Big Blue Button API Scripts

Title	Big Blue Button API Scripts
Author	Julian St. Hill
Date	March 19, 2021
Description	This document details the functioning of each script used to implement the Big Blue Button API (BBB API).
Version	1.0
Amendments	None

Table of Contents

1.0 Overview

The purpose of this document is to detail the functioning of each script used in the implementation of the BBB API with third party applications. This document can be used to implement the BBB API interface on any server or it can be used by anyone who would need to make modifications to the interface in the future.

2.0 In Scope

The following items will be discussed in this document.

- The database scripts
- The class helper file
- The database helper file
- The BBB API endpoints functions

3.0 Out of Scope

The following items are considered out of scope in this document.

- Testing – No testing or test scripts are mentioned.
- Code Snippets – Little to no code will be explained within this document. It will however, list all pertinent function names and give functionalities for each.

4.0 The Database Scripts

The scripts to create a database, tables and load initial data into the tables are all contained within the bbb_db_scripts.sql file. The script creates a database called malldemodb. This database will contain two tables.

- The mall_sessions table – This table stores session information for each room created. Some of the fields stored in this table are:
  - room_name – The unique name for each room created.
  - attendeePW – The password for an attendee joining the meeting.
  - moderatorPW – The password for the moderator of the meeting.
  - createTime – The time the meeting was create. This is represented as a big integer.
  - number_of_participants – The maximum number of participants that can join a meeting.
  - meetingID – This is the unique GUID for each meeting room.

- The bbb_secrets table – This table stores the domain of the BBB server and its security key.
    - id – This is an auto incremented identity field with an initial value of 1.
    - url – The domain of the BBB server or more precisely the URL path to the BBB APIs on the BBB server.
    - secret_key – This is the unique security key for the BBB server.

It is very important that the scripts in this file are run before attempting to use the bbb_api_lib.php. Failing to do this will result in the API calls failing.

5.0 The Class Helper File

Data that are reused throughout the bbb_api_lib.php script are stored in an object using a class structure. The properties in the class are all declared as private and they are accessed or modified using public class methods. The class file is called bbb_api_class.php and it contains the following properties.

- urlAPIPath – Path to the BBB API endpoints on the BBB server
- secret - Variable to access secret code
- maxParticipants - Maximum number of participants in a meeting room
- bannerColor - The background color of the banner
- bannerText - Text to display in the banner of the meeting room
- userID - Unique ID for each user
- attendeePSW - Attendee password
- moderatorPSW - Moderator Password.

The class object has a constructor which is used to initialize all of the properties above with the exception of 'secret'. The 'secret' property is set by a setter called *set_secretKey*. There is a corresponding *get_secretKey* to retrieve the value.

6.0 The Database Helper File

This script file (dbfunctions.php) does all the actual database work. It has functions to create a database connection, close a connection, run a select query on the database and return the results, insert a new record into a table, and update any record in the database. These functions were designed to take the SQL script as a parameter and execute it. This design gives flexibility to execute any SQL statement against the database.

The list of functions includes:

- connectDB – Creates a MySQLi connection to the MySQL database.

- closeDBConnection – This function is called at the end of every function to close the database connection.
- getRecord – This function executes a select statement on the database. The SQL statement is passed as a string parameter.
- insertRecord – This function executes an insert statement on the database. The SQL insert statement and the list of values to be bond are passed as parameters.
- updateRecord – This function updates a record in the database. The SQL statement is passed as a string parameter.

7.0 The BBB Endpoints Functions

The bbb_api_lib.php file is the work horse. It is the link between the client app and the BBB server. The code in this file allows the client to do the following via API calls.

- Create a meeting room and start the meeting.
- Join a user as a moderator.
- Join a user as a member.
- Test if a specific meeting is running.
- Get information on a specific meeting currently running.
- Get information on all currently running meetings.
- End a specific meeting.

7.1 Creating Meeting

A meeting can be created in one of two ways. It can be created by calling the createMeeting function or by calling the joinModerator function. If the user chooses to call the joinModerator function before calling the createMeeting function, a check is first made to see if the specific meeting is running and if it isn't then the createMeeting function is called. This eliminates the need for a moderator to manually create a meeting before joining.

The createMeeting function takes two parameters.

- Meeting Name – The unique name of the meeting to be created or started.
- API Variables – This parameter is created and populated from the bbb_api_class object.

If successful in creating the meeting specified, it will return an XML result with the meeting information.

A check is made by the function to see if the parameters for the specific meeting already exists in the *mall_sessions* table. If they do, they are retrieved and used to fire up the meeting again. If they are not found, the function will create the meeting and insert the new parameters for the meeting into the mall_sessions table for future reference.

7.2 Join as Moderator

As mentioned in section 7.1 the Create Meeting this function is dual purpose. It can create a new meeting if one doesn't currently exist and joins the user to the meeting as a moderator, or it can simply join a user to an existing meeting.

The function takes as parameters:

- The name of the meeting
- The name of the moderator
- The API Variable

It returns a unique string containing the URL and encrypted code to enter the room as a moderator. Each time a meeting is started the *createTime* property in the *mall_session* table is updated.

7.3 Join as Attendee

The function joinAttendee is used to call the BBB API endpoint to join a user and a member (attendee) without any moderator rights. Like the joinModerator function, it first checks to see if the meeting is running. If it is, the user is redirected to the meeting room. However, if the specific meeting hasn't started the user is notified that the meeting does not exist and the script ends.

7.4 Is Specified Meeting Running

This function (isMeetingRunning) calls the BBB API endpoint to check to see if a specified meeting is currently running. It takes as parameters the name of the meeting to check and the API values to use. The function returns *true* or *false* depending on whether the specified meeting is currently running or not.

7.5 Get Meeting Information

Information on a specific meeting can be obtained via the getMeetingInfo function call. This function takes two parameters (meeting name and API variables) and returns an XML result. This XML can be decoded and displayed to the user or it can be used for other purposes.

7.6 Get Meetings

This function gets information on all actively running meetings on the BBB server. The results are returned in XML format.

7.7 End Meetings

Meetings can be ended from within the BBB green light user interface or by calling the endMeeting function with the name of the meeting to end.