

SPORTCAMBODIA WEBSITE IMPLEMENTATION

**INSTALLATION DOCUMENTATION V1.0**

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | DIGIGREEN KYODAI | **Date** | 21/06/2016 - |
| **Company** | KIT | **Version** | 1.0 |
| **Issued to** | A2A Town Cambodia Ltd. | **Reviewer** | Leo |

Table of Contents

[1. Introduction 1](#_Toc454291061)

[2. System Specification 2](#_Toc454291062)

[3. Operating System Installation 2](#_Toc454291063)

[4. Python Installation 2](#_Toc454291064)

[5. Django Setup 3](#_Toc454291065)

[6. Requirement Files 4](#_Toc454291066)

[7. Setting Up Project in Local System 4](#_Toc454291067)

[8. Revision Log 7](#_Toc454291068)

[9. Appendices 7](#_Toc454291069)

[10. Approval 8](#_Toc454291070)

# Introduction

As a part of the Sport Management System implementation project, this document outlines the steps to set up the system by using the source code.

The main objectives of this document is to:

* It is to provide the steps to install the system
* It is to provide the steps to solve any problem occurring while installing
* It is to provide the reference for system technician and developer

The installation documentation sets out in more detail the installation steps of the Sport Management System with visual aid and explanation on the specific steps.

This document is made to meet the contractual terms, and once the basic requirements (v1.0) are agreed, any change is subject to a change control process.

# System Specification

Before getting into the installation steps, first we need to understand the specification and scope of our system.

* Our Sport Management System is a Web Based Management Application.
* Our system runs on any Linux platform like Kali-Linux, Ubuntu…etc
* Our system uses Python language.
* Our system is based on Django framework.
* Our system runs using NoSQL database.
* Our system integrates with Google Appengine to facilitate the database control.

|  |  |
| --- | --- |
| Subject | Specification |
| Application | Web Based Management Application |
| Platform | Linux platform |
| Programming Language | Python |
| Framework | Django |
| Database Language | NoSQL |
| Integrated SDK | Google Appengine |
| Payment Gateway |

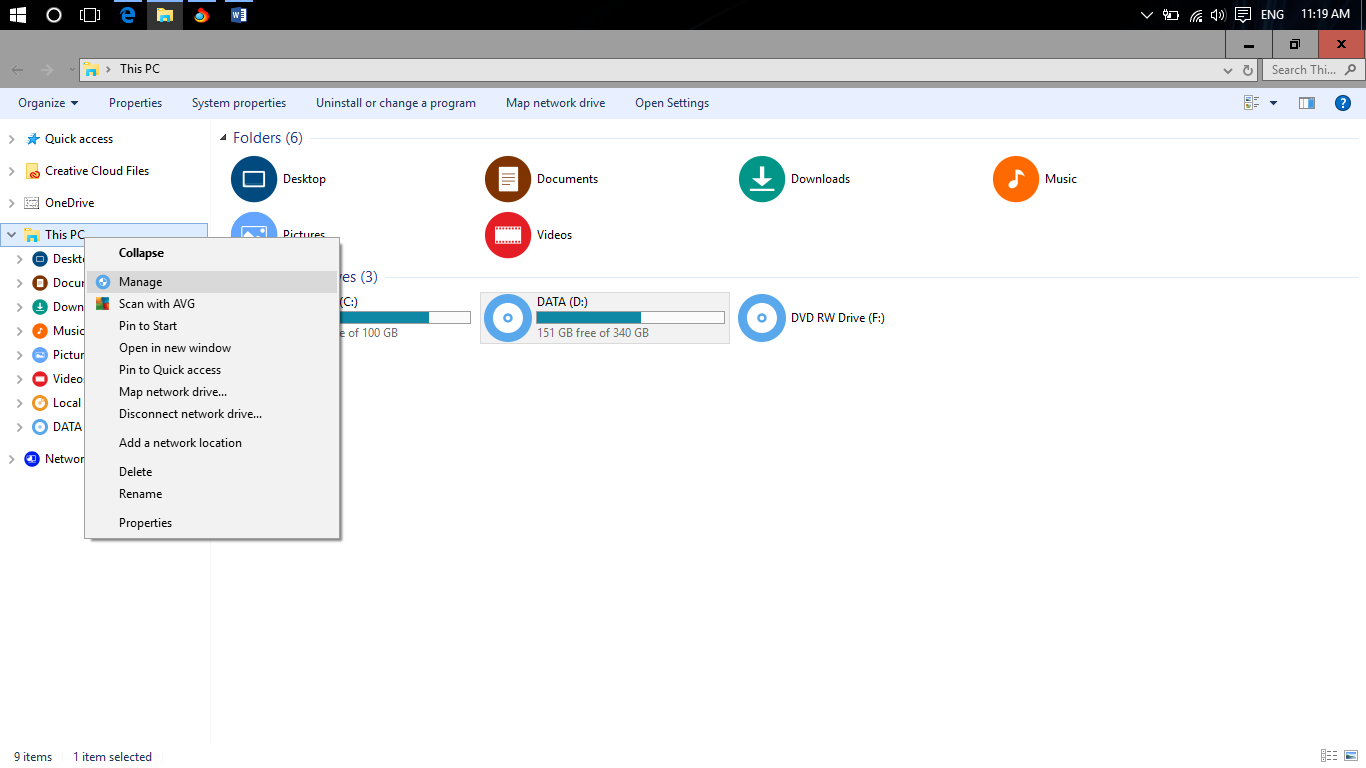
# Operating System Installation

Note\*: To those who do not have Linux operating system installed or running in their machine, this part of the document will aid you in installing the Linux OS.

