

ASSIGNMENT TASK 1
CERTIFICATION REGISTRATION PORTAL



TEAM 1

Vankadari Sivanjanesh

Golamari Rahul Reddy

Ashutosh

Gowtham

Nitish Kondapaka

Requirement Specifications

Business Case 1: Users should be able to register their Certification details in the Web application and post registration they should be able to search their details.

Business Case 2: Web Application should use appropriate cost-effective Data Bases/Storages for implementing the above BC1.

Business Case 3: Valid Users should be able to login to the Web Application, Invalid Users should get an error message.

Business Case 4: Post registration, Users should get notification over mail.

Use Case 1: To login the Web app, User should be able to register their details.

Use Case 2: Authorized user with official mail id's (use Gmail for this implementation) should be able to login to the Web Application, Un Authorized users should get an error message.

(Make use of Token service in the above use case)

Use Case 3: User should be able to store their Certification details in appropriate Databases/

Storages post successful registration.

User Case 4: Users should be able to search their multiple certifications with search option in the

Web Application.

User Case 5: Appropriate notifications should be sent to registered Mail-ID, when an addition or deletion of certification is made in the portal.

Function Case 1: User should have Submit/Cancel button at the registration page.

Function Case 2: Successful registration confirmation should be shown to user after submitting the registration form.

Function Case 3: Registered Users when recording their Certification details, parameters to be considered are: Employee Name, CSP) AWS/Azure/GCP), Certification level, Certification Name, Certification ID (unique), Date of Certification, Expiry Date of Certification, Validity.

Technology Used For Task

Here are the few things to understand about the language and services we have used for the Task1

Front- End:

HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheet (CSS) and scripting language such as JavaScript.

CSS stands for Cascading Style Sheets and is used by web pages to help keep information in the proper display format. The major purpose of css is for describing the presentation of web pages, including colors, layout, and fonts that allows adapting the presentation to different types of devices, such as large screens, small screens, or printers.

JS is a scripting programming language that allows you to implement complex features on web pages. It enables you to create dynamically updating content, control multimedia, animate images , and pretty much everything else.

Back End:

Firebase

1) Firebase Authentication (Authenticating using password and google accounts).

Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more.

2) Cloud function (for sending mails).

Cloud Functions is a Google Cloud feature that allows functions to run in the cloud. In this article I'll show you the use of Cloud Functions in Firebase, creating an e-mail sending function that will work through web requests.

3) Cloud Firestore (for storing the certificate details).

Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud. Like Firebase Realtime Database, it keeps your data in sync across client apps through Realtime listeners and offers offline support for mobile and web so you can build responsive apps that work regardless of network latency or Internet connectivity. Cloud Firestore also offers seamless integration with other Firebase and Google Cloud products, including Cloud Functions.

4) Firebase storage (for storing pdf or jpg format of the certificate).

Cloud Storage for Firebase lets you upload and share user generated content, such as images and video, which allows you to build rich media content into your apps. Your data is stored in a Google Cloud Storage bucket — an exabyte scale object storage solution with high availability and global redundancy. Cloud Storage for Firebase lets you securely upload these files directly from mobile devices and web browsers, handling spotty networks with ease.

Hosting:

GCP was used to host web app, used services are Compute Engine, Cloud Storage, Cloud Firestore and Cloud Functions.

Let's understand what type of web app it is:

This web app is used for registering user cloud certification details where first user will create an account then login into the account after this they can put certification details about what cloud certification they have completed, after submission those details will get stored in Cloud Storage. So, in future if user want to get the details they can directly search their certificate details through their Certification Field.

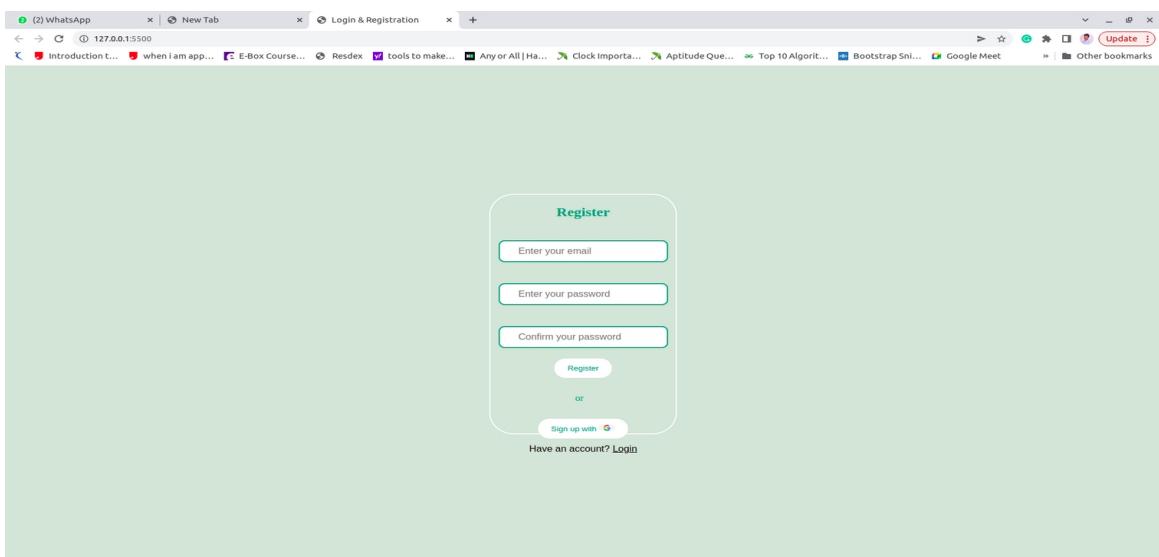
Requirement 1: Post registration, Users should get notification over mail.

1. **use case1** : To login the Web app, User should be able to register their details.
2. **User Case 5:** Appropriate notifications should be sent to registered Mail-ID, when an addition or deletion of certification is made in the portal.

User should register to the website and verify the mail by clicking on the verification link sent to the registered mail before they tried to login.

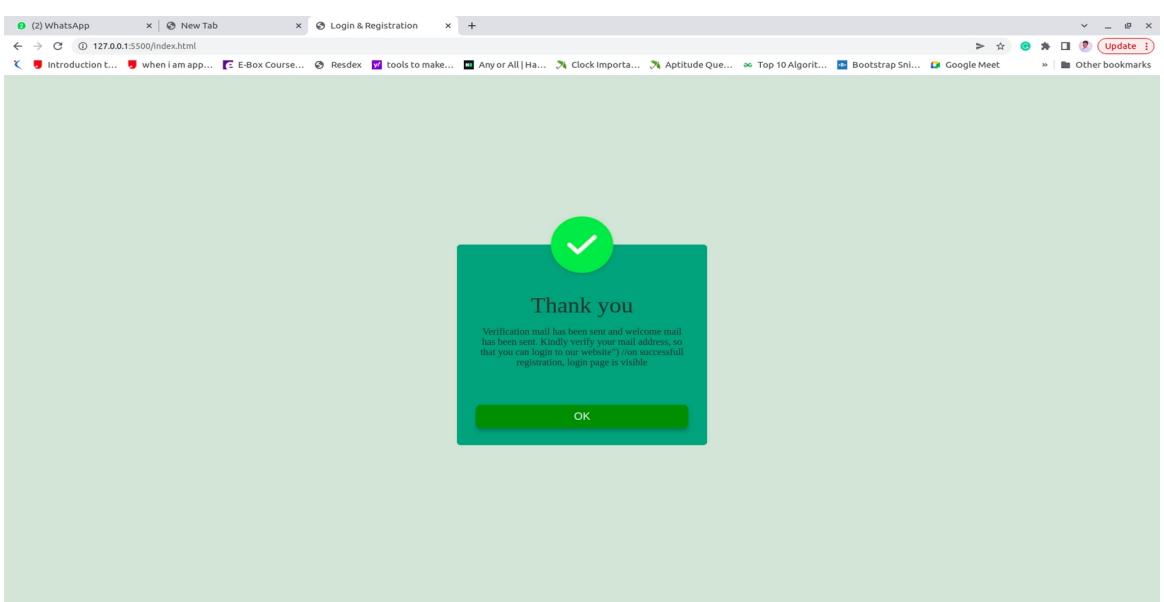
1.1 : Registering the User.

