Bahir Dar Unversty

Bahir Dar Institute Of Technology Faculity Of Computing

Department: Software Engineering

Project Name: Android OS

Name : Tiruneh Getachew

ID : BDU1602613

Submitted To: Lec Wendimu B

Sobmission Date: 16/08/17

❖ Introduction

This document explains how to set up the Android operating system on a computer using virtualization. Virtualization means creating a software-based version of an operating system that runs on your computer, alongside your regular operating system. This is useful for testing, learning, and using Android without needing a separate physical device.

➢ History of Android Operating System

Android began in 2003 when Andy Rubin and his team founded Android Inc. Initially, they wanted to create a smarter operating system for digital cameras. However, they later decided to focus on building an operating system for mobile phones. In 2005, Google bought Android Inc. and started developing Android based on the Linux system. Their goal was to compete with other mobile operating systems. The first Android phone, the HTC Dream, was released in 2008. It came with basic applications like Gmail, Maps, and YouTube. Since then, Android has gone through many updates, with major releases like Gingerbread, Ice Cream Sandwich, Lollipop, and Android 10 and later versions. These updates have improved the user interface, added new features, and enhanced security. Today, Android is used on billions of devices worldwide, including phones, tablets, TVs, and even cars.

➢ Background

Android is an open-source operating system, meaning its code is freely available, and it’s built on the Linux platform. It’s mainly designed for devices with touchscreens, like smartphones and tablets. Android has become the most popular OS in the world because it’s flexible, has a huge selection of apps, and works on many different devices. Because it's open-source, developers can explore and change the system. Running Android on a virtual machine helps connect mobile OS development with regular computer environments, which is great for learning.

➢ Motivation

This project aims to provide practical experience in installing and managing an operating system using virtualization tools like VMware. By exploring Android in a virtual environment, students can learn how it works, how to install it, and how it performs without needing an actual Android device. It also encourages experimentation and helps build technical skills. Virtualization allows students to create realistic system simulations in a safe, reversible environment.

➢ Objective

The goals of this project are:

- To successfully install the Android operating system on VMware Workstation.
- To understand virtualization and how Android interacts with virtual hardware.
- To document the installation process, including any problems encountered and their solutions, as well as details about the file system.
- To improve problem-solving skills by addressing practical errors during installation.

- To assess the educational and developmental advantages of using virtualized Android systems.

By meeting these goals, students will gain a better understanding of how operating systems work, the benefits of virtualization, and how Android behaves in virtual environments.

❖ Hardware and Software Requirements

To install and run Android in a virtual environment using VMware, you need to ensure your computer meets certain hardware and software requirements. These requirements are necessary for the virtual machine to operate smoothly and avoid performance or compatibility issues.

➢ Hardware Requirements

Your computer should have the following minimum hardware:

- **RAM (Memory):** At least 4 GB, but 8 GB or more is recommended for better performance.
- **Storage:** At least 20 GB of free disk space for the virtual disk and installation files.
- **Processor:** Intel or AMD processor with Virtualization Technology (VT-x or AMD-V) enabled in the BIOS. This is crucial for running 64-bit operating systems virtually.
- **Graphics:** Basic integrated graphics is enough, but a dedicated GPU will improve display quality and responsiveness.
- **Other:** Internet connection to download the Android ISO and VMware software, and for any additional installations.

These minimum specifications will allow you to start Android, install apps, and explore its features without causing your computer to slow down or freeze.

➢ Software Requirements

You'll need the following software:

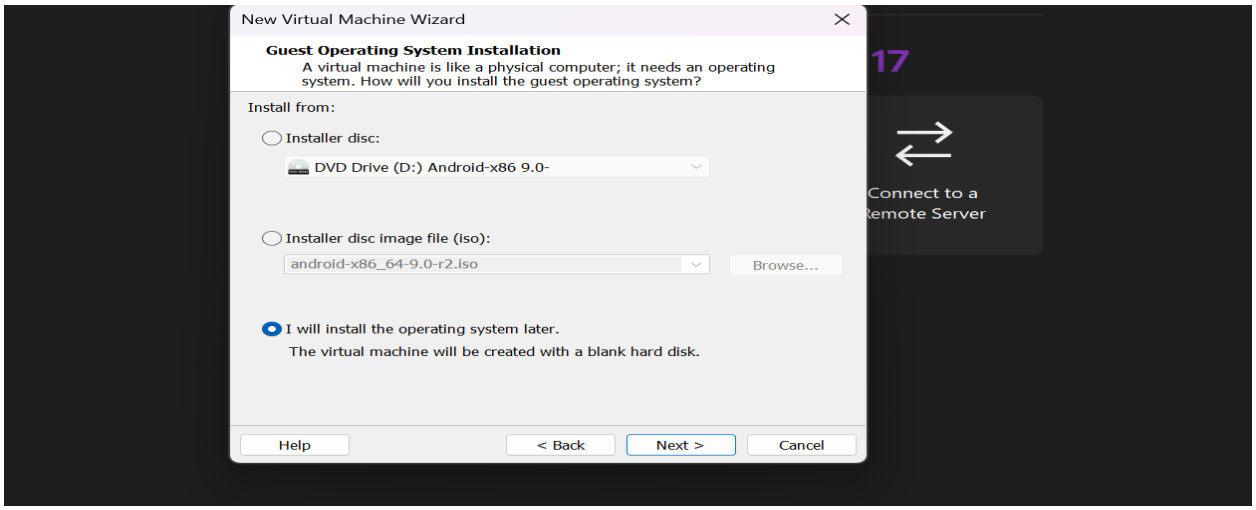
- **VMware Workstation Pro or VMware Workstation Player:** VMware is a virtualization tool that allows you to run multiple operating systems on one physical machine. VMware Player is free for personal use, while Workstation Pro has more advanced features. It helps in creating virtual disks, allocating resources, and booting from ISO files.
- **Android-x86 ISO Image:** Android-x86 is a project that allows Android to run on regular computer hardware (x86 architecture). You'll use this to install Android as a guest operating system in VMware. For example, the Android-x86 9.0 ISO (Pie) is a stable and widely compatible version. You can download it from the official site: www.android-x86.org.
- **Optional Tools:** File extraction tools (like WinRAR or 7-Zip) to open compressed files, and a PDF reader or browser to view installation guides.

