MTRN4230 Robotics GROUP PROJECT

Introduction

This group project is about designing of a pick and place robot manipulator in a simulated environment. Every group/team has seven people involved. The instruction about forming group has been shared in MS Teams *Files/General/GroupProject/GroupProjectMTRN4230.mp4*.

This document outlines the important aspect of this project as well as due dates, marking criteria and deliverables.

Problem description

One of the typical applications for robot arms is performing pick and place tasks in various industrial setups. It involves

1. Detection of an object based on sensor information.
2. Using proper gripper for a robot arm.
3. Actuation of a robot arm towards the target and picking the object
4. Moving to the pre-determined destination location to place the object
5. Placing the object at the destination location

Many components of the above task are adjustable; and based on the type of the object or complexity of the environment, this task can be easy or difficult.

Let us consider an automated warehouse as pharmaceutical product distribution centre (DC) that has automated order and fulfilment systems. The pickers (people or technology used to pick items and collect them to fulfil the order) are robot arms programmed for the pick and place tasks. This warehouse as a modern DC ensures, small orders are processed as fast as possible and the next day delivery is guaranteed for their customers. These commitments make the warehouse very dependant to the automation. Robot pickers are part of warehouse control system (WCS).

Chemists, as the main customers, order the products in small amounts and the next day delivery ensures products are delivered to them on time. This encourages them to order less but more frequent. It reduces the cost of stock accumulation/maintenance and improves the diversity of the products on the shelves without losing serviceability.

How an order is fulfilled

Orders are not coming in bulk but in a smaller amount. So, it means DC needs to operate as a retailer but with a broad range of orders and huge numbers. With all the planning still order fulfilment / picking the products in small amounts and diverse profile is a challenge. The way the stock is maintained in the warehouse is done by warehouse management system (WMS). WMS uses various methods and algorithms to club products and pre-order bulk amounts of them to maintain and replenish the warehouse at the same time WMS ensures the customer orders can be fulfilled based on stock in hand, then the orders are sent to WCS for fulfilment.

One of the approaches WMS/WCS uses is: mixing products in one container to reduce waste of storage capacity i.e. mix tote with dividers. It means one container or tote can hold three products at the same time shown in Figure 1 and it is the responsibility of the picker to pick from a correct section within a tote based on the order details provided.

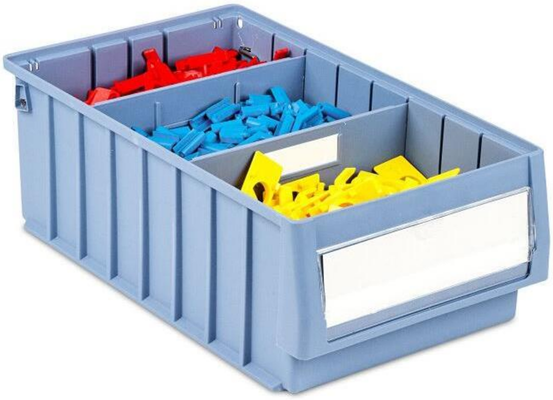


Figure 1 Tote with dividers (https://www.alibaba.com/product-detail/Industrial-Plastic-Storage-Tote-Bins-with_60631536472.html)

Regarding the storage of mix totes, an automated storage and retrieval system is used to store and retrieve totes from the racks as shown in Figure 2. The totes to be stored are fed either from the decant lines or totes that already used to pick items from. Retrievals are done to send totes to robot pickers to fulfil the orders.

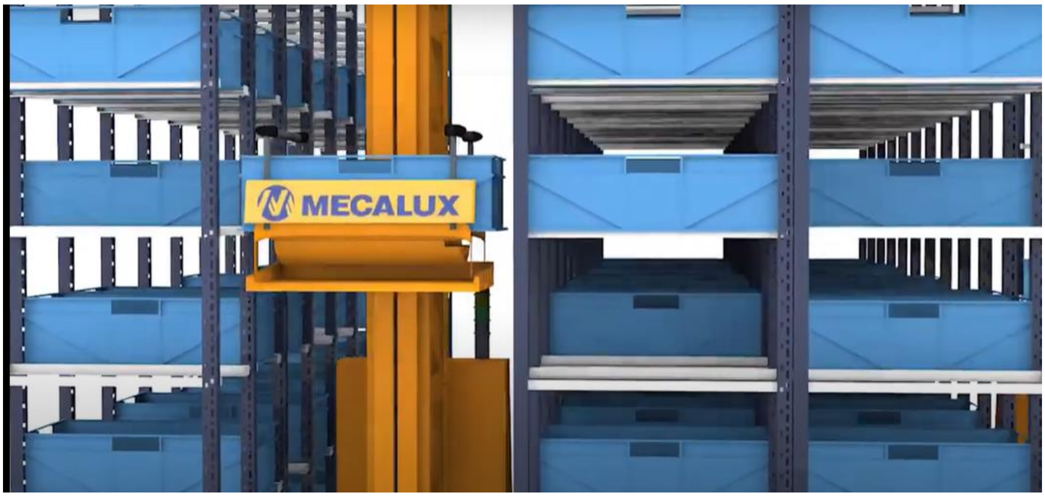


Figure 2 Rack storage system (https://www.interlakemecalux.com/warehouse-logistics-videos/miniload-automated-warehouse-boxes)

Picker robots

Picker robots are used as shown in Figure 3 in an illustrative setup. The robot receives a tote containing products, it picks items based on the order details sent to the robot and places them in another container that finally will be shipped to the customer.

