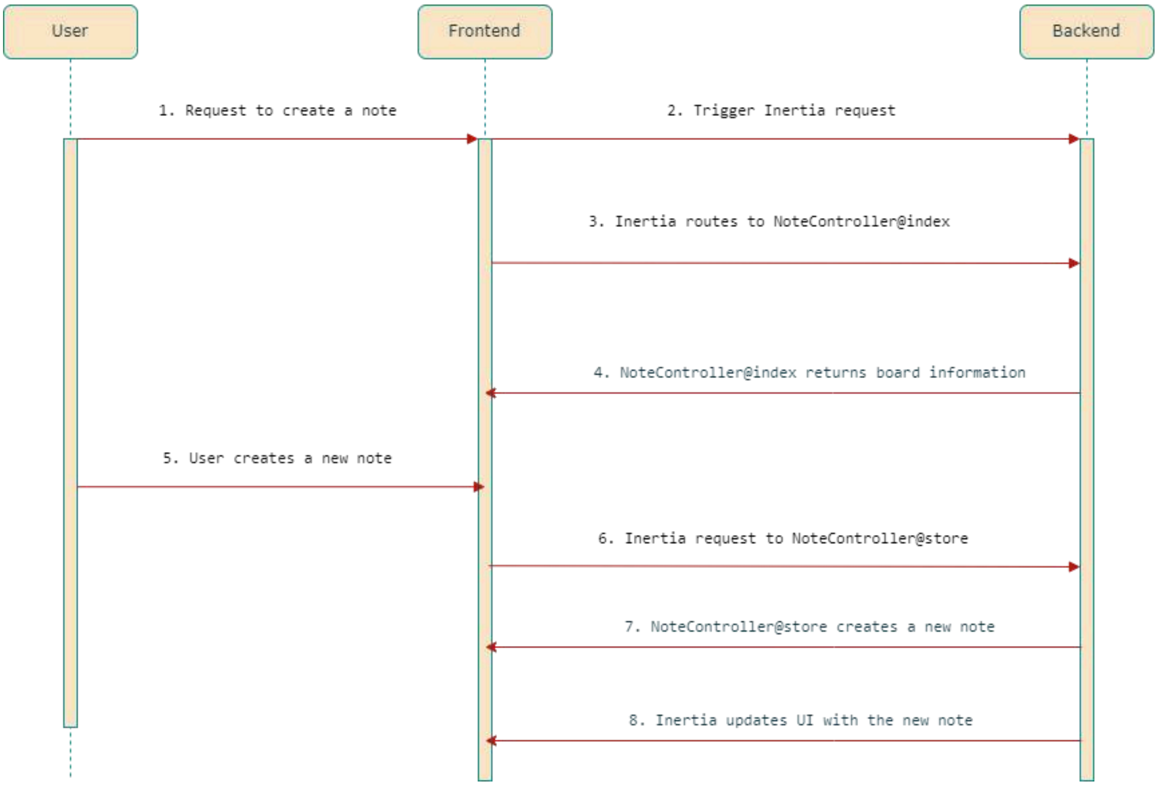
Class Diagram of the System

This class diagram provides a structured overview of the major components, their relationships, and key functionalities within the UniNotes system. Each package contains classes and methods that encapsulate specific responsibilities, contributing to the overall functionality of the system.



Sequence Diagram of a core functionality

In order to gain a basic understanding of the interactions between the system components we have included a sequence diagram of a core functionality of UniNotes, which is *Creating a new note by the user.*


User
Frontend
Backend
1. Request to create a note
2. Trigger Inertia request
3. Inertia routes to NoteController@index
4. NoteController@index returns board information
5. User creates a new note
6. Inertia request to NoteController@store
7. NoteController@store creates a new note
8. Inertia updates UI with the new note

System Deployment

The deployment of the application is on Built-in Apache for macOS. Here are the instructions for running the application.

1. Ensure you have installed the adequate php and react versions. As well as Apache, if needed.
2. You unzip the UniNotes.zip file to your desired location.
3. You open 4 terminals on your computer and navigate to the directory which you have unzipped.
4. In each of the terminals you run these commands:

   ```
   php artisan serve // to start the server
   ```

   ```
   npm run dev  // to start the client
   ```

   ```
   php artisan queue:work // to start the queue
   ```

5. In one of the terminals, in order to run the background job you run

   ```
   run crontab -e
   ```

   and add this command to run the custom made command:

   ```
   * * * * * cd /path/to/your/project && /usr/local/bin/php artisan check:notes >>
   /path/to/your/project/storage/logs/laravel.log 2>&1
   ```

6. You should now be able to open the application on your localhost address.
7. To see how long you have until another execution of the background job run

   ```
   php artisan schedule:list
   ```

   And you should get an output in the terminal similar to this

   ```
   * * * * *  php artisan check:notes
   ..............................................................................................
   ```

   ```
   Next Due: 14 seconds from now
   ```

8. To see the stored data from the backgound job you can look up the laravel.log file.

Software and/or System Testing with suitable test cases

Test Strategy

UniNotes adopts a comprehensive testing strategy to ensure the reliability, functionality, and security of the application. We've done end-to-end testing on the application using Laravel's Dusk.

Testing Scenarios:

Dashboard Test

- Verify User Login
- User Authorization
- Go to Profile Page
- Click on Button
- User Logout

Tests: 5 passed (18 assertions)

Duration: 21.43s

Notes Test

- Verify User Login
- Open Board
- Create Note
- Select Note
- Edit Note
- Change Note Color
- Delete Note
- Export Notes to CSV file
- Back to the Board
- Clear Board

Tests: 12 passed (59 assertions)

Duration: 87.68s

Board test

- Verify User Login
- Create Board
- Delete Board
- Open Board

- Edit Board
- Back to Dashboard
- Clear Dashboard

Tests: 7 passed (40 assertions)

Duration: 42.5s

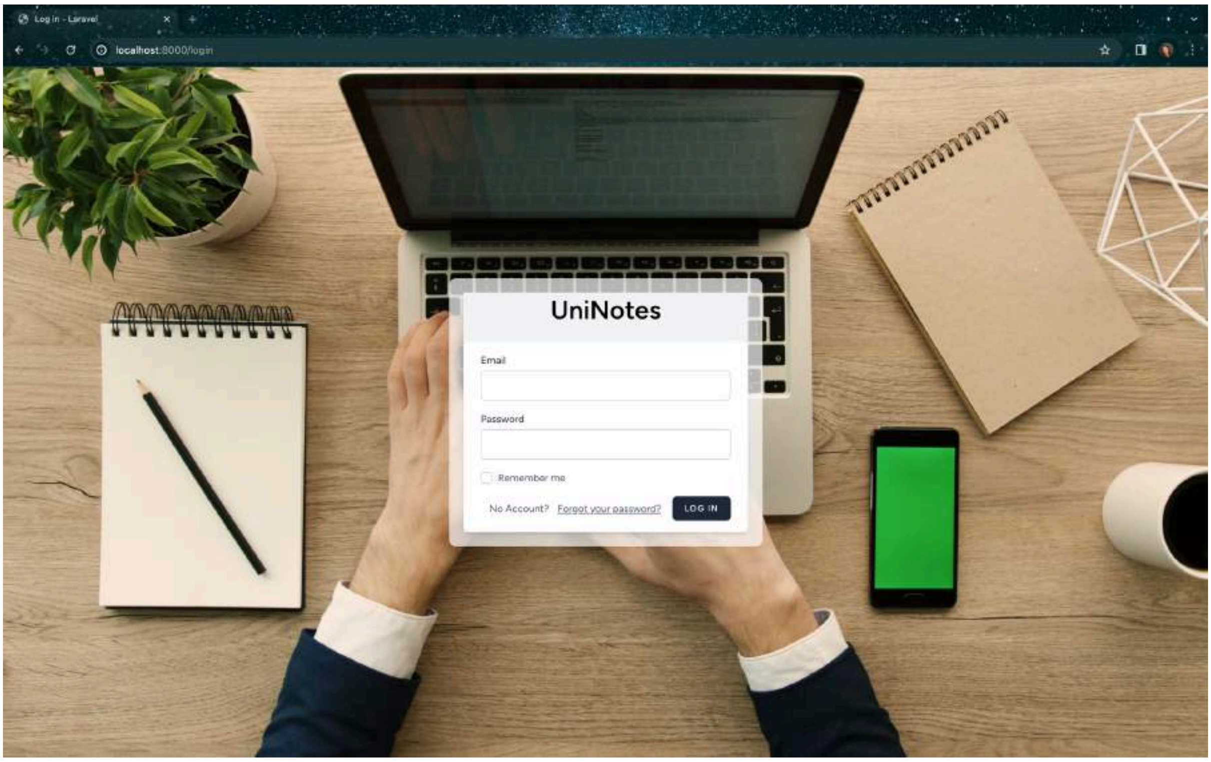
Application Walkthrough

This guide will lead you through scenarios covering fundamental user interactions to advanced features, providing a profound understanding of UniNotes as a platform for collaborative note-taking and efficient student collaboration.

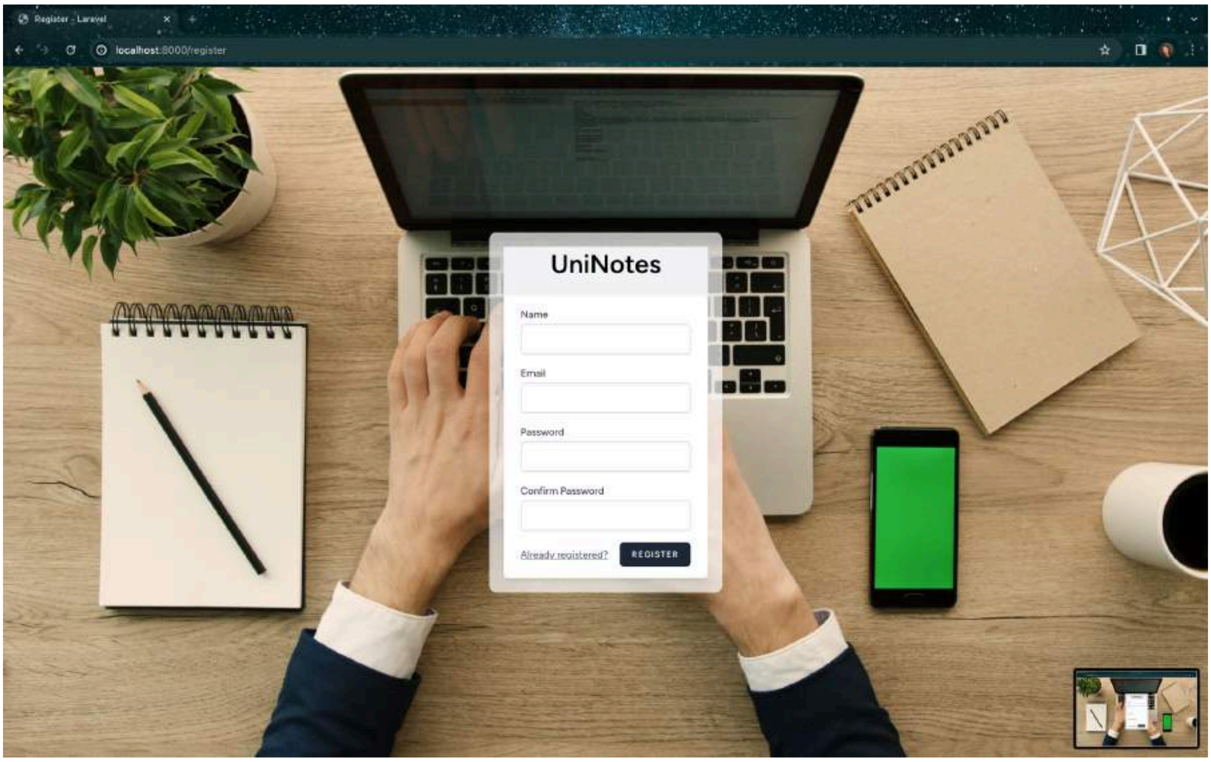
User Registration and Login

User Registration:

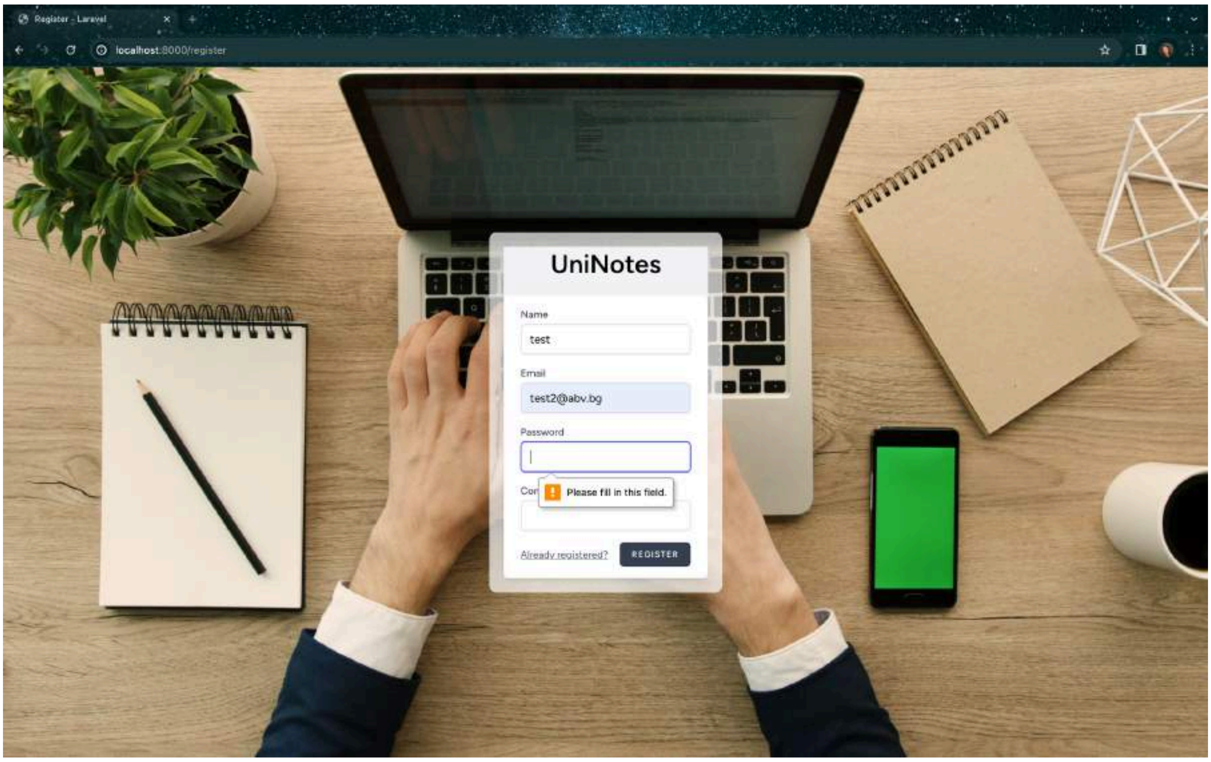
1. Navigate to UniNotes Start Page. Here you can see the Login Form.



2. To create a profile click on **No Account?** to be redirected to the Registration Page.



3. Complete the registration form by providing a name, email, and password, as well as confirmation for the password. In case of having an account already, click on **Already registered?** to be redirected back to the Login Page.

