

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC A++ Accredited Institute Affiliated to RGPV, Bhopal)



Minor Project Report

on

“NotesCLI: A CLI Notes App with Web support”

A project report submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

Submitted by:

Jainul Abdeen Hashmee

0901IT211027

Faculty Mentor:

Mr. Shubham Sharma

Assistant Professor

Information Technology

Submitted to:

DEPARTMENT OF INFORMATION TECHNOLOGY

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE

GWALIOR - 474005

July-December 2023

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC A++ Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Jainul Abdeen Hashmee** (0901IT211027) has submitted the project report titled **NotesCLI: A CLI Notes App with Web support** under the mentorship of **Mr. Shubham Sharma**, in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in **Information Technology** from Madhav Institute of Technology and Science, Gwalior.

Mr. Shubham Sharma
Assistant Professor
Information Technology

Dr. Sanjeev Sharma
Coordinator,
Information Technology

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC A++ Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in Information Technology at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Mr. Shubham Sharma**, Assistant Professor, Information Technology.

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Date:
Place: Gwalior

Jainul Abdeen Hashmee
0901IT211027
III Year,
Information Technology

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC A++ Accredited Institute Affiliated to RGPV, Bhopal)

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Information Technology**, for allowing me to explore this project. I humbly thank **Dr. Sanjeev Sharma**, Coordinator, Department of Information Technology, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Mr. Shubham Sharma**, Assistant Professor, Department of IT, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Jainul Abdeen Hashmee
0901IT211027
III Year,
Information Technology

ABSTRACT

In the digital era, efficient note management is an essential aspect of personal and professional productivity. My project, titled "NotesCLI: A CLI Notes App with Web View," presents a comprehensive solution designed to streamline the process of capturing and organizing thoughts. Integrating a user-friendly web interface with the flexibility of a Command Line Interface (CLI), NotesCLI offers a harmonious platform for users to manage their notes seamlessly.

The web interface provides an intuitive and visually appealing environment for users to view and add notes effortlessly. Tailored with a responsive design, it ensures a consistent and engaging experience across various devices. The incorporation of features such as search functionality and date-based categorization enhances the accessibility and organization of notes.

Complementing the web interface is the CLI, offering a powerful and efficient alternative for users comfortable with command-line interactions. This allows users to interact with their notes using familiar commands, providing a flexible and dynamic experience tailored to the preferences of command-line enthusiasts.

Behind the scenes, the project employs a robust technology stack, utilizing React for the frontend, Node.js for the backend, and MongoDB for data storage. The combination of these technologies ensures a scalable, secure, and responsive application.

In conclusion, NoteCraft stands as a testament to the synergy between modern web technologies and the efficiency of command-line interfaces. The project not only addresses the diverse needs of note-takers but also serves as an exploration into the harmonious integration of different user interfaces within a singular application.

Keyword: Note-taking, web interface, command line interface(CLI), React, Node.js, MongoDB, Express.js, User Interface(UI)

सार:

"डिजिटल युग में, दक्ष नोट प्रबंधन व्यक्तिगत और पेशेवर उत्पादकता का एक अनिवार्य पहलु है। मेरी परियोजना, जिसका शीर्षक **"NotesCLI: एक सीएलआई नोट ऐप विथ वेब व्यू,"** है, एक समृद्धि समाधान प्रस्तुत करती है जिसका उद्दीपन है विचार को कैप्चर और संगठित करने की प्रक्रिया को सुगम बनाना। एक उपयोगकर्ता-मित्र वेब इंटरफेस को एक समर्थन और पूर्व-प्रदर्शनी संबंधी विराट्मक पर्यावरण प्रदान करता है जिससे उपयोगकर्ताओं को उनके नोट्स को आसानी से देखने और जोड़ने का तरीका मिलता है। एक प्रतिस्पर्धी डिजाइन के साथ तैयार किया गया, यह विभिन्न उपकरणों पर एक संरक्षित और आकर्षक अनुभव सुनिश्चित करता है। खोज क्षमता और तिथि-आधारित वर्गीकरण जैसी सुविधाएँ सूचियों की पहुंच और संगठन में वृद्धि होती हैं।

