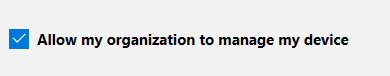
# OCG Mega Max development setup

## Machine Setup

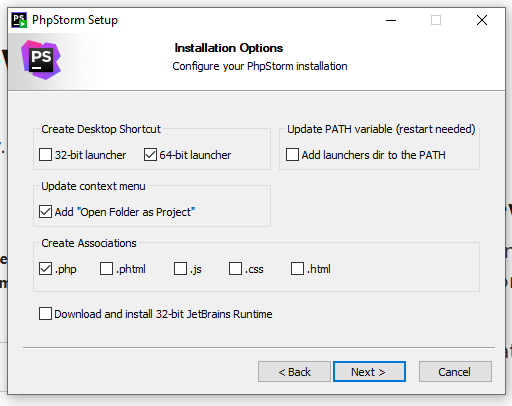
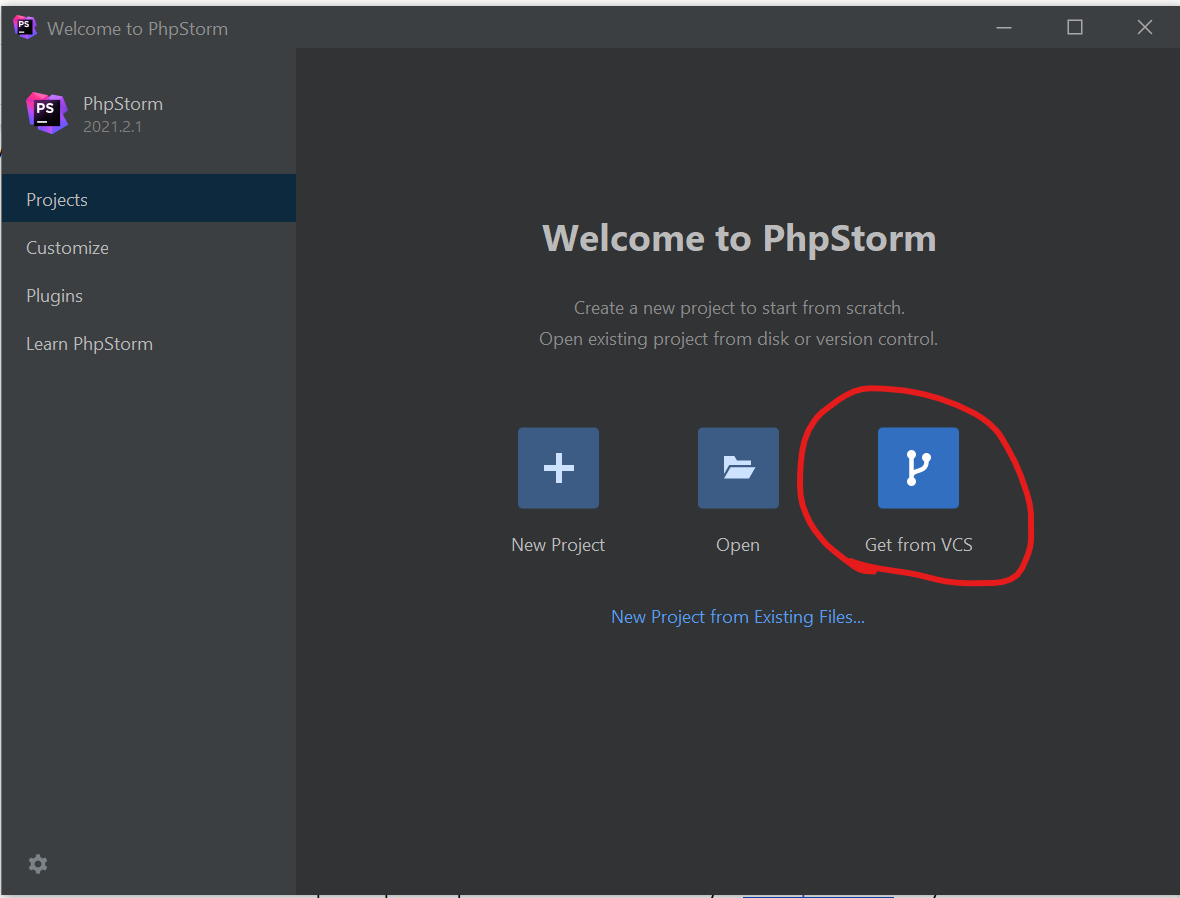
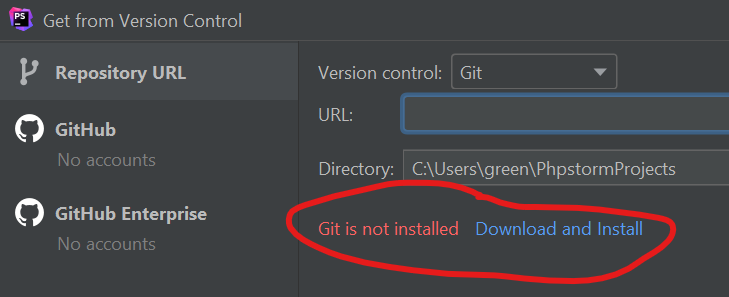
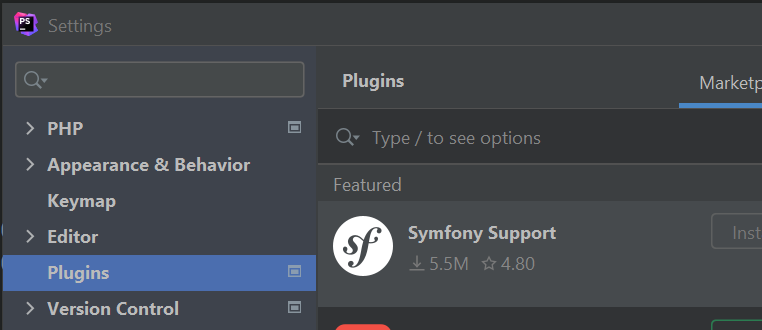
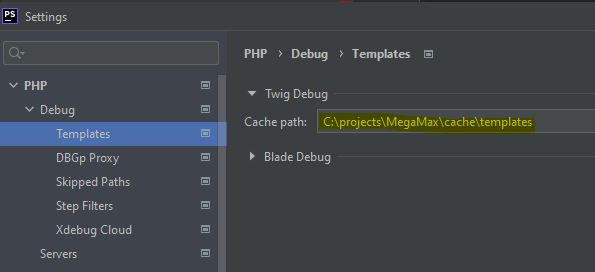
1. setup windows:
   1. Create a local account. This seems to be the easiest. Later on you can set up one drive with your OCG Account.
2. [MS Teams](https://www.microsoft.com/en-us/microsoft-teams/download-app#desktopAppDownloadregion)
   1. Un-check "Allow my organization to manage my device"  
      
3. [7-zip 64 bit](https://www.7-zip.org/download.html)
4. Ocgcompanies account.
   1. dgordhamer@ocgcompanies.com
5. Install Chrome
6. Some windows have super small fonts on the title bar?
   1. <https://superuser.com/questions/1173048/some-applications-fonts-are-too-small>
7. [Notepad++](https://notepad-plus-plus.org/downloads/)
8. [VSCode](https://code.visualstudio.com/download)
9. Install node: <https://nodejs.org/en/download/>
   1. Chocolatey is not required.

