University Student's Schedule Management Systemst

CDS-302 Summer 2024

Alizeh Murtaza

June 19, 2024

Abstract

The following report will delve into a database created for university students to anage their schedules to the maximum efficiency and convenience. The report will descrive the structure of the database, queries that can be used in order to retrive information from the database, and the applications of the database on real life students, counselors, etc.

1 Introduction

University students have many responsibilities, which can include classes, jobs, as well as extracurriculars. In order to make their lives easier and manage their time better, this project will create a database that can keep track of a student's responsibilities to allow for a smooth personal schedule daily. The intended users for this database would be university students as well as advisors/counselors who can be given access at the student's discretion to provide advice or support.

2 Database Design

2.1 E-R Diagram

The Entity Relationship(ER) Diagram will be able to visualize the structure of the database and break down the different entities and attributes that the database consists of. The entities that the ER Diagram consists of are the following: Students, Classes, Work Schedules, Exams, Assignments, Extracurriculars, and Personal Dates.

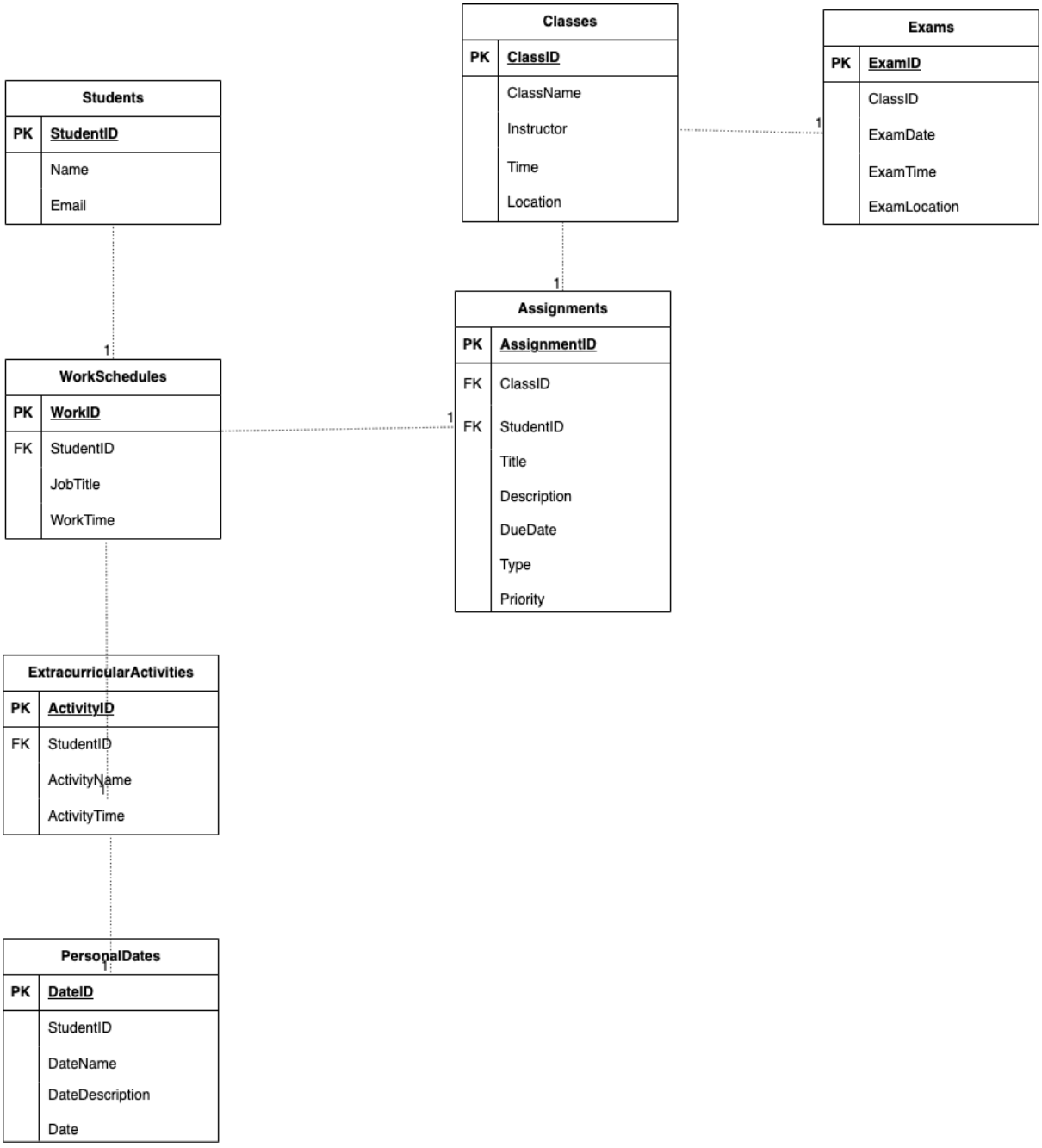

Figure 1: Entity-Relationship Diagram (ERD)

2.2 Relational Database Schema

The Relational Database Schema uses the information from the ER Diagram and essentially turns the entities into tables. The tables that the Relational Database Schema consists of are: Students, Classes, Work Schedules, Exams, Assignments, Extracurriculars, and Personal Dates.

