Drupal Overall –

Name

Slogan

Requirements to install:

[module filter](http://drupal.org/project/module_filter) module https://www.drupal.org/project/module\_filter

Ubuntu package ftpd

[ctools](https://drupal.org/project/ctools) module

[sweaver](http://drupal.org/project/sweaver) module

CKEditor module [CKEditor](http://drupal.org/project/ckeditor)module (a WYSIWYG editor)

Calendar module

Youtube module

Flex slider module <https://www.drupal.org/project/flexslider>

Superfish module

Entity module <https://www.drupal.org/project/entity> included in drupal 8 core

Entity reference <https://www.drupal.org/project/entityreference> included in drupal 8 core

View module <https://www.drupal.org/project/views>

Advanced help module https://www.drupal.org/project/advanced\_help

Event management and calendar

Date <https://www.drupal.org/project/date>

Calendar module <https://www.drupal.org/project/calendar>

Date iCal module <https://www.drupal.org/project/date_ical>

Libraries API <https://www.drupal.org/project/libraries>

You'll need to use an FTP login that has write permission to the /var/www/sites/all directory. Note: if you can't get this to work, you can always install modules the old way. This might be slightly more convenient.

**Lab 5: Install Drupal**

In this lab you will install the Drupal CMS on your server and perform some initial configuration.

*Hint:* since you will be doing lots of work that requires root permission, you may wish to log in as the root user so you don't have to put "sudo" before every command. To do so you can type "sudo su root". (If you aren't doing extensive root work, it's better just to use sudo when you need root permissions.)

**Tasks**

php-mysql, mcrypt, php-mcrypt, php-gd, php-xml, php-imagick, and php-mbstring

then restart server with service apache2 restart

install mysql-server for mysql database server. Log in as sudo su root

create database called “drupal7”

create database user called “drupal7”

give drupal7 user permission to read and modify the drupal7 database. Grant all privileges

install drupal

1. Right click on tar.gz version to copy link address
2. Uncompress
3. Now we need to move the contents of the drupal-7.56 directory into the /var/www/html directory. You can do so with the command

cd drupal-7.56; mv \* .htaccess ..

You need to specifically move the .htaccess--because its name starts with "." it is treated as "hidden"; it won't be moved unless you specifically add it to the command.

Delete the index.html file in /var/www/html so that the new index.php file will open by default.

**Set up a new drupal site**

1. Enable Clean URLs for your website. Lab6-1
2. Drupal needs to be able to send email to users—for example, to send them an initial password. Your ubuntu machine is not initially set up as a mail server. To set it up, use apt-get to install the **mailutils** and **postfix**mail server package. If postfix wasn't previously installed, you'll be asked some configuration questions. Set up your server as an "Internet site". You will be asked for the fully qualified domain name. Use the domain name I set up for your lab server—*mylogin*.calvincs.com

Test your mail server by typing "mail *youremailaddress*" and sending an email to yourself. Note that to end the body of the mail message, you can type control-D.

1. **cron** jobs run every hour. Add a cron job using wget to call cron.php every how following [these instructions](https://www.drupal.org/docs/7/setting-up-cron-for-drupal/configuring-cron-jobs-using-the-cron-command). You'll actually need to add two lines to your crontab file, one for mylogin.calvincs.com and one for proj.mylogin.calvincs.com.
2. You will probably have to increase the memory limit for php in your settings.php file. 128MB should be enough for just about any modules you wish to add. Basically you will edit /var/www/html/sites/default/settings.php and add the following at the end of the file:   
   ini\_set('memory\_limit', '128M');  
   Also make this change for your development site. If you want them, you can get detailed instructions [here](http://drupal.org/node/207036).
3. Theme.

**Set up your lab website**

1. Create a **basic Web page** with a bit of information about your lab site using Add Content -> Basic Page. Use the title "My lab website" and some text in the body.
2. Enable the Forum module. Click "configure" to add an additional forum container. Add a forum in it. Then add a forum topic inside the forum. Check out the URL /forum to make sure that it worked correctly.
3. Menus - Main menu – add link
4. Create an "About" page for your site with some made-up information about your site. Set the URL path to about before saving the page.
5. The contact module lets you set up forms where users can contact the site managers by email. Enable the contact module, and then configure it and click "Add Category." Enter a category "Contact Us" and add your email address in the recipient box. Configure it to send an email that says "Thanks for your feedback. We'll get back to you as soon as possible." Make it the default. Now check it out by visiting the page /contact.
6. Configure the **blocks** on your web site with Structure -> Blocks. Remove the "Powered by Drupal" block in the footer.
7. Add a new **role** to your website of *editor* at People -> Permissions -> Roles. Editors should have all the permissions on Comment, Contact, and Forum, and in the Node section, all permissions except the first three, which have warnings.
8. Figure out how to add links for the About and Contact pages, centered in the footer block.
9. In these labs and in your term project, you'll be making changes to your websites that have the potential to break them. We're going to use git, a version control system, basically as a backup for now.

Once your sites are running, initialize a git repository for your /var/www directory:  
cd /var/www; git init  
Commit your current /var/www directory with   
git add \*  
git commit -m "Initial commit"

Repeat the last two commands (with a different message) every time you want a new backup.