User should register to the website by entering the details

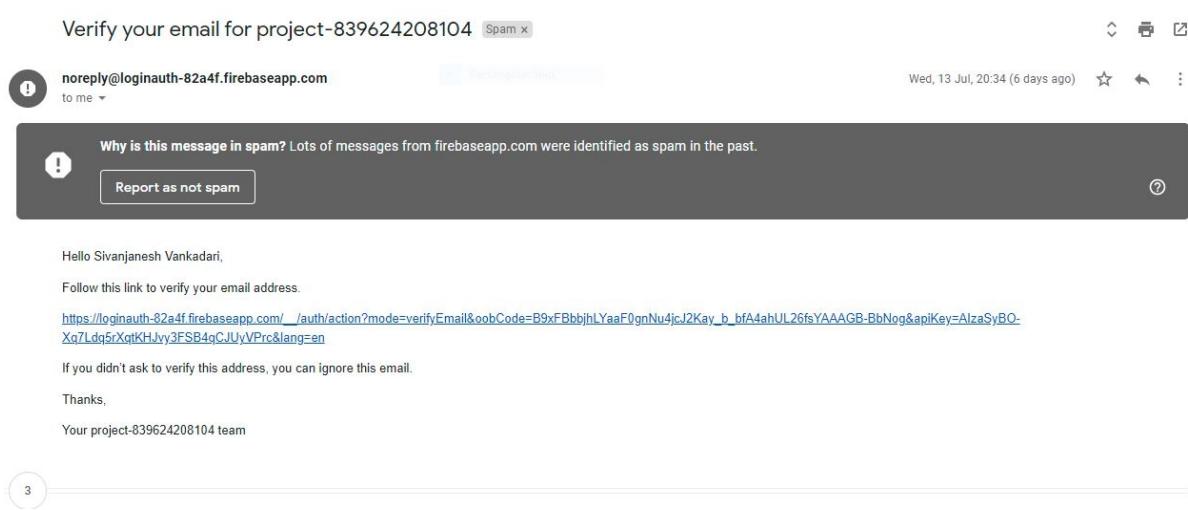


1.1.1 (Registration Page)

when user clicks on register button by entering valid mail a popup will be shown like verification mail has been sent to the registered mail.

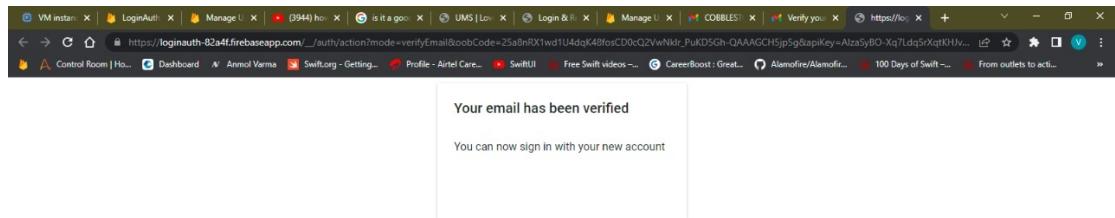


When user clicks on verification link user will get verified.



1.1.2 (Email verification image)

After user sees verified your email. A popup will show.

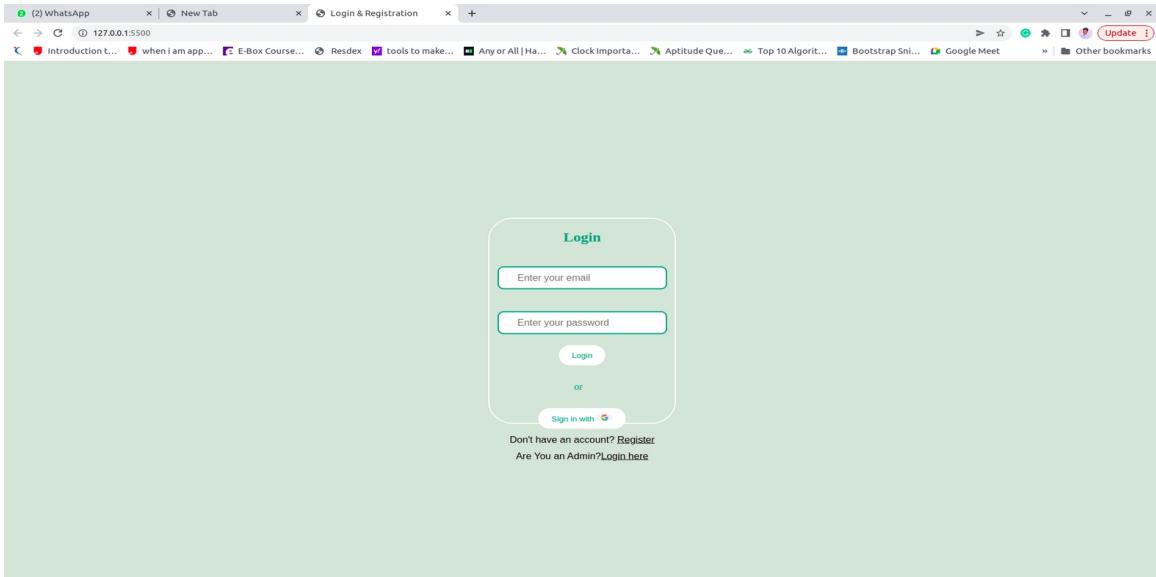


1.1.3 (user verified popup images)

Valid Users should be able to login to the Web Application, Invalid Users should get an error message.

