**Design & Build Lab (DaBL) reservation system (PHP)**

11 December 2019

American University

|  |  |
| --- | --- |
| **Version** | **Date** |
| 1.0.0 | 12.11.2019 |

**TABLE OF CONTENTS**

**DaBL reservation system (PHP) overview 3**

**Code Layout 3**

**Example Code Change 6**

**Summary 7**

# *DaBL reservation system (PHP) overview*

The DaBL reservation system intended for the users who want to reserve machines at DaBL. The DaBL reservation system split into two-part, front-end HTML and back-end PHP, this documentation is focused on explain back-end PHP code. All these PHP files should can be loaded by the front-end HTML web pages.

This PHP documentation is for the reservation system of Design and Build Lab. It will describe all of the functionalities in this system. Users can easily edit or add new features through this documentation. PHP functionalities wrote into three PHP files. **main.php, Event.php, Idatabase.php**.

PHP files should use the **cacert.pem** certification file which is loaded by PHP compiler. Administrators can use **phpinfo()** to find the **php.ini** and change the curl to the directory of **cacert.pem.** Also, the Google Calendar API also provided on GitHub which had been made as **vendor.rar** this file should be unzip before run reservation system.

Administrators can maintain the back-end PHP for the DaBL reservation system through the instructions of this documentation.

***Code Layout***

The following sections describe the main files you will interact with and their purposes separated between each of the separate components of the application.

**Backend:**

* **main.php**
  + This file deals with all the processes that DABL users reserving any machine at DABL. HTML will use HttpRequest to execute this file.
  + Get a string from html, and split it into the data format (email, startDateTime(reservation start time), endDateTime(reservation end time), machineName).
  + Check the user’s email in the database or not.
  + If the user’s email is not in the database, and prompt the deny information on HTML website.
  + If the user’s email is in the database, then check did user sign the waiver at DABL or not.
  + If the user didn’t sign the waiver and prompt the deny information on HTML website.
  + If the user signed the waiver, then check did the user finish the required training or not.
  + If the user didn’t finish the required training and prompt the deny information on HTML website.
  + If the user finished the required training, then check is there any conflict with the schedule.
  + If there is a conflict with the schedule and prompt the deny information on HTML.
  + If there is no conflict with the schedule, then add a new event on the calendar and prompt reservation success.
* **Event.php**
  + This file deals with all the event-related things, and this file needs google calendar API library, which had been installed on the server and can download through “PHP Composer”. Also, this file will do some processes to all the data from HTML to reserve event on google calendar.
  + This file should put under the same folder with the **credentials.json**, **token.json and token.pickle** which is provided by Google Calendar API for authority purpose.
  + This file gets the data format (email, startDateTime(reservation start time), endDateTime(reservation end time), machineName) which is sent from **main.php**
  + After it gets all the data needed by Google Calendar API, it can create a event to the calendar of account which is authorized by administrators.
  + This file also deals with the events conflict. It will grab the in schedule event to check is there any conflict.
* **IDatabase.php**
  + This file deals with all the database-related things. It already set up all connection information include server name, user name, password, database name. Also, this file checks all the user data in the database and returns a true or false value to let **main.php** check data in the database or not.
  + It will get the data that is sent from **main.php** and query mySQL database to check data.

***Example Code Change***

The most likely scenario will be the addition of a new machine for the lab. The DaBL machine is added into the MySQL database, where it will no longer function with the reservation system. The steps you would take to fix the PHP are as follows:

* Add the new machine name to the database on server
* Add the new google calendar to administrator account for the new tool
* Get the new google calendar ID
* In **Event.php**, add one line to **function getCalendarId($machineName)** before default case in the
  + case ‘machine name on frontend page’:

$this->calendarId=’google calendar Id you get’;

break;

* Before change **IDatabase.php**, administrators have to define **a new parameter name** for the new machine which only works in **IDatabase.php.**
* In **IDatabase.php**, do some change to **function fetch\_All()** before “\n”
  + .$row[‘machine name in database’].
* In **IDatabase.php**, add a line to **function search\_ByEmail($email)** in **while($row = mysqli\_fetch\_array($result))** section
  + $ **new parameter name** =$row[‘machine name in database’];
* In **IDatabase.php**, add a line to **function check\_Data($email,$colname)** in **while($row = mysqli\_fetch\_array($result))** section
  + $**new parameter name** =$row[‘machine name in database’];
* In **IDatabase.php**, add a line to **function check\_Data($email,$colname)** in **switch($colname)** section
  + case ‘machine name on frontend page’:

if($**new parameter name**==1){return TRUE;}

else return {FALSE;}

* After these steps be done, DaBL reservation system should works correctly. If not, please check these steps should be followed properly or check the error message of each file.

***Summary***

Hopefully, that provides some perspective on the reservation system of ***PHP*** backend and

the administrator can add new machines for DaBL to the system easily through following these steps in this documentation.

For more information, check out the Google Calendar API instructions. And please feel free to reach out to the team with any questions or support that you need. Thanks again - and enjoy!