## Dev Setup

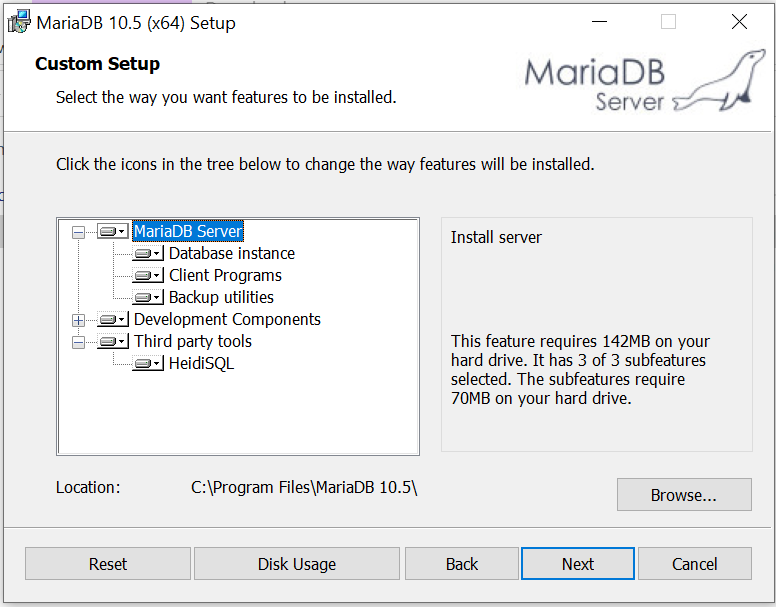
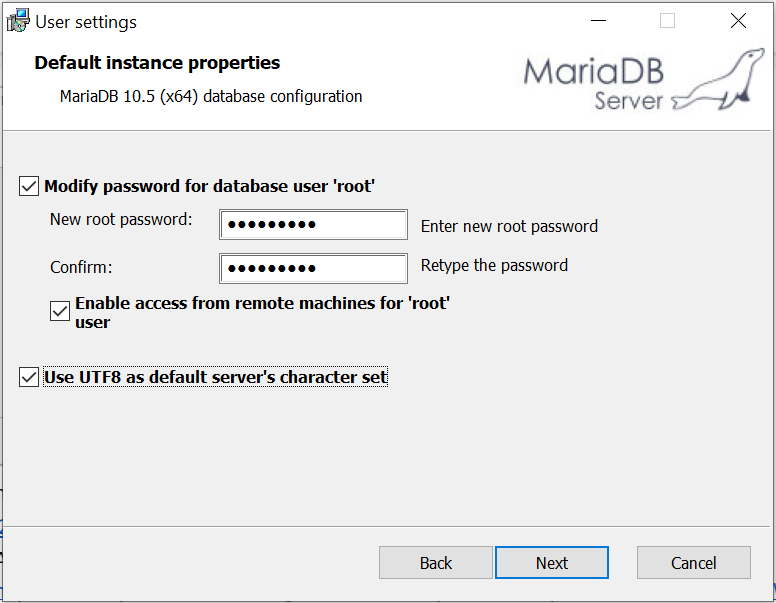
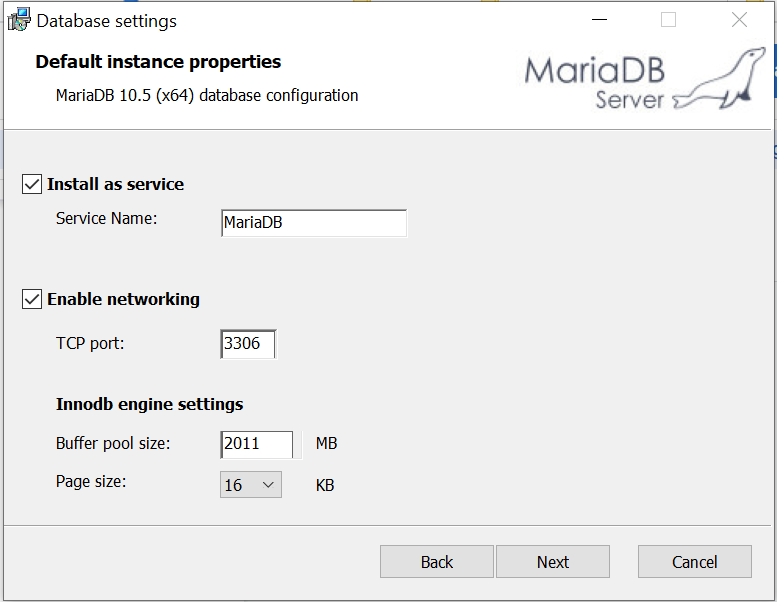
### Gitlab.com

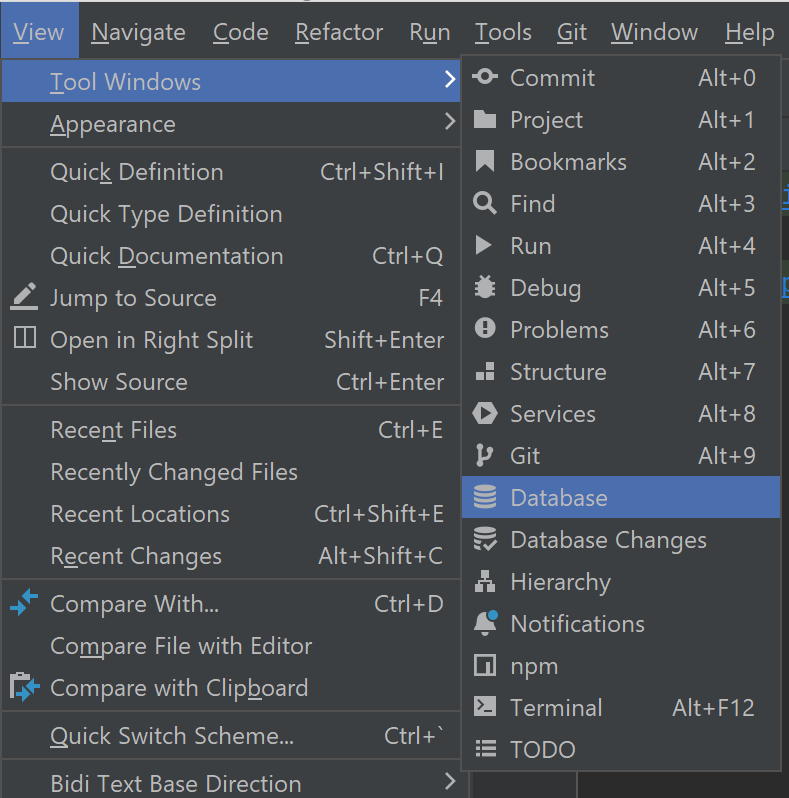
1. Create an account with ocg email.
2. Ask to be added to <https://gitlab.com/ocgcompanies>
3. You won't be able to pull or push repositories via HTTPS until you [set a password](https://gitlab.com/-/profile/password/edit) on your account
4. Create an SSH Key and add to your gitlab account.
   1. <https://gitlab.com/-/profile/keys>
   2. Use whichever type is recommended. I was unable to get ed25519 to work so I went with:
   3. ssh-keygen -t rsa -b 2048 -C JakeLaptop
5. If you need to change the key used for gitlabt:
   1. Generate a new key.
   2. Edit .ssh/config file and add an entry for the new key for the gitlab.com host. Like so:
      1. Host gitlab.com
      2. IdentityFile ~/.ssh/id\_rsa\_2048-dlg
   3. Remember you may need to update your git user.name so commits will accurately identify you.
      1. git config user.name "username1"
6. Sync to C:/vcs/MegaMax-develop/ (You can do this with PHPStorm, see next step)

### Install PHPStorm (Latest)[PHPStorm](https://www.jetbrains.com/phpstorm/download/download-thanks.html?platform=windows)

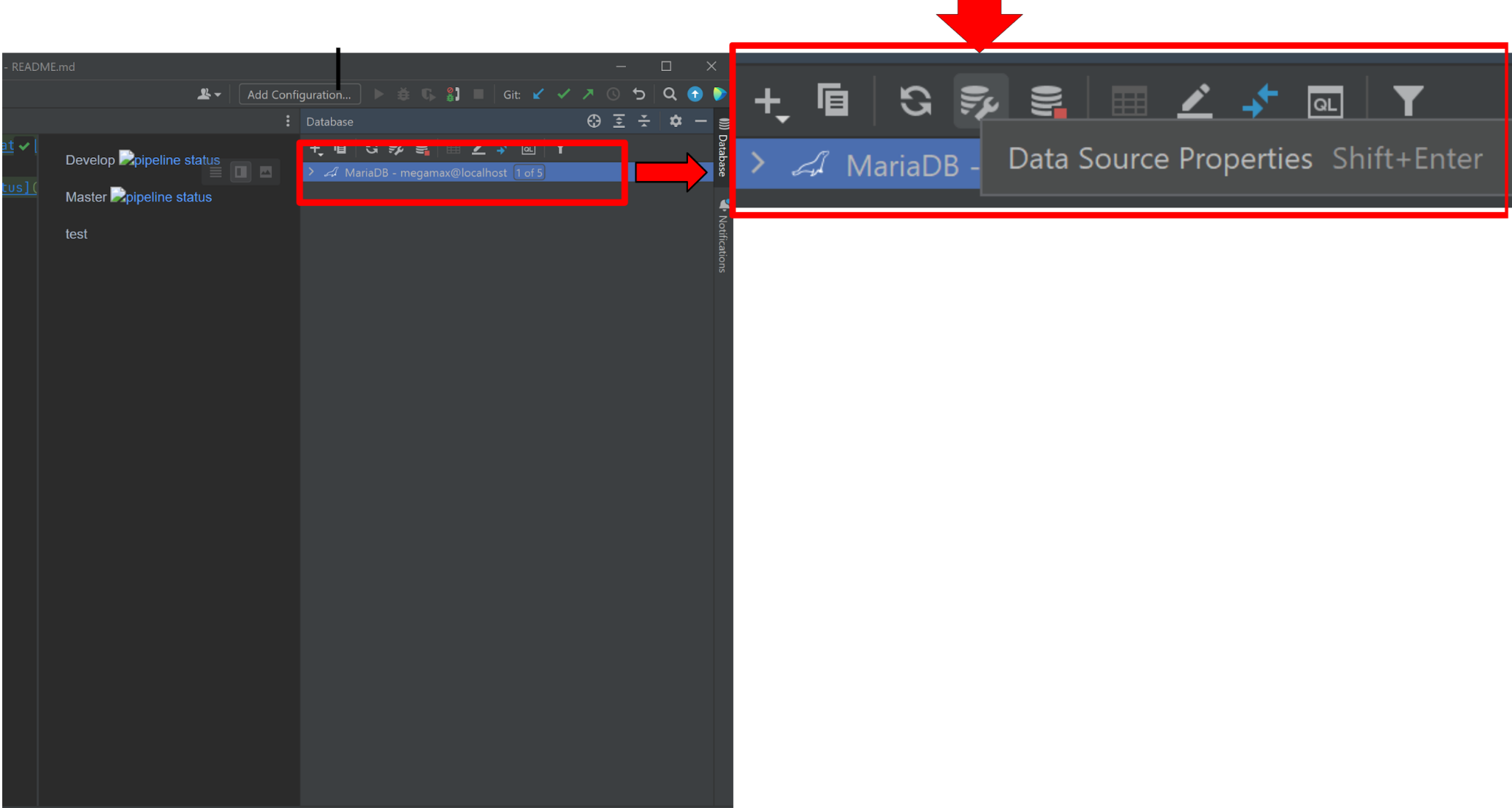
1. 
2. Install GIT:
3. 
4. 
5. Then clone <https://gitlab.com/ocgcompanies/MegaMax.git>
   1. Use your gitlab account
   2. We’ll get to configuring/building the project later.
6. Add plugin support for symphony:
7. Ctrl-alt s (settings)
8. Plugins: Symphony Support
9. Enable Twig debugging by setting the “Cache path” for twig templates:
   1. Ctrl-alt-s (settings)
   2. 