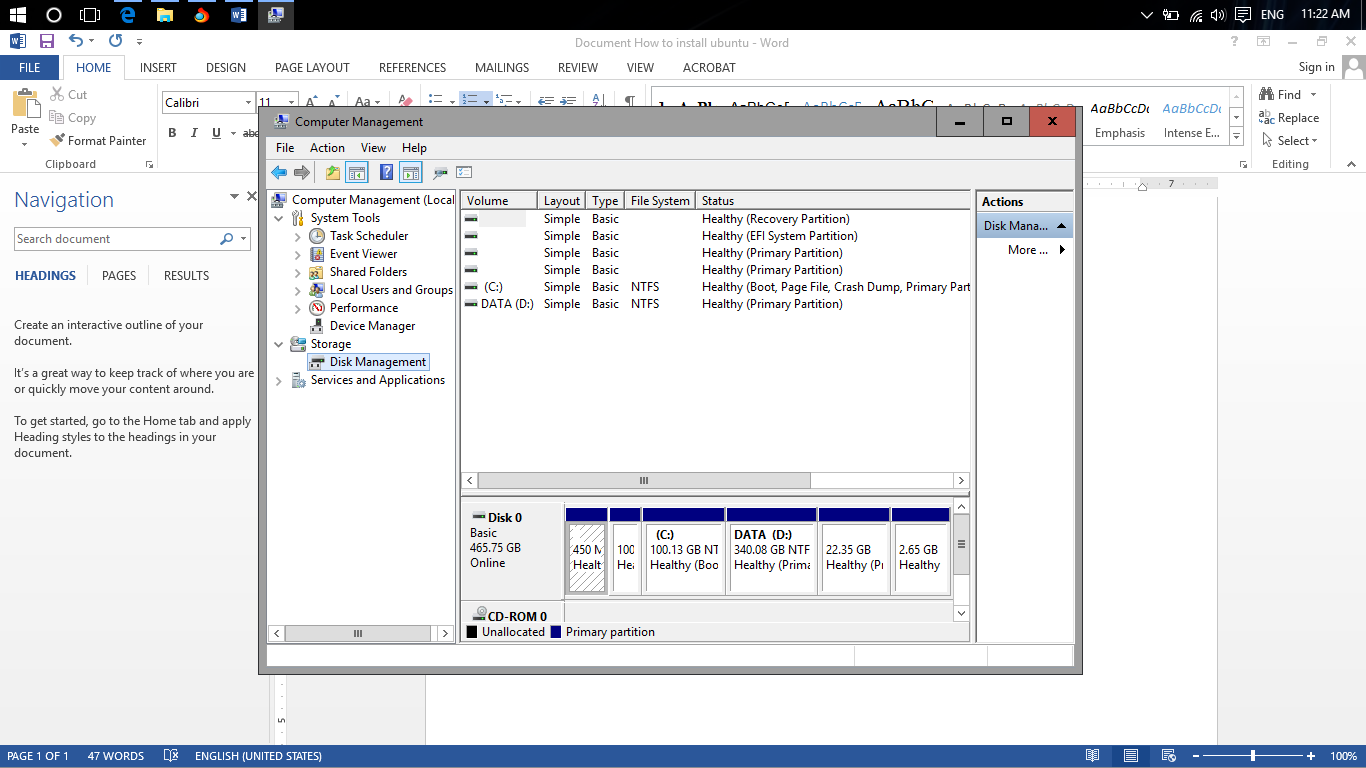
This part of the document detailed on the steps to install the linux operating that our system is going to work on.

First we divide drive from drive for installing Ubuntu:

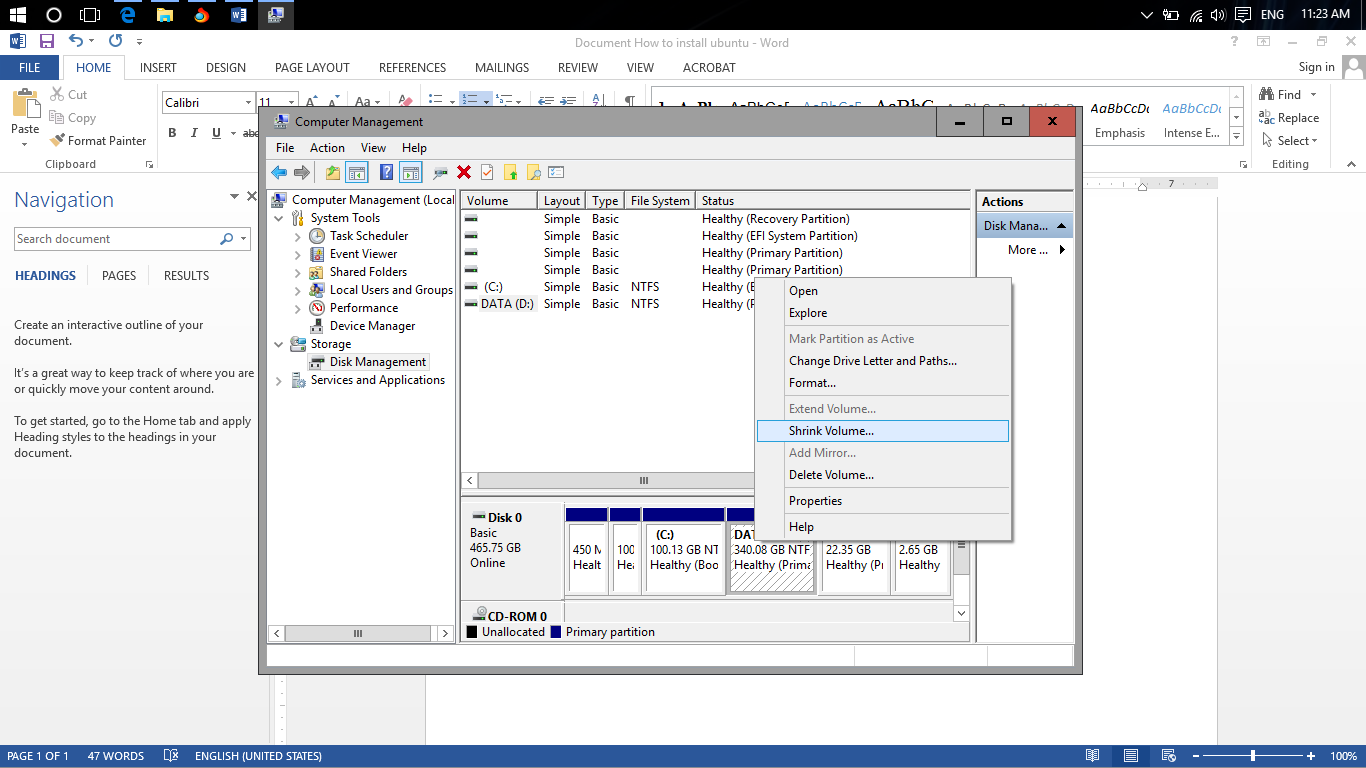
* Division drive from Local disk drive D:
  1. Go to right click on This PC
  2. Choose Manage



