AHMEDABAD UNIVERSITY | School of Engineering and Applied Science

Cloud Computing

Title: Cloud Based Compiler

Mentor: Dr. Sanjay Chaudhary

Members:

1401025 – Shreyas Patel

1401075 – Ashutosh Kakadiya

1401076 – Mihir Gajjar

1401086 – Harsh Mehta

Introduction

Build cloud based web application that provides developer an interface which helps in code compilation, execution, analysis and also provides code optimisation related suggestion if required in code.

Work Done

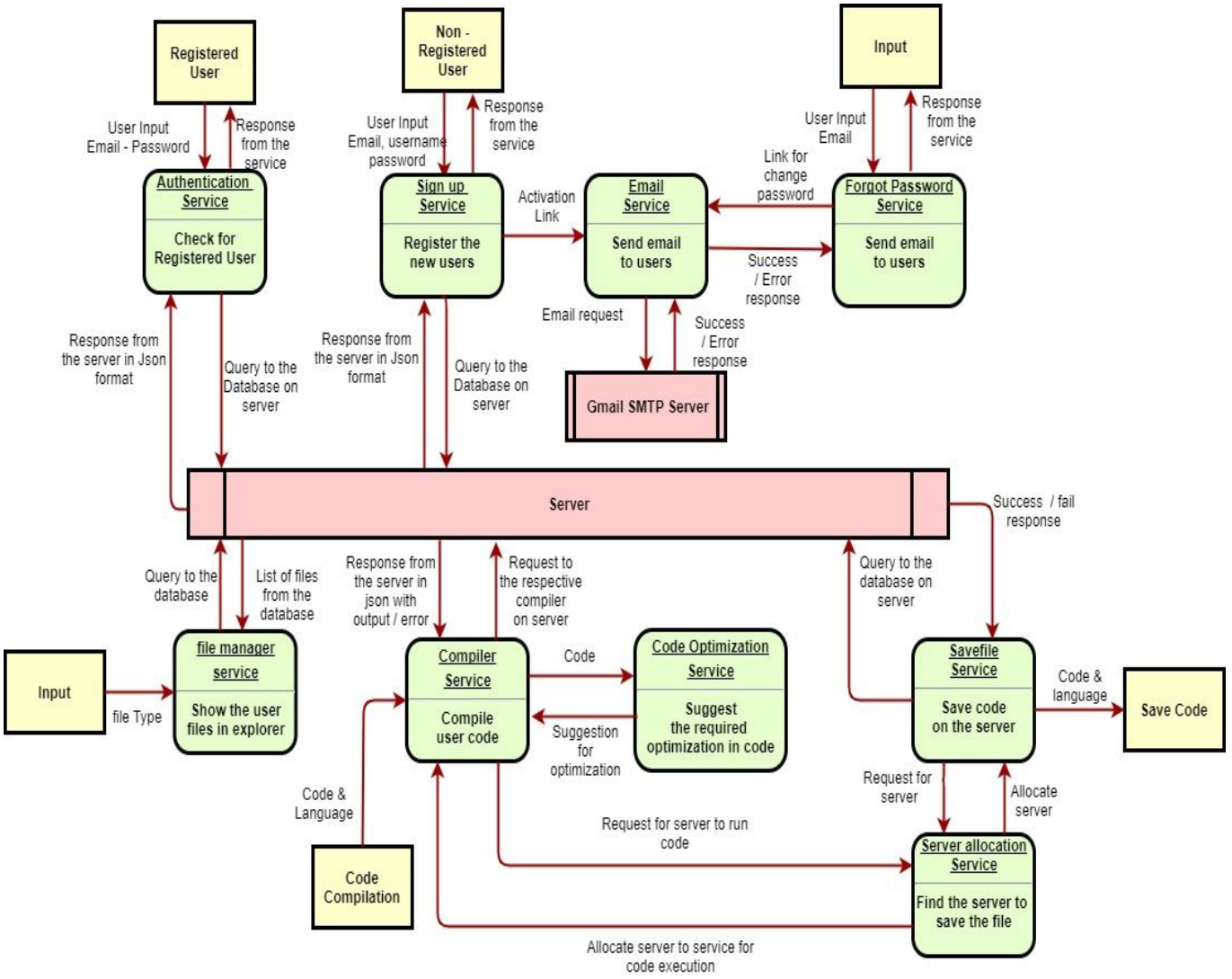
1. Successfully integrated compilers at server side
2. Created service oriented architecture for Code Editor
3. Implemented authentication service for user SignIn / SignUp
4. Designed a database and integrated it with the system
5. Implemented file storage system – Users can store their code on cloud
6. Users can show their activity on their dashboard
7. Developed an optimization service

Tools / Packages

- Front-end technology
    i. HTML , CSS, Javascript
- Back-end technology
    i. Python, Flask, Jinja
- Database
    i. MySQL

Dataflow Diagram

- This diagram shows how data flows in the system



Description of the services

1. Authentication service:

This service compares the credentials provided by the user with the database of authorized users' information. If the credentials match, the process is completed and the user is granted authorization for access. If the credentials do not match, the user is not granted access and and is requested to enter valid credentials to gain access.

2. Sign-up service:

If the user is not registered, then the user can use this service and enter his details (sign - up) which will be added to the database of authorized users. This service provides the email service with the activation link that the user can use to complete the registration process.

3. Email service:

This service, using the Gmail SMTP Server, sends emails to the users containing the link for completing the registration process or for changing the password in case the user forgot the password.

4. Forgot password service:

In case the user has forgotten the password, this service can be used to change the password. It provides the email service with the link that the user can use to change the password.

5. File manager service:

This service shows the user all the files that the user has saved. In addition, it also provides the user, specific types of files which the user wishes to see.

6. Compiler service:

This service performs the primary function i.e. it performs the compilation of the codes for the users. It requests the server allocation service to allocate a machine for the compilation of the code. The user provides the service as an input: the code and the language in which it has to be compiled. If the compilation of the code was successful, then this service calls the code optimization service which takes the code as an input and provides suggestions for optimization. In this case, the output of the code and the suggestions for optimization of the code are both returned to the user. If the compilation of the code was not successful, it returns the error which occurred during the compilation process.

At this stage, the compiler service compiles the code for C, C++, Java and Python.

7. Code Optimization Service:

This service takes the file containing the code as an input, analyzes the code and provides suggestions, which if implemented can optimize the performance in terms of the memory usage or the run-time of the code.

At this stage, this service optimize the code for Java and Python.

