CSWeb User's Guide

Version 8.1.6

International Programs Center
Population Division
U.S. Census Bureau

4600 Silver Hill Road
Washington, DC 20233

cspro@lists.census.gov

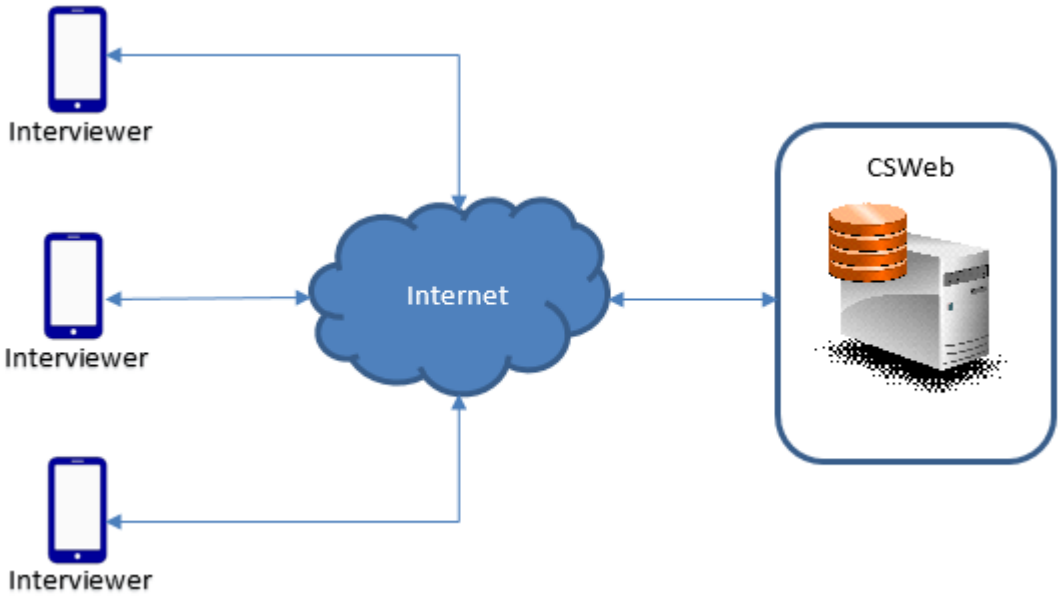
Table of Contents

Table of Contents 1
CSWeb User's Guide 2
Introduction to CSWeb 2
Server Setup 4
Minimum Server Requirements 4
Apache 5
Apache Server Setup 5
Apache CSWeb Setup 5
IIS 7
IIS Server Setup 7
IIS CSWeb Setup 8
Production 10
Running CSWeb in Production 10
CSWeb Best Practices 10
PHP 11
MySQL / MariaDB 11
Apache / IIS 11
CSWeb Server Optimization Quick Reference 12
OPcache 12
APCu 13
FastCGI Settings 13
Application Pool Settings 13
PHP-FPM Pool (/etc/php/[version]/fpm/pool.d/www.conf) 13
Apache Settings (/etc/apache2/apache2.conf or /etc/httpd/conf/httpd.conf) 14
Response Compression (mod_deflate) 14
WAMP (IIS + Windows) 14
LAMP (Apache + Linux) 14
Upgrading 16
Upgrading from Previous Versions 16
How to ... 17
Accessing Data 17
Managing Roles 17
Adding Users 19
Data Settings 19
Additional Configuration Options 20
Upload Files 21
Map Report 22
Sync Report 23
Troubleshooting 25
Get Help 26

CSWeb User's Guide

Introduction to CSWeb

CSWeb is a web application, a CSPro synchronization service, that allows users to securely transfer cases (questionnaires) or files between client devices running CSEntry and a web server. CSWeb requires a web server running PHP and MySQL. CSWeb 8.0 and greater support both the original and JSON formatted dictionaries.



Direct synchronization between interviewers and central server over the Internet

CSWeb eliminates the need to transfer data files by allowing users to instead transfer cases between client devices and a web server. CSEntry data entry applications that are configured to use CSWeb synchronization take advantage of *Smart Synchronization*, a feature that transfers only cases that are added or modified since the last successful synchronization. Smart Synchronization reduces data transfer costs and Internet bandwidth usage.

CSWeb uses a MySQL database on the server to store the cases for the census or survey data entry instrument. Unlike a file based synchronization that requires application designers to manage the concatenation of data files, CSWeb allows users to download a single file containing all the cases for the data entry instrument (once the cases are transferred from the client devices to the server).

The code to run CSWeb can be downloaded from the CSPro download webpage. More information about how to use CSPro's synchronization features can be found in the CSPro User's Guide Synchronization Overview.

CSWeb is intended for large surveys and censuses. For small- to medium-sized surveys, Dropbox or FTP synchronization is sufficient and far simpler to set up than CSWeb. For Dropbox synchronization, no server is required at all, making it very easy to get started. In addition, setting up CSWeb requires knowledge and experience with server administration and website technologies. If you do not have experience setting up and maintaining web servers, you will find setting up CSWeb to be very difficult.

This guide contains the following information:

Server Setup

- Minimum Server Requirements
- Apache Server Setup

- Apache CSWeb Setup
- IIS Server Setup
- IIS CSWeb Setup
- Running CSWeb in Production
- CSWeb Best Practices
- CSWeb Server Optimization Quick Reference
- Upgrading from Previous Versions

How to ...

- Accessing Data
- Managing Roles
- Adding Users
- Data Settings
- Upload Files
- Map Report
- Sync Report
- Troubleshooting
- Get Help

Server Setup

Minimum Server Requirements

Below are the minimum requirements to run CSWeb on an Apache or IIS server.