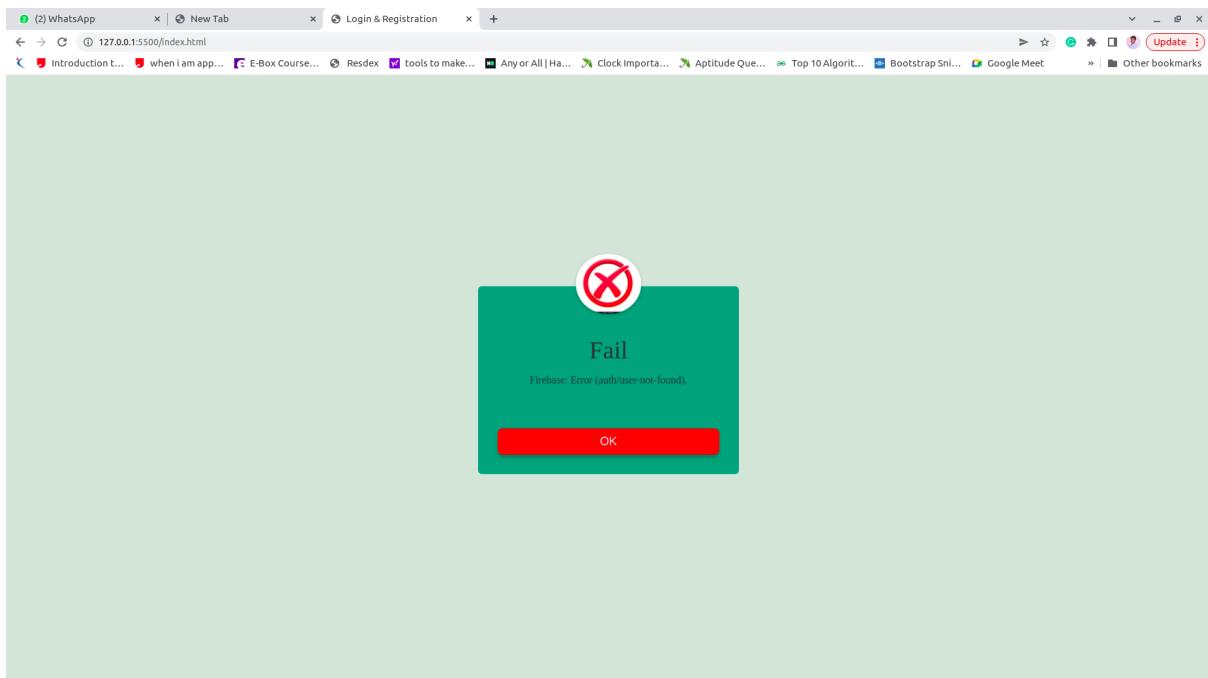
1.2 : Loggin the User.

After user sees verified your email. users can login

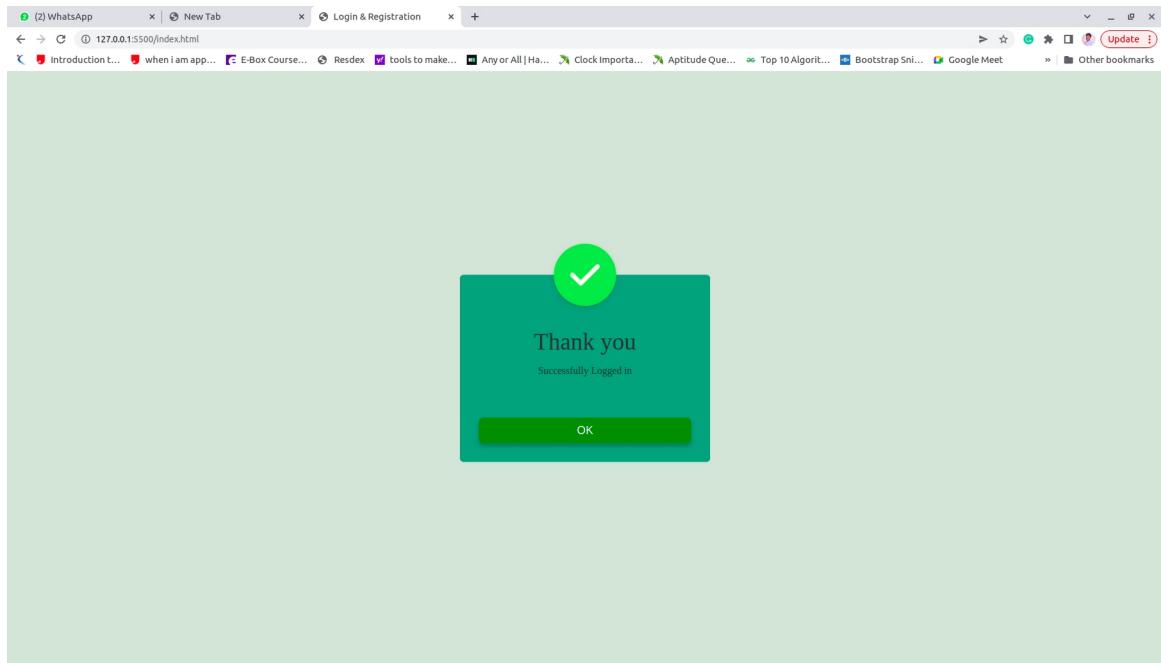


1.2.1 (Login Image)

if user gives incorrect username or password. An error popup will be thrown

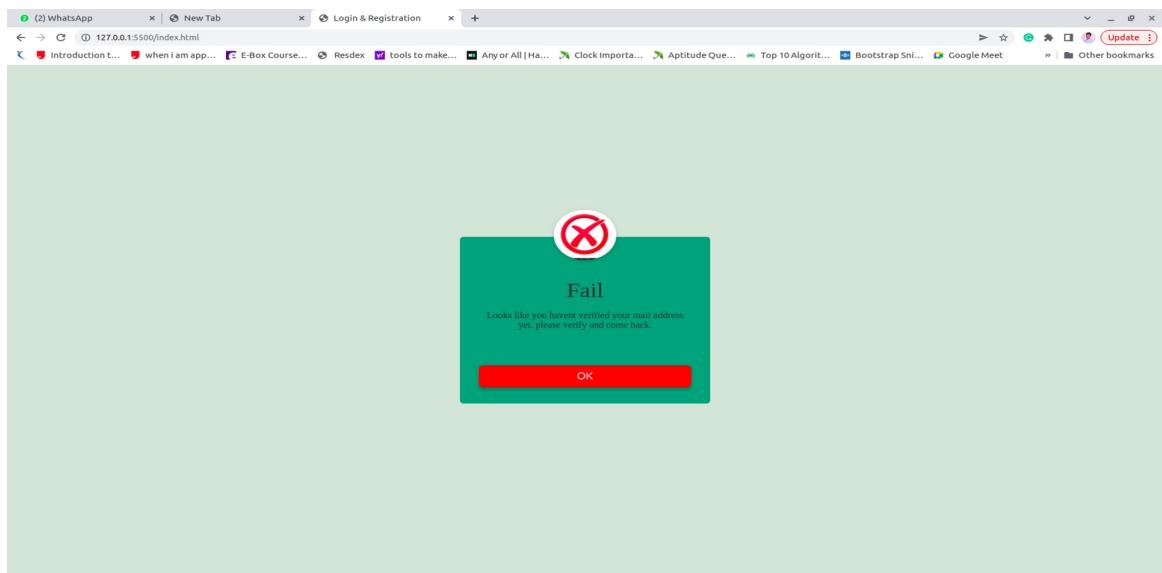


After user enter details for login and clicks Login Button. A popup will shows.



1.2.2 (Successfully login popup)

If a user tried to login with unverified mail , an error will be thrown.



