



Available Online through

www.ijptonline.com

CLOUD BASED JAVA COMPILER FOR ANDROID SMARTPHONES

Dr P. Jeyanthi¹, Nishant Rao², Ketan Ketu³

Department of Information Technology, Sathyabama University, Chennai -600119.

Email: jeyanthiprabhu@yahoo.com

Received on 04-03-2016

Accepted on 29-03-2016

Abstract

In the proposed work, cloud mobile media which is a centric multimedia streaming system. Aim of the proposed work is to provide a Streaming service of user's own multimedia contents on Smart-phone at anytime and anywhere. Cloud Mobile Media Provides Management and streaming service for user's multimedia Contents using local storage in user's Smart-phone, user's personal NAS, user's cloud storage and Cloud-Centric Media Network and Media Analytics. Cloud resource management and control in Infrastructure-as-a-Service (IaaS) and novel cloud-based systems and applications in Software-as-a-Service (SaaS).

In other words, this work investigates the feasibility of cloud mobile media to provide the Cloud-Centric Media Network and Media Analytics based mobile cloud service model. Specifically compared the energy costs for uploading and downloading a video file to and from cloud mobile media with the energy costs of encoding the same video file on a smart phone. The aforementioned comparison was performed by using HTTP and FTP Internet protocols with 3G and Wi-Fi network interfaces. The experimental results shows that cloud mobile media provides the smart phones with much multimedia functionality and saves smart phone

Keywords: Cloud-Centric Media, Infrastructure-as-a-Service (IaaS), Software-as-a-Service (SaaS).

1. Introduction

The number of smart phones and mobile applications are growing rapidly. Though smart phones are expected to have PC-like functionality hardware resources such as CPUs, memory and batteries are still limited. To solve these resources problem, many researchers have proposed architectures to use server resources in the cloud for mobile devices. We propose a conceptual architecture of Android as a Server Platform, which enables multiple Android application on cloud server via network. In this paper we elaborate on the idea of computation offloading and present a practical system, called Cuckoo that can be used to easily write

and efficiently run applications that can offload computation. Cuckoo is targeted at the Android platform, since Android provides an application model that fits well for computation offloading.

2. Literature Review

Software as a Service is a feature of cloud computing and this feature is being implemented in this project. It is nothing but providing a software without installing it on the client device or machine. The Software as a Service (SAAS) is that the software are uploaded in the cloud server, whenever the client request the software to the cloud server, the cloud server will provide the software, which is chargeable in rental manner. This process will be of use to reduce the client system load. To compile and run Java program on our computer, we need to have a working installation of the Java Development Kit (JDK) from Sun Microsystems.

Cloud computing is the upcoming area in the real Networks, but to utilize this cloud computing resource, computer like Hardware is required. Cloud Computing is not easy job to access in our mobile device. Computer system are not as handy as mobile handsets. Computer systems are not available to as many people as mobile phones. So to fulfill the requirements of the users we are implementing the java code in android smart phones using Online Java Compiler which helps us to create, compile, and run the file on the cloud server, instead of running in mobile phones. It also helps them to learn programming languages instead of learning new operating systems.

3. Proposed Systems

Cloud Computing Application can be initiated using Android Smart Phones. We are implementing Software as a Service (SAAS) for Cloud Computing. SAAS is the Cloud Computing Resource, used for the service of without installing that Software in the Device. Here, we are compiling the code using Android Smart phones without installing Software in Mobile Phone.

Implementing cloud computing architecture for mobile devices. Android can utilize software as a service (SAAS) Process from the cloud server, without installing the software in the Android mobile. This features allows students to do Java programming anywhere, anytime using just mobile interface. It allows Compiling and Executing Java programs directly through the Android mobile so that they can concentrate on the programming concepts rather than learning to operate new technologies. We will execute small Java program through Android Mobile it reduce the time consistency. This allows students can to do Java programming anywhere, anytime using just mobile interface. No need of integrated development environment (IDE) because we can directly compile the program. We no need to pay while compiling the program in our Android Smart Phone.

CLOUD COMPUTING AS SAAS & COMPILE JAVA PROGRAM



4. Algorithms Description

In this project, two types of algorithms are used which explains that how the project works using the cloud server.

Those two algorithms have been discussed below:

4.1 Load Balancing Algorithm

In this algorithm, Firstly analysis of different Virtual Machines (VM) of load balancing algorithms is done. Secondly, a new VM load balancing algorithm has been proposed and implemented for the SAAS framework in Simulated cloud computing environment, i.e. 'Weighted Active Monitoring Load Balancing Algorithm' using tools, for the data centric to effectively load balance requests between the available virtual machines assigning a weight, in order to achieve better performance parameters such as response time and Data processing time.