Manually verify:

- **Apache 2.0** or above or **IIS** 7.0 or above
- URL Rewrite Module enabled

Requirements that will be verified by setup script:

- MySQL 8 or greater
- PHP 8.1+
- **Settings in php.ini**
    - enable_post_data_reading on
    - post_max_size=8M
    - **Extensions in php.ini**
        - extension=php_curl.dll
        - or allow_url_fopen=On
        - extension=php_fileinfo.dll
        - extension=php_openssl.dll
        - extension=php_pdo_mysql.dll
        - extension=php_pdo.dll
        - extension=php_dom.dll
        - extension=php_zip.dll (necessary with some installations of PHP)
- **Guzzle** (PHP HTTP client) one of the following must be true
    - PHP 8.0+
    - or extension=php_curl.dll
    - or The CA bundle is installed
- **Files directory**
    - Must exist and be writeable

The above requirements are sufficient to set up a CSWeb server for testing. To use CSWeb for a production data collection operation see Running CSWeb in Production.

Apache

Apache Server Setup

The setup of an Apache server is beyond the scope of this document. Fortunately, there already exists an extensive amount of documentation on the topic. Below we have provided links for getting started.

Web Development Environment

There are a number of different web development environments that will simplify the setup and allow you to run and test CSWeb locally. If you are developing using Windows you may find WampServer helpful. It will install and configure Apache, MySQL, and PHP.

Apache Server

If you are looking to setup a server to host CSWeb you will need to install and configure Apache, MySQL, and PHP.

URL Rewrite Module

Regardless of your setup the Apache **rewrite_module** must be enabled. To do this you will need to edit the **httpd.conf** file. There may be multiple copies of this file on your computer. Once, you have located the file make a back up and then uncomment the following line **LoadModule rewrite_module modules/mod_rewrite.so**. Then restart Apache.

Apache CSWeb Setup

Below we have provided a general overview of the steps required to run CSWeb on an Apache server. In this example, it is assumed WampServer has been installed and configured. **WampServer is only meant for a development environment.** WampServer's configurations for Apache, MySQL, and PHP will be optimized for development and not performance or security. In a production environment WampServer (or XAMPP) should not be used. Instead, Apache, MySQL, and PHP should be installed individually. On Linux this is referred to as the LAMP stack and on Windows as the WAMP stack.

Add Project Files

Copy the **root directory** of your CSWeb project to **<Drive>:\wamp64\www**. The result may look like **C:\wamp64\www\csweb**. Note that with Apache, the case of the folder name matters. **CSWeb** and **csweb** will require different URLs to access the server. For simplicity, we recommend using all lowercase e.g. **csweb**.

Start Apache and MySQL

Make sure Apache and MySQL are running.

Add Database

Add a new database to use for CSWeb using the MySQL command line, MySQL workbench, or phpMyAdmin. Create a user with access to the new database. For security reasons the database user should have a corresponding password.

Run Setup Script

Open a browser and in our case navigate to **localhost/csweb/setup**. The setup script will first check to make sure your server meets the prerequisites. If your server does not pass, refer to the Minimum Server Requirements. Update your server and run the script again. If your server passed the script will ask for the last few configuration details.

1. **Database name**: the name of the database you created earlier.

2. **Hostname**: this will typically be **localhost**.
3. **Database username**: by default this will be **root**.
4. **Database password**: by default this may be **blank**. Do not use the default password for a live server.
5. **CSWeb admin password**: create a password to pair with the default user **admin** to log into CSWeb.

Now verify the final two fields.

1. **Path to files directory**: for this setup, **C:\wamp64\www\csweb\files**
2. **CSWeb API URL**: for this setup, **http://localhost/csweb/api**

If everything is correct you will hit next and receive the "Setup Complete!" message. From here log in using **admin** as the username and the **CSWeb admin password** you just created. Once you have run the setup script you will **not** be able to run it again unless you delete **src\config.php**.

IIS

IIS Server Setup

Below are the prerequisites for setting up an IIS server.

Install Internet Information Services (IIS)

On the lefthand column of **Programs and Features** select **Turn Windows features on or off**. Here you are able to select the check box to turn on **Internet Information Services**. Also turn on **IIS Management Console** under Web Management Tools under IIS, and **CGI** under Application Development Features under IIS

Install PHP 8.1 or above for Windows

The x64 Non-Thread Safe is the most stable for IIS Services Version 10. Unzip this into a directory that has permission to execute on windows. If you do not setup the permissions, then a fatal exception will appear

Install PHP Manager

You can find the PHP Manager from within the **IIS Manager** by clicking **Get New Web Platform components**. Select the **Downloads** tab and install **PHP Manager 1.5.0 for IIS 10**. Once this is done open the IIS Manager, find the **PHP Manager** under **Features View**, and use **Register new PHP version** to select the php-cgi.exe of the PHP you downloaded. You can use the PHP manager to change the PHP version or PHP settings you want IIS to use.

Handle Additional Verbs

In the **IIS Manager** select the **Features View** and then double click **Handler Mappings**. Right-click on **PHP_via_FastCGI** or **CGI** and select **edit**. Click **Request Restrictions...**, select the **Verbs** tab, select **One of the following verbs** and enter **GET,HEAD,POST,PUT,PATCH,DELETE,OPTIONS**. You may then be prompted to double quote the path under **Executable**.

Install URL Rewrite Module

Microsoft's web installer will also install the **URL Rewrite** module. However, if you use another installation method you may need to install it manually. You can verify whether it is installed or not in IIS Manager's **Features View**. For more information and to download the URL Rewrite module click here.

Edit php.ini

Locate the php.ini file for the new installation of PHP. You may have multiple copies on your computer. The default install path will be under **Program Files (x86)** or **Program Files** depending on whether you installed the 32 or 64 bit version. The root of the install directory will be **PHP**. You will find the php.ini file in the subdirectory that was named for the **version** you installed. You may need to open the file as an administrator to have permission to edit it. Open **php.ini** with **Notepad** and search for **;extension=pdo_mysql**. Uncomment it by removing the colon. Copy and paste **extension=php_fileinfo.dll** on the next line. Save and close the file.

