**PharmGenius Data Model**

Here you will find an explanation of the database in use for the PharmGenius website. The database was constructed with phpmyadmin by reverse engineering the data model that was in use for the previous version of the site that was stored on the Google App Engine.

Additional tables and fields were added as was deemed necessary while working on the migrating the site to PHP/MySQL.

A note about date columns: All created\_date columns in the tables will default to the current date/time when a row is inserted, thus negating the need to specifically insert the SYSDATE when inserting to the table. All last\_updated\_date columns are set to automatically update to the current date/time when a record is updated.

**Tables**

admin\_email\_addresses

Stores some data about email addresses that the application can use to send emails.

Note: Future versions may be well served to replace this table with a generic admin\_config table that just stores some key-value pairs for determining some admin settings.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| email\_type | VARCHAR | 20 | Primary key. Stores a string value representing the type of email to be sent. This is so the stakeholder can specify different email addresses for different types of emails. |
| email\_address | VARCHAR | 256 | The email address to send the email to for the given email\_type. |

answer

Stores answers to questions.

Note: This table has a backup table named deleted\_answer that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_answerBackup, which inserts data into the deleted\_answer table anytime data is removed from the answer table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| answer\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| question\_id | INT | 10 | Foreign key to question table. The question this answer belongs to. |
| answer\_text | VARCHAR | 256 | The answer text. |
| correct\_flag | ENUM | - | Indicates if the answer is the correct answer. Valid values are Y or N. |
| times\_chosen | INT | 10 | The number of times the answer has been chosen. |
| created\_date | DATETIME | - | The date the answer was created. |
| last\_updated\_date | DATETIME | - | The date the answer was last updated. |
| last\_updated\_by | VARCHAR | 20 | Foreign key to user table. Who last updated the answer. |

answer\_log

Stores a log of all the answers given by the users.

Note: This table has a backup table named deleted\_answer\_log that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_answerLogBackup, which inserts data into the deleted\_answer\_log table anytime data is removed from the answer\_log table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| answer\_log\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| question\_id | INT | 10 | Foreign key to question table. The question this answer was answering. |
| answer\_id | INT | 10 | Foreign key to answer table. The answer chosen. |
| username | VARCHAR | 20 | Foreign key to the user table. The user answering the question. |
| attempt\_number | INT | 10 | The number of times the user has attempted to answer the question. First time answering would have attempt\_number = 1. |
| correct\_flag | ENUM | - | Indicates if the answer is the correct answer. Valid values are Y or N. |
| score | INT | 10 | The score earned from this question. |
| created\_date | DATETIME | - | The date the answer was created. |

bug\_report

Stores records of bug report submissions from users.

Note: This table has a backup table named deleted\_bug\_report that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_bugReportBackup, which inserts data into the deleted\_bug\_report table anytime data is removed from the bug\_report table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| report\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| username | VARCHAR | 20 | Foreign key to the user table. The user reporting the bug. |
| report\_text | VARCHAR | 350 | The report text. |
| created\_date | DATETIME | - | The date the answer was created. |

category

Stores information about the categories for the quizzes.

Note: This table has a backup table named deleted\_category that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_categoryBackup, which inserts data into the deleted\_category table anytime data is removed from the category table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| category\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| category\_name | VARCHAR | 50 | Unique index. Can only be one category for any given name. The name of the category. |
| description | VARCHAR | 256 | A description of the category. |
| verified\_flag | ENUM | - | Indicates if the category has been verified. Valid values are Y or N. |
| active\_flag | ENUM | - | Indicates if the category is active. Valid values are Y or N. |
| created\_date | DATETIME | - | The date the category was created. |
| created\_by | VARCHAR | 20 | Foreign key to user table. Who created the category. |
| last\_updated\_date | DATETIME | - | The date the category was last updated. |
| last\_updated\_by | VARCHAR | 20 | Foreign key to user table. Who last updated the category. |

flagged\_questions

Stores questions that have been flagged by users for review.

Note: This table has a backup table named deleted\_flagged\_questions that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_flaggedQuestionsBackup, which inserts data into the deleted\_flagged\_questions table anytime data is removed from the flagged\_question table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| flagged\_question\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| question\_id | INT | 10 | Foreign key to question table. The question this answer was answering. |
| problem\_text | TEXT | - | The description of the problem entered by the user. |
| flagged\_by\_user | VARCHAR | 20 | Foreign key to the user table. The user flagging the question. |
| created\_date | DATETIME | - | The date the question was flagged. |

question

Stores information about the questions in the quizzes.

