



Daffodil
International
University

NoteNexus - A Web-Based Tool for Creating and Organizing Notes

Submitted By

Main Uddin

141-35-620

Department Of Software Engineering

Daffodil International University

Supervised By

Mr. A.H.M Shahariar Parvez

Associate Professor

Department of Software Engineering

Daffodil International University

A project presented as part of the requirements for earning a Bachelor of Science in Software
Engineering

Date of submission: 18-Aug-2024

APPROVAL

This project titled on “**Note Nexus**”, submitted by **Md. Main Uddin (ID: 141-35-620)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



Dr. Imran Mahmud
Associate Professor & Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman



Md. Maruf Hasan
Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1



Md. Shohel Arman
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2



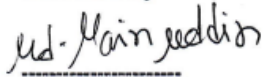
Dr. Md. Sazzadur Rahman
Professor
Institute of Information Technology
Jahangirnagar University

External Examiner

DECLARATION

I announce hereby that I am rendering this study document under **Mr. A.H.M Shahariar Parvez, Associate Professor**, Department of Software Engineering, Daffodil International University, I, therefore, state that this work or any portion of it was not proposed here therefore for Bachelor's degree or any graduation.

Submitted By



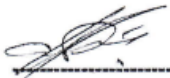
Md. Main Uddin

ID: 141-35-620

Batch: 13-B

Department of software engineering
Daffodil International University

Supervised By



Mr. A.H.M Shahariar Parvez

Associate professor

Department of software engineering
Daffodil International University

ACKNOWLEDGEMENT

Principally, I'm profoundly Acknowledging the supreme guidance that has made the realization of my final year project a reality. I wish to express my sincere thanks to Mr. A.H.M Shahariar Parvez, Associate Professor at Daffodil International University, with the Professor's relentless dedication, exceptional mentorship, and unwavering support. His insightful feedback, meticulous supervision, and invaluable advice throughout the many drafts have been Central to the project's successful achievement.

I also wish to acknowledge the significant contributions of my mentor, Mr. Md. Khaled Sohel (Assistant Professor), whose guidance has been invaluable. Special thanks go to Dr. Imran Mahmud, Head of the Department of SWE, with his crucial support.

Finally, I am deeply appreciative of my parents' steadfast encouragement and understanding, without whom this Expedition could not have been accomplished.

ABSTRACT

NoteNexus is a comprehensive note-taking web application designed using Next.js and Firebase, aimed at enhancing digital note organization and management. This project transcends basic note-taking functionalities by offering an intuitive and user-friendly interface, enabling users to effortlessly create, edit, delete, and archive notes. A key feature of NoteNexus is its advanced organizational capabilities, which enable users to assign importance levels and utilize tags for improved sorting. This capability not only streamlines the management of notes but also boosts efficiency by assisting users in effectively arranging their tasks. This system is designed to cater to both novice and experienced users, ensuring ease of navigation and an optimal user experience. Ultimately, NoteNexus aspires to provide a sophisticated, flexible, and user-centric solution for all digital note-taking needs, promoting efficient note management and enhanced productivity.

Table Of Contents

Content

APPROVAL	i
DECLARATION	ii
ACKNOWLEDGEMENT	iii
Table Of Contents	iv-v
Chapter 1: Introduction	1
1.1 Project Overview	1
1.2 Objective of the Project	1
Chapter 2: Architecture Review	2
2.1 Project Viability Study	2
2.1.1 Technological Viability	2
2.1.2 Operationalized Viability	2
2.2 Feature Specifications Requirements	3
2.3 Project System Requirements	3
2.4 Architecture Behavior Requirements	4
2.5 System Performance	4
Chapter 3: System Architecture	5
3.1 Class Diagram	6
3.2 Use Case Diagram	7
3.3 Activity Diagram	8
3.4 Sequence Diagram	9
3.5 ERD Diagram	10
Chapter 4: Software Development Resources	11
4.1 Code Development Tool	11
4.2 Coding Language	11
4.3 User Experience Design	11
4.4 Data Management System	11
4.5 Application Deployment Or Infrastructure	11

Chapter 5: End-to-End Testing	12-13
5.1 Feature Validation	12
5.1.1 Aspects to Validation	12
5.2 Quality Assurance Plans	12
5.2.1 Testing Methodology	12
5.2.2 Validation Benchmarks	12
5.2.3 Validation Timeline	12
5.3 Validation Scenarios	13
Chapter 6: Operating Instructions	14-20
6.1 Welcome Page	14
6.2 Workflow	14
6.2.1 Sign Up	15
6.2.2 Login	12
6.2.3 List of Notes	12
6.2.4 Create Form	12
6.2.5 Select Priority	12
6.2.6 Sort Note List	12
6.2.7 Trash List	12
6.2.8 Edit Note Colors	12
6.2.9 Archived Note List	12
6.2.10 Pinned Note Placement	12
6.2.11 Export and Copy Notes	12
Chapter 7: Conclusion	21
7.1 Development Reference	21
7.2 Restrictions	21
7.3 Prospective Development	21
7.4 Reference	22

CHAPTER 1

INTRODUCTION

1.1 Project Overview

NoteNexus is a comprehensive note-taking software built using React.js (Next.js) and Firebase. It simplifies digital note organization and management, providing efficient and user-friendly features for all your note-taking requirements.

1.2 Objective of the Project

The designed platform is developed to offer a highly efficient and user-friendly solution for managing notes. It goes beyond basic functionalities, enabling individuals to easily generate, adjust, remove, and organize their notes, meeting various demands in digital note management. Its key feature is its advanced organizational tools, which allow individuals to set priorities and apply tags for better sorting. This functionality not only simplifies the management process but also enhances efficiency by helping individuals track and arrange their tasks proficiently. Focusing on user experience, the platform is built to be accessible to both beginners and experienced users, ensuring straightforward navigation. In essence, this solution aims to be a flexible and effective tool, delivering a superior note management system marked by adaptability, structure.

ARCHITECTURE REVIEW

2.1 Project Viability Study

2.1.1 Technological Viability

Thorough review NoteNexus confirms its solid foundation and capacity for growth. Essential factors encompass:

- Rendering dynamic web pages efficiently is accomplished with Next.js, guaranteeing quick and smooth user interactions.
- Connecting with Firebase enables live data storage and syncing, delivering immediate updates for users.
- The system is designed to scale effectively, accommodating growing user numbers and increasing data volumes without performance degradation.
- Intelligent resource allocation ensures efficient utilization, adapting dynamically to varying user demands.
- The system maintains high responsiveness, ensuring a smooth user experience even as the user base expands.

2.1.2 Operationalized Viability

From an operational standpoint, the platform emphasizes ease of use with a clear and user-friendly design. Its practical viability is illustrated by:

- NoteNexus is crafted with a focus on user experience, providing an accessible interface and a layout that accommodates a wide range of technical skill levels.
- The system optimizes user interactions to improve overall satisfaction and reduce the time required to get accustomed to it.
- Built-in accessibility tools guarantee that users from various backgrounds can engage with the platform effectively.
- Ease of use is prioritized, with a simple introductory process and support for various hardware and web.