* 1. Go to Storage and then Disk management

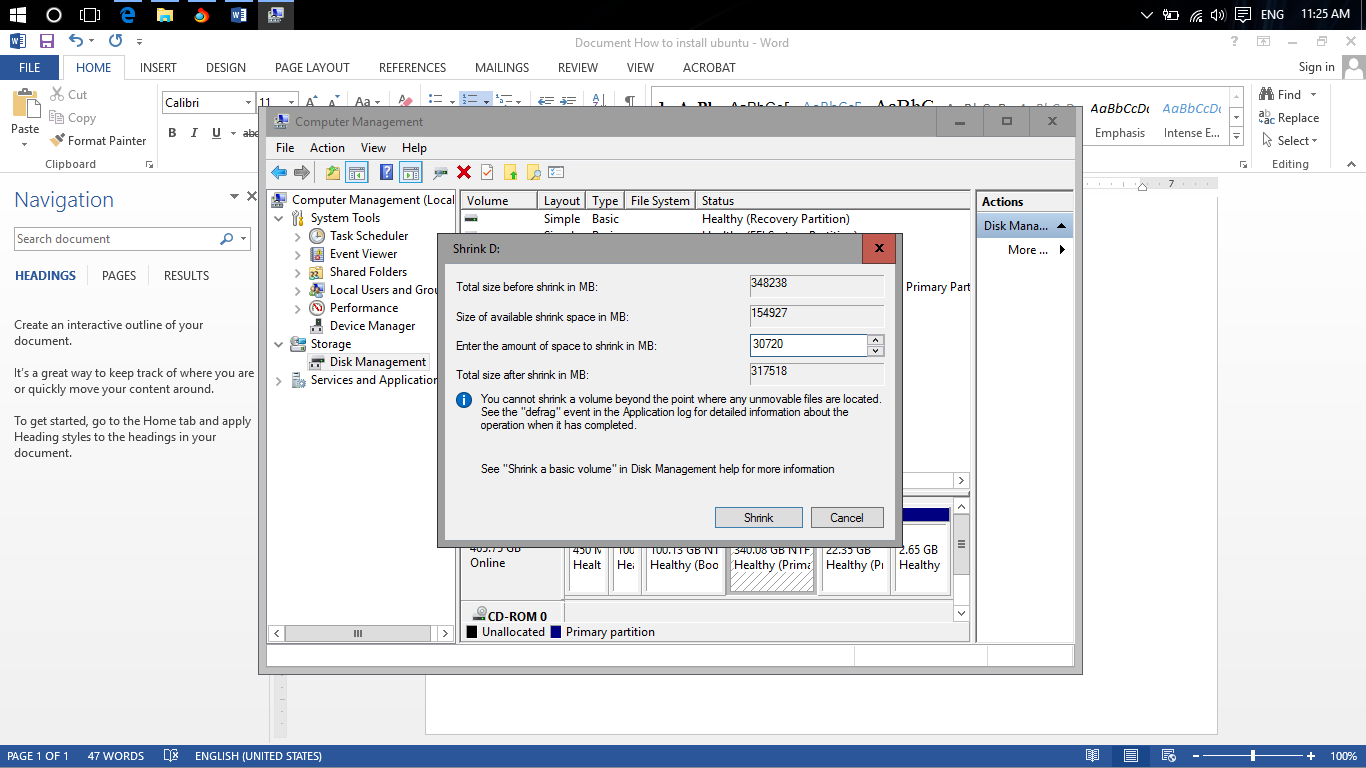


* 1. Right click on drive: D and choose Shrink Volume..

