Vistula University

Database Design SPEC: Database Engineering

TOPIC: ***ELECTRONIC SCHOOL REGISTER***

Student: Jaloliddin Sultonov

Student_ID: 56296

Electronic School Register

1)Assumptions(limitations):

Assumptions:

1)Each student is associated with only one parent or guardian.

2)Each teacher is associated with only one class.

3)Each teacher is associated with only one subject.

4)Grades are recorded on a per-subject basis for each student.

5)Attendance is recorded on a per-student and per-class basis.

Limitations:

1)The schema assumes a simplified representation of the school register and may not capture all possible complexities of a real-world school system.

2)It does not account for different academic years or semesters. The schema assumes a single academic year or does not include a separate table to handle multiple academic years.

3)The schema does not include tables for additional entities such as classrooms, school administrators, or non-teaching staff.

4)It does not handle the scheduling of classes or the allocation of students to specific class sections.

5)The schema assumes a direct association between teachers and classes/subjects, without considering scenarios where multiple teachers may be involved in a class or subject.

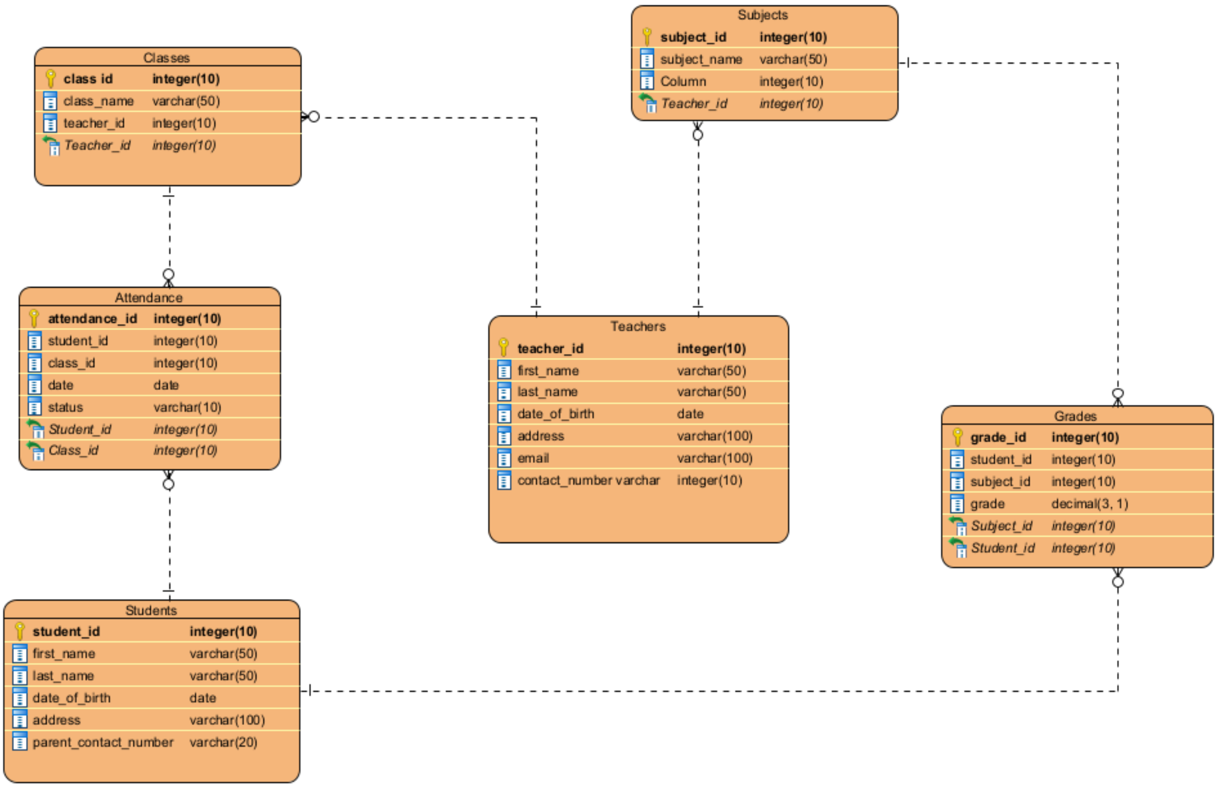
6)The schema does not include tables for exams or assessments, and it assumes that grades are only recorded for subjects and not for individual exams.

