

StudySync

Team Name: **Painguin**

Team Members:

Md Mahir Faysal	East West University
Md. Saidul Islam Rifat	Rajshahi University of Engineering & Technology

1. Introduction & Problem Statement

In many academic settings, students often face challenges in effective self-assessment and collaborative learning outside traditional classroom hours. Existing online platforms may offer static content or lack the specific interactive features needed for dynamic question-setting, real-time performance analysis, and peer-to-peer knowledge exchange. The current methods for students to clarify doubts or understand recurring patterns in questions are often informal, inefficient, and lack structured feedback.

Our project, **StudySync**, aims to address these gaps by providing an interactive web-based platform that empowers question setters (e.g., educators, senior students) to create and manage quizzes, offers students a dynamic environment for self-assessment, and facilitates a structured student-led Q&A forum for collaborative doubt resolution.

2. Solution: **StudySync** Web Application

StudySync will be a web application designed to enhance the learning experience through three core pillars:

1. **Streamlined Question & Exam Management:** Allow designated users to set, schedule, and activate quizzes/exams with ease.
 2. **Actionable Performance Insights:** Provide interactive visual feedback on question patterns for setters, enabling them to understand common student misconceptions.
 3. **Student-Led Collaborative Q&A:** Create a dedicated space where students can post confusions and receive answers from their peers.
-

3. Key Features

For the Therap JavaFest competition, we will focus on implementing the following core features as our Minimum Viable Product (MVP):

A. User Management & Roles:

- Secure User Registration and Login.
- Two primary user roles:
 - **Question Setter:** Can create/manage questions and exams, view detailed results.
 - **Student:** Can take exams, participate in Q&A.

B. Question & Exam Management (Setter Functionality):

- **Question Creation:** Question Setters can create new multiple-choice questions (text-based questions with predefined options and a single correct answer).
 - Ability to specify question text, options (A, B, C, D), and the correct answer.
- **Exam Creation:** Question Setters can create new exams/quizzes.
 - Assign a title and description to the exam.
 - Add existing questions to an exam.
- **Exam Status Control:** Question Setters can change the status of an exam (e.g., DRAFT, ACTIVE, COMPLETED).
 - An "Activate Exam" function to make it available for students.
 - A "Mark as Completed" function.

C. Student Exam Taking:

- Students can view a list of all ACTIVE exams.
- Students can select an exam and attempt it.
- Upon submission, students receive their score and a simple pass/fail indication.

D. Basic Exam Results & Pattern Analysis (Setter Functionality):

- For any COMPLETED exam, the Question Setter can view:
 - The total number of students who attempted the exam.
 - The average score for the exam.

- **Question-wise Performance Analysis:** For each individual question, a visual representation (e.g., simple bar chart) showing the percentage of students who answered it correctly versus incorrectly. This will help identify "tricky" questions or common learning gaps.
- **Implementation Note:** Data for charts will be prepared on the Spring Boot backend and rendered using a simple JavaScript charting library (e.g., Chart.js) on the frontend.

E. Student-Led Asynchronous Q&A Forum:

- **Post Confusion:** Students can post new "confusion" questions, potentially tagged by relevant exam or topic.
 - Each question will have a title, description, and the student who posted it.
- **Answer/Comment:** Any student can provide an answer or comment on an existing "confusion" question.
- **View Q&A:** Students can browse all posted questions and their respective answers in a forum-like structure.
 - Ability to sort by most recent or most active.
- **Implementation Note:** This feature will be asynchronous (similar to a standard web forum), meaning new answers appear upon page refresh, simplifying the initial development complexity.

Future Enhancements (Beyond MVP, for future phases or stretch goals):

- Multiple question types (True/False, fill-in-the-blanks).
- Timed exams.
- Detailed student-specific learning curve analytics across multiple exams.
- Real-time notifications for new Q&A posts or answers (using WebSockets).
- Integration of AI for automated question generation or feedback.
- Ability to attach images/files to questions or Q&A posts.

4. Technology Stack

In line with Therap JavaFest requirements and best practices for modern web development, we propose the following stack:

- **Backend:**
 - **Language:** Java
 - **Framework:** Spring Boot (RESTful APIs).
 - **Database:** PostgreSQL (or MySQL) – a powerful and widely-used relational database.
 - **ORM:** Spring Data JPA with Hibernate.
 - **Security:** Spring Security (for user authentication and authorization).

- **Build Tool:** Maven.
- **Frontend:**
 - **Core Technologies:** HTML, CSS, JavaScript.
 - **Framework/Templating:** React **OR** Thymeleaf with plain JavaScript/jQuery for dynamic elements.]
 - **Charting Library:** Chart.js (a simple yet powerful JavaScript library for charts).
 - **Styling Framework (Optional):** Bootstrap or Tailwind CSS (for responsive and quick UI development).