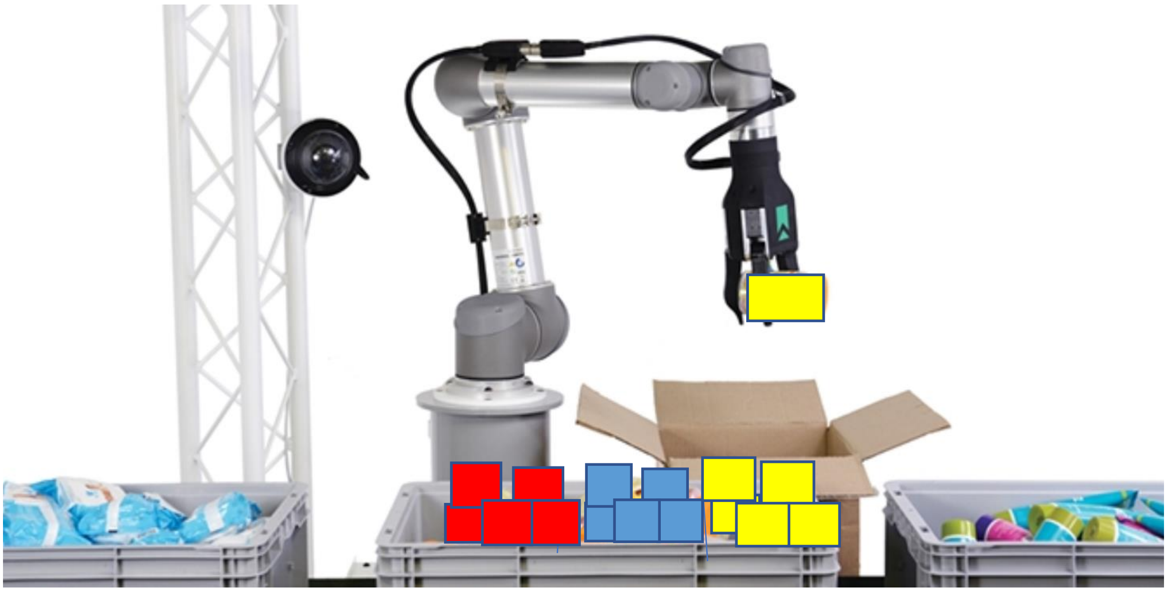


Figure 3 picker robot arm (https://www.roboticstomorrow.com/images/articles/12618.jpg)

As shown in Figure 3, the robot uses a vision system to identify the item e.g. yellow box need to be picked. The gripper is actuated, and specific number of yellow items are picked based on the command from WCS.

Task outline

1. Research about the implementation of pick and place task by a robot arm and software architecture of such a system
   - It is expected that the team first comes up with a design by doing some research. It is important to decide about the main structure of your codebase, the algorithm, and main components. You can use the information in this document and information shared with you during the tutorial to find your path. The goad is: you come up with ideas! It can be very simplified or very complex.
     a. The pick and place task should be done by MATLAB/Simulink and ROS as the minimum requirement. If you would like to do it fully in ROS it is okay. However, the programming language in ROS should be Python.
     b. You do not need to have a complex simulation. The minimum requirement is defined here:
       - One robot arm UR5
       - One RGB-D camera (Kinect)
       - Picking stationary objects (not moving objects)
       - Detection of objects based on classifier and/or image processing techniques
       - Detection of objects based on shape of the object
       - Detection of objects based on colour of the object
       - Detection of objects based on colour and shaper of the object
       - Using depth data from RGB-D (Kinect) in the pick and place task

- Location of objects before pick should be randomised when the program starts but the shape and colours are pre-determined
- Minimum 10 objects need to be picked and placed
- Text-based or the graphic user interface to
  - select number of picks robot need to do and
  - option to select colour and/or shape of the objects to pick as the goal
- Robot movements should be based on online information coming from simulation and calculated not pre-defined or pre-determined movements or path. Only one predefined HOME position is allowed and the destination for placing the order is pre-determined

2. Formulate forward and inverse kinematics in MATLAB for the robot arm
3. Trajectory planning for robot manipulator
4. Design of object detector with a classifier based on RGB and depth data (RGB-D sensor)
5. Completing the design of the simulation platform in ROS as an external simulator using a UR5 robot
    - ROS is used to create an external simulator. Ros is supplied as virtual matching with basic component, so no need to install ROS or initialise the environment. However, the provided model components and controllers need to be updated to cater for the pick and place task. No Python or C++ programming in ROS is expected/required however models in Gazebo need to be added/updated to support your design using Unified Robotic Description Format (URDF) that is an XML file format.

    Note: Gripper model and gripper controller are not provided. It is expected that the team selects a gripper and sets up a controller in ROS however no need to create a gripper from scratch, using existing models (can be found online) is allowed to minimise programming in ROS. The gripper type can be a suction or fingers or any other type depending on the design you will decide.

6. Simulate pick and place task.
    - Simulation software MATLAB/Simulink and ROS are used
7. Teamwork and workload distribution
    - teamwork and continuous communication among group members matter and it is pre-requirement for marking
        - Use your MS Team group for communication instead of emails to have a good track of team interactions. It is needed for marking and tracking purpose
    - fair workload distribution
        - Team needs to show how every team member is actively helping the team has a task and is sharing part of the load in a fair manner

    Note This semester most of us did not have opportunity to meet and team up. We do not have the opportunity to find right people for our team and select our teammates. However, we can show the way we utilise tools and available resources in a smart way to compensate for the physical distance. In addition, it's a typical example of the real-life that we cannot always select our teammates, e.g. you join a company with an existing team, we can practice how to make most of it based on what we know and what we have.

8. Project management
    - There is no expectation for comprehensive project management since it is not project management course, however, it is expected that teams are following a path and is mindful of the deadline and the progress with the basic planning and tracking tools or documents, for example, teams need to have
        - Project Gantt chart
        - Project planning and task allocation
        - Budgeting time and resources

Checkpoints and deliverables

Checkpoint	Due date	Deliverables	Mark
1	W7	○ Showing the team has formed, MS Teams created ○ Showing the project plan and Gantt chart have been prepared and are tracked ○ Role of every person in the team is clearly defined and described ○ Evidence of continues communication ○ Environment for simulation and programming is prepared ○ Simulation model in ROS is progressing	5%
		○ Short report (1%) ○ Video from the model and environment as far as it is ready. i.e. Demo of what you have done so far (2%) ○ Project planning documents (1%) ○ Code (1%) ○ Team communication evidence (no mark but without this item, this checkpoint is considered 0)	
2	W8	○ Simulation model in ROS is ready. Objects can be generated; models are working, and image data can be read from RGB-D ○ The robot can do simple movements by sending commands ○ Path planning works and forwards and inverse kinematic works	5%
		○ Short report (1%) ○ Video of the working system as far as it got i.e. Demo of what you have done so far (2%) ○ Code (2%) ○ Team communication evidence (no mark but without this item, this checkpoint is considered 0)	
3	W11	○ The image processing part is working. ○ Object detection is ready. ○ The user interface is ready to select the target shape or colour and number of picks ○ Depth data can be analysed and fed to the robot controller ○ Robot movement is controlled with detected objects and pick and place task	5%
		○ Short report *(1%)*	