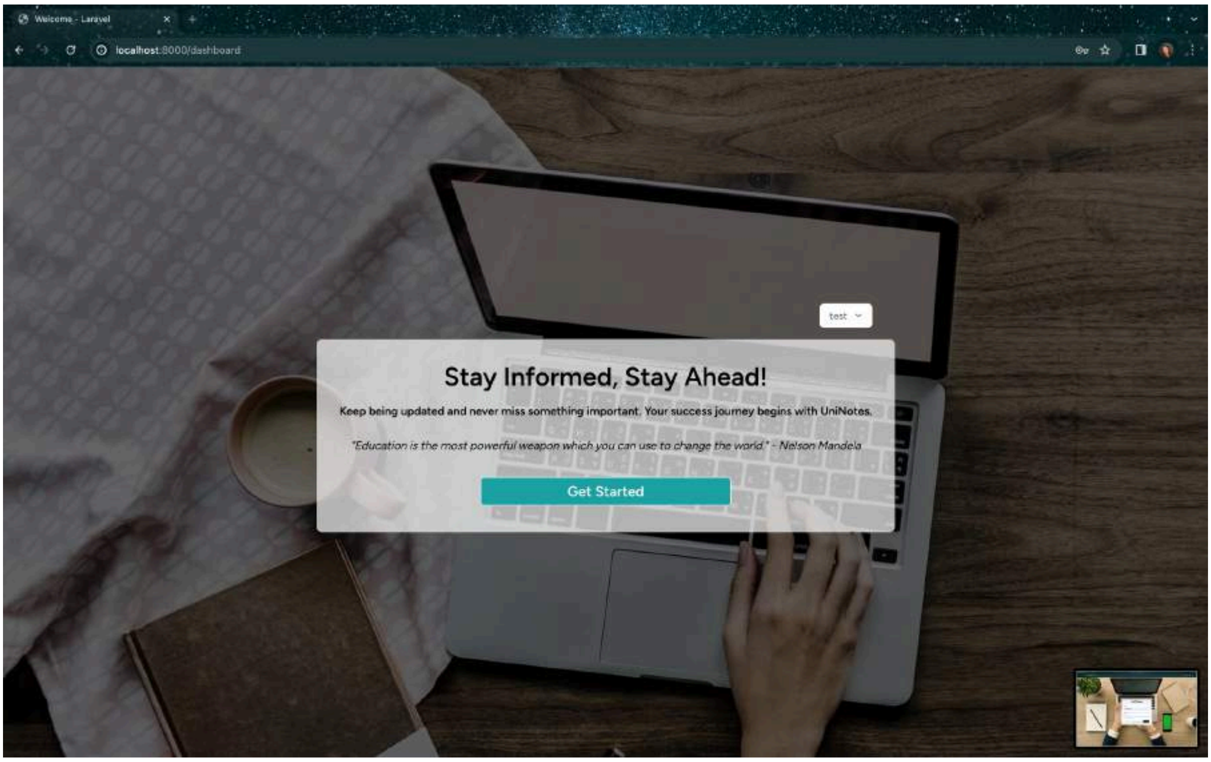


Name: test

Email: test2@abv.bg

Please fill in this field.

4. Submit the registration form by clicking on **Register** to create the profile and you will be redirected to the Home Page.



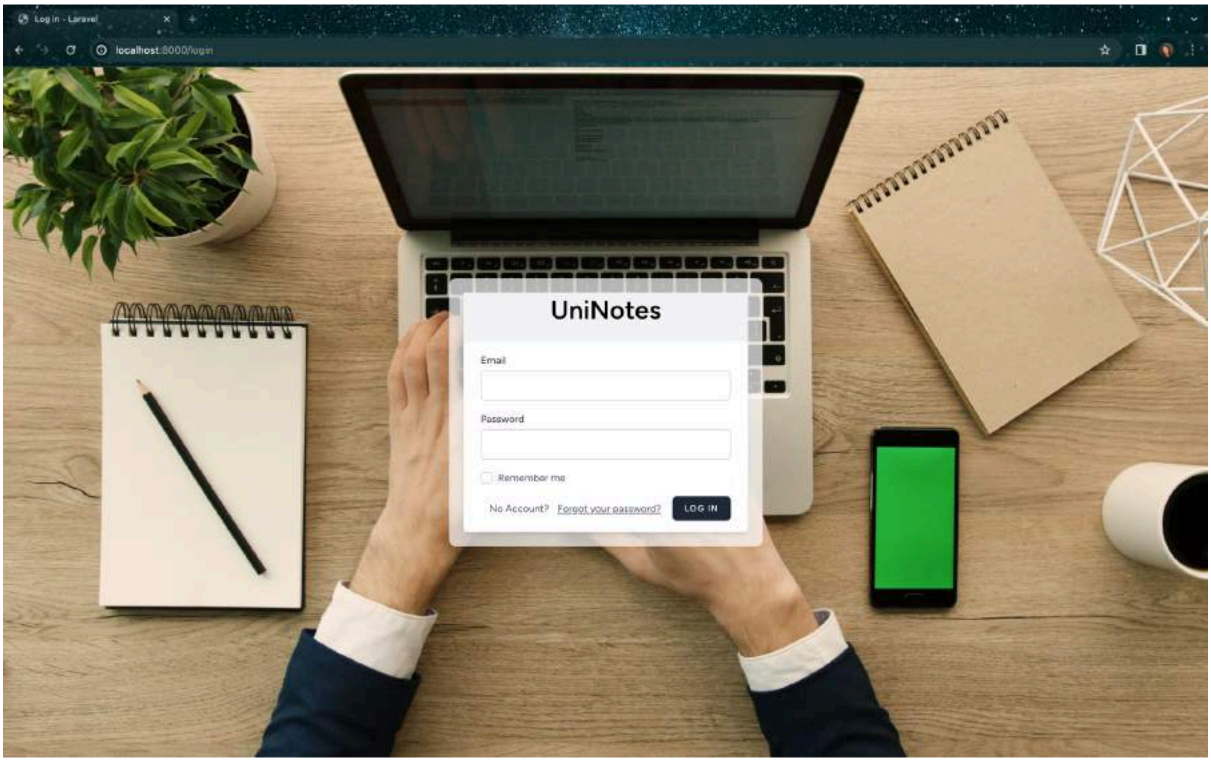
Stay Informed, Stay Ahead!

Keep being updated and never miss something important. Your success journey begins with UniNotes.

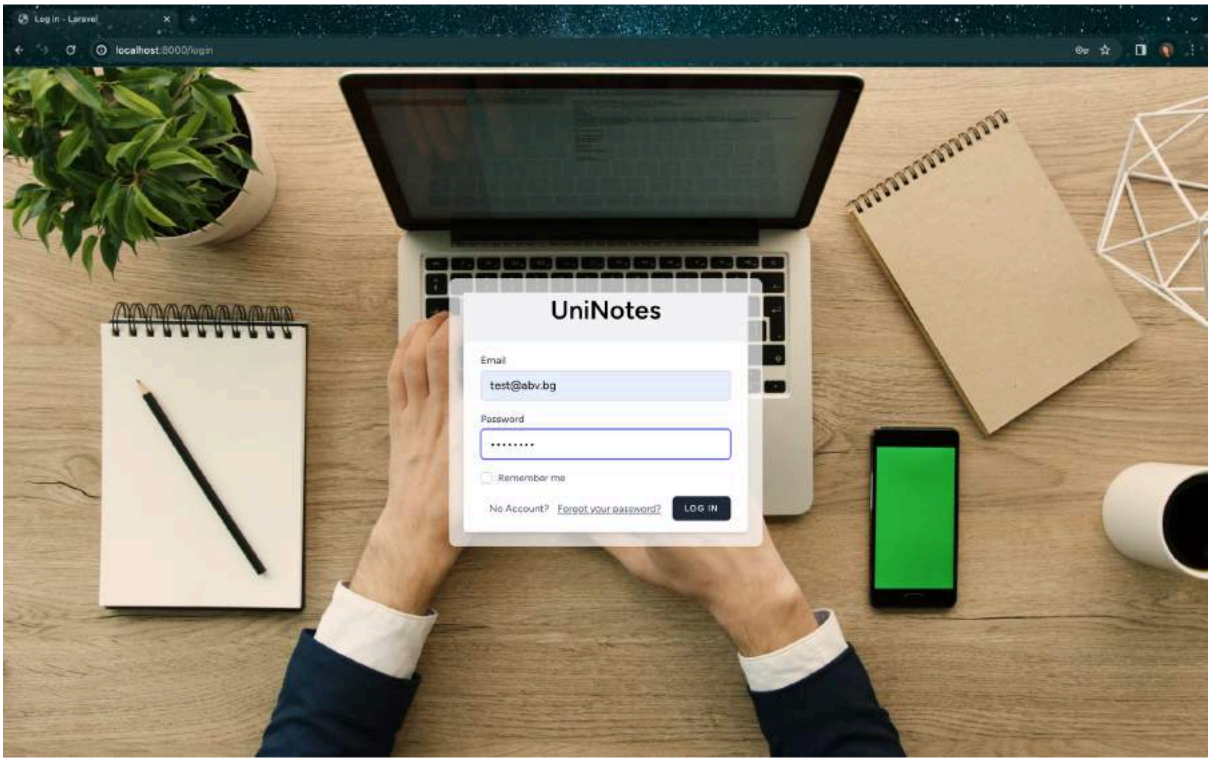
"Education is the most powerful weapon which you can use to change the world." - Nelson Mandela

User Login:

1. Navigate to UniNotes Start Page. Here you can see the Login Form.



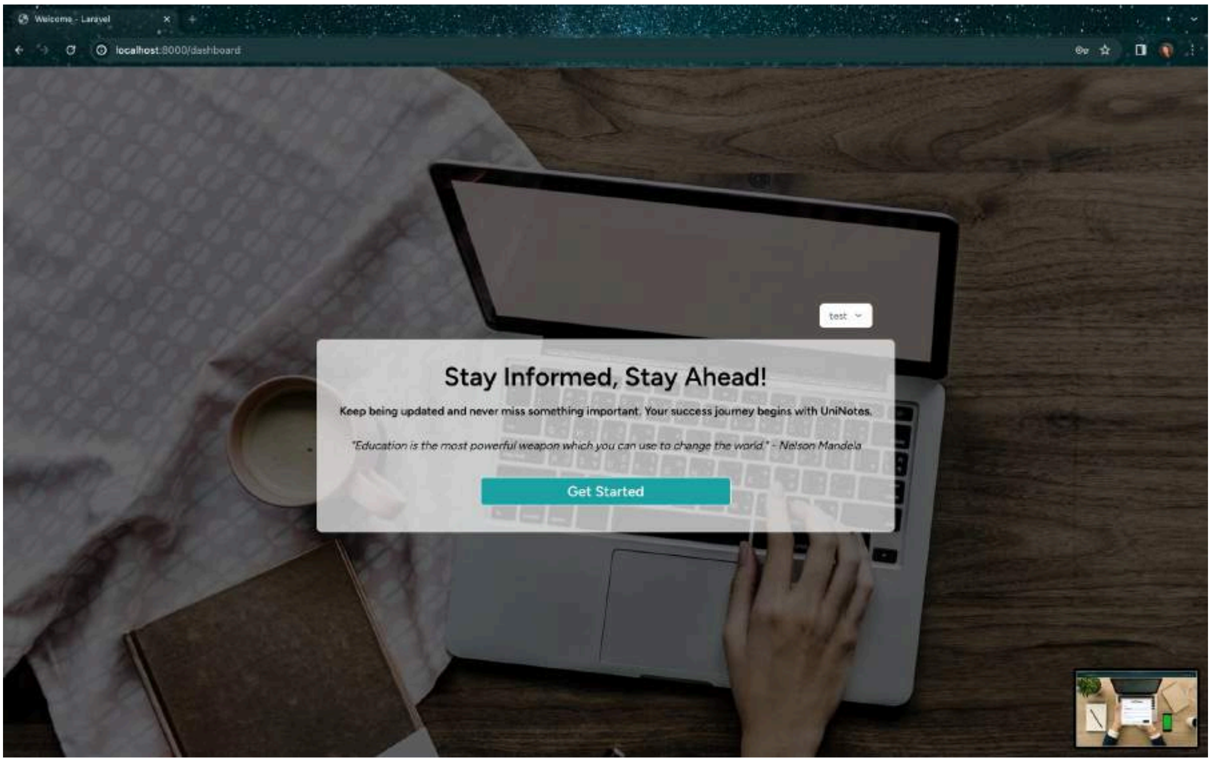
2. Enter the registered email and password. In case of a forgotten password click on **Forgot your password?** and follow the steps provided there.



Email: test@abv.bg

Password: •••••••••

3. Submit the login form and you will be redirected to the Home Page.



Stay Informed, Stay Ahead!

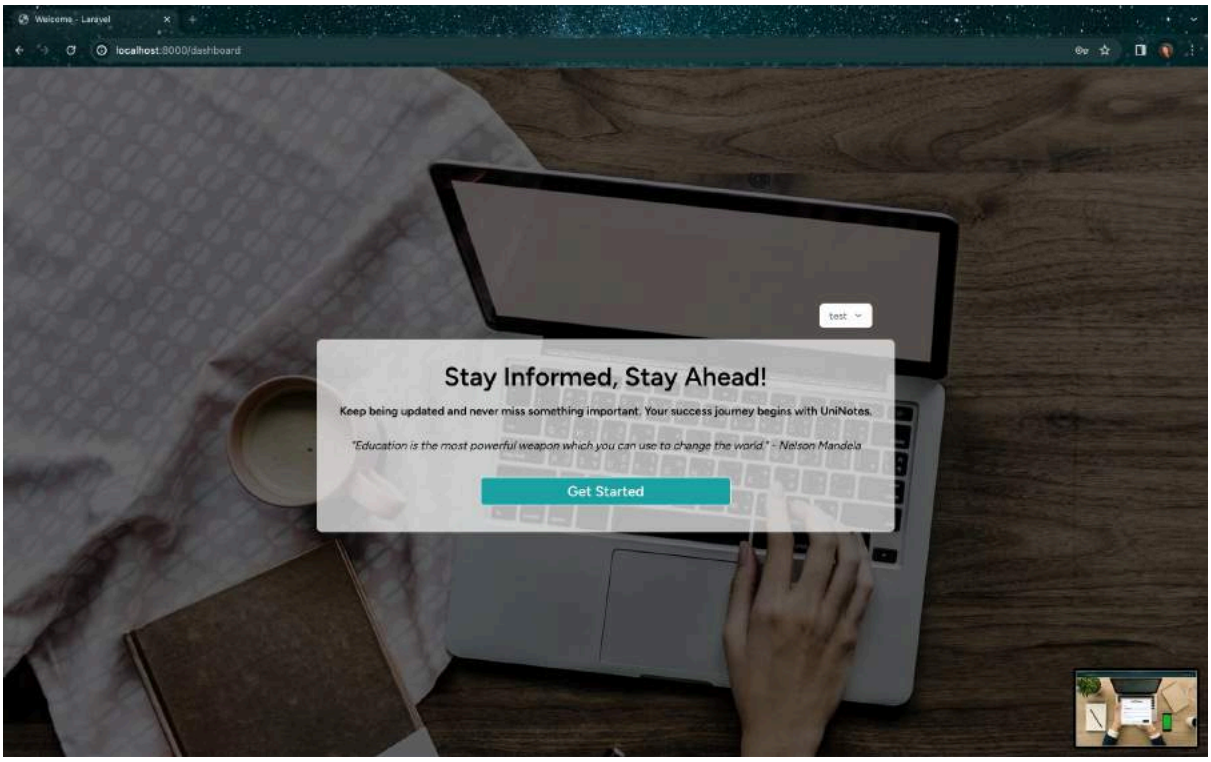
Keep being updated and never miss something important. Your success journey begins with UniNotes.