2.2.1 Students

```
CREATE TABLE Students (
    StudentID INT PRIMARY KEY,
    Name VARCHAR(100),
    Email VARCHAR(100)
);
```

2.2.2 Classes

```
CREATE TABLE Classes (
    ClassID INT PRIMARY KEY,
    ClassName VARCHAR(100),
    Instructor VARCHAR(100),
    Time VARCHAR(50),
    Location VARCHAR(100)
);
```

2.2.3 WorkSchedules

```
CREATE TABLE WorkSchedules (
    WorkID INT PRIMARY KEY,
    StudentID INT,
    JobTitle VARCHAR(100),
    WorkTime VARCHAR(50),
    FOREIGN KEY (StudentID) REFERENCES Students(StudentID)
);
```

2.2.4 Exams

```
CREATE TABLE Exams (
    ExamID INT PRIMARY KEY,
    ClassID INT,
    ExamDate DATE,
    ExamTime TIME,
    ExamLocation VARCHAR(100),
    FOREIGN KEY (ClassID) REFERENCES Classes(ClassID)
);
```

2.2.5 Assignments

```
CREATE TABLE Assignments (
    AssignmentID INT PRIMARY KEY,
    ClassID INT,
    StudentID INT,
    Title VARCHAR(100),
    Description TEXT,
    DueDate DATE,
    Type VARCHAR(50),
    Priority VARCHAR(50),
    FOREIGN KEY (ClassID) REFERENCES Classes(ClassID),
    FOREIGN KEY (StudentID) REFERENCES Students(StudentID)
);
```

2.2.6 ExtracurricularActivities

```
CREATE TABLE ExtracurricularActivities (
    ActivityID INT PRIMARY KEY,
    StudentID INT,
    ActivityName VARCHAR(100),
    ActivityTime VARCHAR(50),
    FOREIGN KEY (StudentID) REFERENCES Students(StudentID)
);
```

2.2.7 PersonalDates

```
CREATE TABLE PersonalDates (
    DateID INT PRIMARY KEY,
```

```
    StudentID INT,
    DateName VARCHAR(100),
    DateDescription TEXT,
    Date DATE,
    FOREIGN KEY (StudentID) REFERENCES Students(StudentID)
);
```

3 Data Insertion

The next step for creating the database was inserting fake data that could mimic a typical university student's schedule into it to view how it can be used.

3.1 Sample Data for Students

```
INSERT INTO Students (StudentID, Name, Email) VALUES
(1, 'Alizeh Murtaza', 'alizeh@gmail.com'),
(2, 'Makda Asgedom', 'makda@gmail.com'),
(3, 'Arsema Techane', 'arsema@gmail.com');
```

3.2 Sample Data for Classes

```
INSERT INTO Classes (ClassID, ClassName, Instructor, Time, Location) VALUES
(101, 'CDS', 'Dr. Bob Johnson', '09:00-10:30', 'Room 111'),
(102, 'Econ', 'Dr. Talia Smith', '11:00-12:30', 'Room 222'),
(103, 'Math', 'Dr. Robert Kent', '3:00-5:30', 'Room 333');
```

3.3 Sample Data for WorkSchedules

```
INSERT INTO WorkSchedules (WorkID, StudentID, JobTitle, WorkTime) VALUES
(1, 1, 'Front DEsk Manager', '10:00-12:00'),
(2, 2, 'Teacher Assistant', '7:00-8:00'),
(3, 3, 'Tutor', '8:30-9:00');
```

3.4 Sample Data for Exams

```
INSERT INTO Exams (ExamID, ClassID, ExamDate, ExamTime, ExamLocation) VALUES
(1, 111, '2024-06-20', '09:00:00', 'Room 101'),
(2, 222, '2024-06-21', '11:00:00', 'Room 102'),
(3, 333, '2024-06-22', '03:00:00', 'Room 103');
```

3.5 Sample Data for Assignments

```
INSERT INTO Assignments (AssignmentID, ClassID, StudentID, Title, Description, DueDate, Type, Priori
  (2, 222, 2, 'Project 1', 'Econ', '2024-06-16', 'Project', 'Medium'),
  (3, 333, 3, 'Quiz 1', 'Math', '2024-06-17', 'Quiz', 'High');
```

3.6 Sample Data for ExtracurricularActivities

```
INSERT INTO ExtracurricularActivities (ActivityID, StudentID, ActivityName, ActivityTime) VALUES
(1, 1, 'Basketball', '8:00-9:30'),
(2, 2, 'Music Club', '7:00-7:30'),
(3, 3, 'Drama Club', '3:00-4:30');
```

3.7 Sample Data for PersonalDates

```
INSERT INTO PersonalDates (DateID, StudentID, DateName, DateDescription, Date) VALUES
(1, 1, 'Birthday', 'Alizeh\'s Birthday', '2024-06-25'),
(2, 2, 'Anniversary', 'Makda\'s Anniversary', '2024-07-01'),
(3, 3, 'Doctor Appointment', 'Arsema\'s Doctor Appointment', '2024-06-30');
```

4 Data Analysis: Query Examples

4.0.1 Retrieve All Students

```
SELECT * FROM Students;
```

The query results in a list of each student in the studentable's student ID, name, and email.