Note: This table has a backup table named deleted\_question that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_questionBackup, which inserts data into the deleted\_question table anytime data is removed from the question table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| question\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| quiz\_id | INT | 10 | Foreign key to quiz table. The quiz this question belongs to. |
| question\_text | VARCHAR | 500 | The text of the question. |
| score | INT | 10 | The base score a user earns for answering the question correctly. Defaults to 100. |
| times\_correctly\_answered | INT | 10 | The number of times the question has been answered correctly. |
| times\_answered | INT | 10 | The number of times this question has been answered. |
| explanation | VARCHAR | 500 | An explanation about the question and its answer. |
| verified\_flag | ENUM | - | Indicates if the category has been verified. Valid values are Y or N. |
| active\_flag | ENUM | - | Indicates if the category is active. Valid values are Y or N. |
| removed\_flag | ENUM | - | Indicates if the question has been removed. Valid values are Y or N. |
| removed\_date | DATETIME | - | The date and time the question was removed. |
| image\_url | VARCHAR | 256 | The url of the image associated with this question. |
| created\_date | DATETIME | - | The date the question was created. |
| created\_by | VARCHAR | 20 | Foreign key to user table. Who created the question. |
| last\_updated\_date | DATETIME | - | The date the question was last updated. |
| last\_updated\_by | VARCHAR | 20 | Foreign key to user table. Who last updated the question. |

question\_review\_history

Stores a history results of question reviews. (NOT CURRENTLY IMPLEMENTED)

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| review\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| question\_id | INT | 10 | Foreign key to question table. The question this review belongs to. |
| review\_outcome | VARCHAR | 500 | What the outcome of the review was. |
| review\_date | DATETIME | - | The date the question was reviewed. |
| reviewed\_by | VARCHAR | 20 | Foreign key to user table. The user that reviewed the question. |

quiz

Stores data about the quizzes.

Note: This table has a backup table named deleted\_quiz that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_quizBackup, which inserts data into the deleted\_quiz table anytime data is removed from the quiz table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| quiz\_id | INT | 10 | Primary key. Auto-increments for each new record. |
| category\_id | INT | 10 | Foreign key to the category table. The category that this quiz belongs to. |
| quiz\_name | VARCHAR | 50 | The name of the quiz |
| description | VARCHAR | 256 | A description of the quiz. |
| active\_flag | ENUM | - | Indicates if the quiz is active. Valid values are Y or N. |
| removed\_flag | ENUM | - | Indicates if the quiz has been removed. Valid values are Y or N. |
| removed\_date | DATETIME | - | The date and time the quiz was removed. |
| created\_date | DATETIME | - | The date the quiz was created. |
| created\_by | VARCHAR | 20 | Foreign key to user table. Who created the quiz. |
| last\_updated\_date | DATETIME | - | The date the quiz was last updated. |
| last\_updated\_by | VARCHAR | 20 | Foreign key to user table. Who last updated the quiz. |

user

Stores data about the users.

Note: This table has a backup table named deleted\_user that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_userBackup, which inserts data into the deleted\_user table anytime data is removed from the user table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| username | VARCHAR | 20 | Primary key. The username for this user |
| password | VARCHAR | 60 | The user’s encrypted password. |
| email\_address | VARCHAR | 256 | The user’s email address. |
| score | INT | 10 | User’s current score. |
| admin\_flag | ENUM | - | Indicates if the user is an admin. Valid values are Y or N. |
| owner\_flag | ENUM | - | Indicates if the user is the owner. Valid values are Y or N. |
| active\_flag | ENUM | - | Indicates if the user is active. Valid values are Y or N. |
| registered\_date | DATETIME | - | The date and time the user registered. |

user\_profile

Stores profile data about the user

Note: This table has a backup table named deleted\_user\_profile that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_userProfileBackup, which inserts data into the deleted\_user\_profile table anytime data is removed from the user\_profile table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| username | VARCHAR | 20 | Primary key. The username for this user |
| full\_name | VARCHAR | 50 | The user’s name. |
| year | VARCHAR | 10 | User’s year in school. |
| employer | VARCHAR | 50 | User’s employer |
| interests | VARCHAR | 256 | User’s interests |
| bio | VARCHAR | 500 | Bio about the user. |
| image\_url | VARCHAR | 256 | The url to the image for the user’s profile picture. |
| created\_date | DATETIME | - | The date the profile was created. |
| last\_updated\_date | DATETIME | - | The date the profile was last updated. |
| last\_updated\_by | VARCHAR | 20 | Foreign key to user table. Who last updated the profile. |

votes

Stores votes that users have given the questions. The dual primary key means that each user can only have one vote per question.