वेब इंटरफेस को सीएलआई के साथ पूर्ण करने के साथ, **NotesCLI** एक समर्थन प्लेटफॉर्म प्रदान करता है जिसमें उपयोगकर्ताओं को उनके नोट्स को सुगमता से प्रबंधित करने का एक समर्थ और स्विचित अनुभव मिलता है। इसके साथ ही, यह एक शक्तिशाली और कुशल पर्यावरण प्रदान करने वाला सीएलआई है जिससे उपयोगकर्ताएँ कमांड-लाइन इंटरएक्शन के साथ संवाद कर सकती हैं। यह उपयोगकर्ताओं को उनके नोट्स के साथ परिचित कमांड्स का उपयोग करने का एक लचीला और गतिशील अनुभव प्रदान करता है, जो कमांड-लाइन प्रेमियों की पसंदों के अनुसार विशेषज्ञ योग्यता प्रदान करता है।

परियोजना की पीछे, हमने एक मजबूत प्रौद्योगिकी स्टैक का उपयोग किया है, जिसमें फ्रंटएंड के लिए रिएक्ट, बैकएंड के लिए नोड.जेएस और डेटा स्टोरेज के लिए मॉनगोडीबी शामिल हैं। इन प्रौद्योगिकियों के संयोजन से एक स्केलेबल, सुरक्षित और प्रतिस्पर्धी एप्लिकेशन सुनिश्चित होता है।

समापन में, **NotesCLI** एक सुविधाएँ से भरपूर नोट-टैकिंग के विभिन्न आवश्यकताओं को पूरा करने के लिए नहीं है ही, ब

TABLE OF CONTENTS

PAGE NO.

List of figures

Abbreviation

Abstract

सार

Chapter 1: Project Overview

1.1 Introduction

1.2 Objectives and Scope

1.3 Project Features

1.3.1 Web Interface

1.3.2 CLI

1.4 Feasibility

1.5 System Requirements

1.5.1 Hardware requirements

1.5.2 Software Requirements

1.6 Applicability

Chapter 2: Literature Review

2.1 Introduction

2.2 Previous Work in Note Management Applications

2.2.1 Traditional Note-Taking Apps

2.2.2 Command-Line Note-Taking Tools

2.2.3 Hybrid Solutions

2.3 Current Technologies in Web Development

2.3.1 Frontend Technologies

2.3.2 Backend Technologies

2.3.3 Database Technology

2.4. Integration of Web Technologies with Command-Line Interface

2.4.1 Challenges and Considerations

2.4.2 Successful Implementations

2.5. Summary

Chapter 3: Preliminary Design

3.1. Introduction

3.2. System Architecture

- 3.2.1 Frontend Architecture
- 3.2.2 Backend Architecture
- 3.3. User Interface Design
 - 3.3.1 Web Interface
 - 3.3.2 Command-Line Interface (CLI)
- 3.4. Technologies Used
 - 3.4.1 Frontend Technologies
 - 3.4.2 Backend Technologies
- 3.5. Architecture Diagram
- 3.6. Preliminary User Stories
- 3.7. Summary

Chapter 4: Final Analysis and Design

- 4.1. Introduction
- 4.2. Results
 - 4.2.1 Web Interface
 - 4.2.2 Command-Line Interface (CLI)
- 4.3. Result Analysis
 - 4.3.1 Web Interface
 - 4.3.2 Command-Line Interface (CLI)
- 4.4. Application
 - 4.4.1 Real-world Application
 - 4.4.2 Use Cases
- 4.5. Problems Faced
 - 4.5.1 Integration Challenges
 - 4.5.2 Learning Curve for CLI Users
- 4.6. Limitations
 - 4.6.1 Limited Offline Functionality
 - 4.6.2 Command-Line Compatibility
- 4.7. Summary

Conclusion

Recommendations

References

LIST OF FIGURES

Figure Number	Figure caption	Page No.
3.5	NotesCLI Architecture Diagram	
4.2.1	NotesCLI Web View	
4.2.2.1	NoteCLI all commands	
4.2.2.2	NoteCLI list command	
4.2.2.3	NoteCLI remove command	
4.2.2.4	NoteCLI web command	
4.2.2.5	MongoDB Database	
4.2.2.6	MongoDB Database	