❖ Installation Steps

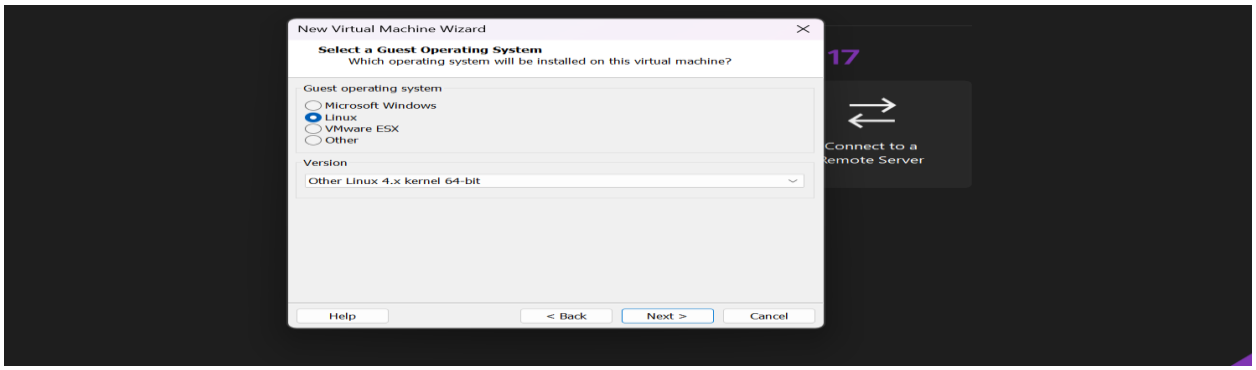
This section describes how to install Android-x86 on VMware Workstation or VMware Player. The process involves creating a virtual machine, selecting the Android ISO image, setting up the virtual hardware, and installing the operating system.

1. **Launch VMware:** Open VMware Workstation or Player and click on "Create a New Virtual Machine."

2. Choose "Custom (Advanced)" for more control over the setup.
3. **Configure Virtual Machine Settings:**
    - Hardware Compatibility: Choose the latest version.
    - Installation Method: Select "I will install the operating system later."

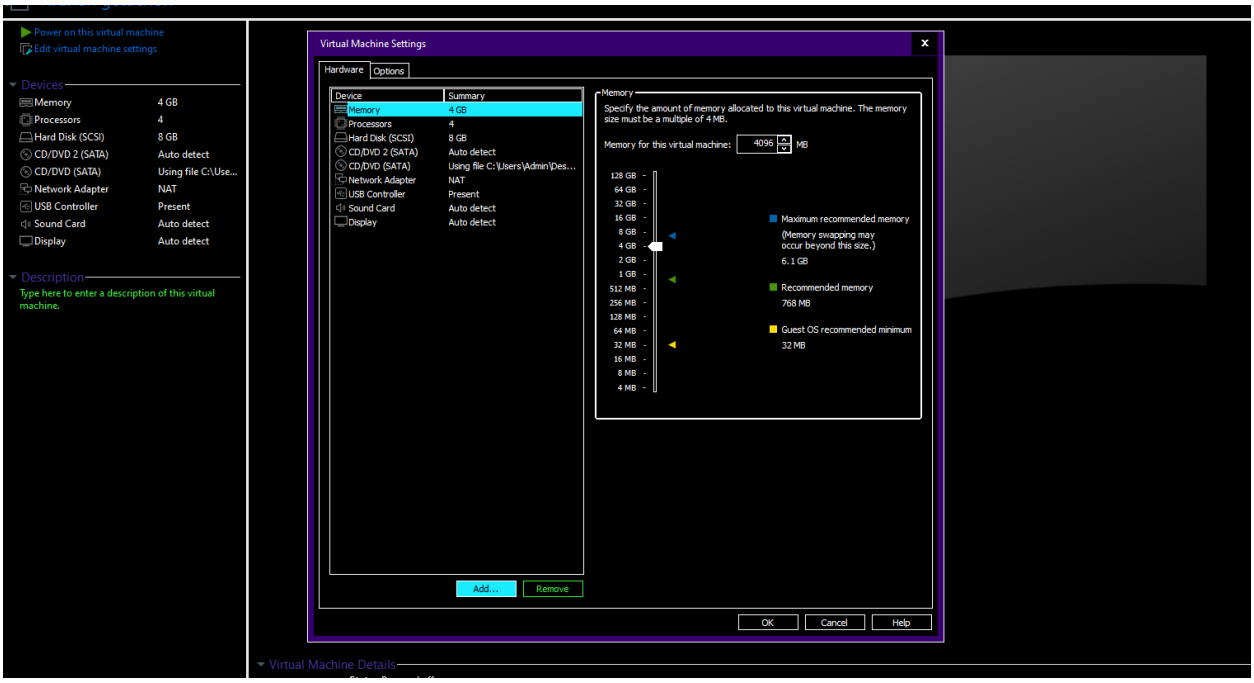


- Guest OS: Select "Other" and "Other (64-bit)."