Lab 7

1. Before you start, back up your website databases by exporting their contents into files with commands such as

mysqldump -u root -p drupal\_*mysite* > *mysite-todaysdate*.sql

It's possible to mess up your web site, so you should make a backup before making significant changes. You can restore your website to an earlier version by deleting the database and restoring an older version. I won't add backup instructions to future labs—instead, you should make it a regular practice.

Similarly, before you upgrade a Drupal version, installing packages, etc. (or after you have made changes and your site is working the way you want it to), you should do a git commit. That is,  
cd /var/www/   
git add \*  
git commit

1. Install the Ubuntu package ftpd to enable you to use Drupal's update manager. In order for this to work, the FTP user you log in with must have write permission to the /var/www/html/sites/all directory.

Now install the [module filter](http://drupal.org/project/module_filter) module using the web-based method, i.e. download the zip file into /var/www/sites/all/modules and unzip it. Enable module filter and update manager modules.

1. Now you should have a new link, "install new module", available on your module page. Use it to install install [ctools](https://drupal.org/project/ctools) and [sweaver](http://drupal.org/project/sweaver) for styling your website. While you are at it, install the CKEditor module [CKEditor](http://drupal.org/project/ckeditor)module (a WYSIWYG editor). You'll need to use an FTP login that has write permission to the /var/www/sites/all directory. Note: if you can't get this to work, you can always install modules the old way. This might be slightly more convenient.
2. Make sure you have Administrator, Editor, and Authenticated User **roles** set up on your server with appropriate permissions.
3. Configure **roles** so that editors and authenticated users can access CKEditor and administrators can administer it.
4. Enable the blog module and create a sample blog entry. Make sure it is present on the /blog page.
5. Add a link to the blogs in your primary menu.
6. Set up permissions so that editors (and administrators) can create and edit their own blog entries.
7. Enable sweaver. Use it to customize some aspect of your theme. Then disable it again.
8. Install and enable Mollom or reCAPTCHA to control spam.
9. Configure **roles** so that editors and authenticated users can access CKEditor and administrators can administer it.
10. Set up permissions so that editors (and administrators) can create and edit their own blog entries.

**Hints**

**Ubuntu**

* Updating apt's package list: apt-get update
* Finding a package: apt-cache search *packagename*
* Installing a package: apt-get install *packagename*
* Restarting Apache: /etc/init.d/apache2 restart
* Upgrading all packages to the latest version: apt-get upgrade

**Drupal**

* Back up a database

mysqldump -u root -p databasename > databasename-[date].sql

* Installing a module with download URL [url]:

sudo su root

cd /var/www/sites/all/modules

wget [url]

tar xvfz [package].tar.gz

rm [package].tar.gz

exit

**SSH**

* [Instructions](http://cs.calvin.edu/courses/cs/108/snelesen/references/remote.html) for connecting to departmental machines from off campus via SSH

[From S. Nelesen]

**Hosting**

* [Clubs.calvin.edu contact form](https://docs.google.com/a/students.calvin.edu/forms/d/1J1dM1ha3gMo1h8FMQiaQ0tGvUNwsY7TvD7zP4xMJKec/viewform)

**Software tools**

* [Ubuntu](http://www.ubuntu.com/) -- Linux OS
* [jEdit](http://www.jedit.org/)
* [Prototype](http://www.prototypejs.org/) -- JavaScript library
* [Drupal](http://drupal.org/) -- Community plumbing

**Firefox Plug-ins**

* [Firebug](http://getfirebug.com/) -- essential Web development plug-in
* [DOM Inspector](http://getfirebug.com/)

**Web References**

* [Simple vi lessons](http://www.jerrywang.net/vi/)
* [Vi cheat sheet](http://media.smashingmagazine.com/wp-content/uploads/2010/05/VI-Help-Sheet-01-large2.jpg)
* [VIM cheat sheet](https://rumorscity.com/wp-content/uploads/2014/08/10-Best-VIM-Cheat-Sheet-05.jpg)
* [HTML](http://www.w3schools.com/tags/default.asp)
* [CSS](http://www.w3schools.com/CSS/CSS_reference.asp)
* [CSS [Wikipedia]](http://en.wikipedia.org/wiki/Comparison_of_layout_engines_%28Cascading_Style_Sheets%29)
* [Apache](http://httpd.apache.org/docs/1.3/configuring.html)
* [JavaScript](http://www.w3schools.com/jsref/default.asp)
* [Prototype](http://www.prototypejs.org/api)
* [PHP](http://www.php.net/docs.php)
* [MySQL](http://dev.mysql.com/doc/refman/5.4/en/)

**Books**

* *HTML & XHTML: The Definitive Guide*, Chuck Masciano and Bill Kennedy (O'Reilly) -- reference
* *Eric Meyer on CSS: Mastering the Language of Web Design*, Eric Meyer
* *More Eric Meyer on CSS*, Eric Meyer
* *CSS: The Definitive Guide*, Eric Meyer -- reference
* *Using Drupal*, Byron, berry, Haug, Eaton, Walker, and Robbins (O'Reilly)
* *Designing Web Usability*, Jakob Nielsen -- a standard usability text