1.2.3 (Error login popup)

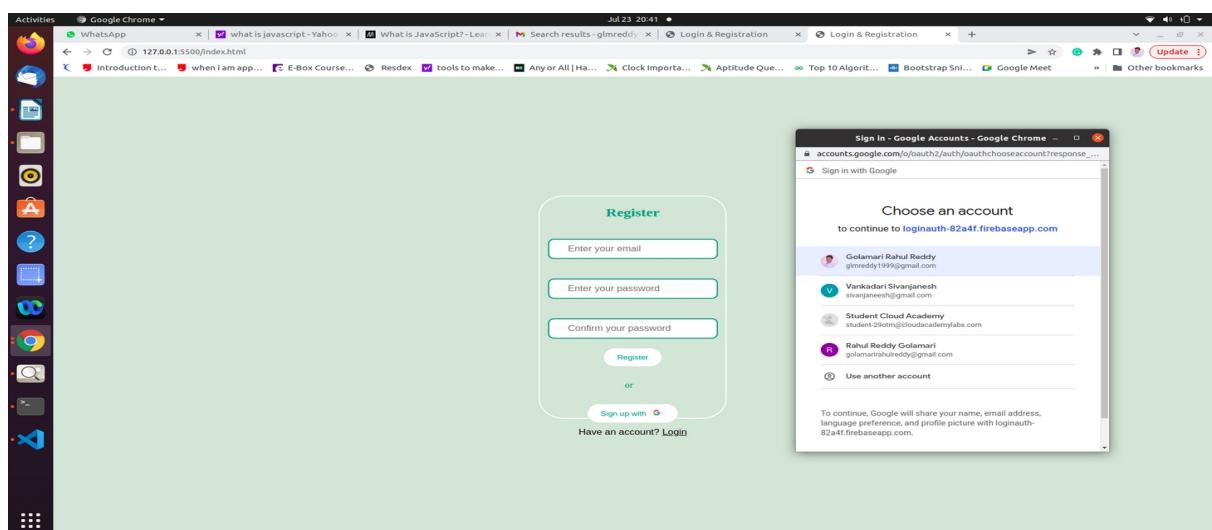
Use Case 2: Authorized user with official mail id's (use Gmail for this implementation) should be able to login to the Web Application, Unauthorized users should get an error message.

(Make use of Token service in the above use case)

User should register to the website through mail and verified mail popup will be visible before they tried to login.

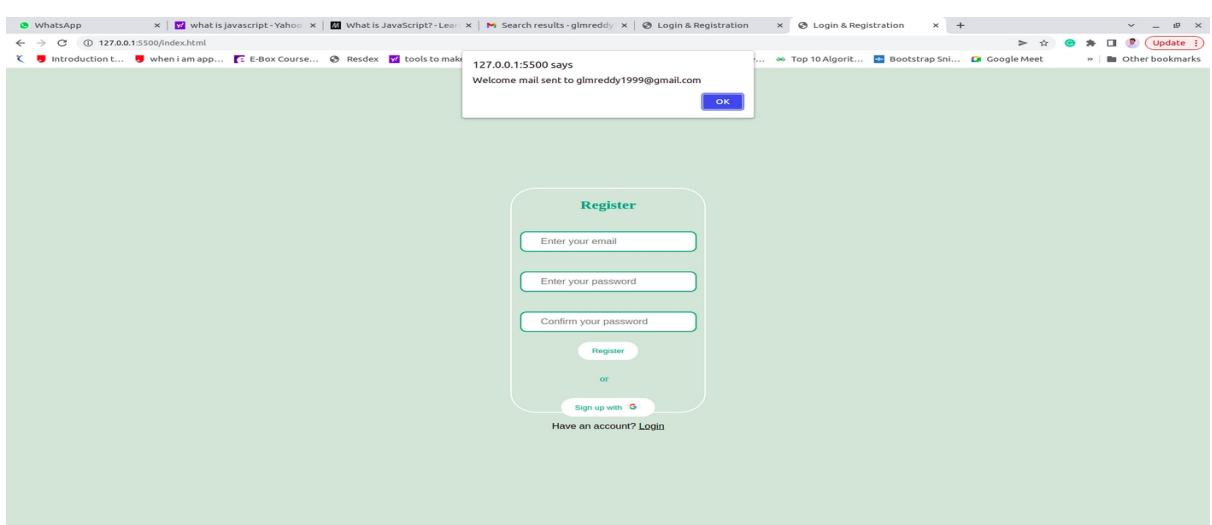
2.1 : Registering the User with Official mail

User should register to the website by selecting the mail.



2.1.1 (Registration Page Image)

when user clicks on register button by selecting valid mail a verification is done automatically and sends welcome mail.

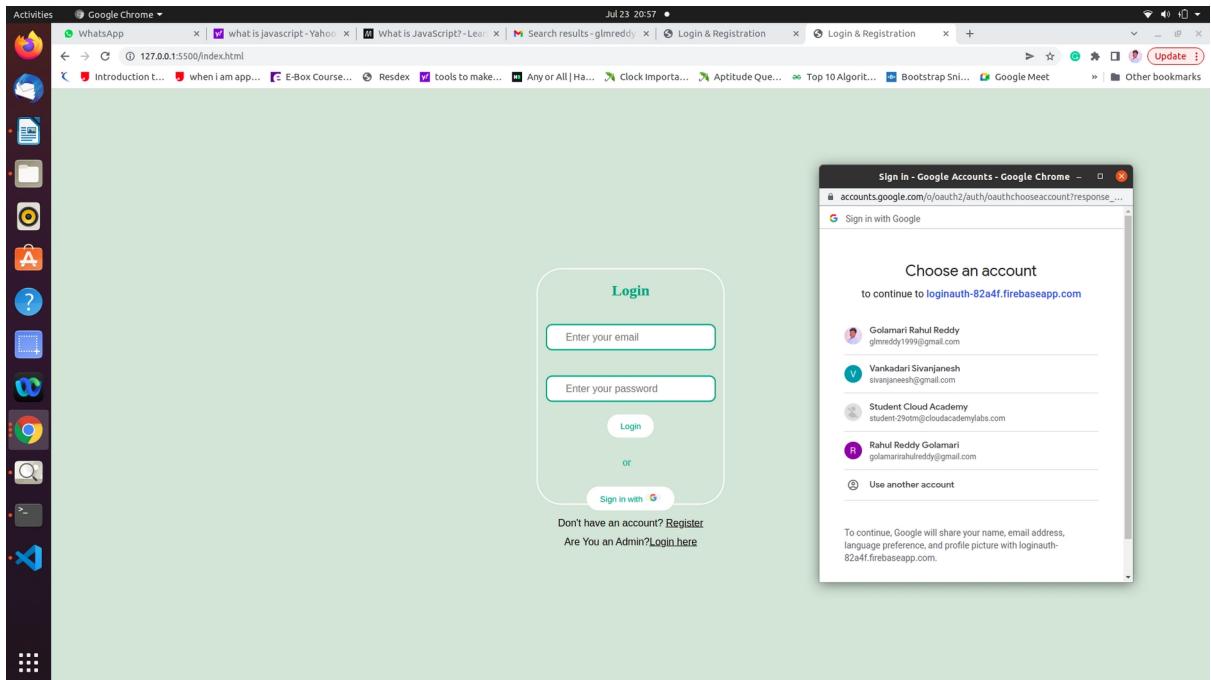


2.1.2 (Successfully Registered)

Valid Users should be able to login to the Web Application, Invalid Users should get an error messages.

