Database Documentation

Written By Marcus, Checked by Ras

January 24, 2024

Contents

1 Introduction

This document outlines the database structure for a web-based application designed for easier PRISM and ITP assigning. The primary goal of this system is to streamline the process of assigning students to either PRISM projects or internships, thereby eliminating the need for manual allocation by the teachers.

The database plays a crucial role in this automation by storing and managing essential data, including student profiles, project and internship opportunities, and the assignment records.

The database is structured to support the following key functionalities:

- Storing comprehensive student information such as academic strengths, interests, and skill sets.
- Maintaining detailed records of PRISM projects and ITP opportunities, including the type of opportunity, etc.
- Facilitating the assignment of students to these opportunities, ensuring an optimal match based on their profiles and the specific requirements of each opportunity.

2 Database Schema Overview

The database for the student assignment system comprises key tables designed to store and manage data efficiently:

- **Users Table:** Manages user credentials and roles, storing user IDs, passwords, roles, and contact information.
- **Students Table:** Contains detailed student profiles, including academic and personal information.
- **Teachers Table:** Holds data about faculty members, their IDs, and specializations.
- **Opportunities Table:** Lists PRISM and internship opportunities with details like type, duration, specializations, and teacher in charge.
- **ITP and PRISM Tables:** Provide specifics about each opportunity, including company and project details.
- **Assigned Table:** Records student assignments to opportunities. A composite key of OpportunityID and StudentID is suggested.
- **Tags and TagKey Tables:** Facilitate categorization of students and opportunities for efficient matching.

3 Table Schemas

This section presents the structure of the database through detailed schemas of each table, including column names, data types, key indicators, and descriptions.

3.1 Users Table

Column Name	Data Type	Key	Description
UserID	int	PK	Unique identifier for each user
Password	nvarchar(256)		Hashed password for security
Role	int		User role (0 for teacher, 1 for student)
Email	nvarchar(256)		User's email address
DateRegistered	datetime		Date and time of user registration
FullName	nvarchar(256)		Full name of the user
Deleted	int		Indicates if the user is active (0) or deleted (1)

3.2 Students Table

Column Name	Data Type	Key	Description
StudentID	char(7)	PK	Unique identifier for each student
UserID	int	FK	Identifier for the user, linked to Users table
ResumeLink	nvarchar(256)		Link to the student's online resume
Specialisation	nvarchar(3)		Student's field of study
Citizenship	nvarchar(64)		Citizenship status
GPA	float		Grade Point Average
ProjRank	nvarchar(max)		Self-assessed project ranking

3.3 Teachers Table

Column Name	Data Type	Key	Description
UserID	int	PK	Unique identifier for each teacher
Specialization	nvarchar(256)		Specialization of the teacher

3.4 Opportunities Table

Column Name	Data Type	Key	Description
OpportunityID	int	PK	Unique identifier for each opportunity
StartDate	datetime		Start date of the opportunity
EndDate	datetime		End date of the opportunity
Deleted	int		Indicates if the opportunity is active (0) or deleted (1)
Slots	int		Number of available slots
Description	nvarchar(256)		Brief description of the opportunity
Specialisation	varchar(3)		Required specialization for the opportunity
TeacherID	int	FK	Teacher overseeing the opportunity, linked to Teachers table
Company	nvarchar(256)		Company offering the opportunity
CitizenType	nvarchar(256)		Citizenship requirement for the opportunity

3.5 ITP Table

Column Name	Data Type	Key	Description
OpportunityID	int	PK, FK	Identifier for the ITP, linked to Opportunities table
JobRole	nvarchar(128)		Role or position offered in the internship

3.6 PRISM Table

Column Name	Data Type	Key	Description
OpportunityID	int	PK, FK	Identifier for the PRISM project, linked to Opportunities
Type	nvarchar(256)		Type or category of the PRISM project
Title	nvarchar(128)		Title of the PRISM project

3.7 Assigned Table

Column Name	Data Type	Key	Description
OpportunityID	int	FK	Identifier of the assigned opportunity
StudentID	char(7)	PK, FK	Identifier of the student assigned to the opportunity
Comments	nvarchar(256)		Additional comments or notes regarding the assignment

3.8 Tags Table

Column Name	Data Type	Key	Description
TagID	int	PK	Unique identifier for each tag
TagName	nvarchar(256)		Name or label of the tag

3.9 TagKey Table

Column Name	Data Type	Key	Description
TagID	int	FK	Identifier of the tag, linked to Tags table
StudentID	char(7)	FK	Identifier of the student associated with the tag
OpportunityID	int	FK	Identifier of the opportunity associated with the tag

4 Version History

This section chronicles the evolution of the database schema, highlighting significant changes and refinements made over time to better meet the system's requirements.