"Education is the most powerful weapon which you can use to change the world." - Nelson Mandela

Board and Note Management

Create a New Board:

1. Navigate to the Home Page. If you need help logging in, refer to instructions in the previous section.

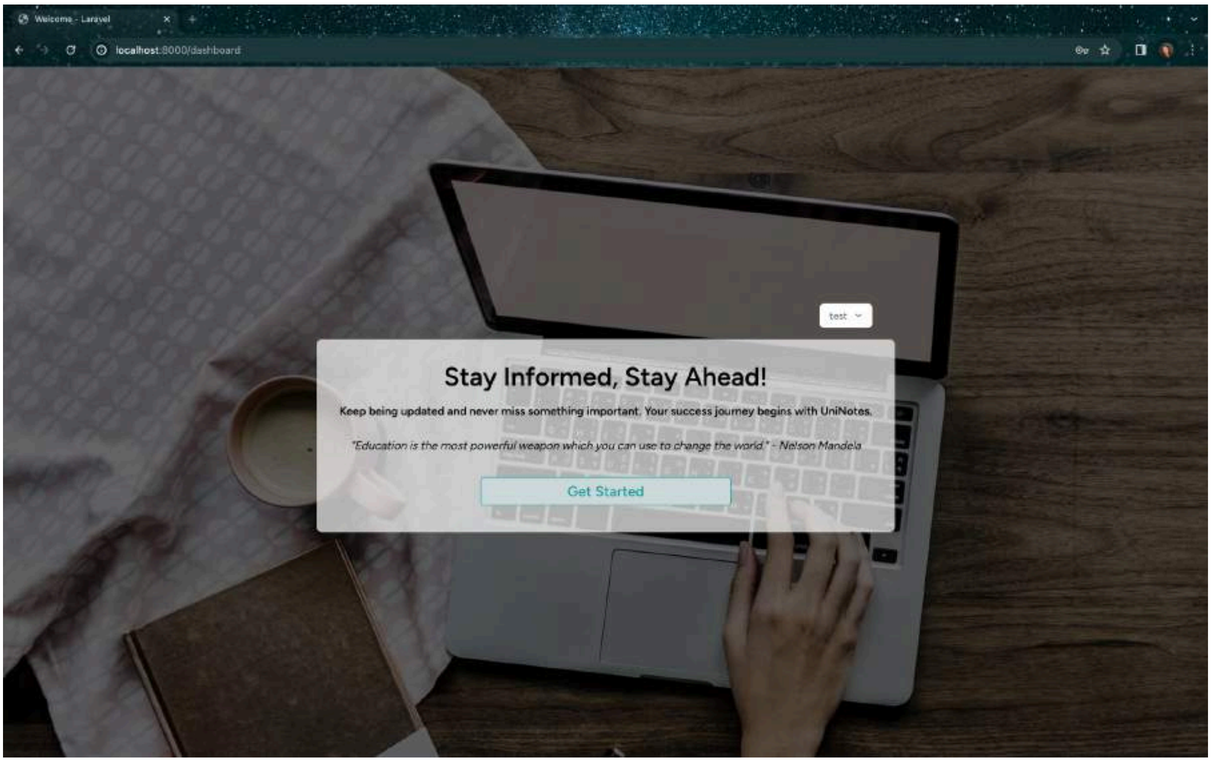


Stay Informed, Stay Ahead!

Keep being updated and never miss something important. Your success journey begins with UniNotes.

"Education is the most powerful weapon which you can use to change the world." - Nelson Mandela

2. Click on **Get Started** to navigate to the dashboard.



Welcome - Laravel

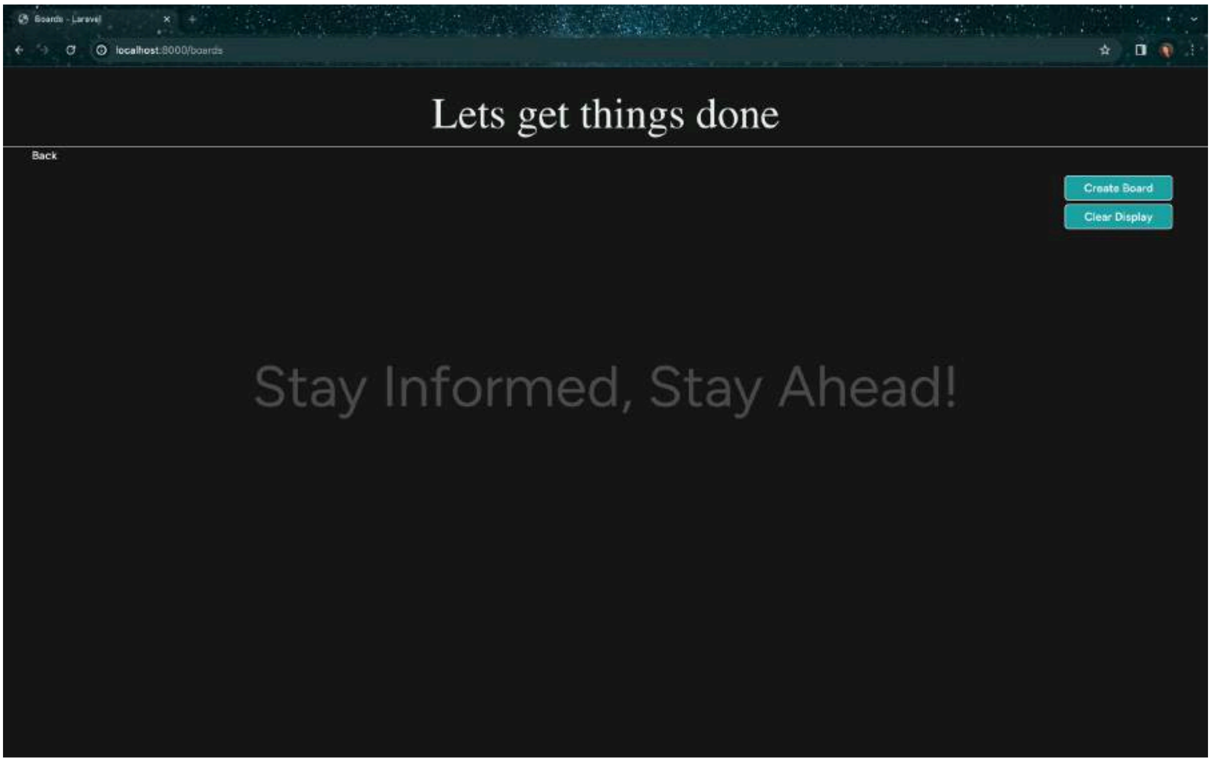
localhost:8000/dashboard

test

Stay Informed, Stay Ahead!

Keep being updated and never miss something important. Your success journey begins with UniNotes.

"Education is the most powerful weapon which you can use to change the world." - Nelson Mandela



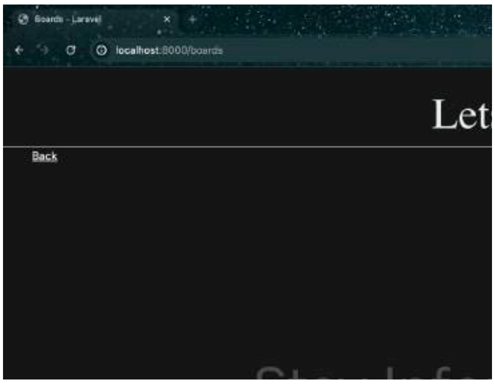
Boards - Laravel

localhost:8000/boards

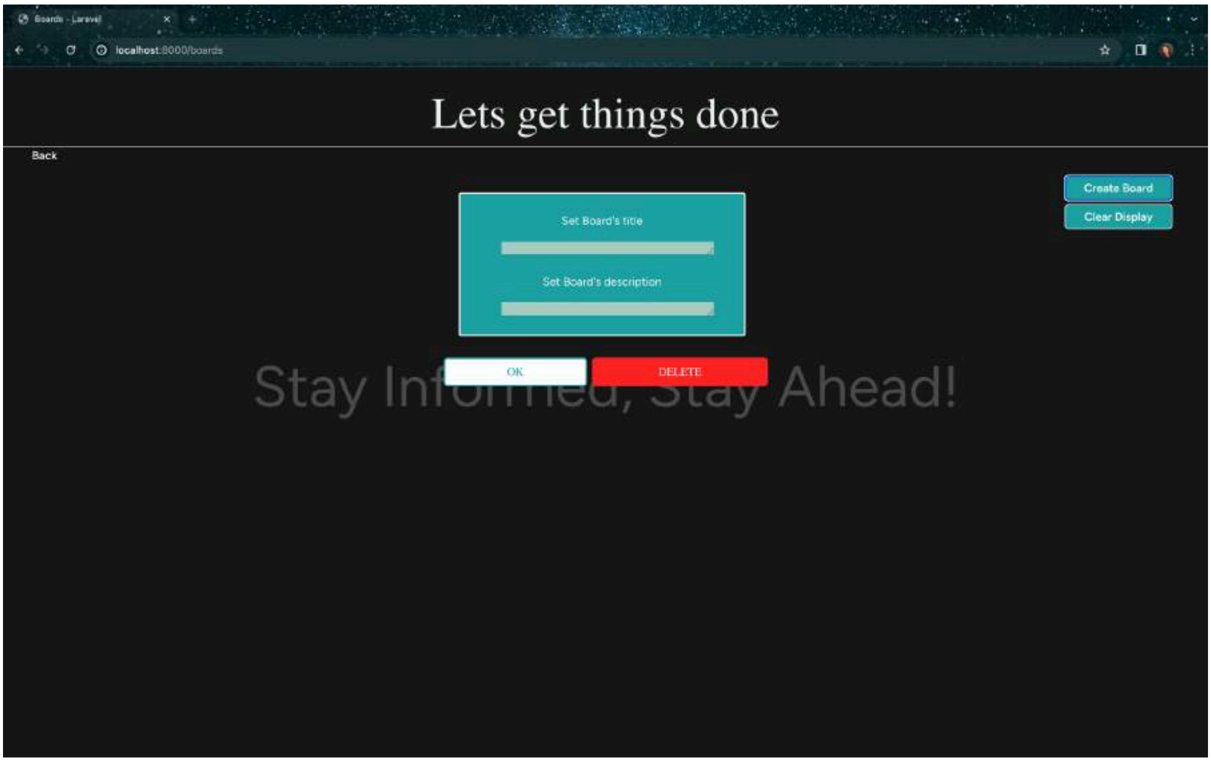
Lets get things done

Stay Informed, Stay Ahead!

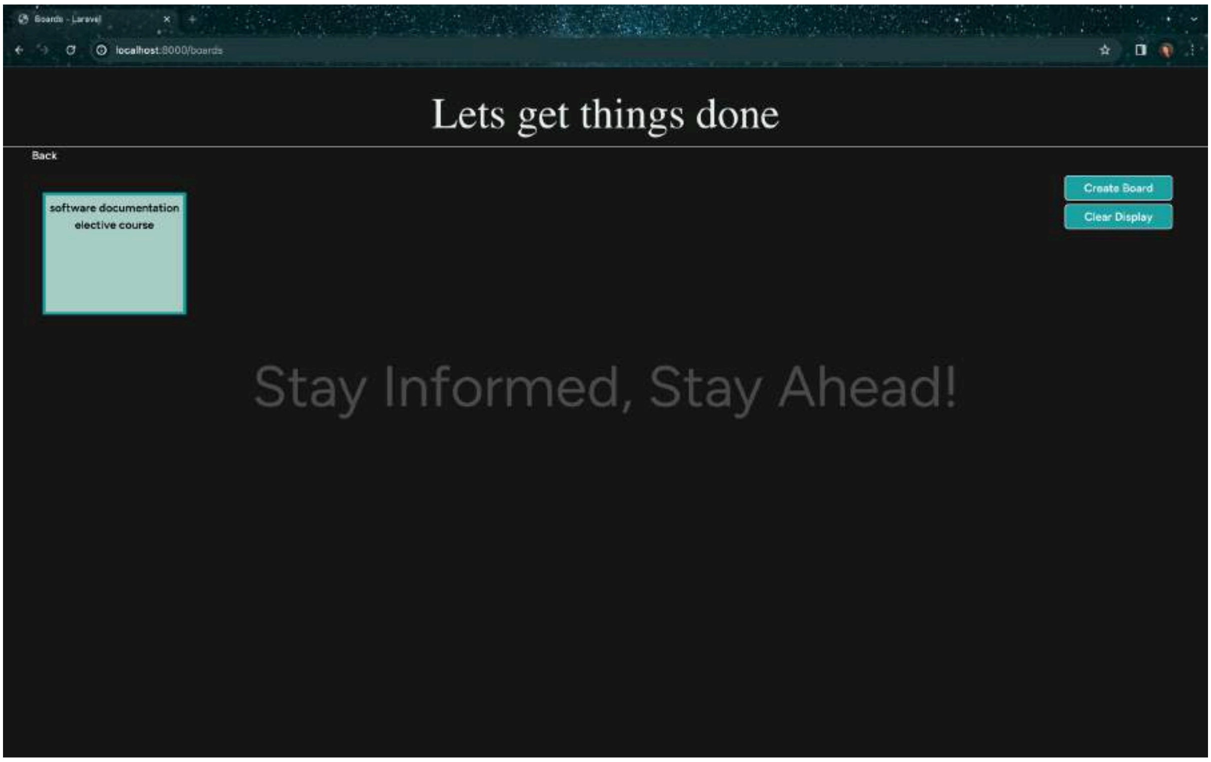
3. To go back to the Home Page click on **Back** in the top left corner of the screen.



4. To create a new board click on **Create Board** and add a title and description for it.



5. Click on **OK** and it should appear on the Dashboard.

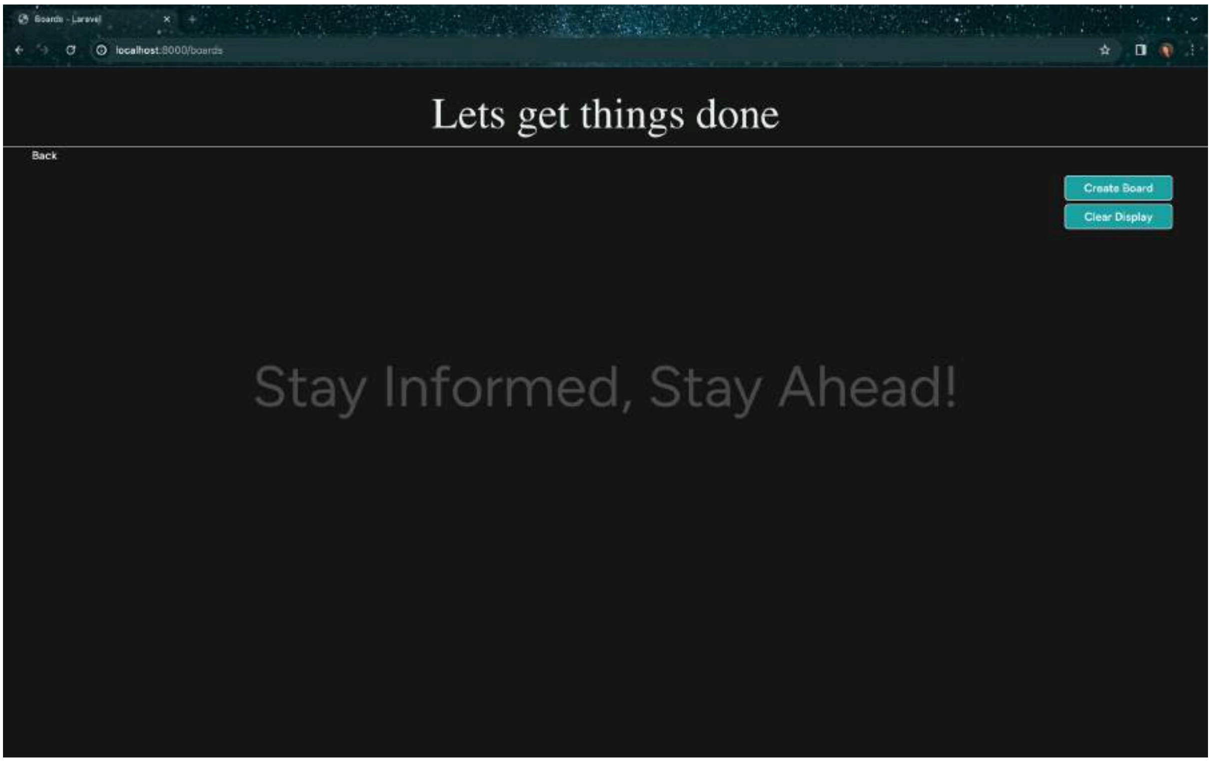


Lets get things done

software documentation elective course

Stay Informed, Stay Ahead!

6. If you want to delete the current boards on the Dashboard click on **Clear Display** and the Dashboard should be clear.

