SCIENXT
Scienxt Journal of Computer Science & Information Technology
SJCSIT
VOLUME-1 || ISSUE-1
JAN-APR || 2023
Scienxt Center of Excellence (P) Ltd.
www.scienxt.com

Scienxt Journal of Computer Science & Information Technology
2023; Volume-1; Issue-1, pp. 1-11

Cloudroid

Mrudul Borse[1], Akash Ranajkar[2], Parth Sakure[3]*, Vaibhavi Sidam[4], Prof. Nikita Kapadnis[5]

[1,2,3,4,5]Department of Computer Engineering,
SCTR’s Pune Institute of Computer Technology
Dhankawadi, Pune, India
Email: parthsakure01@gmail.com

https://zenodo.org/record/7965470

*Corresponding author: Parth Sakure

Abstract:

The Babbage Difference Engine, created in the 1800s, is considered the first computer ever built. Despite its historical significance, the machine's memory storage capacity was quite limited, with just 675 bytes available. Fast forward to the present day, and technological advancements have progressed to the point where most of us carry smartphones in our pockets that boast storage capacities well beyond what the Babbage Difference Engine could ever have imagined. However, despite our smartphones being capable of storing more than 12 gigabytes of data, we still sometimes find ourselves running out of space. This is particularly true in a world where digital media has become an integral part of our daily lives, with high-resolution photos and videos, music, and various other files taking up valuable storage space.

As a result, the demand for extra storage has increased exponentially, and technology companies have responded by introducing various mediums for storing our data. External hard drives, cloud storage, and even physical storage devices like USB flash drives have become commonplace, providing us with additional storage space to store our important files. In today's society, with the vast amount of data being generated every day, the need for ample storage space is only expected to increase. As technology continues to evolve, we can expect to see even more innovative storage solutions being developed to meet the growing demands of users.

The storage that smartphones provide can be divided into 2 parts, one is used for storing application data, and the other is used for storing actual/personal data. While installing new applications into a smartphone there are times when the apt amount of storage is not available due to which we are asked to uninstall some pre-existing applications to accommodate the new application. There are times when uninstalling pre-existing applications is not possible as it will lead to loss of data.

In today's digital age, the amount of data generated every day is growing exponentially. This has led to the development of increasingly powerful mobile devices, which are capable of handling complex tasks and running a wide range of applications. However, these devices have limited storage capacity, which can make it challenging for users to store all of their files and applications on their devices.

To address this challenge, we aim to provide a platform that allows users to access virtual cloud-based storage, where they can install and use apps in real time without investing physical memory on their devices. This means that users can access a wide range of applications without worrying about running out of storage space on their devices.

By providing a virtual storage platform, we aim to reduce the reliance on physical memory storage and enable users to access the applications they need without any constraints. Our platform offers a scalable and cost-effective solution for users who require access to a large number of applications without having to worry about storage limitations on their de- vices.

Overall, we believe that our platform has the potential to revolutionize the way users access and use applications on their devices, making it easier and more convenient than ever before.

Cloudroid uses container-based technology to simulate Android OS on mobile/desktop devices so that users can access a virtual smartphone as and when required without any hardware restrictions provided they are connected to an active internet service.

Keywords:

Babbage Difference Engine, Cloud-based storage, Cloudroid, Android systems, IP Tunnelling, Remote Desk Protocol (RDP)

1. Introduction:

In today's fast-paced digital world, smartphones have become an essential tool for communication, entertainment, and productivity. However, a prevalent issue that many smartphone users face is the limited functionality and efficiency of mobile applications. This can be due to a range of factors, including alack of hardware resources like memory, or out-of-date operating system versions.

One of the most common problems that users faceis running out of storage space on their devices. As users install more applications and generate moredata, the limited storage capacity of smartphones canquickly become a major constraint. This can cause devices to slow down, and even crash, making it challenging for users to access the applications they need.Another issue is outdated operating system versions. As mobile technology evolves, developers arecontinuously releasing new versions of operating systems that provide improved functionality and security features. However, many users are often unableto update their operating systems due to hardwarelimitations or compatibility issues with older devices.These challenges can severely restrict the functionality and efficiency of mobile applications, causingfrustration and inconvenience for users. Fortunately,there are solutions available that can help alleviatethese issues. For example, cloud-based storage solutions can provide users with additional storage space,enabling them to store more data and run more applications on their devices. Similarly, updating tothe latest operating system version can provide userswith access to new features and improved security, enabling them to get the most out of their devices.

The emergence of smartphones has revolutionizedthe way we interact with technology, allowing us to communicate, work, and access information on-the-go. However, as technology continues to advance at an exponential pace, the lack of the latest features and computational power can quickly render a smartphone obsolete. This creates a strong incentive for users to upgrade to newer models, leading to a constant stream of new device releases and a booming smartphone industry.