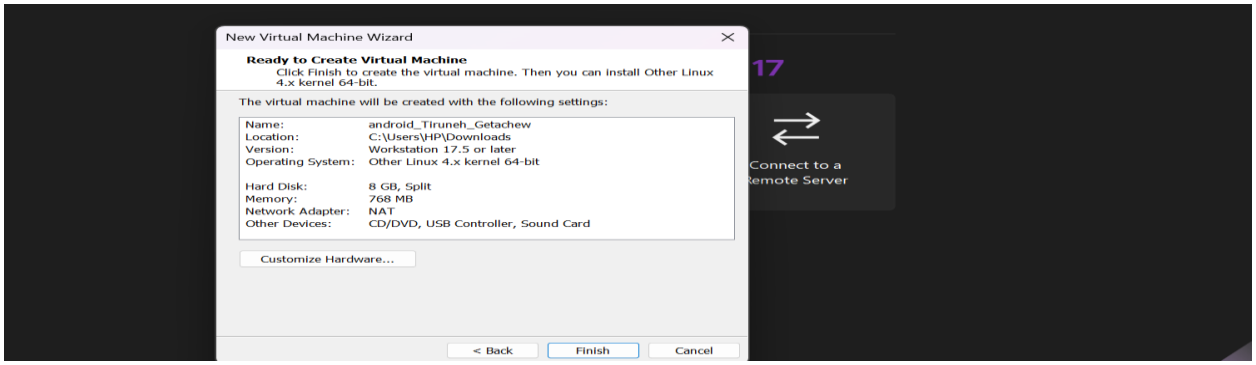
- Name our Virtual Machine: Give it a name like "Android_YourName" (e.g., Android_ Tiruneh_Gtetachew) and choose where to save the files.

4. **Set Processor and Memory:**
   - Processor: 2 cores (or 1 core for less powerful computers).
   - Memory (RAM): Minimum 2048 MB (2 GB), Recommended: 4096 MB (4 GB).

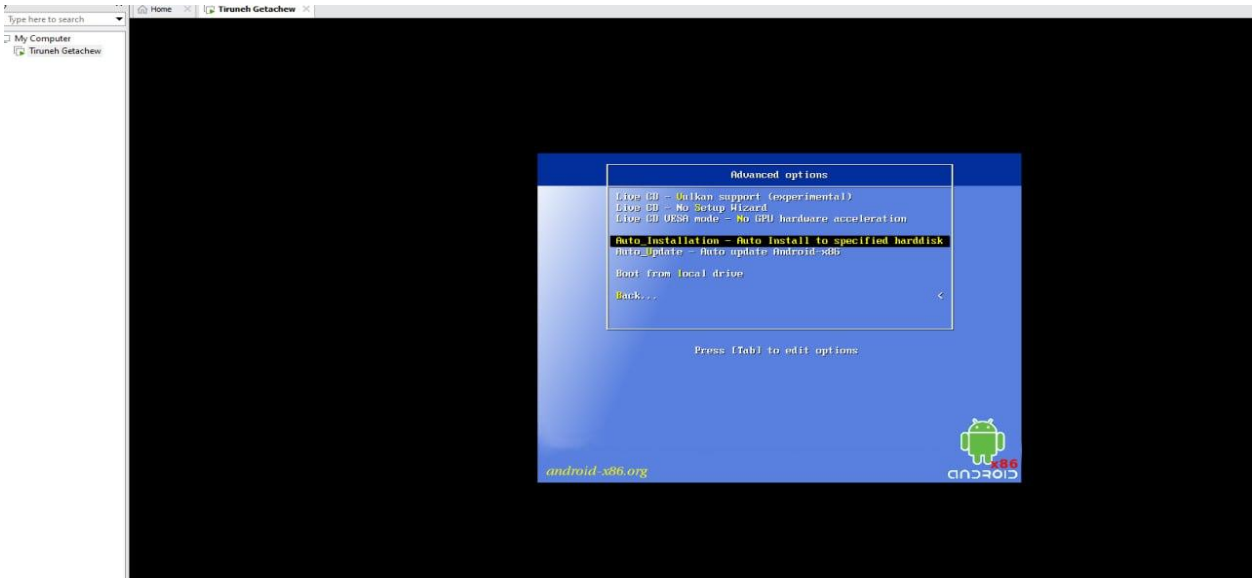


5. **Configure Network and Disk:**
   - Network Type: Use "NAT" (default).
   - I/O Controller: LSI Logic (Recommended).

- Virtual Disk: Create a new virtual disk, choose "SCSI" as the disk type, set the size to 8–20 GB, and select "Split into multiple files."

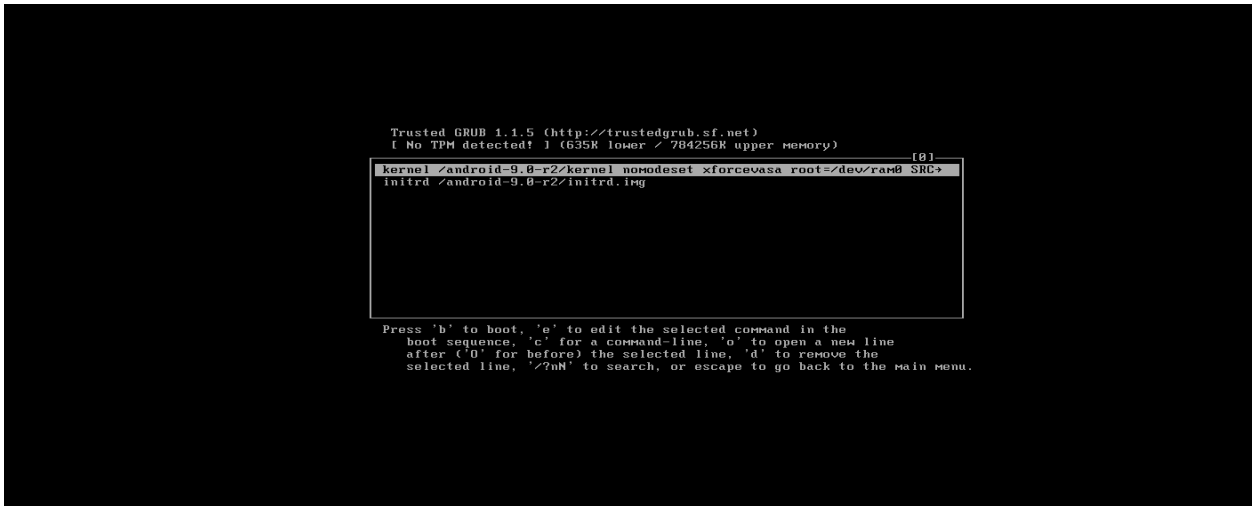


6. **Mount Android ISO:**
   - Go to "VM Settings."
   - Under "CD/DVD (IDE)," choose "Use ISO image file."
   - Select the Android-x86 ISO file.
7. **Advanced Options:**
   - Debug mode: Helps in troubleshooting boot problems.
   - VESA mode: Tries different screen settings for display issues.
   - Install in Debug mode: Installs Android with more technical details for developers.
   - Auto screen resolution: Fixes display problems in VMware or VirtualBox.
8. **Auto Installation:**
   - This option installs Android automatically, without asking questions about partitions or formatting.