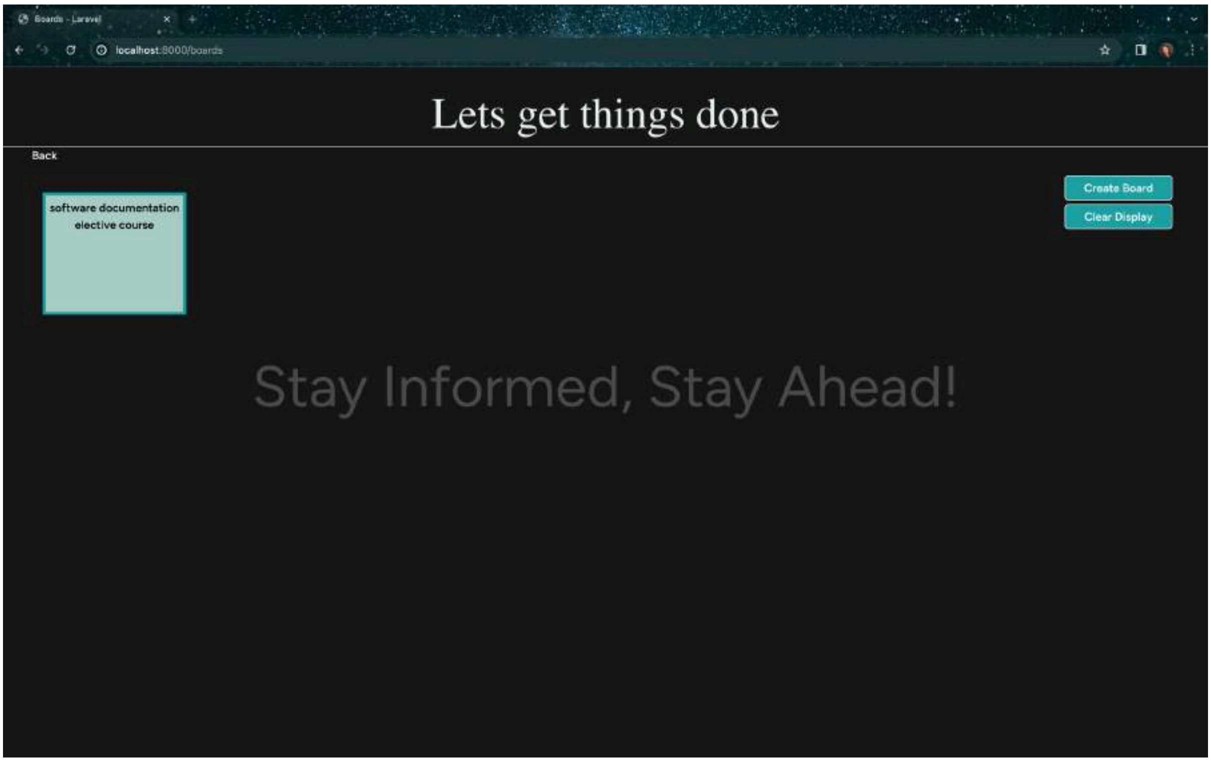


Lets get things done

Stay Informed, Stay Ahead!

Edit Board:

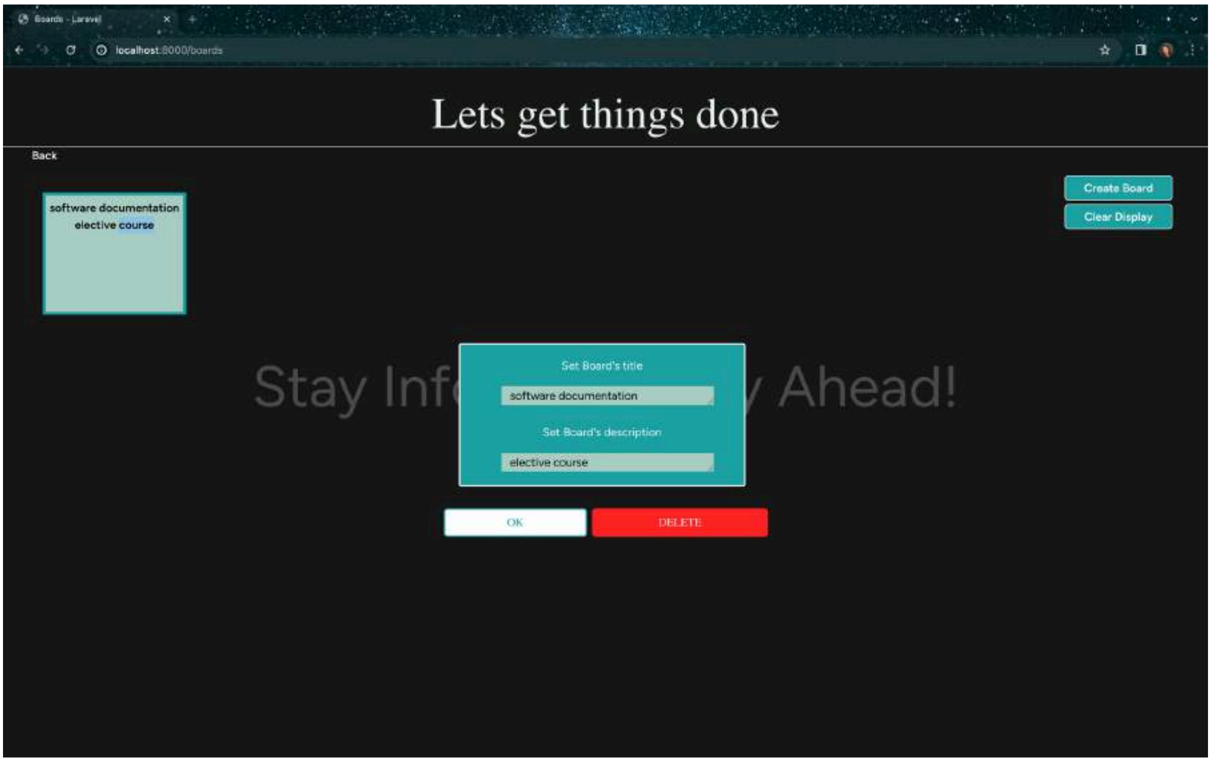
1. To edit a board navigate to the Dashboard.



software documentation
elective course

Stay Informed, Stay Ahead!

2. Double click on the desired board to be able to edit it.



software documentation
elective course

Stay Informed, Stay Ahead!

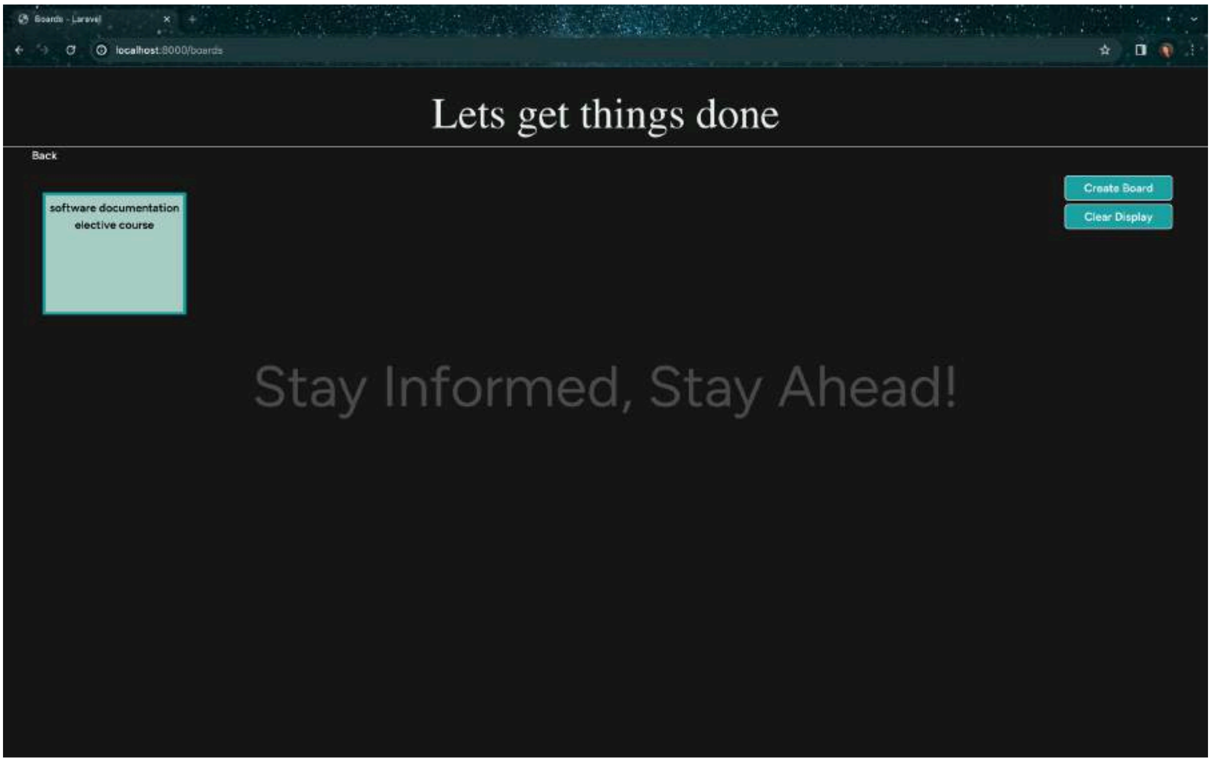
Set Board's title

software documentation

Set Board's description

elective course

3. After completing your desired edits click **OK** and the edits should be up to date.



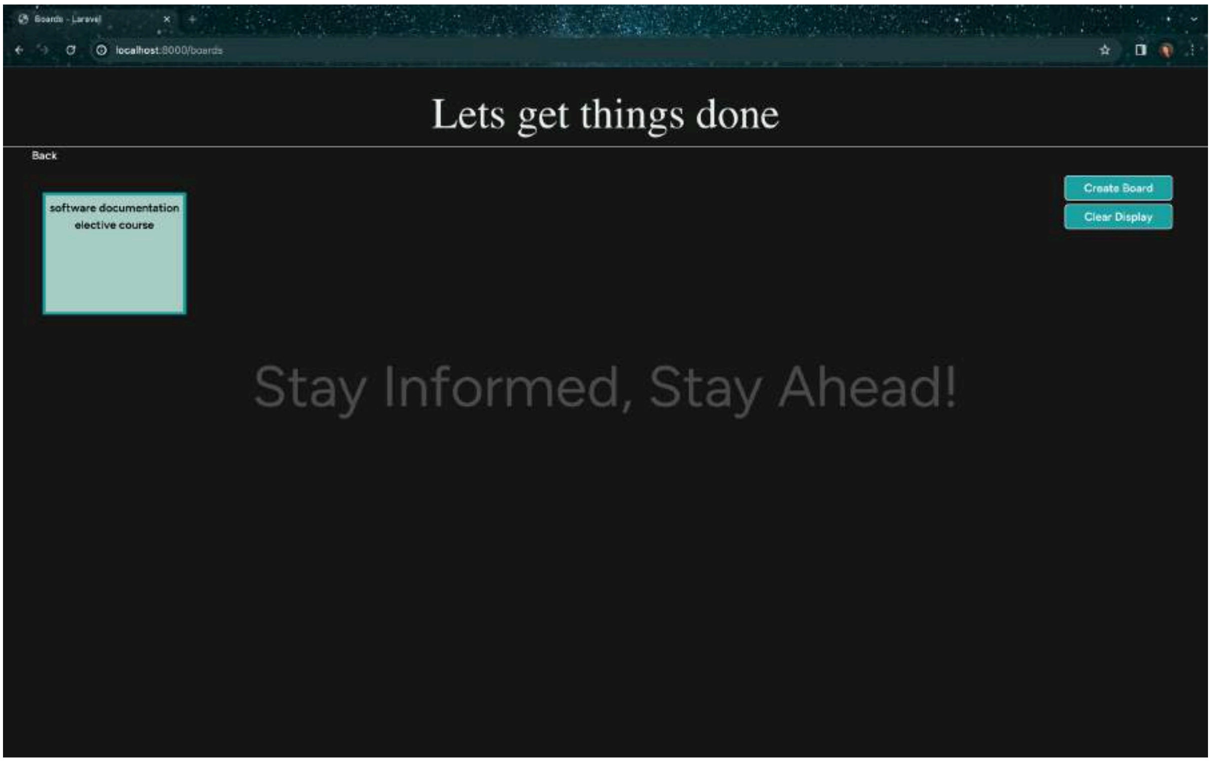
Lets get things done

software documentation elective course

Stay Informed, Stay Ahead!

Add Note to a Board:

1. Navigate to the Dashboard.

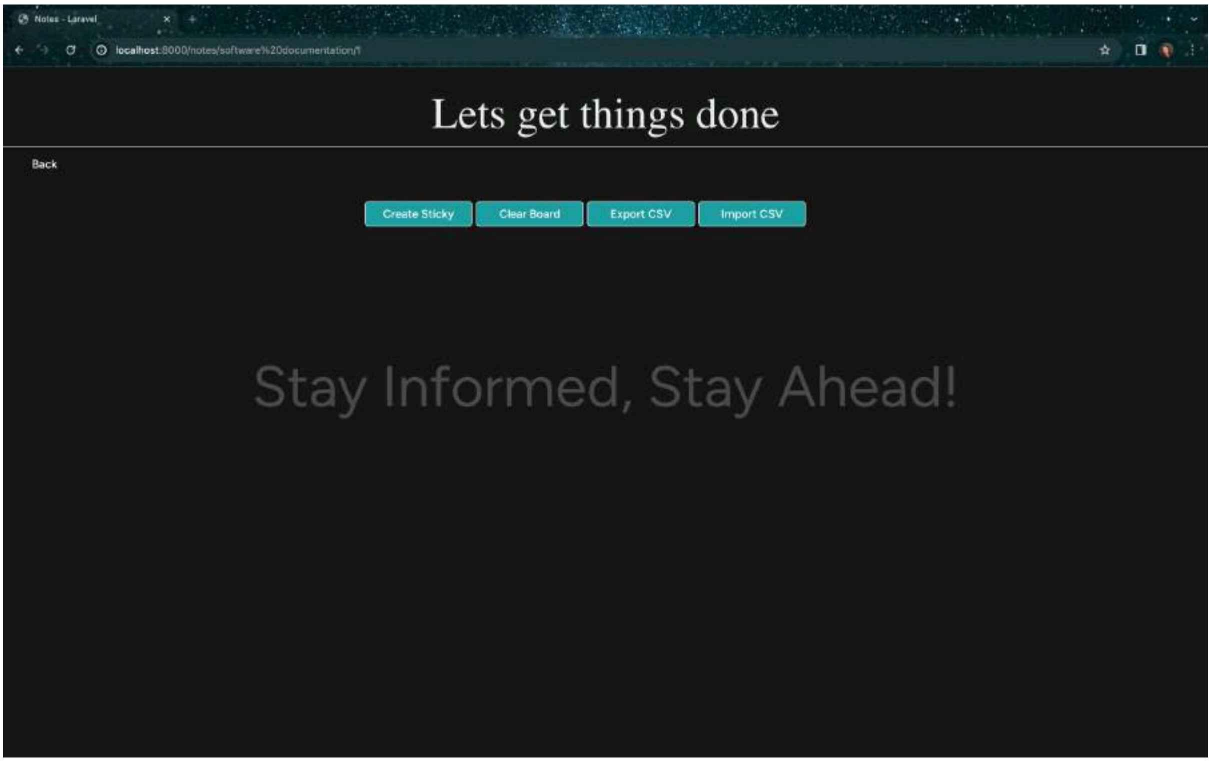


Lets get things done

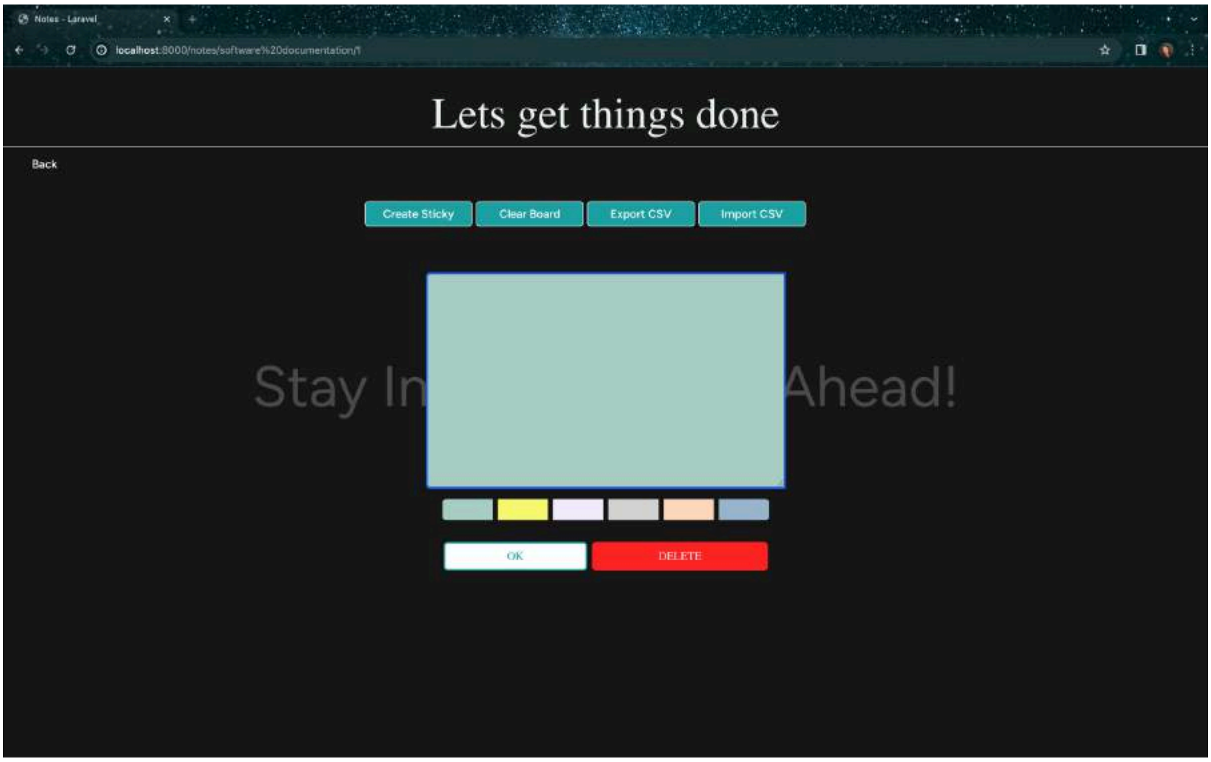
software documentation elective course

Stay Informed, Stay Ahead!