Install MySQL Server 5.5.3+

Download the MySQL Server here. For help with installation see the documentation. Do not forget to start the server. If you installed MySQL Server as a service, you can start it by doing the following:

1. Win + R
2. Run services.msc
3. Right-click on MySQLXX
4. Start

Alternatively, if you did not install MySQL server as a service or have a web development environment like WampServer installed, you can start MySQL Server from the command line. You will need to open a command prompt and change directory to the directory that contains **mysqld.exe**.

```
> mysqld
```

IIS CSWeb Setup

Below are the prerequisites for running CSWeb on an IIS server.

Add Project Files

Copy the **root directory** of your CSWeb project to **<Drive>:\inetpub\wwwroot**. The result may look like **C:\inetpub\wwwroot\csweb**.

Update Directory Permissions

You will find the below directories in your document root. Their permissions will need to be updated as shown for **IUSR** and **IIS_IUSRS** group accounts.

- **files**: Read and Write
- **var**: Read and Write
- **src**: Read and Write
- **app\config**: Read and Write

Let us use the **files** directory as an example. Right-click on **files** and select **Properties**. Then click the **Security** tab. Press the **Add...** button and type **IUSR** under **Enter the names to select**. Press the **Check Names** button and confirm your changes. Select the user name you just added, **IUSR**, then update the permissions under **Permissions for IUSR** so that **Read** and **Write** are allowed. Make sure to assign the same permissions for **IIS_IUSRS** group as well. Now repeat this process for **var**, **src**, and **app\config** setting the correct permissions.

Add Database

Do not forget to start the server. If you installed MySQL Server as a service you can start it by doing the following:

1. Win + R
2. Run services.msc
3. Right-click on MySQLXX
4. Start

Alternatively, if you did not install MySQL server as a service or have a web development environment like WampServer installed, you can start MySQL Server from the command line. You will need to open a command prompt and change directory to the directory that contains **mysqld.exe**.

```
> mysqld
```

Next, you can use MySQL Monitor to add a database. Connect to MySQL Server. You will need to open a command prompt and change directory to the directory that contains **mysql.exe**.

```
> mysql -u root -p
```

Now, add the database.

```
> create database <name-of-your-database>;
```

As an example, I have named my database **cspro**.

```
> create database cspro;
```

To verify our work, we can list the databases.

```
> show databases;
```

If you would prefer not to use the command line we recommend MySQL Workbench.

Start IIS

If IIS is installed you will find the **Internet Information Services (IIS) Manager** shortcut under **Administrative Tools** in the **Control Panel**. Double click the shortcut to launch the IIS Manager. In the **Connections** tree on the left-hand side expand **Sites** and select **Default Web Site**. You will see your document root below this. On the right-hand side there is an **Actions** panel. Under **Manage Website** click **Start**.

Run Setup Script

Open a browser and in our case navigate to **localhost/csweb/setup**. The setup script will first check to make sure your server meets the prerequisites. If your server does not pass, refer to the Minimum Server Requirements. Update your server and run the script again. If your server passed the script will ask for the last few configuration details.

1. **Database name**: the name of the database you created earlier.
2. **Hostname**: this will typically be **localhost**.
3. **Database username**: by default this will be **root**. This is created during the installation of the MySQL Server.
4. **Database password**: by default this may be is **blank**. Using the default password is a security issue. This is set during the installation of the MySQL Server.
5. **CSWeb admin password**: create a password to pair with the default user **admin** to log into the web app.

Now verify the final two fields.

1. **Path to files directory**: for this setup, **C:\inetpub\wwwroot\csweb\files**
2. **CSWeb API URL**: for this setup, **http://localhost/csweb/api**

If everything is correct you will hit next and receive the "Setup Complete!" message. From here log in using **admin** as the username and the **CSWeb admin password** you just created. Once you have run the setup script you will **not** be able to run it again unless you delete **src\config.php**.

Production

Running CSWeb in Production

Using CSWeb for a production data collection operation requires additional steps in order to make the server accessible to devices in the field and to ensure proper data security.

Domain Name

In order to connect to the server from devices outside your local network, such as tablets in the field, you will need to register a domain name for your server. The instructions in this document describe using http://localhost to access your CSWeb server. This will only work when accessing CSWeb from the server itself. In order to access the server from a tablet or another computer you will need domain name such as http://census.gov. There are many companies online that can register domain names. If you already have a website then you may be able to use that domain, or a subdomain, for your CSWeb server.

Network Security

If your server is connected to the Internet, it is important to ensure that you have network security systems in place to prevent unwanted intrusion and access to your data. Such systems will generally include a firewall. Your firewall must be configured to allow HTTP and/or HTTPS traffic in order for devices to connect to CSWeb over the Internet. If you plan to store confidential survey data on your CSWeb server you should seek assistance from an expert in server security.

SSL Certificate

TLS/SSL encrypts the communication between your server and devices in the field. TLS/SSL is also known as https. If you use CSWeb without https, passwords and data are transferred as plain text and may be intercepted during transmission. For confidential data it is important to use TLS/SSL to encrypt all data being transferred between devices in field and the server. This can be done easily by configuring the web server (Apache or IIS) to use https instead of http. This requires an SSL certificate for your domain. There are various companies and organizations that can provide SSL certificates.

See also: CSWeb Best Practices, CSWeb Server Optimization Quick Reference

CSWeb Best Practices

Security

Securing your CSWeb installation is critical, especially for sensitive survey or census data. Consult your systems administrator to review the following:

- MySQL should not be exposed directly to the Internet.
- HTTPS should be enabled for all CSWeb traffic.
- User accounts should follow the principle of least privilege.

For large-scale censuses: Some organizations establish additional layers of security, such as placing MySQL in a DMZ and restricting device sync to a closed mobile network (APN) so that data never travels over the public Internet. This approach has been used successfully in national census deployments to ensure complete network isolation between field devices and backend infrastructure. In these cases administrators ensured that:

- The CSWeb dashboard is not publicly accessible on the Internet.

- The REST API is restricted to known devices only.

Performance Tuning

The default CSWeb installation is suitable for most deployments. However, for large-scale surveys or censuses, server tuning may significantly improve reliability and response times under heavy load.