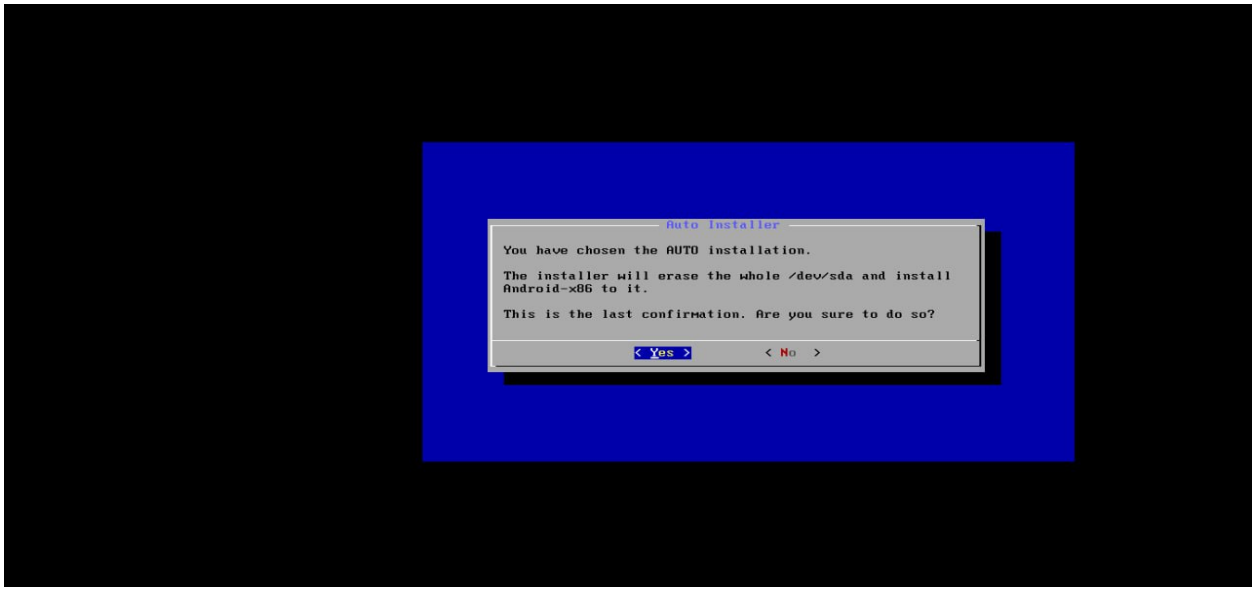


9. **Boot and Start Installation:**

- Power on the virtual machine.
- Select "Installation - Install Android to hard disk" from the boot menu.



- If you choose "Yes" in the confirmation step, it will erase the virtual disk and install Android-x86 with default settings.

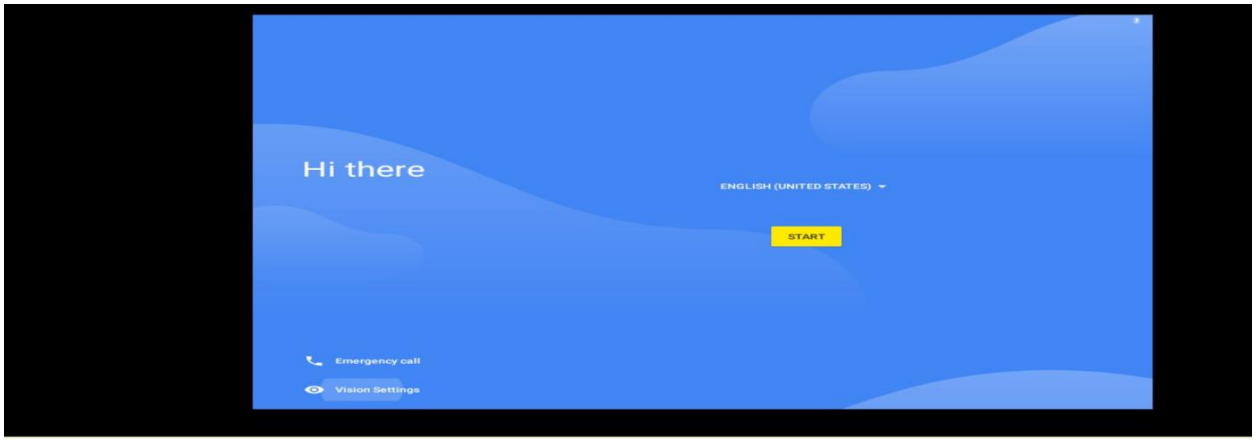


- The installer will copy the system files to the virtual hard disk.

- After installation, the system will prompt for a reboot.
- The Android boot animation will appear, followed by the setup screen.