8. Savefile service:

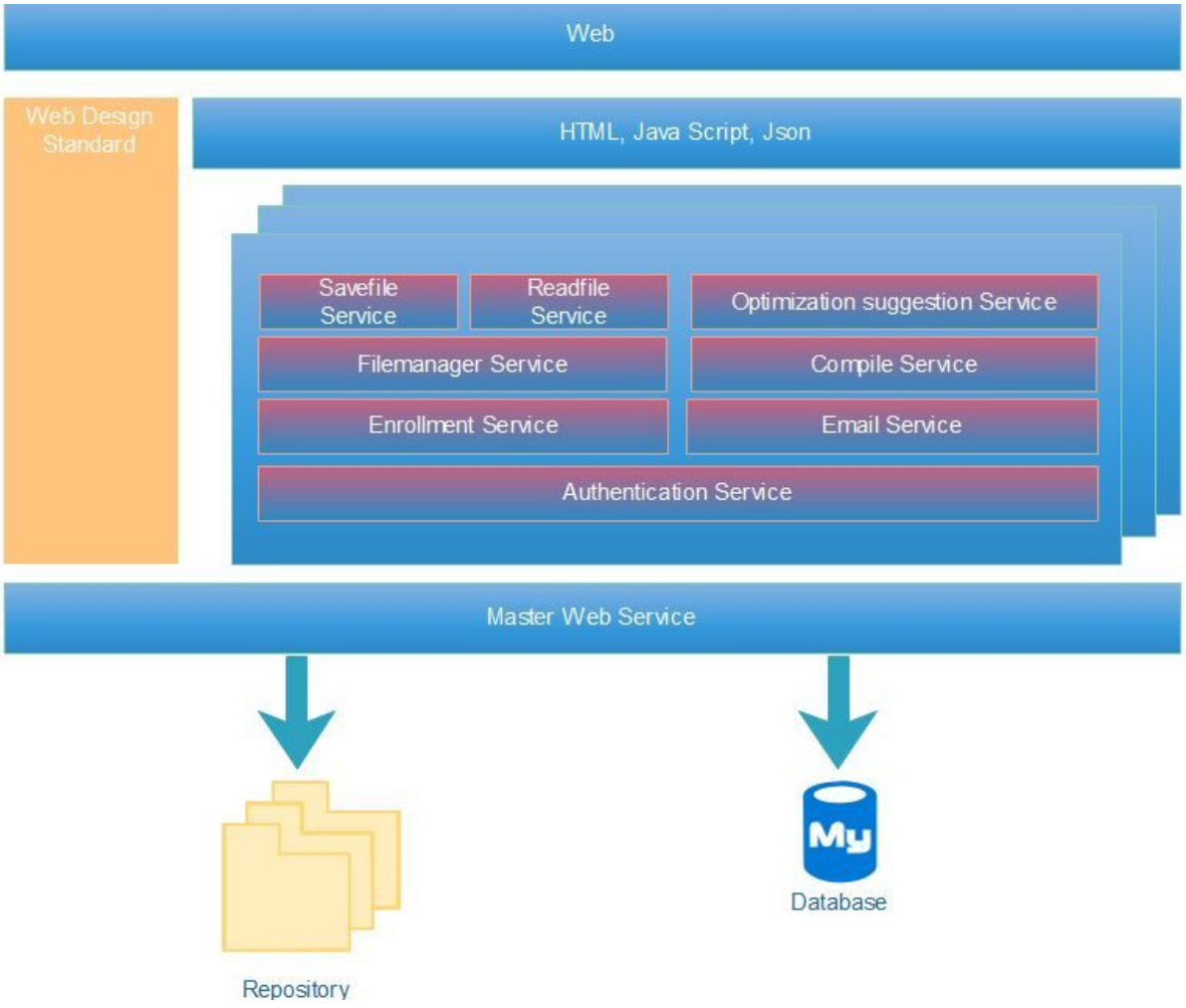
This service is used to save the code entered by the user in the database so that it can be accessed in the future. It requests the server allocation service to allocate a machine in which the code can be stored.

9. Server allocation service:

This service performs the function of allocating the machines and distributing the jobs for compilation and data storage. It checks the status for memory usage and storage availability in various machines that are available on the server side and allocates the machine which will be most suitable for compilation of the code or for storing the file containing the code.