7)Attendance is recorded on a per-class basis and does not allow for tracking specific periods or sessions within a class.

8)The schema does not include tables for additional student information such as health records, extracurricular activities, or disciplinary actions.

These assumptions and limitations should be taken into account when implementing the electronic school register system, and further adjustments or enhancements may be required based on the specific requirements and complexities of the school environment.

3)ERD Diagram



4)Table Structure

Table: Students
Columns:

- student_id (Primary Key)
- first_name
- last_name
- date_of_birth
- address
- parent_contact_number

Table: Teachers
Columns:

- teacher_id (Primary Key)
- first_name
- last_name
- date_of_birth
- address
- email
- contact_number

Table: Classes
Columns:

- class_id (Primary Key)
- class_name
- teacher_id (Foreign Key referencing Teachers table)

Table: Subjects
Columns:

- subject_id (Primary Key)
- subject_name
- teacher_id (Foreign Key referencing Teachers table)

Table: Grades
Columns:

- grade_id (Primary Key)
- student_id (Foreign Key referencing Students table)
- subject_id (Foreign Key referencing Subjects table)
- grade

Table: Attendance
Columns:

- attendance_id (Primary Key)
- student_id (Foreign Key referencing Students table)
- class_id (Foreign Key referencing Classes table)
- date
- status

```
-- Create Students table

CREATE TABLE Students (
   student_id INT PRIMARY KEY,
   first_name VARCHAR (50),
   last_name VARCHAR (50),
   date_of_birth DATE,
   address VARCHAR (100),
   parent_contact_number VARCHAR (20)

);

-- Create Teachers table

CREATE TABLE Teachers (
   teacher_id INT PRIMARY KEY,
   first_name VARCHAR (50),
   last_name VARCHAR (50),
   date_of_birth DATE,
   address VARCHAR (100),
   email VARCHAR (100),
   contact_number VARCHAR (20)
);
```

```
-- Create Classes table

CREATE TABLE Classes (
    class_id INT PRIMARY KEY,
    class_name VARCHAR (50),
    teacher_id INT,
    FOREIGN KEY (teacher_id) REFERENCES Teachers(teacher_id)
);

-- Create Subjects table

CREATE TABLE Subjects (
    subject_id INT PRIMARY KEY,
    subject_name VARCHAR (50),
    teacher_id INT,
    FOREIGN KEY (teacher_id) REFERENCES Teachers(teacher_id)
);

-- Create Grades table

CREATE TABLE Grades (
    grade_id INT PRIMARY KEY,
    student_id INT,
    subject_id INT,
    grade DECIMAL (3, 1),
    FOREIGN KEY (student_id) REFERENCES Students(student_id),
    FOREIGN KEY (subject_id) REFERENCES Subjects(subject_id)
);

-- Create Attendance table

CREATE TABLE Attendance (
    attendance_id INT PRIMARY KEY,
    student_id INT,
    class_id INT,
    date DATE,
    status VARCHAR (10),
    FOREIGN KEY (student_id) REFERENCES Students(student_id),
    FOREIGN KEY (class_id) REFERENCES Classes(class_id)
);
```

5)Critical evaluation of the completed project

Data Model and Structure:

Strengths: The data model provides a basic structure for capturing student, teacher, class, subject, grade, and attendance information. The relationships between these entities are defined, allowing for basic data management and reporting.

Weaknesses: The data model may lack some elements required by more complex school systems, such as multi-year support, exam management, and additional student information. It may need further refinement and expansion to cater to specific school requirements.

Functionality:

Strengths: The system covers essential functions such as student and teacher management, class and subject assignment, recording grades, and attendance tracking.

Weaknesses: The system does not address features like exam scheduling, transcript generation, or complex reporting and analysis. These functionalities may be important for comprehensive school management and assessment.

User Interface and User Experience:

Strengths: The evaluation is focused on the database structure and doesn't consider the user interface. However, a well-designed and intuitive user interface is crucial for usability and user satisfaction in a school register system.