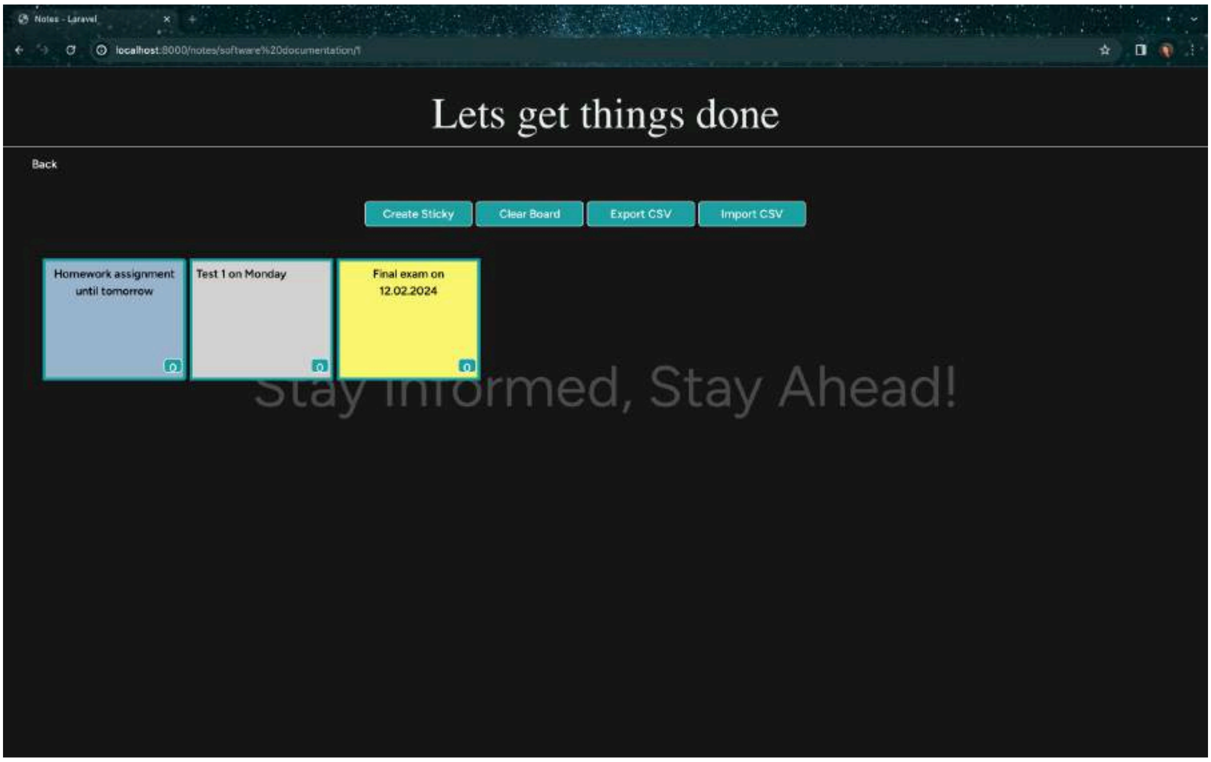
2. Select an existing board.



3. Click on **Create Sticky** to add a note to the board.



4. Fill in desired text and select the desired color. Afterwards click **OK** and the note will be added to the board.



Lets get things done

Homework assignment until tomorrow

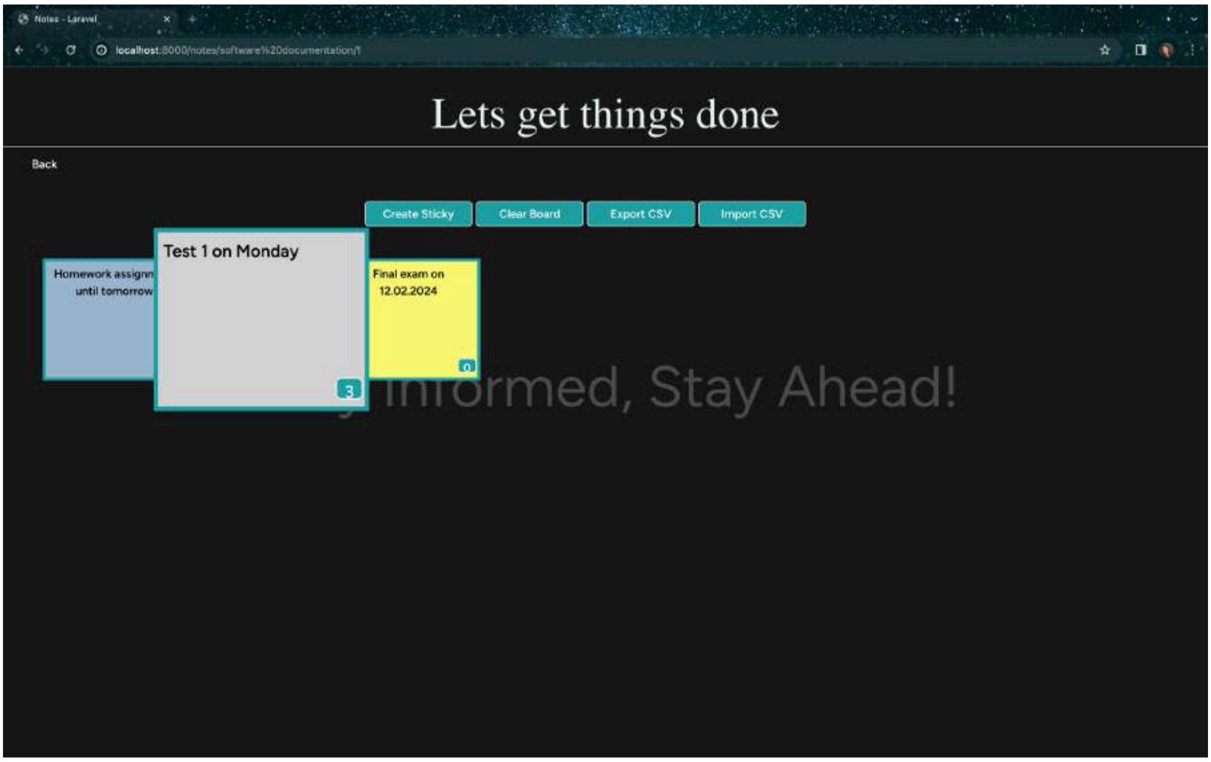
Test 1 on Monday

Final exam on 12.02.2024

Stay informed, Stay Ahead!

Edit Note:

1. Navigate to the desired board. Hover over the note you want to edit.



Lets get things done

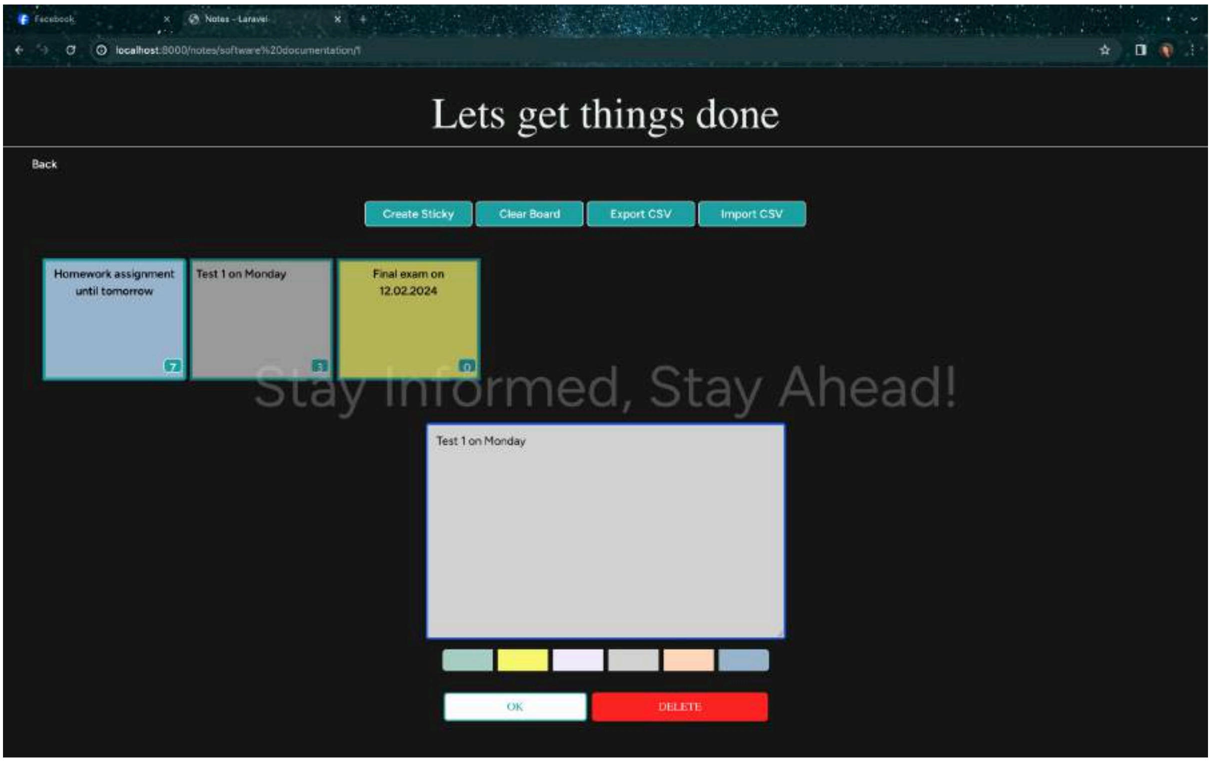
Homework assignm until tomorrow

Test 1 on Monday

Final exam on 12.02.2024

informed, Stay Ahead!