		○ Video of the working system as far as it got i.e. Demo of what you have done so far *(2%)* ○ Code *(2%)* ○ Team communication evidence (no mark but without this item, this checkpoint is considered 0)	
Final	W13	○ The gripper is ready and attached to the robot ○ Pick and place tasks with initialisation from the user interface works and camera data is processed online, and the robot moves accordingly to pick and place 10+ objects with define goal for the system.	15%
		○ Final comprehensive report it should collect and conclude the previous reports (3%) ○ Demo of the result with running different scenarios as video or interactive video (7%) ○ Code (4%) ○ Feedback about the overall experience with the project including mistake the team did as lessons learnt or note of smart moves or decisions that helped you to progress faster than you expected (1%) ○ Team communication evidence (no mark but without this item, this checkpoint is considered 0)	

Getting started

1. Form your group
2. Create MS Teams group and add the lecturer and tutors
3. Start setting up the simulation and programming environment
4. Create your team structure and project planning
5. Capture team skills and allocate time and tasks to people
6. Organise roles for the team members including, project manager, ROS specialists, Robot specialists, Image processing/computer vision specialists and so on. It should not be *everyone doing everything*!
7. Download the VM shared with you
8. **Follow the tutorial video to run ROS and MATLAB**

Assessment overview

Item	***Group Project?*** *(# Students per group)*	***Weight***	***Learning outcomes assessed***	***Assessment criteria***	***Due date and submission requirements***	***Deadline for absolute fail***	***Marks returned***
Group project checkpoint 1	*Yes (up to 7 people)*	*5%*	*1,3,4,5*	*Upload r intermediate report + video + code*	*Week 7: 11:59pm, Friday via Teams*	*Week8:11:59pm, Wed.*	*1 week after submission*
Group project checkpoint 2	*Yes (up to 7 people)*	*5%*		*Upload intermediate report + video + code*	*Week 8: 11:59pm, Friday via Teams*	*Week9:11:59pm, Wed.*	*1 week after submission*

Group project checkpoint 3	*Yes (up to 7 people)*	*5%*		*Upload intermediate report + video + code*	*Week 11: 11:59pm, Friday via Teams*	*Week12:11:59pm, Wed.*	*1 week after submission*
Group project final checkpoint	*Yes (up to 7 people)*	*15%*		*Upload final report + video + code*	*Week13: 11:59pm, Friday via Teams*	*Week14:11:59pm, Wed.*	*1 week after submission*

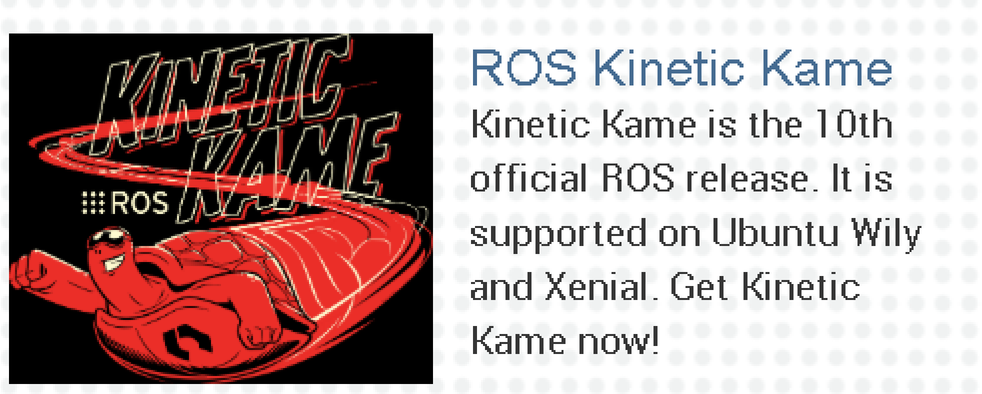
How to use Prepared ROS Kinetic Environment

If you want to avoid setting up your environment, you may start using the VM shared with you to speed up. Some of the main packages.

- Ros Kinetic [http://wiki.ros.org/kinetic/Installation/Ubuntu]
- Catkin [http://docs.ros.org/melodic/api/catkin/html/user_guide/installation.html]
- MoveIt [https://moveit.ros.org/install/]
- Universal Robot [https://github.com/ros-industrial/universal_robot]
- Ur5_ROS-Gazebo [https://github.com/lihuang3/ur5_ROS-Gazebo]

What is ROS

"The Robot Operating System (ROS) is a set of software libraries and tools that help you build robot applications. From drivers to state-of-the-art algorithms, and with powerful developer tools, ROS has what you need for your next robotics project. And it's all open source." [ros.org]