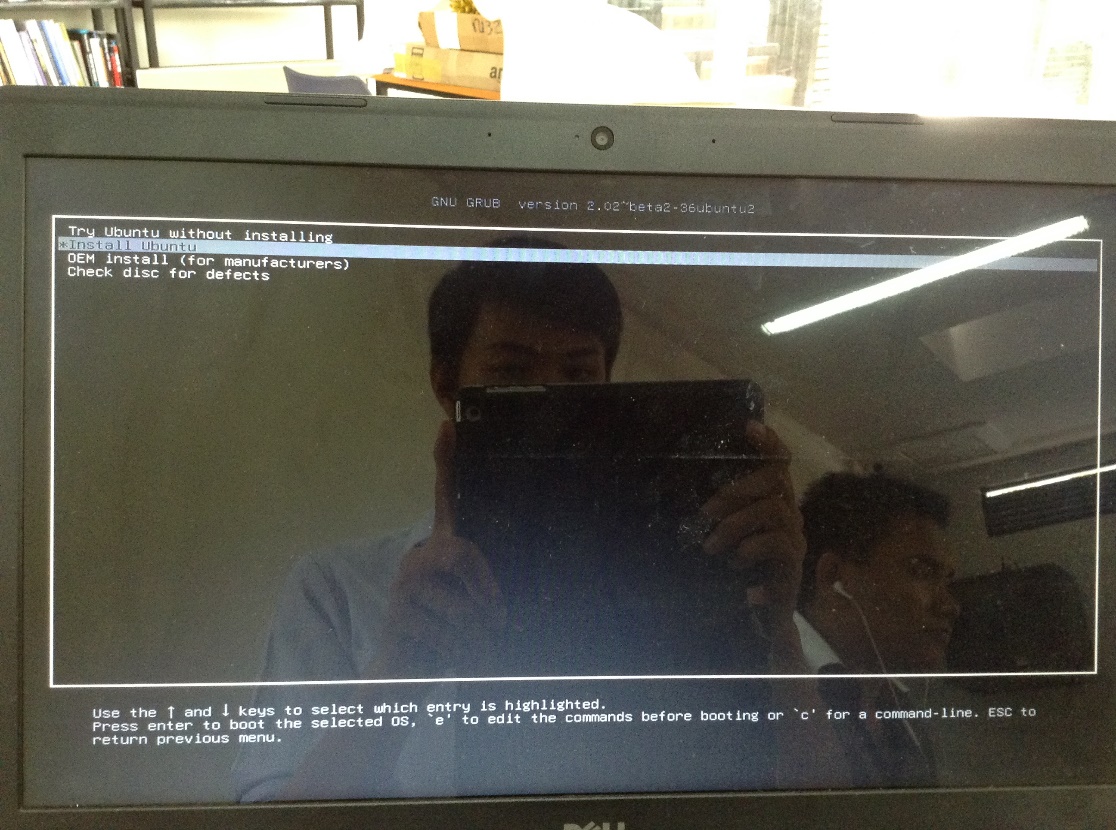
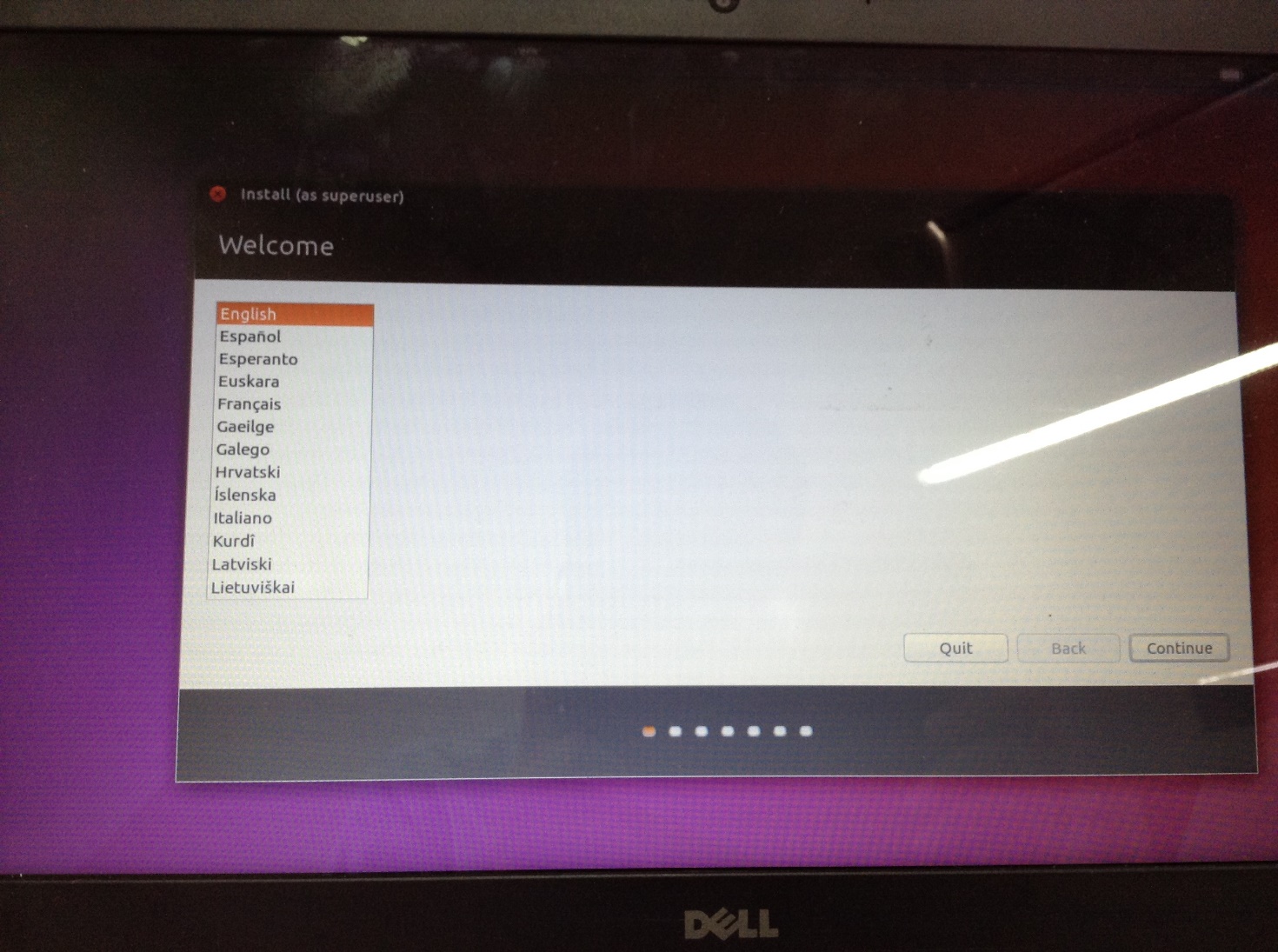