Consider performance tuning if you are experiencing any of the following:

- Enumerators are reporting slow or failed syncs.
- The server becomes unresponsive or sluggish during peak data collection periods.
- Sync times are noticeably longer than expected given your network conditions.

Before making any changes, consult a qualified systems administrator. The following areas are most relevant:

PHP

Default PHP settings are conservative and not optimized for production workloads. Key areas for your administrator to review:

- **OPcache** — By default, PHP compiles your application code from scratch on every single request. OPcache stores the compiled result in memory so subsequent requests are served much faster. This should always be enabled in a production environment.
- **APCu** — Allows frequently used application data such as lookup tables and reference dictionaries to be stored in memory rather than read from disk on every request. Particularly beneficial under high concurrent load.
- **Production error reporting** — Default PHP error settings are intended for development environments and can expose sensitive system information. Production settings should be confirmed before going live.

MySQL / MariaDB

Default MySQL and MariaDB settings are designed for minimal hardware and may not perform well under heavy concurrent load. Key areas for your administrator to review:

- **InnoDB buffer pool** — This is the most impactful MySQL setting. It controls how much data MySQL keeps in memory rather than reading from disk. The default value is very small and should be increased proportionally to the available server RAM.
- **InnoDB log file and buffer sizes** — Controls how MySQL handles write operations in memory before committing them to disk. Proper sizing reduces disk I/O significantly under high concurrent write load, which is the typical pattern for field data sync workloads.
- **Connection limits** — MySQL has a maximum number of simultaneous connections. This should be set to comfortably exceed the expected number of concurrent sync requests to prevent connection errors during peak collection periods.

Storage: Using SSD storage for the MySQL data directory can significantly improve performance under concurrent load and is worth considering for large-scale deployments. Spinning disk (HDD) performance degrades under high numbers of simultaneous read and write operations, which is the typical pattern when many enumerators sync at the same time.

Apache / IIS

Default web server settings may limit the number of concurrent requests that can be processed, causing queuing under heavy load. Key areas for your administrator to review:

- **PHP process pool size** (PHP-FPM pool on Apache / FastCGI pool on IIS) — When a sync request arrives, the web server hands it to a PHP process to handle. There is a limit to how many PHP processes can run simultaneously. By default this limit is set conservatively. Tuning helps handle peak traffic.

- **Worker process timeouts** — If a PHP process takes too long to respond, the web server terminates it and returns an error to the device. Default timeout values can cause legitimate sync requests to be cancelled prematurely under heavy server load. Adjusting timeouts helps server with processing the requests better.
- **Response compression** — The web server can compress data before sending it to field devices, reducing the amount of data transmitted over the network. This is particularly beneficial for enumerators working in areas with poor or limited connectivity — less data means faster syncs and fewer failed requests. Both Apache and IIS support compression natively with no additional software required.

See also: CSWeb Server Optimization Quick Reference

CSWeb Server Optimization Quick Reference

Applies to: IIS + PHP FastCGI + MySQL on Windows Server (WAMP) and Apache + PHP-FPM + MySQL on Linux (LAMP)

This is a quick reference for systems administrators.

MySQL (`my.ini`)

Setting	Recommended Value	Note
`innodb_buffer_pool_size`	50% of total RAM	Most impactful single setting — keeps working data in memory
`innodb_buffer_pool_instances`	`8`	Reduces contention under concurrent load
`innodb_log_buffer_size`	`64M`	Sized for one second of peak write traffic
`innodb_log_file_size`	6% of buffer pool size	e.g. 16G pool → 1G log file. Reduces log rotation frequency without excessive recovery time
`innodb_flush_log_at_trx_commit`	`2`	Flushes to OS cache per commit, to disk per second — appropriate for field data workloads
`max_connections`	`500`	Set above expected peak concurrent connections
`thread_cache_size`	`100`	Reduces thread creation overhead during connection spikes
`back_log`	`250`	Absorbs short connection queue bursts
`slow_query_log`	`1`	Keep enabled as safety net
`long_query_time`	`10`	Only logs catastrophic queries — no overhead under normal operation

Storage: SSD for the MySQL data directory is strongly recommended for large-scale deployments. HDD performance degrades significantly under high concurrent write load.

PHP (`php.ini`)

OPcache

Setting	Value
`zend_extension`	`php_opcache.dll` (Windows) / `opcache.so` (Linux — usually auto-loaded)
`opcache.enable`	`1`
`opcache.memory_consumption`	`256`
`opcache.interned_strings_buffer`	`16`
`opcache.max_accelerated_files`	`20000`

Setting	Value
opcache.validate_timestamps	0 — OPcache must be manually reset after deployments
opcache.revalidate_freq	0
opcache.enable_file_override	1
opcache.cache_id	csweb_cache — required on Windows/IIS for shared cache across worker processes

APCu

Setting	Value
extension	php_apcu.dll — obtain from pecl.php.net
apc.enabled	1
apc.shm_size	32M
apc.ttl	0 — static reference data; cache clears on IIS restart
apc.enable_cli	0
apc.serializer	php

Windows: APCu is not bundled with PHP. Download the NTS x64 DLL matching your PHP version from pecl.php.net and place it in your PHP ext directory.

Linux: Install via sudo apt install php[version]-apcu or sudo pecl install apcu.

After any code deployment: run opcache_reset() or restart the PHP process to clear stale cached bytecode.

IIS (WAMP)

FastCGI Settings

IIS Manager → FastCGI Settings → Edit PHP entry

Setting	Default	Recommended
maxInstances	4	50 — set below MySQL max_connections
instanceMaxRequests	200	10000
activityTimeout	30s	60s
requestTimeout	90s	120s
PHP_FCGI_MAX_REQUESTS (env var)	500	10000 — must match instanceMaxRequests

Application Pool Settings

IIS Manager → Application Pools → [pool] → Advanced Settings

Setting	Default	Recommended
Idle Time-out	20 min	0 (disabled)
Regular Time Interval recycling	1740 min	0 (disabled) or schedule off-peak
Maximum Worker Processes	1	1 — do not increase; APCu is not shared across worker processes

---

Apache (LAMP)

PHP-FPM Pool (/etc/php/[version]/fpm/pool.d/www.conf)

PHP-FPM manages the pool of PHP worker processes on Linux. The equivalent of IIS maxInstances is controlled by the pm (process manager) mode and its associated settings.