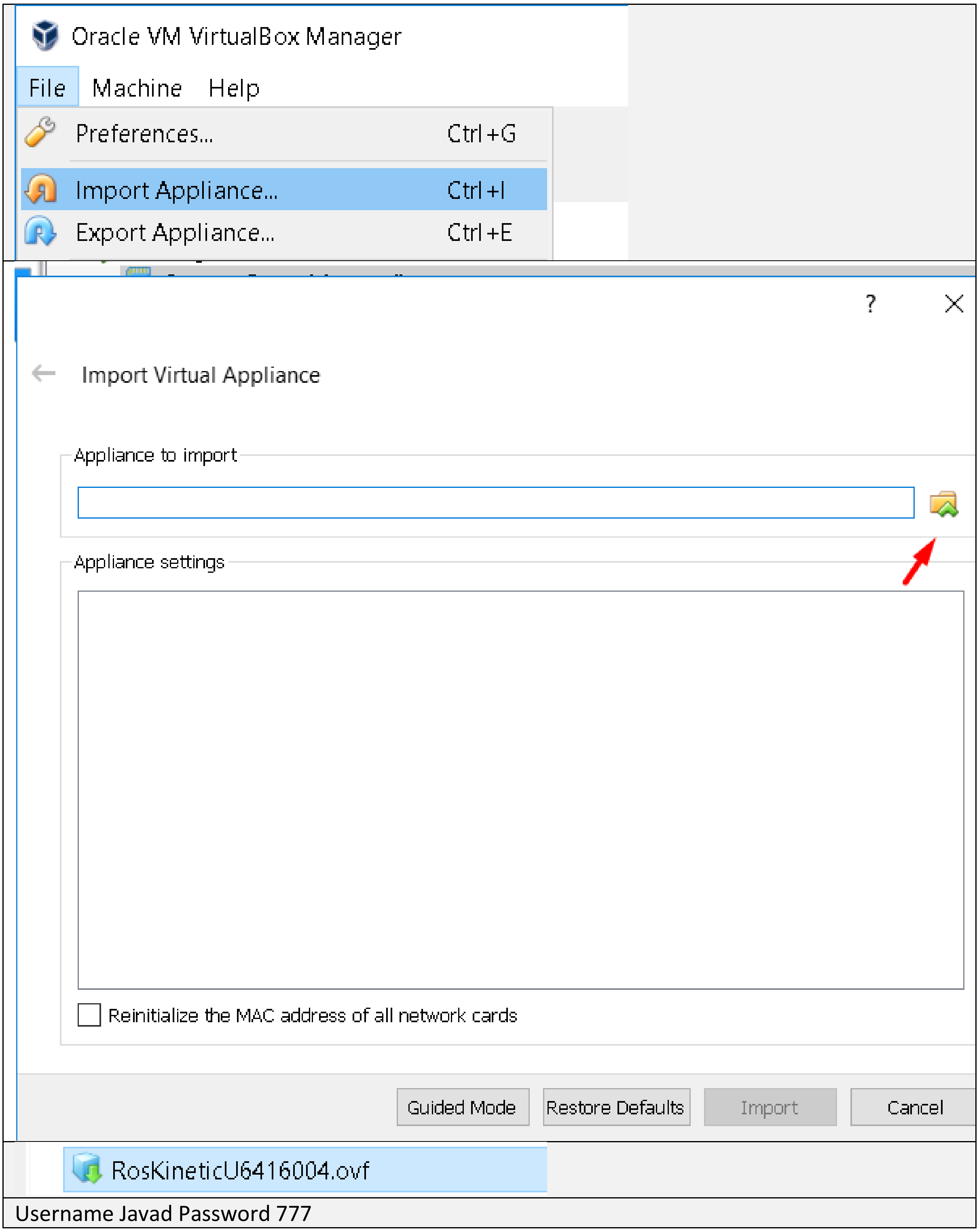
ROS Kinetic Kame

Kinetic Kame is the 10th official ROS release. It is supported on Ubuntu Wily and Xenial. Get Kinetic Kame now!