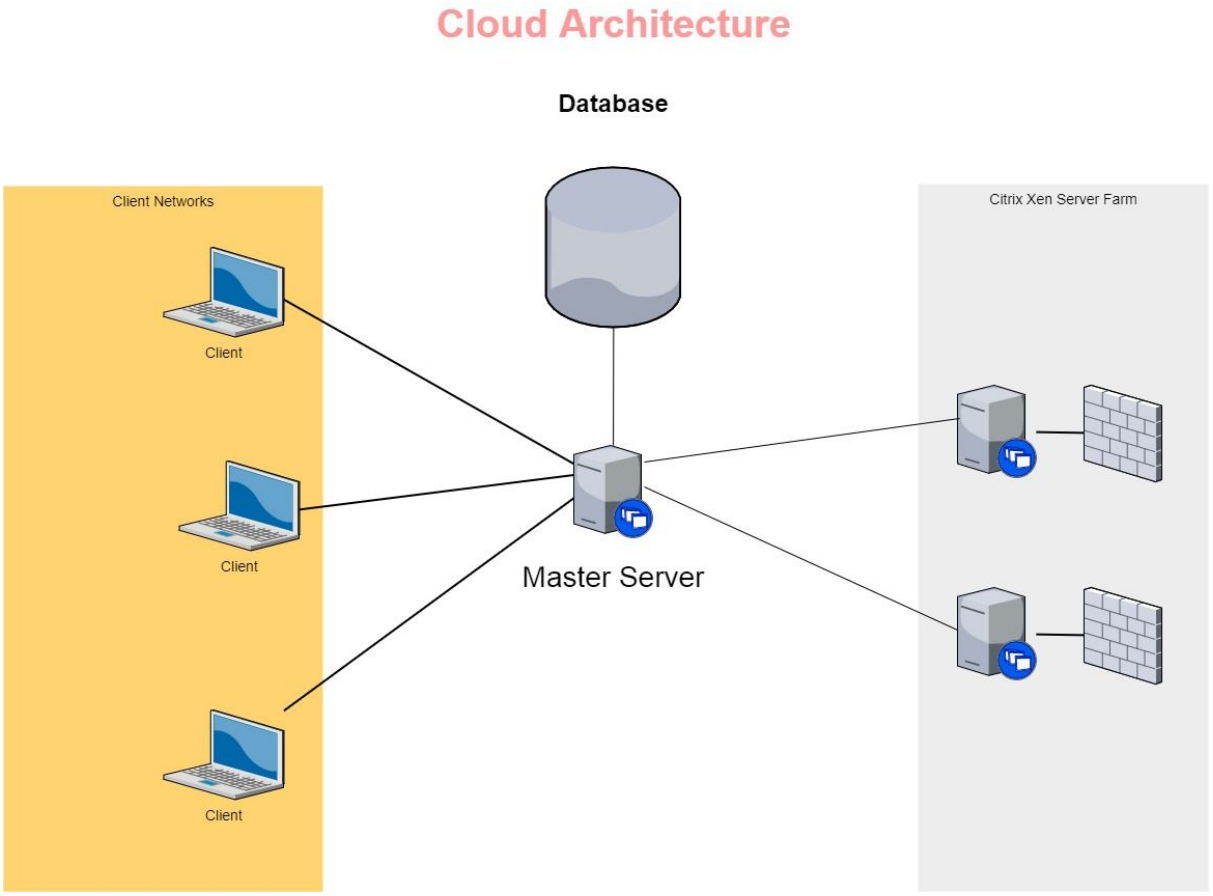
Application Architecture

- Below architecture show how our application is organized

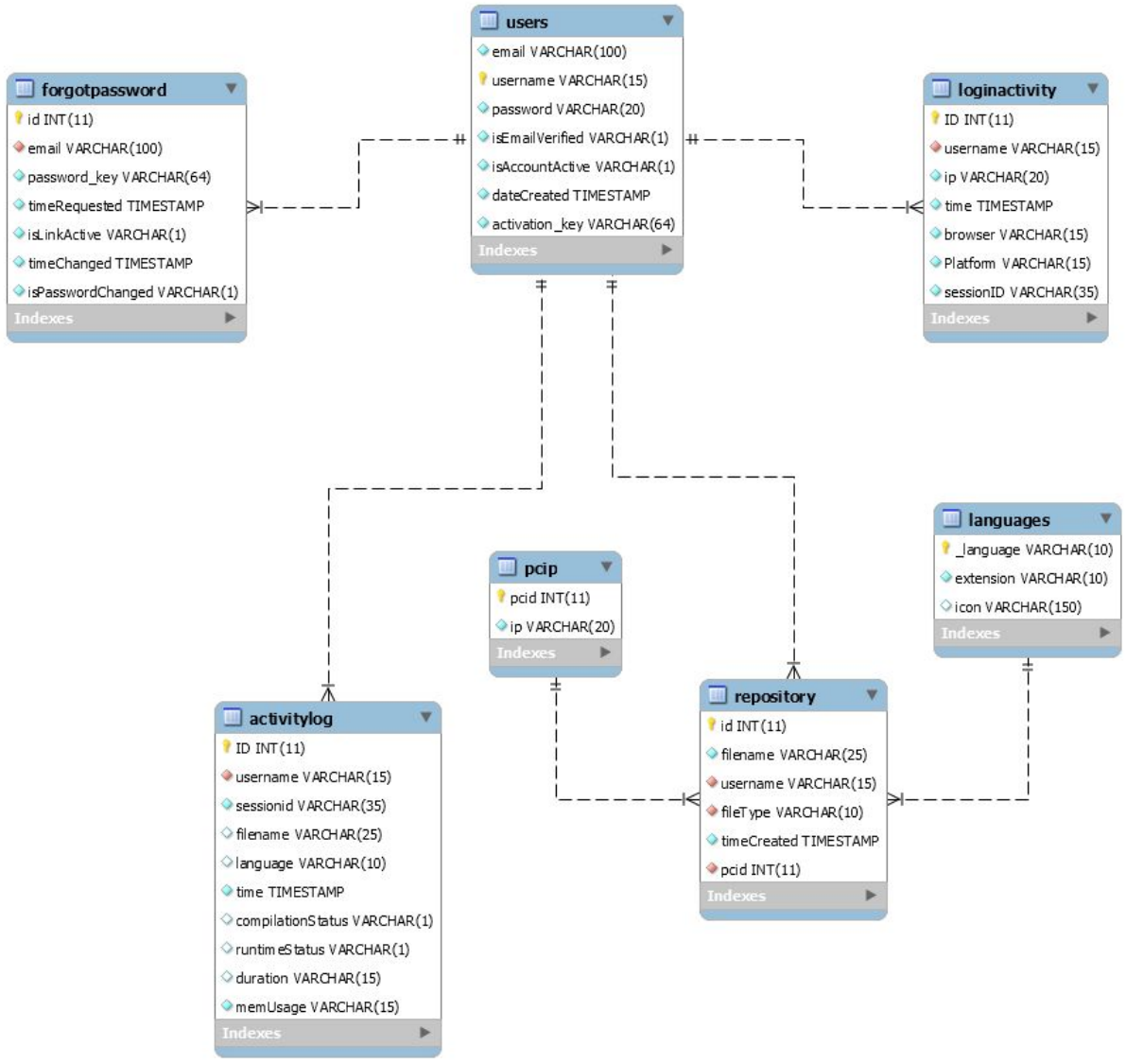


Cloud Architecture

- Below architecture show how our cloud is organized

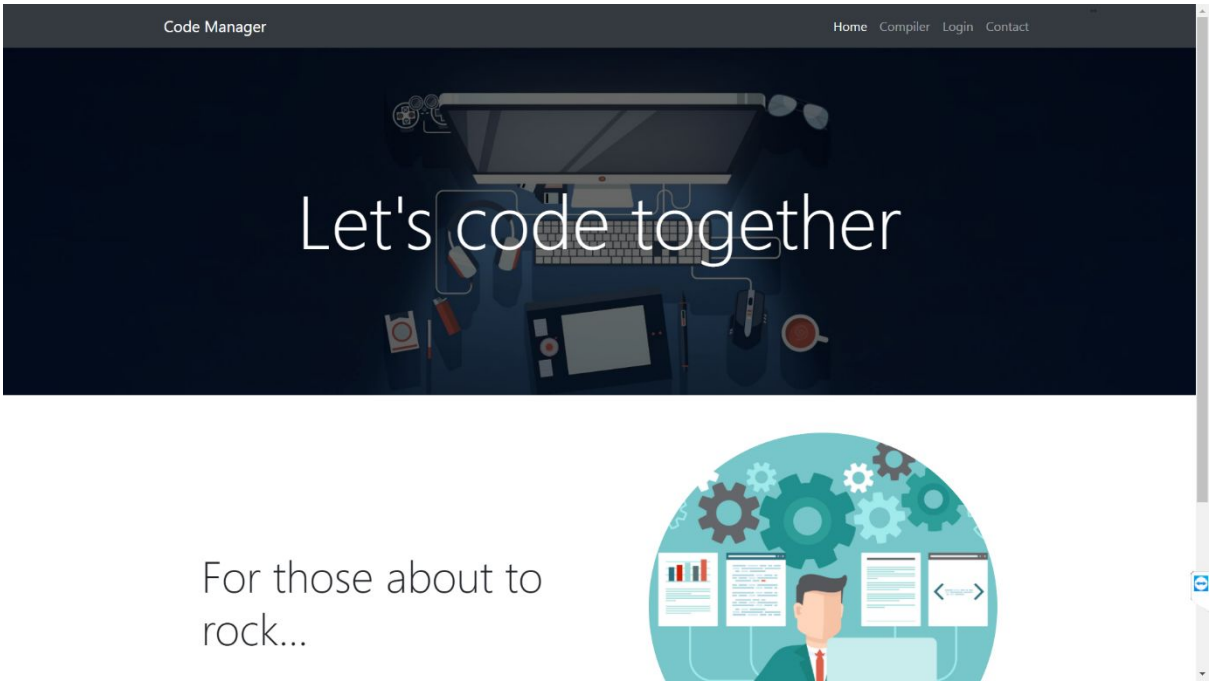


Database Schema

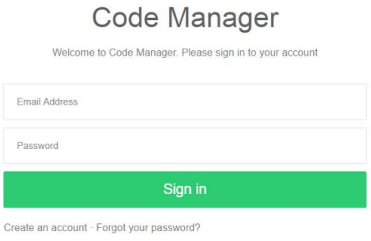


Screenshots

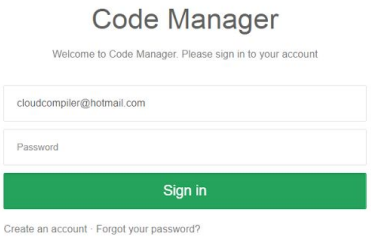
- Home Screen



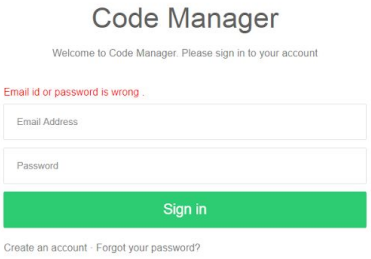
- Login Screen



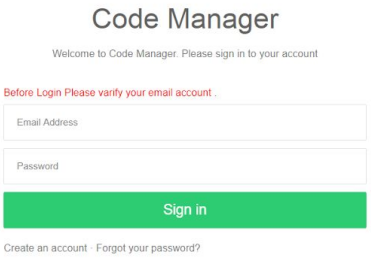
- Error based on user side input validation