2.2 Feature Specifications Requirements

Fundamental functionalities of the software are carefully outlined to offer a thorough and user-centered approach. Key aspects include:

- A streamlined method allowing individuals to sign up and create customized profiles.
- Robust login protocols designed to protect user information.
- Seamless management, modification, removal, and storage of electronic entries.
- User-friendly interface enabling efficient organization and handling of notes.
- Users have the option to categorize each note by assigning distinct importance ratings to manage their tasks more effectively.
- Optimized tools for task management elevate user productivity.
- User-friendly features for categorizing notes via labels ensure organized retrieval.
- An efficient labeling system facilitates structured note organization.
- Intuitive features are implemented to enrich overall usability.
- Incorporation of shared note-editing features to improve collaborative efforts.
- Search capabilities are integrated to enable users to swiftly find notes by entering relevant terms.
- Adaptive layout delivers a uniform and intuitive interaction across all types of platforms.
- Ongoing enhancements and revisions based on user input and changing requirements.

Well-established criteria act as the foundation for the software while ensuring users benefit from a rich and adaptable environment tailored to handle a broad spectrum of note management and organization needs.

2.3 Project System Requirements

- Chrome, Firefox, Safari, and Edge should all deliver uninterrupted operation to ensure smooth functionality across these major web browsers.
- Refine the design to ensure uniform functionality and aesthetics across various web browsers.
- Develop the software to operate effectively in diverse network environments, from high-speed connections to limited bandwidth conditions.
- Incorporate capabilities like offline functionality or adaptive performance to improve user satisfaction during varying network conditions.
- Embed Firebase into the app's framework to enable instant data storage and syncing.
- Utilize Firebase's features to implement robust security measures for user authentication and data protection.

2.4 Architecture Behavior Requirements

- Delimit performance criteria to ensure that user activities are managed promptly and efficiently.
- Refine programming and optimize asset management to reduce waiting times and improve overall efficiency.
- Implement state-of-the-art security measures to safeguard user information during both data transfer and storage.
- Create a design centered on user needs to offer a clear and user-friendly interface.
- Perform user experience evaluations to gather insights and enhance the interface based on real-world usage.
- Introduce redundancy protocols and frequent data storage measures to safeguard operational stability and data preservation.
- Adhere to performance criteria to reduce the likelihood of operational failures and ensure consistent performance.

2.5 System Performance

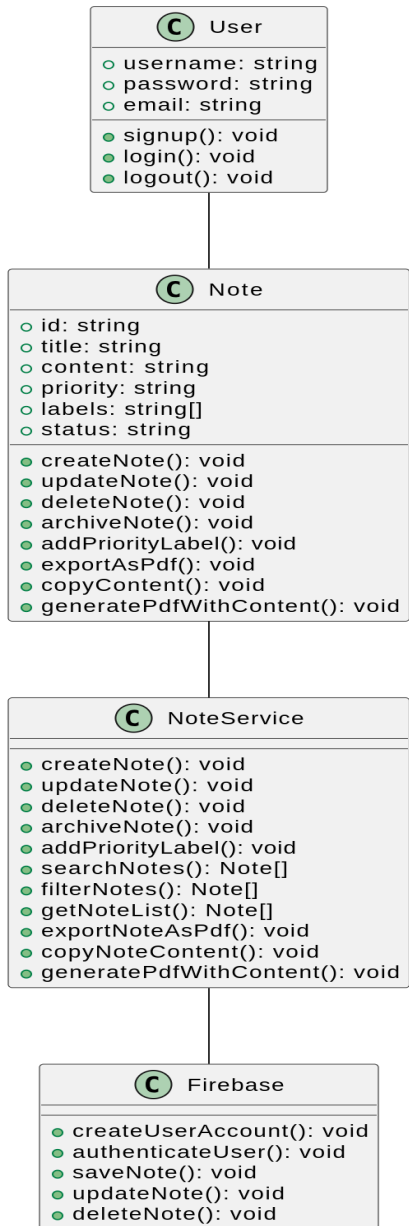
- NoteNexus is finely tuned for optimal performance, allowing for rapid and efficient access to notes.
- By prioritizing minimal delays, the platform ensures an interface that reacts quickly, improving overall user satisfaction.
- The focus on operational efficiency in NoteNexus's design guarantees that interactions are both seamless and user-friendly.
- The ability to quickly access notes is central to the platform's design, enhancing the efficiency and satisfaction of its use.
- Engineered for consistent and rapid performance, the platform aligns with diverse user needs across different tasks.
- Optimization is integrated throughout NoteNexus, ensuring that all user interactions are handled with precision.
- Rapid retrieval of notes is a core element embedded deeply within NoteNexus design strategy.
- NoteNexus prioritizes optimal efficiency to ensure a seamless and pleasant information management process.
- NoteNexus's consistent and rapid functionality ensures it remains a dependable and accessible tool for managing notes. Reliability in delivering swift and effective performance makes NoteNexus a dependable and easy-to-use note-taking solution.

