## **Requirement Analysis: User Management and Authentication for ILE Platform**

****1. System Overview:****

This document outlines the requirements for user management and authentication functionalities within the integrated learning environment (ILE) platform. This system will support three user roles: admin, lecturer, and student.

****2. Functional Requirements:****

****2.1 User Roles and Responsibilities:****

* ****Admin:****
  + ****Login:**** Authenticate with existing credentials to access the platform.
  + ****User Management:**** Create, update, and delete user accounts (lecturer, student).
    - Reset user passwords.
    - Assign user roles (lecturer, student).
    - View all user information.
  + ****System Management:****
    - Configure system settings and integrations.
    - Manage user roles and permissions.
    - Monitor system performance and generate reports.
* ****Lecturer:****
  + Login with existing credentials.
  + View assigned courses and manage course materials.
  + Conduct live lectures via video conferencing.
  + Manage collaborative coding sessions for students.
  + Grade assignments and provide feedback.
* ****Student:****
  + Login with existing credentials.
  + Access enrolled courses and course materials.
  + Participate in live lectures and interact with lecturers.
  + Use the in-built IDE for coding practice and assignments.
  + Collaborate with peers in real-time coding sessions.
  + Submit assignments.
  + View grades and feedback.

****2.2 Authentication:****

* Implement a secure and reliable authentication mechanism.
* Support login for admins using existing credentials (no signup functionality).
* Utilize secure password hashing and storage techniques.
* Consider two-factor authentication (2FA) for additional security (optional).

****2.3 User Data Management:****

* ****Data Source:****
  + Import student and lecturer information from external sources (JSON, CSV, etc.) at system configuration or upon request.
  + Admins should have the ability to define the specific data format (JSON, CSV, etc.) for import.
  + Define a mechanism for handling potential data import errors (e.g., invalid format, duplicate entries).
* ****User Account Management:****
  + Admins only have the ability to create, update, and delete user accounts (lecturer, student).
  + Students and lecturers cannot create or manage their accounts directly.
  + System should store user information securely, including:
    - User ID (unique identifier)
    - Name
    - Email address
    - Role (admin, lecturer, student)
    - Additional relevant information (optional, e.g., department, student ID)
* ****Data Access Control:****
  + Implement role-based access control (RBAC).
  + Grant access to specific functionalities and data based on the user's role.
  + Admins have full access to all functionalities and data.
  + Lecturers and students only have access to functionalities and data specific to their roles.

1. ****Non-Functional Requirements:****

* ****Security:****
  + Ensure the system adheres to best practices for data security and user privacy.
  + Implement measures to prevent unauthorized access, data breaches, and other security vulnerabilities.
* ****Performance:****
  + User login and data retrieval should be efficient and responsive.
  + Consider scalability to accommodate future growth in user base and data volume.
* ****Usability:****
  + The user interface for user management should be intuitive and easy to use for administrators.
  + Provide clear error messages and guidance for data import and user management tasks.

1. ****Success Criteria:****

* The system successfully authenticates users based on their roles.
* User data is imported successfully from designated formats (JSON, CSV).
* Admins can effectively manage user accounts and access control.
* Users can access the platform and utilize functionalities based on their roles.
* Security measures are implemented to protect user data and system integrity.

****5. Open Questions:****

1. How will password reset functionality be implemented for admins?
2. What type of user information will be included in the imported data files?
3. What actions should be taken in case of data import errors?
4. How will user roles and permissions be visualized and managed within the system?