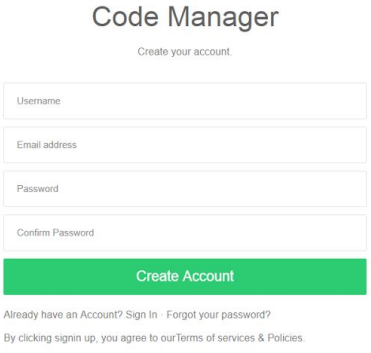
- Error based on server side validation



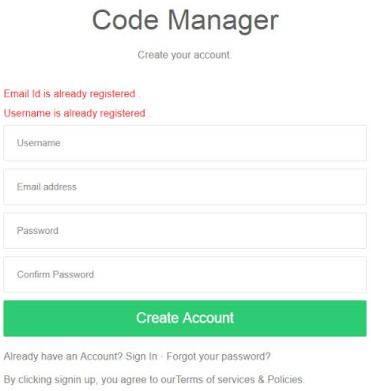
- User need to verify the account before login by clicking the link sent to user's mail account



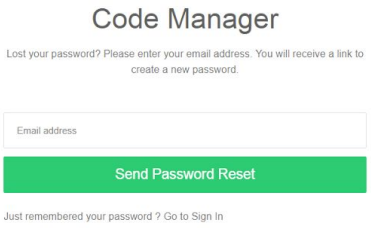
- Sign up screen



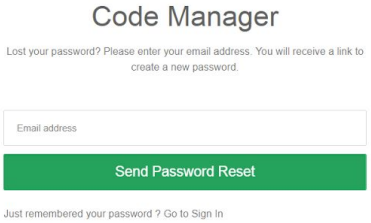
- Server side validation for sign up



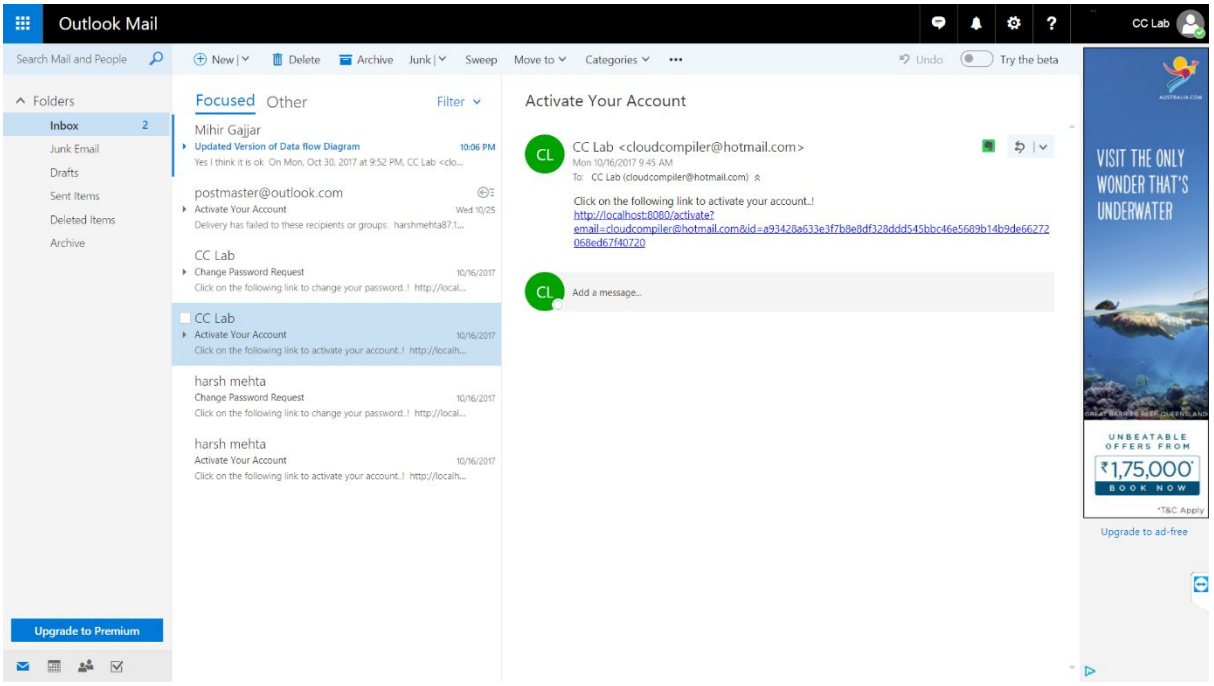
- Forgot password Screen



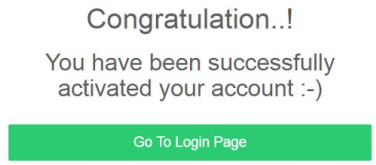
- User side validation for forgotpassword