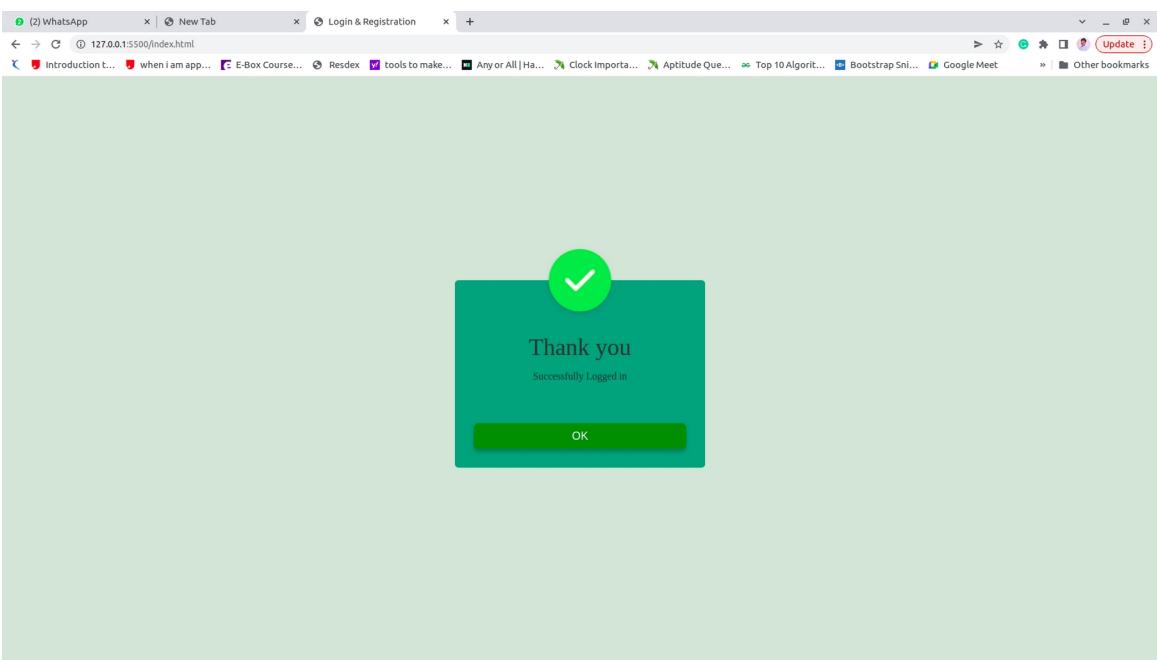
2.2 : Logging the User

After user gets welcome mail. user can login



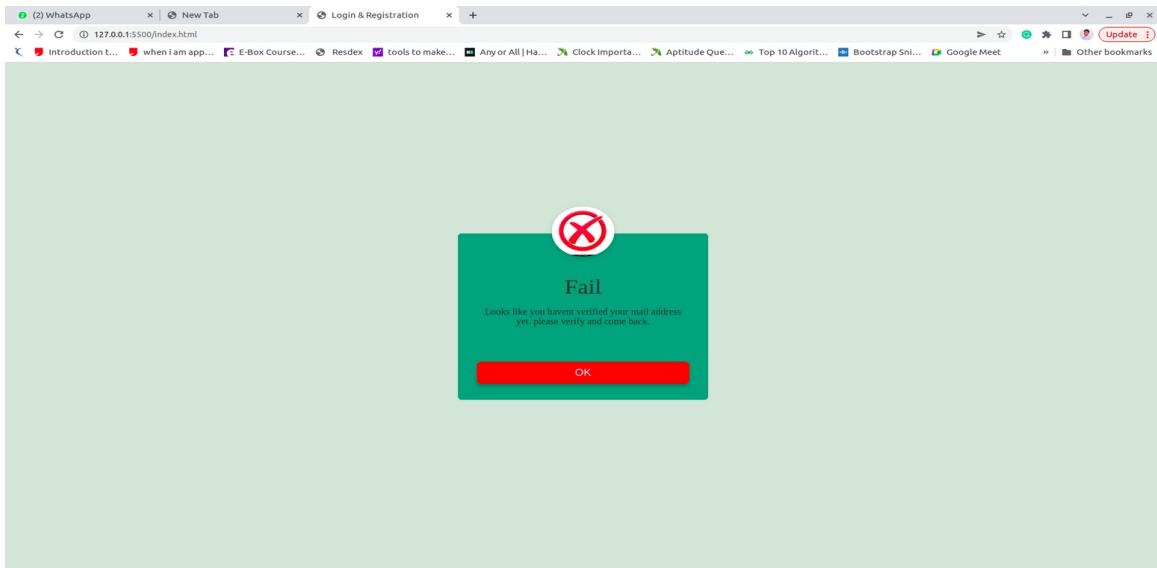
2.2.1 (Login Image)

After Login is Successfully. A popup will shows up.



2.2.2 (Successfully login popup)

If a user tried to login with unregistered or unverified mail , an error will be thrown.



2.2.3 (Error login popup)

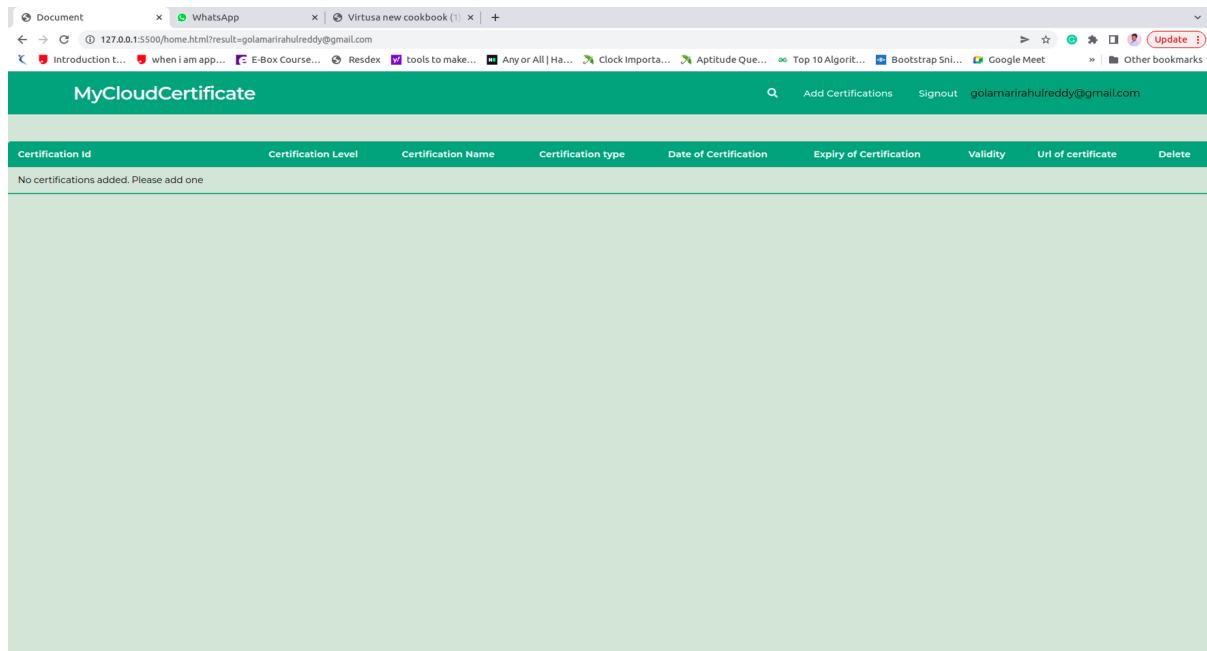
Use Case 3: User should be able to store their Certification details in appropriate Databases/ Storages post successful registration.

User Case 5: Appropriate notifications should be sent to registered Mail-ID, when an addition or deletion of certification is made in the portal.

To add and store user certification details. We have created an HTML CERTIFICATION REGISTRATION FORM and cloud FIRESTORE DATABASE where user can enter their details and store the details in Google Cloud Firestore Database.