Run the setup

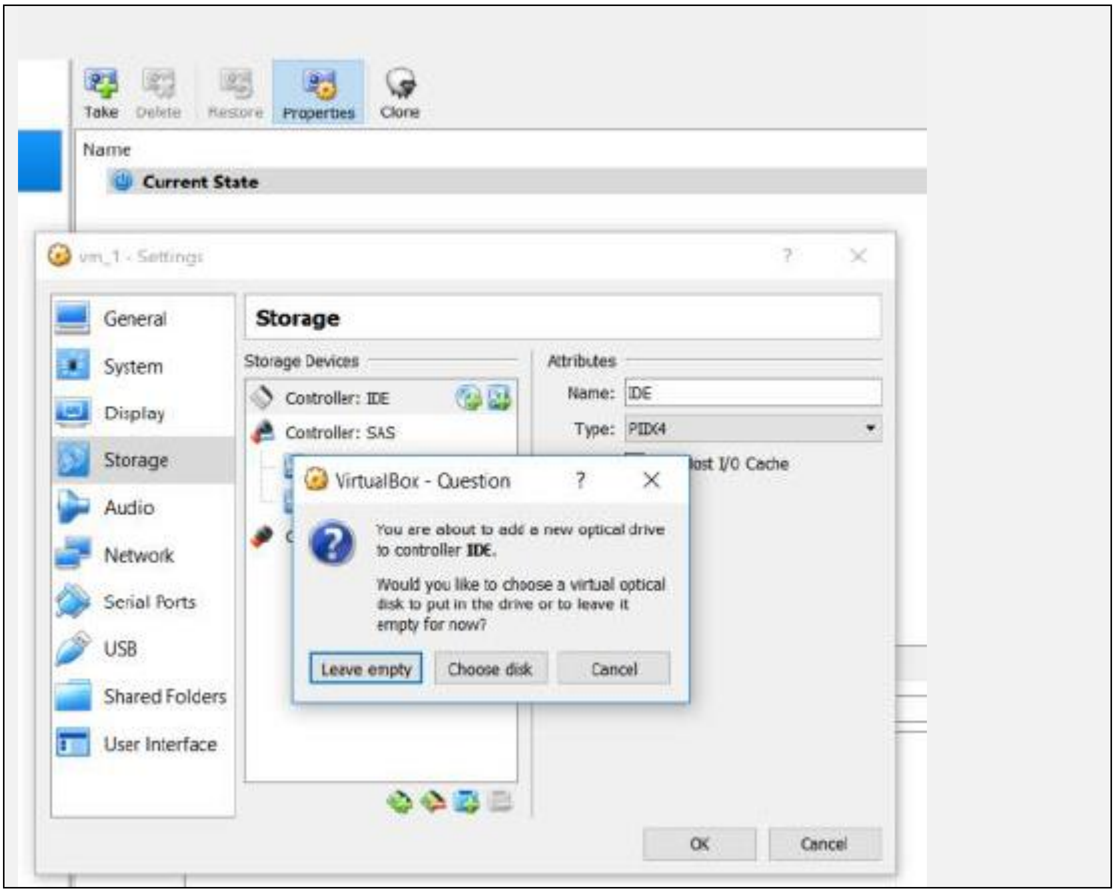
VirtualBox-5.2.22-126460-Win.exe

How to import image in virtual box



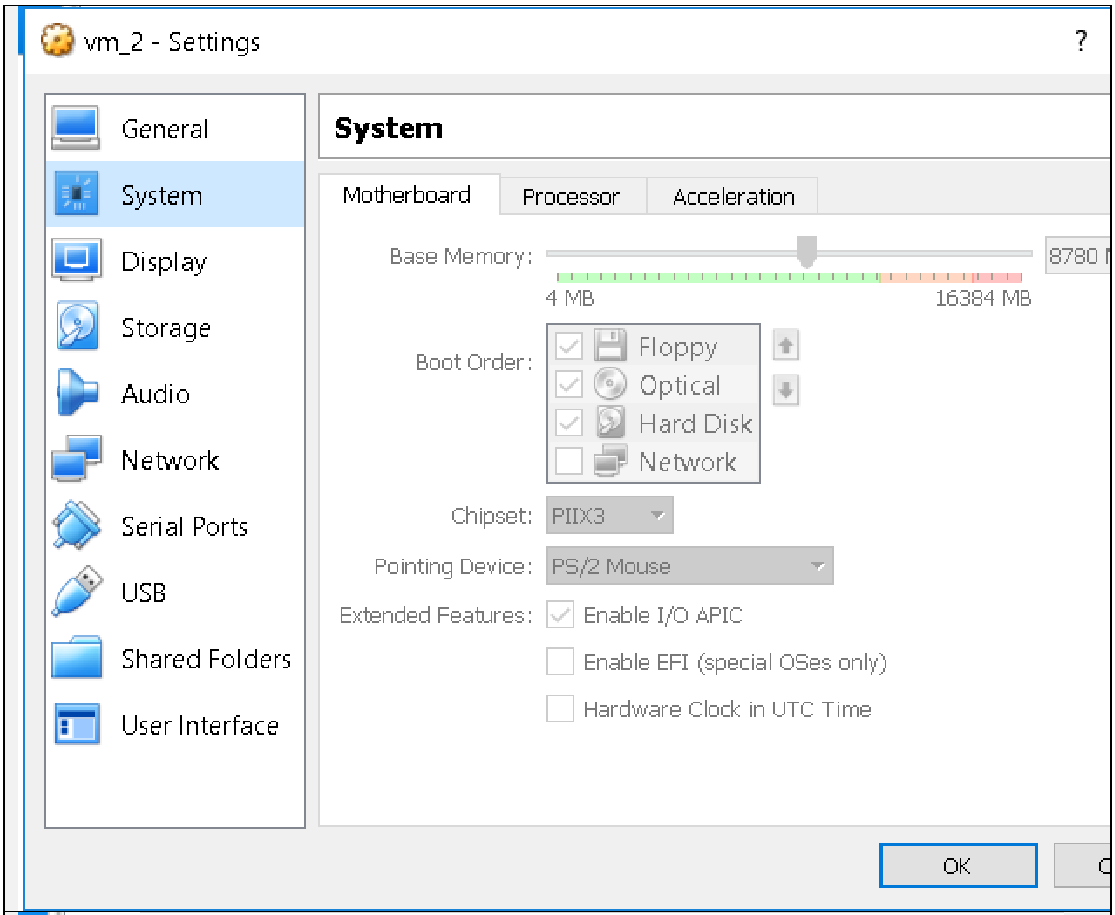
Username Javad Password 777

Install Virtual Box Guest Additions



You are about to add a new optical drive to controller **IDE**.

Would you like to choose a virtual optical disk to put in the drive or to leave it empty for now?



