1. System requirements
2. Clone the website project from github
3. Install Drupal in your localhost
4. Troubleshooting
5. Prerequisite software

# Localhost installation

You’ll need Apache, php and MySQL running on your computer system.

XAMPP - multiplatform

<https://www.apachefriends.org/index.html>

WAMP – Windows

<http://www.wampserver.com/en/>

MAMP – Macintosh

<http://www.mamp.info/en/>

# Git installation

This is the link to the Git website but you could also look at Github for more info.

<http://git-scm.com/>

## There are lots of tutorials available.

Maybe you will find one of these helpful :

<https://www.atlassian.com/git/>

<https://www.youtube.com/playlist?list=PLCOe2mg1ZXrnb8vlMfFMBnpItL4rho-3L>

In depth commentary on submodules and somewhat-long videos on the subject

<http://randyfay.com/content/drupal-deployment-git-submodules>

# Drush installation

Scroll down to see the links in the readme file

<https://github.com/drush-ops/drush>

Windows Install guide

<https://www.drupal.org/node/594744>

Windows Install guide configuration

<https://www.drupal.org/node/1843176>

Godaddy install

<https://support.godaddy.com/help/article/12396/how-to-install-drush-on-cpanel-shared-hosting>

1. Cloning the repository

# Github

<https://github.com/yserviceclub/website>

### The instructions in this paper are for cloning the master branch called ‘website’ using the git command line.

Alternatively, download the zip file from github and place in the correct directory for your system.

Clone the repository into its own folder and make the master branch active

git clone --branch website https://github.com/yserviceclub/website.git

### Git now has a ‘remote’ called ‘origin/website’ for updating your local repository.

### The branch is up to date with its origin and if required you can select a specific version by using tags.

The repository “HEAD” will go to that commit.

View available tags (you may see lots of tags pertaining to Drupal, look for the prefix letter ‘ v ’.

git tag

Show annotations (comments) associated with a specific tag called v0.2

git show v0.2

Set your branch to the revision v0.2

git checkout v0.2

Because Drupal modules were cloned as submodules they retain their remotes for updating through git.

### Whenever you clone or pull for updates, the submodules must be explicitly updated.

This command should update any modules found in sites/all/modules without actually going into each directory and telling git to update it. I can’t say if the –init option is important beyond the initial installation but if any new modules were introduced then it may be required.

git submodule update –init –recursive

### You now have a clone of the repository. Use these git commands for whatever purpose.

git status

git log

git log -1 HEAD

### Updating the repository

You can simply pull and checkout if needed but don’t forget the submodules

git pull

git checkout v0.3

git submodule update –init –recursive

1. Drupal Installation
2. Create database
3. Install.php settings
4. Configure the file system
5. Enable Backup\_and\_Migrate module
6. Restore the latest database backup

### You will need to create a database using your local server software.

phpMyAdmin will let you create a database.

* No prefix needed
* Use whatever name and password you like

### Go to your localhost directory for this project using your favorite browser.

Drupal’s install.php will launch. Just follow instructions.

Localhost/website

#### Your configurations won’t have any effect on other installations because the settings.php file is ignored by the git repository .

### Set the private files path which is required for the backup module

Admin menu » configuration » media » file system

sites/default/private

### Now that the site is working you still need to enable the Backup\_and\_Migrate module

Scroll past the core modules or use your browser’s find text feature.

Don’t forget to save.

Admin menu » modules

### Finally, go to the backup module and restore the latest database

Admin menu » configuration » system » Backup\_and\_Migrate

### Select the RESTORE tab and Restore from a saved backup

Choose the file you want and leave settings at default should restore the database

#### Now all the modules are enabled and configured. All the menus, blocks and views should look as intended.

1. Troubleshooting

Especially after an update, try to clear cache then try restoring the database again.

If still not working, try pulling again from the remote, checkout a tag or branch, update submodules, clear cache, restore database, clear cache again.

### Clear cache using Drush

Drush cc all

### Without Drush you need to consider Drupal cache and Views cache

(there may be other caches, I really recommend using Drush)

Admin menu » configuration » development » performance

Admin menu » structure » views » [ settings tab ] » [ advanced tab ]