

# **PROJECT NAME: NOTE TAKING APP**

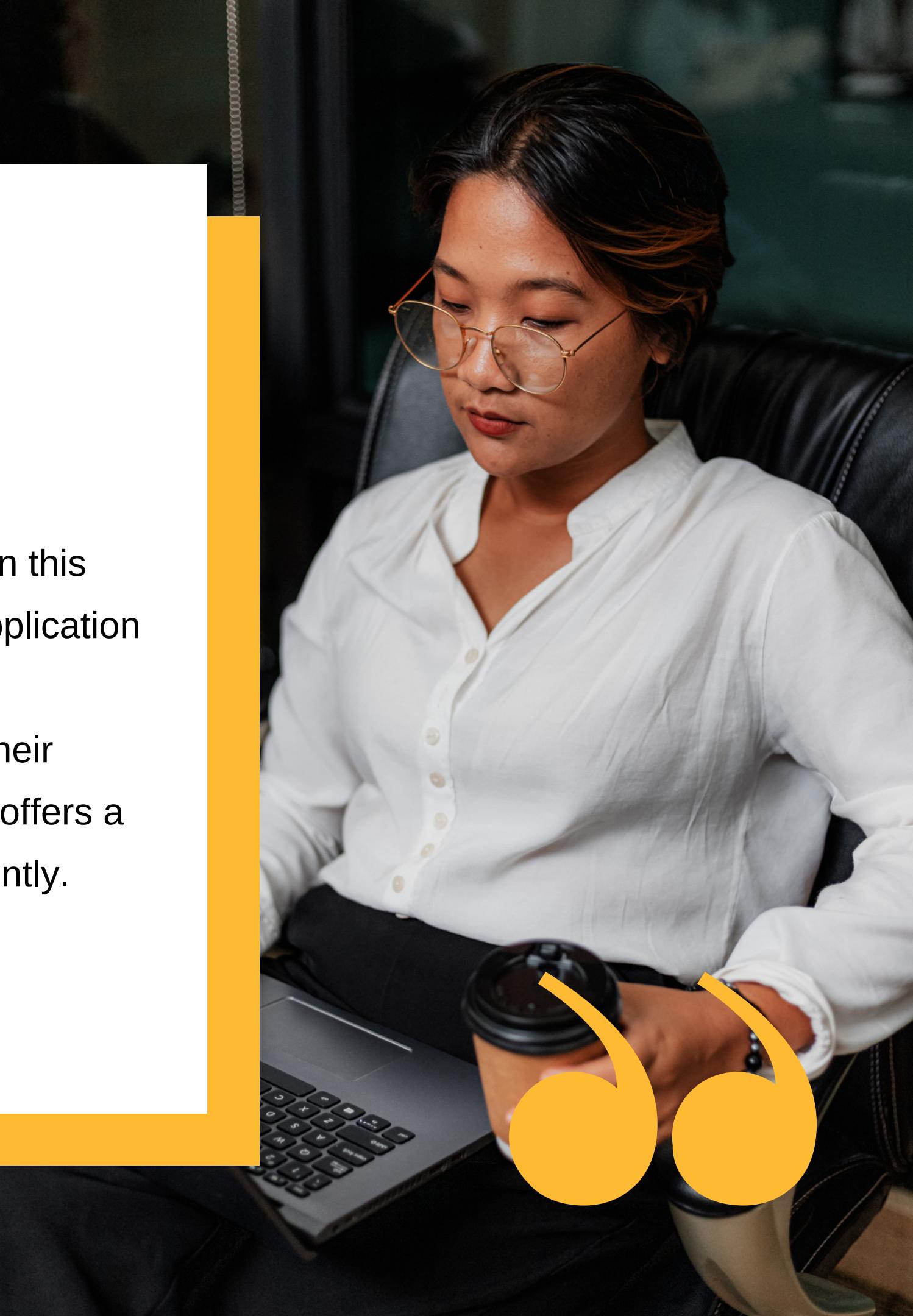
**Submitted By:**  
**Udit**  
**2110992026**  
**G19**

