NATIONAL UNIVERSITY
of Computer & Emerging Sciences

Course: SE1001 Introduction to Software Engineering.

Instructor: Prof Engr. Umar Haroon

Submitted by: Muhammad Rehan

Roll no: 22P-9106

Class: BSE-2A

Date: March 29th, 2023.

Assignment no 2
Department of Computer Science

Part no 01: Name

ProGen OS Alchemist: Build Your Perfect Operating System

Explanation:

"ProGen" is short for "**progenitor**," which means an ancestor or a parent of something, in this sense, the **Linux kernel**. OS is for the operating system. Alchemist refers to someone who transforms something ordinary into something extraordinary, just like an alchemist transforms base metals into gold. **Alchemist** could refer to a tool that transforms essential hardware components into a **powerful operating system** in the context of the software.

Part no 02: Requirements

User Requirement Specification:

This software tool shall create a custom operating system for a user's PC. This software shall allow users to select the desired components and features in their operating system and create a custom **ISO** file to install the operating system. Users shall select various parameters, such as **hardware compatibility**, **software packages**, and **configurations,** to customize the operating system image. The software shall use an automated process to generate the ISO file, making it easy for users to create custom operating systems per their specific needs.

System Requirement Specification:

This project allows users to create a customized operating system tailored to their specific needs and preferences. It offers a wide range of options and functions to optimize system performance, including the choice of **high-performance hardware components**, a file system with **high I/O throughput** and **low latency**, and customizable **kernel settings** and **modules**. Users can allocate more resources to specific applications or processes and **optimize network settings** for low latency and high throughput. The project also provides customizable **memory settings** and options for **low-level system components** and

power management. Additionally, users can choose from various **desktop environments**, **theme options**, **default applications**, and **pre-installed software** for specific purposes. The project also includes options for **system resource allocation**, optimizing the system for the particular usage, and **system updates** and **maintenance options**. Finally, users can customize system security features, choose system **language and locale settings**, and more.