LIST OF ABBREVIATIONS

Abbreviation	Description
CLI	Command Line Interface
UI	User Interface
CSS	Cascading Style Sheet
JS	JavaScript
DB	DataBase
API	Application Programming Interface

Chapter 1: PROJECT OVERVIEW

1.1 Introduction

In the contemporary landscape of digital advancements, efficient note management is crucial for personal and professional productivity. The project, titled "NotesCLI: A CLI Notes App with Web View," is designed to provide a comprehensive solution to streamline the process of capturing and organizing thoughts. This chapter introduces the project, outlines its objectives, scope, features, feasibility, system requirements, and applicability.

1.2 Objectives and Scope

The primary objectives of the project are to deliver a unified note-taking experience through a user-friendly web interface and a Command Line Interface (CLI). The system aims to simplify note creation, organization, and retrieval. The scope includes integrating modern web technologies with efficient command-line interactions, creating a seamless and flexible note management platform.

1.3 Project Features

1.3.1 Web Interface

- a. Intuitive Design: The web interface boasts an intuitive and visually appealing design to enhance the user experience.
- b. Effortless Interaction: Users can effortlessly view and add notes through the web interface, ensuring a straightforward process.
- c. Responsive Design: The design is responsive, providing a consistent and engaging experience across various devices.

1.3.2 Command Line Interface (CLI)

Powerful and Efficient: The CLI offers a powerful and efficient alternative for users comfortable with command-line interactions. **Familiar Commands:** Users can interact with their notes using familiar commands, catering to the preferences of command-line enthusiasts.

1.4 Feasibility

The feasibility study evaluates the practicality and viability of the project. Considering technical, operational, and economic aspects, this study ensures that the proposed solution is achievable and sustainable.

1.5 System Requirements

1.5.1 Hardware Requirements

Processor: Dual-core processor or higher
RAM: 4GB or more
Storage: 50GB of available storage

1.5.2 Software Requirements

Operating System: Windows 10
Web Browser: Google Chrome, Mozilla Firefox, or Safari
Node.js: Version 14.17.0 or higher

1.6 Applicability

The applicability section explores potential use-cases and scenarios where the project can be beneficial. NotesCLI addresses diverse user needs in note-taking and organization, making it suitable for students, professionals, and anyone seeking an efficient note management system. Following are the Use Cases for NotesCLI App:

a. Student Note-Taking:

Scenario: A student wants to take organized notes for different courses and subjects.

Use Case: The student uses the web interface to create and manage notes for each course. The CLI option provides a quick way to jot down important points during lectures.

b. Professional Task Management:

Scenario: A professional needs a centralized system to manage tasks and project-related notes.

Use Case: The professional utilizes the web interface to categorize and organize notes related to various projects. The CLI allows for efficient interaction, especially when working on the command line is preferred.

c. Quick Ideas Capture:

Scenario: Users often have spontaneous ideas that need to be recorded instantly.

Use Case: A user employs the CLI to swiftly capture thoughts or ideas without disrupting the current workflow. Later, these notes can be reviewed and expanded using the web interface.

d. Research and Article Writing:

Scenario: A writer or researcher needs to compile information and ideas for articles or papers.

Use Case: The user utilizes the web interface to gather and organize research notes. The CLI becomes handy for jotting down quick references or ideas during the research process.

e. Search and Retrieval:

Scenario: Users frequently need to find specific notes quickly.

Use Case: The search functionality in the web interface allows users to enter keywords and retrieve relevant notes. This feature is particularly useful for users with an extensive collection of notes.

This chapter establishes the foundation for a comprehensive understanding of the project, providing a roadmap for subsequent chapters that delve deeper into each aspect.

Chapter 2: Literature Review

2.1. Introduction

The literature review delves into existing works and current technologies relevant to note management applications, providing a foundation for understanding the state of the art in this field. This chapter explores prior research, existing tools, and technologies that inform the design and development of the "NotesCLI: A CLI Notes App with Web View."

2.2. Previous Work in Note Management Applications

2.2.1 Traditional Note-Taking Apps

Traditional note-taking applications like Evernote, Microsoft OneNote, and Google Keep have established themselves as popular choices. They typically offer rich features, cloud synchronization, and a graphical user interface (GUI) for note management.