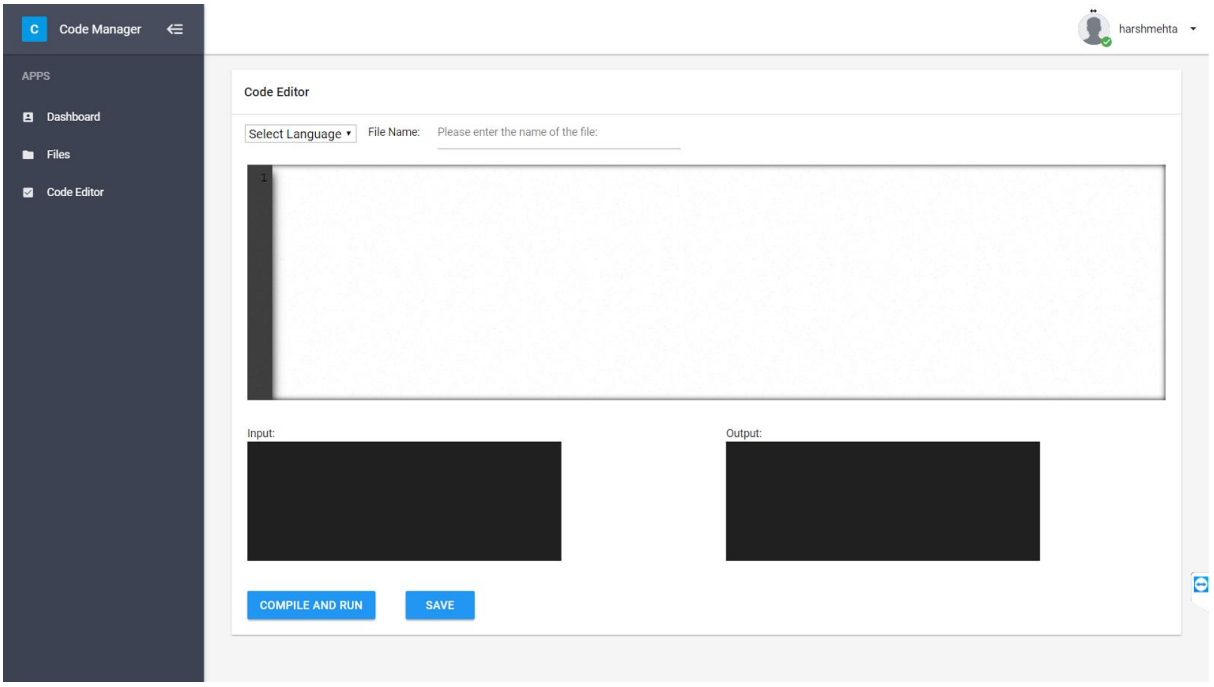
- Link sent to user to activate account



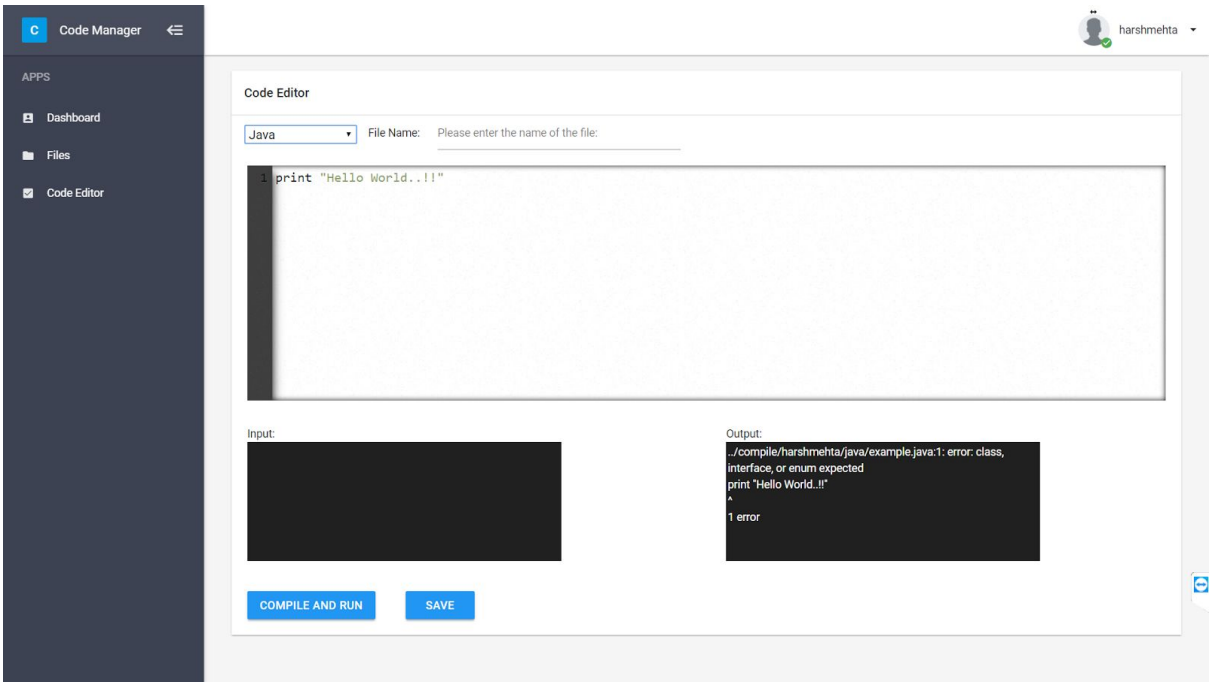
- When user go to the link for activation of an account



