

PeerLearn Labs

PeerLearn — Sprint Plan (v2)

Date: 2025-09-25

1. Purpose

This document defines the sprint-based development plan for PeerLearn. It lists tasks, priorities, and expected deliverables aligned with the MVP use cases.

2. Sprint Breakdown

Sprint 1 (Weeks 1–2) — Foundations

Priority: High

Tasks: Set up backend with authentication, configure SQL database for users, connect MongoDB for notes, implement room CRUD, and set up the React frontend.

Deliverable: Login, room creation, persistence layer.

Sprint 2 (Weeks 3–4) — Real-Time Collaboration

Priority: High

Tasks: Integrate real-time chat/presence, collaborative editor with CRDT sync, awareness indicators.

Deliverable: Multi-user note editing with chat.

Sprint 3 (Weeks 5–6) — Flashcards & Quizzes

Priority: Medium

Tasks: Implement flashcard generation, quiz feature, and results storage.

Deliverable: Flashcards and quizzes with stored results.

Sprint 4 (Weeks 7–8) — Progress Tracking & Gamification

Priority: Medium

Tasks: Add XP system, display progress dashboard, and issue badges.

Deliverable: Progress tracking visible in dashboard.

Sprint 5 (Weeks 9–10) — Security & Hardening

Priority: High

Tasks: Enforce HTTPS, secure cookies, add rate limiting, configure key storage.

Deliverable: Hardened security baseline.

Sprint 6 (Weeks 11–12) — Optional Voice Study

Priority: Low

Tasks: Implement voice chat prototype using WebRTC.

Deliverable: Prototype voice chat.

Sprint 7 (Weeks 13–14) — Polish & Demo Prep

Priority: Medium

Tasks: UI polish, documentation, bug fixing.

Deliverable: Demo-ready app.

3. Priority Legend

High: Must be implemented for MVP success.

Medium: Important, improves usability but not strictly MVP.

Low: Optional or experimental features.

4. References

References: ASP.NET Identity, EF Core, MongoDB, SignalR, Yjs, OpenAI API, Hugging Face, OWASP HttpOnly, rate limiting, data protection, and coturn.