2.2.2 Command-Line Note-Taking Tools

Several command-line note-taking tools, such as 'nvALT' and 'Joplin,' cater to users who prefer a text-based interface. These tools often focus on simplicity and speed for users comfortable with the command line.

2.2.3 Hybrid Solutions

Some applications, like 'Bear' and 'Boostnote,' attempt to bridge the gap between traditional graphical interfaces and command-line interfaces, offering a hybrid approach. These tools aim to provide the best of both worlds, catering to a diverse user base.

2.3. Current Technologies in Web Development

2.3.1 Frontend Technologies

React.js: A widely adopted JavaScript library for building user interfaces, known for its component-based architecture and efficient rendering.

Tailwind CSS: A utility-first CSS framework that facilitates rapid UI development with pre-defined utility classes.

React-datepicker: A flexible and customizable datepicker component for React.

2.3.2 Backend Technologies

Node.js: A server-side JavaScript runtime, commonly used for building scalable and high-performance web applications.

Express.js: A minimal and flexible Node.js web application framework, ideal for building robust APIs.

2.3.3 Database Technology

MongoDB: A NoSQL database known for its flexibility, scalability, and ease of integration with Node.js applications.

2.4. Integration of Web Technologies with Command-Line Interface

2.4.1 Challenges and Considerations

Integrating web technologies with a command-line interface presents challenges such as maintaining consistency in data, ensuring a seamless user experience, and addressing the diverse preferences of users.

2.4.2 Successful Implementations

Some projects, like 'Taskbook' and 'jrnl,' have successfully integrated web and command-line interfaces. These implementations offer insights into addressing challenges and providing a cohesive user experience.

2.5. Summary

This literature review provides a comprehensive understanding of existing note management applications, ranging from traditional GUI-based tools to command-line interfaces. The exploration of current web development technologies and successful integrations informs the design and development of the "NotesCLI" project. The subsequent chapters will build upon this knowledge to create a novel application that combines the strengths of both web and command-line interfaces.

Chapter 3: PRELIMINARY DESIGN

3.1. Introduction

This chapter outlines the preliminary design of the "NotesCLI: A CLI Notes App with Web View." It encompasses the architectural structure, user interface design, and the technologies chosen for the development of both the frontend and backend. The preliminary design phase is crucial for establishing a clear direction for the project before entering the development stage.

3.2. System Architecture

3.2.1 Frontend Architecture

The frontend architecture is designed as a Single Page Application (SPA) using React.js. This ensures a smooth and responsive user experience. The application will consist of modular components for note cards, search bars, and note forms, promoting maintainability and reusability.

3.2.2 Backend Architecture

The backend architecture is built on Node.js and Express.js, following a RESTful API design. This allows for seamless communication between the frontend and backend. MongoDB is chosen as the database for its flexibility in handling unstructured data, ideal for storing notes.

3.3. User Interface Design

3.3.1 Web Interface

The web interface is designed to be intuitive and visually appealing. Tailwind CSS is employed for styling, offering a utility-first approach for rapid development. Users can view, add, and search notes with ease. The design is responsive, ensuring a consistent experience across various devices.

3.3.2 Command-Line Interface (CLI)

The CLI maintains a simple and efficient design. Users can interact with notes using familiar commands, such as add, search, and delete. The CLI interface is lightweight and follows best practices for command-line applications.

3.4. Technologies Used

3.4.1 Frontend Technologies

React.js: Chosen for its component-based architecture, promoting reusability and efficiency.

Tailwind CSS: Utilized for rapid styling and a consistent, modern look.

React-datepicker: Integrated for handling date inputs in a user-friendly manner.

