

Executive Summary

Hive Space is a personalized study application designed to help both individual and collaborative learning. The application allows students to organize notes, set goals, track progress, and maintain consistent study habits. Users can create study groups with friends, communicate through built-in chat, and share notes within their groups. Hive Space also integrates AI tools that generate summaries and quizzes from study materials to help students review and test their understanding . By combining collaboration, productivity tools, and AI-assisted learning in a single platform, Hive Space provides students with an interactive environment that supports more effective studying.

Background

Students today lack a dedicated tool to bring peer collaboration and accountability into their study routines. A platform that has everything a student needs local note storage, study groups, goal setting, AI- quizzes and summaries. Hive Space removes the difficulty of having multiple apps and keeps students focused. Improving student motivation through competitive leaderboards and shared group goals, encouraging consistent study habits for better academic outcomes.

While applications like Duolingo, Forest, Good Notes, and Strava each solve one part of the problem, no single platform has them all. Hive Space fills this gap by combining a social network, group management, AI-assisted studying, and goal tracking into one place.

Experimental Setup

System Specifications

- Mobile application for note storage, study goals, and productivity tracking
- AI-generated summaries and quizzes from study notes
- Group collaboration and goal settings
- Tracking of study sessions, streaks, and performance statistics

Experimental Setup

System Constraints

- All authentications traffic must be secured using HTTPS
- Authentication tokens must expire and be validated
- System must comply with FERPA privacy requirements
- Backend must maintain fast response times using caching mechanisms
- AI-generated summaries and quizzes must return results within an acceptable response time
- Quiz and summaries must be saved locally to prevent storage issues and data loss

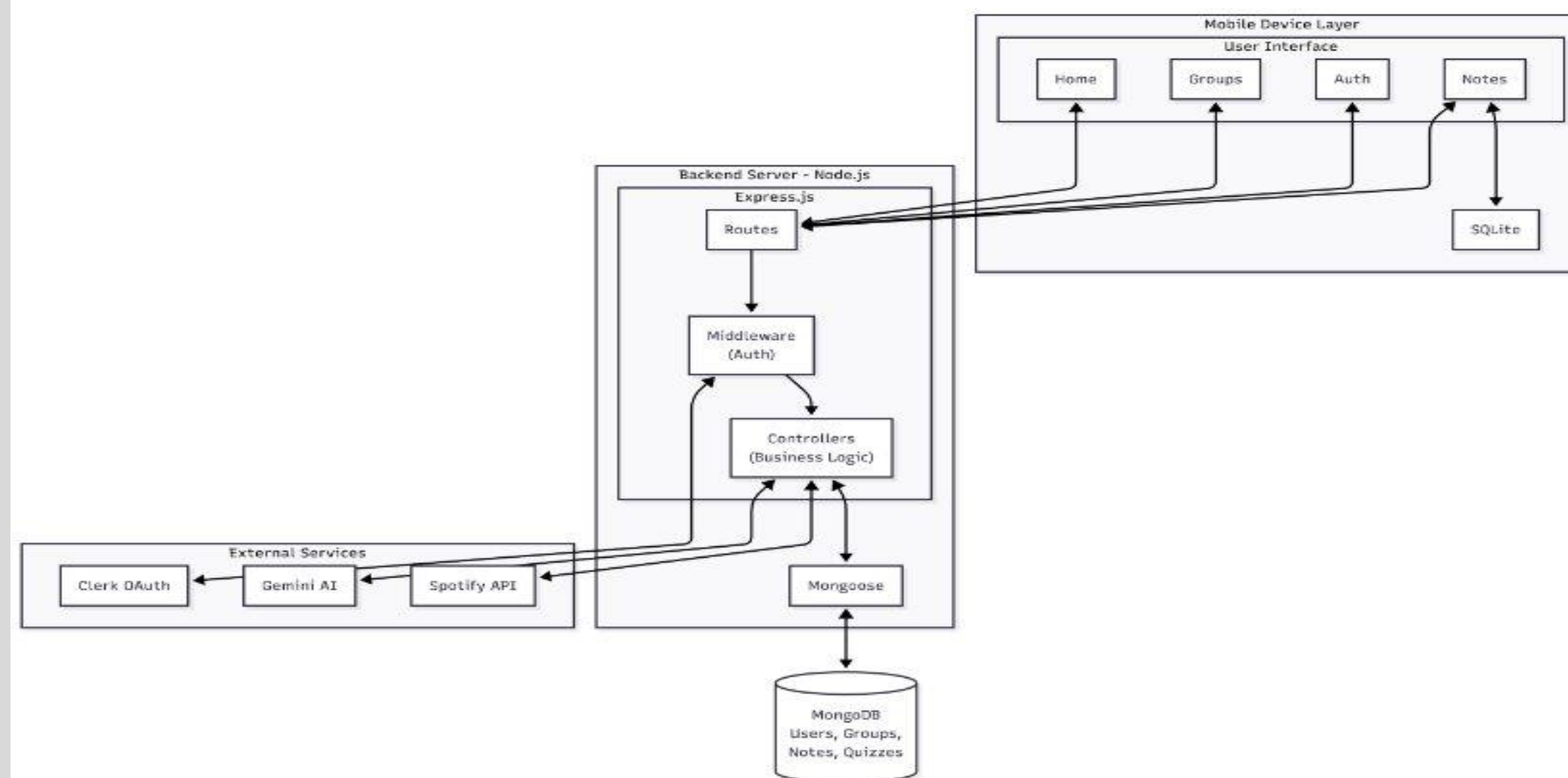


Figure 1. System Design

Experimental Test Plan

Authentication:

