

# Sri Lanka Institute of Information Technology

## Programming Applications and Frameworks(IT3030)

Continuous Assignment – 2025, Semester 1

Initial Document

GROUP ID: DS-25



### **GROUP MEMBERS**

K.Gamsakini – IT22557674

A.Vithusana – IT22221896

# Project Description

## Overview:

LearnHub is a modern, interactive platform designed to facilitate skill-sharing and learning among users. Inspired by LinkedIn, LearnHub allows users to share their expertise, track their learning progress, and connect with others who have similar interests. The platform supports a wide range of skills, including coding, cooking, photography, and DIY crafts, making it a versatile space for both learners and experts.

The primary goal of LearnHub is to create an engaging, user-friendly platform that fosters a sense of community among skill-sharing enthusiasts. By providing tools for sharing skills, tracking learning progress, and interacting with others, LearnHub aims to support users in achieving their learning goals and maintaining a productive learning environment. The platform will empower users to share their knowledge, learn from others, and stay motivated through interactive features and community support.

The target audience for LearnHub includes skill-sharing enthusiasts of all levels (beginners, intermediate, advanced), professionals looking to share their expertise, learners seeking structured learning plans and resources, and anyone interested in acquiring new skills or improving existing ones.

The expected outcomes of the platform include a fully functional skill-sharing and learning environment, increased user engagement through interactive features like likes, comments, and notifications, and positive feedback on the platform's usability, features, and impact on users' learning journeys.

LearnHub is more than just a platform; it's a community-driven space where users can come together to share, learn, and grow. By leveraging modern technologies and a user-centric design, LearnHub will provide a seamless and enjoyable experience for users, helping them stay motivated and achieve their learning goals. The platform will empower users to share their knowledge, learn from others, and stay engaged through interactive features and community support.

## **Key Features:**

### **Skill Sharing Posts:**

- Users can upload up to three photos or short videos (maximum 30 seconds) per post.
- Each post includes a description to provide context and details about the shared skill.

### **Learning Progress Updates:**

- Users can post updates on their learning journey, such as completed tutorials or newly acquired skills.
- Predefined templates help users structure their progress updates, making it easy to share their achievements.

### **Learning Plan Sharing:**

- Users can create and share structured learning plans, including topics, resources, and timelines.
- Plans can be updated as users progress, allowing for dynamic and evolving learning paths.

### **Interactivity & Engagement:**

- Users can like and comment on posts to engage with others.
- Comments can be edited or deleted by the user who posted them.
- Post owners have the ability to delete comments on their own posts.

### **User Profiles & Social Features:**

- Each user has a profile page that displays their skill-sharing posts and activities.
- Users can follow other users to see their posts and updates.
- Profiles are publicly visible to encourage interaction and networking.

**Notifications:**

- Users receive real-time notifications for likes and comments on their posts.
- Notifications help users stay engaged and informed about their interactions.

**Authentication:**

- Users can log in using their existing social media accounts via OAuth 2.0, ensuring a seamless and secure login experience.

**Real-Time Chat App:**

- Users can engage in direct one-on-one or group conversations with others on the platform in real time.

**Technology Stack:**

- Backend: Java Spring Boot (REST API)
- Frontend: React.js
- Database: MySQL (Relational database for structured data storage)
- Authentication: OAuth 2.0 (for secure login using social media accounts)
- Real-Time Communication: WebSocket (for notifications and messaging)
- Version Control: Git with GitHub Workflow for continuous integration and deployment.

## **Development Process:**

### **Planning:**

- Define project scope, requirements, and milestones.
- Create wireframes and design mockups for the platform.

### **Design:**

- Develop a user-friendly interface with a focus on ease of use and engagement.
- Design the database schema and API endpoints.

### **Development:**

- Backend: Develop RESTful APIs using Java Spring Boot for user management, content sharing, and learning progress tracking.
- Frontend: Build responsive and interactive UI components using React.js.
- Integrate third-party services (OAuth 2.0 for authentication).

### **Testing:**

- Conduct unit testing, integration testing, and user acceptance testing (UAT) to ensure the platform is bug-free and performs well.
- Test real-time features like notifications and messaging.

### **Deployment:**

- Deploy the platform for scalability and reliability.

### **Maintenance:**

- Provide ongoing support, bug fixes, and feature updates based on user feedback.
- Monitor platform performance and security to ensure a smooth user experience.

# Functional Requirements

## 1. User Authentication and Authorization

### User Registration:

- Users can register using their email or social media accounts (OAuth 2.0).
- Users must provide a unique username, email, and password during registration.

### User Login:

- Users can log in using their registered email and password or via social media accounts (Google, Facebook, etc.).
- Implement JWT (JSON Web Tokens) for secure session management.

### User Roles:

- Two primary roles: Regular User and Admin.
- Regular users can create, share, and interact with content.
- Admins can manage users, posts, and platform settings.

### Password Recovery:

- Users can reset their password via a secure email link.

## 2. Skill Sharing Posts

### Create Posts:

- Users can create posts to share skills, including:
- Up to 3 photos or short videos (max 30 seconds per video).
- A description of the skill being shared (text input with a character limit of 500).