4.1 Initial Setup

Creation of Core Tables:

- *Users*: Holds user credentials and roles.
- *Students*: Contains student-specific information.
- *Teachers*: Stores teacher details.
- *Opportunities*: General information about opportunities.
- *ITP and PRISM*: Separate tables for different types of opportunities.
- *Assigned*: Links students to opportunities.
- *Tags and TagKey*: For tagging/categorizing students and opportunities.

4.2 Refinement and Streamlining

Alteration in Opportunity Management:

- Change of *OpportunityID* type from *int* to *nvarchar(5)* in ITP, PRISM, and Opportunities tables.
- *TeacherInCharge* field moved from ITP to Opportunities table.
- Adaptation of the *Assigned* table to accommodate the new *OpportunityID* format.

4.3 Enhancing Student Profiles

Expanding Students Table:

- Addition of *Descr*, *Interests*, *Frameworks*, and *ProjRank* fields to Students table.

Adjustment in Tagging System:

- Modification of the structure of the *TagKey* table.

4.4 Centralizing Opportunity Information

Consolidation of Opportunity-Related Data:

- Inclusion of *TeacherInCharge* in the Opportunities table.
- Simplification of ITP and PRISM tables, focusing on specific opportunity details.
- Simplification of the structure of the *Assigned* table.

4.5 Emphasizing Demographics and Eligibility

Inclusion of Demographic Data:

- *Citizenship* added to the Students table.
- *CitizenType* added to the Opportunities table.

4.6 Final Revised Database

Major Changes in Database Structure:

- *Users Table*: Redefined to include additional fields like FullName and Deleted.
- *Teachers Table*: Introduced to handle teacher-specific data.
- *Students Table*: StudentID now a primary key; relationship with Users table established.
- *Opportunities Table*: Redesigned to include a more comprehensive set of fields.
- *ITP and PRISM Tables*: Simplified and made more specific to their respective programs.

- *Tags and TagKey Tables*: Unchanged.
- *Assigned Table*: Redefined to handle associations between students and opportunities.

5 Stored Procedures Overview

This section provides an overview of some of the stored procedures in the database, detailing their functionality.

5.1 UpdateStudent

Purpose: Updates a student's information in the Students table.

- Accepts parameters for student ID, name, resume link, GPA, interests, frameworks, and project rank.
- Updates the respective student's record based on the provided student ID.

5.2 UpdatePRISM

Purpose: Updates details of a PRISM opportunity in both the Opportunities and PRISM tables.

- Takes parameters for opportunity ID, title, type, description, supervisor, slots, specialization, start and end dates, company, and citizenship type.
- Updates the Opportunities table for general opportunity details and the PRISM table for specific PRISM-related information.

5.3 RemoveOpportunity

Purpose: Marks an opportunity as deleted in the Opportunities table.

- Accepts an opportunity ID as a parameter.
- Sets the 'Deleted' status to 1 (deleted) for the specified opportunity. Here are some (but not all) of the examples that we have written.

5.4 EnsureTagExists

Purpose: Ensures a tag exists in the Tags table, creating it if it doesn't.

- Accepts a tag name as input.
- Checks if the tag exists; if not, it inserts the new tag and returns its ID.

5.5 AddTeacher

Purpose: Adds a new teacher to the Teachers table.

- Takes parameters for teacher ID, name, and specialization.
- Inserts a new record into the Teachers table with the provided details.

6 Conclusion

This document has presented a comprehensive overview of the database designed for the automated assignment of students to PRISM projects and internships. It details the structure and evolution of the database, emphasizing efficiency and user-specific needs.