Setting	Default	Recommended

Setting	Default	Recommended
pm	dynamic	dynamic — leave as-is
pm.max_children	5	50 — maximum PHP worker processes; set below MySQL max_connections
pm.start_servers	2	10 — processes spawned on startup
pm.min_spare_servers	1	5 — minimum idle processes kept ready
pm.max_spare_servers	3	20 — maximum idle processes kept ready
pm.max_requests	500	10000 — requests per worker before recycling; reduces process churn
request_terminate_timeout	0	120s — terminates hung workers; align with PHP max_execution_time

After changes: restart PHP-FPM with `sudo systemctl restart php[version]-fpm`

Apache Settings (`/etc/apache2/apache2.conf` or `/etc/httpd/conf/httpd.conf`)

Setting	Default	Recommended
KeepAlive	On	On — allows connection reuse across multiple requests
KeepAliveTimeout	5s	5s — leave as-is for REST API workloads
MaxKeepAliveRequests	100	500 — increase for high-volume sync workloads

MPM Worker / Event settings (`/etc/apache2/mods-enabled/mpm_event.conf`):

Setting	Default	Recommended
MaxRequestWorkers	150	400 — maximum simultaneous connections
ThreadsPerChild	25	25 — leave as-is
ServerLimit	16	16 — leave as-is

Note: Apache with PHP-FPM should use **mpm_event** or **mpm_worker**, not **mpm_prefork**. Prefork is incompatible with PHP-FPM and significantly less efficient under concurrent load. Confirm with: `apache2ctl -V | grep MPM`

Response Compression (`mod_deflate`)

```
# Add to apache2.conf or a .conf file in conf-available/
<IfModule mod_deflate.c>
    AddOutputFilterByType DEFLATE application/json
    AddOutputFilterByType DEFLATE application/json; charset=utf-8
</IfModule>
```

Enable the module with: `sudo a2enmod deflate && sudo systemctl restart apache2`

Restart Reference

WAMP (IIS + Windows)

Change Made	Action Required
php.ini changes	Restart Application Pool
OPcache / APCu settings	Restart Application Pool
New PHP extension added	Restart Application Pool
FastCGI settings	iisreset
my.ini MySQL changes	Restart MySQL Service

LAMP (Apache + Linux)

Change Made	Action Required

Change Made	Action Required
php.ini changes	sudo systemctl restart php[version]-fpm
OPcache / APCu settings	sudo systemctl restart php[version]-fpm
PHP-FPM pool changes	sudo systemctl restart php[version]-fpm
Apache config changes	sudo systemctl restart apache2
my.cnf MySQL changes	sudo systemctl restart mysql

Post-Change Validation

PHP — confirm via `phpinfo()`:

- OPcache section present → Opcode Caching: **Enabled**
- `cache_id` shows **csweb_cache**
- APCu section present and enabled

MySQL — confirm settings took effect by running in MySQL client:

```
SHOW VARIABLES LIKE 'innodb_buffer_pool_size';
SHOW VARIABLES LIKE 'innodb_log_file_size';
SHOW VARIABLES LIKE 'innodb_flush_log_at_trx_commit';
SHOW VARIABLES LIKE 'max_connections';
```

Verify each value matches what was set in `my.ini`. If a value still shows the old default, MySQL did not pick up the config change — confirm the correct `my.ini` path is being used and restart the MySQL service.

See also: CSWeb Best Practices

Upgrading

Upgrading from Previous Versions

If you already have an older version CSWeb installed on your server and you want to upgrade to a newer version of CSWeb follow the steps below. *The upgrade script will not upgrade to another major release. For example, the upgrade script will not upgrade from CSWeb 7.x to 8.x. Instead, install CSWeb 8.x as a new installation.*

The upgrade script will not upgrade from CSWeb 8.0 to 8.1. Instead, install CSWeb 8.1 as a new installation.

Backup Existing Data

While the upgrade should normally preserve any existing data in your database, to be extra safe, you should first backup any data that has been synced to the server. Download each of the data files on the server to a CSPro DB (.csdb) file. See Accessing Data for instructions on how to download the data.

Update Project Files

Download and unzip the CSWeb source code and copy it to the csweb directory on your server. The csweb directory is where you originally copied the project files to during doing the initial installation. For example, for wampserver it would likely be **C:\wamp64\www\csweb**. The new files will overwrite the files from the original installation.

Start Apache and MySQL

Make sure Apache and MySQL are running.

Run Upgrade Script

When upgrading CSWeb, the database schema may change to support new features. In addition to updating the project files, you need to run the upgrade script to migrate the database to the new structure. Open a browser and browse to **csweb/upgrade**. For example, if you have installed on localhost then navigate to **http://localhost/csweb/upgrade**. The upgrade script will check to see if the database needs to be migrated. If it does, it will display an **Upgrade** button. Click this button to start the migration.

If everything is correct you will receive the "Upgrade Complete!" message. From here log in to CSWeb as usual.

How to ...

Accessing Data

There are two ways to access data stored on CSWeb by using the Data Manager tool that is included as part of the CSPro suite. The difference in approaches determines whether CSWeb data is fully downloaded onto your local machine.

Once you navigate to CSWeb's **Data** tab, you will see a list of dictionaries along with a **Manage in CSPro** column with two icons.

View Data

- Clicking the Data Manager icon launches Data Manager and displays the data on the server.
- This opens the data using a CSWeb data source identified using a CSPro URI.
- The CSWeb data is not fully downloaded onto your machine. Only cases that you view will be transferred.

Download Data

- Clicking the PFF icon downloads a PFF file that contains details about the dictionary.
- Running this PFF will launch Data Manager to download or synchronize the data.
- The CSWeb data will be fully downloaded onto your machine into a single CSPro DB file.