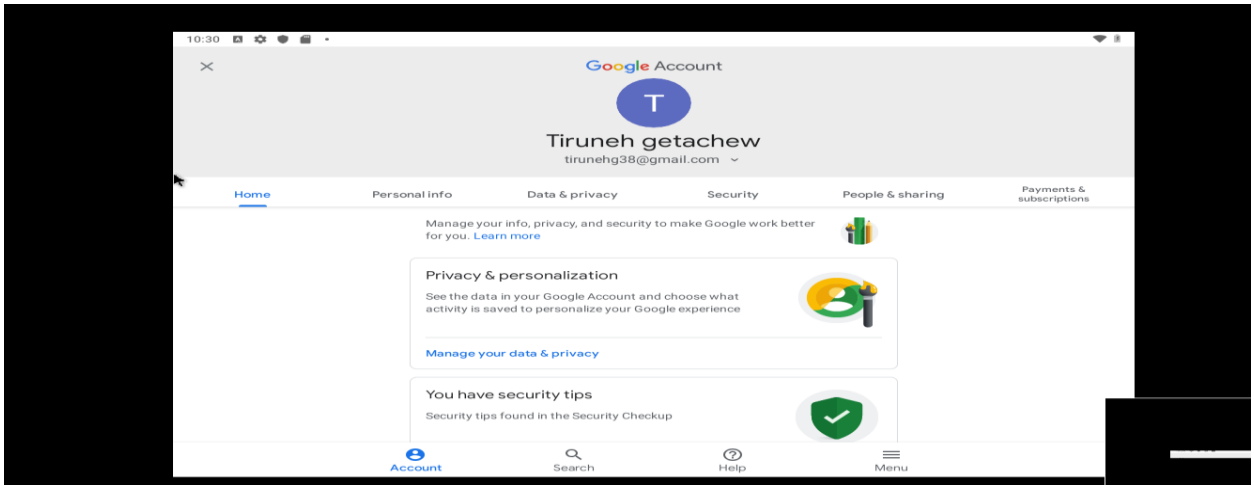
10. **Create Partitions for Android**
    - Choose Create/Modify Partitions
    - Use New -> Primary -> Allocate full disk size -> Bootable -> Write -> Quit
    - Select the created partition (e.g., sda1)

11. **Format and Install**

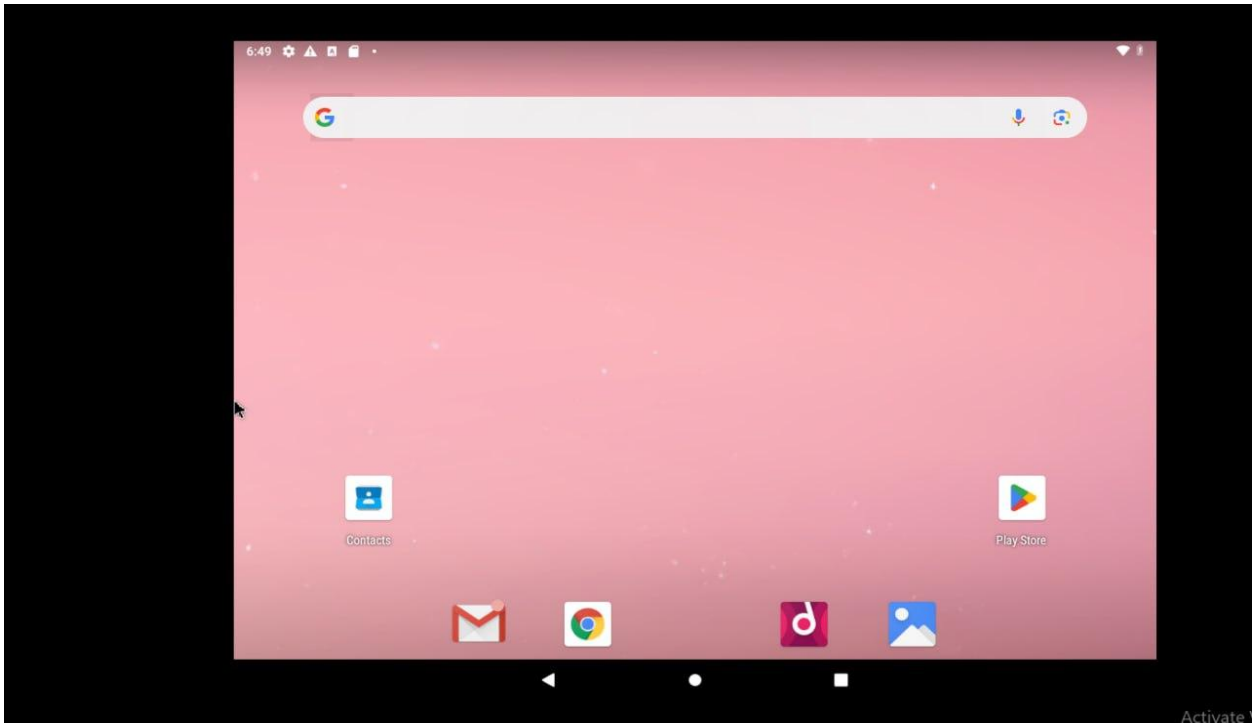


- Format the partition with ext4 filesystem

- When prompted:
  - Install GRUB bootloader
  - Install EFI GRUB2 (if prompted)
  - Make system directory read-write
- Create a user account using your full name