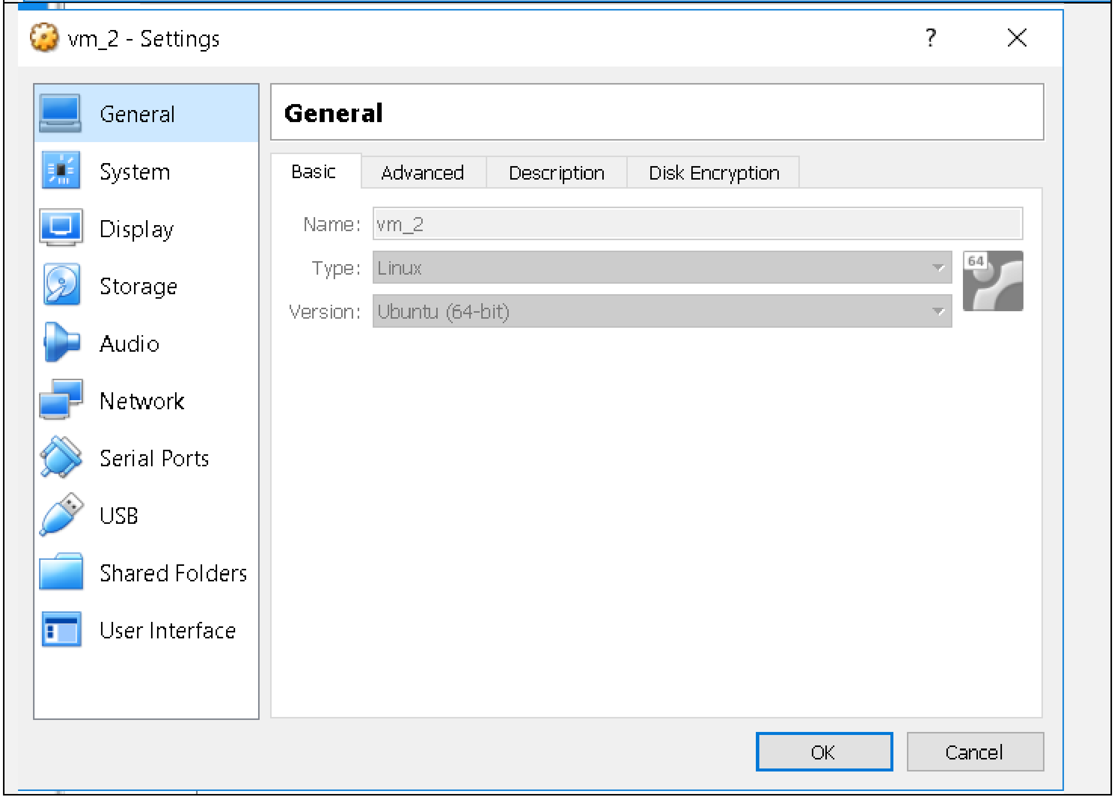
Base Memory: 8780 M

Boot Order:
☑ Floppy
☑ Optical
☑ Hard Disk
□ Network

Chipset: PIIX3

Pointing Device: PS/2 Mouse

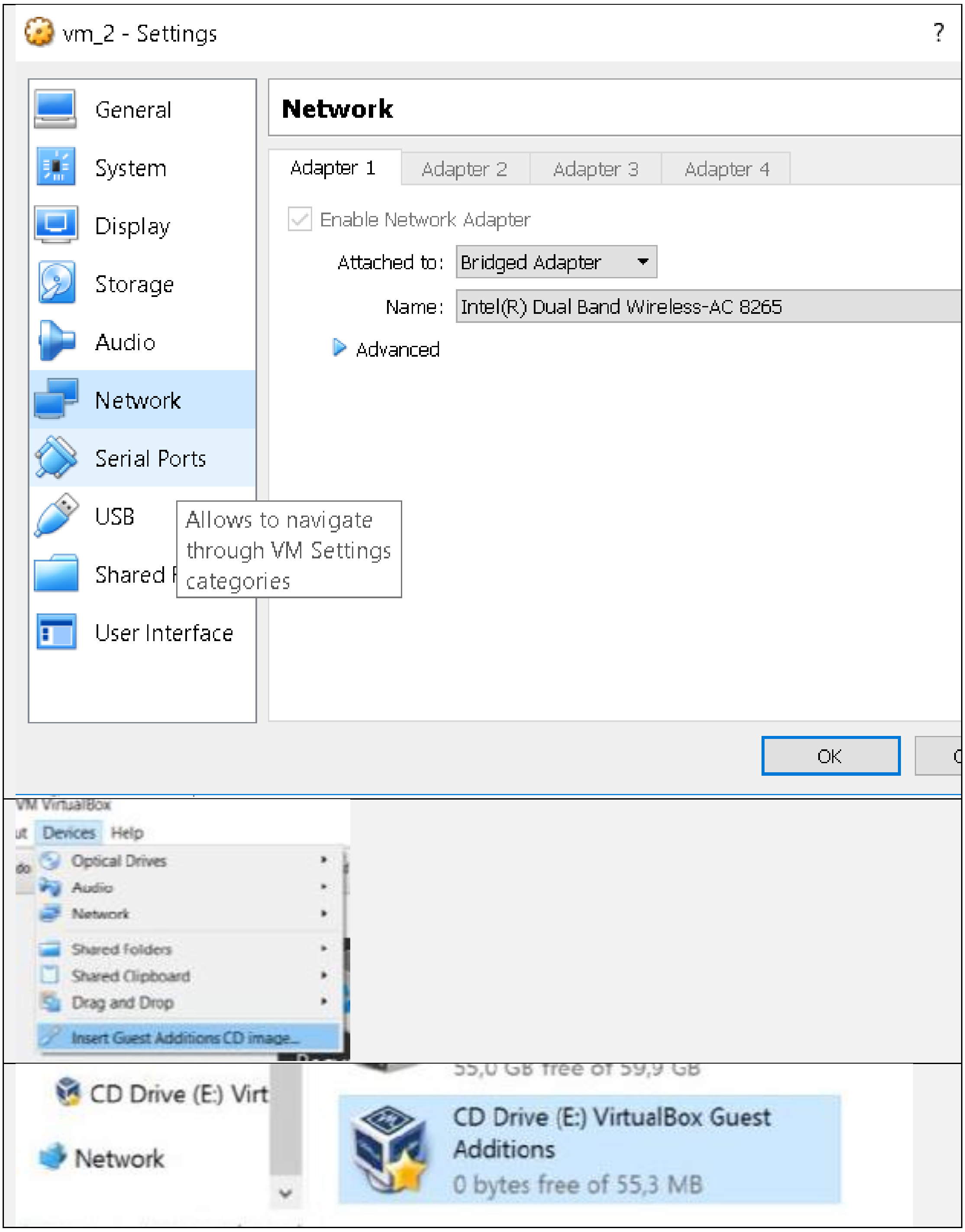
Extended Features:
☑ Enable I/O APIC
□ Enable EFI (special OSes only)
□ Hardware Clock in UTC Time



Name: vm_2

Type: Linux

Version: Ubuntu (64-bit)



