# Adobe Connect Recording Downloader

## Documentation by Ken Mayer

# May 2, 2017

1. **Compiling Sikuli**
2. **Server info**
3. **To Check on the status of the Recording Server**
4. **To Redo a bad recording download**
5. **To Delete a recording**
6. **To Undelete Recordings**
7. **To Change the Configuration of the Recording Server**
8. **To restart the Recording Server**
9. **Reading error logs on the Recording Server**
10. **Recording Server Flowchart**
11. **MYSQL Database structure**

## Compiling Sikuli

The files in the **Sikuli\_DISTRIBUTION2017\_05** folder need to be extracted, and then the file **rec\_config\_SAMPLE.py** needs to be edited (add your specific configuration variables) and renamed **rec\_config.py**. Then select all of the files in that folder (but not the folder itself!) and zip them to a zip file (in Windows you can Send to🡪zip file). Rename this zip file **AdobeRec.skl**, changing the .zip extension to .skl. Now Sikuli can read this file.

## Server info

This program runs Adobe Connect and uses the Sikuli screenreader to interpret the recording and click buttons and enter info into input fields. Therefore it needs to be set up on a server with graphic user interface. I set it up on a Windows Server 2012 R2, with the following software

* Python 2.7
* Sikuli IDE 1.0.0 Win32/Win64
* PHP 5.3 with PEAR extension
* MYSQL

The PHP programs populate a MYSQL database Adobe Connect

For PHP programs the institution-dependent constants are in the file **int\_config.php**

For Python (Sikuli) programs the institution-dependent constants are in the file **rec\_config.py**

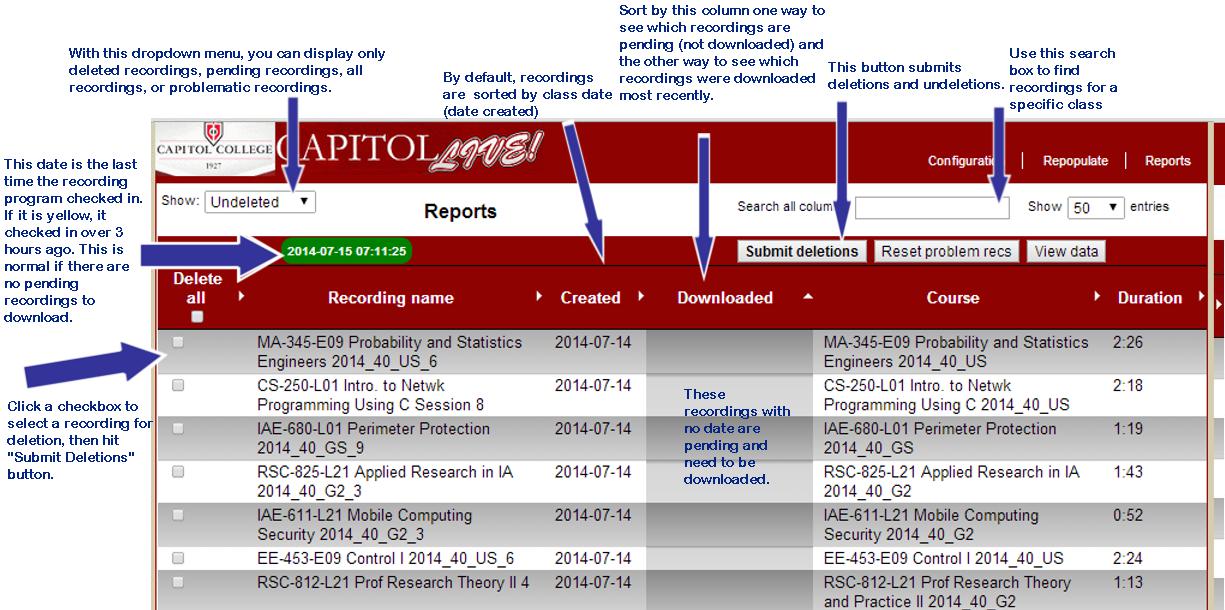
In this documentation, variables will appear with their variable names in bold surrounded by hashtags for example **#PHP\_Pass#**

