# Cloud Operations

Techlogix Cloud Operation Team provides Infrastructure Support including Servers and Application Monitoring, Cost Management and Capacity Planning. Techlogix Cloud Operation offers an Industry Leading Cloud Service Level Agreement (SLA) with 100% uptime guarantee.

**Contents**

[***Cloud Operations*** 1](#_Toc503287825)

[Purpose 4](#_Toc503287826)

[Architecture 4](#_Toc503287827)

[Host Machine Setup 5](#_Toc503287828)

[Installation of Ubuntu server 5](#_Toc503287829)

[Install packages 5](#_Toc503287830)

[Moodle Deployment 6](#_Toc503287831)

[Setup MySQL 6](#_Toc503287832)

[Create a Database 6](#_Toc503287833)

[Download and Setup Moodle 6](#_Toc503287834)

[Deployment of Multiple Moodle Instances 8](#_Toc503287835)

[Setup virtual host in Apache 8](#_Toc503287836)

[Office 365 integration with Moodle 9](#_Toc503287837)

[Plugins Installation 9](#_Toc503287838)

[Register Application in Azure 9](#_Toc503287839)

[Enter Azure application credentials into Moodle 12](#_Toc503287840)

[Configure the Office 365 support plugin 13](#_Toc503287841)

[References 14](#_Toc503287842)

Revision Chart

This chart contains a history of this document’s revisions. The entries below are provided solely for purposes of illustration. Entries should be deleted until the revision they refer to has actually been created.

The document itself should be stored in revision control, and a brief description of each version should be entered in the revision control system. That brief description can be repeated in this section.

It can be removed from the client version of the document, however the local copy of the client version should contain the revision chart and version column should exhibit the “Client Version” in it.

Revision Chart

| Version | Author(s) | Description of Change | Effective Date |
| --- | --- | --- | --- |
| 1 | Ammar Sajid | Initial draft created for distribution and review comments | November 24, 2017 |
| 2 | Saad Saood | Update diagram | December 5,2017 |

# Purpose

The purpose of this document is to provide a complete guide for the deployment of Moodle. The document is supposed to be used for the deployment of Moodle version 3.3 on Ubuntu server 16.04

# Architecture

The deployment model that we followed achieves multi-tenancy but keeping in mind customer’s data security. The architecture looks like this;

**Using Moodle URLs**

moodle-test1.almusnet.com

moodle-test2.almusnet.com

moodle-test3.almusnet.com

RMI Users

SSUET Users

CECOS Users

**Moodle-3**

**Moodle-2**

Apache

VHost-1 VHost-2 VHost-3

**Moodle-1**

DB-3

DB-2

DB-1

DB SERVER

The current architecture uses a single host machine for multiple Moodle instances. Each Moodle instance would have its own Database, data directory and codebase. The real advantage of this architecture is to reduce the infrastructure and operational cost by optimizing the utilization of underlying infrastructure but keeping data isolation for each user.

# Host Machine Setup

## Installation of Ubuntu server

Install Ubuntu server 16.04 on a physical or virtual machine. Make sure the machine has access to internet. This machine can be used for all Moodle instance.

## Install packages

Open up terminal and run the following commands to install required packages;

* Install pending updates
  + $ sudo apt-get update
* Install apache server
  + sudo apt-get install apache2 mysql-client mysql-server php7.0 libapache2-mod-php7.0
* Install PHP
  + $ sudo apt-get install php7.0 libapache2-mod-php7.0
* Install MySQL and set root password for MySQL server
  + $ sudo apt-get install mysql-client mysql-server
* Install additional packages
  + $ sudo apt-get install graphviz aspell ghostscript clamav php7.0-pspell php7.0-curl php7.0-gd php7.0-intl php7.0-mysql php7.0-xml php7.0-xmlrpc php7.0-ldap php7.0-zip php7.0-soap php7.0-mbstring
* Install Git tool for downloading code from github
  + sudo apt-get install git-core

# Moodle Deployment

## Setup MySQL

* Open MySQL configuration file with write permissions
  + $ sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf
* Add below lines in the **mysqld.cnf** file under **[mysqld]** section. If these lines already exist, then replace.

