Operating Systems [2020-2021]

Tutorial – Installing Linux and the required tools

Note: this tutorial will not be done during the practical classes, although questions will be clarified by the teacher.

Introduction

The goal of this Tutorial is to guide the student through the process of installing the Linux operating system, together with all the tools required for the programming classes and assignments, throughout the semester. In order to preserve the student's current machine, the installation will be made using VirtualBox virtualization software.

Installing VirtualBox

Our first goal is to download and install **VirtualBox**, the Virtual Machine manager, available from https://www.virtualbox.org/wiki/Downloads. You should download the installation package most appropriate to your host system and begin the installation process, as illustrated in Figure 1.

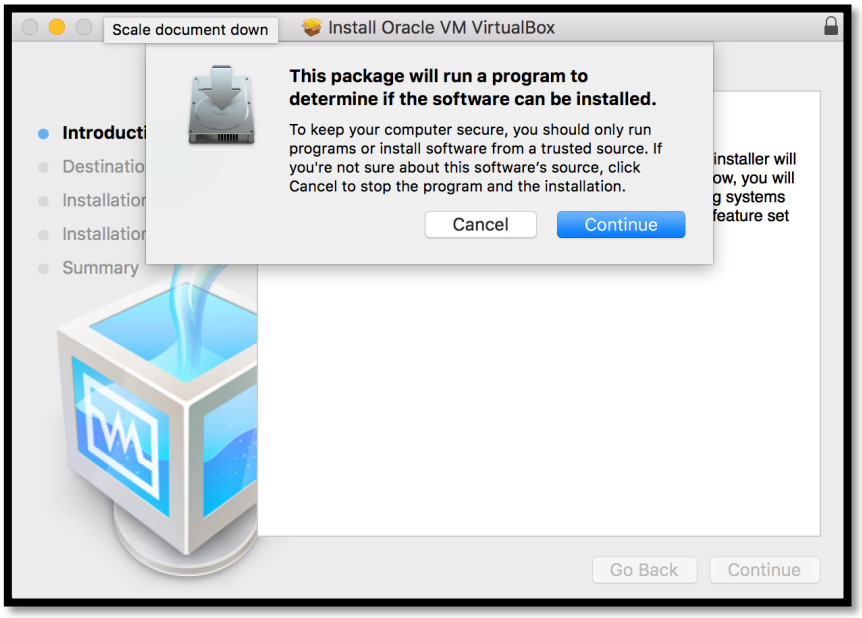

Figure 1 – VirtualBox installation

After installing VirtualBox you should also install the **VirtualBox Extension Pack**, available from https://www.virtualbox.org/wiki/Downloads. The Extension Pack is required to improve integration of the Virtual Machine environment with the host operating system.

Downloading and installing Linux

We next are going to download and install Linux in a new Virtual Machine, using the VirtualBox manager. For this purpose, we begin by downloading an ISO image of a recent version of the Linux Ubuntu operating system (17.04 at time of writing of the assignment), available from:

ftp://ftp.dei.uc.pt/pub/linux/ubuntu/releases/17.04/ubuntu-17.04-desktop-i386.iso

Alternatively, Ubuntu can also be downloaded from the home page of the project, at: https://www.ubuntu.com/download. After having downloaded the ISO image to your computer, create a new Virtual Machine. This process starts by asking information about what type of operating system (and release) we are going to install, and is illustrated in Figure 2.

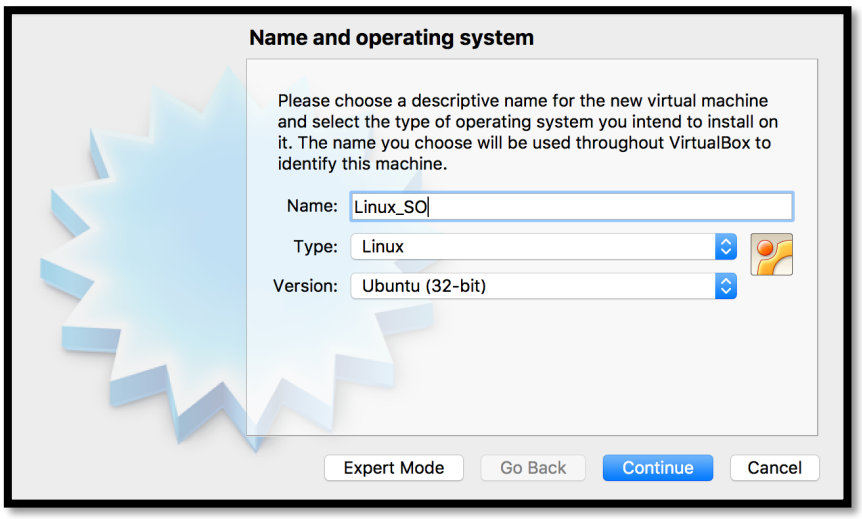

Figure 2 – Creating a new Virtual Machine for Linux

You should allocate the required resources (memory and disk space) to the new Virtual Machine, depending on the availability on your system. This process is illustrated in Figures 3 to 7, and in this example we allocate 2Gb of RAM and 10Gb of disk space to our new Virtual Machine.

Please note: you are advised to allocate at least 7Gb of disk space to the new Virtual Machine and at least 1Gb of RAM (or more, if available).