vm_2 - Settings

Network

Adapter 1

☑ Enable Network Adapter

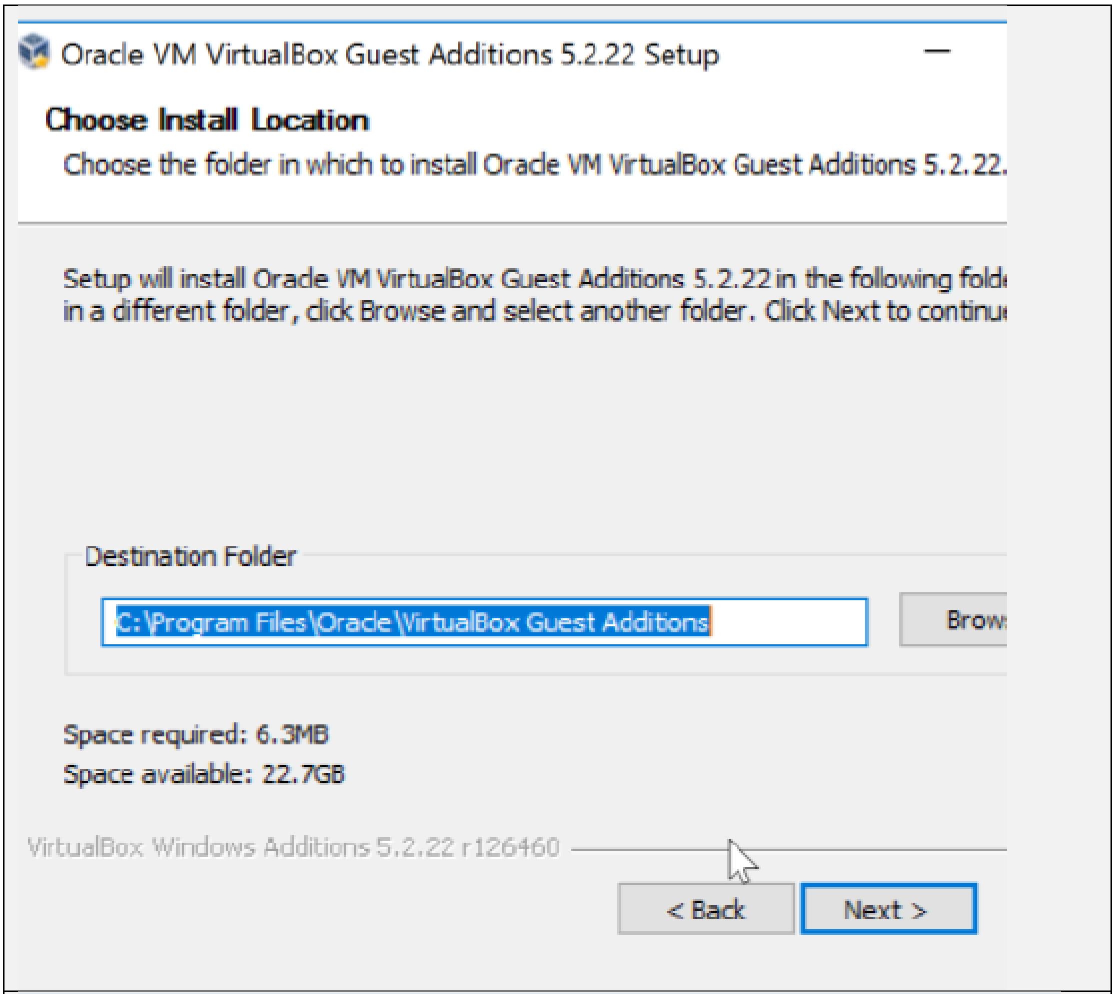
Attached to: Bridged Adapter

Name: Intel(R) Dual Band Wireless-AC 8265

Allows to navigate through VM Settings categories

CD Drive (E:) VirtualBox Guest Additions

0 bytes free of 55,3 MB



Oracle VM VirtualBox Guest Additions 5.2.22 Setup

Choose Install Location

Choose the folder in which to install Oracle VM VirtualBox Guest Additions 5.2.22.

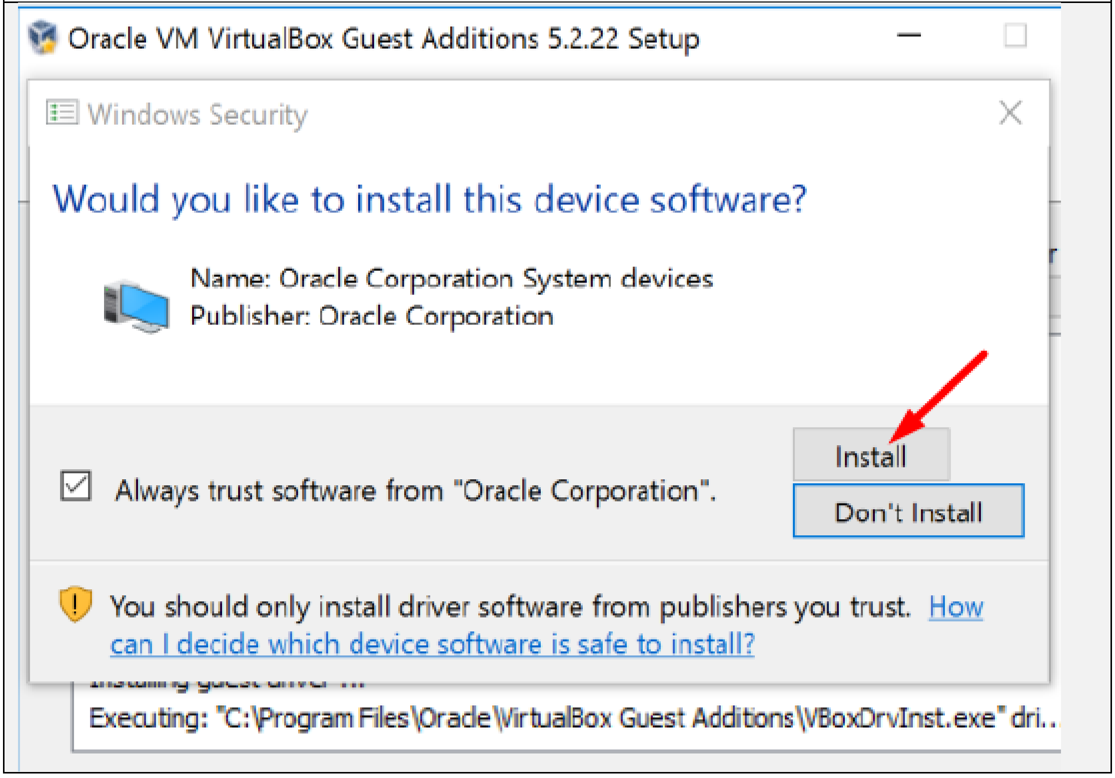
Setup will install Oracle VM VirtualBox Guest Additions 5.2.22 in the following folde
in a different folder, click Browse and select another folder. Click Next to continu