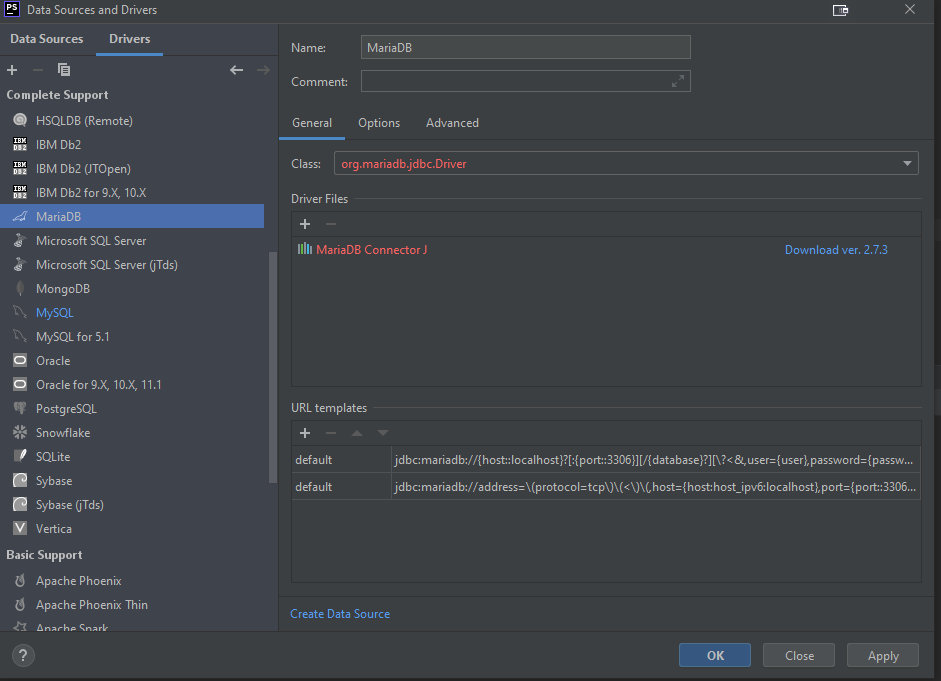
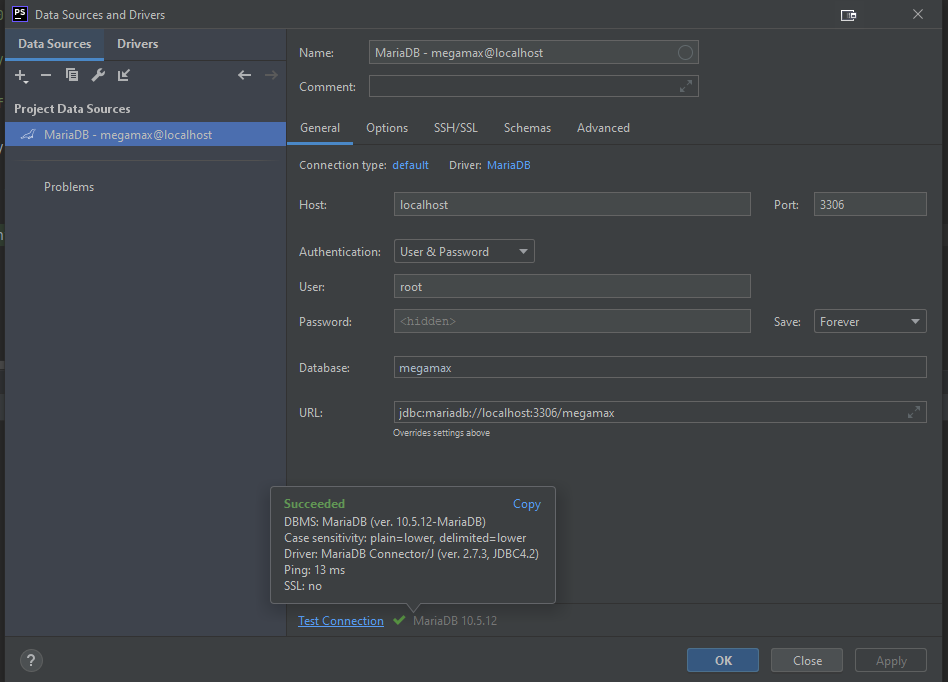
### Download/Install Mariadb

1. [Maria DB: 10.5.9](https://go.mariadb.com/download-mariadb-server-community-thankyou.html?aliId=eyJpIjoibmVLN0U1c3JhN0p3SFdLRSIsInQiOiIyc0FWakJUWHVrelR5Q1MwWDlMZU5nPT0ifQ%253D%253D) (mariadb-10.5.9-winx64.msi)
2. Or <https://mariadb.org/download/>
   1. Choose 10.5.x (as of 8/28/21)
   2. Windows, x86 64, MSI Package
3. Install MariaDB:  
   
   * 1. modify root password to “**Password1**”, use UTF8 as default, and select the “Enable access from remote machines for ‘root’ user” check box:  
        
     2. Install as a service:  
        
4. Optional: Go back and configure PHPStorm to connect to your DB:
   1. Go to the menu bar and select “View -> Tool\_windows -> Database”



In the new tile that opens, select the “Data Source Properties” (or press “Shift + Enter” while the “database” tile is active.



* + 1. Download drivers for MariaDB:
       1. Select the “Drivers” tab in the Data Sources and Drivers  
          
    2. Create new Data Source using above driver:  
       

### Download and install Apache

1. [apache 2.5](https://apache.claz.org//httpd/httpd-2.4.46.tar.gz)
2. Actually used this one from Jake: <https://www.apachehaus.com/cgi-bgin/download.plx?dli=wVFpVUlVUQ18ERntmTWhWTKVlUGR1UvVDZVFVM>
3. Unzip it to **C:/server/apache/apache-2.4.46**
4. Install httpd service so it can run at startup etc:
   1. Open up powershell in administrator mode to C:/server/apache/apache-2.4.46
   2. .\bin\httpd.exe -k install -n httpd
   3. You will get an error:   
      httpd.exe: Syntax error on line 40 of C:/server/apache/apache-2.4.46/conf/httpd.conf: ServerRoot must be a valid directory
   4. Set ServerRoot in httpd.conf: (note: use the full path like so:)  
      Define SRVROOT "C:/server/apache/apache-2.4.46"  
      ServerRoot "${SRVROOT}"
   5. Note, I renamed the directory from apps to server after the above. You need to close all services windows and run httpd.exe -k uninstall -n httpd.
   6. Then run step ii again.
5. **Next configure according to the configs in OCGConfigs project**
6. /Add Virtual
   1. :\server\**httpd-2.4.46-o111k-x64-vc15**\Apache24\conf\extra\**httpd-vhosts.conf**
      1. <VirtualHost 127.0.0.1:80>
      2. <Directory "C:/projects/MegaMax/">
      3. Options +FollowSymLinks -SymLinksIfOwnerMatch
      4. AllowOverride All
      5. Require all granted
      6. </Directory>
      7. ServerName localhost
      8. DocumentRoot "C:/projects/MegaMax/"
      9. </VirtualHost>
   2. Uncomment Include conf/extra/httpd-vhosts.conf in **httpd.conf**
   3. net start/stop httpd
   4. Or through windows services:

