

# Database Documentation

## 0. Key

Primary Key

Foreign Key

## 1. Tables

### Schools

*Info: The relation of schools.*

*Columns: schoolId, name, country, state, city*

*Detail:*

**schoolId** (Type: unsigned int auto\_increment)

- The primary key of the this table. Uniquely identifying each school (primarily college, university, or graduate schools) in the database.

**name** (Type: varchar(128))

- The full name of the school.

**country** (Type: varchar(64))

- The name of the country in which the school is located.

**state** (Type: varchar(64))

- The state (or division within a country) in which the school is located.

**city** (Type: varchar(64))

- The city (or applicable settlement hierarchy) in which the school is located.

### Colleges

*Info: The relation of colleges within a school. This table only applies to Schools who have colleges within them.*

*Columns: collegId, name, school*

*Detail:*

**collegId** (Type: unsigned int auto\_increment)

- The primary key of this table. Uniquely identifying each college belonging to a specific school.

**name** (Type: varchar(128))

- The full name of the college belonging to a specific school.

**school** (Type: unsigned int) **REFERENCES Schools(schoolId)**

- Value representation of the school to which a college belongs.

## **Classes**

*Info: The relation of classes within a school.*

*Columns: classId, school, info, deleted*

*Detail:*

**classId** (Type: unsigned int auto\_increment)

- The primary key of this table. Uniquely identifying each class belonging to a specific school. A class may be cross-listed, see the ClassNames table for more information.

**school** (Type: unsigned int) **REFERENCES School(schoolId)**

- Value representation of the school to which a class belongs.

**info** (Type: varchar(1024))

- Information about the class. Each class must have one set of information even if it is cross-listed as another class who has a different description external to our application.

**deleted** (Type: boolean)

- If this column has a value of TRUE or 1 then the class has been marked as deleted. Otherwise, if this column has a value of FALSE or 0 the class is assumed to still be an available and active class.

## **Majors**

*Info: The relation of majors within a school.*

*Columns: majorId, school, name, abbreviation, college*

*Detail:*

**majorId** (Type: unsigned int auto\_increment)

- The primary key of this table. Uniquely identifying each major belonging to a specific school.

**school** (Type: unsigned int) **REFERENCES School(schoolId)**

- Value representation of the school to which a major belongs to.

**name** (Type: varchar(64))

- The full name of a major.

**abbreviation** (Type: varchar(8))

- The abbreviation or short name of a major.

**college** (Type: unsigned int) **REFERENCES Colleges(collegeId)**

- Value representation of the college to which a major belongs to.

## **ClassNames**

*Info: The relation of class names for a specific class. One class may have many names (specifically cross-listed classes).*

*Columns: class, abbreviation, major, name, unlisted*

*Detail:*

**class** (Type: unsigned int) **REFERENCES Classes(classId)**

- Value representation of the class to which a class name belongs to.

**abbreviation** (Type: varchar(16))

- Abbreviation for the class, usually comprised of the major/department abbreviation and a number.

**major** (Type: unsigned int) **REFERENCES Majors(majorId)**

- Value representation of the major to which this specific class name belongs to.  
Cross-listed classes may belong to different majors but all represent the same class.

**name** (Type: varchar(128))

- The full name of the class. Cross-listed classes may have different names for the same class.

**unlisted** (Type: boolean)

- If this column has a value of TRUE or 1 then a class is no longer listed as the class name in question. Otherwise, if this column has a value of FALSE or 0 then a class is listed as the class name in question.

## **Genders**

*Info: The relation of genders. Currently meeting the ISO/IEC 5218 standard of representing sexes.*

*Columns: genderId, description*

*Detail:*

**genderId** (Type: unsigned tinyint)

- The primary key of this table. Uniquely identifying a gender represented in the relation.

**description** (Type: varchar(32))

- Description of a gender.

## Users

*Info: The relation of users for the Halp applications.*

*Columns: `userId`, `firstName`, `lastName`, `username`, `email`, `phone`, `bio`, `password`, `school`, `major`, `year`, `verifiedTutor`, `joinDate`, `verifiedAccount`, `deleted`*

*Detail:*

**userId** (Type: unsigned int auto\_increment)

- The primary key of this table. Uniquely identifying each user.

**firstName** (Type: varchar(32))

- The first name of a user.

**lastName** (Type: varchar(32))

- The last name of a user.

**username** (Type: varchar(32))

- The username of a user.

**email** (Type: varchar(128))

- The email of a user. In order to verify a user this column must be populated.

**phone** (Type: varchar(16))

- The phone number of a user.

**dateOfBirth** (Type: DATE)

- The date of birth of a user.

**bio** (Type: varchar(128))

- The short description that a user writes about themselves.

**gender** (Type: unsigned tinyint) **REFERENCES Genders(genderId)**

- Value representation of the gender of a user.

**password** (Type: char(60))

- The password hash of a user. Hashed with bcrypt and the salt is stored in the hash.

**school** (Type: unsigned int) **REFERENCES Schools(schoolId)**

- Value representation of the school to which a user attends.