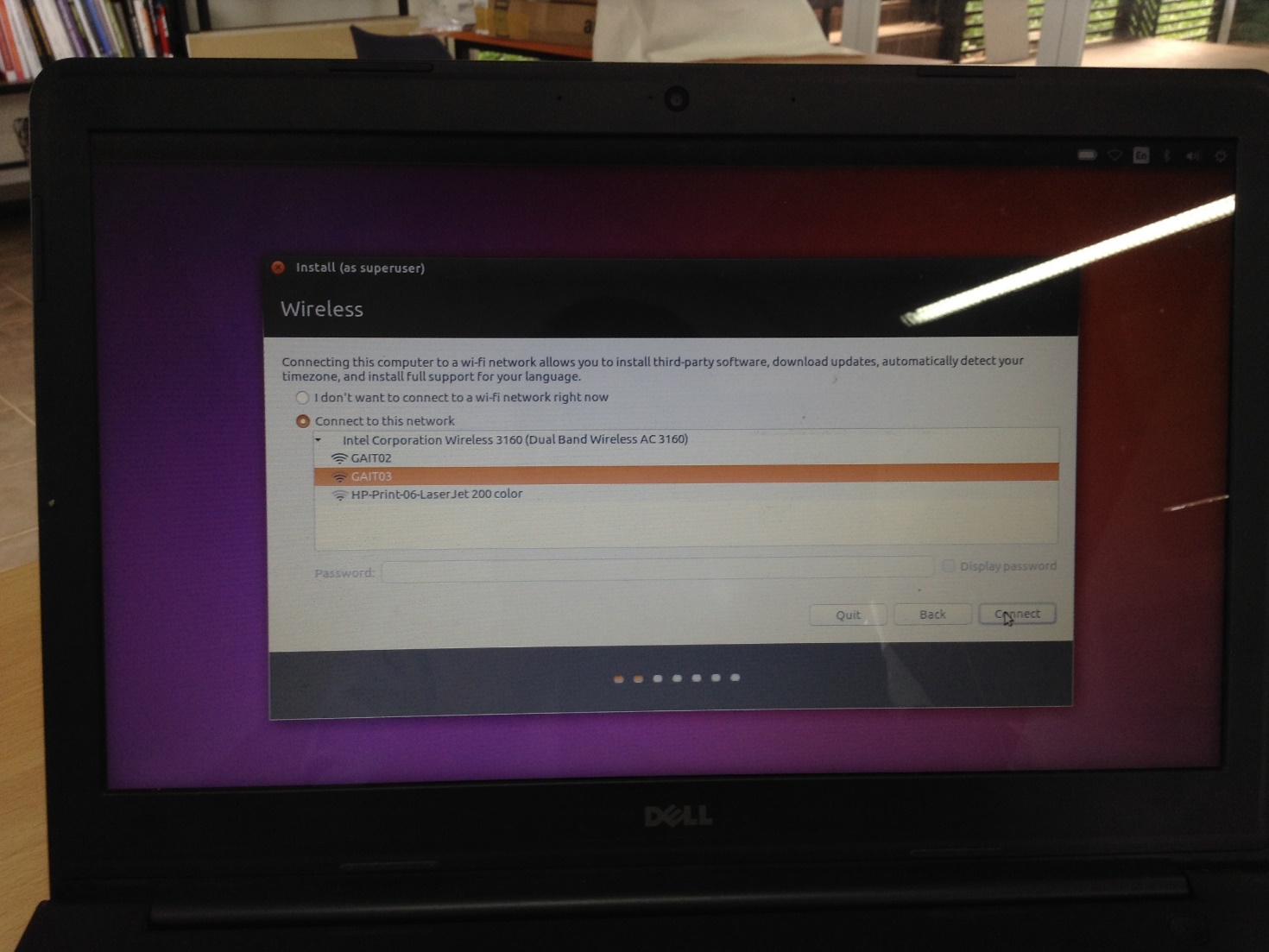
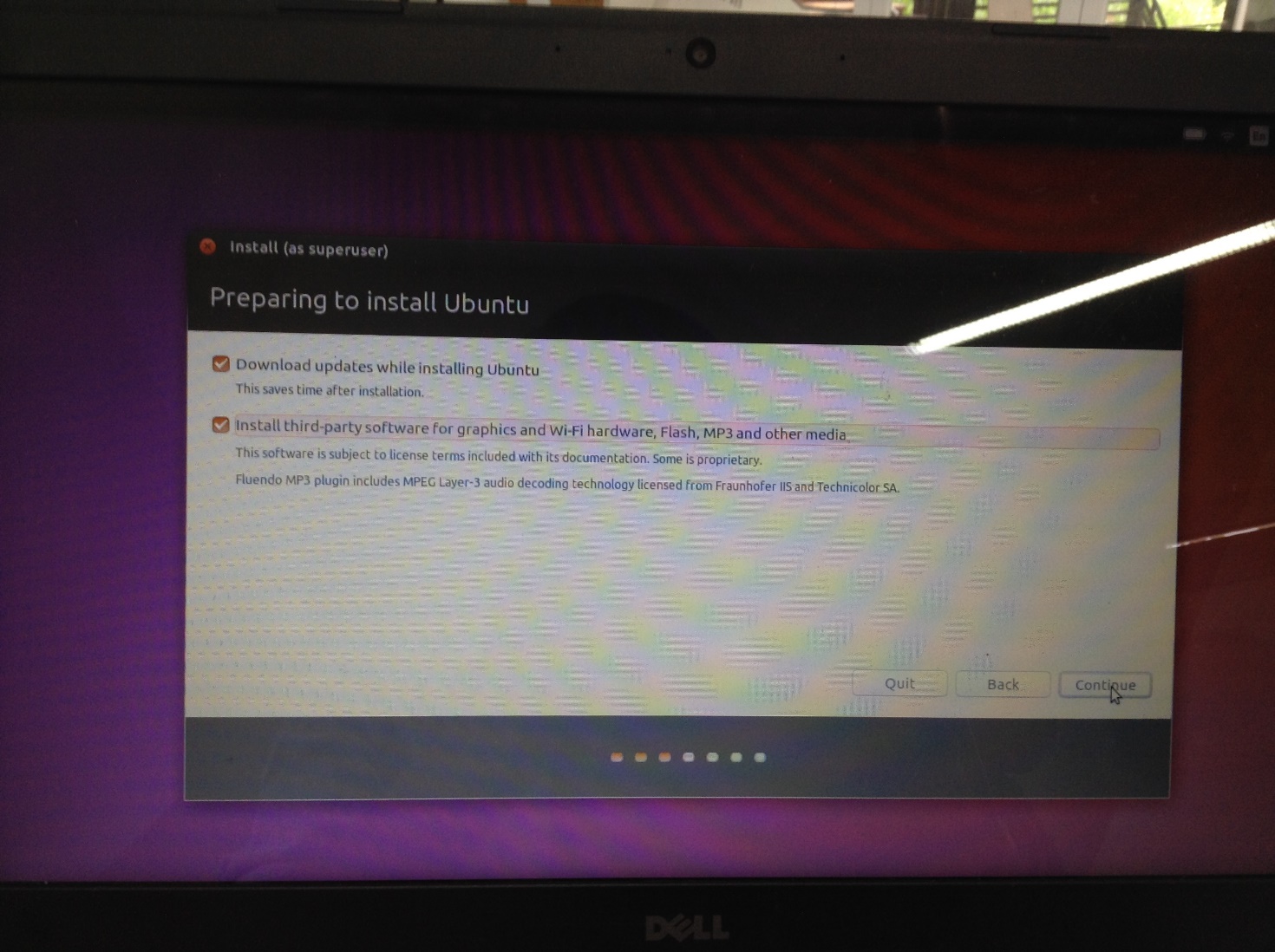