default\_storage\_engine = innodb

innodb\_file\_per\_table = 1

innodb\_file\_format = Barracuda

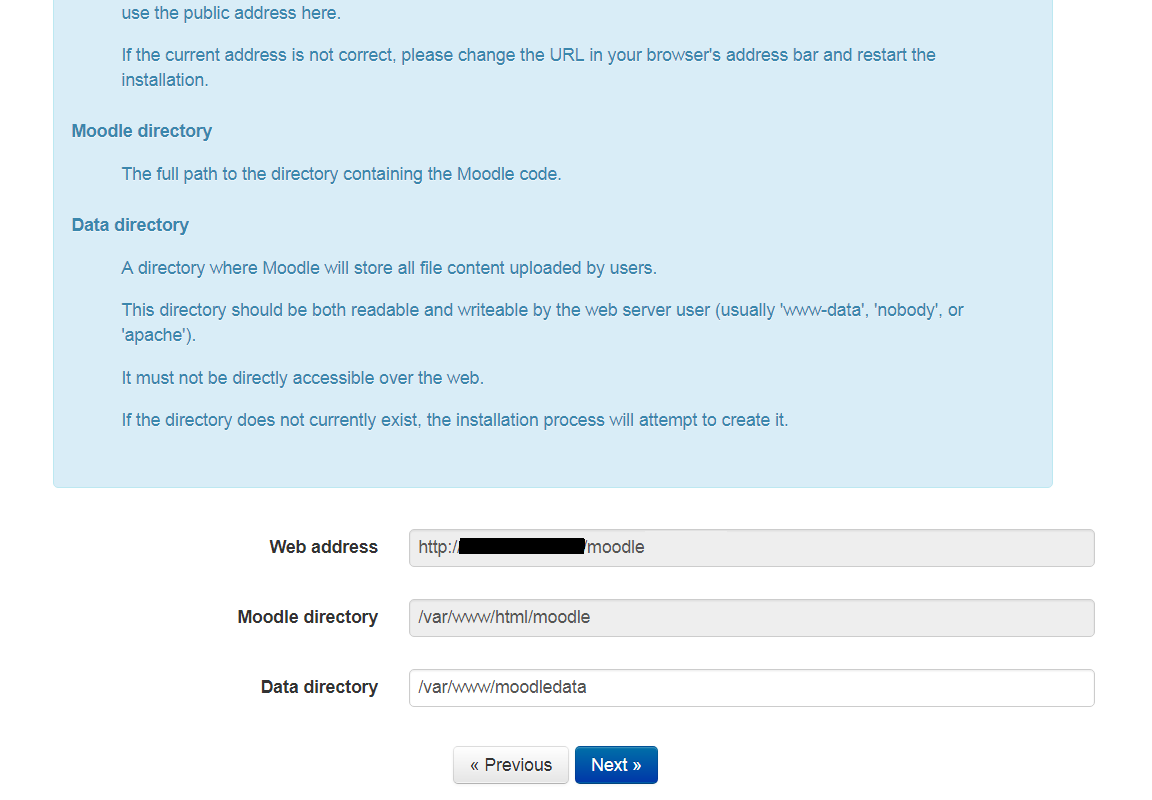
* Restart MySQL service for changes to take affect
  + $ sudo service mysql restart