**major** (Type: unsigned int) **REFERENCES Majors(majorId)**

- Value representation of the primary major of which a user is studying or has studied.

**year** (Type: ENUM('1', '2', '3', '4', '5', 'Post-Graduate', 'Graduate'))

- The year count that a user is currently at in their studies. This only extends to a 5th year for those who have not completed their undergraduate studies in 4 years. After 5 years we no longer track the number of years and turn it over to Post-Graduate and Graduate. Any outliers here will be stuck with the options we give them unless there is a serious response for other options.

**verifiedTutor** (Type: boolean)

- If this column has a value of TRUE or 1 a user has been verified as a tutor. Otherwise, if this column has a value of FALSE or 0 a user has not been verified as a tutor. This

verification may vary by campus. If a campus has a tutoring center with student tutors, the student tutors would be applicable for a verified status.

**joinDate** (Type: datetime)

- The date and time a user joined Halp.

**verifiedAccount** (Type: boolean)

- If this column has a value of TRUE or 1 a user has verified that the email they provided is theirs. If this column has a value of FALSE or 0 a user has not verified that the email they provided is theirs and will not be allowed to proceed using the application.

**deleted** (Type: boolean)

- If this column has a value of TRUE or 1 a user has been deleted and will no longer be found in the application. If this column has a value of FALSE or 0 a user is assumed to still be active.

## **Tutors**

*Info: The relation of tutors for a class.*

*Columns: user, class, grade, year, unlisted*

*Detail:*

**user** (Type: unsigned int) **REFERENCES Users(userId)**

- Value representation of a user

**class** (Type: unsigned int) **REFERENCES Classes(classId)**

- Value representation of a class for which a user is a tutor.

**grade** (Type: ENUM('W', 'Fail', 'Pass', 'NC', 'CR', 'F', 'D-', 'D', 'D+', 'C-', 'C', 'C+', 'B-', 'B', 'B+', 'A-', 'A', 'A+'))

- The grade that a user received for the class that they are tutoring for.

**unlisted** (Type: boolean)

- If this column has a value of TRUE or 1 a user has unlisted the class in question from their available classes to tutor list. Otherwise, if this column has a value of FALSE or 0 a user is assumed to be willing to provide help for the class in question.

## **Posts**

*Info: The relation of posts that users have published.*

*Columns: postId, user, description, bounty, class, postDate, helped, helpedBy, deleted*

*Detail:*

**postId** (Type: unsigned int auto\_increment)

- The primary key of this table. Uniquely identifying each post.

**user** (Type: unsigned int) **REFERENCES Users(userId)**

- Value representation of a user of which a post belongs to.

**content** (Type: varchar(128))

- Content of the post.

**bounty** (Type: varchar(8))

- Very short description of what a user is willing to provide given that a tutor helps them. Ideally this will be a dollar value or some emoji.

**class** (Type: unsigned int) **REFERENCES Classes(classId)**

- Value representation of a class to which a post applies to.

**postDate** (Type: datetime)

- The date and time of the publication of the post.

**resolved** (Type: boolean)

- If this column has a value of TRUE or 1 a user has been successfully helped and the post is closed. Otherwise, if this column has a value of FALSE or 0 a user is assumed to be currently awaiting help.

**tutor** (Type: unsigned int) **REFERENCES Users(userId)**

- Value representation of a user that tutored or is currently tutoring the user of the post in question.

**deleted** (Type: boolean)

- If this column has a value of TRUE or 1 the post in question has been deleted and will not be displayed anywhere on the application. Otherwise, if this column has a value of FALSE or 0 the post in question will still be displayed and is either awaiting a response or has been resolved.

## Messages

*Info: The relation of messages.*

*Columns: messageId, content, postDate, user*

*Detail:*

**messageId** (Type: unsigned int auto\_increment)

- The primary key of the table. Uniquely identifying each message.

**content** (Type: varchar(256))

- The message content.

**postDate** (Type: datetime)

- The date and time that the message was published.

**user** (Type: unsigned int) **REFERENCES Users(userId)**

- Value representation of a user of which this message belongs.

## PostMessages

*Info: The relation of messages that belong to specific posts.*

*Columns: post, message*

*Detail:*

**post** (Type: unsigned int) **REFERENCES Posts(postId)**

- Value representation of a post.

**message** (Type: unsigned int) **REFERENCES Messages(messageId)**

- Value representation of a message that belongs to the post in question.

## **Tokens**

*Info: Login tokens for each user.*

*Columns: user, token, expire, revoked*

*Details:*

**user** (Type: unsigned int) **REFERENCES Users(userId)**

- Value representation of a user to which the token in question belongs to.

**token** (Type: char(64))

- Random 64 character value representing the login token of a user.

**expire** (Type: datetime)

- The date and time the token is set to expire.

**revoked** (Type: boolean)

- If this column has a value of TRUE or 1 the token in question is no longer acceptable. If this column has a value of FALSE or 0 the token in question is still available to use. Only one token per user may be active at a single point in time.