To get notifications through email we used CLOUD FUNCTIONS and for deletion we used firebase storage delete service by first create a reference to that file. Then call the delete() method on that reference,

When user Login to the Web App. Home Page will be visible where user can click on Add Certification Button to enter details of Certification.



(Home Page)

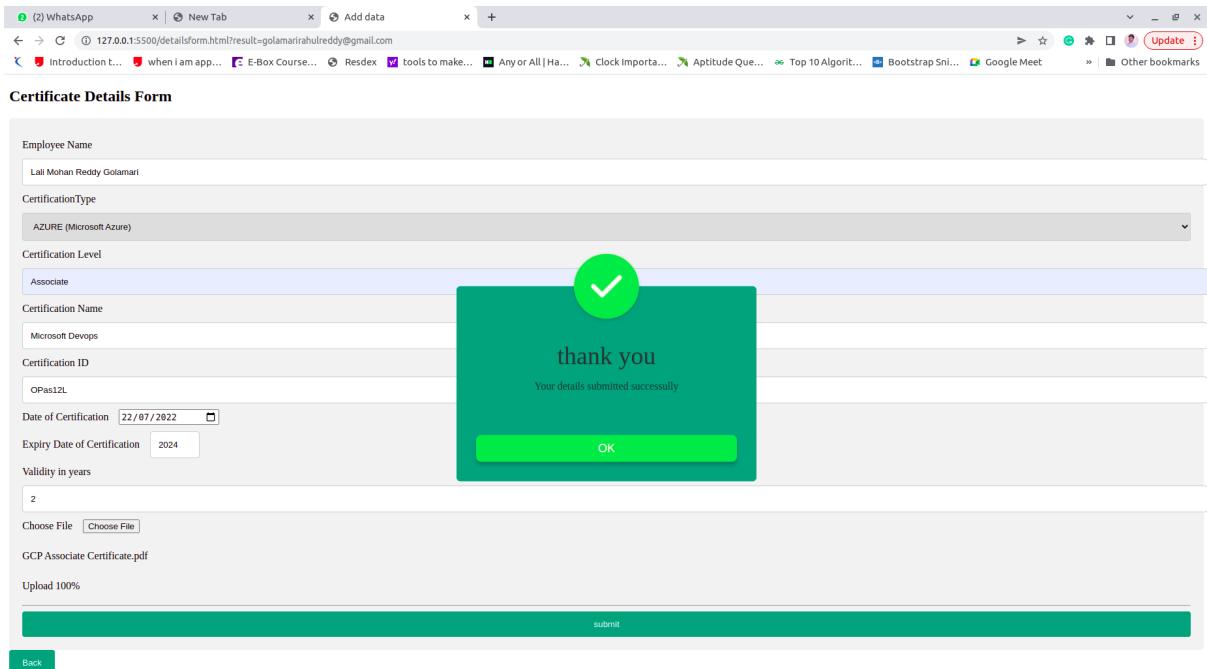
3.1 : Certification Details Registration Form

when the user clicks the add certification button then a certification form appears there user needs to enter details such as certification types, certification level, validity etc.. and clicks submit button.

Employee Name	<input type="text"/>
CertificationType	AWS (Amazon Web Services)
Certification Level	<input type="text"/>
Certification Name	<input type="text"/>
Certification ID	<input type="text"/>
Date of Certification	<input type="text"/> dd/mm/yyyy
Expiry Date of Certification	2016
Validity in years	<input type="text"/> 0
Choose File	<input type="button" value="Choose File"/>
Name of file	<input type="text"/>
<input type="button" value="submit"/>	

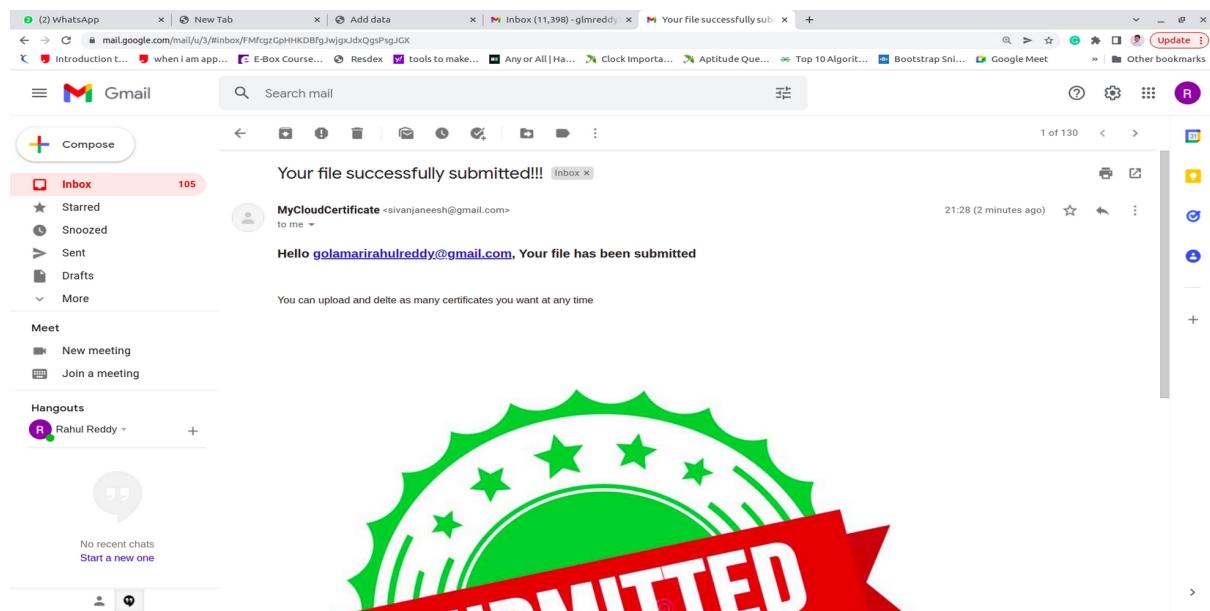
3.1.1 (Certification Form)

After user enters their details and clicks the submit a popup will be shown like successfully submitted. And data will be stored on CLOUD FIRESTORE DATABASE.



3.1.2 (Details Submitted Successfully)

When user clicks on the submit button a popup will be shown like in the **3.1.2 (Details Submitted Successfully)** image. And mail will be sent to the login mail as your file has been submitted.