One of the main reasons why users upgrade tonewer smartphones is the desire for greater and faster computational power. As we rely on our smartphonesfor an increasing number of tasks, from social media and streaming to productivity and gaming, we need devices that can keep up with our demands. This means that processors, RAM, and storage must constantly improve to provide faster speeds, better performance, and more capacity.

Despite the benefits of upgrading to newer smartphones, there are also drawbacks.

Upgrading frequently can be expensive, both in terms of the cost ofthe device itself and the environmental impact of producing and disposing of electronic waste. Addition- ally, newer models may not always be fully compatible with existing technology, leading to frustration and inconvenience for users.

The constant desire for greater computational power and the latest technology drives users to up-grade to newer smartphones frequently, leading to a competitive market and a booming industry. While there are benefits to upgrading, it is important toconsider the potential drawbacks and weigh the costsand benefits before making a purchase.

According to a survey conducted by Statista, 6.92 billion people own a smartphone, which is more than two-thirds of the total world population. Out of this 6.92 billion, nearly 62% change their smartphones every three to four years, that is nearly 650 million kilograms of electronic waste generated every 4 years.

The primary reasons for switching to newer models are:

1. Need for better performance
2. Screen Damage
3. Keeping up-to-date
4. Smartphone no longer works
5.Prettier smartphone
6. Current smartphone too slow
7. Image Quality
8. App Compatibility

Cases where, the smartphone is physically damaged are unavoidable and the user will have to buy a new one but other factors like the need for better performance and current smartphones being too slow can be avoided up to a considerable extent by the use of Cloudroid.

2. Body:

The motto of Cloudroid is “Install once use Everywhere”. We decided to come up with a solution that will provide user to install applications simultaneously without consuming any extra space in the user’ssmartphone. We aim to reduce the need for frequent changing of smartphones due to the need for better computing power and also enable non-android users to use Android applications without owning an Android smartphone.

The application will act as a gateway between the user and a virtual smartphone that is deployed on the cloud. Each virtual smartphone instance will have the latest Android OS with allocated storage memory where the user would be able to install other Android applications as per the user's requirement.

A new instance of the virtual smartphone will be created when the user registers into the Cloudroid platform using the user's email. The user will be required to protect the user's account using a strong password with general validations and a minimum length of 10 characters. The email id will be used as a key to encrypt the virtual smartphone of the user in order to protect it from malicious activities and to maintain the privacy of the user's data.

Cloudroid uses system containers to emulate Android systems. This method results in an isolation and security level similar to virtual machines with an overhead as low as process containers. Cloudroid delivers mobile applications that don't depend on the capabilities of your user's devices and that offload compute, storage, and energy-intensive applications from the end device to the cloud.

The virtual smartphone screen will video streamed to the user in real-time and the user's gestures i.e., tap events, click events, location service, motion of the device, etc. will recorded and sent to the server using APIs and the required action will be performed on the virtual smartphone.

The server will be responsible for maintaining the hardware, creating new containers, managing the memory allocation, and maintaining the security of the overall application. The server will require the capabilities of load balancing and image compression for the video streaming of virtual devices. A constant and high bandwidth internet connection is required to the server to efficiently provide service to the customers. The server will have to use optimized video streaming techniques like transcoding and adaptive bitrate streaming so that the user will be able to access Cloudroid using low bandwidth connections.

3. Architecture:

The system consists of 2 parts:

1. Handheld Device
2. Cloud

3.1. Handheld Device:

The user interacts with a device where the Cloudroid application is installed, allowing access to Cloudroid's cloud-based Android platform. The application re-retrieves data from all available sensors on the device and sends it to the user's designated container in the cloud. This enables the user to access and analyze their device's data remotely, providing greater flexibility and accessibility. The Cloudroid application serves as the bridge between the device's sensors and the user's cloud-based container, making it an essential tool for managing and analyzing device data

3.2. Cloud:

The cloud consists of containers. Whenever a user opens the app allocated container will start. These containers will have a x64-based Android version to run smoothly on the cloud. Each user has a separate container. these containers will process the sensor data send by the application and perform actions on the respective containers. Real-time update of this system is sent back to user.