CHAPTER 3

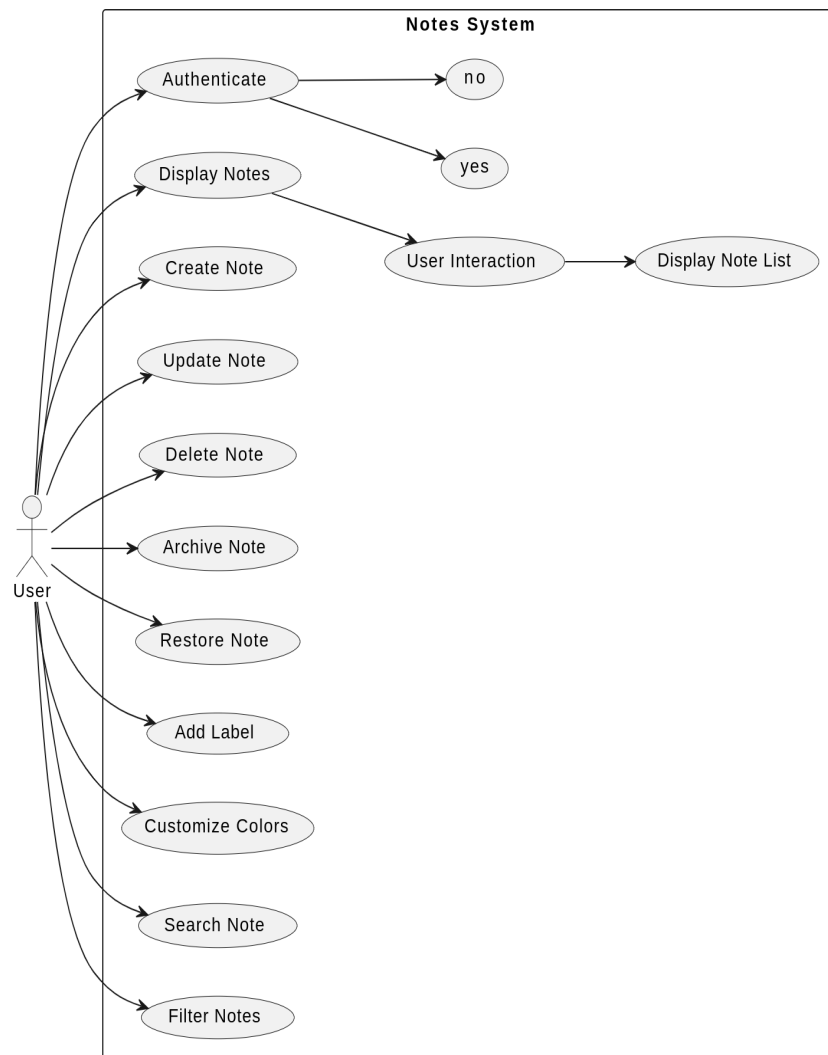
SYSTEM ARCHITECTURE

As we proceed through the upcoming section, we will examine several crucial diagrams, starting with how users engage with the system. The Use Case diagram will guide us by illustrating these interactions and defining the system's operational boundaries. Next, Activity diagrams will reveal the sequence of processes across various elements of the system. The Entity-Relationship Diagram (ERD) will offer insights into the database's structure, emphasizing the relationships between essential entities. Following that, the Sequence diagram will depict the timing and order of interactions across various system modules. Finally, the Class diagram will outline the object-oriented architecture, detailing the attributes and relationships of the classes that facilitate the system's operations. Each of these diagrams contributes to a thorough knowledge of project architecture belonging to diverse viewpoints.

