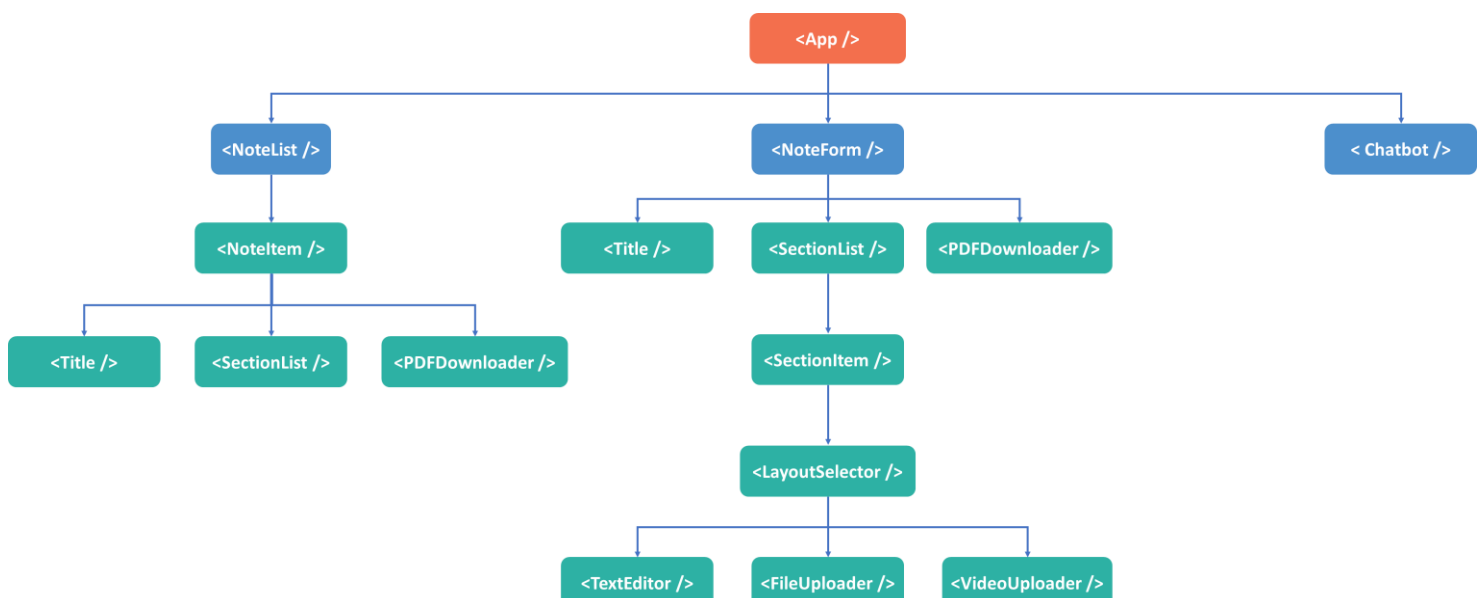


NoteBot App:



Component Hierarchy:

- App (top-level component)
 - NoteList
 - NoteItems
 - NoteItem
 - Title
 - SectionList
 - PDFDownloader
 - NoteForm
 - Title
 - SectionList
 - SectionItems
 - LayoutSelector
 - TextEditor
 - PDFUploader
 - VideoUploader
 - PDFDownloader
 - Chatbot



Data Flow:

1. User interacts with the App component.
2. NoteList component fetches the list of notes from the backend API and renders NoteItems.
3. NoteItems display each note's title and relevant information.
4. Clicking on a NoteItem triggers the rendering of the SectionList for that specific note.
5. Users can create a new note by filling in the Title and SectionList within the NoteForm component.
6. SectionList fetches the sections for the selected note from the backend API and renders SectionItems.
7. SectionItems display the section's content and provide layout options through the LayoutSelector component.
8. Users can choose a layout, and based on their selection, the respective components (TextEditor, PDFUploader, VideoUploader) are rendered within the SectionItem.
9. Users can interact with the TextEditor component to modify the note's content.
10. Users can upload PDFs and videos using the respective uploader components.
11. Users can interact with the Chatbot component to seek assistance from ChatGPT.
12. The PDFDownloader component enables users to download notes as PDF files.

→ Regarding the extension of adding a "Courses" module for publishing notes, by using additional components and functionalities:

- CourseList: Display a list of available courses.
- CourseItem: Render information about each course, including the course name and rating.
- CourseDetails: Provide additional details about a specific course, including the published notes.
- Rating: Allow users to rate published notes.

React would be used for the front-end development, including the components and UI rendering.

Node.js would be used for the back-end to handle API requests, data storage, and communication with the database.