

Hi Prabhu,

I hope you're doing well.

As part of today's learning, I focused on backend migration and API integration to connect the React.js frontend with a Spring Boot backend. We successfully migrated the backend from **Gradle to Maven**, configured the `pom.xml` with required dependencies, and set up an in-memory **H2 database** using `schema.sql` and `data.sql` for development.

We also refactored the `User` model to `AppUser` to avoid conflicts with SQL reserved keywords, and updated related components like `UserRepository`, `UserController`, and `PaymentController`.

On the frontend, I integrated real API data into the `PaymentSummary.js` React component by fetching payment history from the backend using the endpoint:
`http://localhost:8080/api/payments?userId=username`

This replaced the earlier use of dummy data and made the app dynamic and responsive to backend changes. We also resolved several backend errors during migration to ensure smooth communication between frontend and backend.

This session helped me strengthen my understanding of:

1. Backend build systems – Gradle to Maven migration
2. Spring Boot with Spring Data JPA and H2 DB configuration
3. RESTful API development and data exposure
4. Full-stack integration using React and Spring Boot with real-time data fetching and error handling

Now, the application is fully connected end-to-end, and I'm looking forward to building more features on top of this integrated structure.

Best regards,
Yashwanth S