Managing Roles

Overview

Roles in CSWeb control what users can access and modify within the system. Each role defines a set of permissions across features and data. CSWeb provides several predefined roles and allows administrators to create custom roles with granular permission settings.

Understanding Permissions

CSWeb permissions are organized into the following categories:

Category	Description
Dictionaries	Controls access to data dictionaries. Read allows viewing and downloading dictionaries. Write enables creating, modifying, and deleting them.
Data	Controls access to case data within dictionaries. Read allows viewing, downloading, and viewing sync history. Write enables adding, uploading, and modifying case data. Delete permissions are split into Dashboard (via the UI) and API (via API endpoints).
Apps	Controls access to data collection applications. Read allows viewing and downloading. Write enables uploading, modifying, and deleting.

Files	Controls access to files stored in CSWeb. Read allows viewing and downloading. Write enables uploading, modifying, and deleting. Synchronizing paradata requires file permissions.
Reports	Controls access to sync and map reports. Read allows viewing reports. Write enables creating, modifying, and deleting them. Users must also have read access to the associated dictionary data to work with a report.
Users	Controls access to user management. Read allows viewing user lists and details. Write enables creating, modifying, importing, and deleting users.
Roles	Controls access to role management. Read allows viewing roles. Write enables creating, modifying, and deleting roles.
Dashboard Login	Allows users to log into the CSWeb web interface.

Default Roles

CSWeb includes three predefined roles:

Role	Description
Administrator	Full access to all features and data, including user and role management. Use for system administrators. The *Administrator* role can be used for all aspects of CSWeb work except that it cannot be used when accessing data using the CSWeb data source.
Developer	Broad access to dictionaries, data, apps, files, reports, and user management. Cannot manage roles. Use for technical staff.
Standard User	Read/write access to data and files and read-only access to dictionaries and apps. No access to the CSWeb Dashboard. Use for field data collectors.

Creating Custom Roles

Custom roles allow you to create permission sets tailored to specific user groups in your organization. To create a new role:

1. Navigate to the **Roles** page from the sidebar menu.
2. Click the **Add Role** button.
3. Enter a **Role Name**.
4. Configure permissions for each category by checking the appropriate boxes.
5. Enable **Dashboard Login** if users should be able to log into the web interface.
6. Configure **Data Permissions** (see next section).
7. Click **Save**.

Configuring Data Permissions

Data permissions offer two levels of control:

Default Permissions: Permissions apply to all existing and future dictionaries unless overridden. Set these in the **(Default)** row of the Data Permissions table.

Dictionary-Specific Permissions: To override defaults for a specific dictionary, uncheck **Use Default** for that dictionary and configure its permissions independently.

For example, a role for enumerators could have read/write access to the dictionaries for data collected in the field, but only

read-only access to the geocodes dictionary.

Managing Existing Roles

Roles can be edited, copied, or deleted from the **Roles** page using the icons next to each role. A few constraints to be aware of:

- The *Administrator* and *Standard User* roles cannot be modified or deleted.
- Copying a role opens a pre-filled form. Enter a new name and adjust permissions as needed.
- Deleting a role reassigns all affected users to the *Standard User* role.

Adding Users

Each user is composed of a username, first name, last name, role, email (optional), phone number (optional), and password. The role can be a predefined or custom role.

Adding a User

1. Navigate to the **Users** tab of CSWeb.
2. On the right-hand side click the **Add User** button.
3. Enter the username, first name, last name, role, email (optional), phone number (optional), and password.

Import Users

To import multiple users, a CSV file is needed. Each row specifies a single user which has the following format: username, first name, last name, role, password, email (optional), phone number (optional). The CSV file can include a header row. If your CSV file has a header row tick "CSV File has a header row." If an error is detected with any row, no users will be added. Duplicate users are ignored.

1. Navigate to the **Users** tab of CSWeb.
2. On the right-hand side click the **Import Users** button.
3. Browse to the CSV file.

CSV Field Rules:

- First and last name must only contain letters.
- The role is the name of the role (e.g. *Standard User*).
- The password must have at least 8 characters.
- The email and phone number can be blank.

CSV Example Rows:

```
user007, James, Bond, Developer, PasSwOrD7
user008, Pedro, Gonzales, Standard User, PasSwOrD8, pgonzales@example.com, 123-4567
```

Data Settings

You can configure dictionaries to break out questionnaire data into relational tables. Sync report is displayed only for dictionaries that are configured in the settings.

Add Configuration

Click on the "Add Configuration" button on Settings Data page to configure dictionaries.

Source Data: Dictionary in CSWeb database that is to be broken out to relational tables.

Database name: Target database to store the broken out questionnaire data in relational tables. The target database cannot be same as the CSWeb database and must be different for each dictionary.

Hostname: Hostname of the database server.

Database username: Target database username.

Database password: Target database user password.

Use Additional Configuration Options: Upload a JSON file with additional configurations.

Data Configuration Table

Displays the current settings for each dictionary that is configured.

Total Cases: Total number of cases for the dictionary in CSWeb database.

Processed Cases: Number of cases processed for the dictionary in the target database.

Last Processed Time: Time when the last case was processed by the command line process.

Modifying Configuration

Click on the edit or delete button in the data configuration table row to change the dictionary data configuration. Deleting a data configuration only removes the configuration information for the dictionary and does not delete the target database.

Processing Cases

Dictionary data is broken out into relational data by running the command line CSWeb process.

From the command line shell, run the command **php bin/console csweb:process-cases**

Make sure that you have the PHP runtime in your path and that you are running the command from the CSWeb folder.

The command line process uses the data settings configuration for each dictionary and breaks out the case data into the relational tables. The process is configured to run for five minutes each time. You can configure the process to run automatically every five minutes by running it as a cron job on UNIX operating systems or scheduling a task using Task Scheduler on Windows operating systems. The working directory should be set to CSWeb folder location when running the process as a cron job or Windows task. If running the process manually, you may have to run the process multiple times until all the cases are processed, and when new cases are added since the last processed time.