- To handle user authentication, we use Clerk for token generation, expiration and validating all requests to the backend.

AI quizzes and summaries:

- For AI features we use google Gemini API. All requests are made async through the backend, generated quizzes are saved to MongoDB so group members can access them.

Offline Notes Storage:

- All notes are stored locally on user device to using SQLite to ensure user privacy, giving user access to their notes even if they are offline.

Social Network:

- Groups have real-time WebSocket chat, group leaderboards, and groups goals to allow member to track each other's progress and stay connected.

App Performance:

- Authentication tokens are cached on-device removing the need to login every time. Tracking individual service response timing to see end to end performance.

Experimental Results

Server Response Time

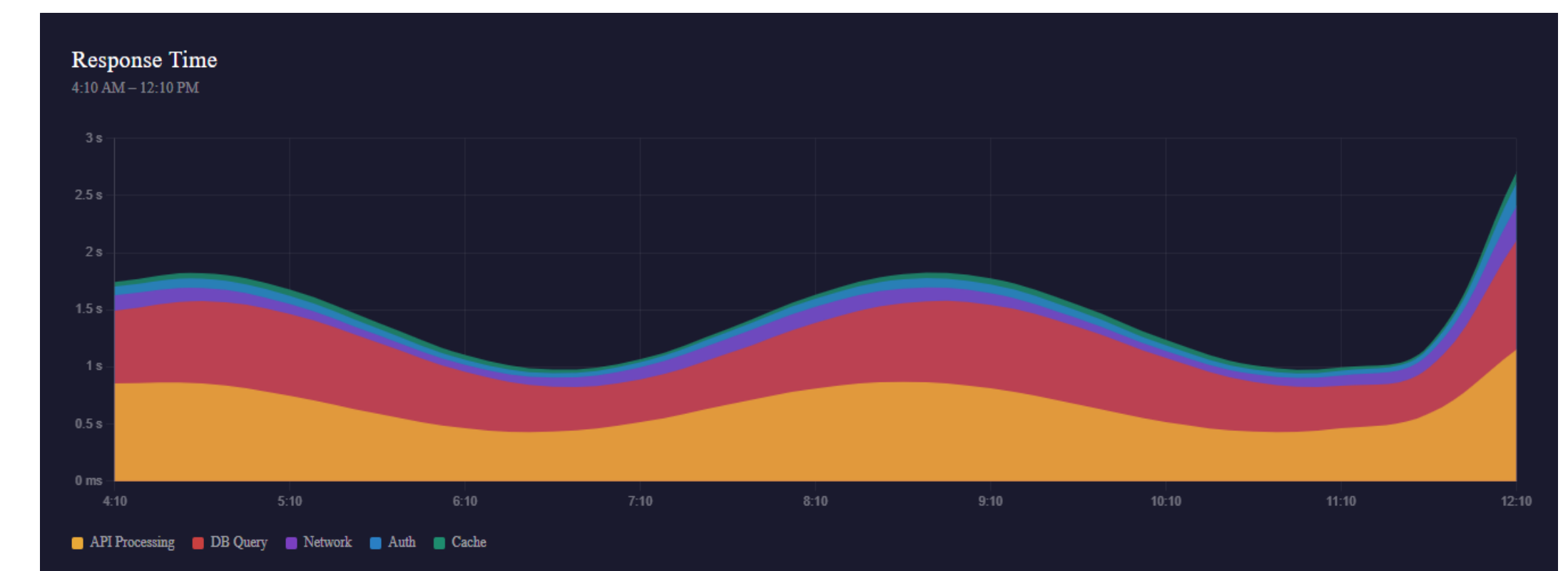


Figure 2. Server Response Times

	Layer	Avg
1	API processing	650 ms
2	DB Query	550 ms
3	Network	100 ms
4	Auth	60 ms
5	Cache	40 ms
6	Total	1400 ms