### php 7.4.8

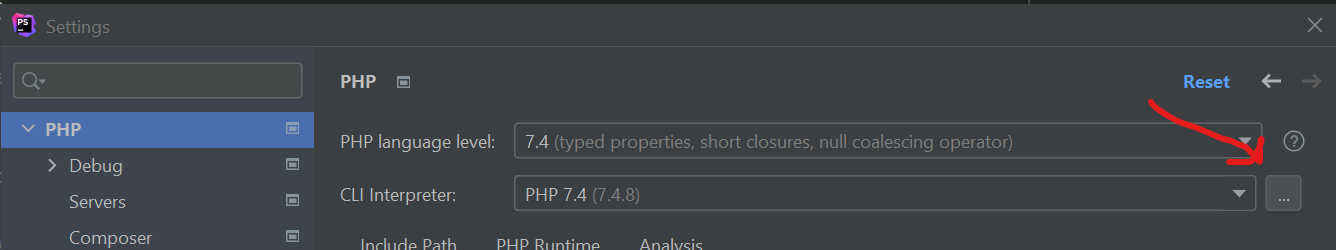
Download:

1. <https://www.php.net/downloads.php>
2. We need the Thread safe version
3. Archived version of 7.4.8: (note ...nts… is the non thread safe version meant for fastcgi which we are not using)  
   <https://windows.php.net/downloads/releases/archives/php-7.4.8-Win32-vc15-x64.zip>
4. Misc: (From the “Which version to I choose?” sidebar.)
   1. vc15: built with Visual Studio 2017 compiler
   2. vc16: built with Visual Studio 2019 compiler

Manual Install: <https://www.php.net/manual/en/install.windows.manual.php>

* + 1. Open new powershell Extract to c:\server\php-7.4.8
    2. Add **C:\server\php\php-7.4.8** to your user **path**
       1. :
          1. Type “**$env:path**” and ensure it has php directory listed.
       2. (OR) command prompt and type: **echo %path%**
       3. Open new command prompt and try it:
          1. **php -r “echo gethostname();”**
          2. It should print your hostname to the console.
    3. Copy php.ini-development to **php.ini and edit** or get latest from [OCGConfigs](https://gitlab.com/dlgocg/ocgconfigs)
    4. Create directories:
       1. **C:\server\logs\profiler**
       2. **C:\server\logs\php**
       3. **C:\server\logs\apache**
    5. Get cacert.pem from OCGConfigs and put it in **C:/server/php/php-7.4.8/cacert.pem**
       1. (I believe) Needed for curl extension

Tell PHPStorm about your php install:

* + 1. **Ctrl+Alt+S** to open settings and select **PHP**
    2. The **CLI Interpreter** might be blank. If so, add a new configuration:
       1. 

### Xdebug <https://xdebug.org/wizard>

1. Follow instructions in the wizard
2. It looks like you need to specify the full name of the DLL instead of just ~~zend\_extension = xdebug~~
3. For example **zend\_extension = php\_xdebug-3.0.4-7.4-vc15-x86\_64.dll**
4. For additional information see <https://xdebug.org/docs/install#configure-php> for more tricks.
   1. which php.ini is active?:
      1. **php -r "echo var\_dump(php\_ini\_loaded\_file(), php\_ini\_scanned\_files());"**
   2. Is Xdebug working?:
      1. **php -r "echo xdebug\_info();"**
   3. **php --version** also mentions “with Xdebug vX.X.X…”
5. PHPStorm integration:
   1. <https://www.jetbrains.com/help/phpstorm/configuring-xdebug.html#integrationWithProduct>
6. **NEVERMIND**. Don’t follow the instructions. Just download the DLL and configure your php.ini like so:

Download Link: <https://xdebug.org/files/php_xdebug-3.0.4-7.4-vc15-x86_64.dll>

Be sure to get the 64-bit version of xdebug 3.0.4-7.4 compiled with vs15

Place the DLL file in “C:\server\php\php-7.4.8\ext\” (or wherever your installation of php-7.4.8 is located).

* 1. [Xdebug]
  2. zend\_extension = php\_xdebug-3.0.4-7.4-vc15-x86\_64.dll
  3. xdebug.idekey = PHPSTORM
  4. xdebug.remote\_enable = 1
  5. xdebug.profiler\_output\_dir=C:\server\logs\profiler
  6. ; not sure if these are needed:
  7. xdebug.mode=debug
  8. xdebug.client\_host=127.0.0.1
  9. xdebug.client\_port=9003

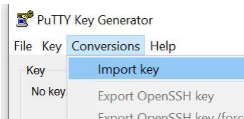
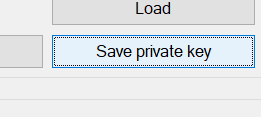
### Replication: (**DRAFT**)

Add this line to your hosts file: (“C:\Windows\system32\drivers\etc\hosts”)

* + 1. 3.133.144.78 ocg-replication

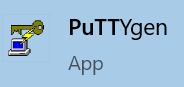
### Generate a public/private key pair:

### **Option 1: Using Command Prompt with the ssh-keygen command**

* + 1. Command prompt: **ssh-keygen**
    2. No need to enter a password
    3. Save the file as C:\Users\green/.ssh/id\_rsa\_[yourname]
    4. Send the .pub file to Jake. Ask him to work his replication magic.
    5. Convert private key to ppk using PuTTYgen:
       1. 
       2. 
    6. Test connection with Putty:

### **Option 2: Generate keys using PuTTYgen application**

### Generate ppk using puTTY:

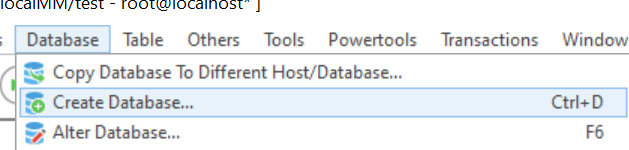
* + 1. Install putty: <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>
    2. Use puttygen  and generate.
    3. Save the public and private key.

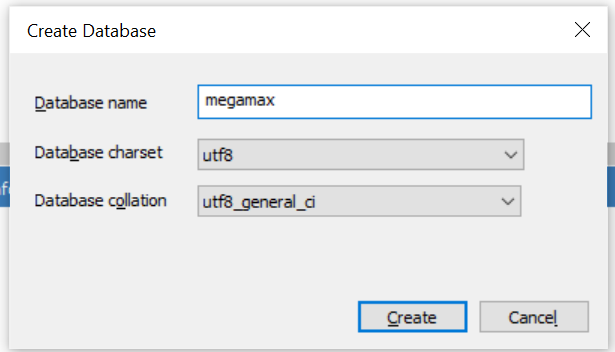
### [SQLYog](https://static.webyog.com/downloads/SQLyog-13.1.7-0.x64Trial.exe)

Install SQLYog. <https://webyog.com/product/sqlyog/trial/>

* When setting up a connection, use the username and password you set for the MariaDB server. This should be connecting to the “localhost” address (which is the same IP Address as “127.0.0.1”)

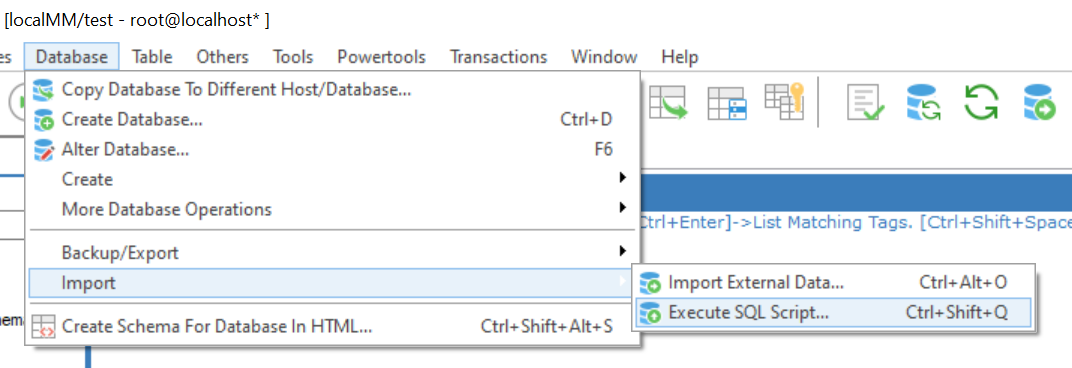
Create DB (Ctrl-D)