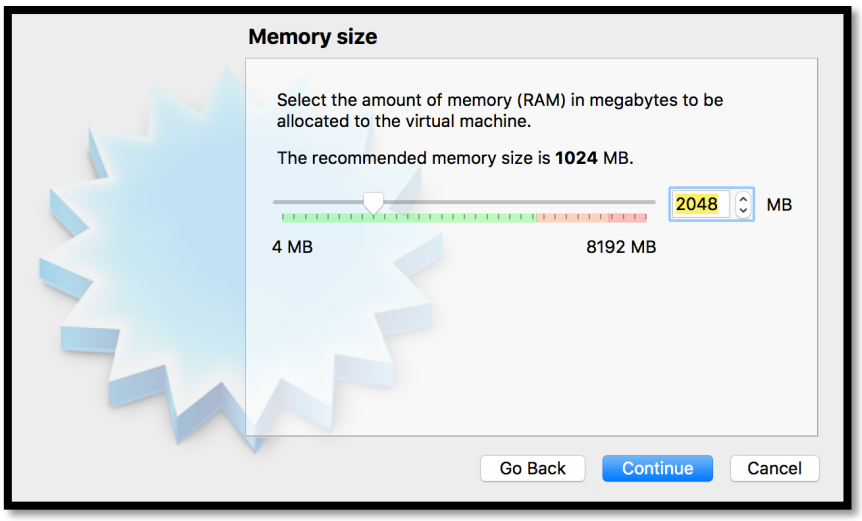

Figure 3 – Configuring RAM for the Linux VM

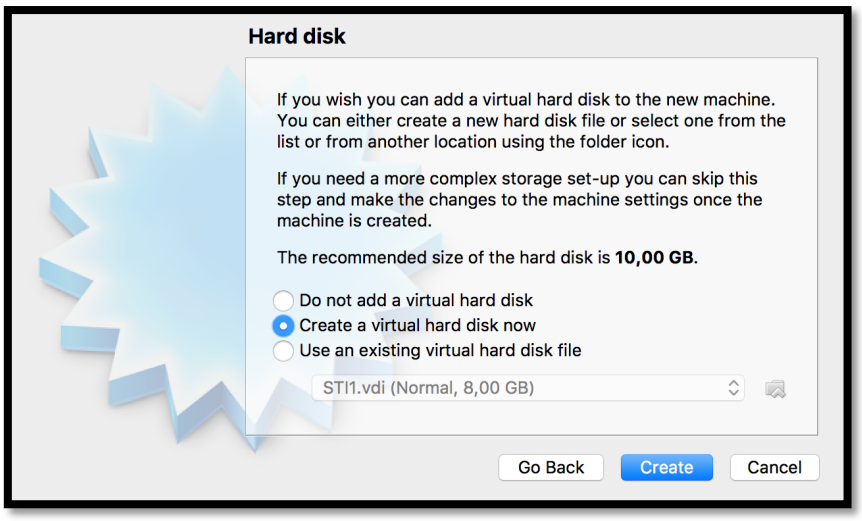

Figure 4 – Configuring the type of Hard Disk

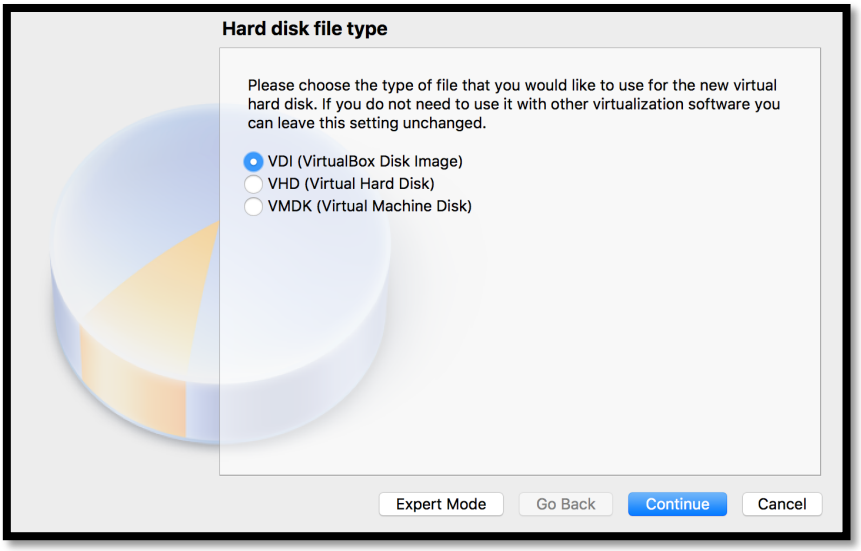

Figure 5 - Configuring the type of Hard Disk (cont.)