See also: Additional Configuration Options

Additional Configuration Options

Additional configuration options can be given by uploading a JSON file on the Data Configuration dialog. Currently, the Process Cases Options is the only option that can be specified. If additional configuration options are not specified, all dictionary variables are included while breaking out the case data into the relational tables.

Process Cases Options

Processing cases through the command line can be slow if you have a large dictionary with a lot of data. The solution is to use the Process Cases Options to specify only the variables in the dictionary that are needed. For example, the dictionary variables used in a report. Only these specified dictionary variables will then be broken out into a relational database when process cases command tool is run.

Include

The include options allow the dictionary variables and records to be targeted for inclusion. The entire dictionary can also be included by not specifying an include option or by specifying an empty array for the include option (see example 1). The include option is processed first.

Exclude

The exclude option allows the dictionary variables and records to be targeted for exclusion. This can be convenient if a record or the entire dictionary was included, but not all dictionary variables are needed. The exclude option is always processed after the include option.

Example 1

This specification includes the entire dictionary but excludes one record:

```
{
  "processCasesOptions":{
    "include": [],
    "exclude": ["HOUSING_REC"]
  }
}
```

Example 2

This specification includes two records and excludes three variables that contain PII:

```
{
  "processCasesOptions": {
    "include": ["PERSON_REC", "HOUSING_REC"],
    "exclude": ["HH_LAST_NAME", "HH_FIRST_NAME", "HH_ADDRESS"]
  }
}
```

Upload Files

The Files page allows a user to manage files and folders on the CSWeb server.

Files Breadcrumb

Above the uploaded files and folders there is a breadcrumb that reflects the current location on the server. The breadcrumb starts with the files folder since this is the destination for uploads. Each folder that is navigated to is reflected in the breadcrumb and is also a link that can be used to navigate to that location.

Upload Files

Files can be uploaded by pressing the *Upload Files* button on the right-hand side. A single file or multiple files can be selected, but they must be within the same folder. Once a file is uploaded, options are available to the right of each file to download,

rename, or delete the file.

Creating Folders

Folders can be created by pressing the *Create Folder* button on the right-hand. This dialog will allow the user to specify the folder name and create it. Once a folder is created, options are available to the right of the folder to rename or delete the folder. The folder itself is a link and can be clicked to navigate to that location.

Multiple Selection and Deletion

Multiple files and folders can be selected by checking the checkbox to the left of each file and folder. Alternatively, the user can check the checkbox above these to select all files and folders. Once a file is selected they can all be deleted by pressing the *Delete Files* button on the right-hand side.

Map Report

The Map Report helps visualize data. If each case has latitude and longitude data associated with it, then it can be displayed on the map as a marker. This allows coverage of a geographical area to be reviewed in a glance. Each marker displays additional information if clicked on and allows the opening of the case for viewing.

Setup Checklist

1. Upload dictionary and data
2. Import labels (optional)
3. Add a database
4. Add a configuration
5. Check Enable Map Report
6. Run CSWeb command line process

Upload Dictionary and Data

Upload the dictionary and then the data to the **Data page**. If the dictionary and data were already uploaded then this step can be skipped. However, the dictionary and data must include **latitude** and **longitude** data for each case. Otherwise, the case will not be displayed on the map.

Import Labels

By default only codes will be shown for id items. Id items appear as values for filters and optionally in the popup dialog. However, a CSV file can be imported that defines code and label pairs, so a label can be displayed in its place. The rules for defining these pairs are as follows:

1. The **first** pair in the CSV file must define the **first dictionary id**
2. Each following pair must correspond to a dictionary id in **consecutive order**
3. Dictionary ids do not require a corresponding pair. However, once a dictionary id has been **ignored no more can be defined**. The code will be displayed.

Labels are imported by clicking the **Import Report Labels** button on the **Data page**.

Add a Database

Add a new database to using the MySQL command line, MySQL workbench, or phpMyAdmin. Create a user with access to the new database. For security reasons the database user should have a corresponding password. This database is different

than the database created during the setup of CSWeb. The database will hold relational data after processing cases.

Add a Configuration

See instructions for Add Configuration. At the bottom of the **Add Configuration** dialog check the **Enable Map Report** checkbox. Then the following configuration details can be entered.

1. **Tile provider**: Allows the selection of the source of base maps (Esri, Mapbox, or local files).
2. **Access token**: Only necessary if Mapbox was selected as the tile provider. Access tokens can be managed from the access tokens page once a Mapbox account has been created.
3. **Latitude**: Specify the latitude field in the dictionary.
4. **Longitude**: Specify the longitude field in the dictionary.
5. **Map popup items (max 5)**: Customize the popup dialog for each marker with additional data from the dictionary. A maximum of five options can be selected.

Run CSWeb Command Line Process

See instructions for Processing Cases. Note that this is not a one-time operation. During data collection the command line process will need to be run repeatedly to update the data displayed on the map.

Viewing a Map Report

Click **Map Report** on the sidebar to display the map.

1. **Zoom**: Located in the top left-hand corner of the map. Press the **+** icon to zoom out and the **-** icon to zoom in. Alternatively, the mouse wheel will zoom in and out.
2. **Filter**: Located below the zoom icon. Two levels of filtering are available. They will correspond to the first two id items in the dictionary. A selected filter will filter which map markers are shown on the map.
3. **Base map**: Located in the top right-hand corner of the map. Select one of four different types (normal, hybrid, satellite, terain) of maps to display.
4. **Map popup dialog**: Click on a marker and a popup dialog will display. By default this dialog will include the case key, latitude, longitude, and a **View Case** link to open the case for viewing. Additional information from **Map popup items (max 5)** will be included if selected.

Sync Report

The Sync Report will summarize the total number of cases uploaded to CSWeb for a particular geography. The report can be run on all dictionaries uploaded to CSWeb. The dictionaries will appear in the order they were uploaded to CSWeb. By default the first dictionary will be displayed and grouped by its first dictionary id.