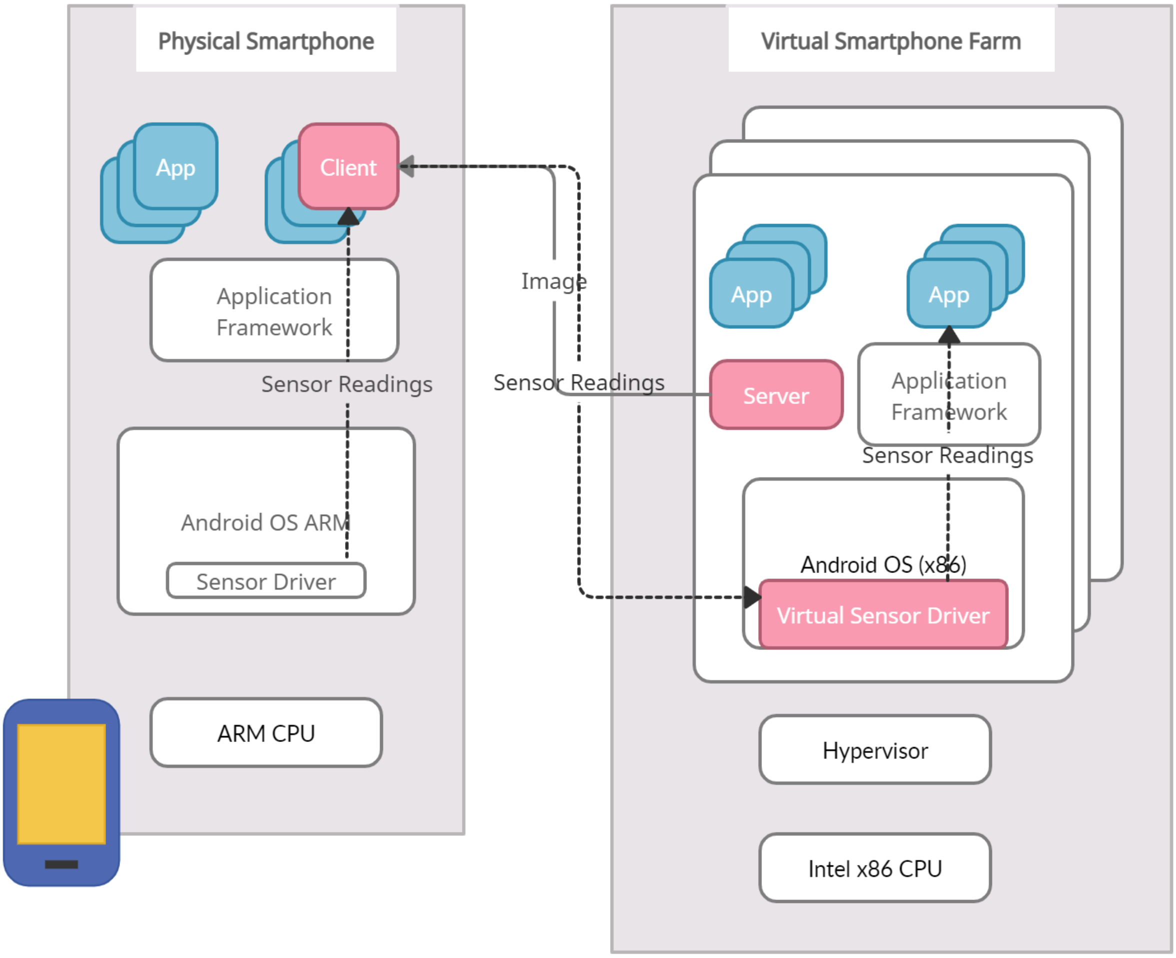

Figure.1: Architecture diagram

4. Literature Review:

4.1. Android as a Server Platform:

Summary: This paper presents several case studies where Android devices were used as servers, such as a mobile web server, a file server, and a sensor data collection server. The authors also discuss the security implications of using Android devices as servers and propose solutions to address potential security issues.

The paper concludes by highlighting the benefits of using Android devices as servers, including cost-effectiveness, scalability, and flexibility. The authors suggest that using Android devices as servers can bea game-changer in the server industry, and the potential applications are endless.

Overall, this research paper explores the potential of using Android devices as servers and presents several case studies to support the authors' argument. The paper provides valuable insights into the benefits and challenges of using Android devices as servers and can be useful for researchers and practitioners in the field of server technology.

Gap: This app has a unique restriction of beingaccessible to only one user at any given point in time,which means that if two or more users attempt to access it simultaneously, an unavoidable conflict arises,resulting in the need for one of the users to either relinquish their access or for the app itself to be re-named to allow for both users to access it separately.This peculiar characteristic can lead to potential issues and inconvenience for users, and it is crucial for individuals to be aware of this limitation when utilizing the app to ensure smooth and uninterrupted usage.

4.2. Cross-Platform Remote DesktopSharing with IP Tunnelling:

Summary: In this paper, the remote desktop application with IP Tunnelling has been proposed ina cross-platform solution to access and use remote servers and their use in the field of education, IT, IoT, and embedded systems for carrying out supplication-oriented actions from a client host to the server.