**Submitted To:**  
**Ms. Pritpal Kaur**

# INTRODUCTION

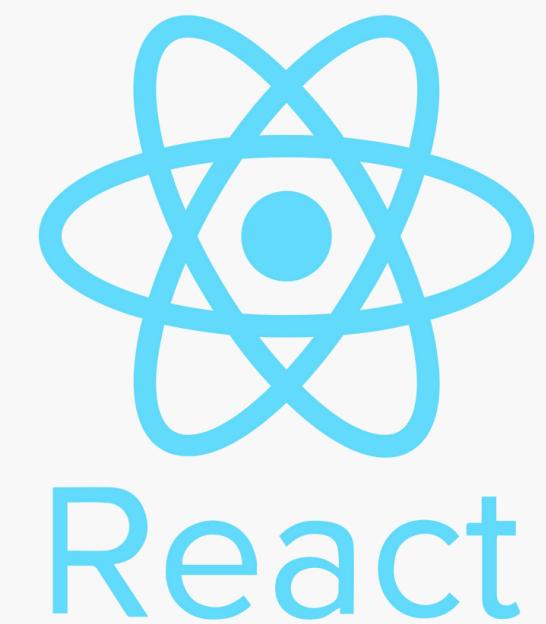
Welcome to the documentation of our React-based Note-Keeping App. In this documentation, we will explore the key components that make up our application and their respective functionalities.

Our Note-Keeping App is designed to help users capture and organize their thoughts, ideas, and important notes in a simple and intuitive manner. It offers a user-friendly interface and a range of features for managing notes efficiently.