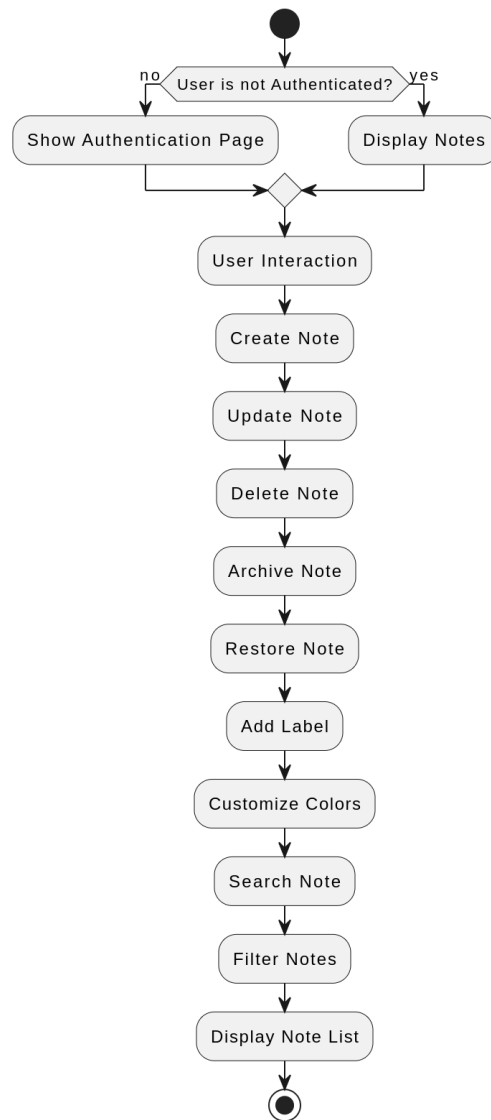
3.1 Class



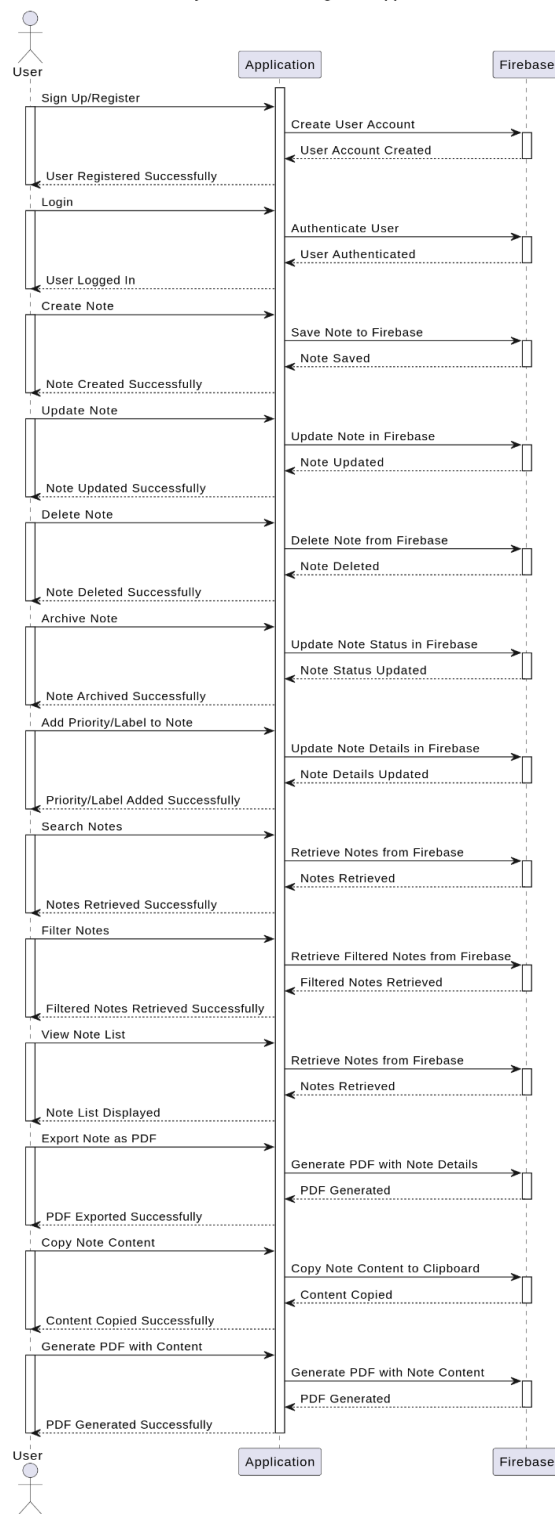
3.2 Use Case



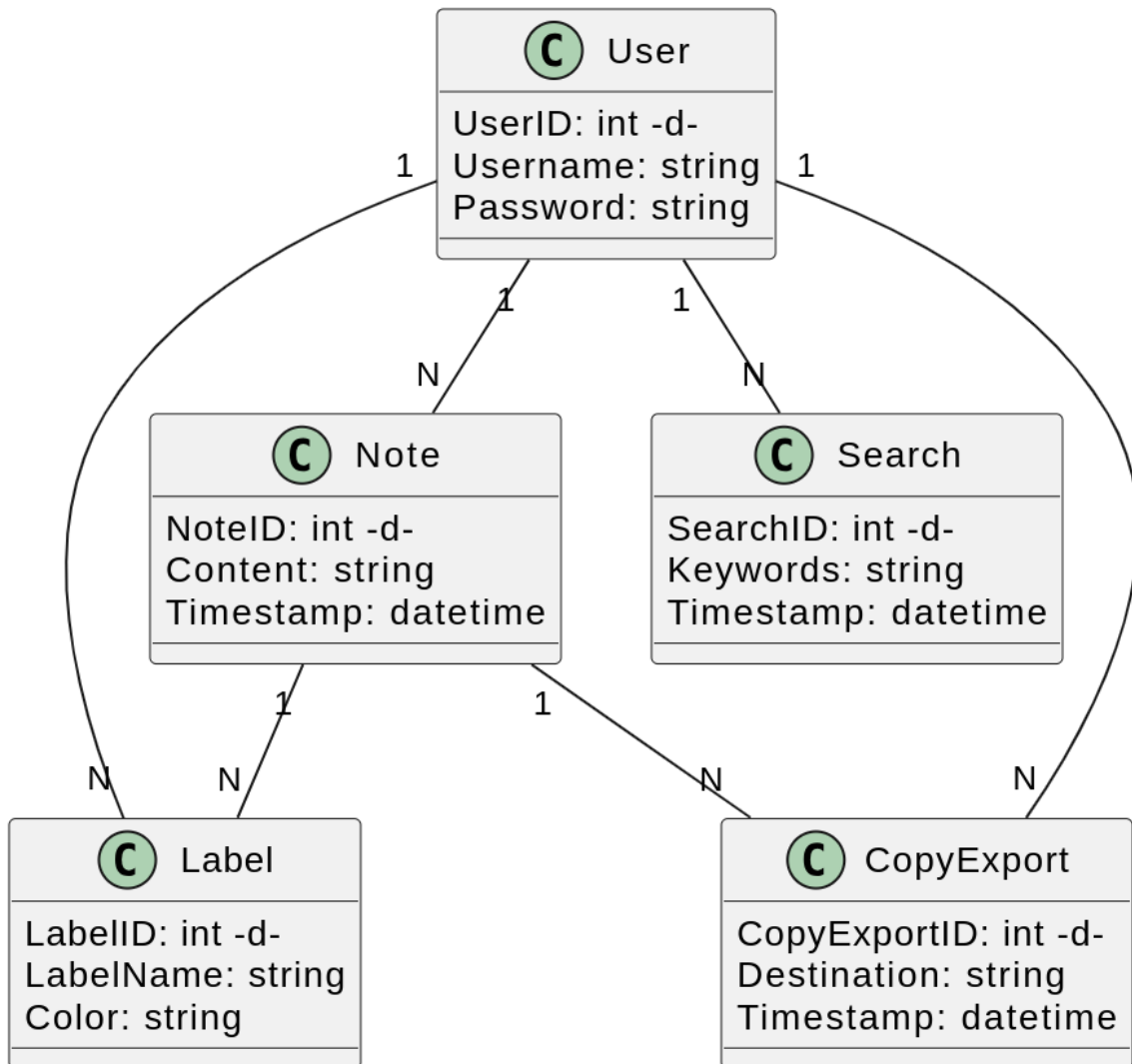
3.3 Activity



3.4 Sequence



3.5 ERD



CHAPTER 4

SOFTWARE DEVELOPMENT RESOURCES

4.1 Code Development Tool

This initiative is built with VS Code serving as one of the necessary software toolkits.

4.2 Coding Language

For frontend development, the project employed (Next.js), a framework built on React, while the backend was powered by Firebase, a NoSQL database.

4.3 User Experience Design

The interface design emphasized ease of use and practicality aiming to provide an excellent user interaction.

4.4 Data Management System

The application utilizes Firebase live data store due to its excellent expandability and smooth connection capabilities

4.5 Application Deployment Or Infrastructure

This software has been launched and maintained by the AWS platform, ensuring dependable performance, easy access for clients.

CHAPTER 5

END-TO-END TESTING

5.1 Feature Validation

5.1.1 Aspects to Validation

- Examine the note entry page to verify that it is easy to use, straightforward, and accessible for all users.
- Ensure the system properly screens user inputs at the time of note entry to prevent mistakes or incorrect data from being recorded.
- Test the process of adding notes on various devices (tablet, Webpage, Handheld device) to confirm a fluid user interface.
- Relevant, review this software's functionality for enabling joint note-taking, ensuring that several users can input data simultaneously without any interference .
- Ensure that the platform logs and retains relevant details during note entry, including timestamps or user IDs, for monitoring and organizational purposes.

5.2 Quality Assurance Plans

5.2.1 Testing Methodology

The approach to testing encompasses both manual and automated methods to guarantee comprehensive evaluation.

5.2.2 Validation Benchmarks

- Success: The functionality for adding entries operates flawlessly and without issues.
- Issue: Users experience problems that prevent them from successfully adding entries.

5.2.3 Validation Timeline

The evaluation will occur in several stages, beginning With test-driven development and advancing through Consolidation, software assessments.