The paper provides a detailed description of the architecture and working of the system, highlighting its features such as encryption, compression, and multi-platform support. The authors also present the experimental results and evaluate the performance of the system in terms of latency, throughput, and resource utilization.

Overall, the paper proposes a viable solution for remote desktop sharing that overcomes the limitations of cross-platform compatibility. The system has potential applications in various fields such as education, telecommuting, and remote collaboration.

Gap: To elaborate further, the software or application is designed to establish a secure and re-liable connection between two geographically separated computers, enabling a seamless exchange of data and information. Unlike cloud-based solutions, this application does not rely on external servers or third-party storage providers for storing data, ensuring complete control. The absence of cloud storage also eliminates concerns related to data breaches, network downtimes, or compatibility issues, providing a more robust and efficient solution for inter-computer communication.

4.3. Android-Based Remote Desktop Client:

Summary: The paper details the architecture and design of the remote desktop client application, which is built using the Android platform and leverages the Remote Desktop Protocol (RDP) for remote access. The authors discuss the key features of the application, including the ability to connect to remote machines, view and control the desktop remotely, and transfer files between the local and remote machines. Overall, the paper offers insights into the development and implementation of a remote desktop client application for Android-based devices, highlighting the benefits it offers and its potential as a useful tool for remote access and control.

Gap: The software application enables users to access the Windows operating system on their Android device, providing a seamless platform that seamlessly integrates the functionalities of both systems. How- ever, while utilizing the software, users may find that the cursor is unable to be moved freely as it would be on a traditional computer system. Instead, users must rely on arrow buttons provided within the software to navigate the interface and control the cursor's movement on the screen. This feature, while not as fluid as a traditional computer system, pro- vides a functional workaround for users seeking to access Windows applications and files from their Android device.

4.4. A framework for executing Android applications on the cloud:

Summary: The paper discusses a framework for executing Android applications on cloud servers, which aims to address the limitations of running resource-intensive applications on mobile devices with limited processing power and memory. The authors propose a solution that offloads the processing of Android applications to cloud servers, allowing users to access the applications from their mobile devices.

The authors describe the architecture of their framework, which consists of a client component running on the Android device and a server component running on the cloud server. The client component captures the user's input and sends it to the server component, which processes the input and sends the output back to the client component for display on the mobile device.

The paper concludes with a discussion of the future directions for the research, including the potential for integrating the framework with other cloud-based services and expanding its capabilities to support a wider range of Android applications.

Gap: Emulators when compared to virtual ma- chines are extremely slow and Emulation falls short of virtualization as far as backup and recovery are considered.

5. Conclusion:

Our upcoming Android-based application will revolutionize the way Android users access storage from the cloud. Instead of relying on physical memory on their devices, users can simply access our cloud-based storage through our application, thereby freeing up precious space on their devices. This application is incredibly user-friendly, as it eliminates the need for users to constantly delete important files or other applications to make room for new ones.

With our application, users can enjoy additional storage space without the need for external hardware devices. This innovative solution will overcome the storage shortage problem faced by many smartphone users, particularly those whose devices have limited storage capacity. As a result, the functionality and efficiency of mobile applications will no longer be severely restricted by limited device storage.

Not only will our application solve storage problems for users, but it will also help reduce the need for frequent smartphone upgrades. Many users currently upgrade their smartphones primarily due to storage limitations. By providing an alternative solution, our application will significantly extend the life of existing devices, saving users money and reducing e-waste generated every year.

We believe that our application will contribute significantly to the goal of sustainability by reducing the amount of excess resources used to manufacture new smartphones. It will also provide a more affordable and sustainable option for users who want to make the most of their existing devices without having to constantly upgrade.

Overall, we are excited to bring this innovative solution to the Android market and offer users a more efficient and sustainable way to access cloud-based storage.

6. References:

[1] M. Toyama, S. Kurumatani, J. Heo, K. Terada, and E. Y. Chen, "Android as a server platform," in *2011 IEEE Consumer Communications and Networking Conference (CCNC)*, pp. 1181–1185, IEEE, 2011.

[2] S. Sridhar, S. Sanagavarapu, and S. Chitrakala, "Cross-platform remote desktop sharing with ip tunneling," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1–7, IEEE, 2020.

[3] A. Kotkar, A. Nalawade, S. Gawas, A. Patward- han, and S. Mangale, "Android based remote desktop client," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 2, pp. 345–348, 2013.

[4] S. Ghorpade, N. Chavan, A. Gokhale, and D. Sap-kal, "A framework for executing android applications on the cloud," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 230–235, IEEE, 2013.

Cite as

Mrudul Borse, Akash Ranajkar, Parth Sakure*, Vaibhavi Sidam, Prof. Nikita Kapadnis. (2023). Cloudroid. https://doi.org/10.5281/zenodo.7965470