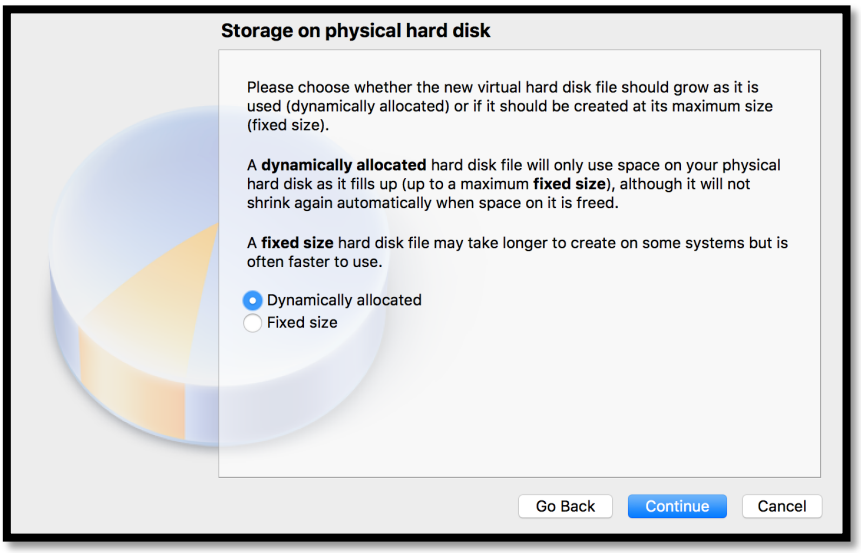

Figure 6 – Configuring storage

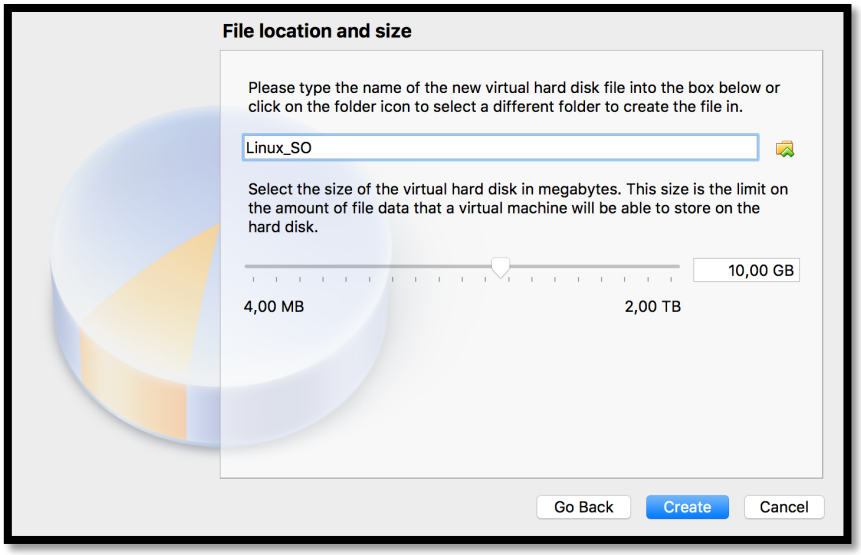

Figure 7 - Configuring disk space

We are now ready to start installing Linux Ubuntu on our just created Virtual Machine in VirtualBox. For this purpose, just boot up the Virtual Machine and locate and select the file containing the ISO image previously downloaded, as Figure 8 illustrates.

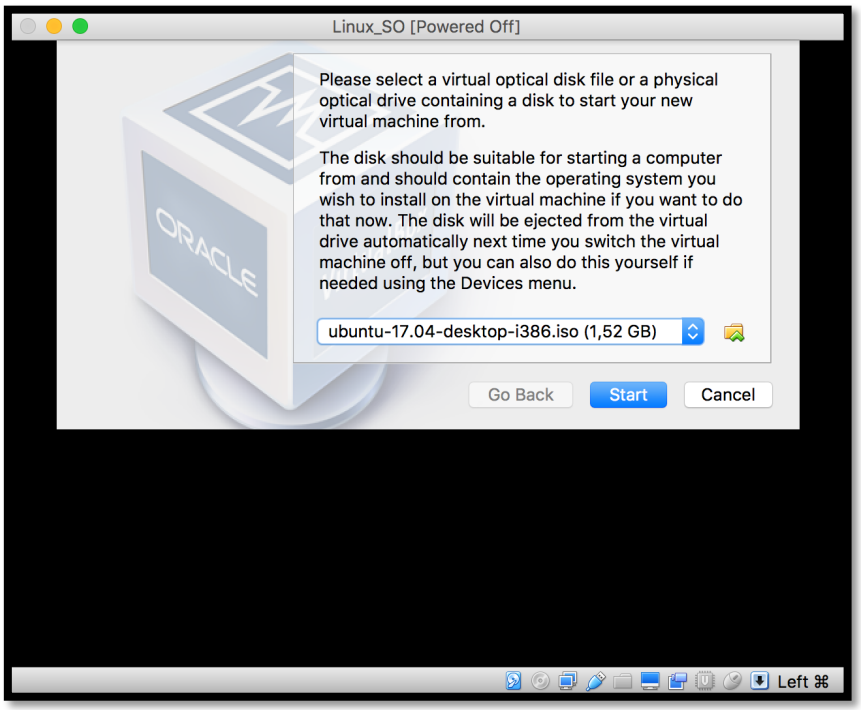

Figure 8 – Selecting the ISO image for the Linux installation

During the installation process, will also be asked for the required information, such as the Location, type of Keyboard and user information. We will also be creating a personal account that you'll use during the semester, as illustrated in Figures 9 to 12.

