A Cloud-based Java Compiler

Sandeep Adhikari[1], Sagar Gurung[2], Amjad PP[3], Chitra M.P.[4]

[1, 2, 3]B.E.(ISE), Sambhram Institute of Technology, Bangalore, India

[4]Assistan Professor (ISE), Sambhram Institute of Technology, Bangalore, India

Abstract: *Java programming language is in one of the most popular language in current industry. People are more excited to learn java because of its various features. But the installation of JDK or IDE de motivate the learner to learn the java. Besides that, the problem of software crash after installing into pc is one of the major problem for java learner. To overcome this problem we have introduced a cloud based java compiler called JCompiler. JCompiler is an online compiler and debugging tool which would allow users to compile and run java codes through their browser without installing java in their system. It would be a very user friendly application which can be installed on one server and can be accessed through internet with the help of any browser. JCompiler would execute code for us with a click of button. We just need to paste our code on the browser, and JCompiler would run it and show the output there itself. Or just try things out when we don't have an interpreter handy. The user need not to have expensive computer to compile the java code, instead, they can have any gadgets with the internet and web browser facility. This application will be more helpful to the java learner.*

Keywords: Cloud Computing, Mobile learning, Compiler, Java

1. Introduction

Technology plays a vital role in 21st century education. Technology transfigured the activities of this era. The way we learn the way we think and the way work [1]. Mobile learning is one of the trend of this era. It appears from the outside to be learning via mobile devices such as smart phones, laptops, Mp3 players, laptops table. It is not only about using mobile devices to access content and communicate but it is also about the mobility of learner and this is great advantage one can take from technology learn when you travels [2].

Generally, codes are written in English language. And this code should be understand by computers to interpret sequence of particular text. In other word the text should be translated from human readable from to machine readable from and is done by compiler.

Cloud computing means flexible self-service is the delivery of demand computing resources. It is a type of internet based computing where network of remote servers are used which are hosted on internet to store manage and process data instead of using personal computer or local server[3].

Having identified the vital role that the technology has played in online learning with access of internet. We are motivated to develop the online compiling application. This research focuses on the development of cloud based (IDE) to compile and run java program from any kind of devices with the faculty of internet access and web browser facility [1].

2. System Description

The system developed will have a built in editor where user can edit or write the java programs. A cloud-based server is to be developed as software as a service to host the compiler that compile and executes user codes . Any IDE can be placed as a compiler. The system will be developed with web based user interface that can be accessed via browser from pc or any kind of device with the access of internet.

A. Compiler

Compilers are used to compile the programs and convert them into one from to another i.e. human readable form to machine readable form or say from written program to executable binaries . A compiler can be divided in two phase on the way they compile, i.e. Analysis phase and Synthesis phase.

In Analysis phase, the compiler will read the source program and divided into core parts and checks the lexical, grammar and syntax error where as the synthesis phase generates the target program with the help of intermediate source code representation.[4]
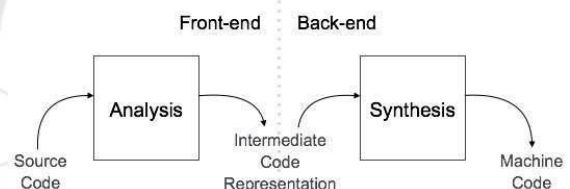

Figure 1: Phases of Compiler

B. Common Cloud service model
Based upon the services offered, clouds are classified in the following ways:

Software as a Service (SaaS)
It includes a complete software offering on the cloud software as a service allows organizations to access business functionality hosted by the cloud vendor on Pay-per-use basis. ex: gmail
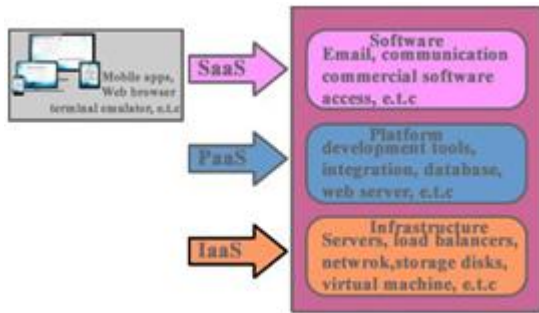
Platform as a Service (PaaS)
A developing platform being delivered as a service. Ex: Google App Engine

Infrastructure as a service (IaaS)
Computer infrastructure such as virtualization on an outsourced basis is being delivered in this model.
Ex: Amazon EC2

Figure 2: Cloud services

3. Functionalities of the System

3.1 User module

The user have to open our application form web browsers and need to sign up with the credential details mentioned in the sign up page. After the registration process is completed the user need to sign in, providing valid username and password then the user is permitted to the dashboard page, where he/she can actually compile there code. This module is necessary because to give the security to the user file. In other words, the user can only access their previous file only after the sign in process.

3.2 Remote Command Processing

In this module we create an architecture where multiple clients connect to a server and send command processing request to the server side scripts. Under this mechanism we create a thin layer of isolation where only UI codes are kept on client side and Interface provides user to just write the type of program they want to execute and the program. All the command processing happens on server side remotely, so there is no need of installing any software on client machine. Even command line arguments can be provided by user in this online application.