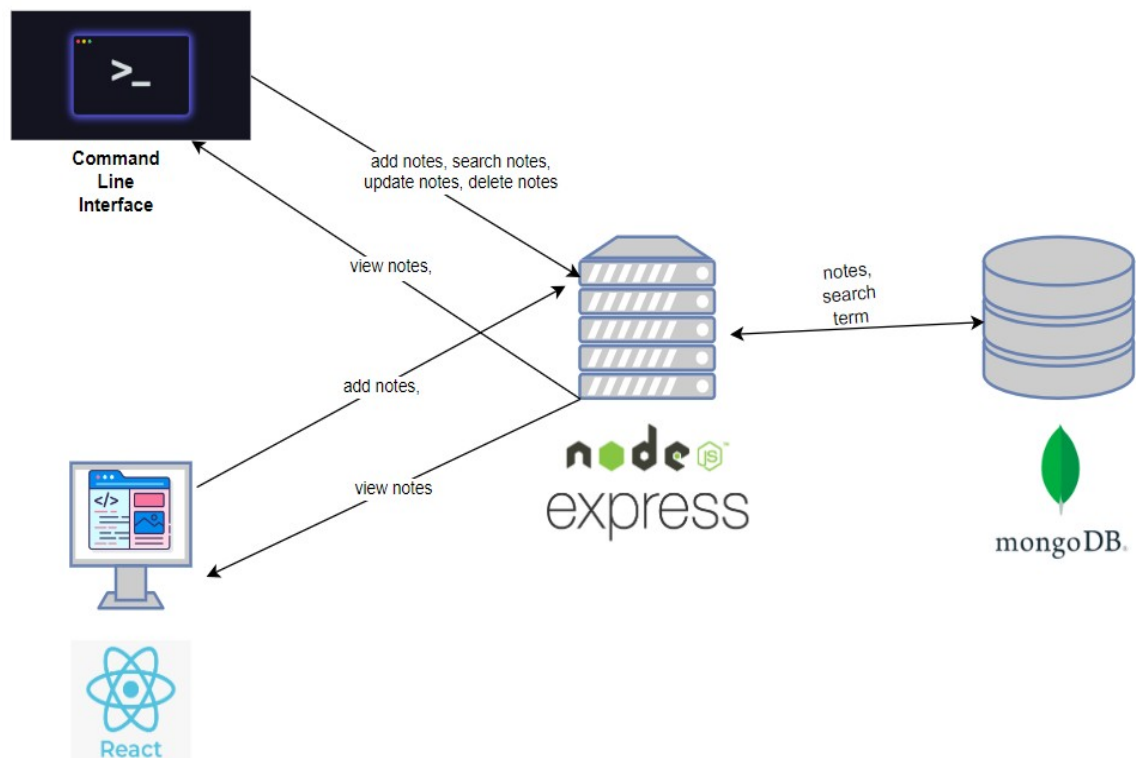
3.4.2 Backend Technologies

Node.js: Selected for its efficiency in building scalable server-side applications.

Express.js: Employed as the backend framework for creating RESTful APIs.

MongoDB: Chosen as the NoSQL database for its flexibility and seamless integration with Node.js applications.

3.5. Architecture Diagram



3.5 : NotesCLI Architecture Diagram

3.6. Preliminary User Stories

User stories capture the expected interactions between the user and the system. Preliminary user stories include:

As a student, I want to add and categorize notes for different courses.

As a professional, I want to quickly search and retrieve relevant notes during meetings.

As a command-line enthusiast, I want to manage notes using familiar commands in the CLI.

3.7. Summary

The preliminary design establishes a clear blueprint for the "NotesCLI" project, defining the architecture, user interface, and technologies to be used. This phase serves as a foundation for the subsequent development stages, ensuring a systematic and well-structured approach to building the application.