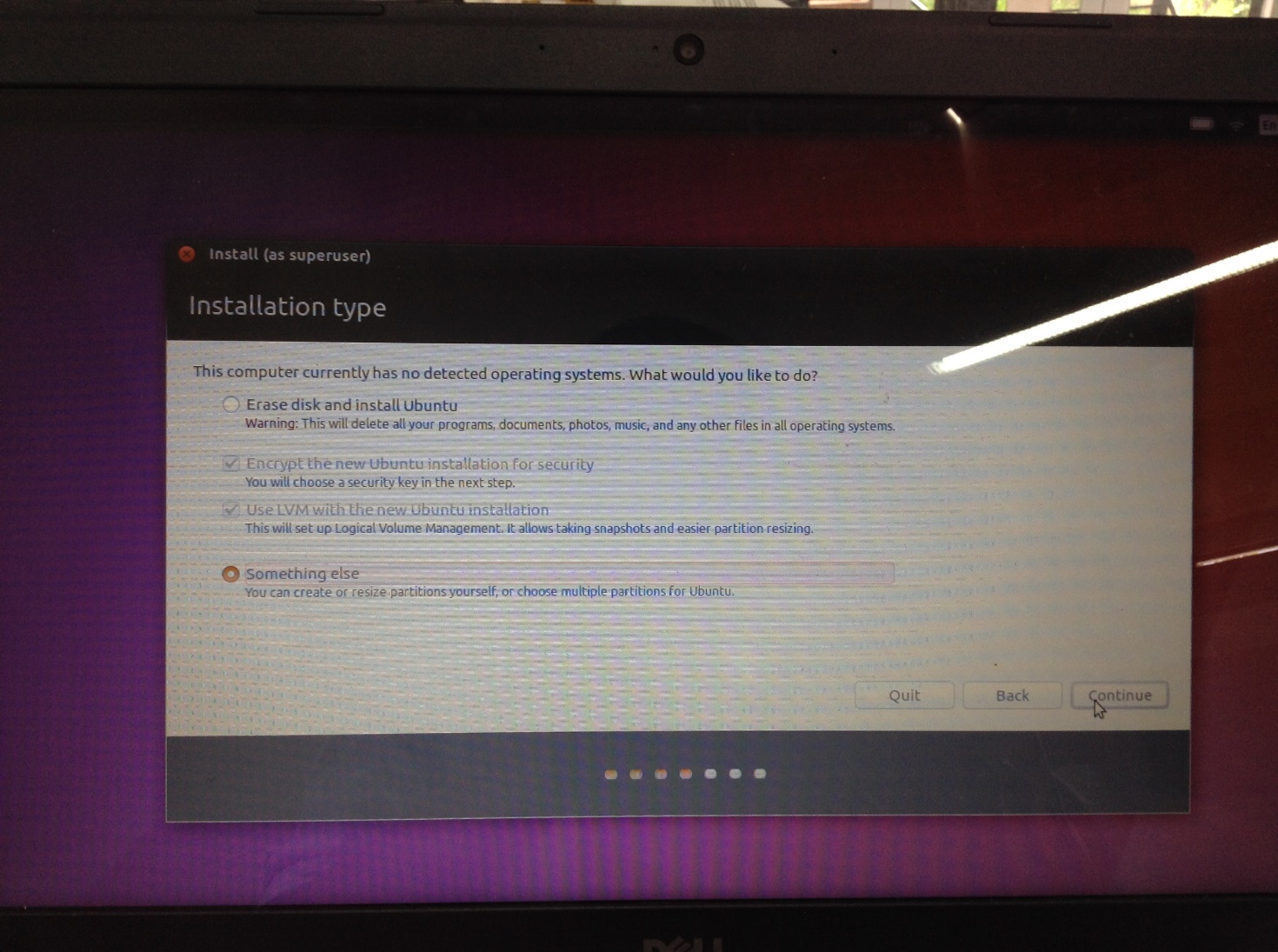
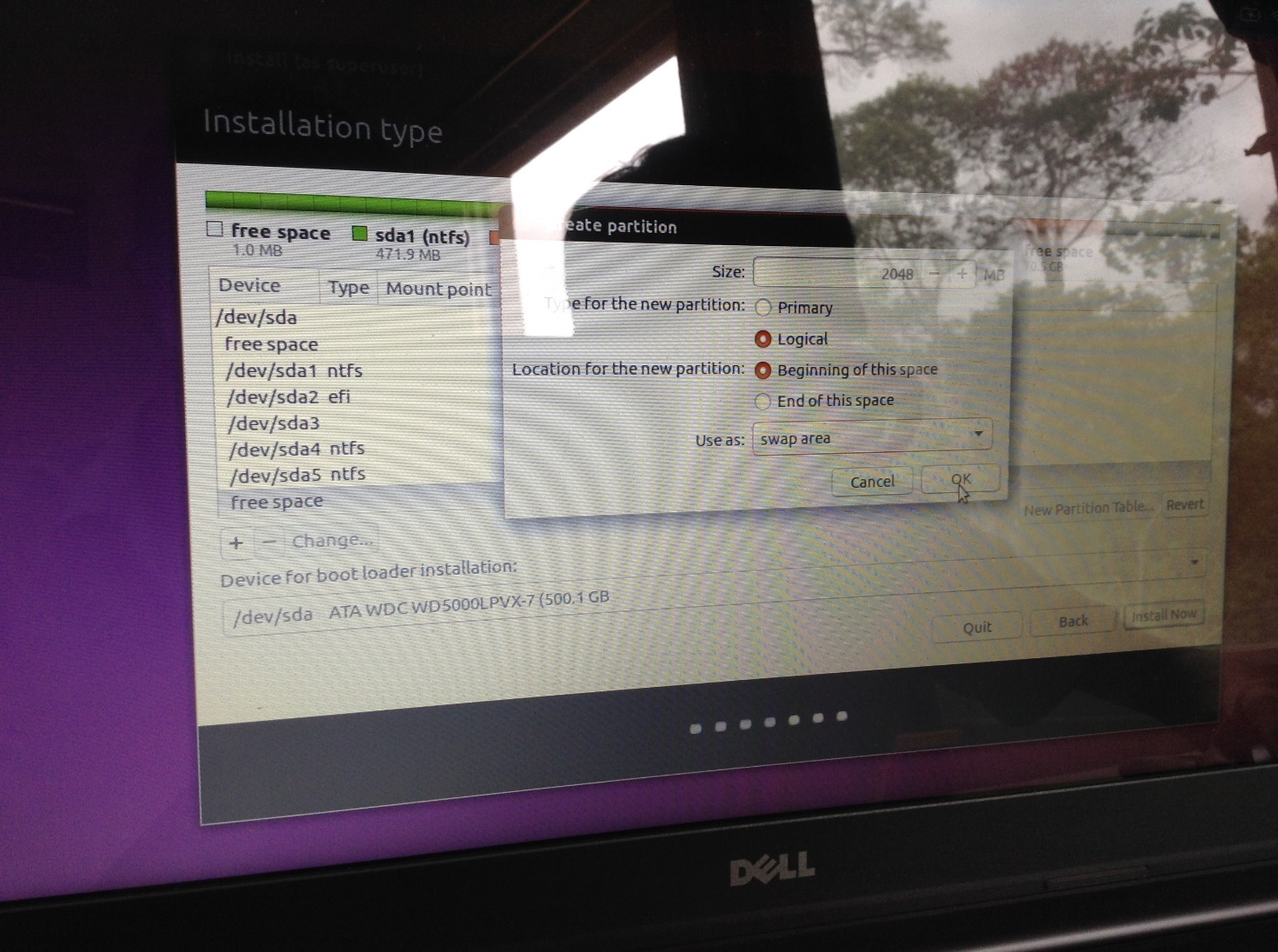