Note: This table has a backup table named deleted\_votes that contains all of the below fields and a deleted\_date field. This table is populated by the trigger trg\_voteBackup, which inserts data into the deleted\_votes table anytime data is removed from the votes table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Type** | **Size** | **Notes** |
| question\_id | INT | 10 | Primary key and foreign key to the question table. |
| username | VARCHAR | 20 | Primary key and foreign key to user table. The user voting on this question. |
| value | TINYINT | 4 | The value this user has given. Valid values are 1, 0, -1. |
| created\_date | DATETIME | - | The date the vote was first cast. |
| last\_updated\_date | DATETIME |  | The last time the vote was updated. |

**Views**

There are a few views in the schema. The views named vw\_correctanswers… facilitate the leaderboard functionality.

vw\_questionVoteValue

This view displays the current total value of votes cast for all the questions. It maps question\_id to value. This is used to determine the number to display next to the votes.

**Triggers**

There are a handful of triggers in the schema that automatically make some changes before or after inserts or deletes. If you are noticing unexpected behavior in the database, this should be the first place you look.

Any triggers with Backup in the name are designed to copy data to a backup table when the original data is deleted.

trg\_addAttemptNumber

This trigger executes before inserting new data into the answer\_log table. It determines the data that should be inserted into the attempt\_number field by counting the number of records that match the question\_id and username in the table, then adding 1 to that value.

trg\_addAttemptToQuestion

This trigger executes after data is inserted into the answer\_log table. This trigger updates the question table’s times\_answered and times\_answered\_correctly fields and increments them when appropriate. The times\_answered\_correctly field will only be incremented when the data just inserted to the answer\_log table had a correct\_flag value of ‘Y’.

trg\_updateQuestionRemovedDate

This trigger executes before updating the question table. If the incoming update has a removed\_flag value of ‘Y’, the update will set the removed\_date column to the current timestamp.

trg\_updateQuestionStatus

This trigger executes after updating the quiz table. If the quiz was set to inactive (active\_flag = ‘N’), this trigger will update active\_flag on all of that quiz’s questions to ‘N’ and vice versa.

trg\_updateQuizStatus

Executes after updating the category table. Much like the previous trigger, this trigger will update the quiz table and set every quiz belonging to the category to inactive if the category was set to inactive, and vice versa.

trg\_updateRemovedDate

Executes before updating the quiz table. If the quiz is about to be removed (removed\_flag = ‘Y’), the update will set the removed\_date column equal to the current timestamp.

**Stored Procedures**

There are a few stored procedures to facilitate certain database functions. Here is what they are and what they do.

sp\_createUserAccount

This stored procedure creates a user account by inserting data into the user table, and then into the user\_profile table. This procedure is used so that the PHP only has to make one call when creating a user, and the user table must be updated first because of the foreign key constraints.

sp\_resetUserScore

This stored procedure will reset a user’s score to zero. To do this, it first copies that user’s data from the answer\_log table to the deleted\_answer\_log table. Then it removes all of this user’s data from the answer\_log table. Finally, it updates the user table for that user and sets the score field to 0.

sp\_createCategoryAndQuiz

This procedure creates a category and quiz in the database. These creations are combined because a category should not be created without a quiz existing to fill it. In fact, the application’s UI should only allow a category to be created if a quiz name is also provided. This accepts the user creating the quiz, the category name and the quiz name. It first inserts data to the category table, then into the quiz table. It does so in this order due to foreign key constraints.

sp\_createQuestion

This procedure creates a question in the database. It accepts the following data: the user creating the question, the question text, the quiz ID of the quiz to which this question will belong, the correct answer text, the text of each of the four answers, and the explanation for the question. The procedure inserts the question, then each of the answers. This is done in this order due to foreign key constraints.