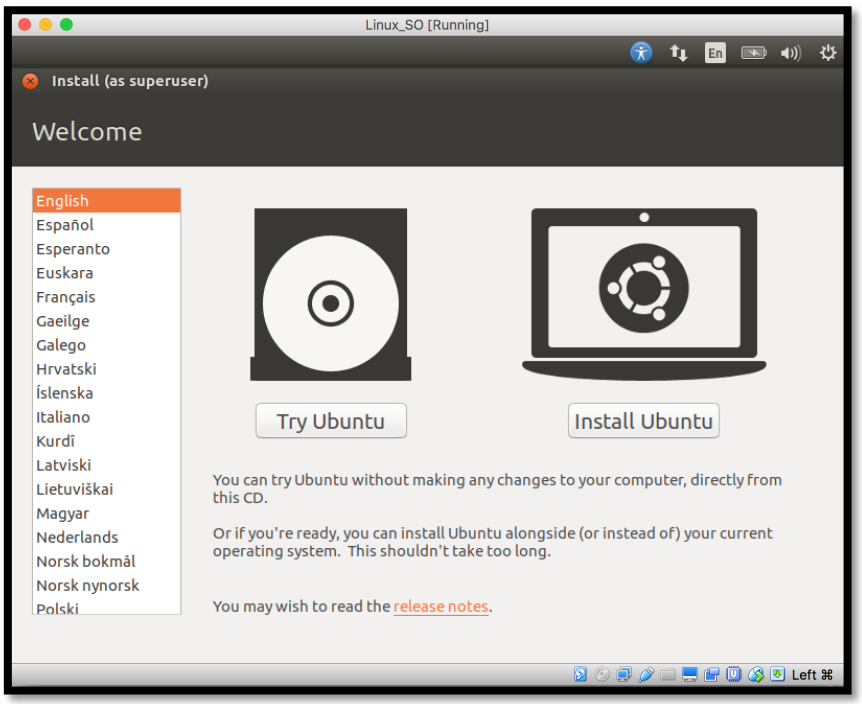

Figure 9 – Start installing Linux Ubuntu

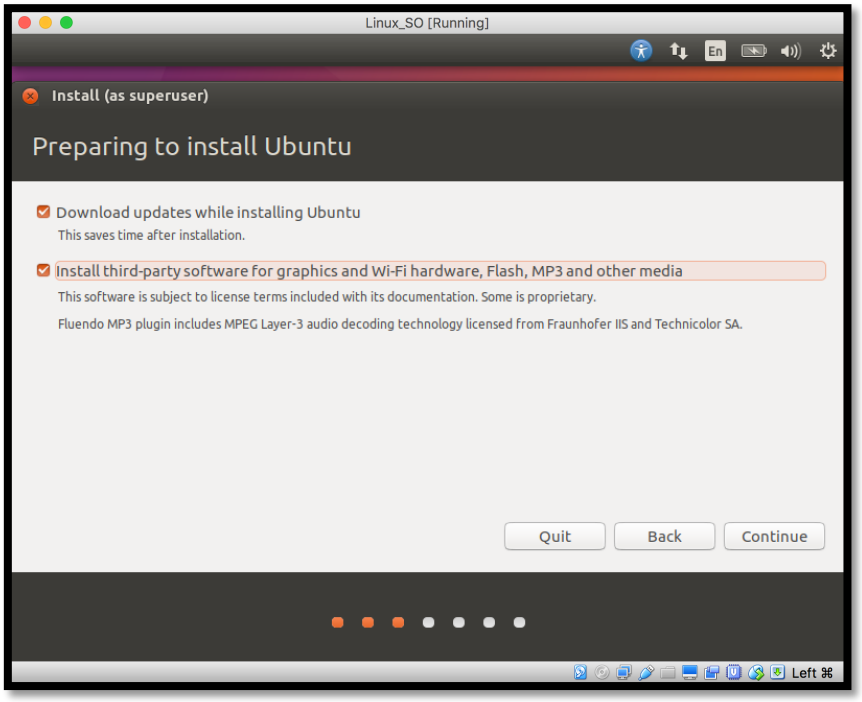

Figure 10 – Download options (during the installation)