Viewing a Sync Report

1. Click on a **dictionary** from the sidebar under Sync Report
2. Select the **dictionary id** from the dropdown above the report

The **column filters** allow further filtering on values that are currently displayed. While **search ids** allows for a global search across all dictionary id rows (the final column, totals will not be searched).

Uploading Labels

By default only codes will be shown in the report. However, a CSV file can be imported that defines code and label pairs, so a

label can be displayed in its place. The rules for defining these pairs are as follows:

1. The **first** pair in the CSV file must define the **first dictionary id**
2. Each following pair must correspond to a dictionary id in **consecutive order**
3. Dictionary ids do not require a corresponding pair. However, once a dictionary id has been **ignored no more can be defined**. The code will be displayed.

The first row of the CSV file is either an optional header or pairs. If the header is included then tic "CSV file has a header row" when importing the CSV file.

If a header row is included and the dictionary ids for the dictionary are province, district, and enumeration area then the format of the first five rows is:

Province Code	Province Label	District Code	District Label
1	Artesia	1	Dongo
1	Artesia	2	Idfu
2	Copal	1	Baja
2	Copal	2	Bassac

Notice the enumeration area is ignored in this example. The province and district will display labels, while the enumeration area will display a code.

Understanding How Dictionary Design Affects Reports

To create useful reports it is important to understand how the design of the dictionary affects the report. Each dictionary id will become a column in the report. Typically, each dictionary id is a unique geography and the report summarizes the information by geography.

Province	District	Enumeration Area	Total
Artesia	Dongo	EA 001	56
Artesia	Idfu	EA 007	33
Copal	Baja	EA 001	0
Copal	Bassac	EA 005	74

If the enumerator id is added as a dictionary id item consider how its position affects the report.

Province	District	Enumeration Area	Enumerator	Total
Artesia	Dongo	EA 001	Enum001	56
Artesia	Idfu	EA 007	Enum002	33
Copal	Baja	EA 001	Enum003	0
Copal	Bassac	EA 005	Enum004	74

Above the enumerator id is the last dictionary id. The report summarizes the information by geography, but also allows the analyst to see which enumerator is working there. Additionally, if the province, district, or enumeration area is selected from the dropdown the enumerator will be ignored.

Enumerator	Province	District	Enumeration Area	Total
Enum001	Artesia	Dongo	EA 001	56
Enum002	Artesia	Idfu	EA 007	33
Enum003	Copal	Baja	EA 001	0
Enum004	Copal	Bassac	EA 005	74

Alternatively, if the enumerator id is the first dictionary id then the emphasis will be on the enumerator. All information is summarized by enumerator and then by geography.

Problematic Dictionary Ids

- **Urban/rural** do not make the key unique, so they do not need to be id items. However, if used they will double the number of rows in the report. Further, at what position will they be added?
- **Longitude/latitude** will likely be unique on their own. Making the report nothing more than a list.

Troubleshooting

Troubleshooting Problems

CSWeb logs errors to a **logs/api.log** file located in the CSWeb sources folder in your web server's root directory. This log file has more detailed information to help you troubleshoot problems.

If you need assistance setting up CSWeb or troubleshooting problems, please email cspro@lists.census.gov. Attach to the email any server logs to help the CSPro development team diagnose the problem.

Reconfigure CSWeb

To reconfigure or reset your CSWeb installation:

1. Locate the CSWeb sources folder in your web server's root directory.
2. Delete the file **src/config.php**.
3. Delete the contents of **var/cache** folder.
4. Open your web browser and launch the CSWeb setup page.

Code or configuration changes not taking effect

Delete the cache if you notice that your changes are not taking effect when you refresh the browser. From the CSWeb root folder the path to the cache folder is *var/cache*. Then refresh the browser.

Common errors during CSWeb setup

Failed to connect to database. SQLSTATE[HY000] [1045] Access denied for user

The database username and database password are not correct. They should be set to the username and password for a MySQL user that has permissions to access the MySQL database.

Failed to connect to database. SQLSTATE[HY000] [2002] php_network_getaddresses: getaddrinfo failed: No such host is known.

The hostname is incorrect. This should be the hostname for the MySQL database, not the server hostname. In most cases the hostname should be "localhost" unless your database and web server are running on different computers. The hostname should NOT include "http://". If MySQL is running on a different port than the default, you should add the port to the hostname. For example, "localhost:3307".

Something went terribly wrong

This is a general error that could be caused by a number of different problems with the CSWeb configuration. For more information check the CSWeb log files and also check the Apache/IIS/PHP error log.

Common errors syncing to CSWeb from a tablet/phone

Failed to connect to localhost

You are using a server URL that uses localhost. Localhost can only be used to access the server from a web browser running on the server itself. To access the server from another computer or device you must use either a domain name or the IP address of the server. If your server does not have a domain name or a static IP address you will need to obtain one. If you are unsure how to do this, you may want to consider using Dropbox synchronization instead of CSWeb.

Unable to resolve host

Either the server URL is incorrect or your device is not connected to the network.

Resource not found on server

The server URL is not correct. The host portion of the URL is probably correct but the part that follows is not. This often happens when "ui" is used instead of "api" at the end of the URL. For example, *https://example.org/csweb/ui* instead of *https://example.org/csweb/api*. Always use "api" when syncing with CSWeb from CSPro and use "ui" only when accessing CSWeb from your web browser.

Issues with CURL library when using IIS

If you run into issues with the CURL library while using IIS a workaround is to disable the curl extension in php.ini and enable allow_url_fopen.

Get Help

To contact the CSPro development team with comments, questions, or to report problems, please contact:

International Programs Center
Population Division
U.S. Census Bureau

4600 Silver Hill Road
Washington, DC 20233

Phone: +1 301-763-1451

Support email: cspro@lists.census.gov

Official website: www.census.gov/data/software/cspro.html

CSPro Users Forum: csprousers.org/forum

When you contact us, please mention that you are using **CSPro 8.1.6**.