## To check on the status of the Recording Server, point your browser to this page:

https://YOURSERVER.net/connect/reports.php?dist**=#PHP\_Pass#**

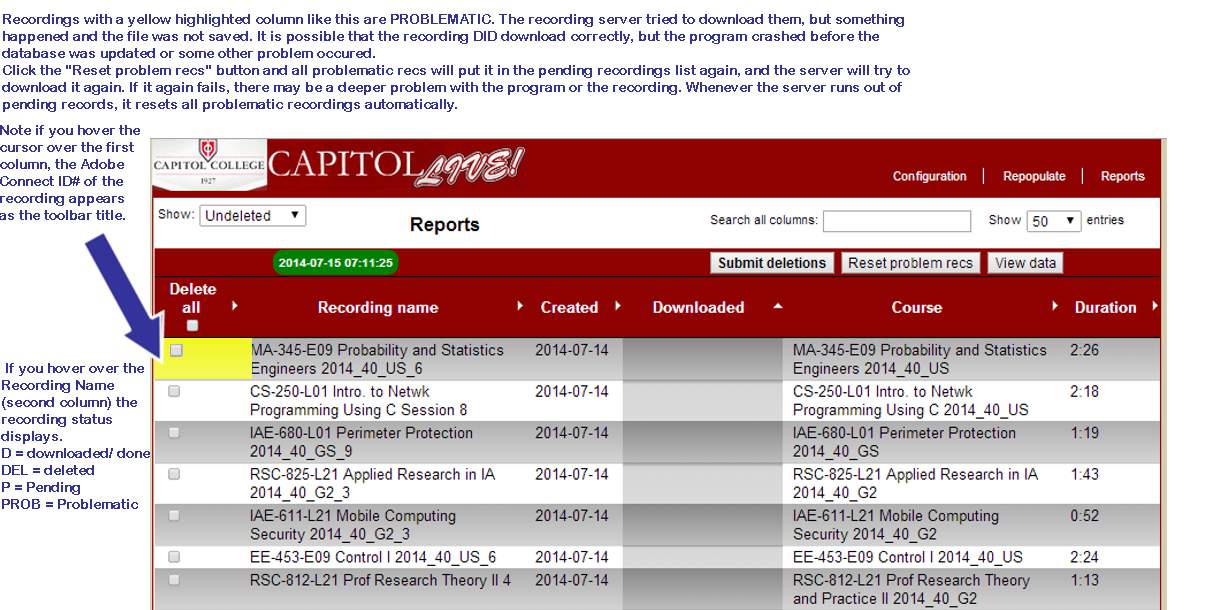
The **?dist==#PHP\_Pass#** is a security device so that random webcrawlers cannot access the pages

You should see a sortable list of recordings, by default showing the most recently recorded classes at the top. If a recording has a blank downloaded column, it either is being downloaded at that moment, is in the queue to be downloaded, or had a problem downloading.



At the top left is the latest time that the recorder server checked in. If there are pending recordings, it should be highlighted GREEN and current (within the last 3 hours). If there are no pending recordings, the recording server can be idle for hours and even days. In that case the time is highlighted YELLOW. If there are pending recordings and the time is yellow, the recording server probably has a problem. See the **To Restart the Recording Server** section below.

The second thing to look for is if there are any recordings highlighted yellow, as in the image below.



Recordings with a yellow highlighted column like this are **PROBLEMATIC**. The recording server tried to download them, but something happened and the file was not saved. It is possible that the recording **DID** download correctly, but the program crashed before the database was updated or some other problem occurred.