Part no 03:   Functional Requirements

- The software shall allow users to choose high-performance hardware components such as the **processor**, **RAM**, and **graphics card** and allow users to select system resource allocation.
- The software shall allow users to choose a file system with **high I/O throughput** and **low latency** for fast disk access, such as **Btrfs or XFS.**
- The software shall allow users to customize **kernel settings** and modules to optimize performance for specific hardware configurations.
- The software shall allow users to allocate more resources to specific applications or processes using **cgroups** or **task sets**.
- The software shall allow users to customize network settings for **low latency** and **high throughput**, such as disabling unnecessary network protocols and optimizing **network stack settings**.
- The software shall allow users to choose low-level system components such as different **init systems**, alternative **display managers**, or **custom kernel modules**.
- The software shall allow users to disable **power management features** to maximize performance.
- The software shall allow users to use **real-time kernel patches** for applications that **require low latency**.
- The software shall allow users to choose a desktop environment such as **GNOME**, **KDE**, **XFCE**, **LXDE**, etc.
- The software shall allow users to customize theme options such as **colors**, **fonts**, **icons**, etc.
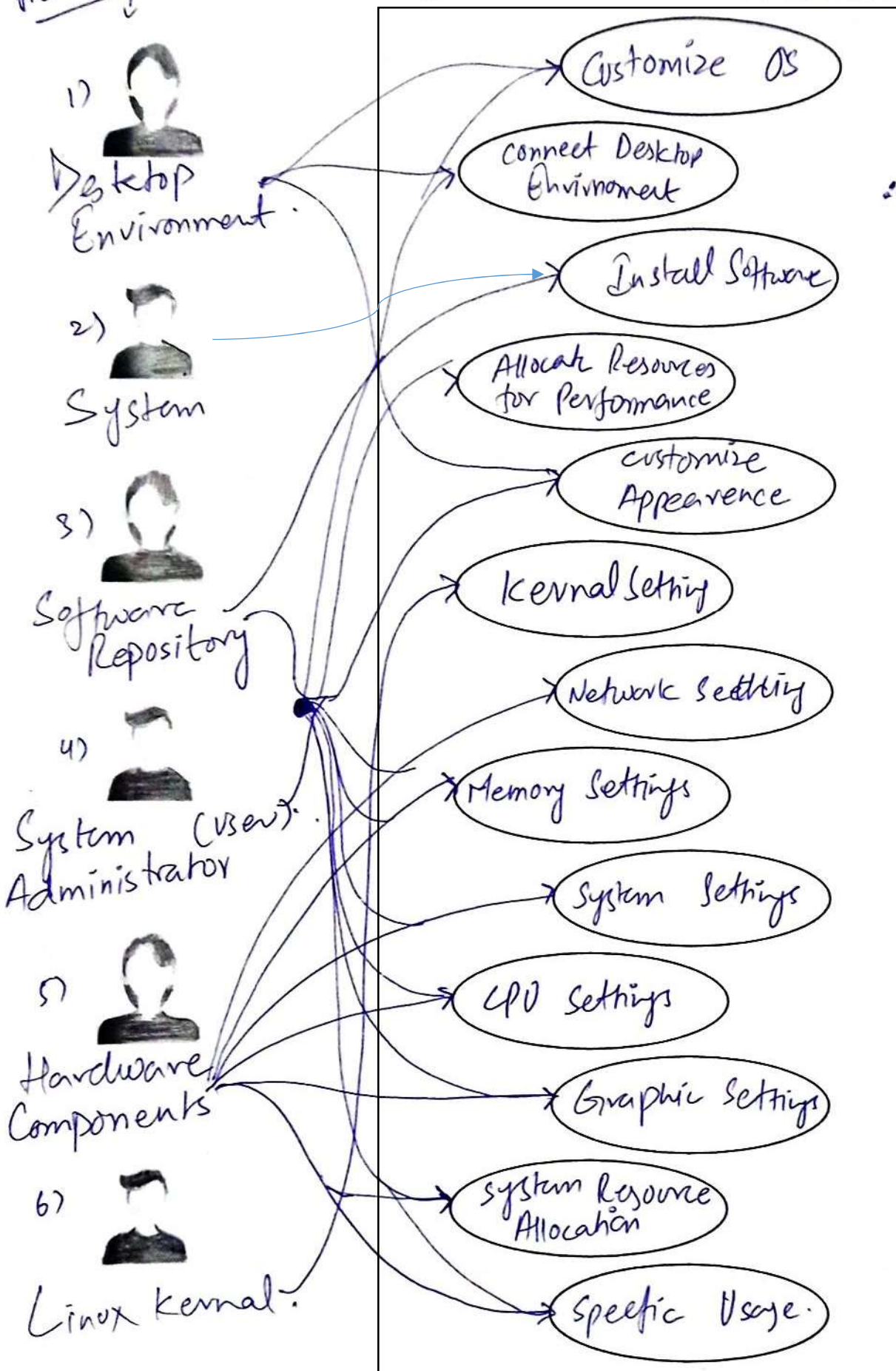
- The software shall allow users to choose default applications such as a **browser**, **email client**, **office suite**, **media player**, etc.
- The software shall allow users to install additional games, **development tools**, **graphic design** software, etc.
- The software shall allow users to choose pre-installed software for specific purposes such as **education**, **entertainment**, **productivity**, etc.
- The software shall allow users to optimize the system for specific usage such as **gaming**, **video editing**, **office work**, etc.
- The software shall allow users to choose system updates and maintenance options such as **automatic** or **manual** updates.
- The software shall allow users to customize system security features such as **firewalls** or **antivirus**.
- The software shall allow users to choose system **language** and **locale settings**.

Non-Functional Requirements

- The software shall be compatible with a wide range of **hardware configurations**.
- The software shall have a user-friendly **interface** for ease of use.
- The software shall provide **clear** and **concise instructions** for each **customization** option.
- The software shall be **stable** and **reliable** without **crashing** or causing **data loss**.
- The software shall be **secure** and **protect** user data from **unauthorized** access.
- The software shall be **scalable** and able to handle many users.
- The software shall be compatible with standard Linux **distributions** and their **package managers**.
- The software shall have **low system requirements** to run smoothly on most **hardware configurations**.
- The software shall perform well and speed up the creation of **custom Linux operating systems**.
- The software shall be well-documented and provide user support for **troubleshooting and assistance**.

Use Case Diagram: ProGen OS Alchemist

Actors:- 6 total.

Use cases :13 total.

∴ User & System is connected to all use cases.

Actors:

1) Desktop Environment.

2) System

3) Software Repository

4) System Administrator (user).

5) Hardware Components

6) Linux kernal.

Use Cases:

- Customize OS
- Connect Desktop Environment
- Install Software
- Allocate Resources for Performance
- customize Appearance
- Kernal Setting
- Network Setting
- Memory Settings
- System Settings
- CPU Settings
- Graphic Settings
- System Resource Allocation
- Specific Usage.