* Database name: "megamax"
* Database charset: **utf8**
* Database collation: **utf8\_general\_ci**

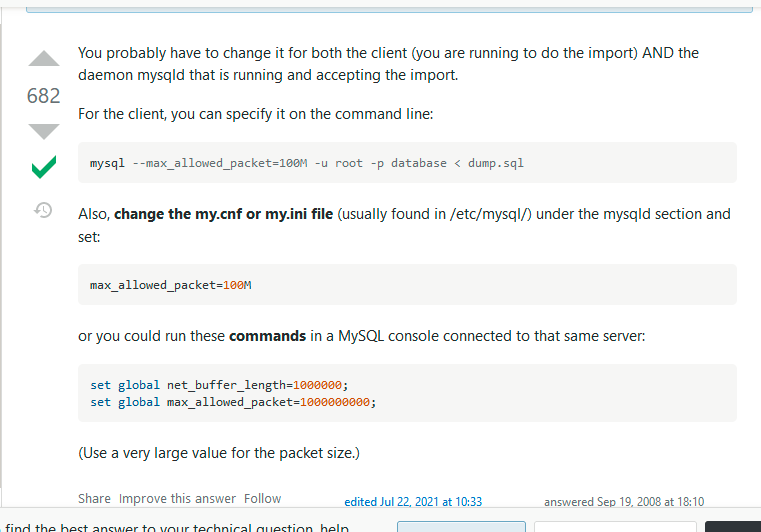
Import DB into the **megamax** database. Be sure to **select** the megamax database in SQLYog **before** importing the .sql file. (5 min)

Menu: Database -> Import -> Execute SQL Script…  


Use the .sql file from a MySQL Dump (on sharepoint)

Sometimes you may run into an error. Update the my.ini file which is in the same location as your database files. On my machine that’s c:\Program Files\MariaDB 10.5\data\**my.ini**

I needed to update my max\_allowed\_packet to 100M. See [this stackoverflow](https://stackoverflow.com/questions/93128/mysql-error-1153-got-a-packet-bigger-than-max-allowed-packet-bytes). (Screenshot Below)



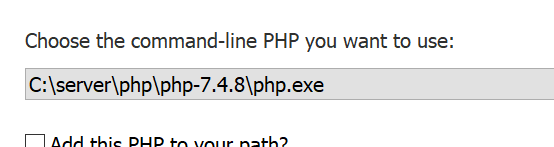
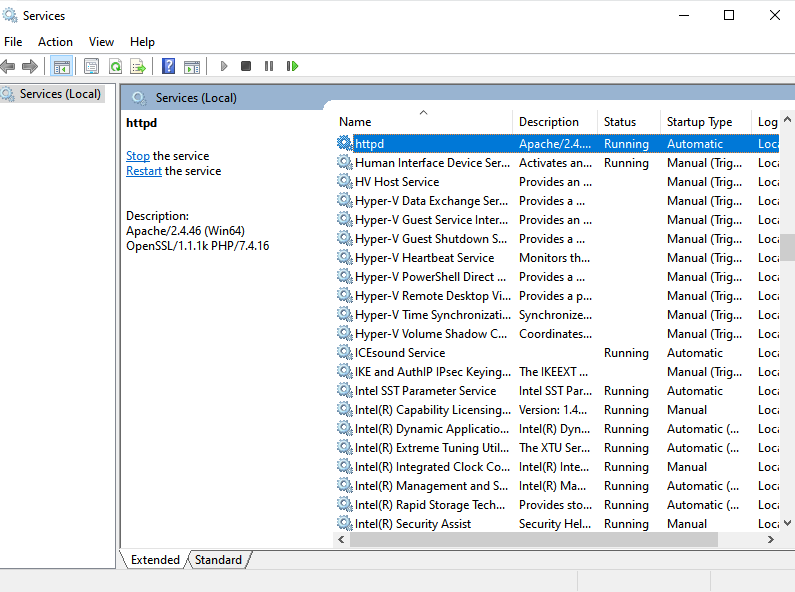
To use the above solution, you will need to open the command prompt, navigate to you installation of MariaDB, go to the “\bin” directory and run “mysql.exe - - user [USER] -p” then enter the password you setup when installing MariaDB as prompted.

**(in this case the user is “root” and the password is “Password1”)**

### [MySql Workbench](https://dev.mysql.com/downloads/workbench/)

1. [Visual C++ 2019 Redistributable](https://support.microsoft.com/en-us/topic/the-latest-supported-visual-c-downloads-2647da03-1eea-4433-9aff-95f26a218cc0)
2. Goofy [Workaroung](https://dba.stackexchange.com/questions/200130/mysql-workbench-crashing-when-connecting) to connect
3. Abandoned. Maybe try this again later?

## Local MegaMax project setup

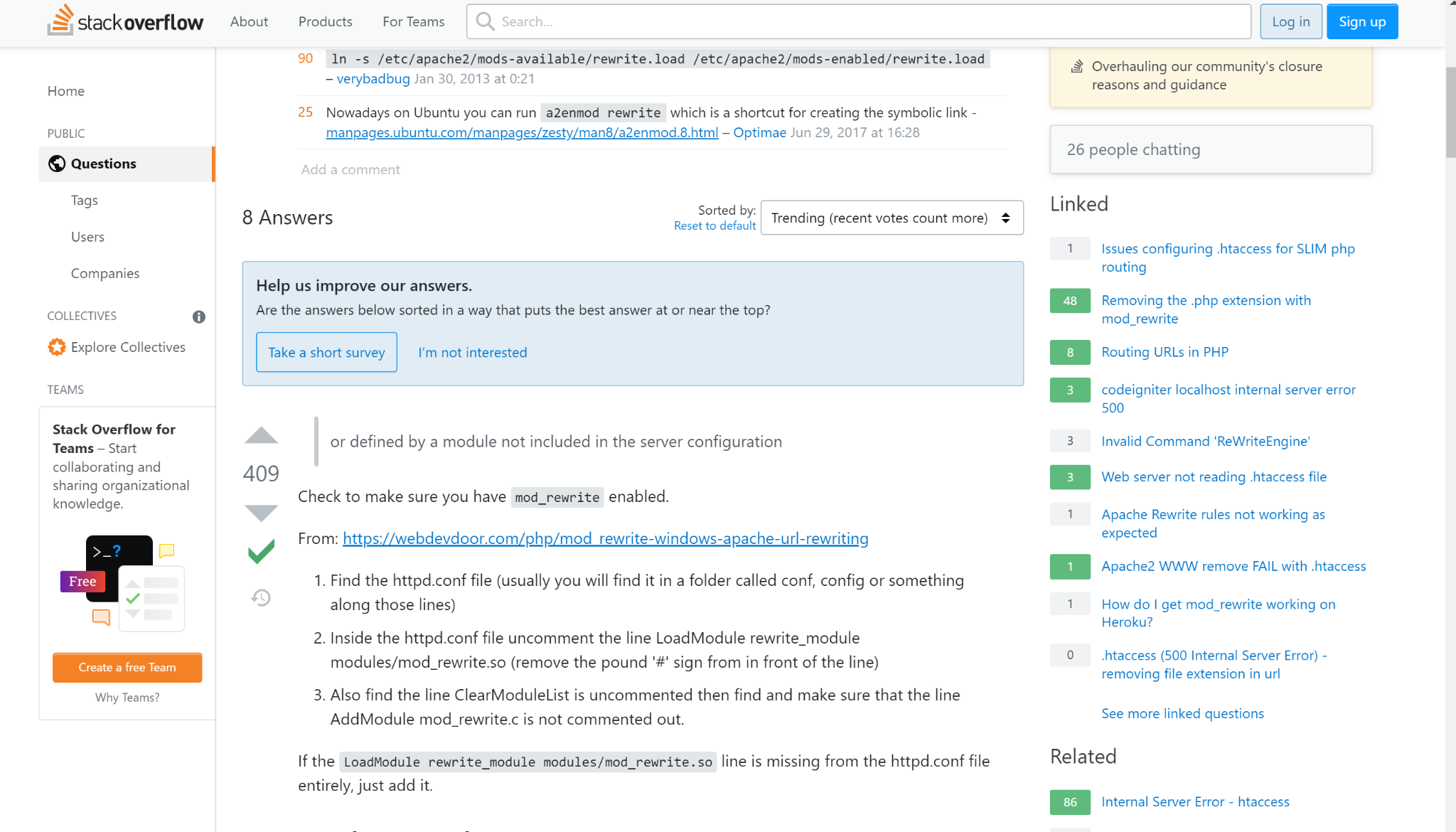
1. Install [composer](https://getcomposer.org/download/)
   1. Windows installer
   2. 
   3. PHP version 7.4.8
   4. C:\server\php\php-7.4.8\php.exe
   5. Proxy: none
   6. Add to System path:
   7. C:\server\php\php-7.4.8
2. Redis on windows
   1. [~~https://redis.com/ebook/appendix-a/a-3-installing-on-windows/a-3-2-installing-redis-on-window/~~](https://redis.com/ebook/appendix-a/a-3-installing-on-windows/a-3-2-installing-redis-on-window/)
   2. ~~Extract to C:\server\redis-2.4.5~~
   3. ~~Get redis from OCGConfigs project~~
   4. ~~Follow instructions in~~ **~~Redis on Windows Release Notes.docx~~**
   5. <https://kasunprageethdissanayake.medium.com/installing-redis-x64-3-2-100-on-windows-and-running-redis-server-94db3a98ae3d>
3. Once PHP and HTTPD (apache) are setup and working, you can run php commands for megamax from the root of the project: For example: from C:/vcs/MegaMax-develop/ you can run:
   * 1. 
4. Setup **megamax.in**i:
   1. One folder above c:\projects\MegaMax (your project folder synced from git) create “configs” folder. Add megamax to that from the OCGConfigs project
   2. This is a hardcoded location, in some places, AND configured in **config.php** so make sure the location is updated there too. We’ll get to that.
   3. webRoot used for links etc.
   4. documentRoot = set to your project location.
   5. Database params set as needed.
   6. **C:\Windows\System32\drivers\etc\hosts** file: add:
      1. 127.0.0.1 megamax.lan
5. C:\projects\MegaMax\inc\**config.php** 
   1. Add your hostname to JWHITE case. To get your hostname from PHP’s perspective run from command prompt:
      1. php -r "echo gethostname();"
6. Run commands: from terminal in the root of the project dir:
   1. Install PHP packages
      1. **composer install**
      2. As of 8/31/2021, 127 installs
   2. Install javascript components:
      1. **npm install**
      2. 1700 + packages installed
7. You are all set. Next steps:
8. Load the login page <http://megamax.lan/login.php>
   1. Missing css and images?
      1. Uncomment all the **combine** and **copy** lines in webpack.mix.js
      2. Run “**npm run dev**”
      3. Revert that file. Otherwise build will take a very long time each time.
   2. Now log in:
      1. <http://megamax.lan/>
      2. Forwards to <http://megamax.lan/login.php>
9. Log Locations:

Apache:

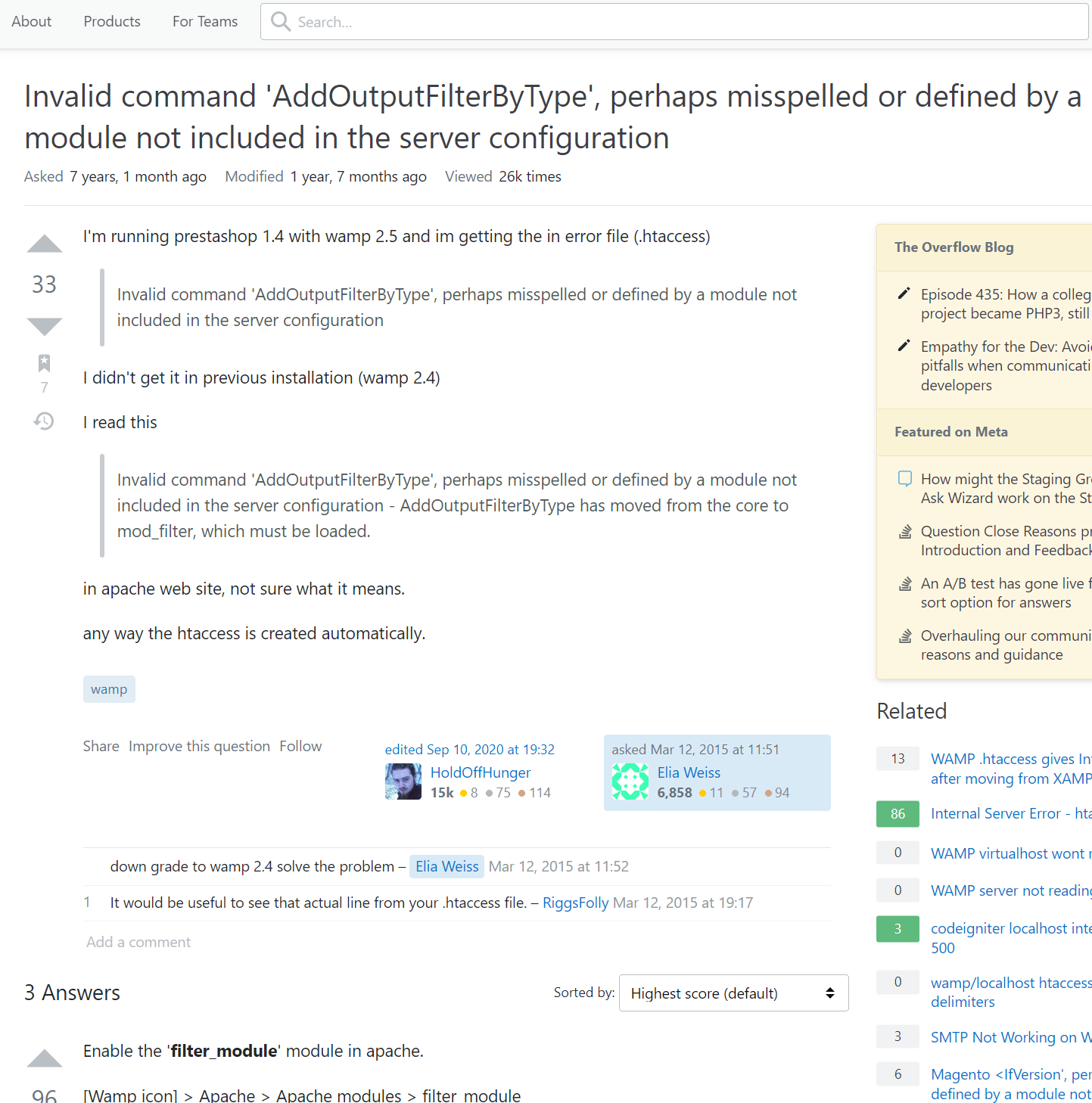
* C:\server\apache\apache-2.4.53\logs

Errors I ran into.