Certification details are stored successfully in the filestore

The screenshot shows the Firebase Cloud Firestore interface. On the left, the navigation sidebar includes 'Project Overview', 'Authentication', 'Firestore Database' (selected), and 'Storage'. The main area displays a document structure under 'golamarrahulreddy@gmail.com'. A specific document named 'OPas12L' is expanded, showing its fields:

certificationname: "Lali Mohan Reddy Golamari"
certificationid: "OPas12L"
certificationlevel: "Associate"
certificationname: "Microsoft Devops"
certificationtype: "AZURE"
dateofcertification: "2022-07-22"
expiryofcertification: "2024-07-22"
validity: "2"

3.1.3 (Details Added in Firestore Database)

certification pdf or image file is stored in storage centre firebase

The screenshot shows the Firebase Storage interface. On the left, the navigation sidebar includes 'Project Overview', 'Authentication', 'Firestore Database', and 'Storage' (selected). The main area lists files in a folder named 'golamarrahulreddy@gmail.com'. One file, 'OPas12L', is selected and shown in a detailed view:

Name	Size ↑	Type	Last modified
OPas12L	168.2 KB	application/pdf	Jul 22, 2022

Detailed view of 'OPas12L':

Name	OPas12L
Size	172,238 bytes
Type	application/pdf
Created	Jul 22, 2022, 9:28:04 PM
Updated	Jul 22, 2022, 9:28:04 PM
File location	(dropdown)
Other metadata	(dropdown)

3.1.4 (Storing Certificate Image)

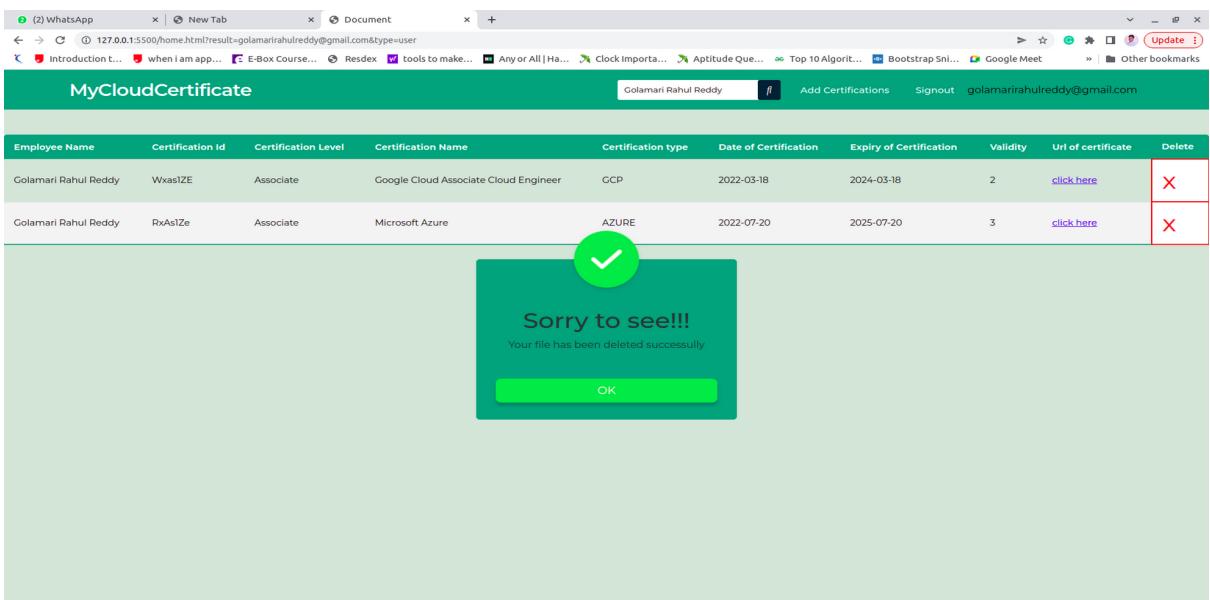
After Adding the details in database. Home page will look like below image

Employee Name	Certification Id	Certification Level	Certification Name	Certification type	Date of Certification	Expiry of Certification	Validity	Url of certificate	Delete
Golamari Rahul Reddy	RxAs1Ze	Associate	Microsoft Azure	AZURE	2022-07-20	2025-07-20	3	click here	X
Lali Mohan Reddy Golamari	OPas12L	Associate	Microsoft Devops	AZURE	2022-07-22	2024-07-22	2	click here	X
Sivanjanesh	YxQs1ZA	Associate	Google Cloud Associate Cloud Engineer	AWS	2022-07-20	2024-07-20	2	click here	X

3.1.5 (Home Page After Adding Data)

User can also delete their certificate details by clicking in the red wrong option of delete field in the Home Page.

When user deleted the file a popup will be shown like the below image.



3.1.6 (Deleting the Details)

And also, mail will be sent as Your file has been Deleted.

The screenshot shows a Gmail inbox with 105 messages. Two messages from 'MyCloudCertificate' are visible, both containing the text: 'Hello golamarrahulreddy@gmail.com, Your file has been deleted You can upload and delte as many certificates you want at any time'. The first message is from 'MyCloudCertificate' and the second is from 'MyCloudCertificate <svivanjaneesh@gmail.com> to me'. The inbox sidebar shows options like Starred, Snoozed, Sent, Drafts, and More. Below the inbox, there's a 'Meet' section and a 'Hangouts' section where 'Rahul Reddy' is listed. At the bottom, there's a 'No recent chats' section with a 'Start a new one' button.

3.1.7 (File Deletion Mail)

use case 4: user should be able to search his certificates using the search bar.

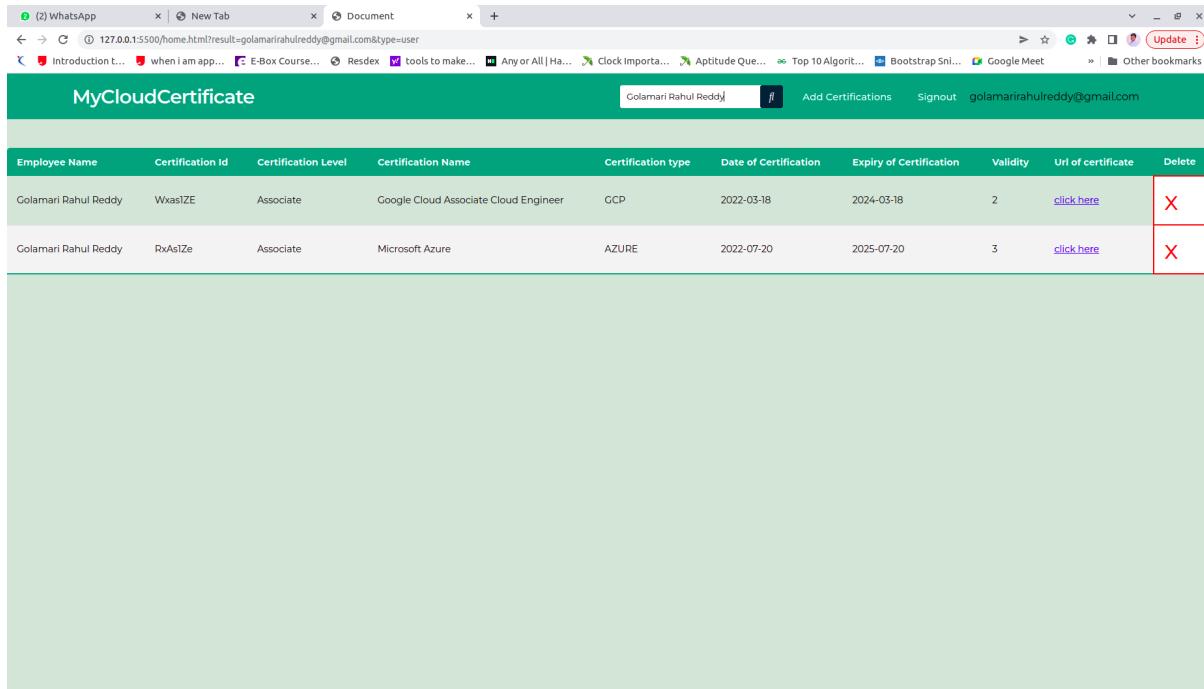
After user added multiple certification.

