## System Design and Architecture

## Introduction

IdeaNest is a comprehensive web-based platform designed to facilitate academic project management, collaboration, and mentorship. The system provides a complete ecosystem for students, mentors, faculty, and Hod sir to manage the entire project lifecycle from idea conception to final approval. This architecture document outlines the modular design approach, technology stack selection, and scalability planning that enables IdeaNest to handle multi-role authentication, project management workflows, and real-time collaboration features.

## Modular Design

### System Architecture Overview

IdeaNest follows a multi-tier modular architecture that separates concerns across distinct layers and components:

#### Core Modules

**Authentication & User Management Module**

* Multi-role authentication system supporting Studen/Faculty/Hod sir/Mentor roles
* Google OAuth integration with profile completion
* Traditional email/password authentication with secure sessions
* Password reset functionality with email verification
* User profile management with image upload and GitHub integration

**Project Management System Module**

* Project submission with multi-file upload validation (images, videos, code, presentations)
* Three-tier approval workflow: Student → Faculty→ Hod sir or Student → Hod sir
* Project categorization (Software/Hardware) with difficulty levels
* Enhanced project details including team size, development time, target audience
* File security with protected uploads and access control
* Real-time project status tracking and editing capabilities

**Ideas System Module**

* Idea sharing platform for project concepts
* Interactive engagement features (likes, comments)
* Content moderation with report system
* Real-time AJAX-powered interactions
* Idea deletion tracking with reason maintenance

**Mentor System Module**

* Comprehensive mentor dashboard with analytics
* Student-mentor pairing with request-based system
* Session management with meeting link integration
* Built-in email functionality with queue management
* Activity tracking and project access permissions

**Hodsir / Faculty Interface Module**

* Faculty features: project assignment, review queue, classification management
* Hod sir features: enhanced dashboard, user management, mentor oversight
* Data export capabilities in multiple formats
* Email configuration and notification dashboard
* Content moderation and system settings

### Module Interaction Architecture

The modular design enhances maintainability through:

* **Independent Component Development**: Each module can be developed and tested separately
* **Reusability**: Common components like authentication and file handling are shared across modules
* **Extensibility**: New features can be added without affecting existing modules

## Technology Stack

### Backend Technologies

**PHP 8.2+**

* Selected for its mature ecosystem and extensive web development capabilities
* Provides object-oriented programming features essential for modular architecture
* Native support for MySQL integration and session management
* Justification: Industry-standard for web applications with strong community support

**MySQL 10.4.28-MariaDB**

* Chosen for relational data management with optimized queries
* Foreign key constraints ensure data integrity across 30+ interconnected tables
* Support for complex queries required by multi-role authentication and project workflows
* Justification: Proven performance for academic management systems with concurrent users

**Apache 2.4 Web Server**

* Provides reliable HTTP server capabilities with mod\_rewrite support
* Essential for URL routing and security configurations
* Justification: Stable, widely-supported web server suitable for production deployment

**PHPMailer 6.10+**

* Implements reliable email delivery system with queue management
* Supports SMTP configuration for automated notifications
* Justification: Industry-standard email library with comprehensive error handling

### Frontend Technologies

**HTML5/CSS3 with Bootstrap Framework**

* Responsive design framework ensuring cross-device compatibility
* Modern web standards for semantic markup and accessibility
* Justification: Proven framework for academic platforms requiring multi-device access

**JavaScript (ES6+) with AJAX**

* Enables real-time interactive features without page refresh
* Supports dynamic content loading for project galleries and idea feeds
* Justification: Essential for modern web applications requiring responsive user experience

### Integration Technologies

**GitHub API v3**

* Provides repository and profile data synchronization
* Enables automatic profile completion for developer students
* Justification: Critical for ICT education platform connecting academic and professional development

**Google OAuth 2.0**

* Streamlines user registration and authentication process
* Reduces barrier to entry for academic users
* Justification: Standard authentication method for educational platforms

### Development and Quality Assurance Tools

**Composer Dependency Management**

* Autoloading and package management for PHP dependencies
* Enables modular development with third-party integrations
* Justification: Industry standard for PHP project organization

**PHPUnit Testing Framework**

* Comprehensive test suite with unit, integration, and functional testing
* Code quality assurance with PHP\_CodeSniffer (PSR-12 compliance)
* PHPStan static analysis for error prevention
* Justification: Essential for maintaining code quality in educational software

## Scalability Planning

### Horizontal Scaling Approach

**Database Optimization**

* Strategic indexing on frequently queried columns (user\_id, project\_id, status)
* Prepared statements throughout the application preventing SQL injection and improving performance
* Connection pooling for efficient database resource utilization
* Potential for read replicas to handle increased query load

**Email System Scalability**

* Queue-based email processing with priority management
* Cron job automation for background email delivery
* Rate limiting to prevent server overload

### Vertical Scaling Considerations

**Server Resource Optimization**

* PHP memory limit configuration for handling large file uploads
* Apache worker process optimization for concurrent user sessions
* MySQL buffer pool sizing for improved query performance

### Problem Identification and Solutions

**Database Performance**

* Identified risk: Complex join queries across project approval workflow
* Solution: Query optimization and selective indexing on approval status fields
* Monitoring: Database query performance logging for continuous optimization

**File Upload**

* Current limit: 10MB per file with PHP processing
* Solution: Progressive enhancement with chunked uploads for larger files
* Future: Direct browser-to-cloud upload bypassing server processing

**Concurrent User**

* Risk: Session management overhead during peak usage (project deadlines)
* Solution: Session storage optimization and connection pooling
* Monitoring: Real-time server resource monitoring with automated scaling triggers

### Reliability and Performance Measures

**Error Handling and Recovery**

* Comprehensive error logging across all system modules
* Database transaction management ensuring data consistency
* Automated backup procedures for data protection
* Graceful degradation for non-critical features during high load

**Performance Monitoring**

* Email delivery tracking and analytics
* User activity logging for usage pattern analysis
* System health monitoring for proactive issue identification

## Conclusion

IdeaNest's system architecture demonstrates a well-planned approach to academic project management platform development. The modular design ensures maintainability and extensibility, while the carefully selected technology stack provides a solid foundation for current requirements and future growth. The scalability planning addresses identified bottlenecks with concrete solutions and migration paths.

The system's emphasis on security, performance optimization, and user experience positions it as a comprehensive solution for educational institutions seeking to modernize their project management and mentorship processes. The modular architecture enables incremental improvements and feature additions without disrupting core functionality, supporting long-term platform evolution and institutional adoption.

**System Design**   
click on preview

Mentor flow: <https://github.com/Vivekchavda1374/IdeaNest/blob/main/System%20Design%20/mentor_system.mmd>

Database flow :   
<https://github.com/Vivekchavda1374/IdeaNest/blob/main/System%20Design%20/database_schema.mmd>

Sequence flow:   
<https://github.com/Vivekchavda1374/IdeaNest/blob/main/System%20Design%20/sequence_diagram.mmd>

Whole System Flow:

<https://github.com/Vivekchavda1374/IdeaNest/blob/main/System%20Design%20/system_design.mmd>

Work flow for project :

<https://github.com/Vivekchavda1374/IdeaNest/blob/main/System%20Design%20/workflow_diagram.mmd>

github project report : <https://github.com/Vivekchavda1374/IdeaNest/blob/main/Report/ARCHITECTURE_OVERVIEW.md>