Jacob Preston

Dr. Charmayne Cullom

BACS 350 – 026

December 6, 2016

Documentation

# Data Dictionary

## Admin Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Length** | **Data Type** | **Why** |
| adminID | 11 | Integer | To have a unique id for each admin |
| First | 15 | Varchar | To store the first name of an admin |
| Last | 15 | Varchar | To store the last name of an admin |
| Username | 25 | Varchar | For the admin to login |
| Password | 25 | Varchar | For the admin to login |

## User Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Length** | **Data Type** | **Why** |
| userID | 11 | Integer | To have a unique id for each user |
| First | 15 | Varchar | To store the first name of the user |
| Last | 15 | Varchar | To store the last name of the user |
| Username | 25 | Varchar | For the user to login |
| Password | 25 | Varchar | For the user to login |

## Song Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Length** | **Data Type** | **Why** |
| songID | 11 | Integer | To have a unique id for each song |
| Title | 50 | Varchar | To store the title for each song |
| Artist | 50 | Varchar | To store the artist for each song |
| Lyrics | 1000 | Varchar | To store the lyrics for each song |

# File Names

I have the files organized into 4 sections: the root, config, partials, styles. The root is used for main pages that display something to the user except for process\_data.php. It is used to process any form data. Partials are chunks of code that are included in other files. The config folder just stores my utility file which has functions. Then, the styles folder contains the css file for styling the website.

## ./BACS\_350/

* account\_details.php contains form for editing account details
* admin\_home.php the home page for an admin
* create\_song.php contains form for creating a new song
* edit\_song.php contains form for editing song information
* error.php a page that loads when an error occurs
* index.php the landing page where the client selects if they are a user or admin
* login\_form.php contains form for logging in
* logout.php a logout confirmation page
* process\_data.php a page to process the post data after a form submits something
* register.php contains a form for registering a new user or admin
* user\_home.php the home page for the user
* view\_song.php contains a form to view details about the song

## ./BACS\_350/config/

* util.php a class with multiple functions used within the site

## ./BACS\_350/partials/

* footer.php the code for a footer that is included in most pages
* header.php the code for a header that is included in most pages
* menu\_bar.php the code for a menu bar that is included in most pages
* songs.php the code for the table of songs used on the home pages

## ./BACS\_350/styles/

* basic\_styles.css all the styling used on the website

# Variables

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Why** | **Where** |
| $\_POST[“admin\_view”] | Used to know if the post data came from the view song button on the admin\_home.php page | process\_data.php |
| $\_POST[“index\_admin”] | Used after the user clicks an option on the index.php page | process\_data.php |
| $\_POST[“index\_user”] | Used after the user clicks and option on the index.php page | process\_data.php |
| $\_POST[“login\_password”] | Used to store in the password variable | process\_data.php |
| $\_POST[“login\_username”] | Used to store in the username variable | process\_data.php |
| $\_POST[“login”] | Used to know if the post data came from the login\_form.php | process\_data.php |
| $\_POST[“register\_firstname”] | Used to store in the first variable | process\_data.php |
| $\_POST[“register\_lastname”] | Used to store in the last variable | process\_data.php |
| $\_POST[“register\_password”] | Used to store in the password variable | process\_data.php |
| $\_POST[“register\_username”] | Used to store in the username variable | process\_data.php |
| $\_POST[“register”] | Used to know if the post data came from register.php | process\_data.php |
| $\_POST[“song”] | Used to get the song id from the song.php page | process\_data.php |
| $\_POST[“user\_view”] | Used to know if the post data came from the view song button on the user\_home.php page | process\_data.php |
| $\_SESSION[“client\_type”] | To store in the session varable if the current user is an admin or a user | process\_data.php  menu\_bar.php |
| $\_SESSION[“db”] | To have access to the database | util.php |
| $\_SESSION[“error\_code”] | To store the error code in the active session in the browser to be used on the error.php page | error.php  process\_data.php  util.php |
| $\_SESSION[“id”] | To store the user id or admin id across pages | util.php |
| $\_SESSION[“status”] | Used to store if the user is logged in at the moment | util.php  menu\_bar.php |
| $\_SESSION[“temp”] | Used to store any variables from post data that need to be used temporarily | process\_data.php  view\_song.php |
| $artist | To store the artist of a song from the database or the input | util.php  view\_song.php  edit\_song.php |
| $conn | To store connection Boolean | util.php |
| $db | To store the instance of MySQLi | util.php |
| $e | To store the exception if the database connection fails | util.php |
| $error\_code | To store a string of text that will give the user a little more details about why they were redirected to the error.php page | error.php |
| $first | To store the first name of the user or admin from either the database or user input | account\_details.php  process\_data.php  util.php |
| $id | To store the id from the session variable id | util.php |
| $IDs | To get an array of song ids | songs.php |
| $last | To store the last name of the user or admin from either the database or user input | account\_details.php  process\_data.php  util.php |
| $lyrics | To store the lyrics of a song from the database or the input | util.php  view\_song.php  edit\_song.php |
| $password | To store the password of the user or admin from either the database or user input | account\_details.php  process\_data.php  util.php |
| $results | The Boolean of the SQL query | util.php |
| $rows | To receive the rows from the database query | util.php |
| $song\_artist | Used to store the artist of a song from the database or user input | view\_data.php |
| $song\_lyrics | Used to store the lyrics of a song from the database or user input | view\_data.php  edit\_song.php |
| $song\_title | Used to store the title of a song from the database or user input | view\_data.php  edit\_song.php |
| $songID | To store the unique id of the song from the database | view\_song.php  util.php  songs.php  edit\_song.php |
| $sql | To store the string of code that will be sent to the MySQL server | util.php |
| $title | To store the title of a song from the database or the input | util.php  view\_song.php  edit\_song.php |
| $username | To store the username of the user or admin either from the database or user input | process\_data.php  util.php |
| $util | To have access to the util class. This class contains many functions including SQL queries | Every .php page |
| $val | To store the song id in the IDs array | songs.php |