## Create a Database

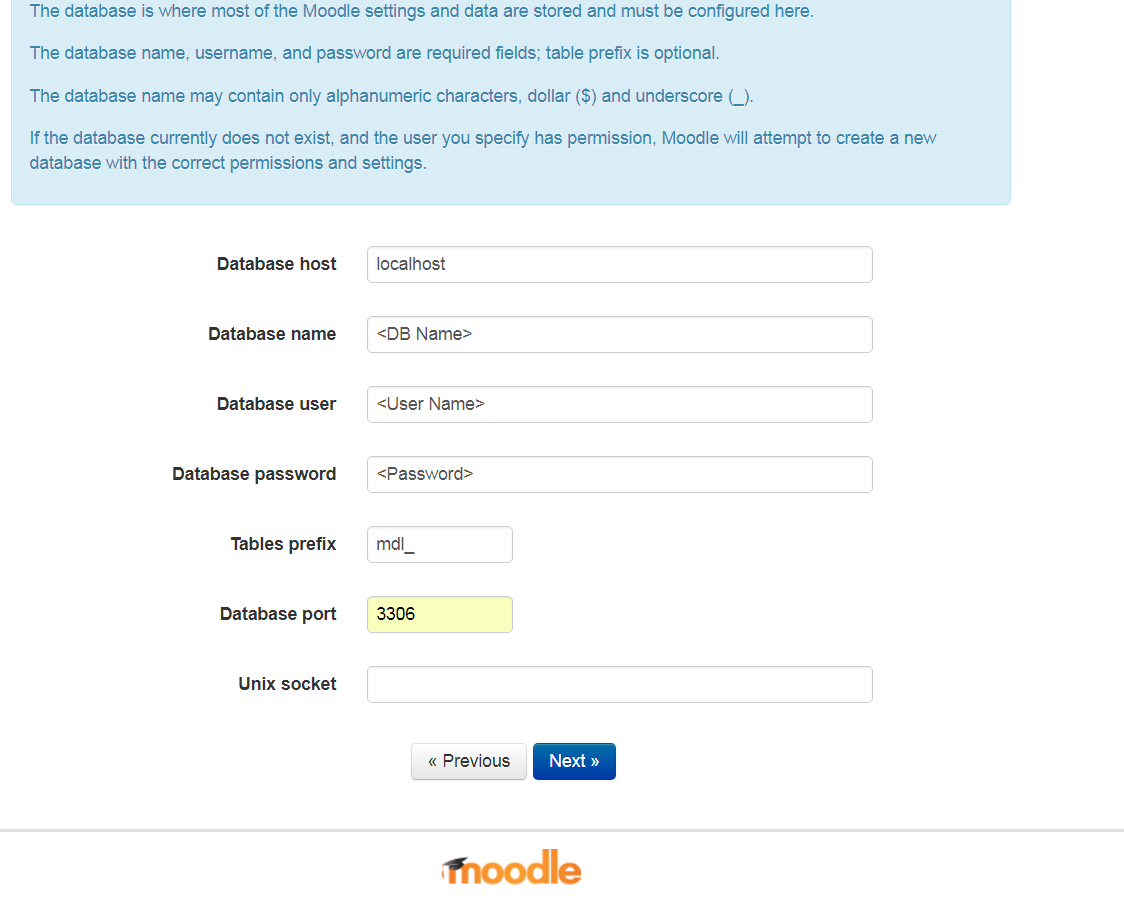
* Now you need to create Databases for Moodle instances. Run the below command to login to MySQL shell.
  + $ mysql -u root –p
* Create an empty database.
  + mysql> CREATE DATABASE **<DB Name>** DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;
* Create a user for the database that will be used by Moodle to access the database and assign permissions. Replace <User Name>, <DB Name> and <Password> with your desired strings.
  + mysql> CREATE USER '**<User Name>**'@'localhost' IDENTIFIED BY '**<Password>**';
  + mysql> GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,CREATE TEMPORARY TABLES,DROP,INDEX,ALTER ON **<DB Name>**.\* TO **<User Name>**@localhost IDENTIFIED BY '**<Password>**';

## Download and Setup Moodle

* Run the following commands to get the code of Moodle version 3.3 from GitHub.
  + sudo git clone git://git.moodle.org/moodle.git
  + cd moodle
  + sudo git branch --track MOODLE\_33\_STABLE origin/MOODLE\_33\_STABLE
  + sudo git checkout MOODLE\_33\_STABLE
* The commands below will copy the downloaded code with correct permission to run/execute.
  + sudo cp -R /opt/moodle /var/www/html/moodle
  + sudo chmod -R 777 /var/www/html/moodle
  + sudo mkdir /var/www/moodledata/
  + sudo chown -R www-data /var/www/moodledata
  + sudo chmod -R 777 /var/www/moodledata
* Open your browser and go to http://<IP\_address>/moodle. Change the data directory if you want to but make sure your data directory has the correct permissions.



* Enter the name and credentials of database that was created in the earlier step.



* Clink next then it will connect with database and insert schema.
* After DB setup, it will ask you to create an Admin account for Moodle management. Keep going till your admin account is setup.

# Deployment of Multiple Moodle Instances

The steps given in the above sections help you to create a single Moodle instance on Ubuntu host. In order to create multiple instances on the same host, we have to repeat the given sections for each instance;

* Create a Database
* Download and Setup Moodle

## Setup virtual host in Apache

We need to setup virtual hosts if we want to setup multiple instance on the same host with different URLS but same IP address.

