

Day 7 task 2 – Work Summary & Steps Done

Goal:

Implement the Statistics Page and Toast Notification System in the frontend of the AI Class Assistant project.

1-Set up Navigation & Routing

File Updated: App.js Added React Router navigation between:

- `/` → Flashcards page
- `/stats` → Statistics page Created Navigation component with active link highlighting.

2. Added Statistics Page (StatsPage.js)

Purpose: Show user's quiz & flashcard progress.

Features implemented: - Fetch progress data from backend API.

- Display summary cards (Quizzes Generated, Flashcards Reviewed, Correct Ratio).
- Display a Bar Chart using Recharts. Dependencies installed: `npm install recharts bootstrap`

3. Created api.js for API Calls

Purpose: Centralize API communication. Code example: `import axios from 'axios'; const api = axios.create({ baseURL: 'http://localhost:5000' }); export default api;`

4. Implemented Toast Notification System

File Added: Toast.js

Purpose: Show success or error messages

. Components: - Toast → Single toast message UI - ToastContainer → Displays multiple toast messages - useToast hook → Functions to add/remove toasts

ToastProvider Context created to make addToast available app-wide.

5. Wrapped App in ToastProvider File

Updated: index.js React 18

Fix: Changed from ReactDOM.render() to createRoot(). Wrapped App inside so all components can use addToast.

6. Connected Toasts in StatsPage

On successful data fetch → Show 'Statistics loaded successfully'. On error → Show 'Failed to load statistics'.

7. Testing Steps

1. Start backend.
2. Start frontend (`npm start`).
3. Visit `/` and `/stats`.
4. Verify stats cards, bar chart, and toast notifications. Result Fully functional statistics dashboard. Toast notifications integrated. Clean API calls via api.js. Navigation ready for scaling