### Edit Posts:

- Users can edit their own posts (description, photos, or videos).

### Delete Posts:

- Users can delete their own posts.

**View Posts:**

- Users can view posts from other users in a feed.
- Posts should display the user's name, profile picture, description, and media.

**3. Learning Progress Updates****Create Progress Updates:**

- Users can post updates about their learning journey.
- Predefined templates are available for common updates (e.g., "Completed a tutorial on React.js").

**Edit Progress Updates:**

- Users can edit their progress updates.

**Delete Progress Updates:**

- Users can delete their progress updates.

**View Progress Updates:**

- Progress updates are displayed on the user's profile and in the feed.

**4. Learning Plan Sharing****Create Learning Plans:**

- Users can create structured learning plans, including:
  - Topics to be covered.
  - Resources (links, books, videos).
  - Completion timelines.

**Edit Learning Plans:**

- Users can update their learning plans as they progress.

**Delete Learning Plans:**

- Users can delete their learning plans.

**View Learning Plans:**

- Learning plans are visible on the user's profile and can be shared with others.

## **5. Interactivity & Engagement**

### **Like Posts:**

- Users can like posts shared by others.
- The number of likes is displayed on each post.

### **Comment on Posts:**

- Users can comment on posts.
- Comments can be edited or deleted by the user who posted them.
- Post owners can delete comments on their own posts.

### **Follow Users:**

- Users can follow other users to see their posts and updates in their feed.

### **Notifications:**

- Users receive real-time notifications for:
  - Likes on their posts.
  - Comments on their posts.
  - New followers.

## **6. User Profiles**

### **Profile Creation:**

- Each user has a profile page displaying:
  - Profile picture.
  - Bio (short description about the user).
  - Skill-sharing posts.
  - Learning progress updates.
  - Learning plans.

### **Edit Profile:**

- Users can edit their profile information (bio, profile picture, etc.).

### **Public Visibility:**

- Profiles are publicly visible to encourage interaction.



## **7. Search and Discovery**

### **Search Users:**

- Users can search for other users by username or skill.

### **Search Posts:**

- Users can search for posts by keywords or tags.

### **Discover Skills:**

- A discovery section suggests popular skills, posts, and users based on user interests.

## **8. Admin Features**

### **User Management:**

- Admins can view, edit, or delete user accounts.

### **Post Management:**

- Admins can view, edit, or delete any post on the platform.

### **Analytics:**

- Admins can view platform analytics (e.g., number of users, posts, interactions).

## **9. Integration with Third-Party Services**

### **OAuth 2.0 Integration:**

- Users can log in using their Google, Facebook, or other social media accounts.

## **10. Real-Time Chat App**

### **User-to-User Communication:**

- Users can send and receive real-time text messages to and from other users.

# Non-Functional Requirements

## 1) Performance

- The system should work quickly, even with many users online at the same time.
- Both the backend and frontend should load and respond within a few seconds for a smooth experience.
- The platform should handle a high number of requests without slowing down.

## 2) Scalability

- The system should be able to handle more users and content as LearnHub grows.
- The database should efficiently store large volumes of data, like posts, messages, and user profiles.

## 3) Security

- Users' personal information should be secure and only accessible to authorized individuals.
- Sensitive data (e.g.:- passwords) should be encrypted to prevent unauthorized access.
- Communication between the client and server should always use secure protocols (e.g.:- HTTPS).
- OAuth 2.0 authentication should be implemented for secure login via social media accounts.

## 4) Usability

- The website should be simple and easy to use, even for non-technical users.
- Navigation should be clear, with helpful instructions and easy-to-understand options.
- The platform should have an intuitive design to encourage skill sharing, progress tracking, and interaction.

## **5) Privacy**

- Users should be able to control who can see their personal information and posts.
- The platform should respect users' privacy, ensuring their data is kept secure and confidential.
- Users should have the option to make their profiles and posts private or public, depending on their preferences.