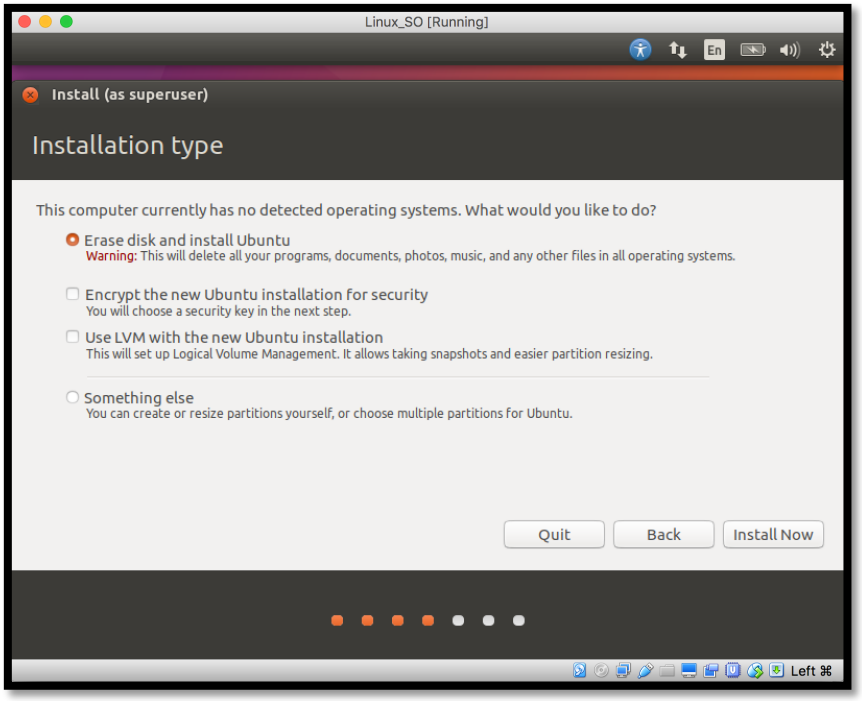

Figure 11 – Selecting the installation type

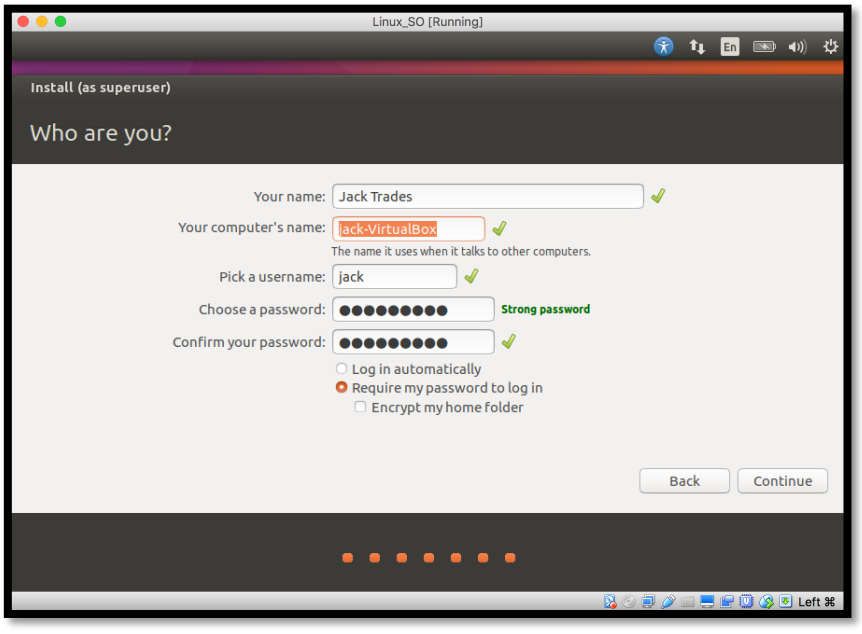

Figure 12 – Creating a (personal) user account on the Linux system

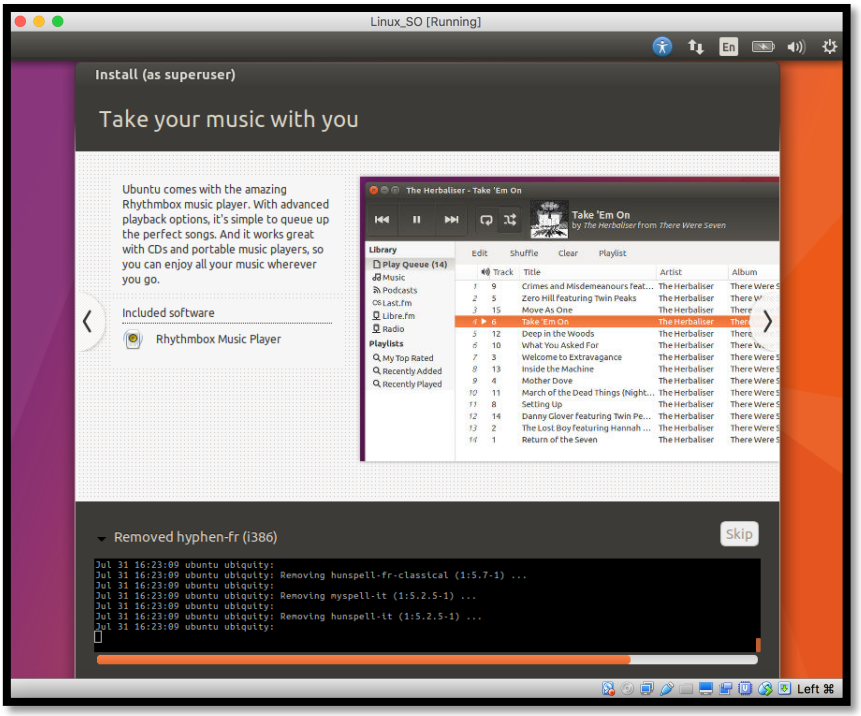

Figure 13 – Installation of the required software

After the installation of the required process (Figure 14), the system will be ready for our first login (using the previously created account), as Figure 15 illustrates.

