WIREFRAME DOCUMENT - EduManage

EduManage Wireframe Architecture

EduManage can be visualized as a three-layer stack: Presentation Layer, Business Logic Layer, and Data Access Layer. Here's a high-level wireframe architecture with tables and flowcharts:

Presentation Layer

- This layer interacts with the user and displays information.
- Technologies: HTML, CSS, Javascript (ReactJS, AngularJS, VueJS)
- Components:
    - Login Screen
    - Dashboard
        - User Management (if applicable)
        - Course Management
            - Add Course
            - Edit Course
            - View Course Details
            - Enroll Students (if applicable)
        - Student Management (if applicable)
            - View Student List
            - Add Student (if applicable)
            - Edit Student Details (if applicable)
        - Assignment Management (if applicable)
            - Create Assignment
            - View Assignment List
            - Submit Assignment (Student)
            - Grade Assignment (Teacher)
    - Reports (Optional)

Flowchart - User Login

A[User Enters Credentials] --> B{Validate credentials}
B --> C{Login Successful} {Credentials Invalid}
C --> D[Display Dashboard]
B --> E[Display Error Message]

Business Logic Layer

- This layer handles the core application logic and business rules.
- Technologies: Python (Django/Flask), Java (Spring), NodeJS (Express)
- Components:
    - User Management (if applicable)
        - User login/logout
        - User creation (if applicable)

    - Course Management
        - Add Course
        - Edit Course
        - Delete Course
        - Enroll Students (if applicable)
    - Student Management (if applicable)
        - Add Student (if applicable)
        - Edit Student Details (if applicable)
    - Assignment Management (if applicable)
        - Create Assignment
        - Edit Assignment
        - Delete Assignment
        - Grade Assignment (Teacher)
    - Data Validation
    - Business Rules Engine

Flowchart - Add Course

A[User Requests Add Course] --> B{Validate course data}
B --> C{Data Valid} {Data Invalid}
C --> D[Save course data to database]
D --> E[Course Added Successfully]
B --> F[Display Error Message]
E|D --> G[Display Success Message and potentially redirect to course list]

Data Access Layer

- This layer interacts with the database and handles data persistence.
- Technologies: MySQL, PostgreSQL, MongoDB
- Database Schema:
    - Users (if applicable) - user_id, username, password, role (optional)
    - Courses - course_id, course_name, description, teacher_id (foreign key)
    - Students (if applicable) - student_id, name, email, course_id (foreign key) (optional)
    - Assignments (if applicable) - assignment_id, assignment_name, due_date, course_id (foreign key)
    - Submissions (if applicable) - submission_id, student_id (foreign key), assignment_id (foreign key), content, grade (optional)
- (There can be additional tables based on specific functionalities)

Flowchart - Save Course

A[Business Logic Layer calls Save Course] --> B{Connect to database}
B --> C{Save course data to Courses table}
C --> D[Save successful] {Save failed}
D --> E[Disconnect from database]
E --> F[Return success message to Business Logic Layer]
B --> G[Handle Database Error]
G --> E

Note: This is a high-level overview, and the specific implementation details will vary depending on your chosen technologies and functionalities. You can extend this wireframe to include additional features and functionalities specific to EduManage.