Let’s say we have two Moodle instances installed “moodle1” and “moodle2” then we have to create two virtual hosts by following the below steps;

* Create two virtual host configuration files under “/etc/apache2/sites-available/” directory. vhost files are the files that specify the actual configuration of our virtual hosts and dictate how the Apache web server will respond to various domain requests.
  + $ touch moodle1.example.com.conf moodle2.example.com.conf
* Open these files one by one in your favorite editor and add the below content. Replace moodle1 with moodle2 for second file.

<VirtualHost \*:80>

ServerAdmin admin@moodle.com

ServerName moodle1.example.com

ServerAlias www.moodle1.example.com

DocumentRoot /var/www/html/moodle1

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

* We can use the a2ensite tool to enable each of our sites like this:
  + sudo a2ensite moodle1.example.com
  + sudo a2ensite moodle2.example.com
* Restart the apache service for changes to take effect. Make sure you don’t see any errors when service restarts.
  + $ sudo service apache2 restart
* Now you need to add DNS records against your domains “moodle1.example.com” and “moodle2.example.com”. If you just want to test these domains before DNS entries, then you can add those entries in your local hosts file.
* Open your favorite browser and hit the URLs. Here you go!

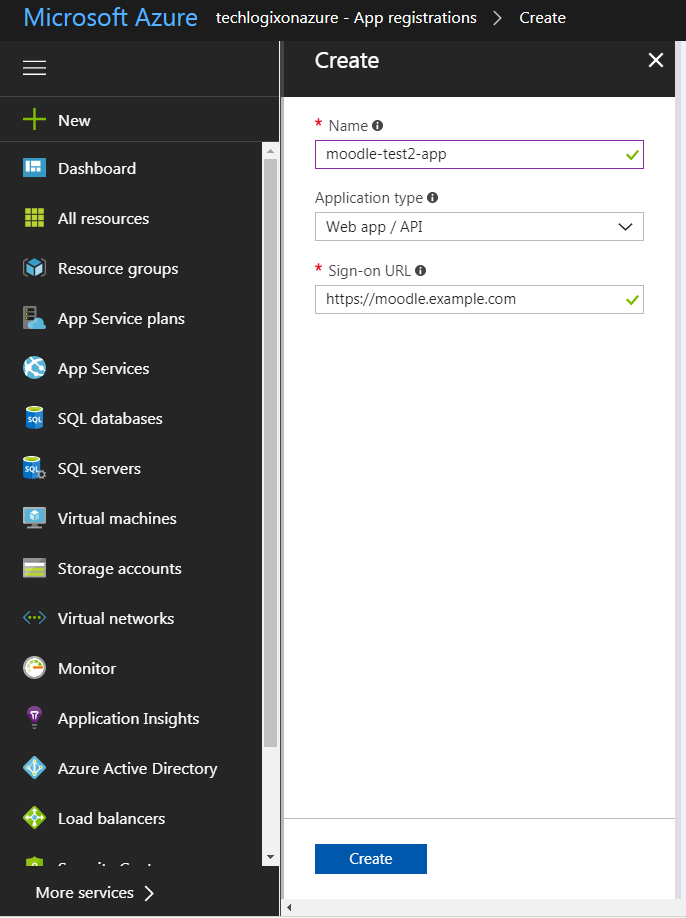
# Office 365 integration with Moodle

## Plugins Installation

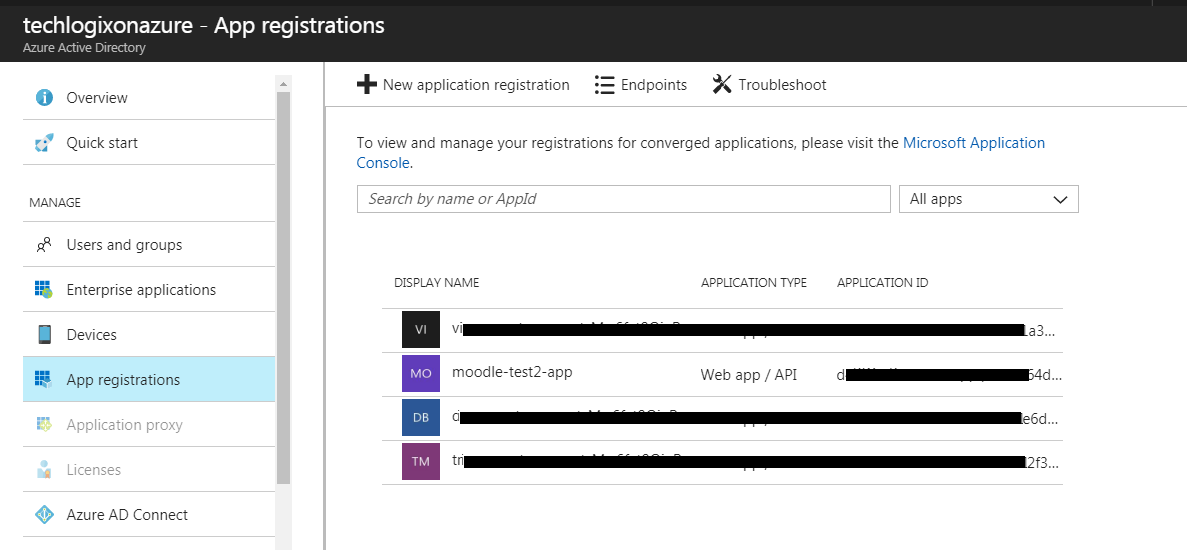
* Download office 365 plugin for Moodle and then copy it .