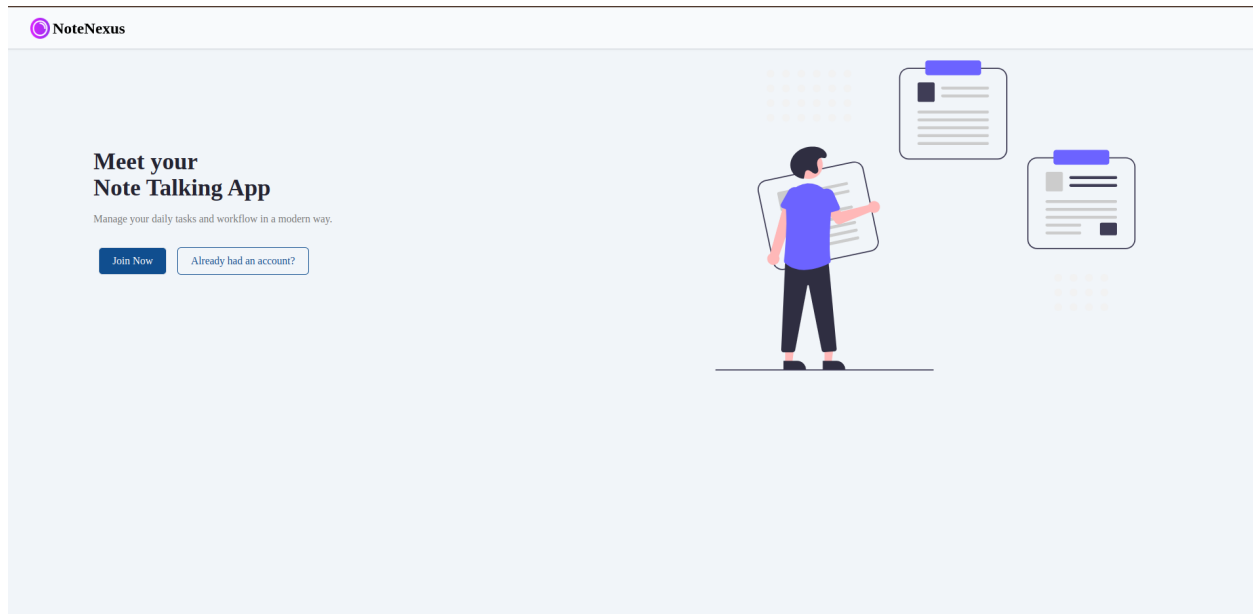
5.3 Validation Scenarios

- Validate user registration success.
- Validate user login success.
- Validate content generation.
- Validate content removal.
- Validate content storage.
- Validate content retrieval.
- Validate content search.
- Validate exporting content as a PDF.
- Validate the coping.

CHAPTER 6

OPERATION INSTRUCTIONS

6.1 Welcome Page



6.2 Workflow

In the forthcoming chapter, we will undertake a visual analysis by reviewing several important diagrams that are key to grasping the intricacies of the initiative. The Use Case diagram will provide insights into how end clients experience the application & outline of boundaries. Following this, Diagrams will display all around the evolving workflows, highlighting the sequence of actions across various system elements.

6.2.1 Sign Up

NoteNexus

Sign Up

First Name

Last Name

Email

Password

Sign Up

[Already Have an Account](#)

6.2.2 Login

NoteNexus

Login

Email

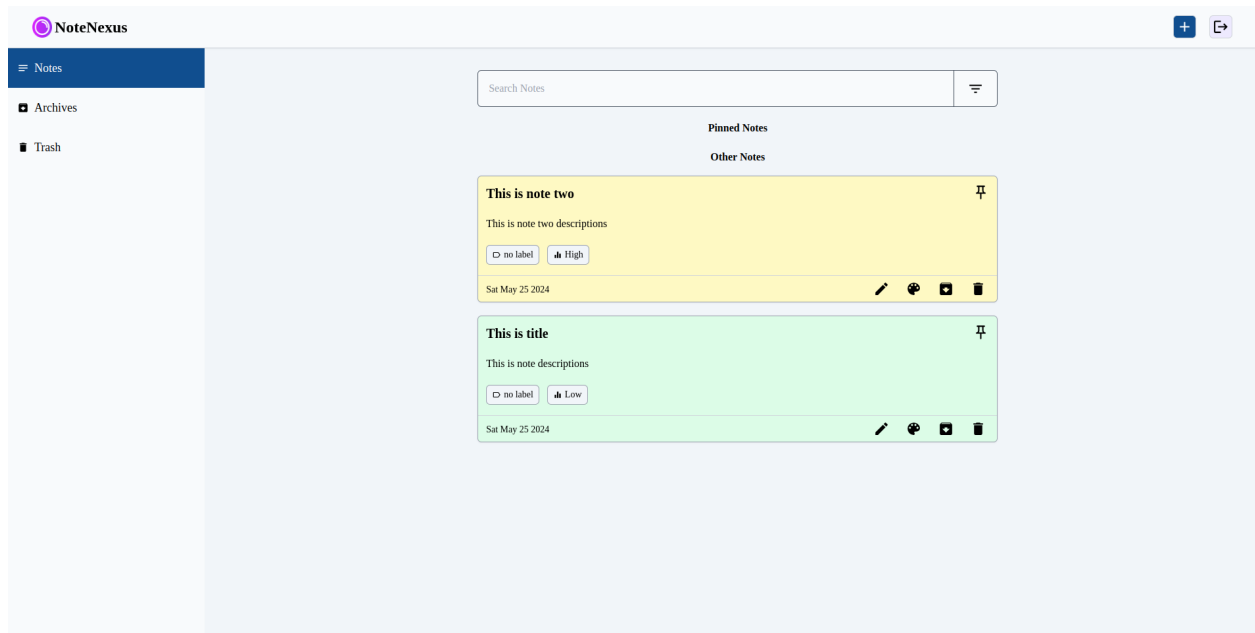
Password

Login

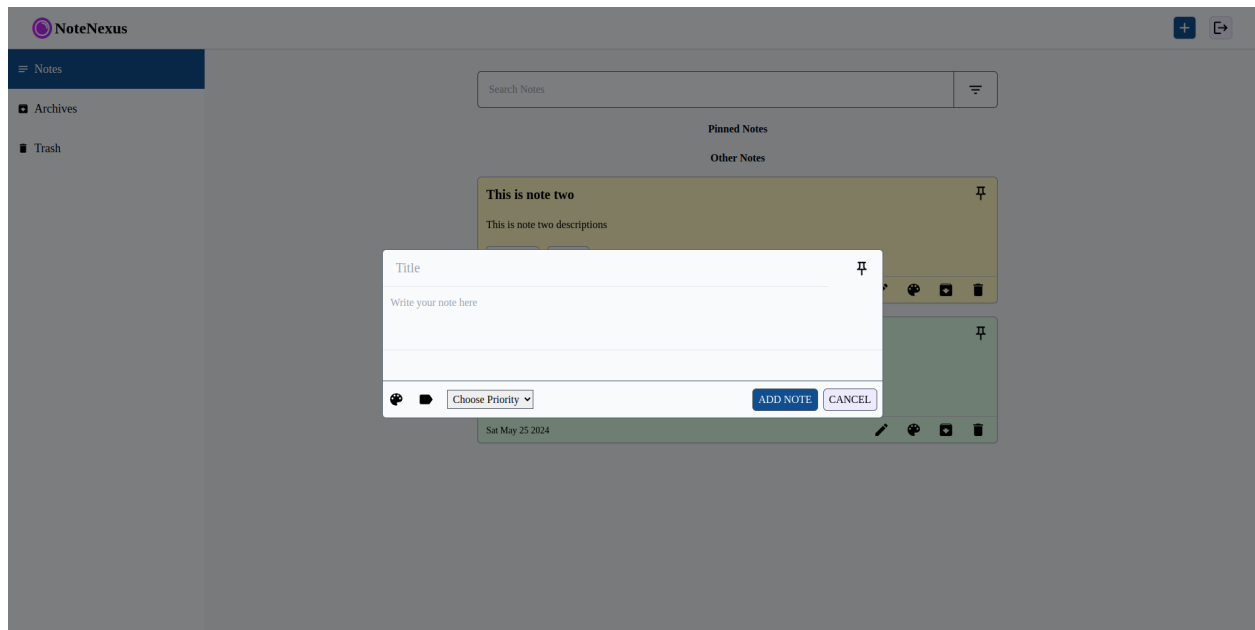
Login As Guest

[Create New Account](#)