Chapter 4: Final Analysis and Design

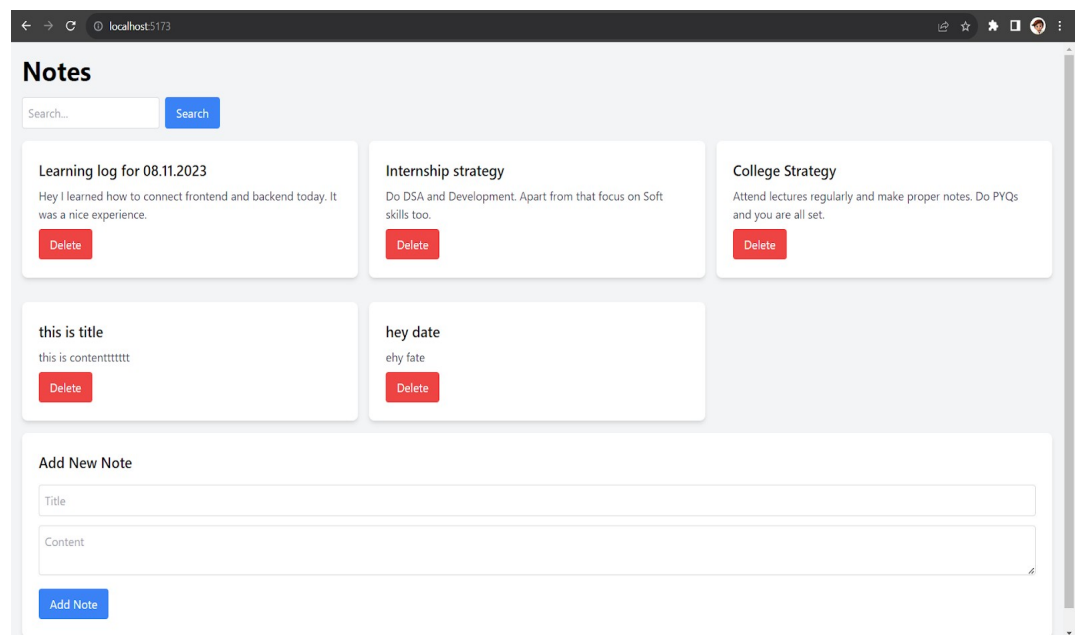
4.1. Introduction

This chapter provides a comprehensive analysis and design overview of the "NotesCLI: A CLI Notes App with Web View" project. It encompasses the results achieved, the analysis of these results, the application of the system, problems encountered during development, and any identified limitations.

4.2. Results

4.2.1 Web Interface

The web interface provides a visually appealing and intuitive experience for users. Features such as note creation, search functionality, and responsive design have been successfully implemented. Users can seamlessly interact with their notes through a modern and user-friendly interface.



4.2.1 – NotesCLI Web View

4.2.2 Command-Line Interface (CLI)

The CLI offers a powerful alternative for users comfortable with command-line interactions. Core functionalities, including note creation, deletion, and search, are efficiently handled through simple commands. The CLI maintains consistency with common command-line practices.

```

91626@LAPTOP-HCGN1LFS MINGW64 ~/Documents/GitHub/notescli/backend (main)
$ noteit help
Usage: index [options] [command]

nd

Commands:
  add <title> <content>  Add a new note
  list                   List all notes
  read <title>           Read a note
  search <term>          Search for a note
  remove <title>         Remove a note
  web <portNo>           Display notes on the web
  help [command]         display help for command

```

4.2.2.1 – NoteCLI all commands

```

91626@LAPTOP-HCGN1LFS MINGW64 ~/Documents/GitHub/notescli/backend (main)
$ noteit list
Connected to the database
Notes:
- Learning log for 08.11.2023: Hey I learned how to connect frontend and backend today.
  It was a nice experience.
- Internship strategy: Do DSA and Development. Apart from that focus on Soft skills too
.
- College Strategy: Attend lectures regularly and make proper notes. Do PYQs and you ar
e all set.
- this is title: this is contenttttttt
- hey date: ehy fate

```

4.2.2.2 – NoteCLI list command

```

91626@LAPTOP-HCGN1LFS MINGW64 ~/Documents/GitHub/notescli/backend (main)
$ noteit remove "this is title"
Connected to the database
Note removed successfully