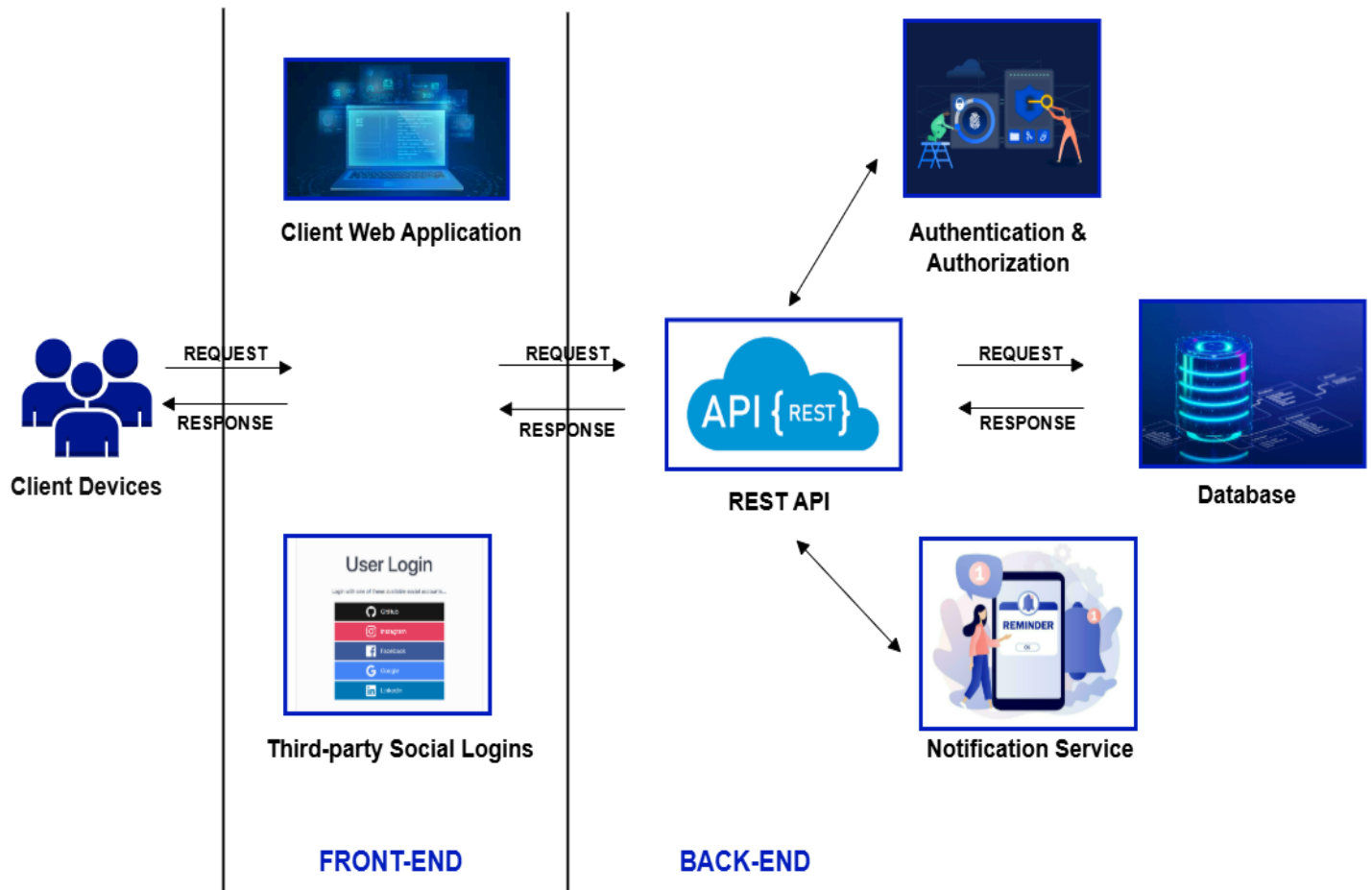
## **6) Reliability**

- The system should be highly available, with minimal downtime or interruptions.
- The platform should recover quickly from errors to ensure smooth operation.
- Regular backups should be done to avoid data loss in case of system failure.

## **7) Accessibility**

- The platform should be accessible to all users, including those with disabilities.

# System Overview Diagram



# **Initial Work Breakdown**

## **Project Planning and Documentation**

### **Tasks:**

- Requirement Gathering:  
Identify functional and non-functional requirements.

### **System Design:**

- Create system architecture diagrams (backend, frontend, database).
- Design Gantt Chart.

### **Documentation:**

- Prepare initial project documentation
- Finalize detailed documentation

### **Responsibility:**

#### **IT22557674 & IT22221896:**

- Both members will collaborate on all tasks, including requirement gathering, system design, Gantt chart and both initial and final documentation.

# **Work Distribution**

## **IT22557674:**

### **User Authentication and Authorization**

- Implement user registration (email and social media integration via OAuth 2.0).
- Implement user login with JWT for secure session management.
- Set up roles for Regular Users and Admin, and manage permissions.
- Implement password recovery functionality.

### **Skill Sharing Posts**

- Develop REST API endpoints for creating, editing, deleting, and viewing posts.
- Handle media uploads (photos and videos).

## **Interactivity & Engagement**

- Implement like, comment, and follow functionalities.
- Set up notifications for likes, comments, and new followers.

## **IT22221896:**

### **Learning Progress UI:**

- Create pages for posting, editing, and viewing learning progress updates.
- Add predefined templates for progress updates.

### **Learning Plan UI:**

- Develop pages for creating, updating, and viewing learning plans.
- Display structured data for topics, resources, and timelines.

### **Search and Discovery:**

- Implement user and post search functionalities.
- Create a discovery section that suggests popular skills, posts, and users.

## **IT22557674 & IT22221896:**

- **Database Design**
- **Admin Features**
  - User management (view, edit, delete user accounts).
  - Post management (view, edit, delete any post).
  - Set up platform analytics for admins (users, posts, interactions).
- **Real-Time Chat App**
  - Develop user-to-user communication via real-time text messaging.
- **Set up GitHub Workflow**
  - Collaborate on Git workflow and task management.
- **Testing & Deployment**

# Gantt Chart