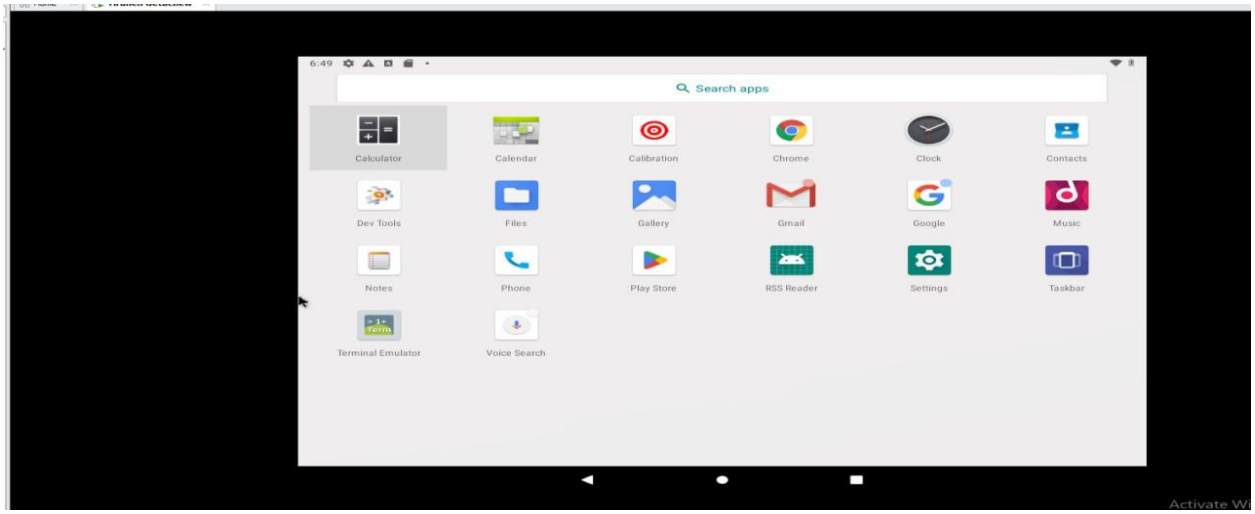


12. **Android is Ready**



- You should now see the Android home screen.
- You can explore settings, install APKs, and test apps.
- Use the mouse and keyboard to interact.
- Android OS is now installed in VMware.

- You can use this setup for testing and experimenting without needing a physical device.

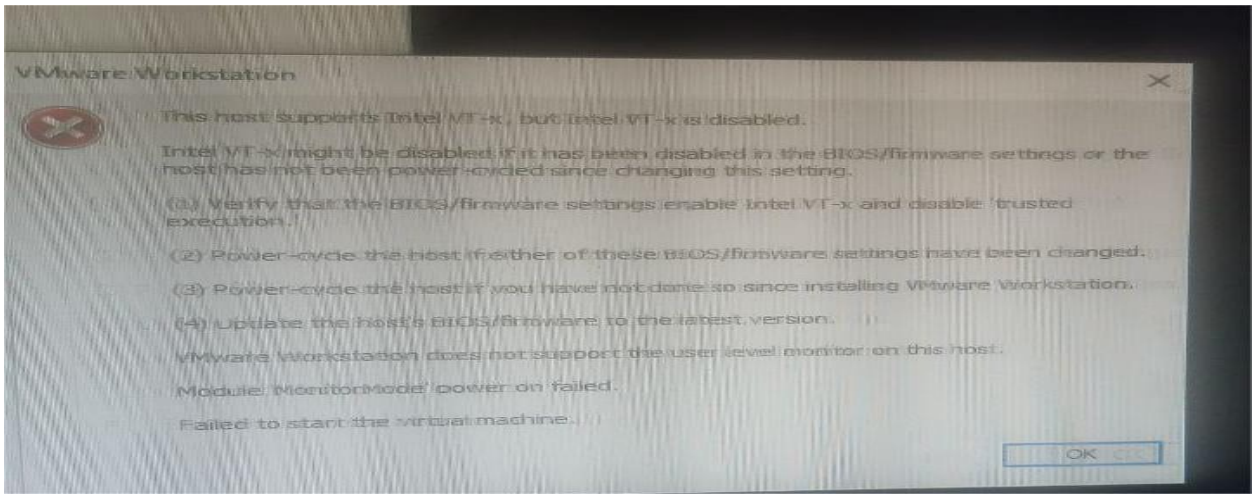


❖ Issues (Problems Faced During Installation)

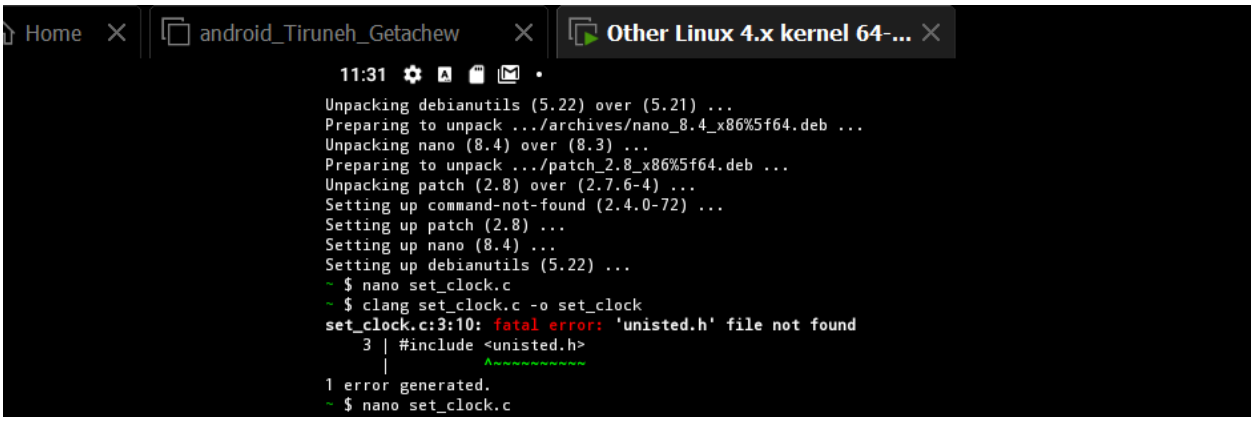
When installing Android OS in a virtual environment with VMware, you might encounter some technical problems. These can be related to compatibility, VMware settings, or how Android behaves in a virtual machine.

Common Issues:

- **VMware Error – Intel VT-x is Disabled:** This error occurs because Intel VT-x (Virtualization Technology) is turned off in the computer's BIOS settings. VT-x is necessary for running 64-bit operating systems in a virtual environment.



- **Mouse or Keyboard Not Responding:** Sometimes, the mouse or keyboard doesn't work correctly within Android after it starts.
- **Low Screen Resolution or Display Problems:** Android might start with a very low resolution, affecting visibility.
- **Boot Loop or Freezing During Installation:** The system may freeze or restart repeatedly during the setup process.
- **ISO File Not Booting:** If the ISO file isn't correctly connected, the virtual machine won't recognize it, and you'll see an "Operating System Not Found" error.
- **Slow Performance or Lag:** Insufficient RAM or CPU allocation can cause Android to run slowly.
- **I also gate formatting error in system call implementation.** See below.