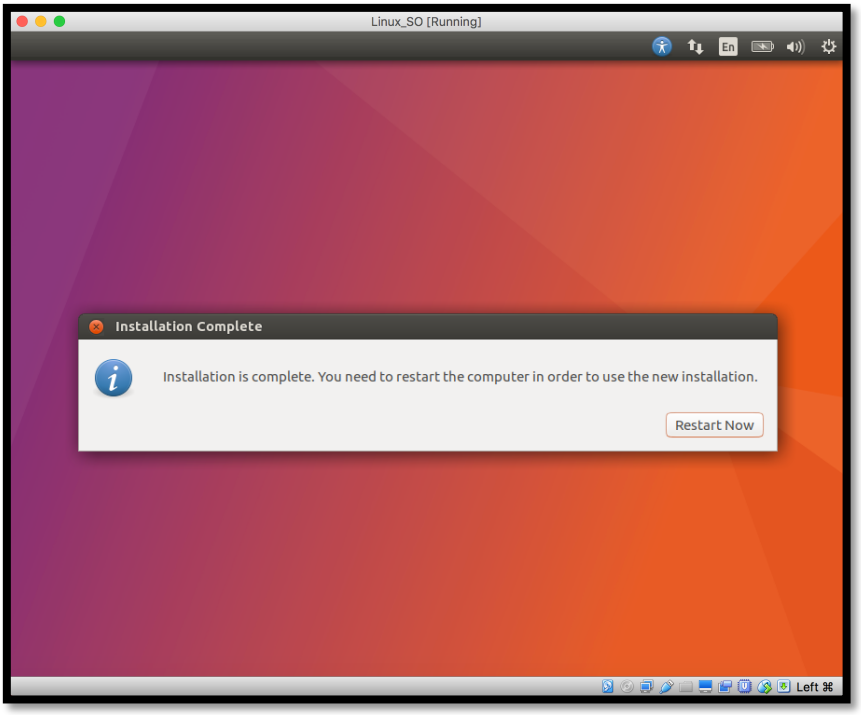

Figure 14 – Conclusion of the installation process

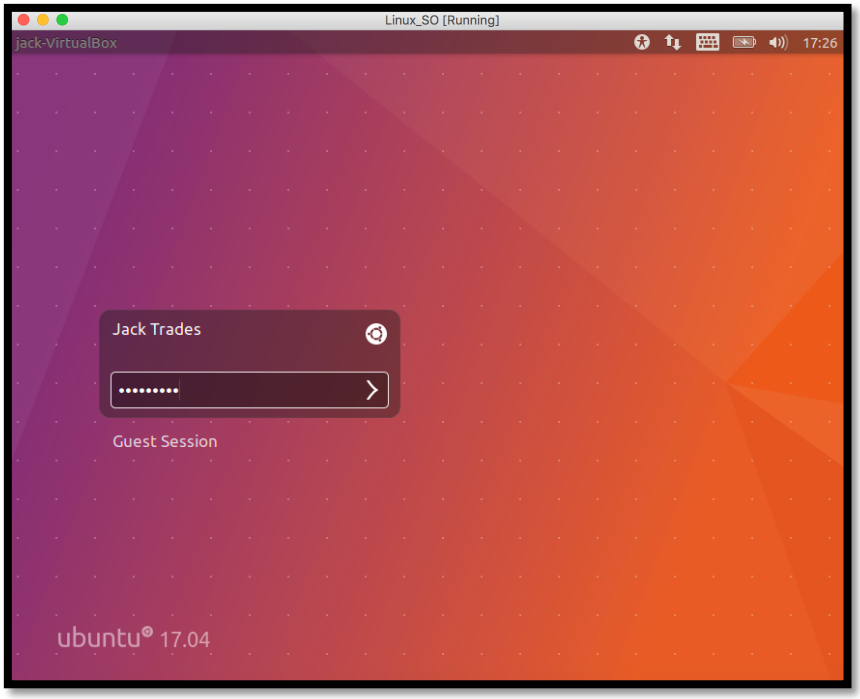

Figure 15 – The Linux system is now available for our first login

As important final installation steps, we should update the system, as illustrated in Figure 16.

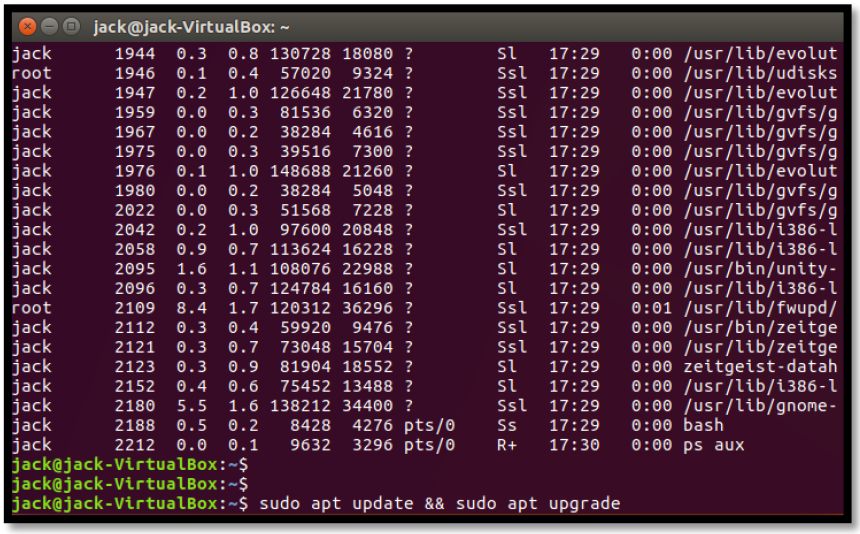

Figure 16 – Updating the system

Bootloader and partitions

A bootloader is the program that loads an operating system when the machine boots up and, in the case of Linux Ubuntu, GRUB2 (**GR**and **U**nified **B**ootloader) is the default boot loader installed. Although the following configurations are not required, they exemplify how the GRUB2 configuration may be changed after the installation of the operating system, as per our previous steps.

All settings related to the configuration of the GRUB2 boot loader are stored in a single configuration file, the **/etc/default/grub** file. As for other configuration aspects in Linux, this file may be edited and changed, in order to fine tune the configuration of the boot loader. Figure 17 illustrates this process, in this case using the *nano* text editor.

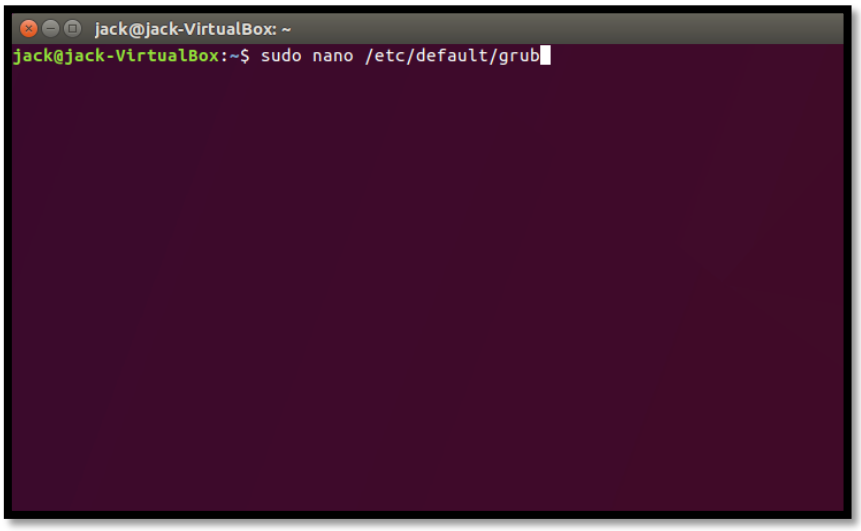

Figure 17 – Editing the GRUB2 configuration file

In Figure 18 we illustrate two configuration settings that are changed for GRUB2: the timeout before GRUB2 boots up the selected operating system (in this example we change this timeout to 20s), and forcing the boot menu to show up at the beginning of the boot process.