3.3 Online Compiler and Executions

All the tasks related to compiling and execution takes place on server side and thus we need to install all the required software's on server system. Our application architecture is designed in J2EE where users can register and login through the server URL using system browser. Once user authentication is completed, user can write programs based on java and which can be executed remotely on server. Here client need not have java installed in their system.

3.4 User History Maintenance

This function takes care of user's file history. For each program user makes are get saved in the user's personal folder on server. So, in case if user wants they can open their previous programs and can open it and re-operate on it. Users can see only their programs and as user opens any program, it gets copied to their IDE and can be used for further processing. Sometimes this type of feature is much required when user gets exception in program and after working for long time want to take break.

3.5 Result Generation on Client Side

Through this module results on java are sent to the client browser and user can review the output. For each input provided by user server first processes the source file, converts it to machine understood language and return results to the user system.

4. System Architecture

The proposed system operates server/client architecture. The client requests for services from the server which forwards the request to the cloud, where IDE is present. The response is returned to the server which interprets it and sends the appropriate result back to the client. The figure below describes the architecture of the system.
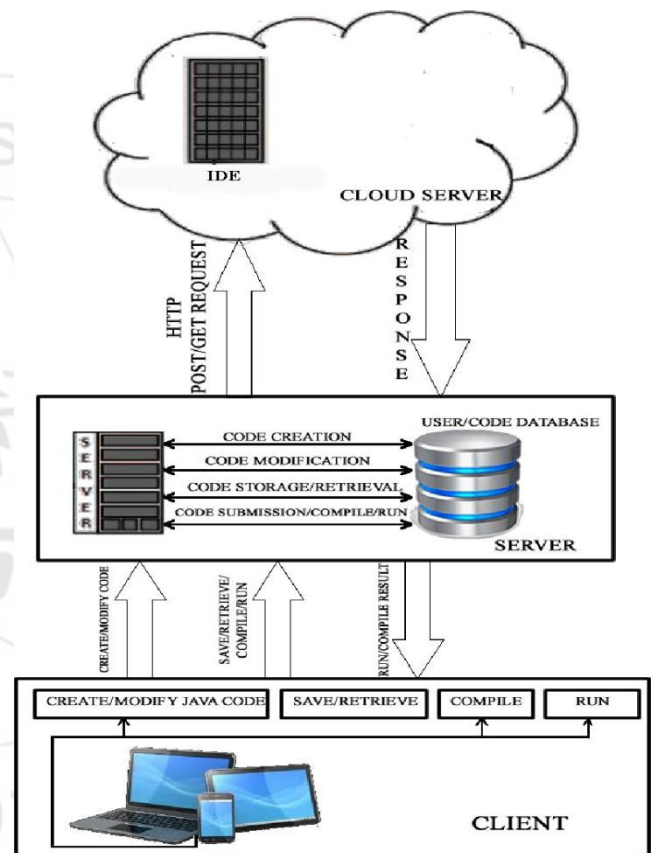
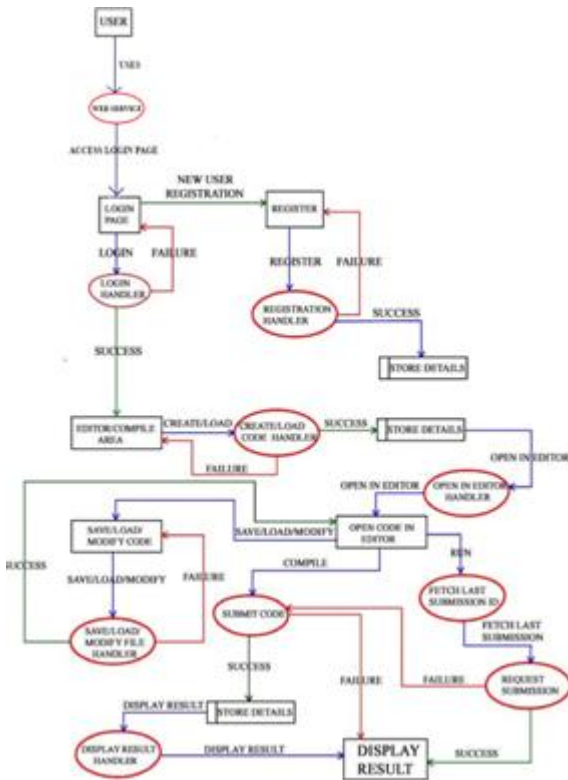**Figure 3:** Architecture of the system

5. Data Flow Diagram

Figure 4: Dataflow diagram of the system

The data flow diagram shows the how user access system from web based UI. The interaction from with cloud compiler and the database for storage are shown.

6. System Requirement

A. Functional Requirements

Development of the android and web application
The system requires a web browser on a PC and non-android devices, a web server to host the server side program that interfaces between the user device and the cloud server and finally a cloud server for does the main processing. The user writes and edits the java programs on his device and send to the server. The server forwards the request to the cloud server which does the processing and returns the response to the web server which interprets the result and sends to the user for display on his device.