4.1 Retrieve All Classes

```
SELECT * FROM Classes;
```

This query results in each class in the Classes table's class ID, name, instructor, time, and location.

4.2 Retrieve Exam Details for All Classes

```
SELECT ClassID, ExamDate, ExamTime, ExamLocation FROM Exams;
```

This query results in details for every exam in all classes including the class ID, exam date, exam time, and exam location.

4.3 Retrieve Work Schedule for a Specific Student

```
SELECT WorkSchedules.JobTitle, WorkSchedules.WorkTime
FROM WorkSchedules
JOIN Students ON WorkSchedules.StudentID = Students.StudentID
WHERE Students.Name = 'Alizeh Murtaza';
```

This query results in the job title and work time for the student named "Alizeh Murtaza," joining the workschedules and students table.

4.4 Retrieve Extracurricular Activities for a Specific Student

```
SELECT ActivityName, ActivityTime
FROM ExtracurricularActivities
WHERE StudentID = 3;
```

This query results in a list of the names and times of all extracurricular activities for the student with the student ID being "3".

4.5 Count Number of Students per Class Based on Assignments

```
SELECT Classes.ClassName, COUNT(Assignments.StudentID) AS NumberOfStudents
FROM Classes
JOIN Assignments ON Classes.ClassID = Assignments.ClassID
GROUP BY Classes.ClassName;
```

This query results in the amount of students that are assignmend to each class through joining the classes and assignments tables.

4.6 Retrieve Assignments Due on a Specific Date

```
SELECT Students.Name, Assignments.Title, Assignments.DueDate
FROM Students
JOIN Assignments ON Students.StudentID = Assignments.StudentID
WHERE Assignments.DueDate = '2024-06-15';
```

This query results in a list of student names as well as their assignment title and due date for assignments that have a due date of June 15, 2024.

4.7 Retrieve Personal Dates for a Specific Student

```
SELECT DateName, DateDescription, Date
FROM PersonalDates
WHERE StudentID = 2;
```

This query results in a list of the name, date, and description of the personal dates the student with the student ID number of "2".

4.8 Retrieve Job Titles for Students with a Specific Job

```
SELECT Students.Name, WorkSchedules.JobTitle
FROM Students
JOIN WorkSchedules ON Students.StudentID = WorkSchedules.StudentID
WHERE WorkSchedules.JobTitle = 'Teacher Assistant';
```

This query results in the names of the students and their job title when their job title is "Teacher Assistant".

4.9 Retrieve Assignments for a Specific Class

```
SELECT Title, Description, DueDate
FROM Assignments
WHERE ClassID = 111;
```

This query results in a list of the assignment title, description, and due date of all assignments due for the class with the class ID of 111.

5 Applications

This database has much more to offer than a simple calendar. With queries that can retrieve, sort, list, etc... the data in multiple ways, there are more uses than just an arrangement of dates. There is also selective access, meaning the users of the database have a certain amount of access to it based on their role. For example, a counselor accurate depiction of the student's workload and life. that does not have a student ID will not be able to access a student's information unless that access is given to them by the student. The student may choose how much or little access they want to give to someone else. This will benefit the student and counselor. The student will be able to allow their counselor to create a plan, advice, etc... based on the schedule and the counselor will be able to have a more accurate depiction of the student's workload and life in general.

This database will not only allow for external people to get a visualization of a student's workload, but also the student themself. Through filling the database with their information and using queries to show the amount of work they must need to get done, the student can also have a better picture of if they can balance their work and life, something crucial for a student's well-being. Keeping track of tasks, exams, assignments, and more will allow the student to be in a better state of mind and make their lives easier or more organized thorugh making sure that personal plans, and work/study does not clash.

This database is also able to be accustomed to each and every student. Based on classes, work, extracurriculars, and personal matters- all of which are entities in the database-, the student can fit their unique needs into the database.

This database has many purposes. Firstly, reducing stress through allowing for a more organized scheduling system. Secondly, succeeding academically through time management and keeping track of important due dates. Thirdly, balancing their work and life through having a visual representation of all of their plans in a single database. Finally, having their support (counselors, therapists, loved ones, etc...) be able to gain insight on to the student's life to better guide and support the student through their academic and peronal journey.

6 Conclusion

The University Student's Schedule Management System database is a simple database, however, has the ability to have a powerful and positive impact on student's lives. It combines the different aspects of a

student's life and provides a platform for time management that aims to increase academic success, time management, reduced stress, and improved support.