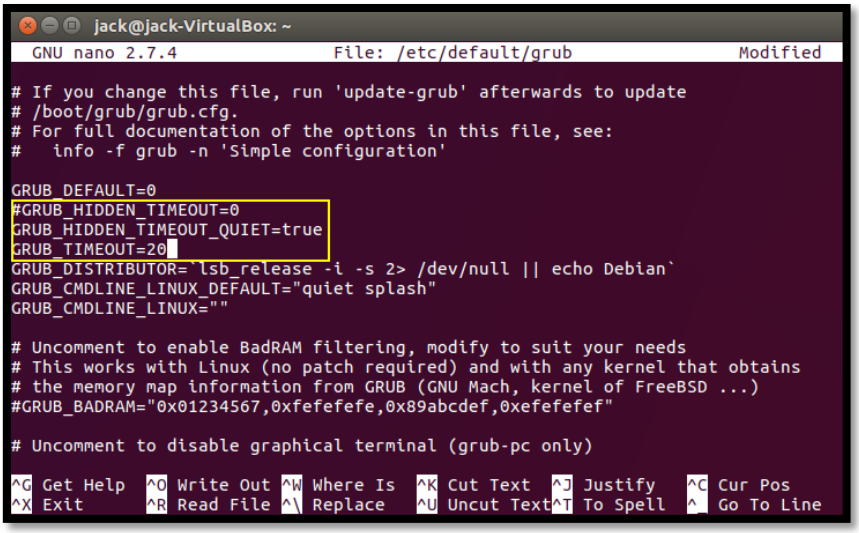

Figure 18 – Some (simple) changes to the GRUB2 configuration

After saving the GRUB2 configuration file, the configuration may be updated by running the command ***sudo update-grub***.

Another fundamental aspect to consider in the configuration of the Linux operating system is that of the disk partitions used by the operating system. In our previous installation of Linux, we have performed an automatic installation and thus the available space (the virtual partition created using VirtualBox) was automatically partitioned and formatted for us. We can analyze the current configuration in terms of the used disk partitions, as illustrated in Figure 19 using the ***fdisk*** command. We may note that, as requested before, the installation has created a Linux partition with 10Gb.

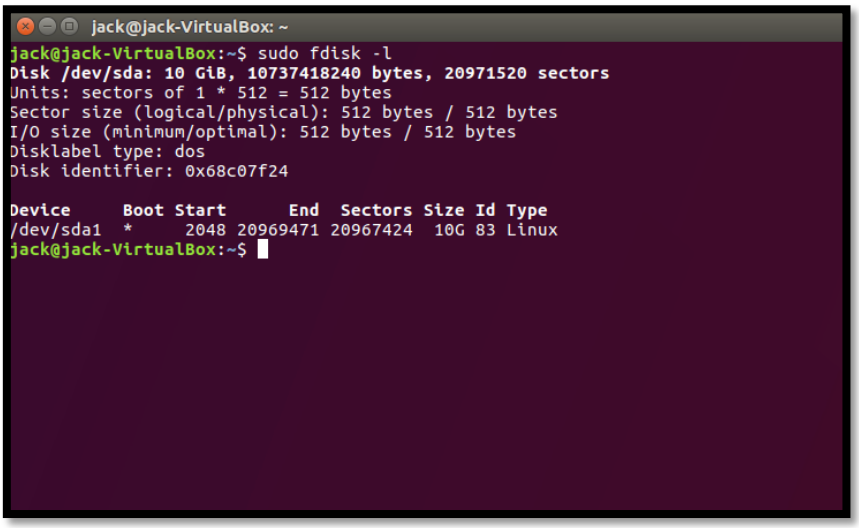

Figure 19 – Available partitions

Another useful command in Linux is ***df***, which allows us to visualize the various mounted partitions, as well as the space available in each partition, as illustrated in Figure 20. As we may observe, the main file system used by the Linux operating system (mounted in the '**/**' root directory from the /dev/sda1 partition) currently occupies approximately 55% of the available 10Gb of disk space.

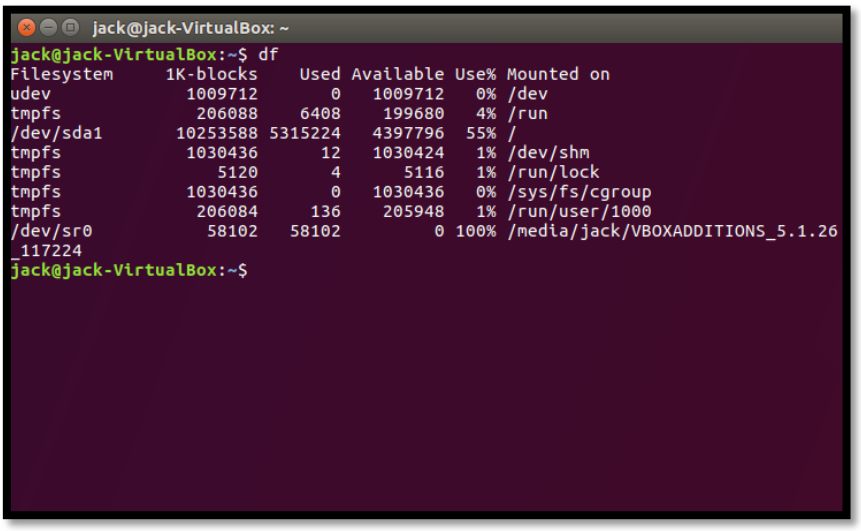

Figure 20 – File system disk space usage

Final configurations

As final configurations, we will install the **Virtual Guest Additions** in Linux (in order to enable integration with the host system), as well as the required **tools for development**, during the semester. Starting with the installation of the Virtual Guest Additions, select the "Insert Guest Additions CD Image" option from the VirtualBox Devices menu and follow the installation process, illustrated in Figures 21 and 22.

