**Project Design Phase**

**Problem – Solution Fit Template**

|  |  |
| --- | --- |
| **DATE:** | **26-06-2025** |
| **Team ID :** | **LTVIP2025TMID58052** |
| **Project Name :** | **LearnHub: Your Center for Skill Enhancement** |

**Instructions:**

* Each section addresses a key problem statement previously identified during user research for LearnHub.
* The "Problem Statement" reiterates the core user issue.
* The "Proposed Solution Fit" details how LearnHub’s specific components, features, and architectural decisions directly solve that problem.

**1. Problem Statement: Lack of Visibility and Progress Tracking in Learning**

Learners on online platforms often feel disconnected from their progress, lacking a clear view of how much of a course they have completed and what remains, leading to decreased motivation and course abandonment.

**Proposed Solution Fit:**

* **Frontend (Client-Side) - Learner Dashboard:** Provides a centralized interface where learners can log in to view enrolled courses, module-wise completion status, and upcoming modules, ensuring transparency and clear progress tracking.
* **Backend (Server-Side) - Progress Tracking Logic:** Manages and updates user progress as they complete video lectures and assignments, ensuring data consistency across sessions.
* **High-Level Data Flow - Real-Time Updates:** RESTful APIs and Axios calls fetch the latest progress dynamically whenever a learner completes a module, updating dashboards seamlessly.
* **External Services/Integrations - Notifications:** Email reminders are triggered for pending modules, gently nudging learners to maintain consistency and reducing course abandonment.

**2. Problem Statement: Complicated Course Enrollment Processes**

Learners often encounter unclear, multi-step processes when enrolling in courses, leading to confusion and drop-offs during the enrollment phase.

**Proposed Solution Fit:**

* **Frontend (Client-Side) - Course Enrollment Interface:** Provides a clear, single-click enrollment feature with visual confirmation and post-enrollment instructions.
  + **UI/UX (Bootstrap/Material UI):** Ensures a clean, intuitive, and accessible interface for learners on mobile and desktop devices.
* **Backend (Server-Side) - Enrollment Management:** Processes learner enrollment securely while maintaining accurate user-course relationships in the database.
* **High-Level Data Flow:** Clear API endpoints handle the enrollment request, confirm it, and return a success response with course access details, ensuring a seamless learner experience.

**3. Problem Statement: Lack of Direct Communication Between Learners and Instructors**

Learners often have unresolved doubts or require clarification on course content but lack structured channels for direct, organized communication with instructors.

**Proposed Solution Fit:**

* **Frontend (Client-Side) - Discussion Forums:** Integrated discussion sections within each course where learners can post questions and interact with instructors and peers.
* **Backend (Server-Side) - Forum and Messaging Logic:** Manages posting, editing, and retrieving discussions linked to specific courses, ensuring organized doubt resolution.
* **Database:** Stores discussions and replies in MongoDB, associating them with relevant course and user records for persistent access.
* **APIs:** RESTful API endpoints manage the creation and retrieval of discussions efficiently, ensuring low-latency access and consistent structure.

**4. Problem Statement: Concerns About Payment Security for Premium Courses**

Learners hesitate to purchase premium courses due to concerns about payment security and potential misuse of payment information.

**Proposed Solution Fit:**

* **Frontend (Client-Side) - Secure Payment UI:** Integrates trusted payment gateways with clear payment flows, ensuring transparency during transactions.
* **Backend (Server-Side) - Payment Handling:** Does not store sensitive payment data; interacts securely with payment gateways via APIs to confirm transactions.
* **Security Measures:**
  + Uses HTTPS for encrypted data transfer during payment processes.
  + JWT-based user authentication ensures that only authorized users can initiate payments.
  + Regular security audits and validation checks protect against vulnerabilities.
* **Deployment Strategy (Render):** Leveraging Render provides built-in SSL certification for HTTPS and secure, scalable infrastructure, further enhancing payment and user data security.

**5. Problem Statement: Concerns Regarding Data Security and Privacy**

Learners share personal and learning progress data on the platform and are concerned about its privacy, potential data leaks, and unauthorized access.

**Proposed Solution Fit:**

* **Backend (Server-Side) - Security Enforcement:**
  + **User Authentication:** Secure login using hashed passwords and JWT tokens for session management.
  + **Authorization Controls:** Role-based access control ensures only authorized users access specific data.
  + **Input Validation:** Prevents malicious injections.
* **Key Non-Functional Considerations:**
  + Data encryption in transit (HTTPS) and at rest in MongoDB.
  + Secure API design and regular security audits.
* **Deployment Strategy:** Render’s infrastructure ensures SSL encryption, automated updates, and robust server security, reducing risks of breaches and maintaining data confidentiality.