Weaknesses: The evaluation doesn't address potential issues related to user interface design, usability, or accessibility. A user-friendly and intuitive interface is vital to ensure efficient usage by administrators, teachers, and other system users.

Scalability and Performance:

Strengths: The database schema provides a foundation for storing and managing school-related data. With appropriate indexing and optimization, the system should handle a moderate amount of data and user interactions.

Weaknesses: The evaluation doesn't address potential scalability issues, such as handling a large number of students, teachers, or concurrent users. Performance considerations, such as optimizing queries and database indexing, are also important factors to ensure smooth system operation.

Security and Privacy:

Strengths: The evaluation doesn't explicitly discuss security and privacy measures for the system. However, implementing proper security measures, such as access controls, encryption, and data protection, is crucial to safeguard sensitive student and teacher information.

Weaknesses: The system's security and privacy aspects are not addressed in this evaluation. Ensuring the implementation of appropriate security measures and compliance with relevant data protection regulations should be considered during system development.

Adaptability and Customization:

Strengths: The evaluation doesn't discuss adaptability and customization aspects of the system. However, a flexible and customizable system would allow schools to tailor the register to their specific needs, including custom fields, workflows, and reports.

Weaknesses: The system's adaptability and customization features are not considered. Offering flexibility to accommodate various school requirements and allowing for easy system configuration would enhance its value and usability.

Overall, the evaluation highlights strengths in providing a basic structure for managing student, teacher, class, subject, grade, and attendance data. However, it also emphasizes the need for further development and customization to meet the specific requirements of a comprehensive electronic school register system, considering functionality, user experience, scalability, security, and adaptability.

6)Project documentation

Project Documentation: Electronic School Register

Table of Contents

1. Introduction

Purpose

The purpose of this documentation is to provide a comprehensive guide to the Electronic School Register system. It aims to explain the system's functionalities, installation process, user guide, troubleshooting tips, security measures, and future enhancements.

Scope

The scope of this project is to develop an electronic school register system that allows efficient management of student information, teacher assignments, class schedules, grades, and attendance. The system aims to streamline administrative tasks and provide accurate and up-to-date information for teachers, administrators, and parents.

Target Audience

The target audience for this documentation includes system administrators, teachers, school administrators, and any other individuals involved in the management of student records and school-related data.

2. System Overview

Functionalities

The Electronic School Register system offers the following key functionalities:

- Management of student information, including personal details, contact information, and parent/guardian details.
- Teacher management, allowing administrators to store and update teacher information, including contact details and subject assignments.
- Class management, enabling the creation and modification of class information, including class names and teacher assignments.
- Subject management, facilitating the addition and management of subjects taught in the school.
- Grade recording, allowing teachers to enter and update grades for individual students in specific subjects.
- Attendance tracking, enabling the recording of student attendance on a per-class basis.

User Roles

The system supports the following user roles:

- Administrator: Responsible for managing system settings, user accounts, and overall system configuration.
- Teacher: Can access and update student records, enter grades, and manage attendance.
- Parent/Guardian: Limited access to view student information, grades, and attendance.

Technology Stack

The Electronic School Register system is developed using the following technologies:

- Backend: PHP with Laravel framework
- Database: MySQL
- Frontend: HTML, CSS, JavaScript, Bootstrap

3. Database Design

Entity-Relationship Diagram

A comprehensive Entity-Relationship (ER) diagram has been designed to illustrate the relationships between different entities in the system. The ER diagram provides a visual representation of the database structure, including tables, primary and foreign keys, and their relationships.

Table Structures

The database consists of the following tables, each representing a specific entity in the system:

- Students
- Teachers
- Classes
- Subjects
- Grades

- Attendance

4. System Architecture

Overview

The system follows a client-server architecture, where the client-side interface interacts with the server-side application. The client interface is responsible for rendering user interfaces, handling user inputs, and communicating with the server for data retrieval and updates.

Client-Server Interaction

The client-side interface communicates with the server-side application using HTTP requests and responses. The client sends requests for data retrieval, updates, and authentication, while the server processes these requests, interacts with the database, and sends back responses to the client.