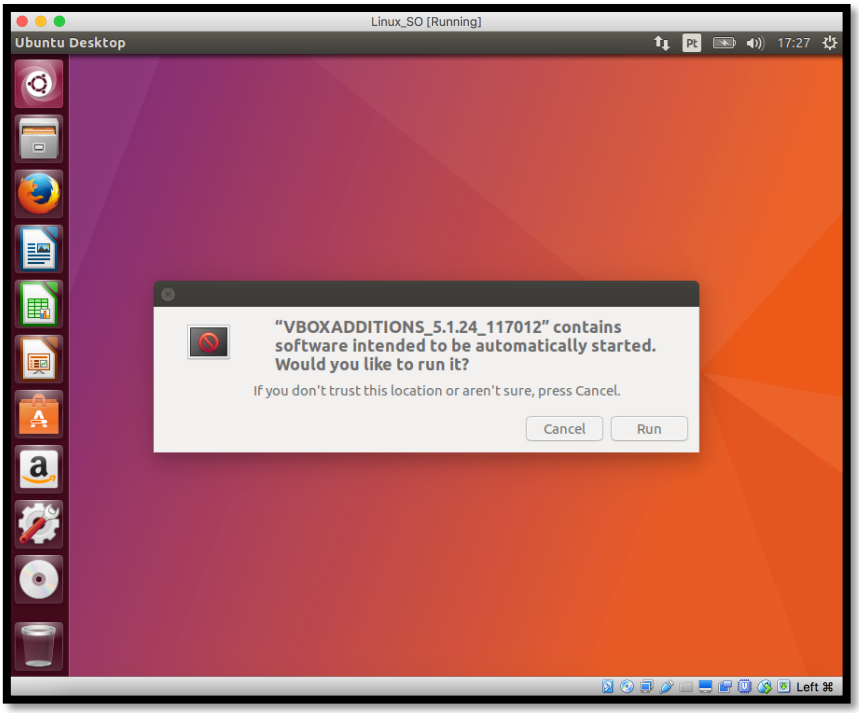

Figure 21 – Installing the VirtualBox Guest Additions

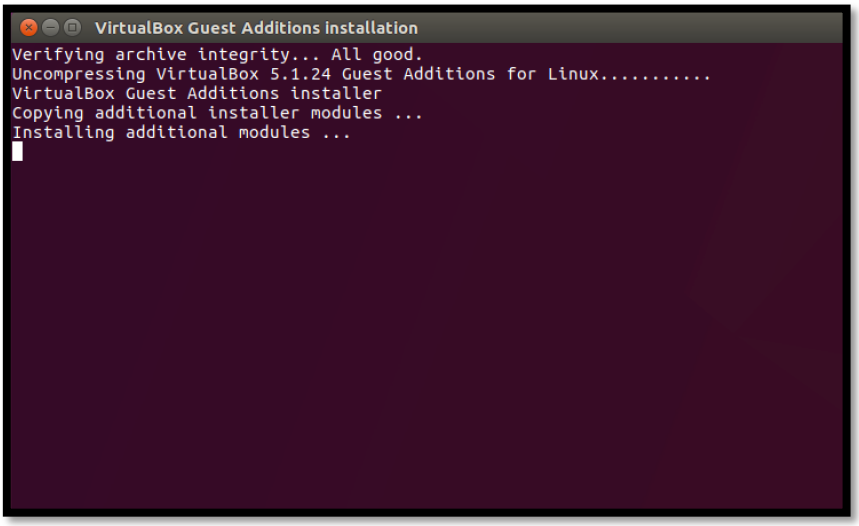

Figure 22 - Installing the VirtualBox Guest Additions (cont.)

Finally, we need to install the required packages for development, during the semester, which are illustrated in the Figure 23.

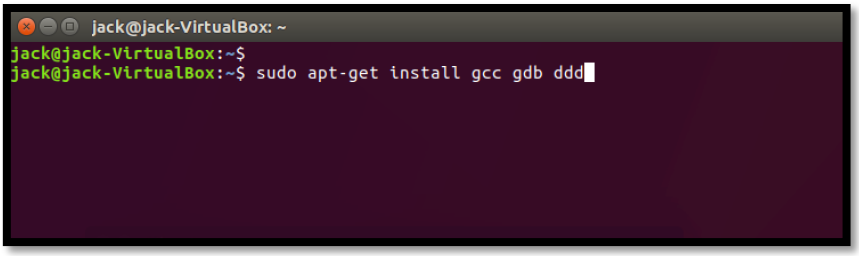

Figure 23 - Installing the required development packages

More information about each of the previously installed packages may be obtained using the **sudo dpkg-query -s** command. In short, **gcc** is the GNU C compiler, **gbd** our source-level debugger and **ddd** a graphical debugger interface to the **bdg** debugger. Finally, please note that further configurations may be required in order to prepare your brand new operating system for using during our classes, in particular in what respects the keyboard layout.