2. Select the note to edit by hovering over it and <u>double clicking</u> on it.



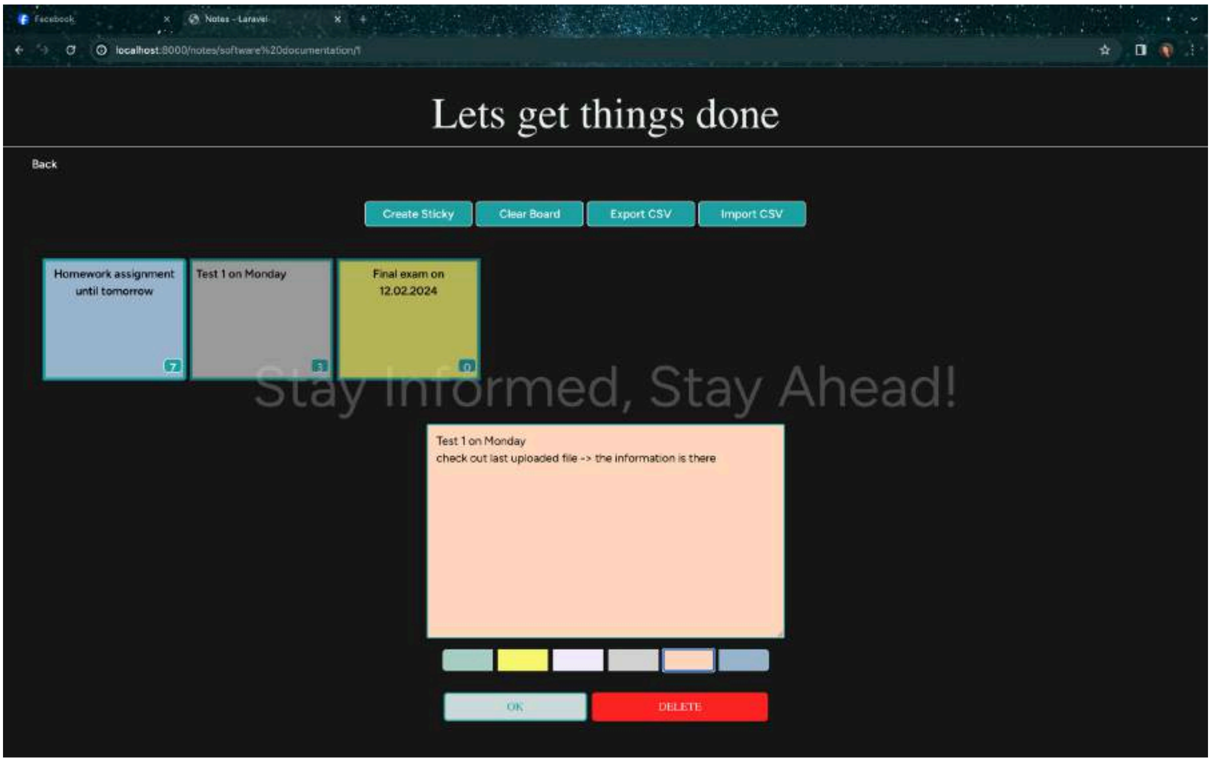
Homework assignment until tomorrow

Test 1 on Monday

Final exam on 12.02.2024

Test 1 on Monday

3. Update the note information form and/or color to your liking.



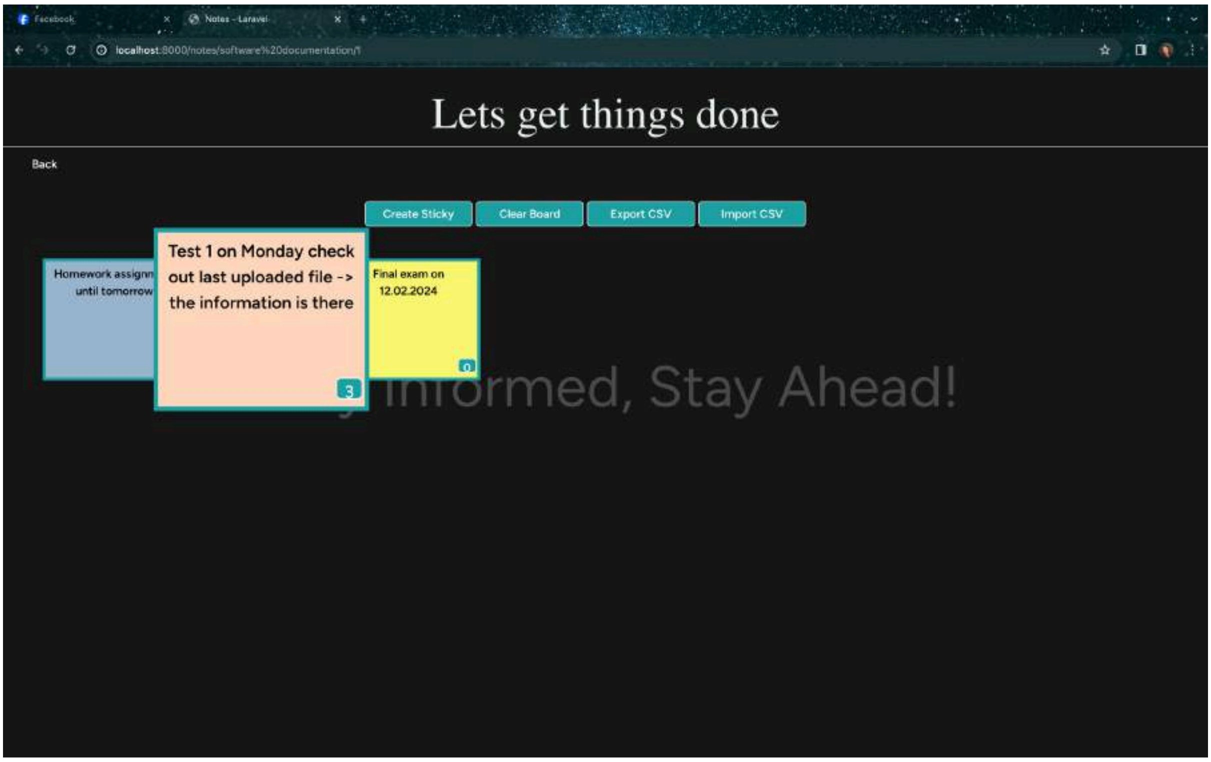
Homework assignment until tomorrow

Test 1 on Monday

Final exam on 12.02.2024

Test 1 on Monday
check out last uploaded file -> the information is there

4. Click on **OK**.



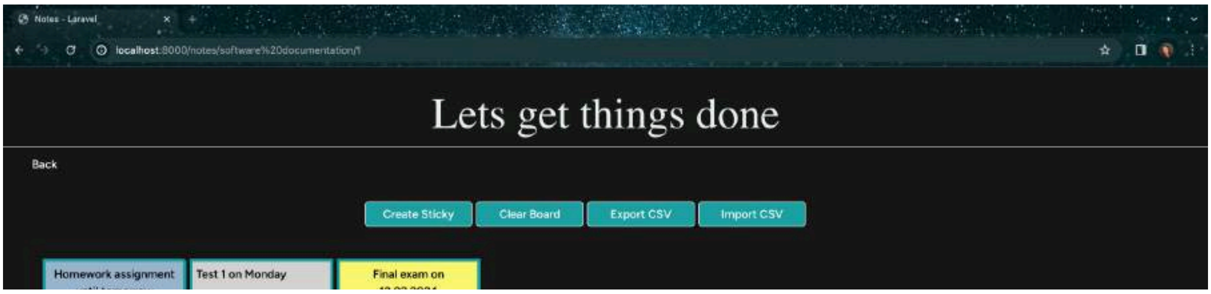
Homework assignm
until tomorrow

Test 1 on Monday check out last uploaded file -> the information is there

Final exam on 12.02.2024

Clear Board:

1. To delete all notes from a board navigate to the desired board.

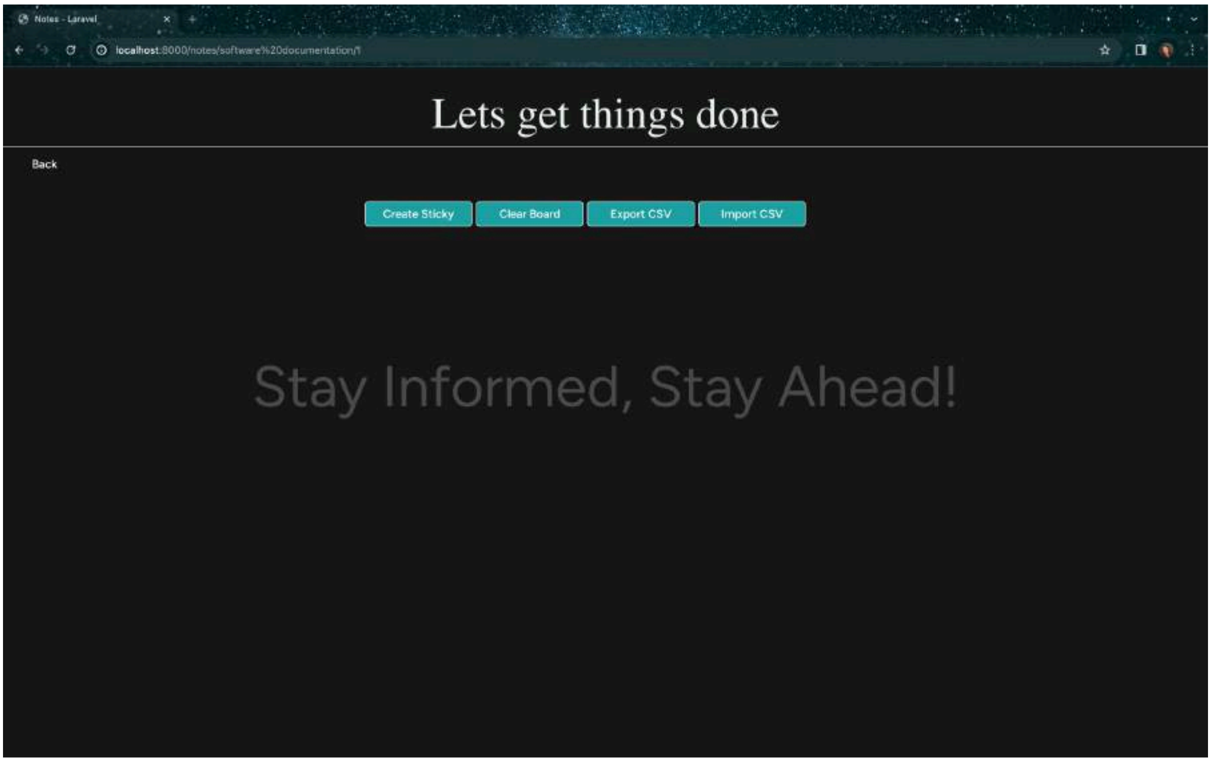


Homework assignment

Test 1 on Monday

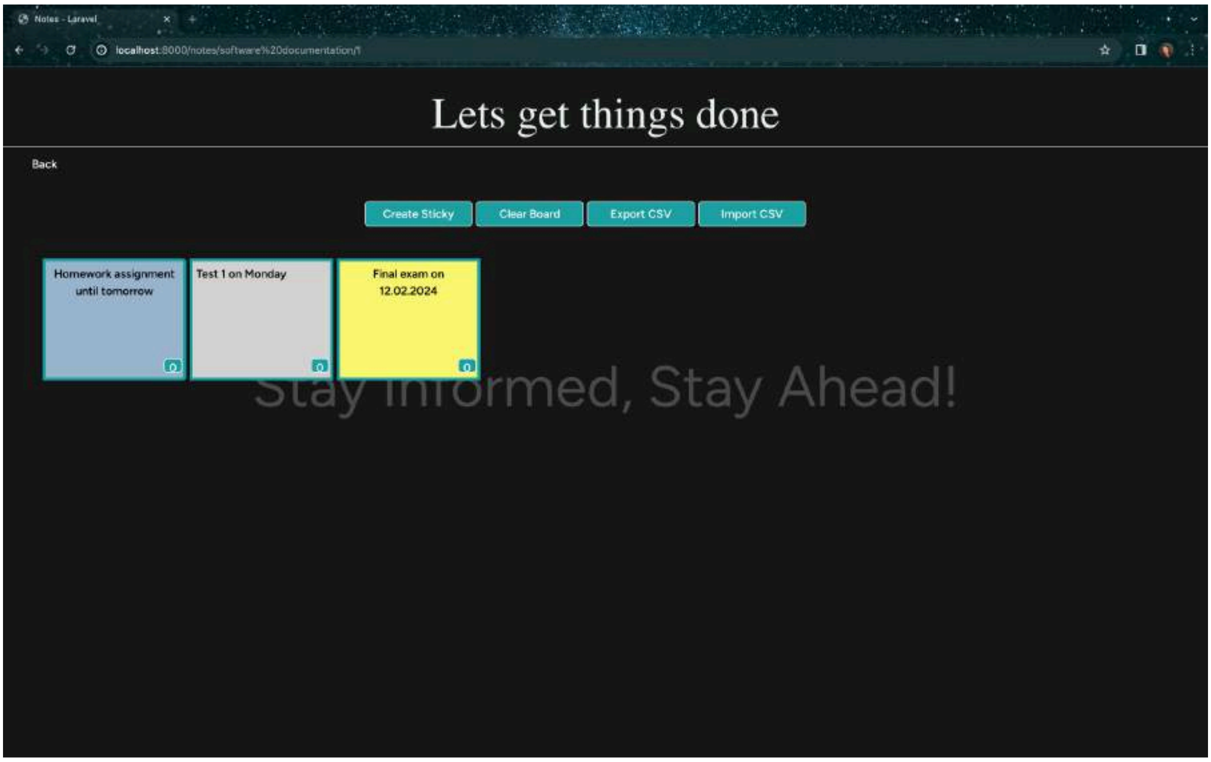
Final exam on

2. Click on **Clear Board.** This should clear all notes from the board.



Upvote Note:

1. To upvote a note hover over it.

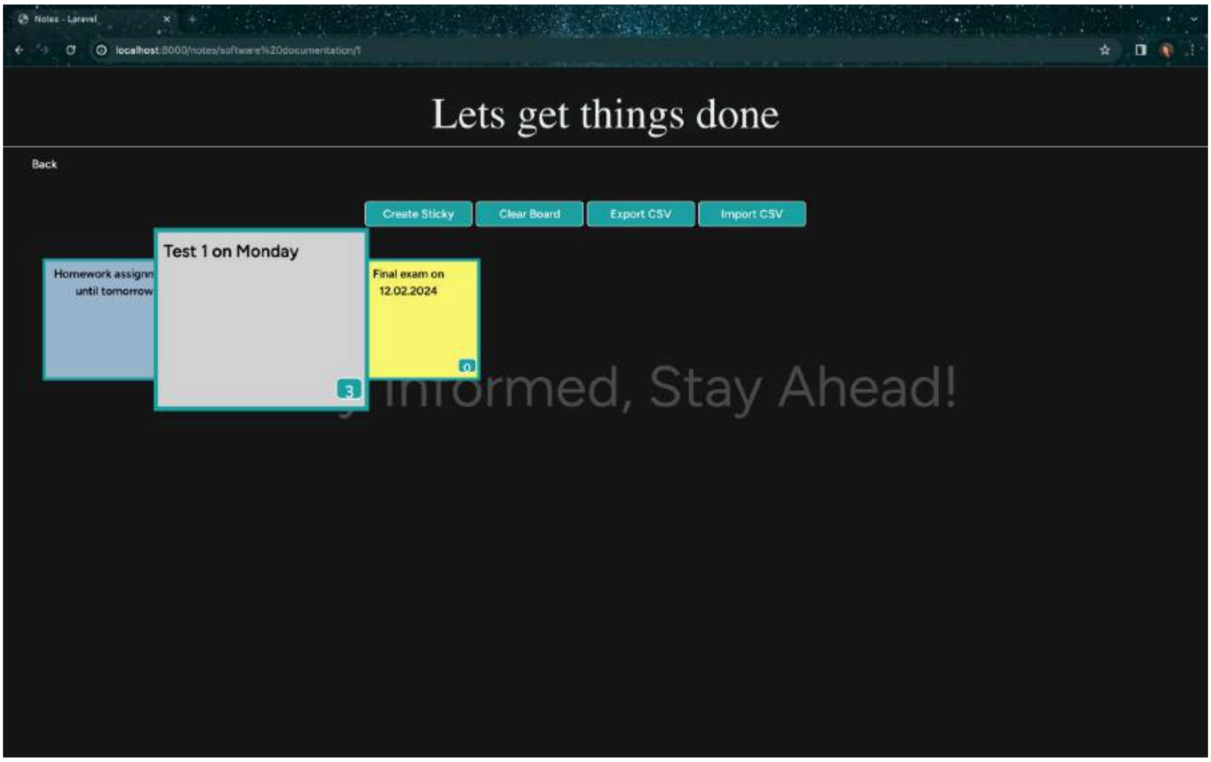


Homework assignment until tomorrow

Test 1 on Monday

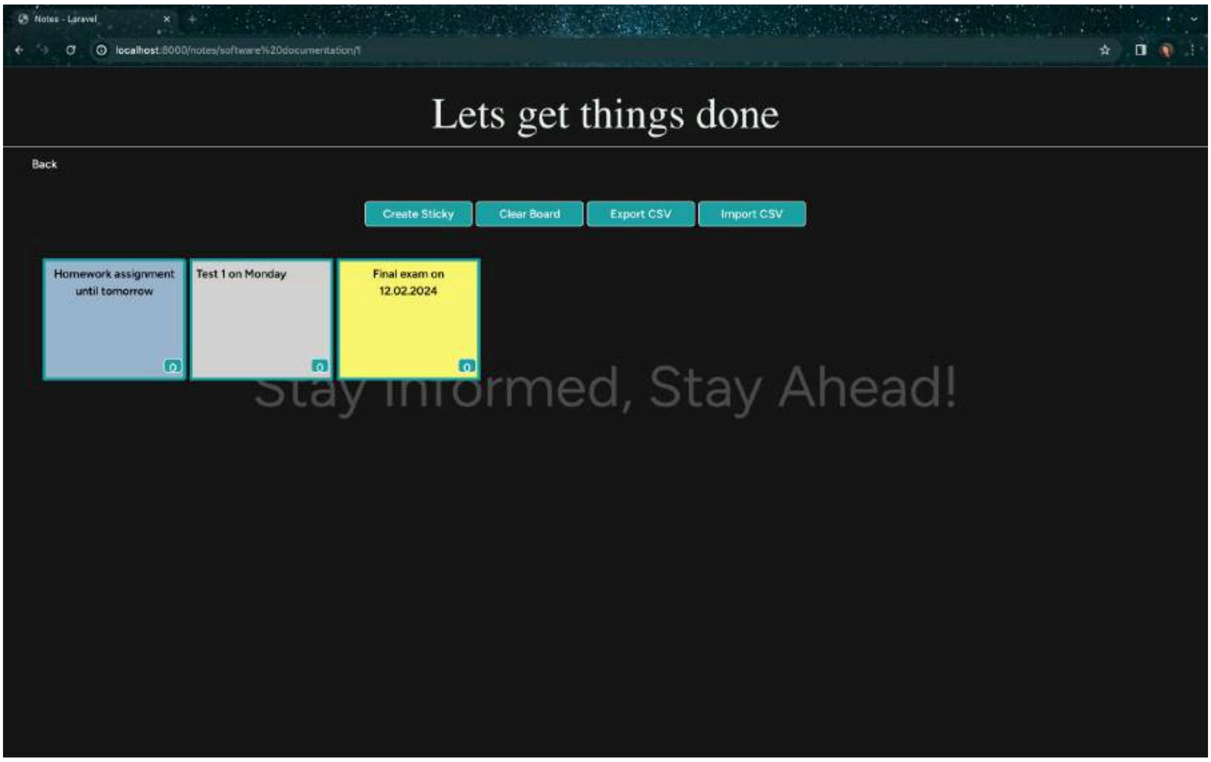
Final exam on 12.02.2024

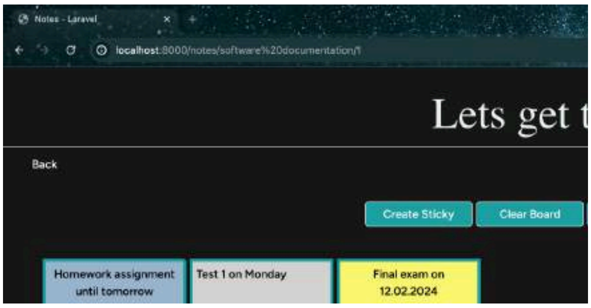
2. Click on the number in the bottom right corner of the note.