Destination Folder

C:\Program Files\Oracle\VirtualBox Guest Additions

Space required: 6.3MB
Space available: 22.7GB

VirtualBox Windows Additions 5.2.22 r126460



Oracle VM VirtualBox Guest Additions 5.2.22 Setup

Windows Security

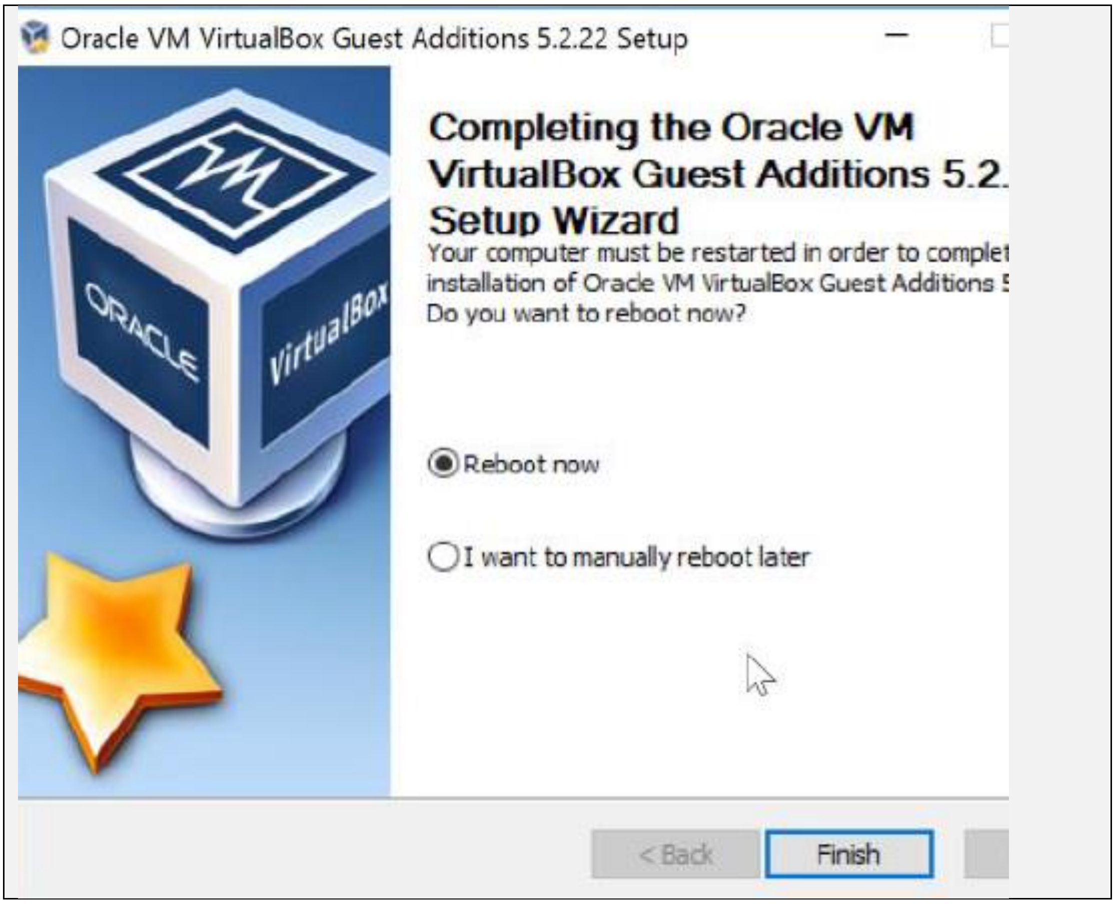
Would you like to install this device software?

Name: Oracle Corporation System devices
Publisher: Oracle Corporation

☑ Always trust software from "Oracle Corporation".

You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

Executing: "C:\Program Files\Oracle\VirtualBox Guest Additions\VBoxDrvInst.exe" dri..



Oracle VM VirtualBox Guest Additions 5.2.22 Setup

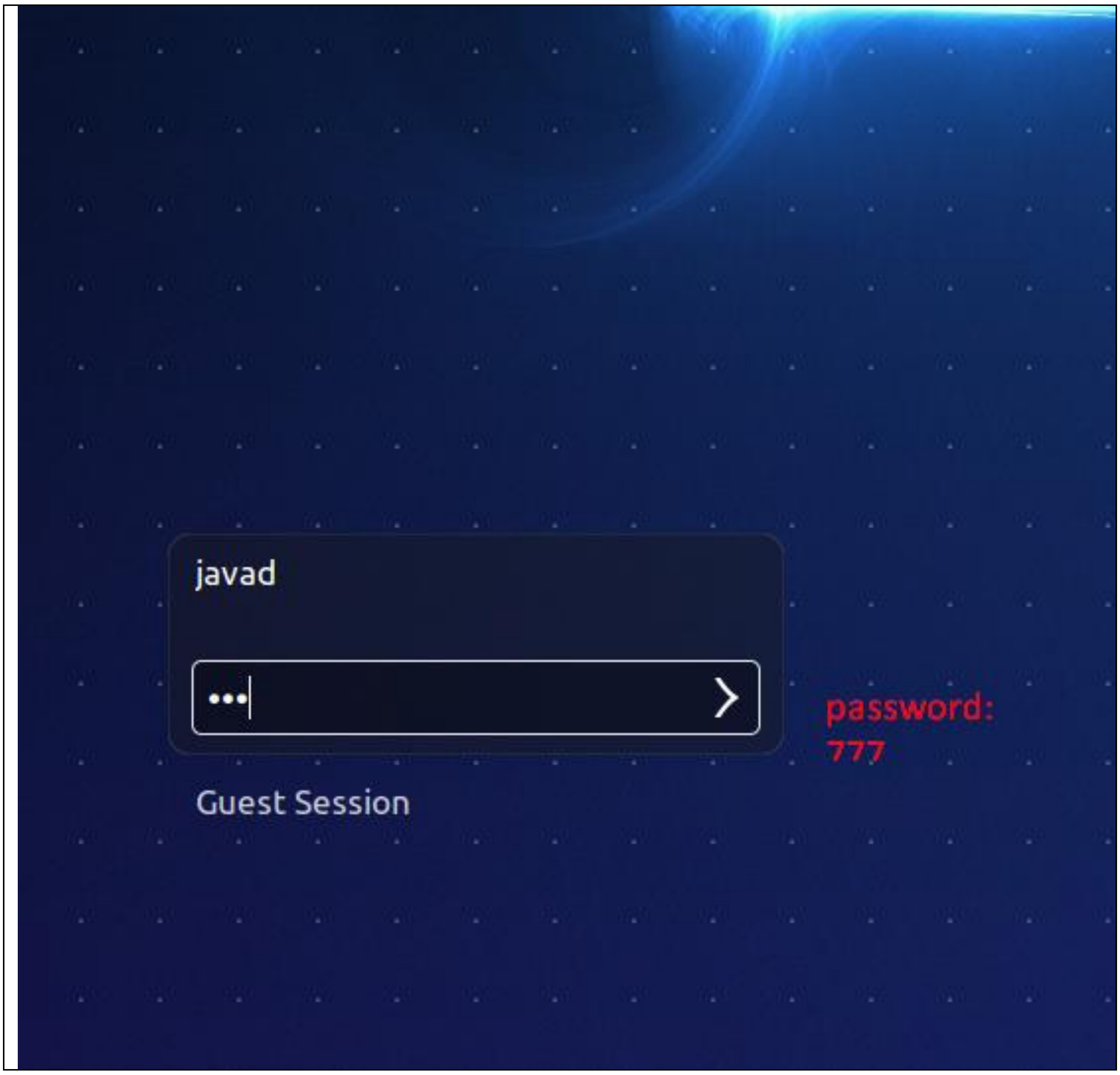
Completing the Oracle VM VirtualBox Guest Additions 5.2. Setup Wizard

Your computer must be restarted in order to complet installation of Oracle VM VirtualBox Guest Additions 5 Do you want to reboot now?

● Reboot now

○ I want to manually reboot later

See some actions



javad

•••

password: 777