  + $ sudo curl -k --max-redirs 10 [https://github.com/Microsoft/o365-moodle/archive/MOODLE\_33\_STABLE.zip -L -o o365.zip](https://github.com/Microsoft/o365-moodle/archive/MOODLE_33_STABLE.zip%20-L%20-o%20o365.zip)
  + $ sudo apt-get install unzip
  + $ sudo unzip o365.zip
  + $ cp -r o365-moodle-MOODLE\_33\_STABLE/\* /var/www/html/moodle
* Now hit the URL of your Moodle instance and install the plugins
* Login to Moodle using admin creds
* Navigate to Site Administration > Plugins > Authentication and click Manage authentication
* Locate the OpenID Connect authentication plugin and click the eye icon to enable
* Click the Settings link for the plugin.
* Verify the Authorization and Token endpoints. These should be set by default but if not, set the endpoints to the following:
  + Authorization Endpoint: <https://login.windows.net/common/oauth2/authorize>
  + Token Endpoint: <https://login.windows.net/common/oauth2/token>
  + Note: “common” in the link should be replaced by your tenant ID in case you have multiple tenants configured against your login
* Note the Redirect URI. This should be the URI of your Moodle instance followed by /auth/oidc. You will need to enter this value into Azure AD later, so note this value and put it aside. For example, <https://moodle1.example.com/auth/oidc/>

## Register Application in Azure

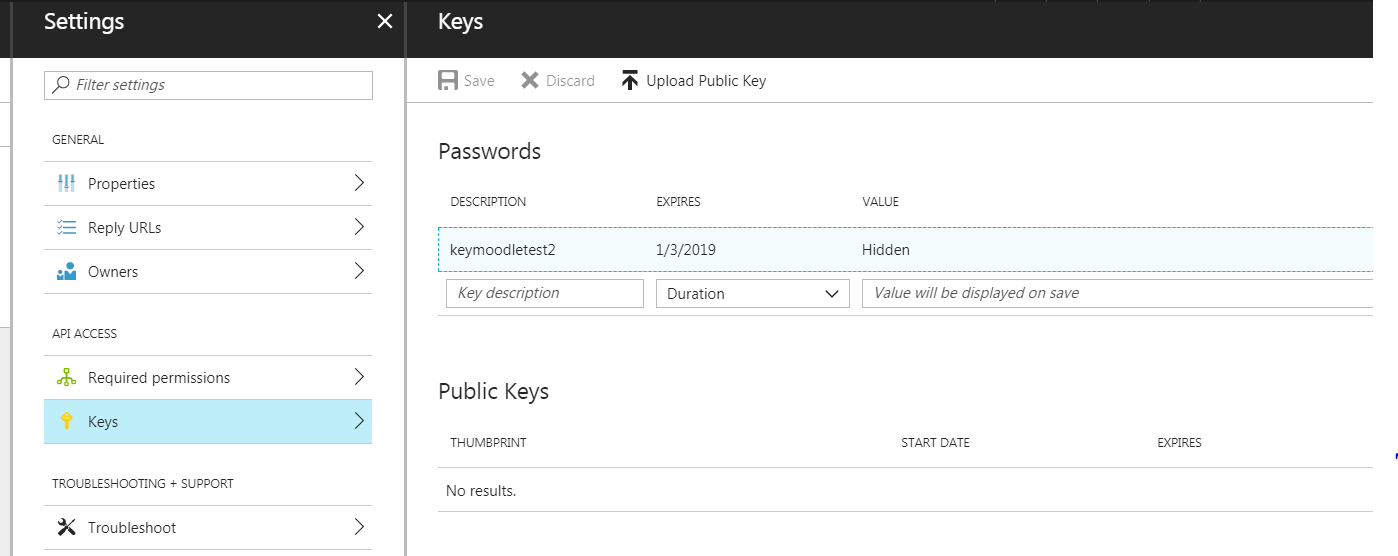
* Click on the Active Directory icon on the left menu, and then click on the desired Azure AD
* Click on “New application registration”. Add a display name, select Application type and enter the URL of your Moodle application then hit create button.