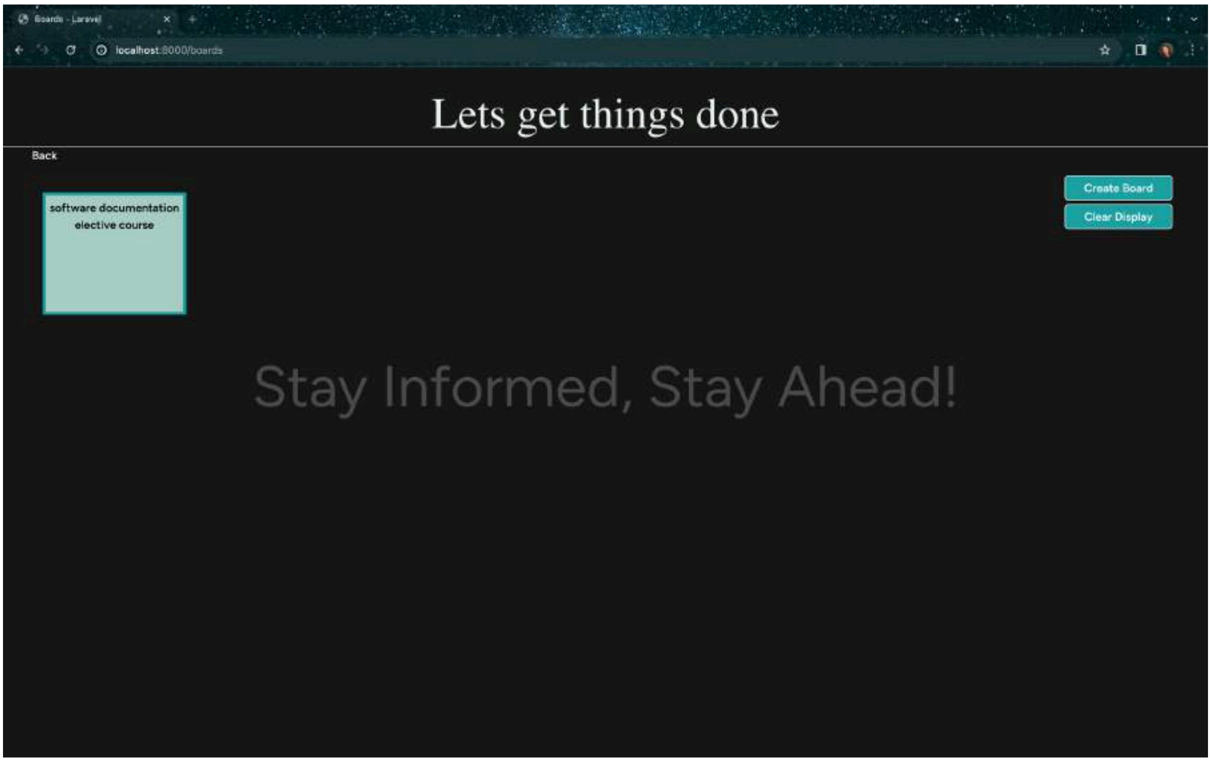
Board Navigation:

1. To switch to a different board on the dashboard, while being in one, click on **Back** in the top left corner.





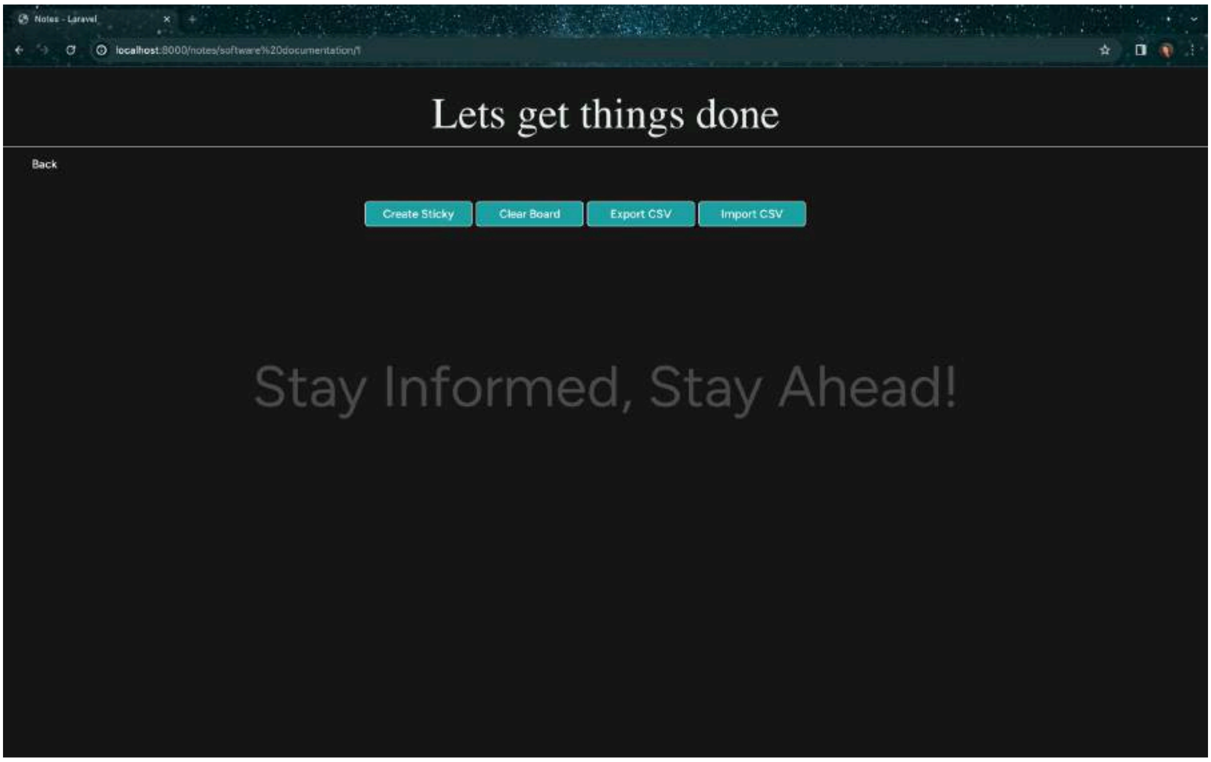
2. You will be redirected back to the Dashboard.



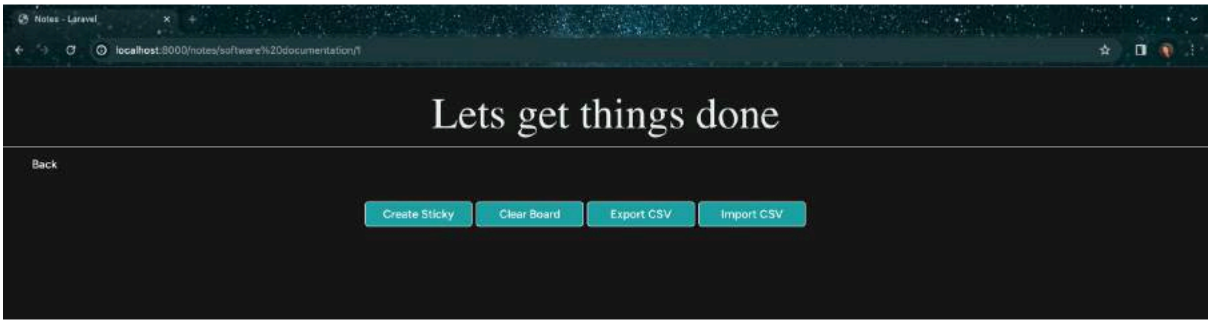
Importing and Exporting Notes:

Import Notes to a Board:

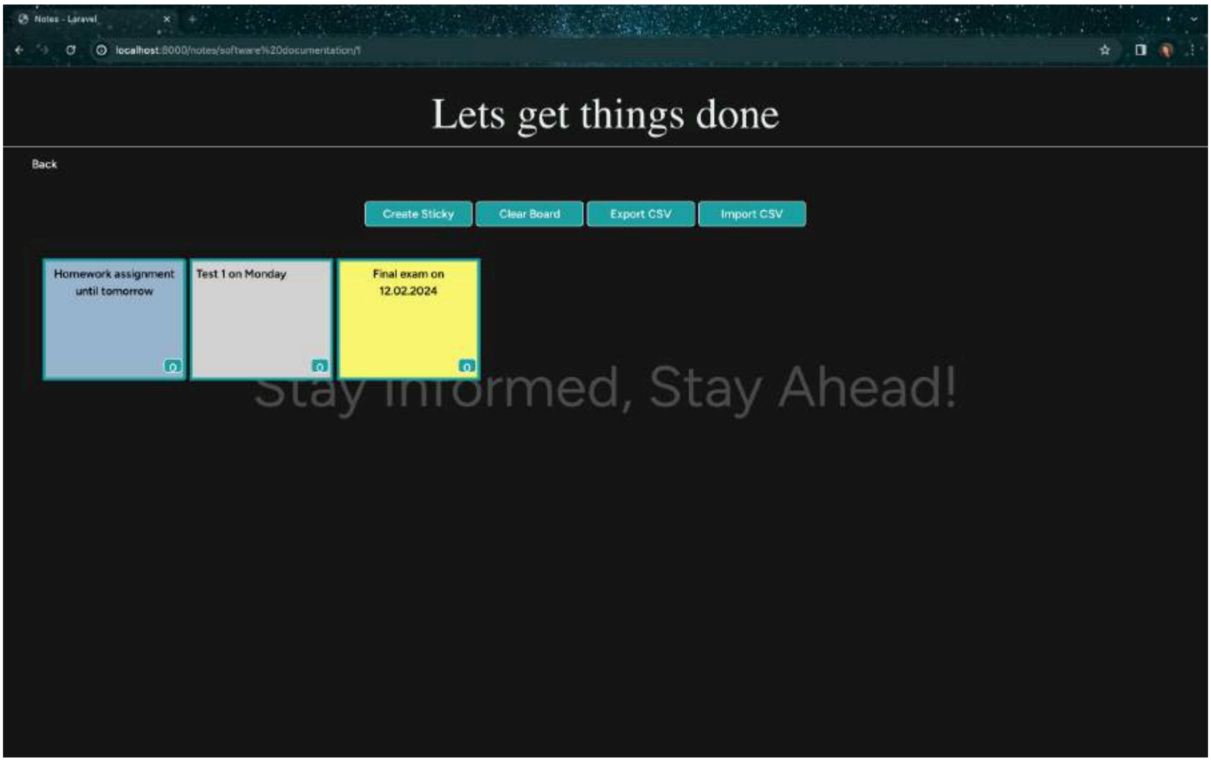
1. Navigate to the board.



2. Click on **Import CSV** button.

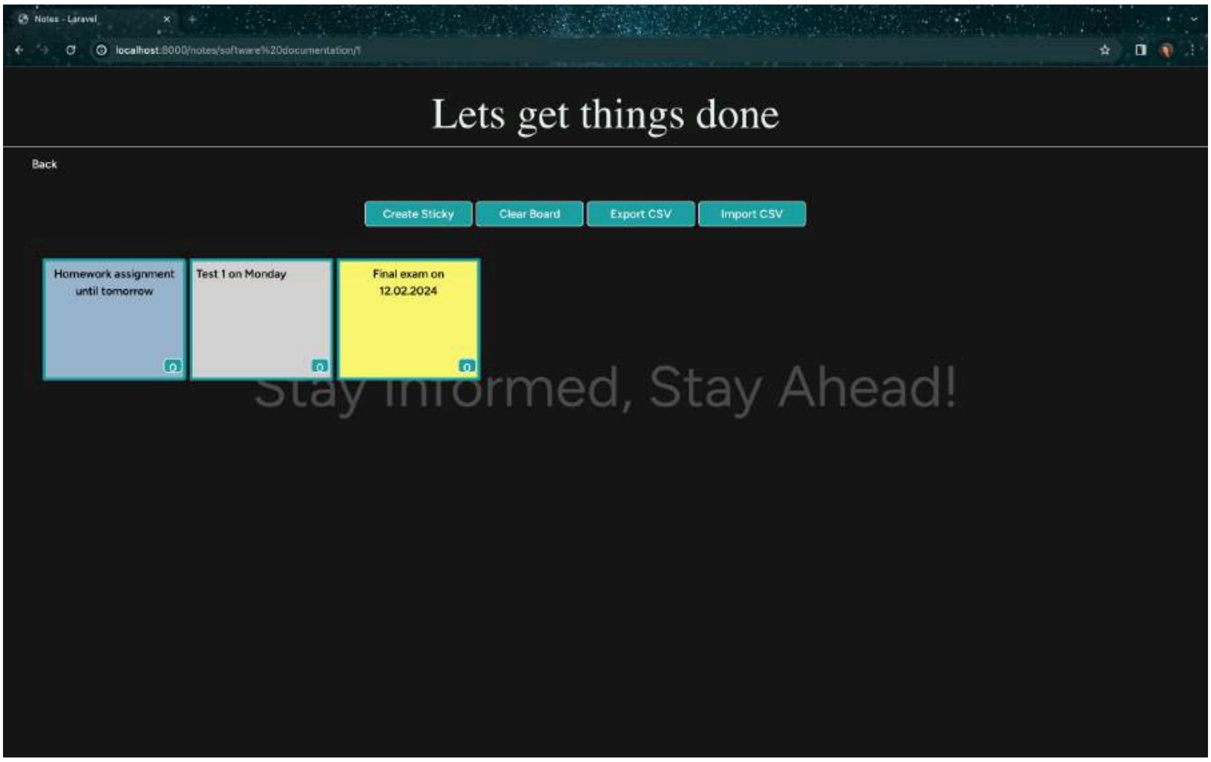


3. Choose a CSV file with notes data. The notes should be imported into the board.

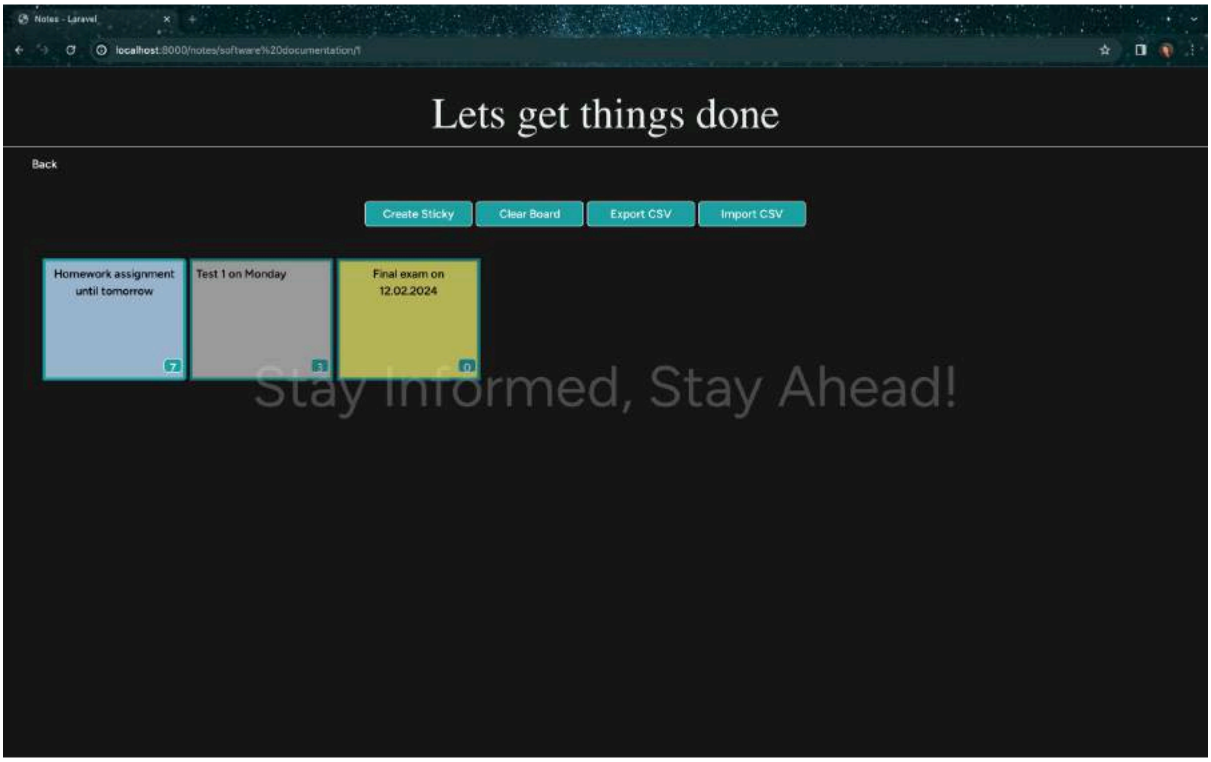


Export Notes from a Board:

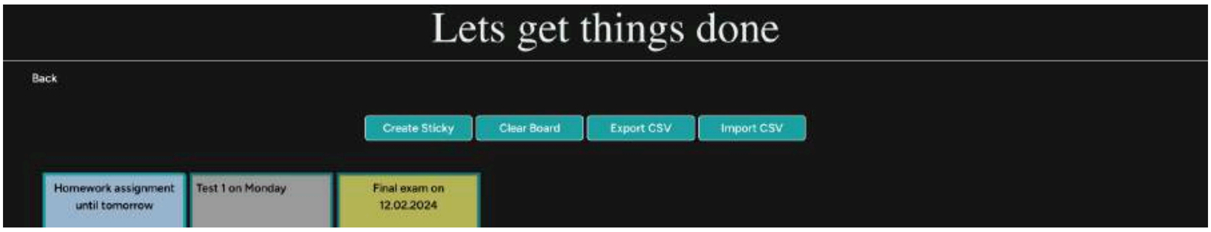
1. Navigate to the board.