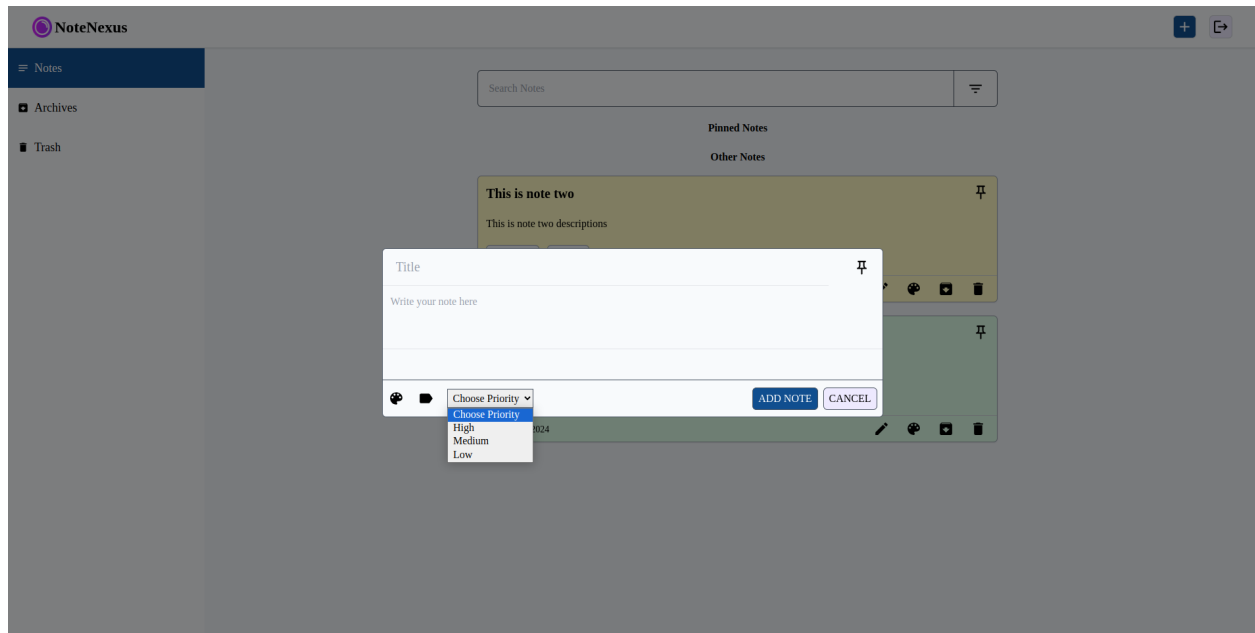
6.2.3 List of Notes



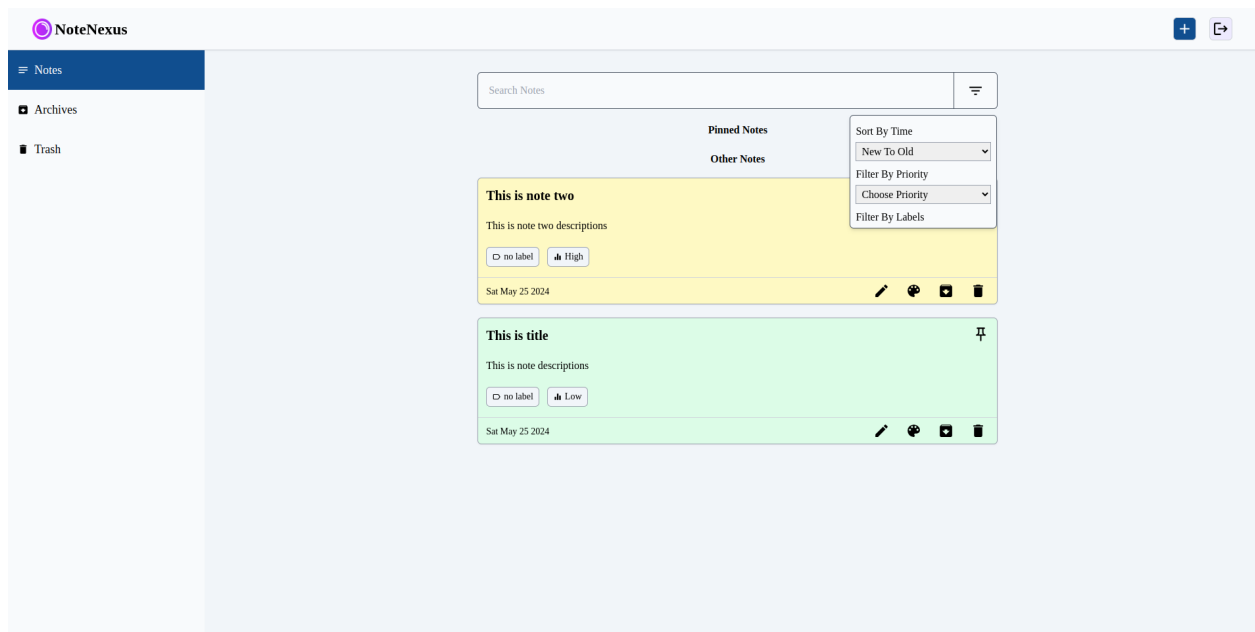
6.2.4 Create form



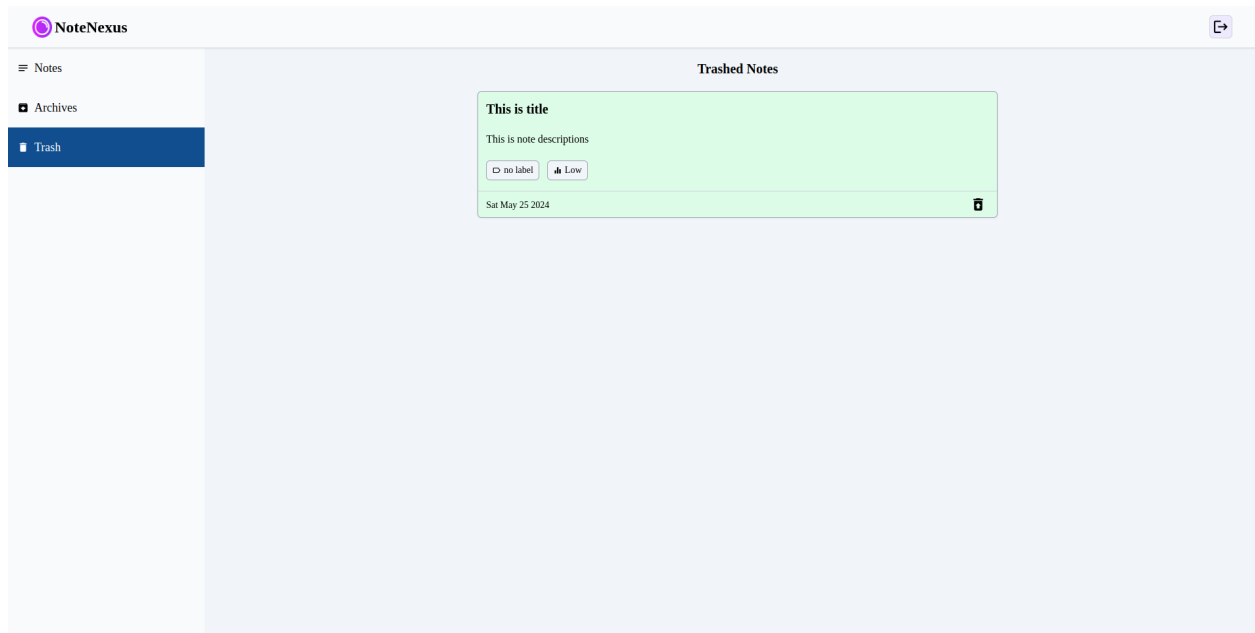
6.2.5 Select Priority



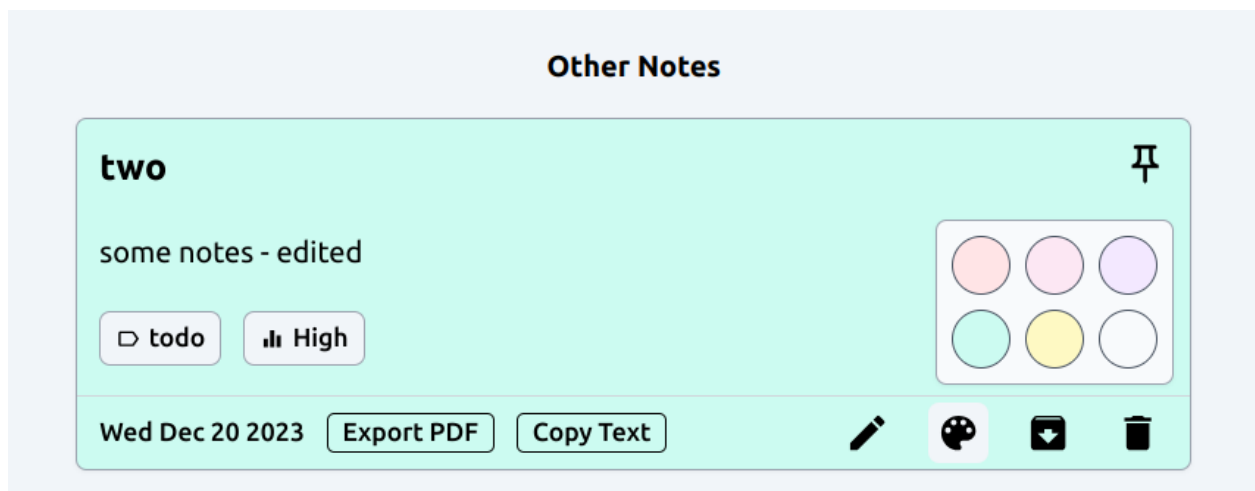
6.2.6 Sort Note List



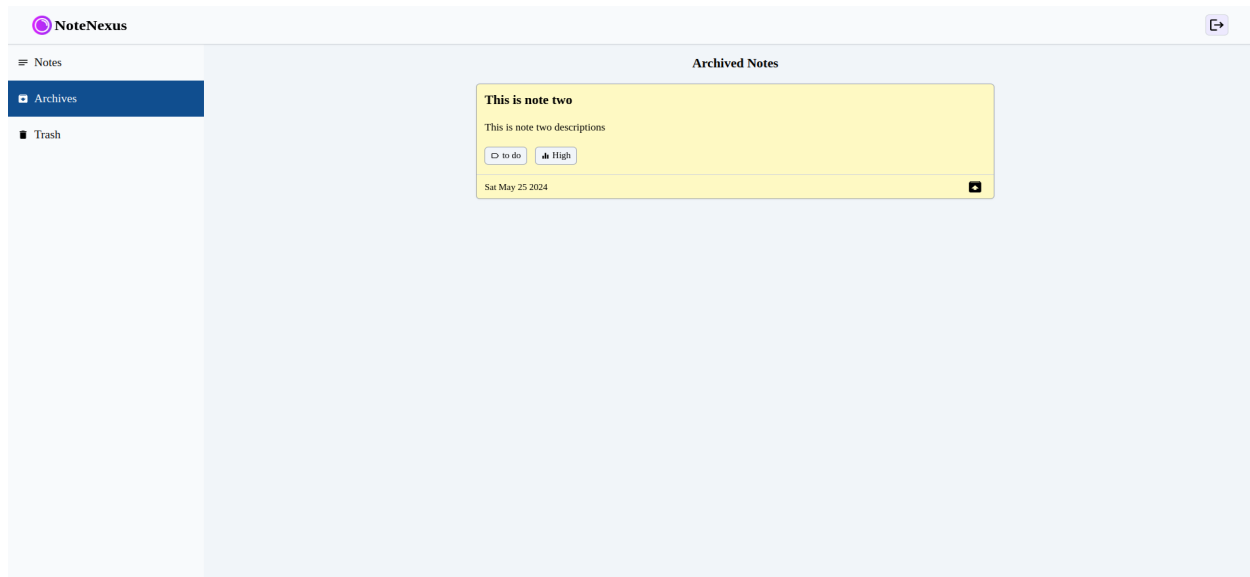
6.2.7 Trash List



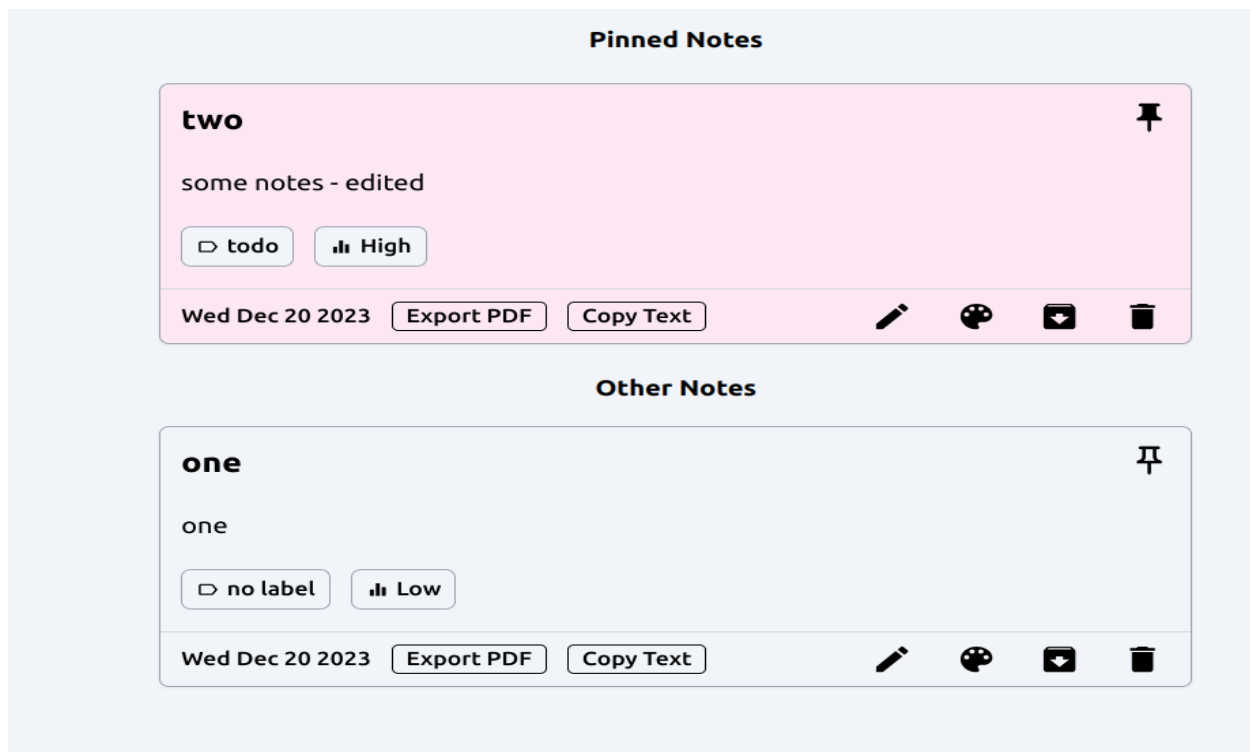
6.2.8 Edit Note Colors



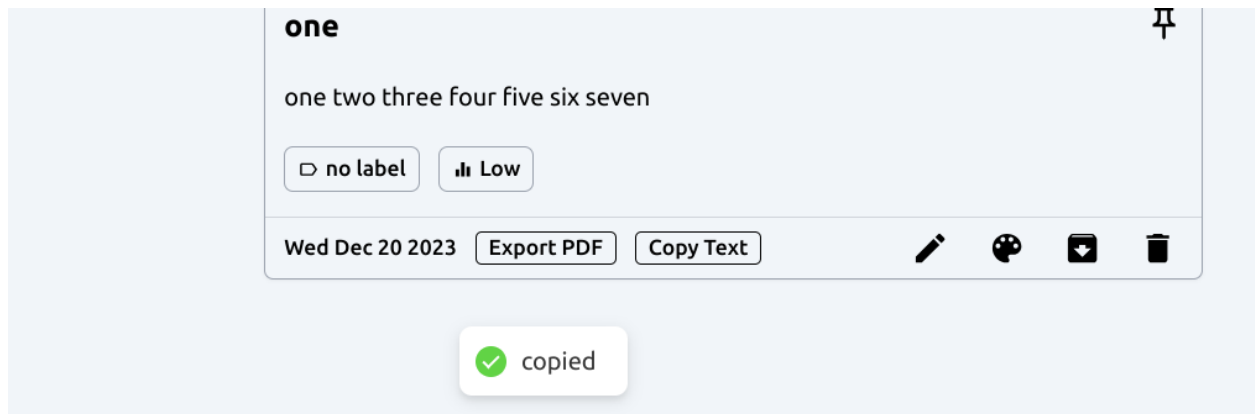
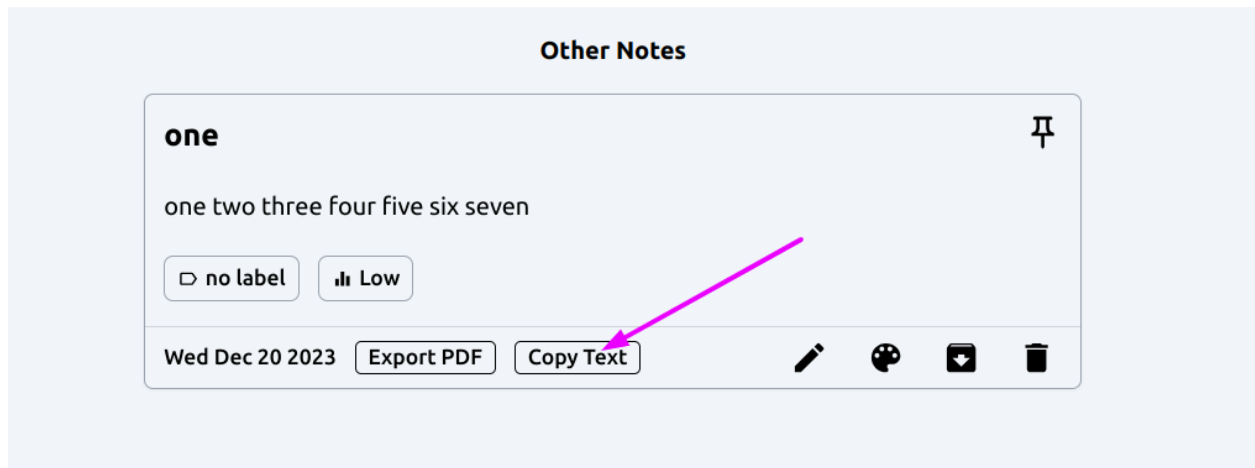
6.2.9 Archived Note List



6.2.10 Pinned Notes Placement



6.2.11 Export and Copy Notes



CONCLUSION

NoteNexus effectively meets demand with streamlined & well-structured electronic note-taking solutions. Major accomplishments include an intuitive user interface, a strong system framework, and smooth connectivity with Firebase.