4.2 Greedy Method Algorithm

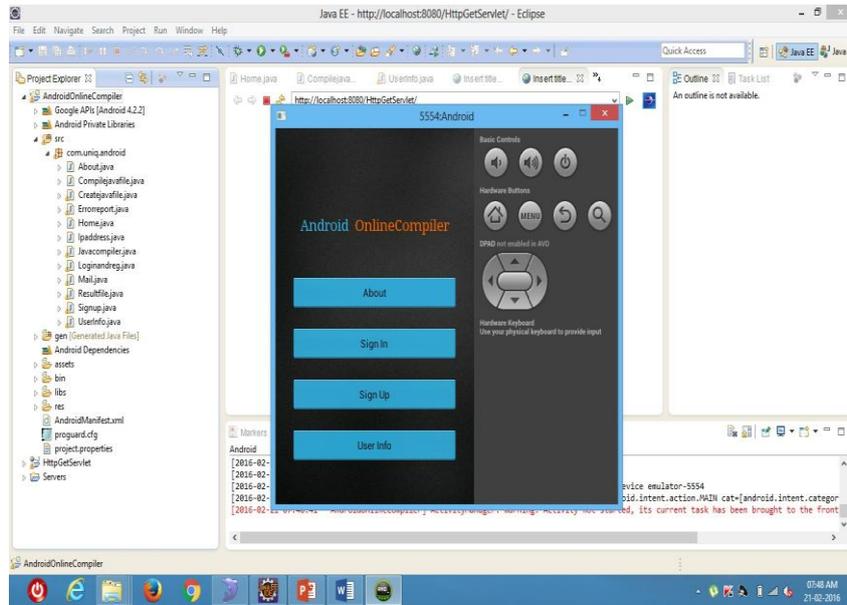
Greedy method algorithms is an approach to solve certain kinds of optimization problems. Greedy algorithms are similar to dynamic programming algorithms in that the solutions are both efficient and optimal if the problem exhibits some particular sort of substructure.

A greedy algorithm builds a solution by going one step at a time through the feasible solutions, applying a heuristic to determine the best choice. A heuristic applies an insight to solving problem, such as always choose the largest, smallest, etc..

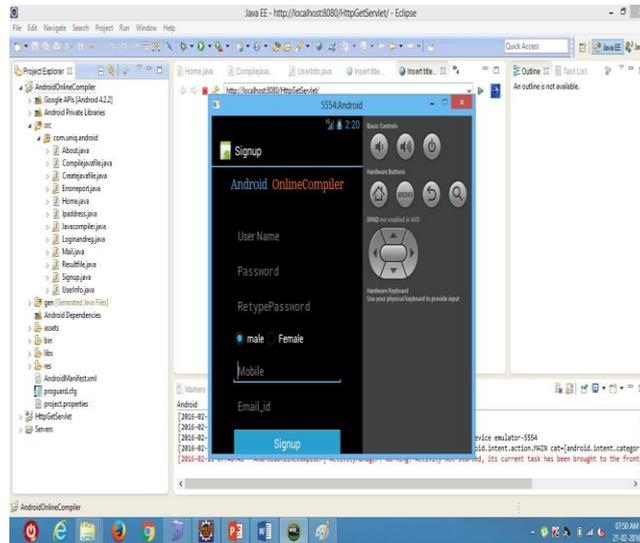
5. Experimental Results

This step is mainly used to explain the step by step process of Online Java Compiler in which all the screen shots will be displayed below:-

