Community Skill Exchange Platform

A web application where users can exchange skills. For example, a designer teaching Photoshop in exchange for coding lessons.

Functional Requirements:

- **User Authentication & Profiles:**

  - **User registration & login (email verification):** Allows new users to sign up and existing users to access their accounts securely, with email confirmation required to activate the account.

  - **Role-based access (Learner/Teacher/Admin):** Assigns distinct permissions and features based on the user's role, ensuring that only Teachers can create listings and only Admins can manage the platform.

  - **Profile customization (skills, bio, location):** Enables users to update their personal information, location, and list the skills they want to learn or teach, which aids in matching.

  - **Profile picture upload & edit:** Users can personalize their profile with a photo, making interactions more trustworthy and engaging.

  - **Password reset via email:** Provides a standard, secure mechanism for users who have forgotten their password to regain access to their account.

- **Skill Listings & Search:**

  - **Create new skill listings (with category & description):** Teachers can publish a detailed post describing the skill they are offering, classifying it, and providing necessary details for Learners.

  - **Filter/search skills by category, rating, or location:** Provides powerful tools for Learners to quickly narrow down the thousands of available skills

to find the most relevant, highly-rated, or nearby options.

- ○ **View detailed skill post pages:** Allows users to see all information related to a specific skill offering, including the full description, teacher profile, and any associated ratings.

- ○ **Favorite or bookmark skills:** Enables Learners to save listings they are interested in for later review or reference, acting as a personal watchlist.

- ○ **Pagination for large lists of skills:** Ensures that search results and skill lists load efficiently by breaking up large datasets into manageable, sequential pages.

- **Skill Exchange Management:**

  - ○ **Request a skill exchange (send/accept invitations):** Facilitates the core transaction: a Learner formally asks a Teacher for a lesson, and the Teacher can then approve or decline the request.

  - ○ **Messaging system between users:** Provides a private, secure channel for users to communicate details, ask clarifying questions, and coordinate before and during an exchange.

  - ○ **Schedule management (calendar integration):** Allows users to propose, agree upon, and manage the date and time for their skill sessions, ideally linking to external calendar services.

  - ○ **Feedback and rating system after session:** Gathers user satisfaction data by allowing Learners and Teachers to rate and review each other post-exchange, building reputation and quality control.

  - ○ **Notifications for requests and messages:** Alerts users immediately to critical activity, such as new exchange requests, scheduling updates, or incoming messages.

- **Admin Dashboard:**

- **Manage users (activate/deactivate accounts):** Gives administrators the ability to enforce rules and maintain community standards by controlling user account status.

- **Approve or reject skill listings:** Allows administrators to moderate the quality and appropriateness of all skills being offered before they are visible to the general public.

- **Track overall platform analytics (users, exchanges, top skills):** Provides administrators with key data on user activity, exchange trends, and popular skills to guide platform strategy and development.

- **Send announcements or warnings to users:** Provides administrators with key data on user activity, exchange trends, and popular skills to guide platform strategy and development.

- **Backup and restore data:** Provides administrators with key data on user activity, exchange trends, and popular skills to guide platform strategy and development.

Non Functional Requirements:

1. **Usability & Performance:**
   - **Availability:** The platform shall maintain an uptime of at least 99.5% during scheduled operational hours to ensure users can request and manage exchanges reliably.
   - **Response Time:** All key actions such as user login, skill search/filter , loading a skill post must complete and display results within 3 seconds under normal load.
2. **Security:**
   - **Data Encryption:** All sensitive user data, including passwords, email verification status, and profile details, must be stored using industry-standard encryption protocols.
   - **Data Integrity:** The platform must implement a daily automated process to backup and restore all data from MySQL to prevent loss from system failure.

- **Unauthorized Access:** The role-based access control (Learner/Teacher/Admin) must be strictly enforced, ensuring that users can only execute functions intended for their role. Enforce email verification during registration and role-based access control (RBAC) for Learner, Teacher, and Admin roles.

3. **Scalability & Maintainability**
   - **Concurrent Users:** The system must be able to support a minimum of 5,000 concurrent active users without degradation in response time.
   - **Code Maintainability:** The codebase must be well-documented and follow established coding standards to allow new developers to implement new features like scheduling management or announcements with minimal effort.