**Phase 3: Development**

In this phase, the Notes App was developed using the design and research done in previous phases. Both frontend and backend parts were implemented in parallel to ensure a smooth integration. The goal was to create a fully functional CRUD application where users can create, view, and delete notes with proper frontend-backend synchronization.

**Frontend Development**

The frontend was built using **React**, focusing on a modular structure. Functional components were created for different parts of the UI:

* **App**: Root component managing layout and state.
* **NoteList**: Displays all notes in a list format.
* **NoteItem**: Represents an individual note with delete functionality.
* **AddNoteForm**: A form to add new notes.

React hooks like useState and useEffect were used to manage and update state dynamically. API communication was handled using **Axios**, which sent HTTP requests to the backend.

After each operation—like adding or deleting a note—the state was updated based on API responses to reflect changes in the UI.

The UI was styled using **Material UI**, providing a responsive, clean, and consistent design.

**Backend Development**

The backend was developed using **Node.js** with the **Express.js** framework. RESTful APIs were implemented to support:

* GET /api/notes — Fetch all notes
* POST /api/notes — Add a new note
* DELETE /api/notes/:id — Delete a note by ID

Middleware like cors and Express's built-in JSON parser were used to handle cross-origin requests and parse incoming data.

For simplicity, notes were stored in-memory during development. The structure was designed to easily integrate a database such as **MongoDB** later for persistence.

**Frontend-Backend Integration**

Integration was done using **Axios** in React to send requests to the Express backend. The application used API responses to synchronize the frontend state with backend data.

To improve responsiveness, **optimistic UI updates** were implemented—updating the UI immediately on user actions and then confirming with the server.

**✅ Output:**

A fully functional Notes App capable of:

* Creating notes
* Displaying notes
* Deleting notes

All features are synchronized between frontend and backend using a clean and modular architecture.