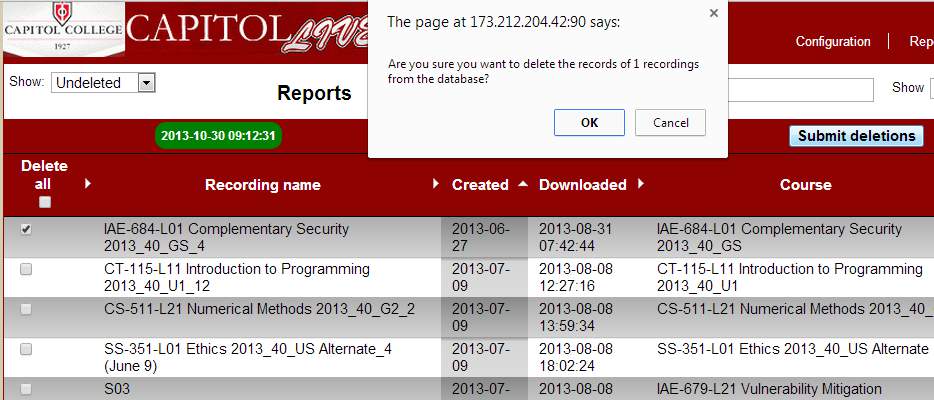
To reset the recordings, click the “Reset problem recs” button. All the problematic recordings will be put in the pending recordings list again, and the server will try to download them again. If it again fails, there may be a deeper problem with the program or the recording. The newest version of the recording server program will ***automatically*** reset problematic recordings after all the pending recordings have been downloaded.

## To redo a bad recording download

If a student or professor tells us that a download is defective, there are are two possibilities. The original recording may have a problem, in which case we can file a ticket with Adobe and they \*might\* be able to fix it. Or the downloaded copy we have may have a problem. Note that our recordings downloads only last about 4 ½ hours, so if the problem is that they stop there, that is deliberate, to prevent wasting processing time on empty recording hours. To reset a recording and have it do it over, simply delete it and then undelete it (see below). It will be added to the queue of pending recordings.

## To delete a recording

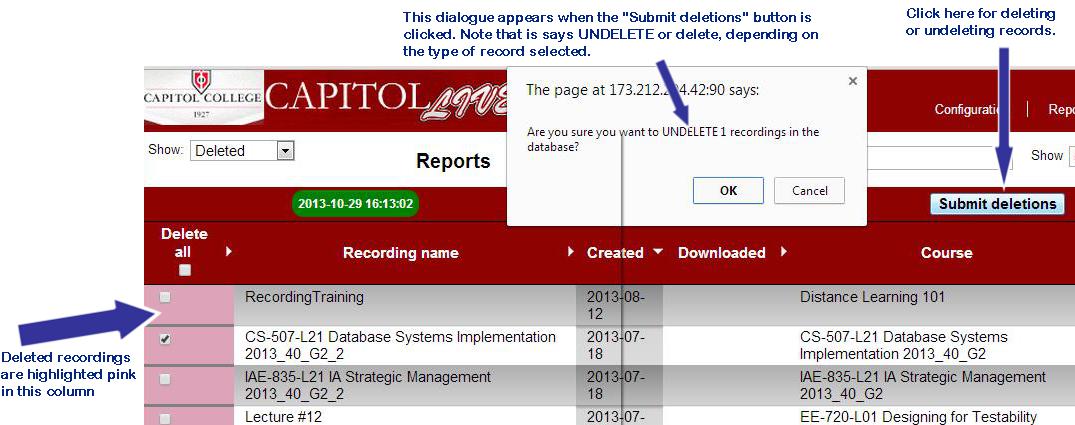
click on the checkbox beside it, and then hit the “Submit deletions” button. The page will confirm your decision. You will not actually delete any files in Adobe Connect or on our recording server, but the database will ignore the recording from now on. It will not try to download it if it was pending.



**You may need to delete a recording if you get a report from a student or a professor that a downloaded recording is corrupt, but the recording hosted at Adobe is OK.** Perhaps the sound or some feature was not working when the recording was downloaded. Simply delete the recording and UNDELETE it (see next page). Then the recording will be on the queue for downloading again. You might want to check the recordings made at the exact same time as the corrupt one (with the exact same downloaded time), to make sure that they are OK. Students might not report them until weeks later.