Then I fix by return back the code easly by typing correctly , “#include <unistd.h>

➢ Solutions to the Issues

Here are the solutions to fix the common problems:

- **Intel VT-x is Disabled:**
  - Restart the computer and enter the BIOS/UEFI setup (usually by pressing Esc, F2, F10, or Del during startup).
  - Find the virtualization settings (usually in the "Advanced" or "System Configuration" tab).
  - Enable "Intel Virtualization Technology (VT-x)" or "Virtualization."
  - Disable "Trusted Execution" if it's present.
  - Save the changes and exit the BIOS.

❖ Filesystem Support

Android-x86 uses the ext4 file system as its main file system during installation.

- **ext4 (Fourth Extended Filesystem):** This file system is used because it's fast, reliable, and includes journaling, which helps in recovering files if the system shuts down unexpectedly. Since Android is based on Linux, it naturally supports ext4. It is designed for good performance and can handle large storage capacities.

❖ Advantages and Disadvantages

There are both advantages and disadvantages to installing Android OS in a virtual environment. It's important to weigh these when considering this setup for development or learning.

➢ Advantages

- **No Need for a Physical Android Device:** You can test and use Android on your computer.
- **Safe Testing Environment:** It's a safe space to experiment without risking damage to a real mobile device.
- **Cost-Effective and Flexible:** This setup is beneficial for students or developers who have limited resources.
- **Quick Snapshots and Rollbacks:** Virtual machines can be easily saved and restored to previous states.
- **Platform Independence:** You can run Android on Windows, Linux, or Mac operating systems using virtualization.

➢ Disadvantages

- **Limited Hardware Support:** Some Android features that rely on specific hardware, like cameras, sensors, or GPS, won't work in a virtual machine.
- **Graphics and Performance Limitations:** Apps that require high-end graphics or animations may run slowly or not at all.
- **Complex Configuration:** Setting up the virtual environment requires some knowledge of virtual hardware and disk partitioning.
- **Storage and RAM Usage:** Virtual machines can use a lot of storage and RAM on your computer, especially when multitasking.

❖ Conclusion

This project has shown that it's possible and useful to install Android OS in a virtual environment using VMware. The process involves creating a virtual machine, configuring hardware and disk partitions, solving compatibility issues, and successfully running Android as a virtual OS. Through this practical experience, we've gained a better understanding of virtualization and how Android works internally. We were able to simulate a working mobile operating system on a computer, making it easier to learn, test, and develop without needing physical devices. It also highlighted the importance of file systems, installing the GRUB bootloader, and managing system resources. Overall, using virtualization for Android is a valuable and convenient method for both learners and developers.

❖ Future Outlook / Recommendations

In the future, there are several ways to expand on this project:

- **Try Newer Android Versions:** Explore installing more recent Android versions like 10, 11, or 12, which may offer more features and better compatibility.
- **Use Advanced Emulators:** Consider using tools like Genymotion, Android Studio Emulator, or Bliss OS, which provide more features and better performance.
- **Explore Android App Development:** Use the virtual Android system to install and test APKs using the Android Debug Bridge (ADB).
- **Improve VM Performance:** Allocate more RAM and enable hardware virtualization in the BIOS settings.
- **Network and File Sharing:** Experiment with setting up network access and sharing files between the Android virtual machine and the host computer.

❖ What, Why, and How of Virtualization in Modern Operating Systems

➢ What is Virtualization?

Virtualization is a technology that allows you to run multiple operating systems on a single physical machine by using a special software layer called a hypervisor. The hypervisor simulates hardware resources like the CPU, RAM, storage, and network interfaces, so that multiple "virtual machines" (VMs) can operate independently and at the same time. Each virtual machine functions like a separate computer with its own operating system, but it's completely isolated from the other VMs. Virtualization simplifies the physical hardware and provides flexible, scalable environments for development, testing, deployment, and learning. For example, running Android OS in VMware on a Windows computer is a form of virtualization. Android "thinks" it's installed on a physical machine, but it's actually running inside a virtual one.

➢ Why is Virtualization Important in Modern Operating Systems?

Virtualization is widely used in IT, cloud computing, software development, and education. Here are some key reasons for its importance:

- **Efficient Resource Utilization:** Running multiple virtual systems on one physical system optimizes the use of CPU, memory, and storage.
- **Cost Reduction:** Organizations can save money by needing less hardware, power, and maintenance.
- **Testing and Development:** Developers can test applications on various operating systems without requiring multiple physical devices.
- **Isolation and Security:** Virtual machines are isolated from each other, so a problem in one VM won't affect others or the host system.

- **Backup and Recovery:** VMs can be easily backed up, cloned, and restored, which improves disaster recovery and system migration.
- **Support for Older Systems:** Virtualization allows older operating systems or software to run, even if they're not compatible with newer hardware.

➢ How Does Virtualization Work?

Virtualization relies on a hypervisor to manage virtual machines. There are two main types of hypervisors:

- **Type 1 Hypervisor (Bare-Metal):** Installed directly on the physical hardware (e.g., VMware ESXi, Microsoft Hyper-V, Xen).
- **Type 2 Hypervisor (Hosted):** Runs within a host operating system (e.g., VMware Workstation, Oracle VirtualBox).

Basic Workflow:

- The hypervisor allocates system resources (CPU, RAM, disk space, network) to each VM.
- The guest operating system (like Android, Linux, or Windows) is installed in the VM as if it were on physical hardware.
- Users can manage and switch between multiple VMs easily.

Use Case Example: Installing Android OS in VMware

In this project, Android OS is installed on a virtual machine using VMware (a Type 2 hypervisor).

- Host OS: Windows (your main system).
- Hypervisor: VMware Workstation or Player.
- Guest OS: Android-x86. [1]

  Use Case Example: Installing Android OS in VMware In your project, Android OS is installed on a virtual machine created using VMware (a Type 2 hypervisor). Here's how virtualization applies: Host OS: Windows (your main system) Hypervisor: VMware Workstation or Player Guest OS: Android-x86 Result: Android runs in a separate, isolated space, allowing app testing, experimentation, and learning without affecting the host.