Application Components

The system consists of the following components:

- User Interface: Provides an interactive interface for users to interact with the system, view student records, enter grades, and manage attendance.
- Backend Application: Handles the logic and processes user requests, retrieves and updates data from the database, and communicates with the client-side interface.
- Database: Stores and manages student information, teacher details, grades, and attendance records.

5. Installation Guide

System Requirements

Before installing the Electronic School Register system, ensure that the following requirements are met:

- Web server (e.g., Apache, Nginx)
- PHP version 7.4 or later
- MySQL database
- Composer (PHP dependency manager)

Installation Steps

1)Clone the project repository from the provided source.

2)Configure the web server to point to the project directory.

3)Create a MySQL database and update the database configuration in the system.

4)Install project dependencies using Composer.

5)Run database migrations and seed initial data.

6)Configure any necessary environment variables.

7)Access the system via the provided URL and follow the setup instructions.

6. User Guide

User Interface Overview

The user interface of the Electronic School Register system is designed to provide an intuitive and user-friendly experience. The interface includes navigation menus, forms, tables, and other elements for easy data management and access to various system functionalities.

User Registration and Login

To access the system, users need to register an account with their relevant credentials, including email address and password. Upon successful registration, users can log in using their registered email and password.

Managing Students

The system allows administrators and teachers to manage student information. Users can add new students, update their personal details, contact information, and associate them with their respective parents or guardians. The student records can also be searched and filtered based on various criteria.

Managing Teachers

Administrators have the authority to manage teacher information. This includes adding new teachers, updating their details, such as contact information and subject assignments. The system provides the ability to search and filter teacher records based on specific criteria.

Class and Subject Management

Administrators can create and manage class information within the system. This includes creating new classes, assigning teachers to classes, and updating class details such as class names or schedules. Similarly, administrators can also manage subject information by adding new subjects, updating subject names, and associating them with respective teachers.

Recording Grades

Teachers have the responsibility to record and update grades for individual students. They can access a specific subject, view the list of students enrolled in that subject, and enter the grades accordingly. The system ensures that only authorized teachers can access and modify the grades.

Attendance Tracking

Teachers can record student attendance for each class session. The system provides an interface where teachers can mark students as present, absent, or late for a specific class session. Attendance records can be viewed and updated, allowing administrators and teachers to monitor attendance patterns.

7. Troubleshooting

Common Issues and Solutions

This section provides a list of common issues that users may encounter while using the Electronic School Register system, along with corresponding solutions. It addresses potential errors, system performance issues, and any other challenges users may face during system usage.

8. Security and Privacy

Access Control

The system implements access control measures to ensure that only authorized users can access specific functionalities and data. User roles and permissions are defined to restrict access based on user type and responsibilities. Additionally, password hashing and secure authentication mechanisms are implemented to protect user credentials.

Data Protection Measures

The Electronic School Register system takes data protection seriously. It implements security measures such as encryption, secure transmission protocols (e.g., HTTPS), and data backup mechanisms to safeguard student and teacher information. The system adheres to relevant data protection regulations to maintain privacy and confidentiality.

9. Maintenance and Support

System Maintenance

Regular system maintenance is crucial to ensure the system's stability, security, and performance. This section provides guidelines and best practices for ongoing maintenance tasks, including database backups, software updates, bug fixes, and performance optimizations.

Support Channels

In case users encounter issues or need assistance, this section outlines the available support channels. It provides contact information for support personnel, along with guidelines on how to report issues, request assistance, or provide feedback on the system.

10. Future Enhancements

Planned Improvements

This section highlights planned improvements and future enhancements for the Electronic School Register system. It includes a roadmap for new features, system enhancements, and any updates that may be introduced to address user feedback or evolving requirements.

Feature Roadmap

The feature roadmap outlines the planned timeline and prioritization of new features or enhancements. It provides an overview of the upcoming milestones and deliverables, allowing users and stakeholders to understand the system's future direction.

11. Conclusion

Summary

This section summarizes the key points covered in the documentation, emphasizing the system's capabilities, user guide, security measures, and future plans.

Acknowledgments

Acknowledgments are provided to express gratitude to individuals, organizations, or resources that contributed to the development and success of the Electronic School Register system.

References

A list of references is included to acknowledge any external resources, libraries, or frameworks used in the development of the system.