## To Undelete Recordings.

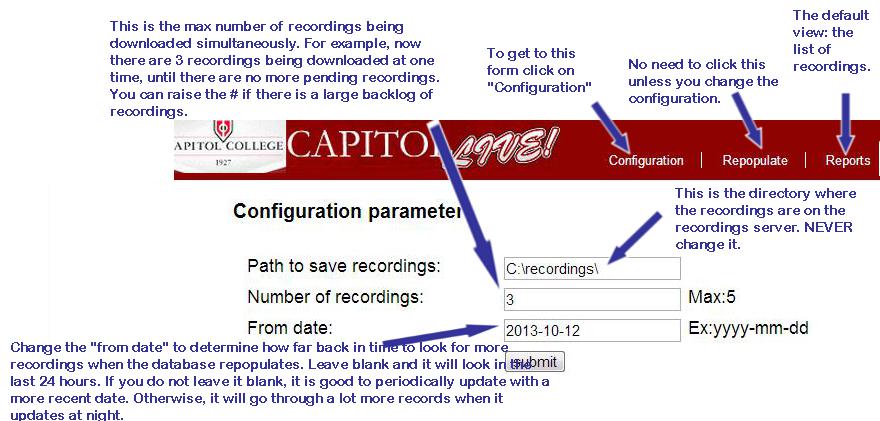
If you display DELETED recordings, you can click their checkboxes to undelete them. After they are deleted they will return to the PENDING queue. The recording server will attempt to download them, even if they had already been downloaded before.



## To Change the Configuration of the Recording Server

Most likely you will not need to change the configurations, but this guide explains how and why. Essentially, the configuration page puts three options into a database.

1. **Where to save the downloaded recordings.**  
   Do not mess with this, or the recordings will not be found by our integration script.
2. **How many recordings should be downloaded at the same time.**Our default mode, shown here, is three. That means that typically the server is playing and downloading three Adobe Connect classes simultaneously. If you were to listen in, you would hear three professors droning on simultaneously. You can have more, particularly if the recording server was down for a few days/weeks and you have a long backlog.
3. **What date the recording server uses to repopulate the database.** Every night, the server looks in Adobe Connect for all recordings past the date here. If the date is blank, it will use the day before the current day.



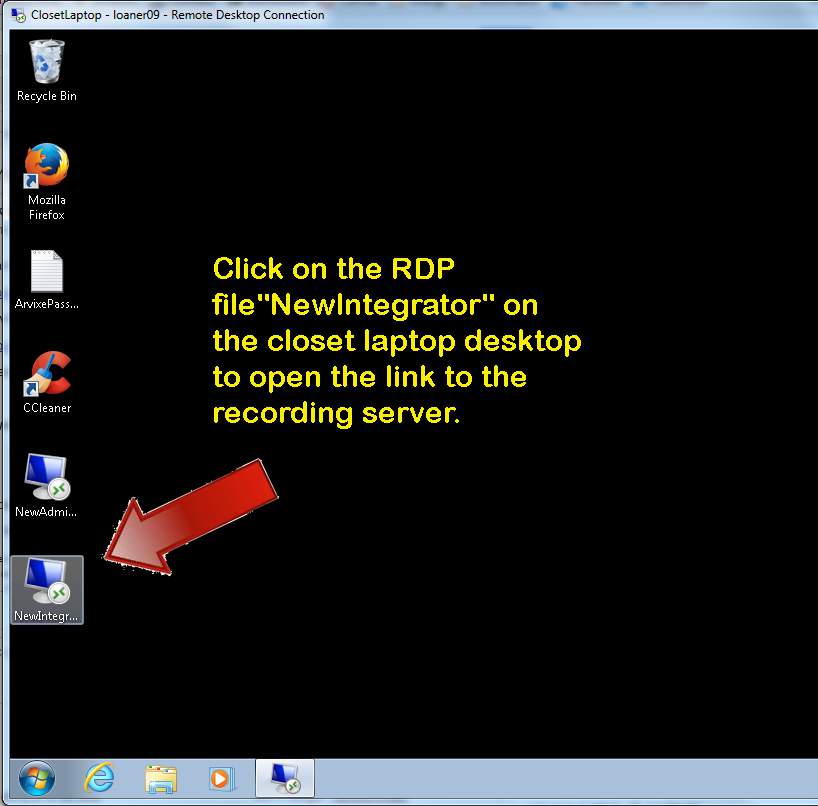
## To restart the Recording Server

If the recording server program is not working, it probably just needs to be started again. Note that this does not mean turning the server off or taking it offline, but rather, connecting via RDP and running the program.

