### **Project Specification Document**

#### 

#### **1. Introduction**

* **Project Name**: elearnia.io
* **Objective**: Develop a comprehensive online community management platform with advanced communication, personalization, and management tools.

#### **2. Features**

##### 

##### **Main Features**

* **Live Chat and Discussion Forums**:
  + **Description**: Allow community members to communicate in real-time via live chat, as well as engage in more in-depth discussions through forums.
  + **Specifications**: Message filtering, real-time notifications, ability to create discussion threads.
* **Feedback and Survey System**:
  + **Description**: Collect user feedback and gather opinions on various topics through surveys.
  + **Specifications**: Create custom surveys, real-time result analysis, survey history.
* **Profile Customization**:
  + **Description**: Allow users to personalize their profiles with additional information, themes, avatars, etc.
  + **Specifications**: Custom themes, visibility preferences management, badge addition.
* **Collaboration Tools**:
  + **Description**: Provide tools that enable community members to collaborate on common projects, such as shared documents or brainstorming sessions.
  + **Specifications**: Collaborative online documents, task boards, shared workspaces.
* **Community Analytics**:
  + **Description**: Provide detailed statistics on member activity, popular posts, engagement, etc.
  + **Specifications**: Customizable dashboards, activity reports, trend tracking.
* **Role and Permission Management**:
  + **Description**: Implement a detailed role management system with specific permissions for administrators, moderators, and members.
  + **Specifications**: Custom role creation, permission management, action history.

##### **Secondary Features**

* **Course Publishing Capability**:
  + **Description**: Allow the creation and management of online courses, with videos, quizzes, and progress tracking.
  + **Specifications**: Intuitive interface for course creators, student tracking, course completion certificates.
* **Social Media Integration**:
  + **Description**: Connect the platform with social networks for greater visibility and easy content sharing.
  + **Specifications**: Direct content sharing, profile connection, contact import.
* **Payment Integration**:
  + **Description**: Allow communities to manage paid subscriptions, donations, or the sale of premium courses and content.
  + **Specifications**: Secure transaction management, integration with payment gateways, subscription management.

#### 

#### **3. Technical Architecture**

* **Backend**:
  + Use Node.js with a framework like Express.js.
  + SQL/NoSQL database to store user information, courses, surveys, etc.
* **Frontend**:
  + React.js for a dynamic and responsive user interface.
* **Integrations**:
  + APIs for AI (if necessary), payment management, social networks, etc.

#### 

#### **4. Hosting and Infrastructure**

* **Cloud Server**: Use a cloud infrastructure like AWS or Azure.
* **Security**: Implement SSL certificates, firewall systems, and regular backups.

#### 

#### **5. Deployment Plan**

* **Development Phases**:
  + **Phase 1**: Development of main features (chat, forum, profile customization, role management, etc.).
  + **Phase 2**: Implementation of secondary features (courses, social media integration, payments).
* **Testing**:
  + Unit, integration, and beta testing.

#### 

#### **6. Maintenance and Support**

* **Maintenance**: Regular updates, bug fixes, continuous improvement.
* **Support**: User support with FAQ, assistance chat, and community forums.