# Invalid command 'RewriteOptions', perhaps misspelled or defined by a module not included in the server configuration #

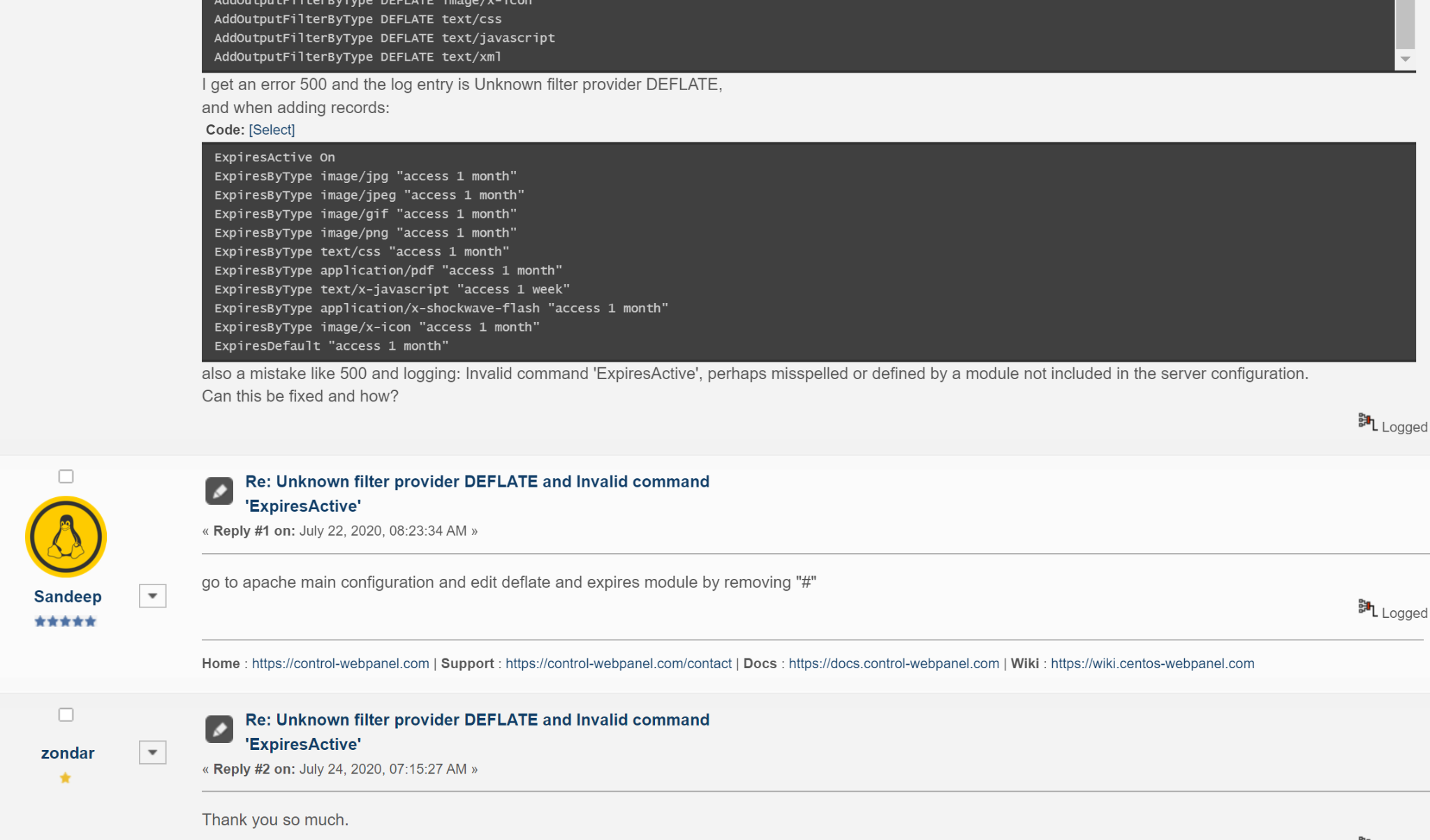


# Invalid Command AddOutputFilterByType



To solve this, Enable the ‘filter\_module’ in httpd.conf.

# Unknown filter provider DEFLATE



To solve this error, enable the deflate module in the httpd.conf file.

1. Create a user
   1. via sql: (doesn’t seem to be working at the time this was written.)
      1. ~~insert into users (name, user\_name, email) VALUES (‘david’, ‘dgordhamer’, ‘~~[~~dgordhamer@ocg.com~~](mailto:dgordhamer@ocg.com)~~’)~~
   2. Update the password for OPeer user (Run from inside the MegaMax Project folder):
      1. ./megamax user:set-password OPeer Password1
   3. Log in ( <http://megamax.lan/login.php> ) with [Owen.peer42@gmail.com](mailto:Owen.peer42@gmail.com) and Password1
   4. Create your user: <http://megamax.lan/views/office/user-management.php>
   5. Add user\_groups to your user: (replace 271 with your new user “ID”. Note, this copies user\_group\_assignment from Jake’s super user.
      1. INSERT INTO user\_group\_assignment
      2. SELECT NULL, **271**, a.uga\_group\_id
      3. FROM user\_group\_assignment a
      4. LEFT OUTER JOIN user\_group\_assignment b ON b.uga\_user\_id = **271** AND a.uga\_group\_id = b.uga\_group\_id
      5. WHERE a.uga\_user\_id = 68 AND b.uga\_id IS NULL;
   6. Log in with your new user.

## Java Setup

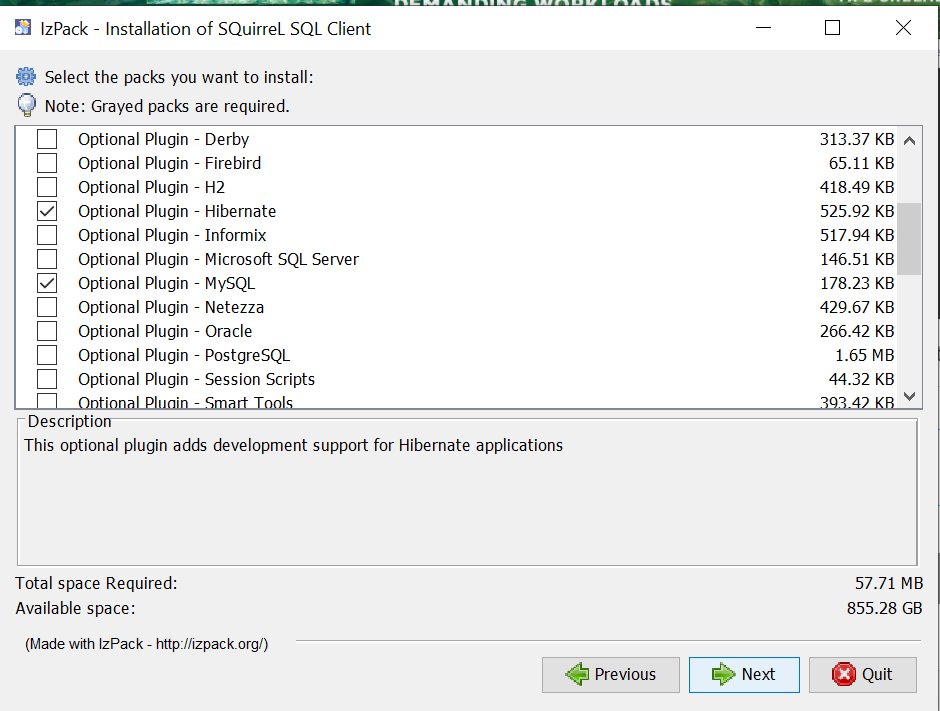
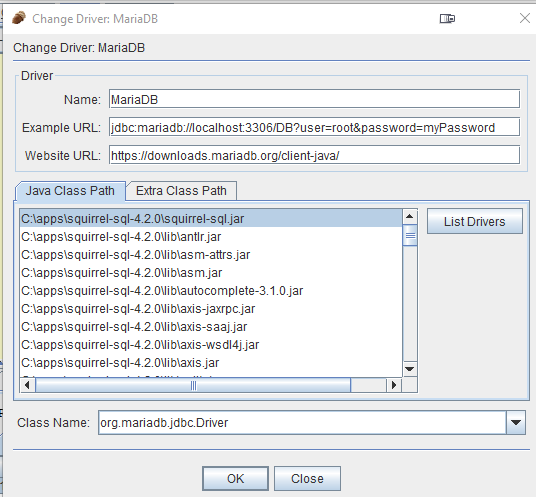
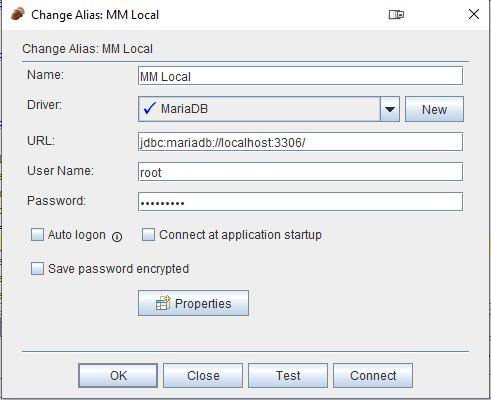
1. Install Oracle’s JDK:
2. Install OpenJDK 9:
   1. <https://jdk.java.net/archive/>
   2. 7 zip to unzip to a tar file
   3. 7 zip to untar the tar to a directory
   4. Rename jdk-9.0.4 to openjdk…
   5. Move to c:/Java
   6. **(OPTIONAL)** Add “C:/Java/openjdk” to JAVA\_HOME (you may need to create this as a new User Variable) if you want this to be the default java installation on your machine.
   7. **(OPTIONAL)** Add “C:\Java\openjdk\bin” to your users “path” variable.

## Squirrel Setup:

1. <http://squirrel-sql.sourceforge.net/#installation>

<https://sourceforge.net/projects/squirrel-sql/files/1-stable/4.2.0/> (use this link to download version4.2.0 which is the last version of squirrel to support java 8+.

When installing from the command line using “java -jar [squirrel install jar file name]”, you will need to run CMD with administrator privileges.

1. 
2. Install MariaDB Connector/J (Java 8 Connector)
   1. <https://mariadb.com/downloads/connectors/>
   2. Example url: jdbc:mariadb://localhost:3306/MegaMax
   3. Create driver sonfig:
      1. 
   4. Create an alias:
      1. 

## Other dns hosts file entries:

127.0.0.1 megamax.lan

3.133.144.78 ocg-replication

3.133.144.78 ocg-replication.local

3.133.144.78 ocg-prod-rep.tunnel

13.59.219.77 ocg-services

13.59.219.77 oauth.ocgcompanies.com

13.59.219.77 sabre.ocgcompanies.com

13.59.219.77 satis.ocgcompanies.com

13.59.219.77 support.ocgcompanies.com

## Tips and tricks:

### Power shell

1. **ctrl-r** reverse lookup of command history. Keep hitting it to see more results
2. echo $Env:Path

### PHP

1. Echo stuff to the console:
   1. php -r "echo 'path:'.session\_save\_path() ;"
   2. php -r "echo \_\_DIR\_\_;"
   3. php -r "echo xdebug\_info();"
   4. php -r "echo var\_dump(php\_ini\_loaded\_file(), php\_ini\_scanned\_files());"
2. Alternative control structures: [Use colon along with endstructure instead of curly braces](https://sodocumentation.net/php/topic/1199/alternative-syntax-for-control-structures). This is helpful with really long code segments inside a for loop.