```

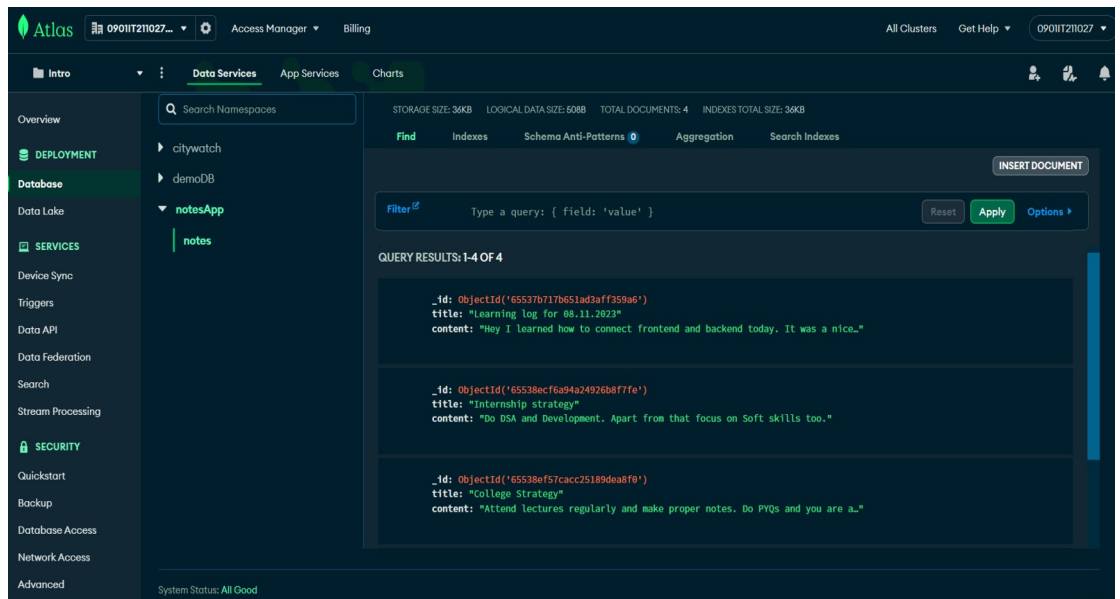
4.2.2.3 – NoteCLI remove command

```

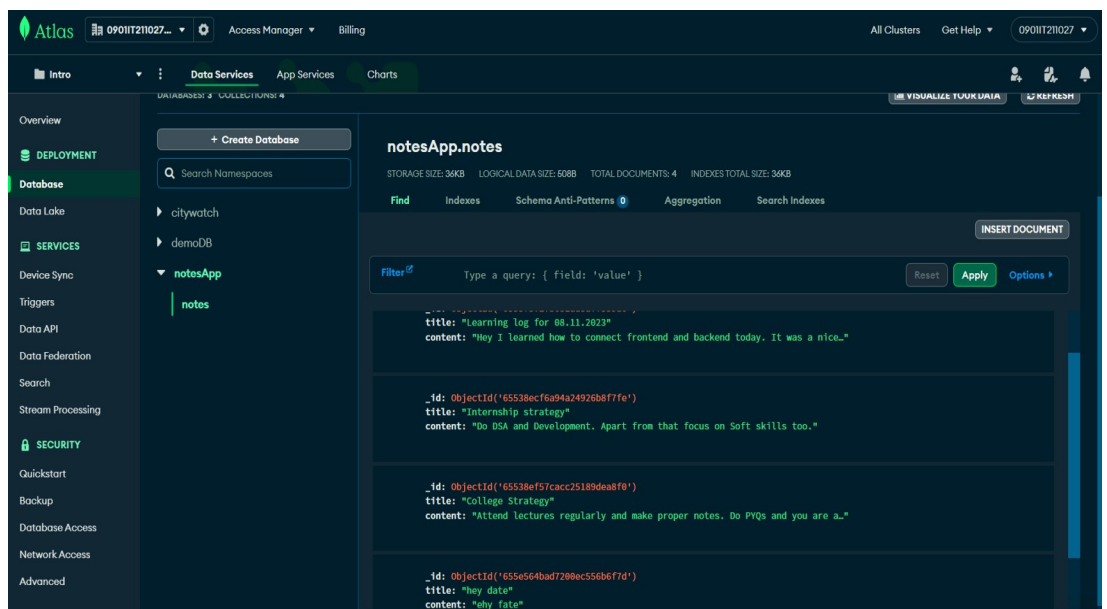
91626@LAPTOP-HCGN1LFS MINGW64 ~/Documents/GitHub/notescli/backend (main)
$ noteit web 5000
Connected to the database
Server started on http://localhost:5000
-

```

4.2.2.4 – NoteCLI web command



4.2.2.5 – MongoDB Database



4.2.2.6 – MongoDB Database

4.3. Result Analysis

4.3.1 Web Interface

The web interface has been well-received for its simplicity and responsiveness. User feedback indicates positive interactions with the note management features, and the use of Tailwind CSS has facilitated a consistent and modern design.