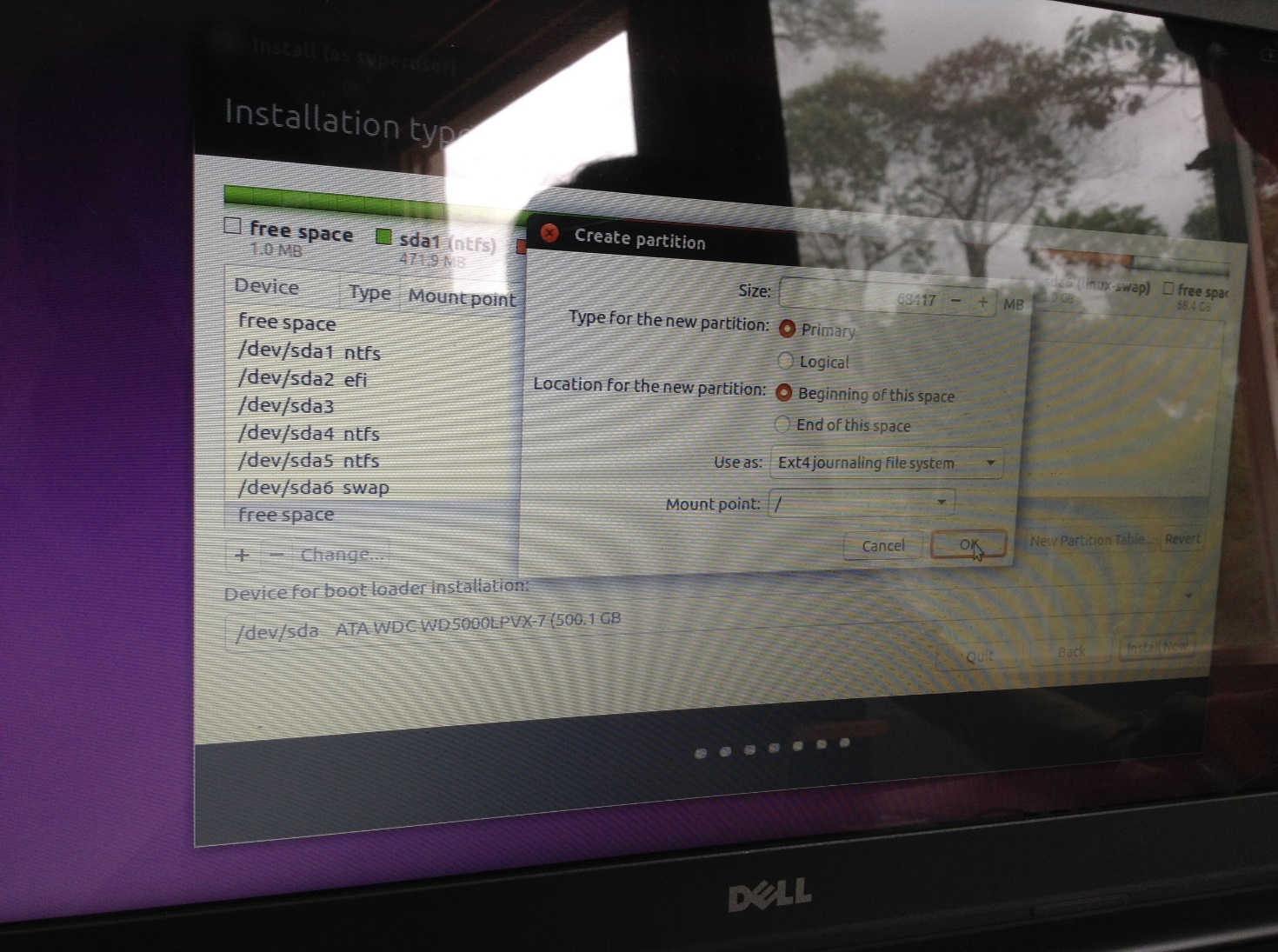
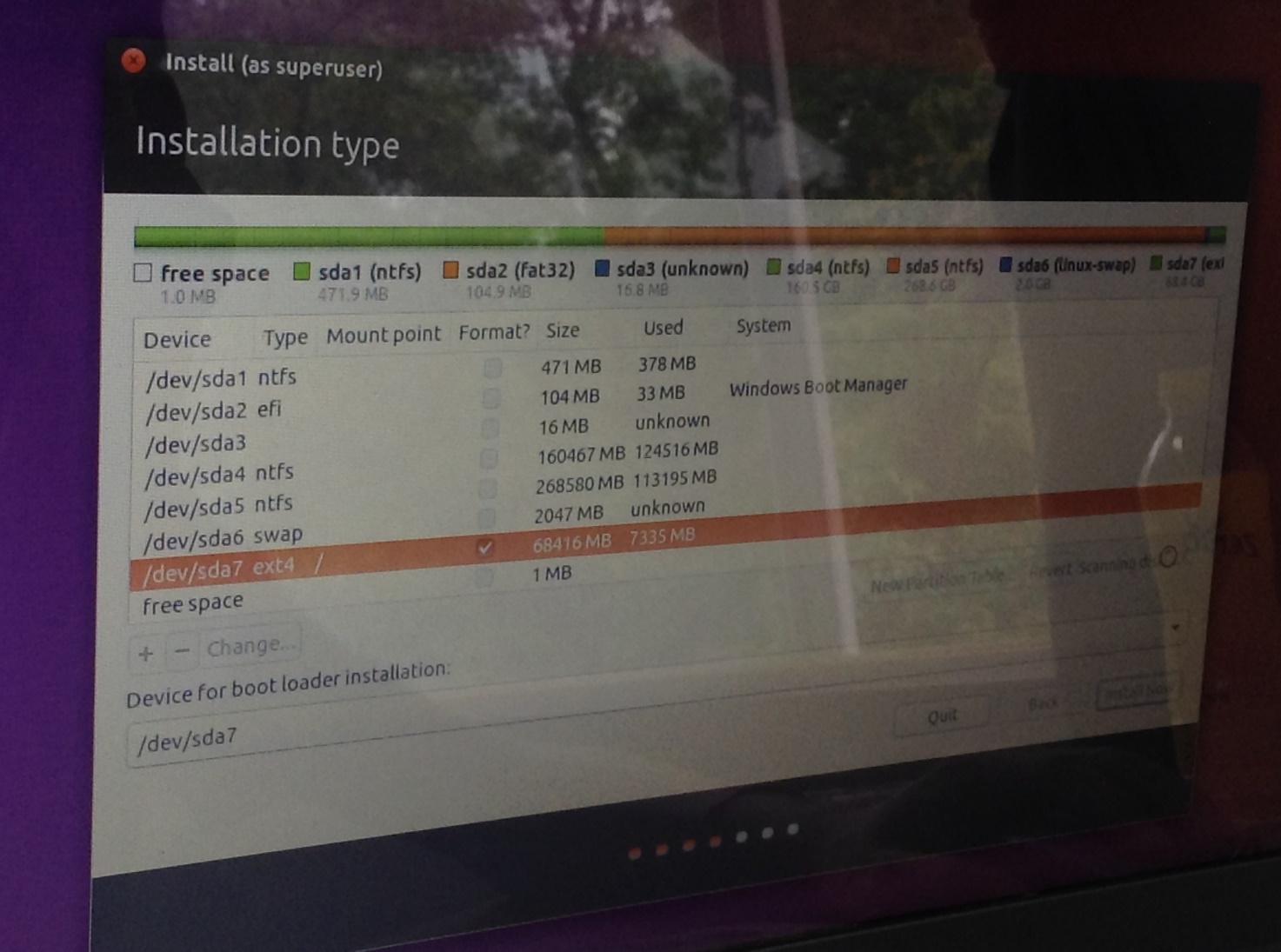