### Megamax project:

1. Set a users password: **./megamax user:set-password users.user\_name Password1**
   1. Note, users.user\_name refers to the DB *table*: users and *column*: user\_name

### PuTTY

1. Export all settings:
   1. regedit /e "c:\temp\putty.reg" HKEY\_CURRENT\_USER\Software\SimonTatham\PuTTY
2. Export just the settings:
   1. regedit /e "c:\temp\putty-sessions.reg" HKEY\_CURRENT\_USER\Software\SimonTatham\PuTTY\Sessions

### Useful server info: (TODO: Incomplete)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Staging |  |  |
|  |  |  |  |
|  |  |  |  |

1. Environments:
   1. Production:
      1. Web Server: ocgmegamax.com
         1. Apache:
            1. logs:
      2. DB: ???
      3. Replication server:
         1. Host:3.133.144.78
         2. Hosts the replication DB instances for Prod, Staging, Sandbox. (Different DB Names in the same instance of MariaDB)
         3. Hosts oauth and sabre (Jake to clarify what this means)
   2. Stage/sandbox:
      1. Web and DB Server:: staging.ocgmegamax.com / sandbox.ocgcompanies.com
2. Configs:
   1. /etc/**php.ini**
   2. /etc/httpd/conf/**httpd.conf**
3. Logs: (note: ocgmegamax\_sandbox is also a directory that houses sandbox)
   1. /var/www/ocgmegamax/public\_html/logs
   2. /var/log/httpd/ocgmegamax/ssl-error.log
4. User info:
   1. Need sudo to access the above logs. Add your user to the “wheel” group which allows sudo.

### Useful linux command:

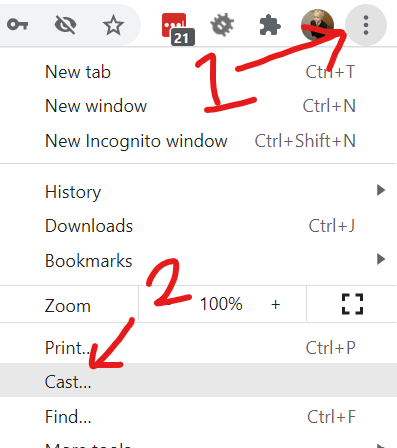
Disk usage: du -a / | sort -n -r | head -n 5

start mariadb: systemctl stop mariadb

### Misc info:

1. 13.59.219.77 ocg-services
   1. This server hosts:
      1. oauth.ocgcompanies.com
      2. sabre.ocgcompanies.com
         1. sabre is used by billing. oauth is used by sabre
      3. satis.ocgcompanies.com
         1. Used by composer
         2. Like our own personal Packagist (<https://packagist.org/>)
      4. support.ocgcompanies.com
         1. Used by PA’s (Christine, Shaq, Alex)
      5. GitLab runner is hosted on this server (<https://docs.gitlab.com/runner/>)

### Office stuff:

1. Connect to wireless display:
   1. Chrome:
      1. 
      2. Then choose your source (desktop/tab/file) and click on the chromecast device.
   2. Windows:
      1. Win+k
      2. explorer.exe ms-settings-connectabledevices:devicediscovery
      3. Then choose your display.

Project Improvements:

1. Documented Deployment process and requirements.
   1. (Maybe have BA deploy after testing in staging.ocgmegamax.com
2. Development Environment setup not documented, and may not be repeatable.
   1. Hire interns that need to have machines set up.

Project overview:

**Npm run dev**

Needed to build js and react/vue dependencies as well as setup some dependencies.

Classes: old non name spaces classes.. (PHP Namespace)

Command: ./megamax commands

Classes-ns

Newer stuff.

AllEncompassingProvider. Kind of like a bootstrap

views/template is the base dir for templates.

Can pass data into the template

Twig is the templating engine from symfony framework.

PSR standards php-fig.org

Container is a facade

Logger facade

Cake php phinx robmorgan/phinx for DB Migrations (Changes) Like Liquibase.

./megamax make:service-provider

Many Line items are specific to DTE

Some interesting paths:

* view/field/payments-lists/6
* vendor/items/
* vendor/items/prices

## Megamax KT cafe (Knowledge Transfer/sharing):

### Database stuff:

Tables: All columns have a representation of the table name in the column name. For example, job table has job\_scheduled\_start\_date, or job\_info\_date. It’s a little redundant, but helps keep track of which field is for which table while joining many tables. I guess the original designers didn’t know about aliases. :)

Question:

How do I refresh my local DB with Prod?

Answer:

Through SQL Yog. Make sure you're connected to both replication and localhost. Be on the replication connection. On the right-side menu with all of the tables, expand the database you want to copy, right click the word "tables", select "copy tables to different host". It's important to be sure you have target host of "localhost" selected. Then select the database you want those tables to be copied to.

Question:

I have to run npm run dev each time I make a change to the vue file. For example, making a change to station-list.vue just to play around...Is there a faster way?

Answer:

run 'npm run watch'

it will detect changes to files when they're saved and automatically rebuild and it will complete almost immediately

### Jobs

MM Jobs status:

“Complete”: So, there really is no status called complete. “Back from field” is the status. What they mean when they say a job is complete is the work is done and it’s ready to be billed etc.

### Routes and FrontController.php

Example: **/api/setup/station** GET call is defined in **routes/setup.php**. It calls the **StationController**->**getAll(**) method.

### Common errors:

Return value of OCG\MegaMax\Controller\FrontController::routeRequest() must be an instance of Laminas\Diactoros\Response, string returned

Cause: report-dashboard.php

### Development Guide:

#### Clients, Services, DAOs:

A client is a consumer that needs to access data from a service. A client should only ever retrieve data from a service. They should never go directly to a DAO, much less use a DB connection directly to get data. This is for several reasons:

1. Separation of responsibilities. A client’s main job and focus should be presenting the data to the users in a usable way. It’s concern is usability
2. Security. //TODO: splain why security is impacted with this pattern.
3. Nuff said I guess.

Services sit between pages and the underlying data. Services should perform and enforce business logic rules.

DAOs site between services and the underlying data storage. In our case, this is the Megamax Database.

### API in PHP/Megamax

Example:

### Deployments

#### Sandbox:

To to a full deployment of all assets (this is skipped because it takes a long time and isn’t needed every time) uncomment all the .combine lines and mix.copy lines in **webpack.mix.js**

## PHP quirks:

### String formatting

* Concatenation is like this: $message = “Wrong inputs. input 1:”.$input1.” input 2:”.$input2; // to easy to mess up.
* Interpolation is like this: $message = “Wrong inputs. input 1: {$input1} input 2:{$input2}; // pretty clear
* Sprintf is like : $message = “Wrong inputs. input 1: {$input1} input 2:{$input2};

### Namespacing:

* Ever see a backslash before a variable? \ (backslash) is the namespace separator in PHP 5.3. Used in the beginning of a variable/function is the Global namespace. See [this StackOverflow](https://stackoverflow.com/questions/4790020/what-does-a-backslash-do-in-php-5-3#tab-top).

### Params:

* You can pass in a null, but you need to give a type hint of “?”. For example: function stuff(**?**string $var1) {...

### Logic:

* How equals comparison works: [How do the PHP equality (== double equals) and identity (=== triple equals) comparison operators differ? - Stack Overflow](https://stackoverflow.com/questions/80646/how-do-the-php-equality-double-equals-and-identity-triple-equals-comp)

## GIT common commands

* List all tags: (filter down to 4.3.1\*)
  + git tag -l ‘4.3.1\*’
* Get the status of my local branch
  + git status
* Get the latest:
  + git fetch (just check to see if updates are available)
  + git pull (check for updates and pulls them down)
* git branch

### SQL KT

#### COALESCE: returns the first non-null value in a list.

#### INNER: default kind of join when you don’t specify LEFT or RIGHT

#### JOIN: left and right joins are the “base” table. Or, the greedy table. So, if you LEFT JOIN, the greedy table will be the table on the left of the JOIN statement (the FROM table). Example: SELECT \* FROM job LEFT JOIN address will give you all jobs even if they don’t have a related address table.

#### Date comparisons: job\_create\_date > (CURRENT\_DATE - INTERVAL 55 DAY)

## AWS

### Network restriction by IP to Megamax EC2 (Ohio Region)

We use [security groups](https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#SecurityGroups:) to whitelist specific ip’s into the boxes