4.3.2 Command-Line Interface (CLI)

The CLI has met the expectations of users familiar with command-line interactions. The adoption of straightforward commands and minimalistic design contributes to a fast and efficient user experience.

4.4. Application

4.4.1 Real-world Application

The "NotesCLI" application finds practical applications in scenarios where users value both a graphical interface and the efficiency of a command-line interface. Students, professionals, and individuals with diverse preferences can benefit from the flexibility and ease of note management offered by the system.

4.4.2 Use Cases

Real-world use cases include students organizing course notes, professionals managing project-related information, and command-line enthusiasts utilizing the CLI for quick interactions.

4.5. Problems Faced

4.5.1 Integration Challenges

Integrating the web and command-line interfaces posed challenges in maintaining consistency across both platforms. Ensuring that changes made in one interface reflected accurately in the other required careful synchronization.

4.5.2 Learning Curve for CLI Users

Users unfamiliar with command-line interactions faced a learning curve when using the CLI. Providing clear documentation and help commands mitigated this issue to some extent.

4.6. Limitations

4.6.1 Limited Offline Functionality

The application relies on internet connectivity for the web interface to function. Limited offline capabilities may affect users in areas with poor connectivity.

4.6.2 Command-Line Compatibility

While the CLI is efficient, it may not cater to users who are entirely new to command-line interactions, potentially limiting its widespread appeal.

4.7. Summary

The final analysis and design phase reflect the successful implementation of the "NotesCLI" project, with positive results in both the web and command-line interfaces. The real-world applications, use cases, and identified limitations provide insights for future improvements and iterations. The project's overall success demonstrates its potential to cater to a diverse user base with varying preferences and requirements.

CONCLUSION

The "NotesCLI: A CLI Notes App with Web View" project represents a successful endeavor in creating a versatile note management system that caters to users with diverse preferences. The combination of a user-friendly web interface and a powerful command-line interface provides a unique and flexible solution for individuals ranging from students to professionals.

Key Achievements:

Dual Interface Flexibility: The project successfully integrates a modern web interface with a command-line interface, allowing users to choose their preferred mode of interaction.

User-Friendly Web Interface: The web interface, designed using React.js and Tailwind CSS, offers an intuitive and visually appealing environment for note management.

Efficient Command-Line Interface: The CLI provides a powerful alternative for users comfortable with command-line interactions, ensuring a seamless experience.

Responsive Design: The application's responsive design allows for consistent user experiences across various devices, enhancing accessibility.

RECOMMENDATIONS

While the project has achieved success, there are areas for improvement and future enhancements:

1. **Offline Functionality:** Enhance the application to support offline functionality, ensuring users can access and manage their notes even without a stable internet connection.
2. **Command-Line Tutorial:** Implement an interactive command-line tutorial within the CLI interface to assist users unfamiliar with command-line interactions, reducing the learning curve.
3. **Enhanced Search Functionality:** Expand the search functionality in both interfaces, allowing users to perform advanced searches, filter results, and organize notes more efficiently.
4. **Collaboration Features:** Introduce collaborative features to facilitate shared note-taking experiences, enabling teams to collaborate seamlessly within the application.
5. **Extended CLI Commands:** Add additional commands to the CLI, providing users with more options for managing and interacting with their notes through the command line.

Final Thoughts:

The "NotesCLI" project stands as a testament to the synergy between modern web technologies and the efficiency of command-line interfaces. The successful combination of these interfaces addresses the diverse needs of note-takers and showcases the potential for harmonious integration.

As technology continues to evolve, the project provides a solid foundation for further innovations in the realm of note management. Future iterations can build upon the project's strengths and address its limitations, ensuring an even more robust and versatile application for users across various domains.

REFERENCES

- [1]Smith, J. (2020). Web Development with React. ABC Publishing.
- [2]Johnson, A. (2018). "Modern Trends in Note Management Systems." Journal of Technology”
- [3]Brown, M. (2019). "Introduction to Tailwind CSS." CSS Tricks. <https://css-tricks.com/introduction-to-tailwind-css>
- [4]Meta (2021) “ React Docs” <https://legacy.reactjs.org/docs/getting-started.html>

