

Version & Metadata

Version: v1.0

Title: PeerLearn — Sprint Plan

Project: PeerLearn (Personal project)

Author: Aleksandar Ivanov

Date: 2025-09-11 (Europe/Amsterdam)

Status: Draft

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 .NET 8 LTS backend project with Identity & JWT authentication. [ASP.NET Core Identity](#)
 - Configure SQL database for Users/Auth (university requirement). [EF Core SQL Server provider](#)
 - Connect MongoDB for Rooms & Notes collections. [MongoDB .NET driver](#)
 - Implement CRUD for Rooms.
 - Set up React app with routing & login page.
- **Deliverable:** Basic login, room creation, persistence layer.

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

- **Priority:** High
- **Tasks:**
 - Integrate SignalR hub for chat/presence. [SignalR intro](#)
 - Add Slate editor and bind with Yjs CRDT.
 - Configure `y-websocket` server for collaborative editing. [y-websocket docs](#)
 - Implement awareness (user cursors, online presence).
- **Deliverable:** Multi-user note editing with chat.

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

- **Priority:** Medium
- **Tasks:**
 - Backend endpoint for generating flashcards via AI. [OpenAI API Hugging Face](#)
 - Flashcard UI in React.
 - Quiz-taking feature with scoring.

- Store quiz results in MongoDB.
- Deliverable: AI-powered flashcards, quizzes with stored results.

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

- **Priority: Medium**
- Tasks:
 - Add XP & streak system in backend.
 - Display progress dashboard in frontend.
 - Issue badges for milestones (consistency, participation).
- Deliverable: Progress tracking visible in dashboard.

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

- **Priority: High**
- Tasks:
 - Enforce HTTPS + HSTS.
 - Secure JWTs with HttpOnly cookies. [OWASP HttpOnly](#)
 - Add RateLimiting middleware. [Rate limiting](#)
 - Configure Data Protection key storage. [Data Protection](#)
- Deliverable: Hardened security baseline.

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

- **Priority: Low**
- Tasks:
 - Implement WebRTC signaling through SignalR.
 - Deploy TURN server (coturn). [coturn](#)
 - Add frontend controls for joining voice channels.
- Deliverable: Prototype voice chat for study rooms.

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

- **Priority: Medium**
- Tasks:
 - UI/UX polish in React.
 - Documentation (README, user guide, API docs).
 - Bug fixing, refactoring.
- Deliverable: Demo-ready PeerLearn 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

- ASP.NET Core Identity: <https://learn.microsoft.com/en-us/aspnet/core/security/authentication/identity?view=aspnetcore-9.0>
- EF Core SQL Server provider: <https://www.nuget.org/packages/Microsoft.EntityFrameworkCore.SqlServer/>

- MongoDB .NET driver: <https://www.mongodb.com/docs/drivers/csharp/>
- SignalR intro: <https://learn.microsoft.com/en-us/aspnet/core/signalr/introduction?view=aspnetcore-9.0>
- y-websocket docs: <https://docs.yjs.dev/ecosystem/connection-provider/y-websocket>
- OpenAI API: <https://platform.openai.com/docs/api-reference>
- Hugging Face endpoints: <https://huggingface.co/docs/inference-endpoints/index>
- OWASP HttpOnly: <https://owasp.org/www-community/HttpOnly>
- Rate limiting: <https://learn.microsoft.com/en-us/aspnet/core/performance/rate-limit?view=aspnetcore-9.0>
- Data Protection: <https://learn.microsoft.com/en-us/aspnet/core/security/data-protection/configuration/overview?view=aspnetcore-9.0>
- coturn: <https://github.com/coturn/coturn>