Table 1. Average Response Times

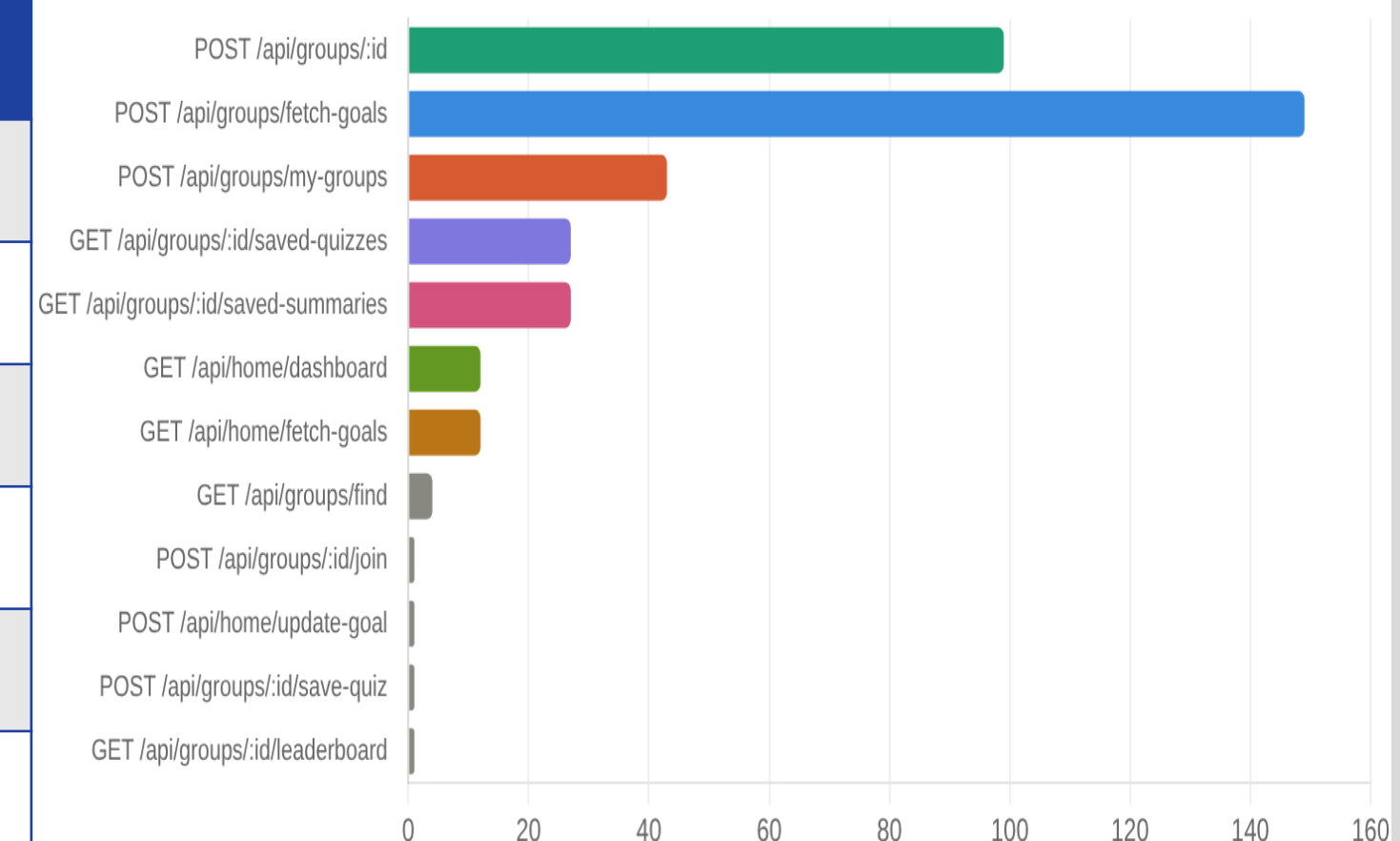


Figure 3. API calls observed

Conclusions

The Hive Space system successfully meets the original goal of creating a unified platform that supports study productivity, collaboration, and AI-assisted learning. The design addresses key specifications and constraints, including secure authentication, reliable data storage, and efficient response times. Throughout the development process, the team gained valuable experience integrating AI services with a full-stack mobile application while ensuring usability, security, and system performance. The results shows that Hive Space can effectively support note organization, study tracking, and collaborative learning within a single platform. Future work will focus on enhancing AI-generated study tools, improving personalization features, and expanding collaboration capabilities. We acknowledge the support and guidance of our professor, **Dr. Chenxi Wang**, throughout the development of this project.

References

1. Clerk. "Frontend API Reference." *Clerk Documentation*, 2025-11-10, clerk.com/docs/reference/frontend-api/2025-11-10/description/versions.
2. Clerk. "Backend API Reference." *Clerk Documentation*, 2025-11-10, clerk.com/docs/reference/backend-api/2025-11-10/description/introduction.
3. Spotify. "Android SDK Documentation." *Spotify for Developers*, developer.spotify.com/documentation/android.
4. Render. "Render Documentation." *Render*, render.com/docs.
5. OpenJS Foundation. "Express 5.x API Reference." *Express.js*, expressjs.com/en/5x/api.html.