Development of the server
The server is created for interfacing between the user and the cloud server. The server requests the cloud server to execute user''s codes and interprets the returned response and sends the result to the user. The server also handles storage of user''s codes, compilation and execution information. A web browser on a PC and non-android devices, a web server to host the server side program that interfaces between the user device and the cloud server and finally a cloud server for does the main processing. The user writes and edits the java programs on his device and send to the server. The server forwards the request to the cloud server which does the processing and returns the response to the web server which interprets the result and sends to the user for display on his device.

B. Other Requirement

Communication requirement
A user is required to have access to the internet via Wi-Fi networks, LAN networks or any other network to use the system.

Database requirement
The database system used is MySQL. PHP''s strong integration with MySQL makes it very much desirable for this system. It is very impressive how easily the two can work together.

7. Implementation

The web browser interface was designed using J2EE, CSS and JQUERY. The EditArea plug-in were integrated with other HTML elements to produce the interface for the system. A responsive HTML theme was used to make the system good and easy to use regardless of the device (non-android devices) used to access the system.
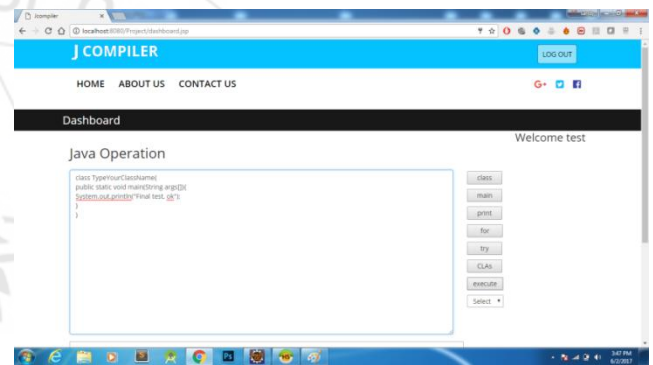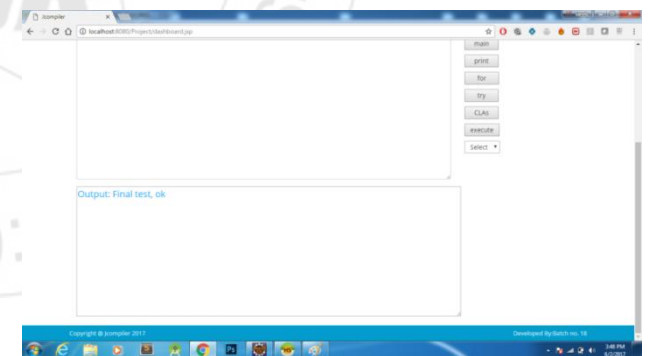

Figure 5: edit area of the application

Figure 4: Output region of application

8. Testing

The server side that mediates between the client (user device or web browser) and the Cloud-based java server (IDE) was developed using the server side language Java Server Page (JSP). It was deployed to an WINDOWS server and tested with no known issues.

The web based system was accessed and tested on the windows 7 environment using with Chrome 14+, Firefox 3.5+, Opera 12+, Safari 4+ with no bugs detected. The system is expected to work fine on other operating systems with the same browsers.

9. Limitation of the System

Execute Limited Language
Our system compiles and executes only java language. The language like C, C++, Perl, ruby cannot be compile or execute.

10. Future Works

Implement a Smart Multimedia Learning System for Java programming language: The "Cloud-based Java Compiler for Smart Devices" developed is one component of a proposed Smart Multimedia Learning System for Java programming language. The other components of this Smart Learning System could be developed and integrated with the already developed Cloud-based Java Compiler for Smart Devices.

Extend to include other programming languages:
The implemented system was strictly for the Java programming language alone. In future works compilers for other programming languages such as C, C++, Python, e.t.c. could be developed and integrated into this system.

11. Conclusion

In conclusion, We propose a very user friendly J2EE based application which can be installed on one server and can be accessed by any of the LAN systems or internet with the help of any browser. JCompiler would execute code for us with a click of button. We just need to paste our code on the browser, and JCompiler would run it and show the output there itself and even give us a short link to download the file to our system.

This application is a new concept and has a lot of scope for up gradation. This application can be enhanced to integrate more programming languages such as C++ and C#. Also this application supports only standalone application programming so it can be further enhanced to support web programming also using PHP, HTML or JSP.

References

[1] Tanko Y. Mohammed; Mohammed Hamada 2016 15th International Conference on Information Technology Based Higher Education and Training (ITHET). "A cloud-based Java Compiler for smart Devices". IEEE spectrum. DOI 10.1109/ITHET.2016.7760742
[2] https://www.jisc.ac.uk/guides/mobile-learning/what-is-mobile-learning
[3] http://www.webopedia.com/TERM/C/cloud_computing.html
[4] https://www.tutorialspoint.com/compiler_design/compiler_design_architecture.htm