Use the program **Remote Desktop Connection**. You will be connection to a virtual machine that is itself using RDP to connect to the server, so it is “watching” the Adobe Connect recordings 24/7

As a shortcut, you should save the info to a RDP file. Clicking on this file will automatically open Remote Desktop Connection with the login information to access the laptop. It will probably still ask for your password, since our network security prefs do not allow them to be stored.



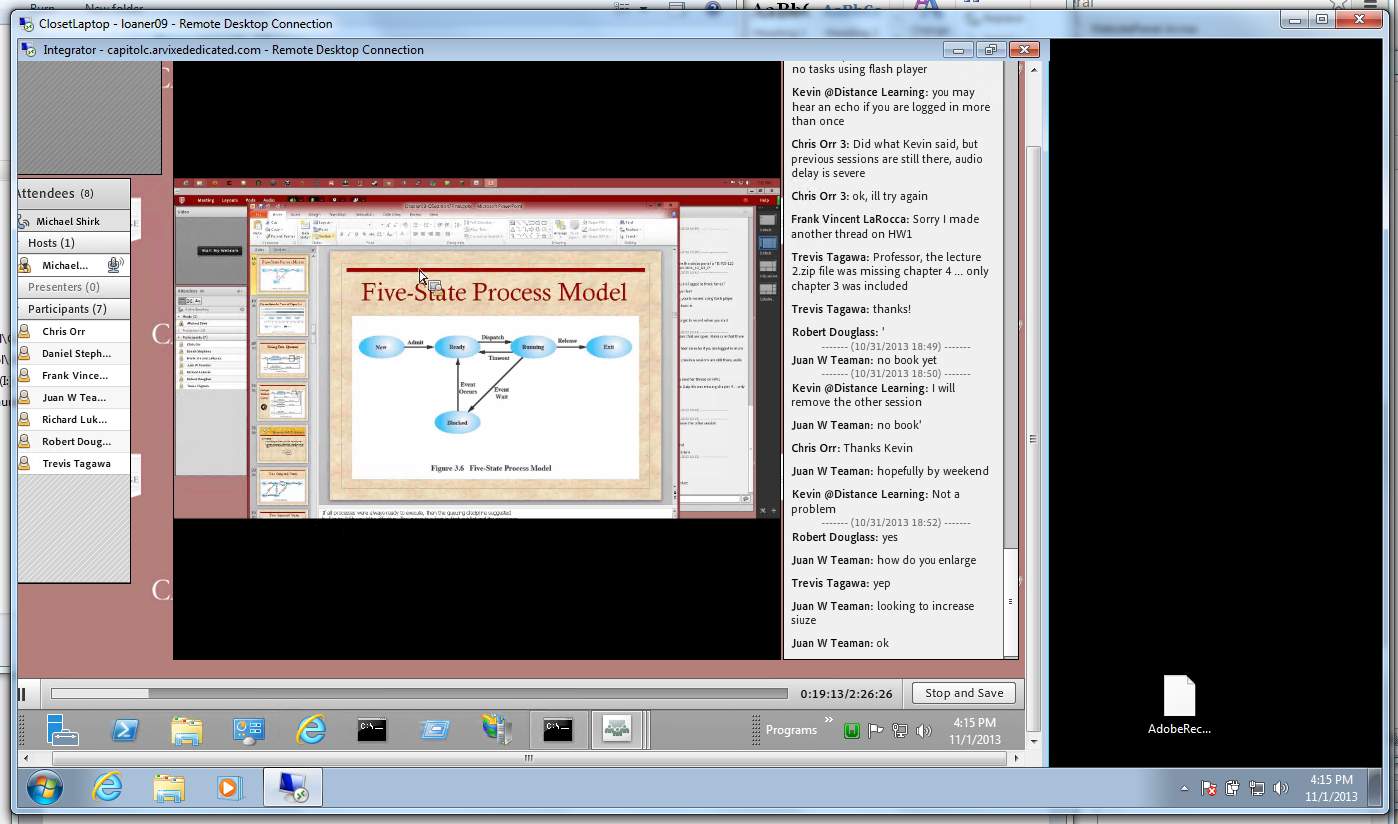
A window will open on your screen with the desktop of the closet laptop. If the laptop is not connected to the recording server, you might see a blank desktop like this:

Click on the RDP file, NewIntegrator to link to the recording server.