# TECHNOLOGIES USED:

- HTML
- CSS
- JavaScript
- React



# KEY COMPONENTS

**01**

App component

**02**

AddNote component

**03**

Header component

**04**

Search component

**05**

Note component

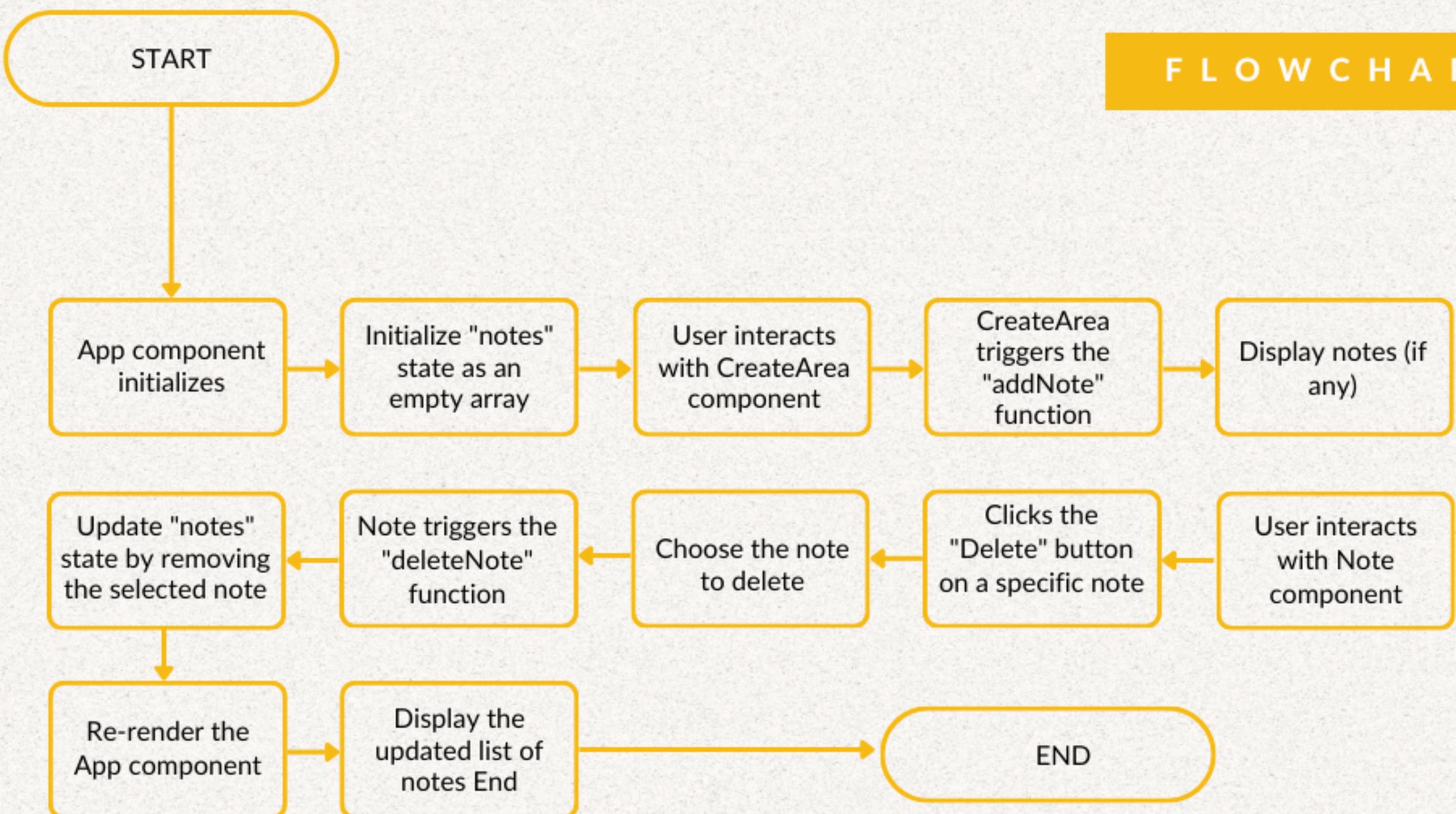
**06**

index.js

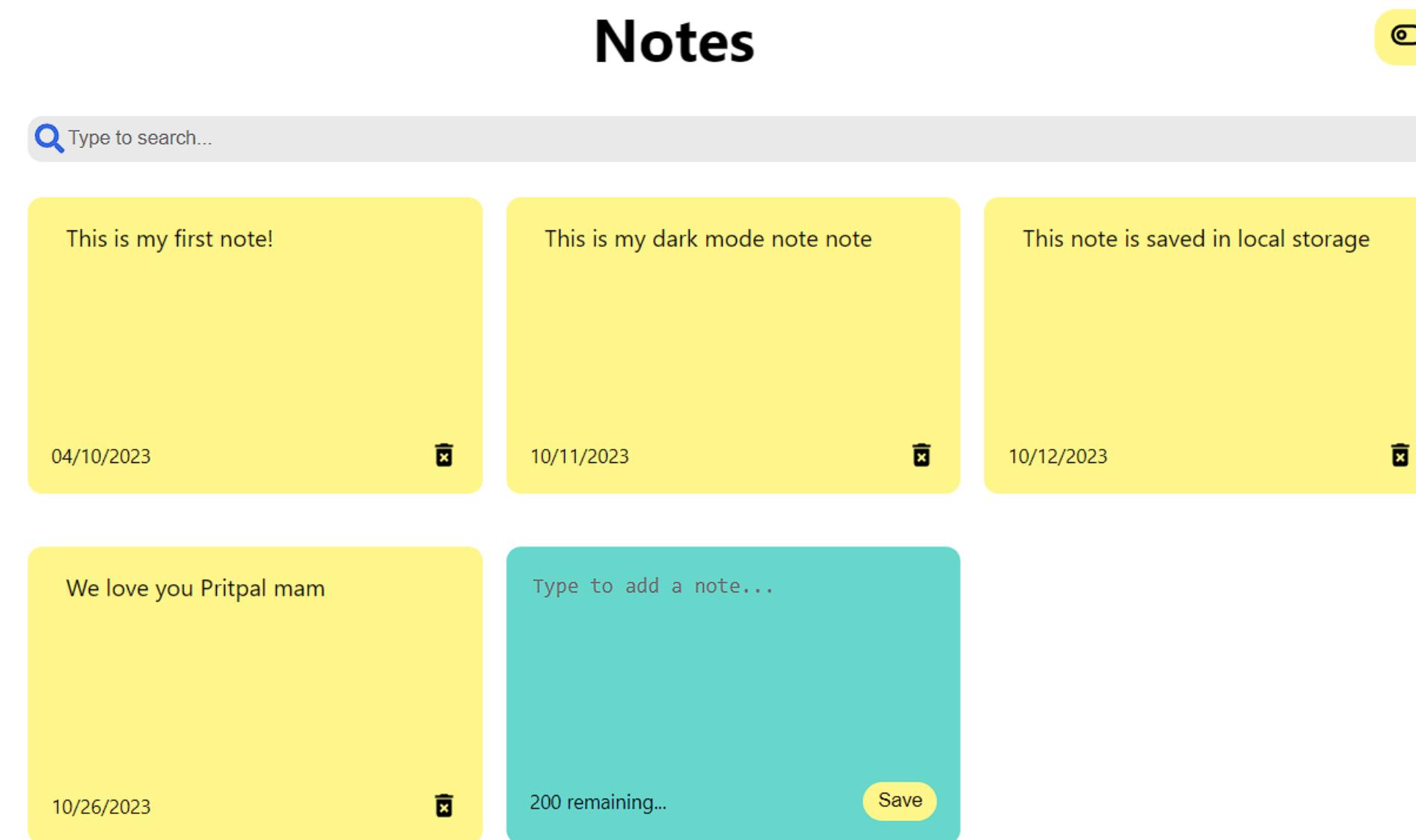


# FLOW CHART

FLOW CHART



# App.js



The "App" component is the core component of the note-keeping application. It manages the state of notes, allowing users to add and delete notes. It also serves as the container for other components like "Header," "Footer," "Create Area," and a list of "Note" components.

# AddNote.js

The screenshot shows a web application titled "Notes". At the top right is a yellow circular icon with a black "O" inside. Below the title is a search bar with a magnifying glass icon and the placeholder text "Type to search...". The main area displays five notes cards:

- A yellow card with the text "This is my first note!" and the date "04/10/2023" at the bottom right, with a small trash icon.
- A yellow card with the text "This is my dark mode note note" and the date "10/11/2023" at the bottom right, with a small trash icon.
- A yellow card with the text "This note is saved in local storage" and the date "10/12/2023" at the bottom right, with a small trash icon.
- A yellow card with the text "We love you Pritpal mam" and the date "10/26/2023" at the bottom right, with a small trash icon.
- A teal card with the placeholder text "Type to add a note..." and the word "Save" in a yellow button at the bottom right. It also shows "200 remaining..." and has a small trash icon at the bottom right.