* 1. Input amount of memory to shrink 30720 . Then press Shrink button.

After than we will have another drive which are not allocated.



* Installation of Ubuntu OS

1. plug USB bootable flash drive of Ubuntu to computer
2. restart computer and always press F12 when the computer starting boot
3. Choose your USB bootable drive
4. Choose Install Ubuntu 
5. Choose Language than click continue 
6. Connect your computer to internet 
7. Check both checkbox 
8. Choose something else 
9. Select free space than press + button on the screen and create swap drive first 
10. Format drive as Ext4 to install Ubuntu 
11. Look at “ Device for boot loader installation :” Choose a drive we want to install Ubuntu OS on 

Just follow the screen until installation finish.

# Python Installation

Getting to know Python.

What is Python?

* Python is a ‘dynamic’ programming language which is one of the most popular of its kind, with perl, PHP, and Ruby also contending for the roost.
* Python is a unique and powerful language, and it appeared everywhere – from scalable web servers that run uninterrupted ‘round-the-clock’, to throw-away scripts that only see a few second of ‘daylight’.
* It can be used for both database and GUI programming, as well as both server-side and client-side programming.
* It is one of the most versatile of its kind.
* It allows for multiple programming paradigms, such as object-oriented, functional, and imperative programming. There is also an automatic memory management and a substantial, comprehensive, and standard library.

Python Installation.

This is an easy step to do with ‘Click and Run’

* First, go to this link <https://www.python.org/downloads/source/>.
* Download the version of the Python that you need.
* Unzip the downloaded file
* Install the file
* Go to Shell or Terminal
* Input command *$ python --version*
* And then you’re all set!

Or else, you can refer to this link. <http://docs.python-guide.org/en/latest/starting/install/linux/> for more detailed instruction.

# Django Setup

Getting to know Django.

What is Django Framework?

* is a free and open source web application framework, written in Python. A web framework is a set of components that helps you to develop websites faster and easier.

Django Installation.

This is a bit complex way of installing the framework. However, we are going to go through the recommended way.

* First, install *pip* in the Shell prompt

|  |
| --- |
| $ pip install -U pip |

* Install a virtual environment and activate it.

|  |
| --- |
| $ pip install virtualenv |

* Install Django

|  |
| --- |
| $ pip install Django |

Or else, you can refer to this link. <https://docs.djangoproject.com/en/1.9/topics/install/> for more detailed instruction.

# Requirement Files

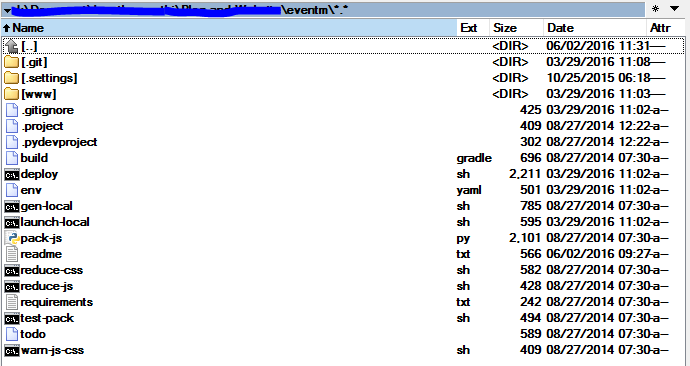
To setup our project, first we need the minimum requirement files.

|  |  |
| --- | --- |
| No. | Subject |
| 1 | Sport Management System Source File. (eventm directory) |
| 2 | Google Appengine SDK downloaded file from <https://cloud.google.com/appengine/downloads> |
| 3 |  |

Note\*: These files are recommended to place inside ‘Home’ rather then ‘Desktop’. And these files should be in parallel directory.

# Setting Up Project in Local System

First, we need to check our source code to ensure that we have enough files without missing a single one.



**After that, we should start setting up a separate virtual environment for our system with virtualenv and virtualenvwrapper.**

First, go to shell prompt, or terminal in linux.

(picture)

Second, go to ‘Home’

|  |
| --- |
| $ cd .. |

Installing virtualenv in our eventm directory

|  |
| --- |
| $ cd eventm  $ pip install virtualenv |

(picture)

Installing virtualenvwrapper in our eventm directory

|  |
| --- |
| $ pip install virtualenvwrapper |

(picture)

Check if our virtualenvwrapper is properly installed. In the command, type in mkvirtualenv and then press double taps to see if the command is found or not.

(piture)

If the command is not found. And so you should go to your Home and find file name ‘.bashrc.sh’, and at the end of the code, add source of the virtualenvwrapper.sh and export it like how it is shown below.

|  |
| --- |
| #.bashrc file  …  export WORKON\_HOME=$HOME/.virtualenvs  source /usr/local/bin/virtualenvwrapper.sh |

(picture)

Save the file and then restarted your terminal or shell prompt or you can run the .bashrc file inside your terminal or shell prompt.

|  |
| --- |
| $ . .bashrc |

And then do the checking again if the mkvirtualenv exist or not. If the commanded is still not found, please check the instruction again or go to <http://stackoverflow.com/questions/12232421/virtualenvwrapper-commands-arent-working>

(picture)

Next, go to our eventm directory again and create virtualenv for our project.

|  |
| --- |
| $ mkvirtualenv eventm |

(picture)

After that, we need to install or update the necessary requirement.

|  |
| --- |
| $ pip install -r requirements.txt |

(picture)

Afterward, we need to convert the less code into css code using node-less

|  |
| --- |
| $ sudo apt-get install node-less |

(pciture)

With this, you are already finished setting up the virtualenv for our project.

**Working on or activate the project.**

First, go to shell prompt, or terminal in linux.

(picture)

Second, go to ‘eventm’ directory

|  |
| --- |
| $ cd eventm |

Next, to work on the virtualenv that we have created.

|  |
| --- |
| $ workon eventm |

(picture)

After that, run gen-local.sh file inside the eventm directory

|  |
| --- |
| $ ./gen-local.sh |

(picture)

Before getting into the last steps, we need to modified some code inside the file named launch-local.sh.

|  |
| --- |
| …  cd ..  cd google\_appengine/  ./…… |

(picture)

After that, run launch-local.sh file inside the eventm directory to test

|  |
| --- |
| $ ./launch-local.sh |

(picture)

With that, your project should be able to up and running. Do not close your terminal, open a new browser, and type in the address as below.

|  |
| --- |
| Localhost:8080 …. /\* This is the main website\*/  Localhost:7000 …./\* This is for the database\*/ |

# Reference

Our documentation are made using various reference from:

|  |  |
| --- | --- |
| No. | Reference |
| 1 | <http://stackoverflow.com/questions/12232421/virtualenvwrapper-commands-arent-working> |
| 2 | <https://www.jetbrains.com/help/pycharm/2016.1/creating-virtual-environment.html> |

# Revision Log

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Change Reference | Reviewed by |
|  |  |  |  |
|  |  |  |  |

# Approval

This document has been approved as the official Business Requirements Document for the project.

ODOO HR Implementation. Changes will be governed by the project’s change management process.

|  |  |  |
| --- | --- | --- |
| Prepared by | Signature | Date |
|  |  |  |
| Approved by | Signature | Date |
|  |  |  |