When the recording server window opens within the laptop window, click on the Start Recording shortcut on the desktop to start the recorder.



When the recording server is working, most of the laptop desktop will be filled with another remote desktop, since the closet laptop is remotely RDPing into the recording server in the cloud. Within that desktop Adobe Connect recordings will be playing. If all is working correctly it should look something like the following image:



As you can see from the icons on the bottom of the inside window, the recording server is running three instances of Adobe Connect simultaneously and also the batch file **Start\_Recording.cmd**. All is well. Often a problem will cause extra dialogue windows to be open. In these cases, you should shut down all of the windows, all Adobe Connect windows, all Internet Explorer windows, and the batch file. Then you can start the batch file again.

## Reading error logs on the Recording Server

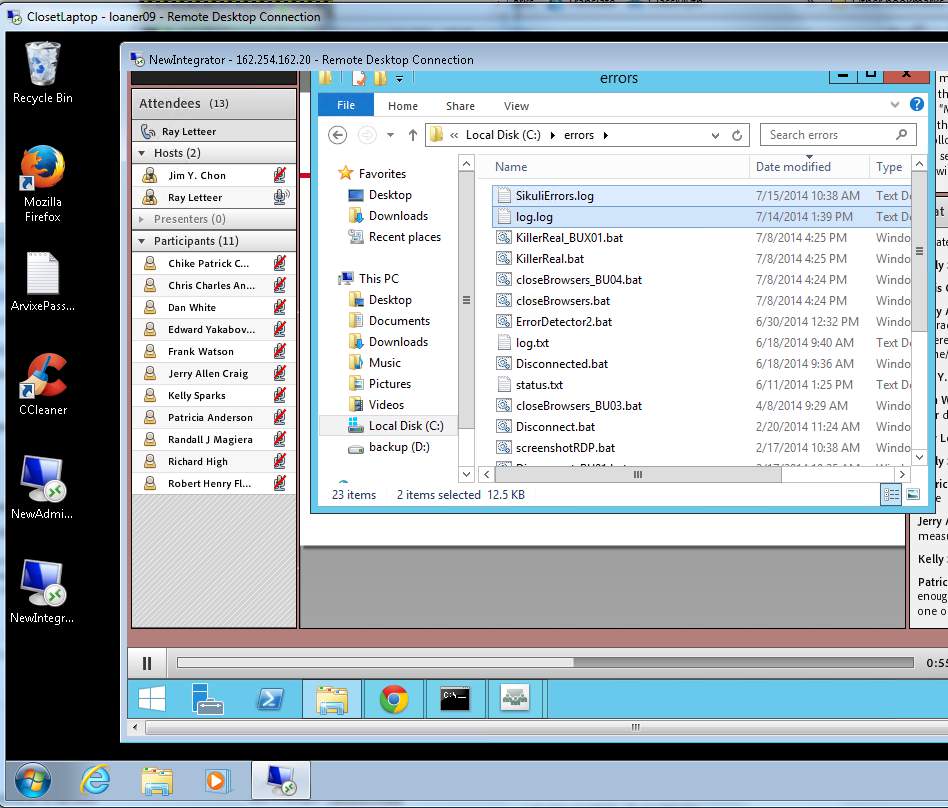
There are two different recording logs

**1. C:\errors\log.log** records any actions done by batch files

**2. C:\errors\SikuliErrors.log** records any actions done by the Sikuli program

If you want to check up on any problems, you should copy the files to your own computer and read them there instead of opening them in the recording server. Otherwise, you might inadvertently “lock” the files and prevent the recording program from adding more entries.

Periodically I store the log files and start new blank ones.