* Click on the app you just created and change settings



* Locate the Application/Client ID, note this value (write it down or copy it somewhere), and set it aside. We'll need it later.
* Locate the keys section of the page and create a client secret key.



* Note this key value (write it down or copy it somewhere) and set it aside. We'll need later.
* Navigate to the Required Permissions. Click the Add (plus sign) and then “Select an API”. Select these APIs from the list if not selected with the default permissions
  + Windows Azure Active Directory (Microsoft.Azure.ActiveDirectory)
  + Office 365 Exchange Online
  + Office 365 SharePoint Online
* Now change the permissions as specified below by just clicking on the API

|  |  |  |
| --- | --- | --- |
| API | Application Permissions | Delegated Permissions |
| Windows Azure Active Directory | Read directory data | * Read directory data * Read all users' full profiles * Sign in and read user profile. |
| Office 365 Exchange Online |  | * Read users’ calendars * Read and write users' calendars |
| Office 365 SharePoint Online |  | * Read items in all site collections * Read and write items in all site collections * Create or delete items and lists in all site collections * Have full control of all site collections * Read user files * Read and write user files |

* Now click on “Grant Permissions”. It would require admin rights on Azure active directory.

## Enter Azure application credentials into Moodle

* Ensure you have enabled the OpenID Connect authentication plugin following the steps a few sections above.
* Navigate to the OpenID Connect authentication plugin's settings page (Site Administration > Plugins > Authentication > OpenID Connect)
* Enter the "Client ID" value you noted earlier from Azure into the "Client ID" box on the screen.
* Enter the client secret key value you noted earlier from Azure into the "Client Secret" box on the screen.
* Click "Save changes" at the bottom of the screen.

## Configure the Office 365 support plugin

* Navigate to Site Administration > Plugins > Local plugins.
* Click Microsoft Office 365 Integration. Scroll down to the Setup section and complete each of the setup settings as follows.
* Application Credentials
  + This should report that the credentials have been set. If not, you need to enter your Azure credentials by following the section above.
* System API User
  + This should report "No user set". Click "Set User"
  + You will be taken to an Office 365 login screen. Log in as a user that has administrator access in your Office 365 subscription.
  + This user is used for system operations that are not specific to a single user - i.e. user sync operations. This user needs to have administrator access to be able to access all needed information. You can change this user later if needed.
* Azure AD Tenant
  + This is the domain name that identifies your Office 365 subscription, for example "contoso.onmicrosoft.com"
* Click Save changes.
* Azure Setup
  + This tool verifies that Azure has been correctly set up. Click the "Update" button to check setup.
  + If the tool reports any missing permissions, return to Azure and ensure that all required permissions have been added to your configured application for Moodle.

# References

* <https://docs.moodle.org/34/en/Step-by-step_Installation_Guide_for_Ubuntu#Procedure>
* <https://www.digitalocean.com/community/tutorials/how-to-set-up-apache-virtual-hosts-on-ubuntu-14-04-lts>
* <https://www.markbrilman.nl/2011/08/howto-convert-a-pfx-to-a-seperate-key-crt-file/>
* <https://unix.stackexchange.com/questions/31378/apache2-invalid-command-sslengine>
* <https://www.digitalocean.com/community/questions/apache2-sub-domain-configuration-with-ssl-on-ubuntu-16-04-lts>
* <https://docs.moodle.org/33/en/Office365#Plugins_.26_Features>
* <https://github.com/Azure/azure-quickstart-templates/tree/master/moodle-singlevm-ubuntu>