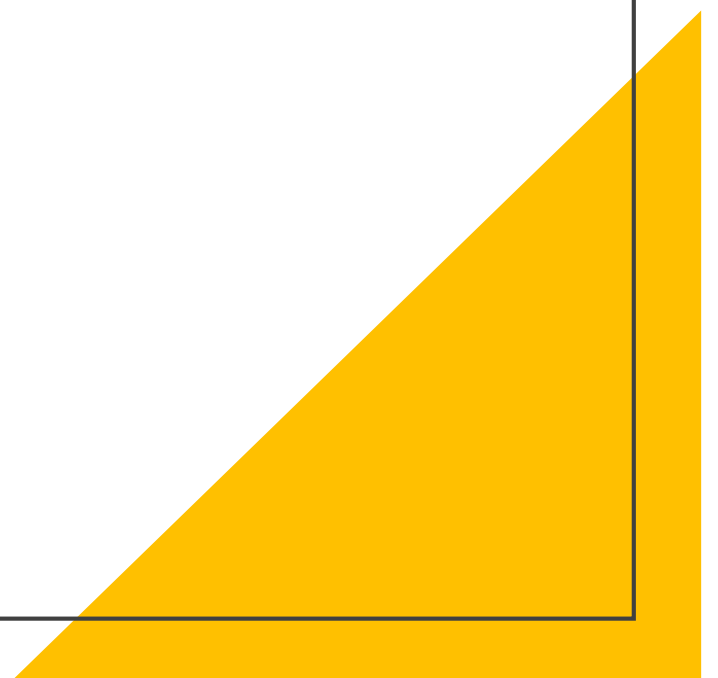


# School Management System

Mahmoud Gamal

MERN Stack Developer



# Description:

Our graduation project focuses on developing a school management system that streamlines administrative tasks, facilitates and enhances the learning experience of students. We will be using a full-stack approach, combining Node.js for the backend and React for the frontend, along with Redux for state management and PostgreSQL for the database



# Key Features:

01

A user-friendly dashboard for administrators to manage student records, schedules, attendance, and grades

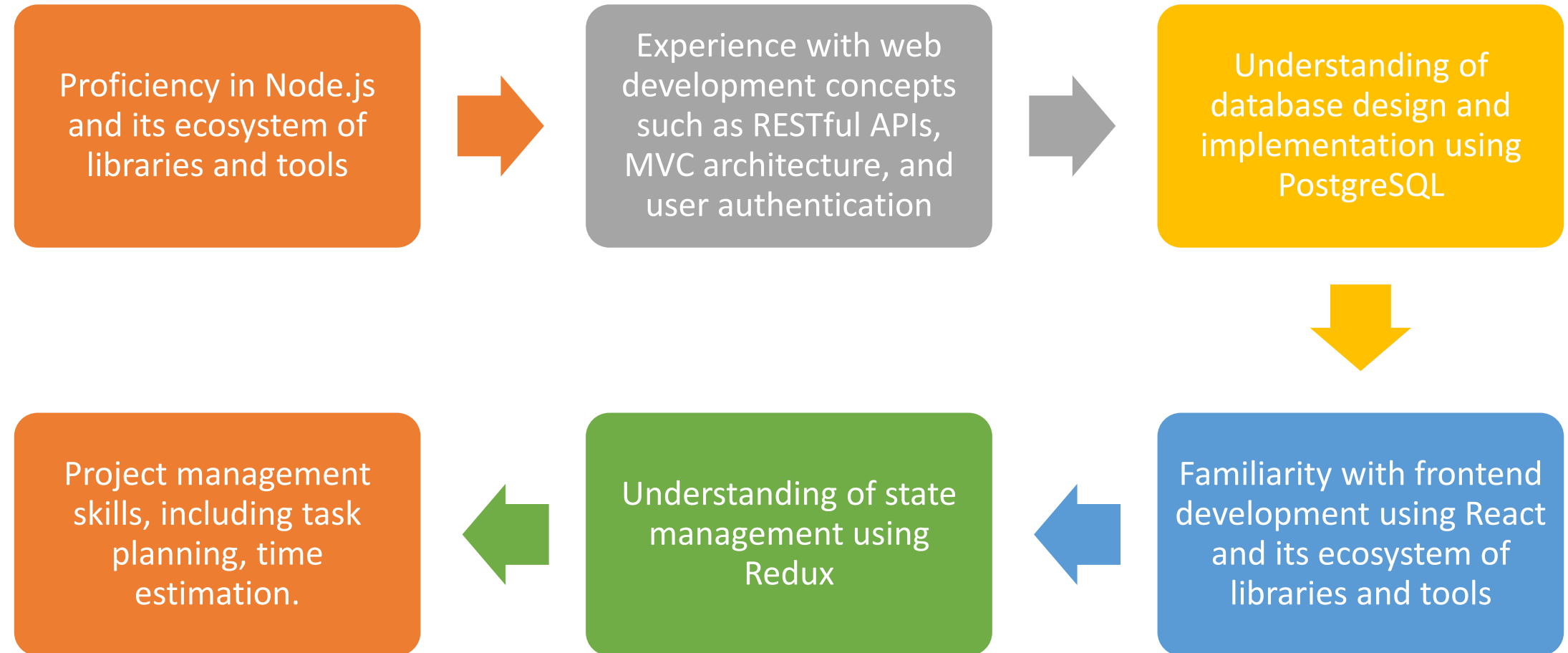
02

Ability to manage student information, including their personal details, enrollment, attendance, and academic performance.

03

Ability to manage courses, including their details, schedules, and enrollment.  
User authentication and authorization system to ensure secure access to the system.

# Skills Developed:



# Workflow:

1. Planning and analysis: Define project requirements, identify stakeholders, and create a project plan.
2. Backend development: Build the backend using Node.js and PostgreSQL, including data models, APIs, and authentication.
3. Frontend development: Build the frontend using React and Redux, including user interface design, state management, and API integration.
4. Testing: Conduct unit tests, integration tests, and end-to-end tests to ensure functionality and quality.
5. Deployment: Deploy the application to a server or cloud service, configure the environment, and ensure scalability and security.
6. Maintenance: Monitor the application, fix bugs, and implement new features based on feedback and user needs.

# Project Plan

**Week 1-2:** Project planning and analysis, including defining project requirements, identifying stakeholders, and creating a project plan.

**Week 3-5:** Backend development, including building the backend using Node.js and PostgreSQL, creating data models, APIs, and authentication.

**Week 6-8:** Frontend development, including building the frontend using React and Redux, designing the user interface, and integrating with the backend.

**Week 9-10:** Testing, including conducting unit tests, integration tests, and end-to-end tests to ensure functionality and quality.

**Week 11-12:** Deployment, including deploying the application to a server or cloud service, configuring the environment, and ensuring scalability and security.

**Week 13-14:** Maintenance, including monitoring the application, fixing bugs, and implementing new features based on feedback and user needs.