2. Select the notes you wish to export by clicking on them once. They should appear with a shadow when selected.



3. Click on **Export CSV.**

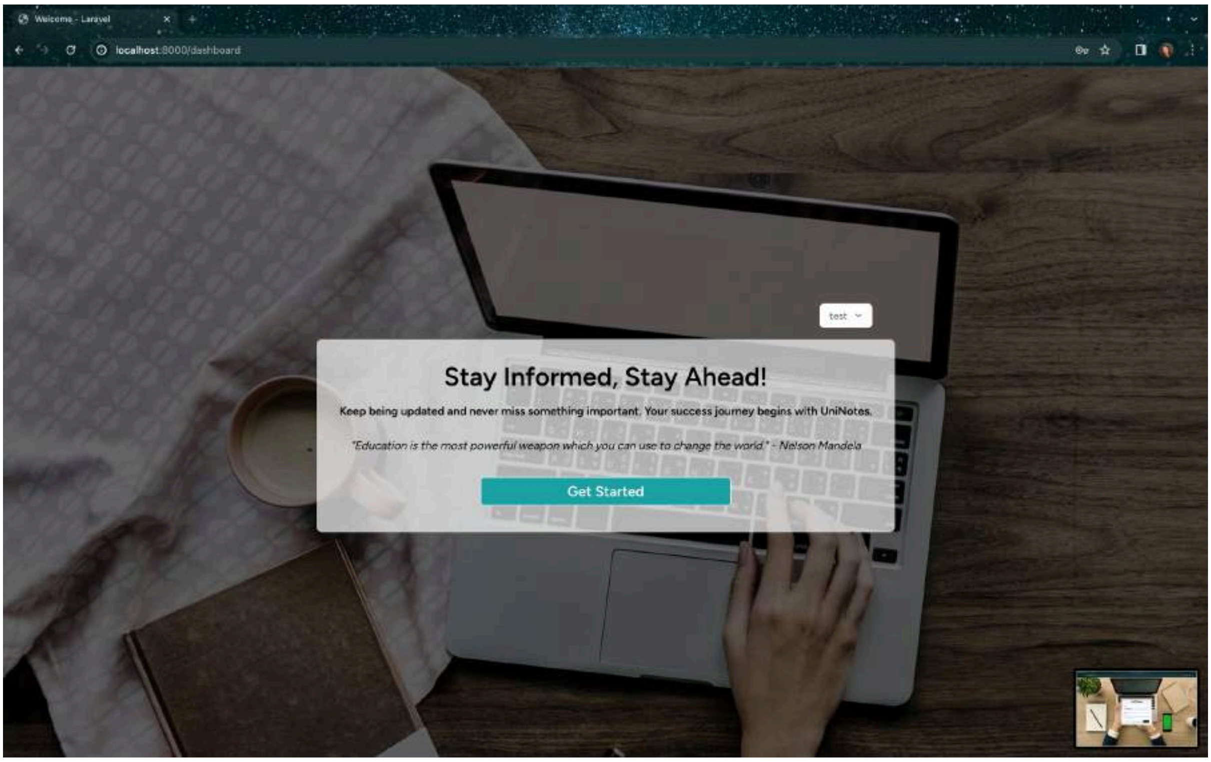


4. Export notes from the chosen board to a CSV file.

User Profile Management:

Modify User Profile:

1. Login in to your profile to be navigated to the Home Page.

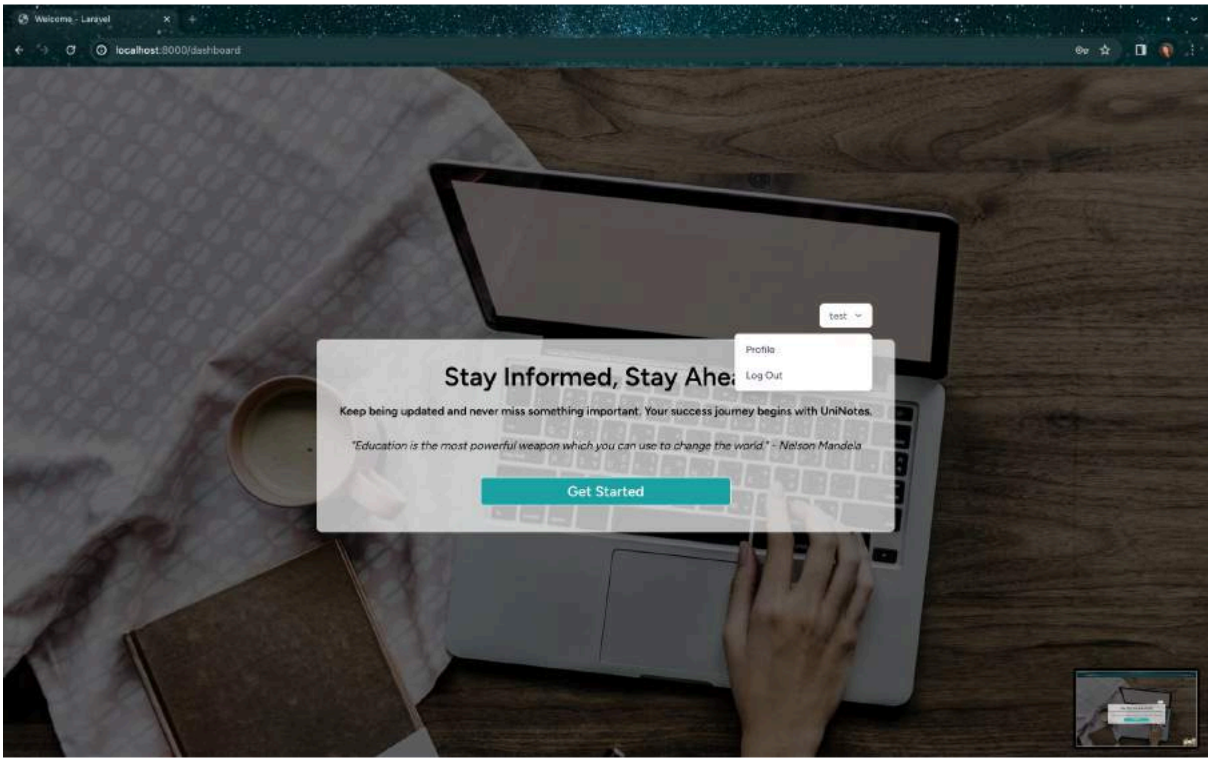


Stay Informed, Stay Ahead!

Keep being updated and never miss something important. Your success journey begins with UniNotes.

"Education is the most powerful weapon which you can use to change the world." - Nelson Mandela

2. Click on the button with your name to open the dropdown menu.

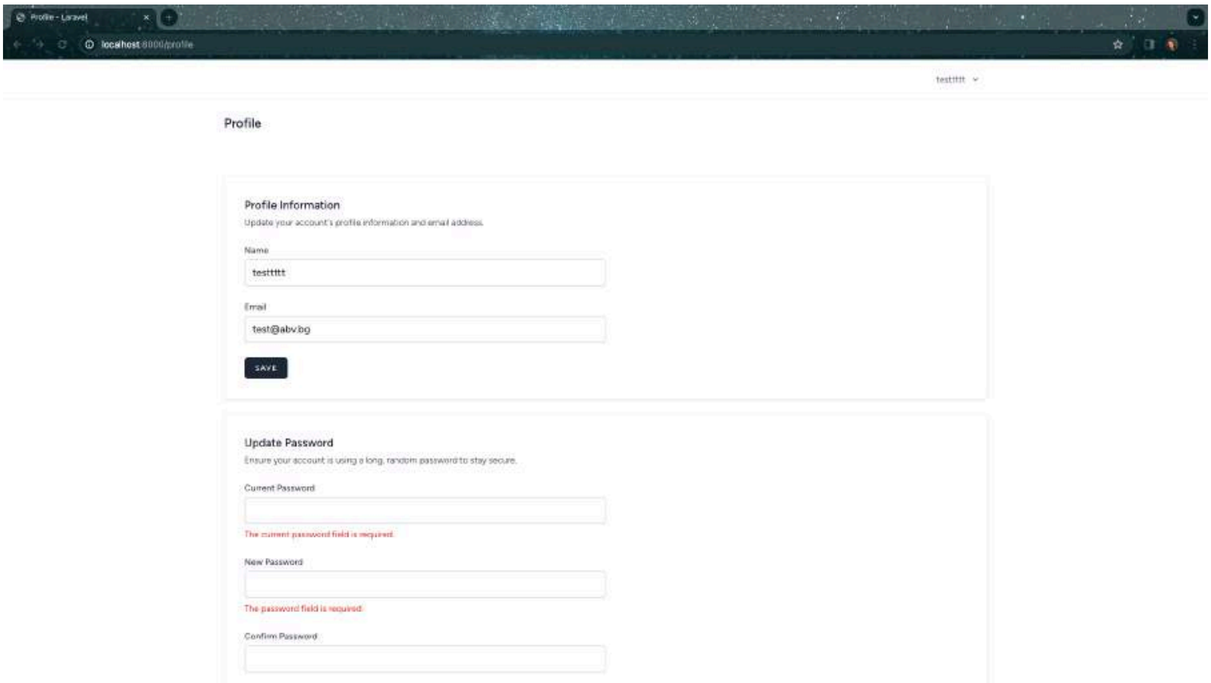


Stay Informed, Stay Ahea

Keep being updated and never miss something important. Your success journey begins with UniNotes.

"Education is the most powerful weapon which you can use to change the world." - Nelson Mandela

3. Select **Profile** and you should be redirected to your Profile Settings page.



Profile

Profile Information

Update your account's profile information and email address.

Name: testtttt

Email: test@abv.bg

Update Password

Ensure your account is using a long, random password to stay secure.

Current Password:

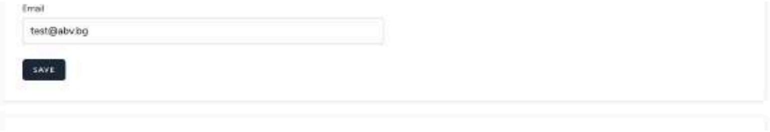
The current password field is required.

New Password:

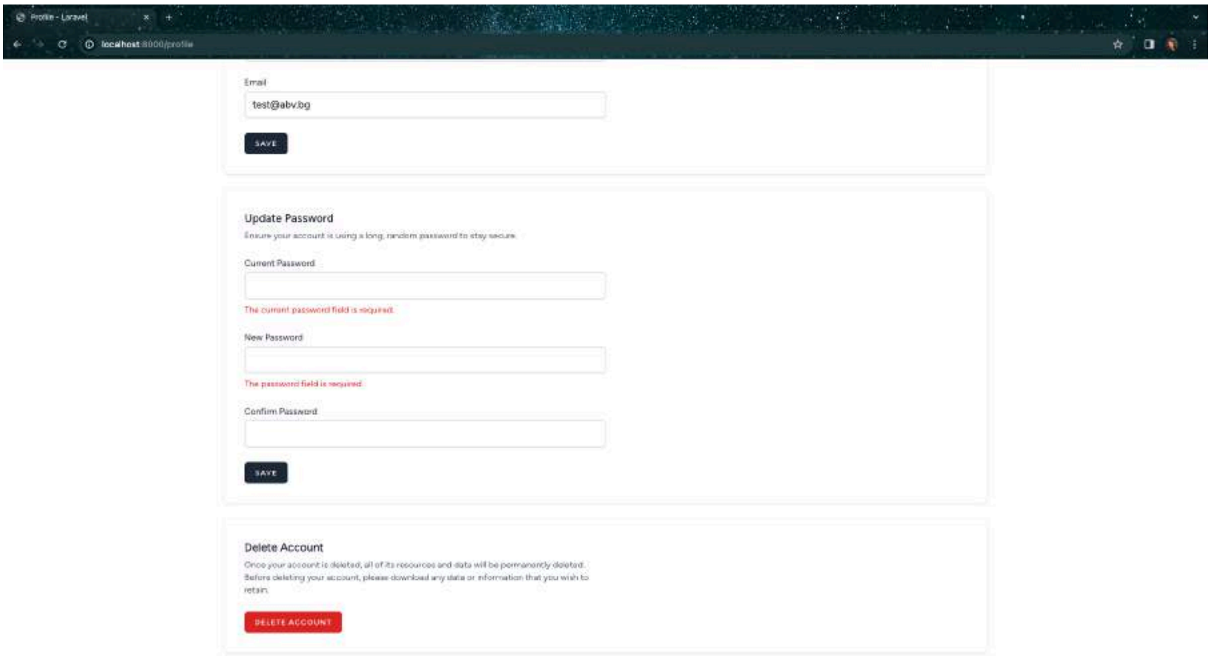
The password field is required.

Confirm Password:

4. Here you can update your name and email, or password, filling out the information in the Profile Information and the Update Password form, respectively. For each form there is a **Save** button on the bottom you should click, in order to save your changes.


Email: test@abv.bg

5. If you wish to delete your profile, scroll to the bottom and click on the **Delete Account** button.



Email

test@abv.bg

Update Password

Ensure your account is using a long, random password to stay secure.

Current Password

The current password field is required.

New Password

The password field is required.

Confirm Password

Delete Account

Once your account is deleted, all of its resources and data will be permanently deleted. Before deleting your account, please download any data or information that you wish to retain.

Conclusion

In conclusion, UniNotes represents a platform for collaborative note-taking among students. Throughout this documentation, we have delved into the components of UniNotes, from its client/server architecture and integration of Laravel and React to its features like collaborative note creation, importing/exporting notes, and responsive design testing. UniNotes streamlines the process of note management and fosters a dynamic environment for knowledge-sharing and collaboration.

As we navigate the landscape of higher education, UniNotes emerges as a reliable companion, providing a secure, efficient, and user-friendly space for students to organize their academic endeavors. The dedication to security, performance optimization, and continuous improvement through automated testing and deployment ensures that UniNotes aligns with the evolving needs of a diverse student community.

As the developers of UniNotes, we remain committed to refining and enhancing this platform to elevate the student experience further. We welcome feedback that contributes to the ongoing evolution of UniNotes.