The "CreateArea" component is responsible for enabling users to create new notes. It offers a form for input, including a title and content field, and a button to submit the note. The component manages its own state and triggers a callback function when a note is added.

# Header.js

## Notes



The "Header" component serves as the application's header, displaying the application's title along with a stylized icon. It provides a visual identifier for the application.

# Search.js

 Type to search...

The "Search" component is responsible for displaying all the notes containing the Searched text.

# Note.js

We love you Pritpal mam

10/26/2023



The "Note" component represents an individual note within the note-keeping application. It displays the note's title, content, and a delete button, allowing users to remove a specific note.

1. In conclusion, we've explored the key components and functionality of the React Note-Keeping App.
2. We've learned how the App component serves as the central hub, managing notes and interactions between other components.
3. The AddNote component empowers you to create and organize your notes with ease.
4. The Header and Footer components enhance the app's visual appeal and relevance with a dynamic copyright
5. Individual notes are represented by the Note component, offering a seamless way to view and delete notes.
6. The index.js file serves as the entry point, connecting the app to the HTML DOM.
7. We've also examined the application flowchart to understand the user interactions and data flow.
8. This app is designed to efficiently organize your thoughts and ideas, making it a valuable tool for productivity.
9. We invite you to explore the React Note-Keeping App and experience the benefits it offers in managing your notes effectively.
10. Thank you for your time and attention throughout this presentation.

# **THANK YOU!**

