

National University of Computer & Emerging Sciences
Karachi Campus



SysLink

Project Proposal
Operating System (CS-2006)

Section: BCS-4J

Instructor Name: Miss Anaum Hamid

Group Members:

Sufiyaan Usmani (21K-3195)

Yousuf Ahmed (21K-4594)

Muhammad Hamza (21K-3293)

Project Title: SysLink

Introduction:

Introducing our newest app, designed for Linux enthusiasts and developers alike. With this app, you can connect to your Linux machine using the secure shell (SSH) protocol, allowing you to remotely access and control your machine from anywhere in the world. But that's not all - this app takes it a step further by implementing system calls. With these calls, you can execute low-level operations on your machine directly from the app, making it easier than ever to manage your Linux system. Whether you're a seasoned Linux user or just getting started, this app is the perfect tool to help you take your system administration to the next level.

Methodology:

1. Requirement Analysis: The first step in developing the app is to understand the requirements of the users. This involves understanding the purpose of the app, the user's needs, and the expected outcomes.
2. Research: The next step is to research the tools and technologies required to develop the app. This includes understanding the SSH protocol, Linux system calls, and any libraries or frameworks that can be used to simplify the development process.
3. Design: Once the requirements have been analyzed and the research has been completed, the next step is to design the architecture of the app. This includes identifying the key components, the interactions between them, and the data flow.
4. Implementation: The actual development of the app takes place in this phase. The app will be developed using programming languages such as Java. The development process should follow best practices such as modularity, abstraction, and error handling.
5. Testing: After the app has been developed, it needs to be tested to ensure that it meets the requirements and is free of bugs. Testing should be done using automated testing tools as well as manual testing to ensure that the app works as expected.
6. Deployment: Once the app has passed testing, it can be deployed to the production environment. This involves configuring the app to work with the Linux system and making sure that it is accessible to users.
7. Maintenance: The final phase of the methodology is maintenance. This involves monitoring the app to ensure that it is working as expected, fixing any bugs that are discovered, and updating the app as needed to meet the changing requirements of the users.

Problem Statement

There is a need for an app that enables users to remotely connect to a Linux machine using SSH protocol and execute system calls. Currently, users have to use command-line tools to perform such operations, which can be time-consuming and cumbersome. Additionally, there are limited options for users to interact with the system beyond the command-line interface. Hence, there is a requirement for an app that simplifies the process of connecting to a Linux machine and allows users to perform system calls directly from the app. This will provide users with a more user-friendly interface and make it easier for them to manage their Linux systems.

Salient Features

File manager: Allow users to browse, manage and manipulate files on their Linux system

Disk usage analyzer: Analyze and visualize disk usage on the Linux system.

Process manager: View and manage running processes on the Linux system.

Terminal emulator: Allow users to run command-line programs and scripts on their Linux system.

Real-time performance metrics: View and monitor system resource usage, including CPU, memory, and disk usage

SSH client: Allow users to connect to remote Linux systems using SSH.

Network configuration: Configure and manage network settings, such as IP address, DNS, and proxy settings

System backup and restore: Allow users to backup and restore their Linux system data and configuration

Tools and Technologies:



Ubuntu



Virtual Box



VMware



C



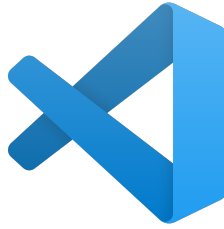
C++



Java



Android Studio



VS Code



Git



GitHub