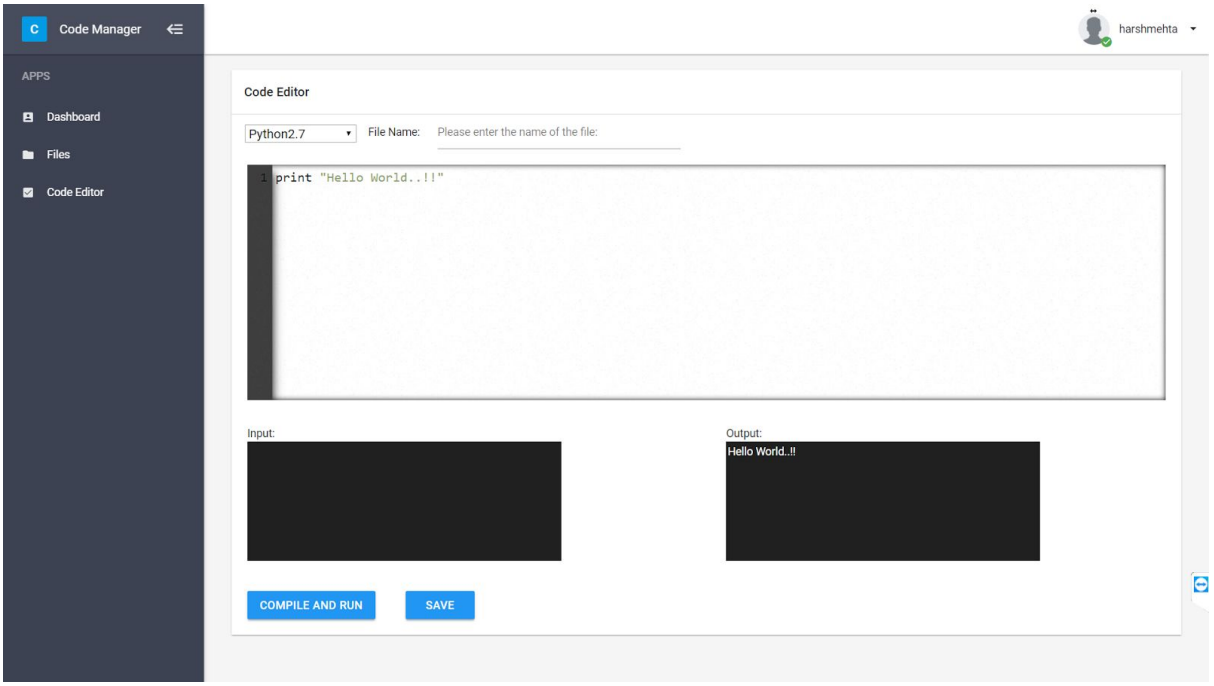
- Code Editor screen



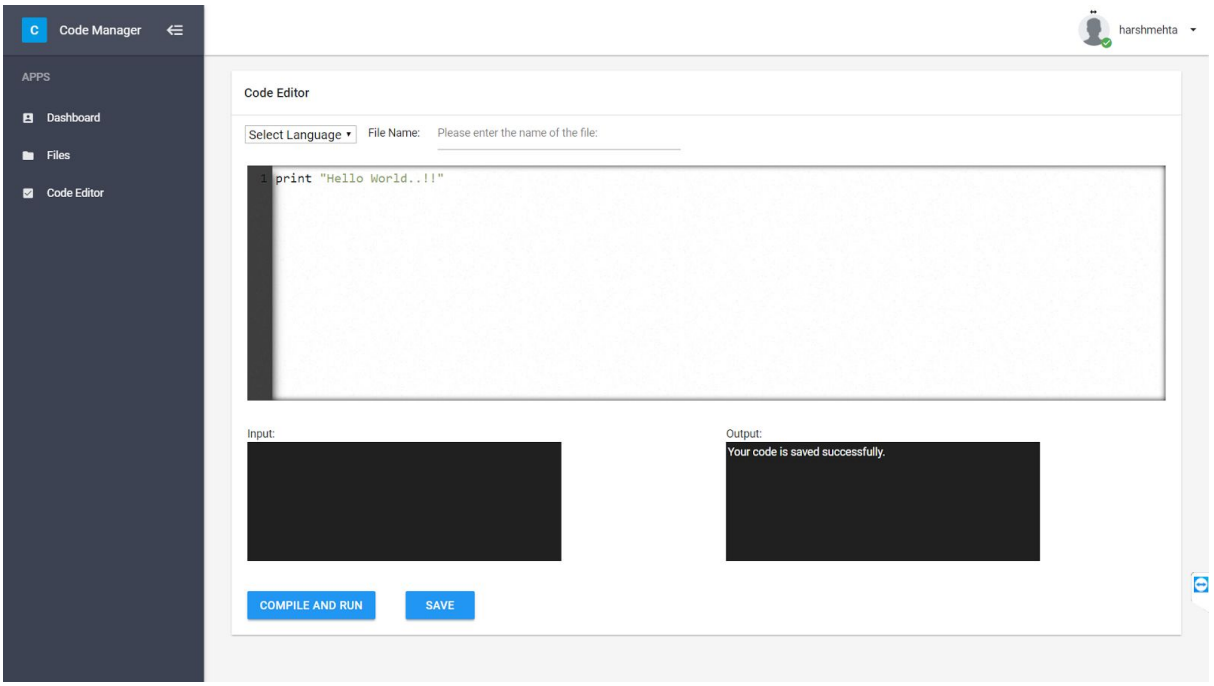
- Compilation error is showing in output