Result: Android runs in a separate, isolated environment, which is useful for testing apps, experimenting, and learning without affecting the main system.

Conclusion

Virtualization has significantly changed modern computing. From cloud data centers to mobile app development, it enables the simulation of entire systems, resource conservation, and improved security. Learning virtualization tools like VMware is a valuable skill for students and software engineers in development and system administration.

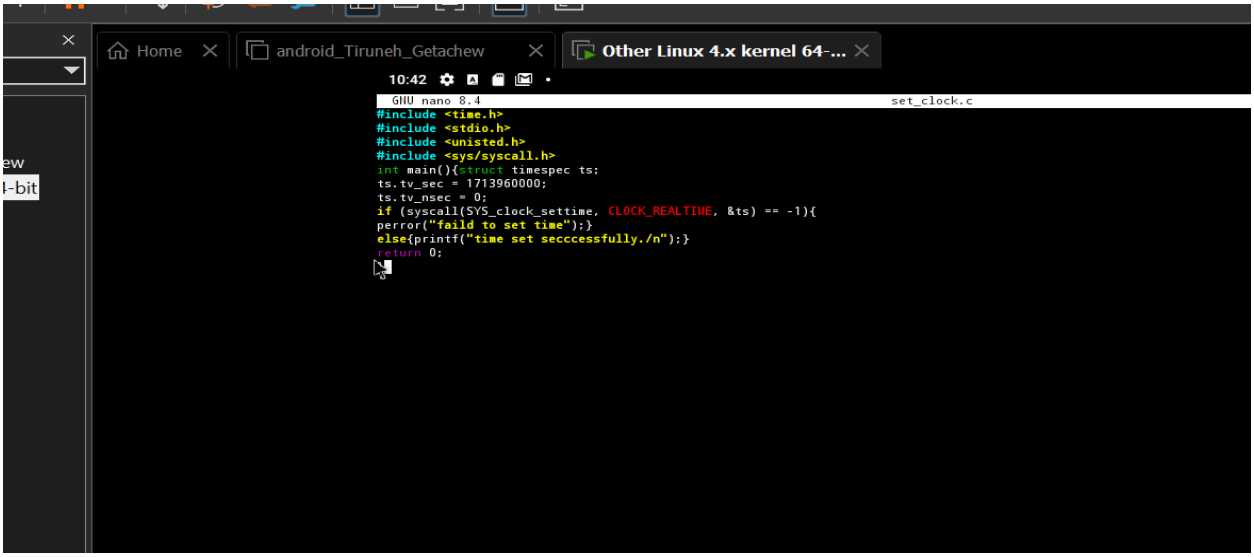
❖ Implementing a Simple System Call

This project includes implementing a basic system call to demonstrate how applications interact with the operating system. System calls are special functions that user-level programs use to request services from the operating system's core (kernel).

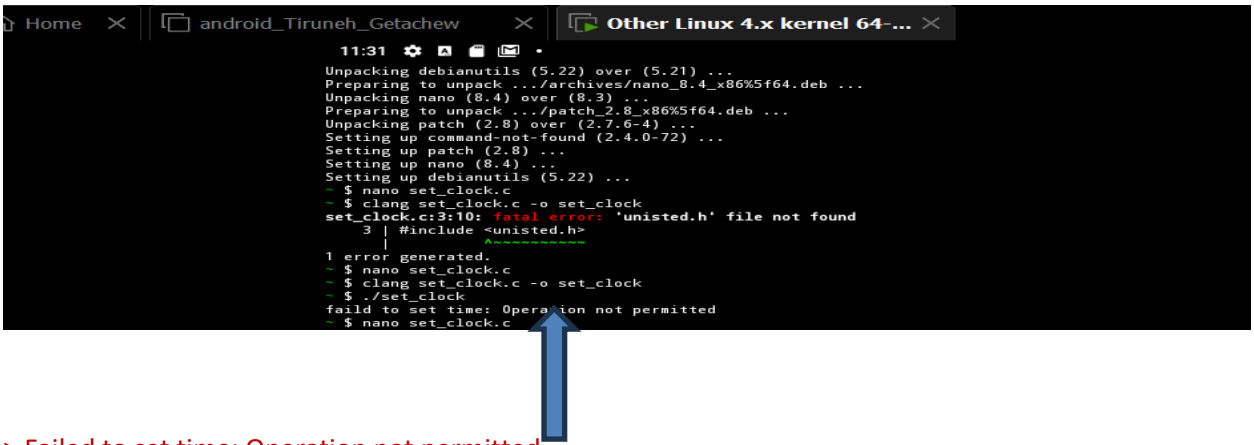
This system call is part of the POSIX standard and allows a process (with appropriate privileges) to change the time for a specific clock — most commonly, CLOCK_REALTIME, which affects the entire system clock.

```
#include <time.h>

#include <stdio.h>

#include <unistd.h>

#include <sys/syscall.h>


int main() {

   struct timespec ts;

   ts.tv_sec = 1713960000;  // Example UNIX timestamp

   ts.tv_nsec = 0;


   // Direct syscall to clock_settime

   if (syscall(SYS_clock_settime, CLOCK_REALTIME, &ts) == -1) {

      perror("Failed to set time");

   } else {

      printf("Time set successfully.\n");

   }
```

```
    return 0;
}
```



When I run the program it display thi image below.



> Failed to set time: Operation not permitted

Means my code works, but Android blocks changing the system time because not root (superuser).

❖ References

- Android-x86 Official Website. *Android-x86 - Porting Android to x86.* http://www.android-x86.org
- VMware Official Documentation. *VMware Workstation User Guide.* https://www.vmware.com
- VMware Official Documentation. *VMware Workstation User Guide.* https://www.vmware.com
- YouTube. *Android-x86 Installation on VMware (Video Guide).* https://www.youtube.com/results?search_query=android+x86+vmware+installation