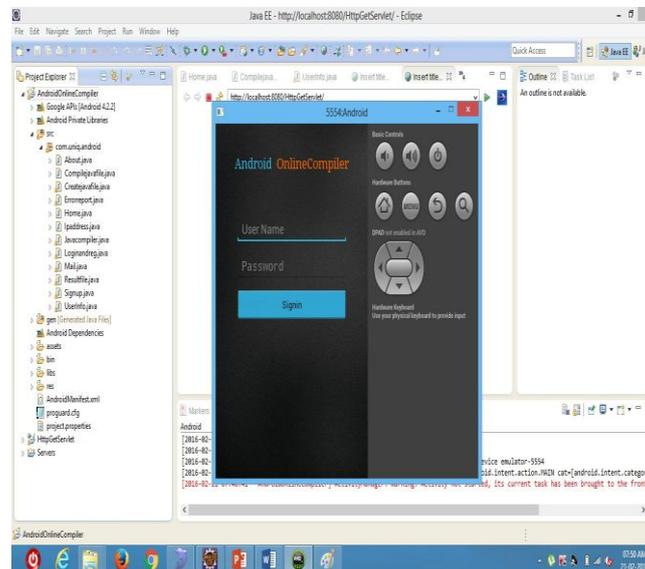
1. Home Page



2. Sign up Page



3. Sign in Page



4. Compiler User Interface



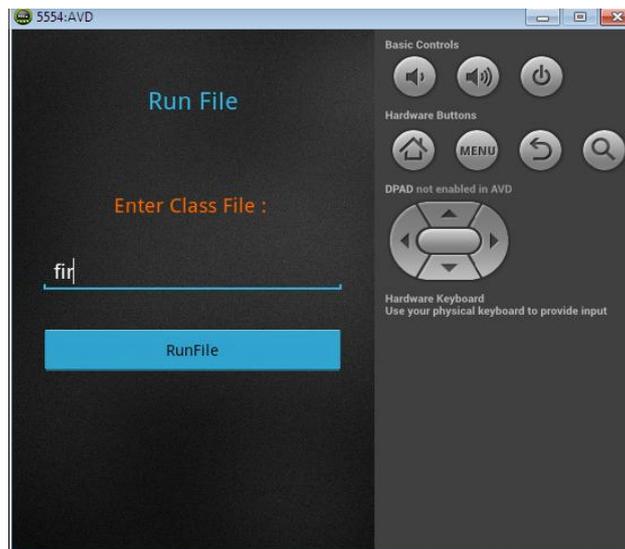
5. Creating a File



6. Compiling of the File



7. Run the File



Explanation

In the above figures, Firstly it has been explained about the home page in which all the buttons have been given for sign up, sign in, etc.. Secondly after clicking on sign up page then sign up page will be displayed in the emulator, then we can do sign up which is followed by the sign in page in which after filling the username and password then click on sign in, it will show that you have been successfully signed in. Thirdly, three buttons will be there in which it says about the creating, compiling, and running a file. After clicking on creating a file it will open a new page where we can write a program and can save the program. Fourthly we can compile the program by clicking on compiling a file button. Lastly, after compiling if error is there then we can update the program and we can save it. If no error is there then we finally run the program by giving filename.class. Hence the output screen will be opened according to the program written in the emulator.

6. Conclusions

This project helps us to compile and execute Java programs directly through the Android mobile so that they can concentrate on the programming concepts rather than learning to operate new operating systems. This feature allows students to do Java programming anywhere, anytime using just mobile interface. This project developed a prototype for android architecture for a multiple system. This proposed system gives more efficient gain when compared with the existing system.

By providing an open development platform, Android offers developers the ability to build extremely rich and innovative applications. Developers are free to take advantage of the device hardware, access location information, run background services, set alarms, add notifications to the status bar, and much, much more.

The application architecture is designed to simplify the reuse of components, any application can publish its capabilities and any other application may then set of services and systems, including a rich and extensible set of views that can be used to build an application, including lists, grids, text boxes, buttons and even an embeddable web browser.

7. References:

1. D. Kovachev, T. Yu, and R. Klamma, "Adaptive computation offloading from mobile devices into the cloud," in Proc. IEEE ISPA, 2012.
2. R. Kemp, N. Palmer, T. Kielmann, and H. E. Bal, "Cuckoo: A computation offloading framework for smartphones," in Proc. MobiCASE, 2010.
3. IBM Corporation. Power ISA v2.07, Appendix D: Example Performance Monitor, May 2015.
4. Intel ® Corporation. Intel 64 and IA-32 Architecture Software Developer's Manual.
<http://www.jikesrvm.org>
5. N. Neelakantam, R. Rajwar, S. Srinivas, U. Srinivasan, C. Zilles, "Hardware Atomicity for Reliable Software Speculation", in International Symposium on Computer Architecture (ISCA), 2007
6. C. K. Shum, F. Busaba, C. Jacobi, "IBM zEC12: The Third-Generation High-Frequency Mainframe Microprocessor", in IEEE Micro, 2015.
7. Javascript engine webpage, [https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey)
8. [US/docs/Mozilla/Projects/SpiderMonkey](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey), Mozilla
9. The DaCapo Benchmark Suite, <http://www.dacapobench.org/>
10. V8 Javascript engine homepage, <https://code.google.com/p/v8/>, Google Corporation.

Corresponding Author:

Dr P. Jeyanthi¹,

Email: jeyanthiprabhu@yahoo.com