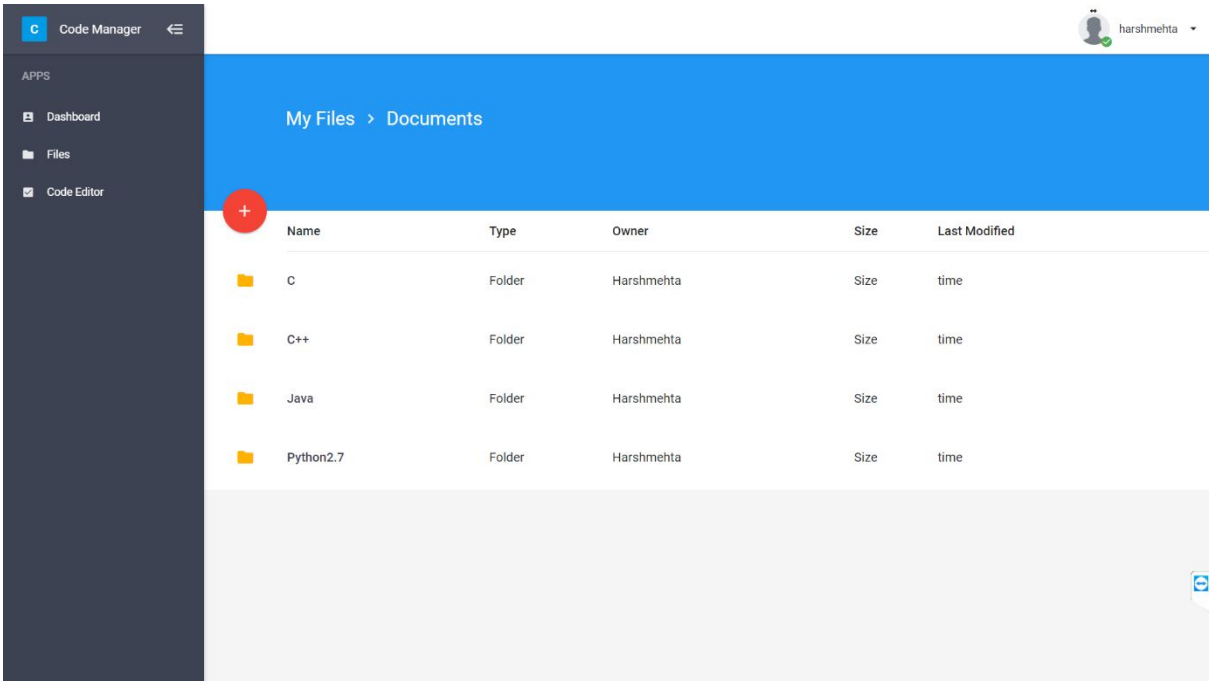
- Successful compilation

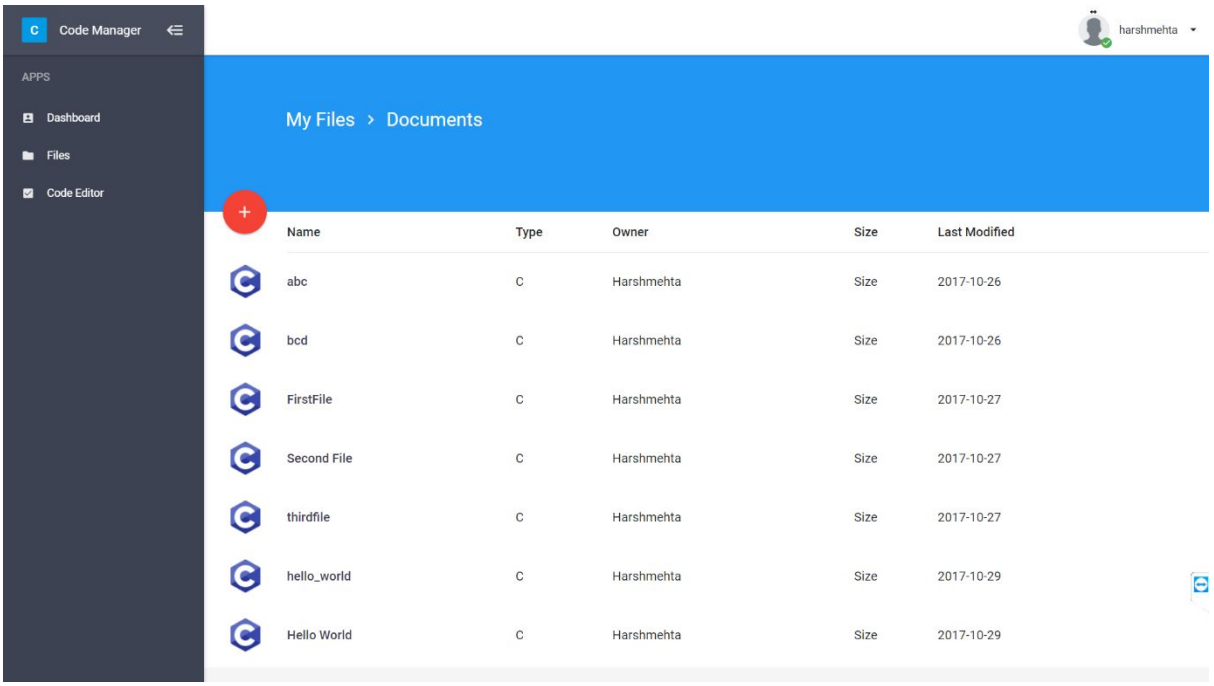


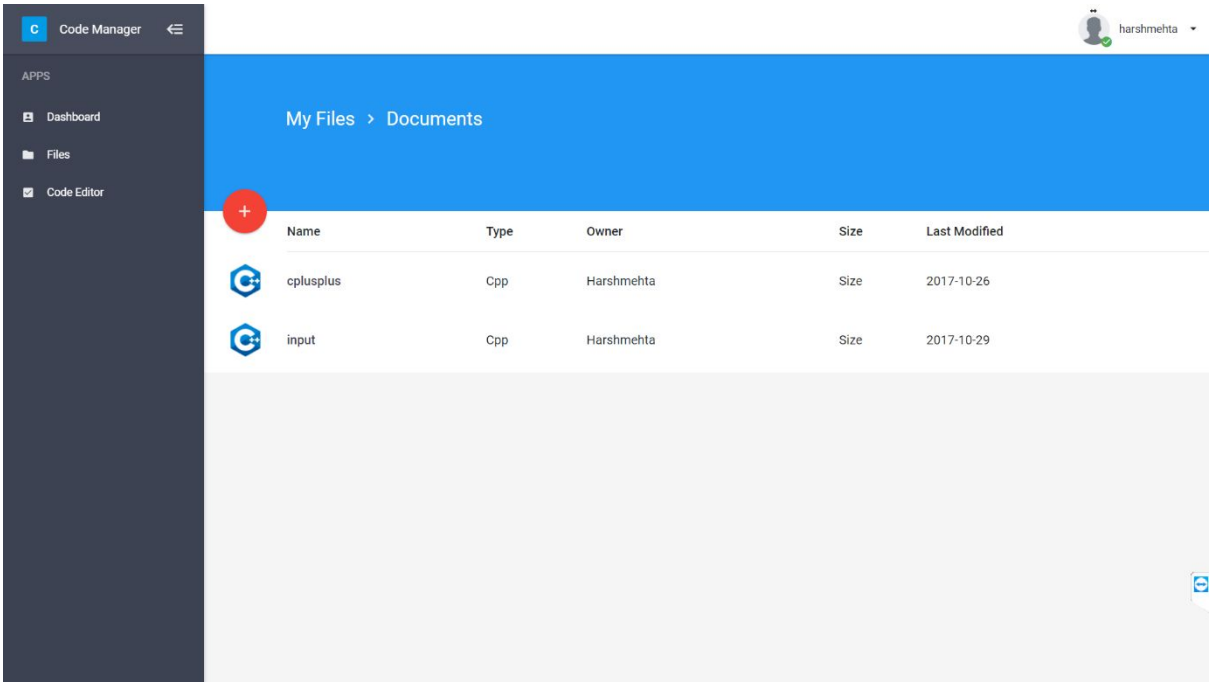
- Save file

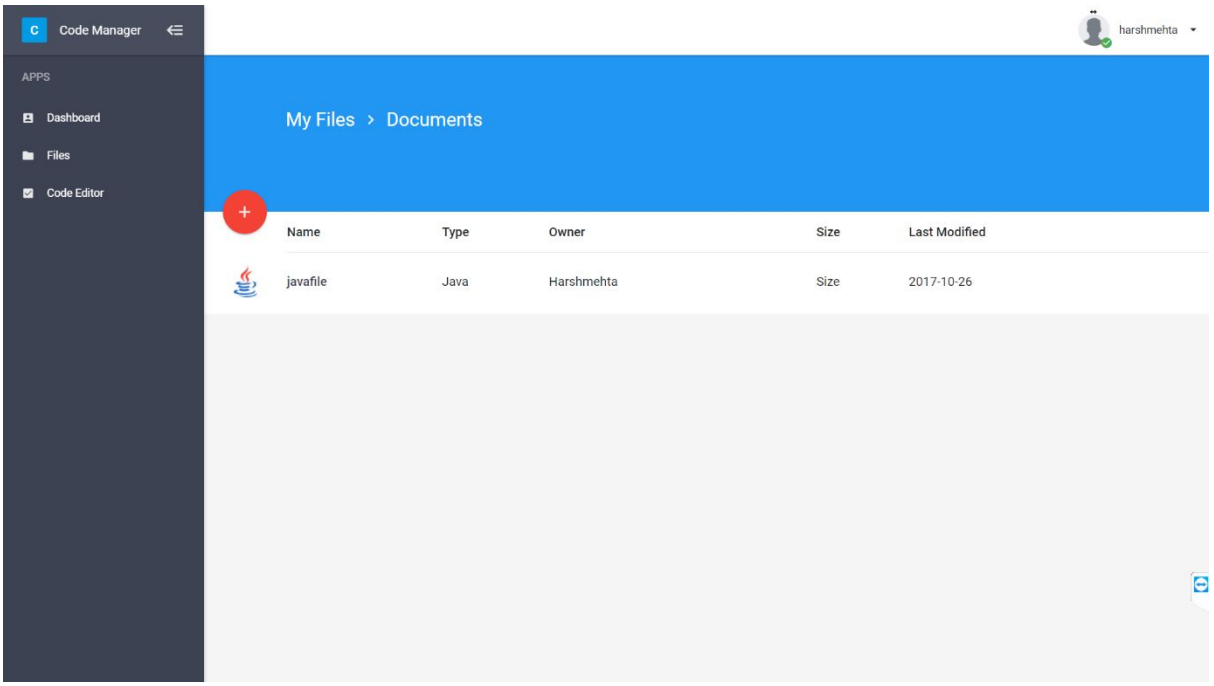


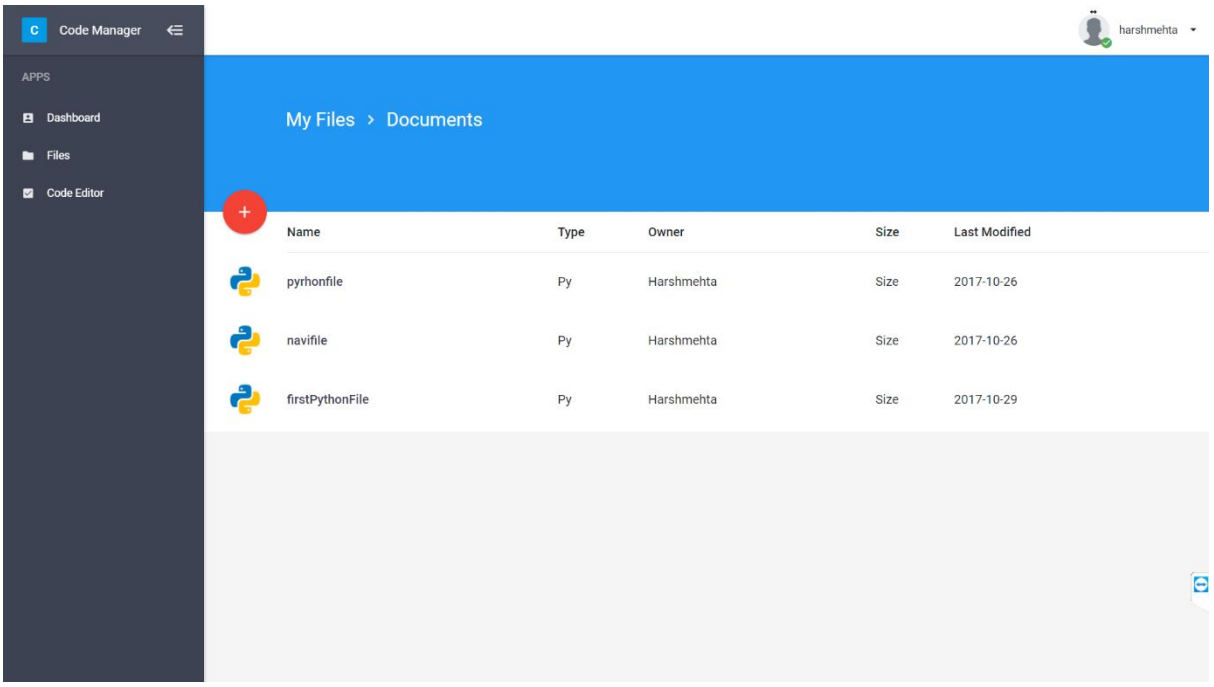
- File manager screen




Code Manager
APPS
Dashboard
Files
Code Editor
harshmehta
My Files > Documents
Name Type Owner Size Last Modified
abc C Harshmehta Size 2017-10-26
bcd C Harshmehta Size 2017-10-26
FirstFile C Harshmehta Size 2017-10-27
Second File C Harshmehta Size 2017-10-27
thirdfile C Harshmehta Size 2017-10-27
hello_world C Harshmehta Size 2017-10-29
Hello World C Harshmehta Size 2017-10-29


Code Manager
APPS
Dashboard
Files
Code Editor
harshmehta
My Files > Documents
Name Type Owner Size Last Modified
cplusplus Cpp Harshmehta Size 2017-10-26
input Cpp Harshmehta Size 2017-10-29


Code Manager
APPS
Dashboard
Files
Code Editor
harshmehta
My Files > Documents
Name
Type
Owner
Size
Last Modified
javafile
Java
Harshmehta
Size
2017-10-26


Code Manager
APPS
Dashboard
Files
Code Editor
harshmehta
My Files > Documents
Name
Type
Owner
Size
Last Modified
pyrhonfile
Py
Harshmehta
Size
2017-10-26
navifile
Py
Harshmehta
Size
2017-10-26
firstPythonFile
Py
Harshmehta
Size
2017-10-29

Future Work

- Code Analysis: Our IDE will show the total memory used by user's program and total computational time of it. For finding total memory, we are going to work with proc - a file system at kernel level mode that contain the process information. Among the various information we are interested in size of data, bss, stack and virtual memory size.
- Steps to be follow
    - As soon as user submit his code, a program at server side will automatically fetch the process ID (pid) of the newly generated process
    - For almost every process running in the system, there is a directory created in /proc/. The directory name is the process id and it contains several useful files which convey important information. The two main files of our concern are **/proc/[pid]/status** and **/proc/[pid]/statm**
    - From, these two files we get memory usage in form of files, and each page is of 1KB we multiply total pages and page size and calculate the memory usage.