****

## Recording Server Flowchart

|  |  |  |
| --- | --- | --- |
| Server - web by cyberscooty - clip art, clipart, computer, informatique, it, ordinateur, server, serveur, web, http://www.star.nesdis.noaa.gov/star/images/meetings/GPM2013/connect.jpg  **Adobe Cloud Servers**  Recordings stored in a collection of xml and flv files which are woven together into an interactive seamless experience.  Between eight and 30 hours of recordings are added each day. | http://iconizer.net/files/Vista/orig/dedicated_server.png **Integration Server**  Authenticates Canvas users and points them to interactive recordings at Adobe and downloadable recordings  Downloads and stores Adobe Connect recordings as they are made  2TB of storage and CPU running Windows Server 2012, MySql, and PHP 5.3, running a PHP program that determines if there are new recordings and puts them into a queue to be downloaded. Then a Sikuli program runs to automatically play the recordings on the server and store them | **End user**  Can access interactive programs on Adobe cloud server or download the recordings from Integration Server through Canvas |
|  | Each class in Canvas has a page linking via BLTI authentication to the Integration Server which links to the correct Adobe Connect classroom and recordings | **Viewing VPS**  Connects to Integration Server with Remote Desktop Access, provides graphics and keyboard for the Sikuli program on the Integration Server. |

## MYSQL Database structure

-- PHP Version: 5.3.26

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

SET time\_zone = "+00:00";

--

-- Database: `cc\_ac\_structure`

--

-- --------------------------------------------------------

--

-- Table structure for table `archive\_views`

--

CREATE TABLE IF NOT EXISTS `archive\_views` (

`starttime` varchar(25) DEFAULT NULL,

`start\_raw` int(14) NOT NULL,

`login` varchar(30) DEFAULT NULL,

`course` varchar(25) DEFAULT NULL,

`recname` varchar(90) DEFAULT NULL,

`duration` int(11) DEFAULT NULL,

`useragent` varchar(160) DEFAULT NULL,

`ip` varchar(20) DEFAULT NULL,

`basestring` text,

`endtime` varchar(25) DEFAULT NULL,

`total` int(10) DEFAULT NULL,

`comment` text,

`end\_raw` int(14) NOT NULL,

`session` char(50) NOT NULL,

`rec\_url` varchar(100) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=utf8;

-- --------------------------------------------------------

--

-- Table structure for table `cc\_configdetails`

--

CREATE TABLE IF NOT EXISTS `cc\_configdetails` (

`root\_path` varchar(255) DEFAULT NULL,

`search\_criteria` varchar(255) DEFAULT NULL,

`number\_of\_instances` int(11) DEFAULT NULL,

`download\_from\_date` date DEFAULT NULL,

`status\_check` datetime DEFAULT NULL,

`Stupid\_unique` tinyint(1) DEFAULT NULL,

UNIQUE KEY `root\_path` (`root\_path`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `cc\_download\_log`

--

CREATE TABLE IF NOT EXISTS `cc\_download\_log` (

`scoid` int(11) NOT NULL COMMENT 'scoid of recording',

`date` date DEFAULT NULL,

`start\_time` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',

`end\_time` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',

`num\_downloaded` int(11) DEFAULT NULL,

PRIMARY KEY (`scoid`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `cc\_recordings\_archive`

--

CREATE TABLE IF NOT EXISTS `cc\_recordings\_archive` (

`scoid` varchar(255) DEFAULT NULL,

`foldername` varchar(255) DEFAULT NULL,

`instructor` tinytext COMMENT 'folder name should be instructor',

`url` varchar(255) DEFAULT NULL,

`meetingname` varchar(255) DEFAULT NULL,

`meetingurl` varchar(255) DEFAULT NULL,

`recordingname` varchar(255) DEFAULT NULL,

`datecreated` varchar(255) DEFAULT NULL,

`duration` int(32) DEFAULT NULL,

`attendance` smallint(4) DEFAULT NULL COMMENT 'from Adobe API',

`date\_modified` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

UNIQUE KEY `scoid` (`scoid`),

UNIQUE KEY `scoid\_2` (`scoid`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `cc\_recordings\_in\_ac`

--

CREATE TABLE IF NOT EXISTS `cc\_recordings\_in\_ac` (

`scoid` varchar(255) DEFAULT NULL,

`foldername` varchar(255) DEFAULT NULL,

`instructor` tinytext COMMENT 'folder name',

`url` varchar(255) DEFAULT NULL,

`meetingname` varchar(255) DEFAULT NULL,

`meetingurl` varchar(255) DEFAULT NULL,

`recordingname` varchar(255) DEFAULT NULL,

`datecreated` varchar(255) DEFAULT NULL,

`duration` int(32) DEFAULT NULL,

`attendance` smallint(4) DEFAULT NULL COMMENT 'student attendance',

`date\_modified` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

UNIQUE KEY `scoid` (`scoid`),

UNIQUE KEY `scoid\_2` (`scoid`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `cc\_recordings\_saved`