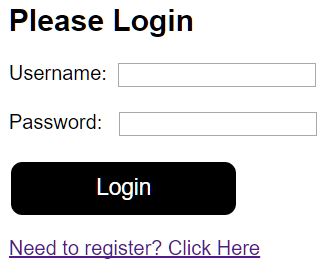
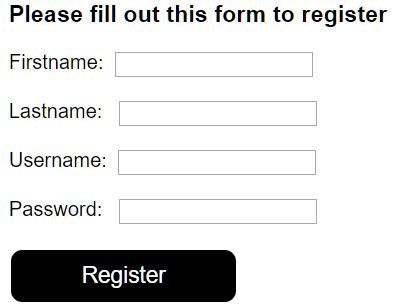
# Instructions

This website is used for getting lyrics for a song.

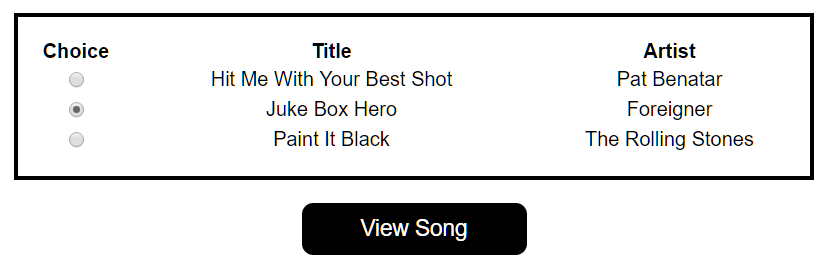
1. On the load page, click the button user if you are a user and admin if you are an admin.



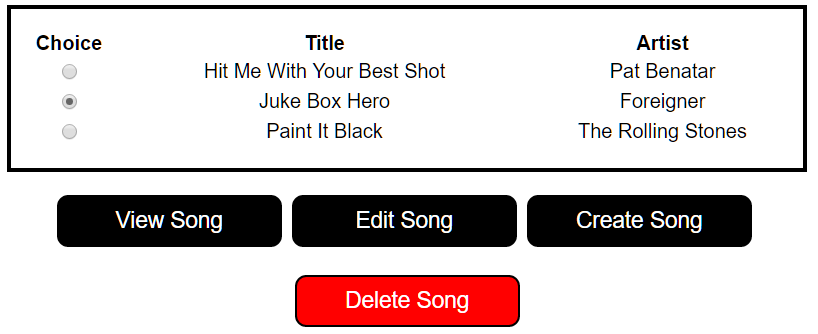
1. Then login with username and password or register as a new user

 or 

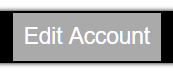
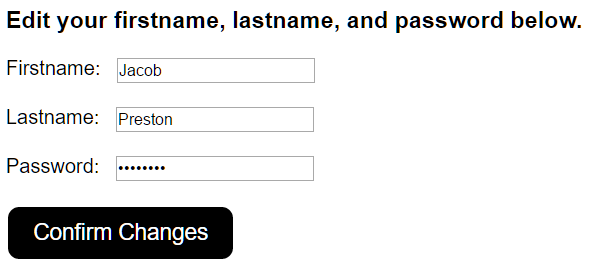
1. Once logged in, you can choose a song and click view song to see its lyrics



* 1. If you are an admin you can create songs, edit songs, or delete songs also



1. You can click edit account to change your first name, last name, or password

then 

1. Once finished you can click logout to logout

then 