7.1 Development Reference

<https://notenexus620.netlify.app/>

7.2 Restrictions

- The system's dependence on instantaneous updates makes it vulnerable to disruptions during periods of connectivity issues.
- Challenges with poor network connectivity can lead to lag or irregularity in live updates, impacting the complete performance.
- The reliability of the application's live features depends on a stable network connection, presenting a possible vulnerability.
- It is important to develop effective measures to minimize the impact of network disruptions on live updates, guaranteeing the application remains reliable under different network conditions.

7.3 Prospective Developments

- Upcoming upgrades could integrate shared features, allowing several users to modify notes concurrently, enhancing cooperation and immediate interaction.
- Improvements could include expanding text customization features, enabling individuals to personalize design features, arrangements, and additional components to enhance visual appeal.
- Integrating additional third-party tools is another potential improvement, enhancing the app's capabilities and providing users with a smooth interaction with their chosen productivity resources.
- Upcoming updates could introduce an enhanced editor with extended functionalities, offering users a more dynamic and refined content management experience.

7.4 Reference

Vercel. (n.d.). Vercel. Retrieved from <https://vercel.com/>

Firebase. (n.d.). Firebase. Google. Retrieved from <https://firebase.google.com/>

bpampuch. (n.d.). *pdfmake*. GitHub. Retrieved from <https://github.com/bpampuch/pdfmake>