--

CREATE TABLE IF NOT EXISTS `cc\_recordings\_saved` (

`scoid` varchar(255) DEFAULT NULL,

`status` char(10) DEFAULT 'P',

`datedownloaded` datetime DEFAULT NULL,

UNIQUE KEY `scoid` (`scoid`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `cc\_rec\_attendance`

--

CREATE TABLE IF NOT EXISTS `cc\_rec\_attendance` (

`scoid` int(11) DEFAULT NULL COMMENT 'recording scoid',

`session\_id` int(11) DEFAULT NULL COMMENT 'meeting session id#',

`transcript\_id` int(11) DEFAULT NULL COMMENT 'unique # for each login',

`course\_sco` int(11) DEFAULT NULL COMMENT 'sco for the course',

`person` tinytext,

`status` tinytext COMMENT 'type of enrollment',

`start` datetime NOT NULL,

`stop` datetime NOT NULL,

`seconds` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `errors`

--

CREATE TABLE IF NOT EXISTS `errors` (

`time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`useragent` varchar(120) DEFAULT NULL,

`ip` varchar(20) DEFAULT NULL,

`login` varchar(25) DEFAULT NULL,

`course` varchar(45) DEFAULT NULL,

`error` varchar(300) DEFAULT NULL,

`number` int(11) NOT NULL AUTO\_INCREMENT,

PRIMARY KEY (`number`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=1 ;

-- --------------------------------------------------------

--

-- Table structure for table `live\_courses`

--

CREATE TABLE IF NOT EXISTS `live\_courses` (

`course\_code` tinytext NOT NULL,

`name` tinytext NOT NULL COMMENT 'course name',

`prof` tinytext NOT NULL,

`prof\_id` tinytext,

`description` tinytext COMMENT 'Adobe Connect description',

`connect\_id` int(10) unsigned NOT NULL DEFAULT '0',

`canvas\_id` int(10) unsigned DEFAULT NULL,

`department\_canvas` smallint(6) DEFAULT '1' COMMENT 'Canvas department number',

`enrollment` tinyint(4) DEFAULT NULL COMMENT 'from Canvas API',

`start` date NOT NULL,

`end` date NOT NULL,

`comment` tinytext,

`day` tinytext NOT NULL,

`time` tinytext,

`visible` tinyint(1) NOT NULL DEFAULT '1',

`last\_updated` timestamp NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`connect\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `recs`

--

CREATE TABLE IF NOT EXISTS `recs` (

`course` varchar(25) DEFAULT NULL,

`course\_sco` int(12) DEFAULT NULL,

`rec\_name` varchar(72) DEFAULT NULL,

`rec\_url` varchar(43) DEFAULT NULL,

`rec\_sco` int(12) DEFAULT NULL,

`date` char(30) DEFAULT NULL,

`duration` int(12) DEFAULT NULL,

`in\_db` tinyint(1) DEFAULT NULL,

`in\_archive` tinyint(1) DEFAULT NULL,

UNIQUE KEY `rec\_sco` (`rec\_sco`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8;

-- --------------------------------------------------------

--

-- Table structure for table `views`

--

CREATE TABLE IF NOT EXISTS `views` (

`starttime` varchar(25) DEFAULT NULL,

`start\_raw` int(14) NOT NULL,

`login` varchar(30) DEFAULT NULL,

`course` varchar(25) DEFAULT NULL,

`recname` varchar(90) DEFAULT NULL,

`duration` int(11) DEFAULT NULL,

`useragent` varchar(160) DEFAULT NULL,

`ip` varchar(20) DEFAULT NULL,

`basestring` text,

`endtime` varchar(25) DEFAULT NULL,

`total` int(10) DEFAULT NULL,

`comment` text,

`end\_raw` int(14) NOT NULL,

`session` char(50) NOT NULL,

`rec\_url` varchar(100) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=utf8;