The screenshot shows the 'MyCloudCertificate' application interface. At the top, there's a navigation bar with tabs for 'Add Certifications' and 'Signout'. The main area displays a table of certifications with the following data:

Employee Name	Certification Id	Certification Level	Certification Name	Certification type	Date of Certification	Expiry of Certification	Validity	Url of certificate	Delete
Golamari Rahul Reddy	RxAstZE	Associate	Microsoft Azure	AZURE	2022-07-20	2025-07-20	3	click here	X
Golamari Rahul Reddy	WxasIZE	Associate	Google Cloud Associate Cloud Engineer	CCP	2022-03-18	2024-03-18	2	click here	X
Sivanjanesh	YxQsIZA	Associate	Google Cloud Associate Cloud Engineer	AWS	2022-07-20	2024-07-20	2	click here	X

4.1.1 (Home Page With Multiple Certificates)

User can search for particular files based on certification fields like certification type or certification Employee etc... , by clicking on search option in the Home Page.



The screenshot shows a web browser window with the title 'MyCloudCertificate'. The search bar contains the text 'Golamari Rahul Reddy'. Below the search bar is a navigation menu with links for 'Add Certifications', 'Signout', and an email address 'golamarirahulreddy@gmail.com'. The main content area displays a table of certification records:

Employee Name	Certification Id	Certification Level	Certification Name	Certification type	Date of Certification	Expiry of Certification	Validity	Url of certificate	Delete
Golamari Rahul Reddy	Wxas1ZE	Associate	Google Cloud Associate Cloud Engineer	GCP	2022-03-18	2024-03-18	2	click here	X
Golamari Rahul Reddy	RxAstZe	Associate	Microsoft Azure	AZURE	2022-07-20	2025-07-20	3	click here	X

4.2.2 (Searching On Certification Field)

use case 5: Has Already been implemented in **use case 1** and **use case 3 for mail notifications.**

Finally user can signout from the web app by clicking on the signout option.

The screenshot shows a web browser window with the following details:

- Title Bar:** Document, WhatsApp, Virtusa new cookbook, +
- Address Bar:** 127.0.0.1:5500/home.html?result=golamarirahulreddy@gmail.com
- Toolbar:** Back, Forward, Stop, Refresh, Home, Update, Other bookmarks
- Page Content:**
 - Header:** MyCloudCertificate, Add Certifications, signout, golamarirahulreddy@gmail.com
 - Table:** A table with columns: Certification Id, Certification Level, Certification Name, Certification type, Date of Certification, Expiry of Certification, Validity, Url of certificate, Delete. The table has a single row with the text "No certifications added. Please add one."

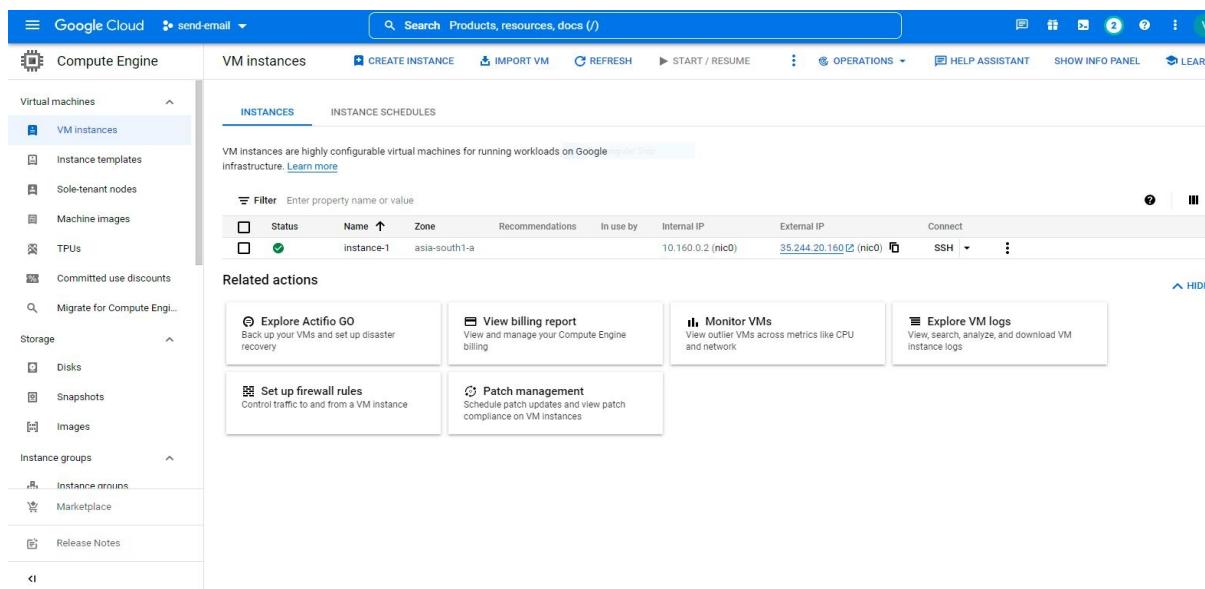
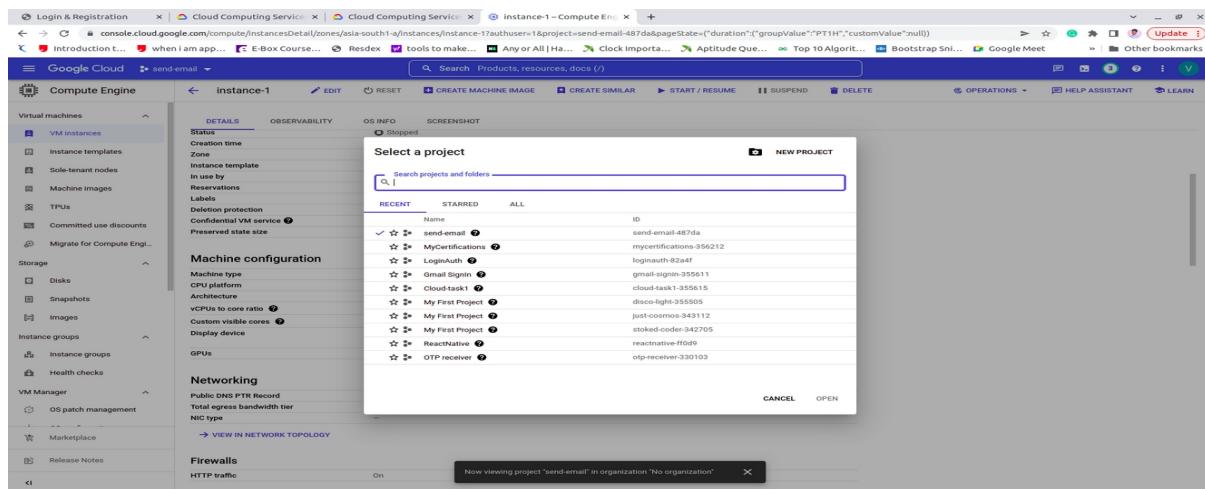
(Sign Out Option)

Hosting the Web App:

GCP Services used:

After completion of web app, user have to host it using Virtual Machine and store the certification details in Cloud Firestore Database in Google Cloud Platform.

Step 1: Login into the GCP Console with your google account and make sure have a project, if not create one project.



Step 2: First, select the compute engine option and click the create VM instance, give the name and select region, zone and choose the machine type for the web app and appropriate configuration details.

Check the below image for details of vm instance.

The screenshot shows the Google Cloud Compute Engine interface. On the left, a sidebar navigation includes 'Compute Engine' (selected), 'Virtual machines' (selected), 'VM instances' (selected), and other options like 'Instance templates', 'Sole-tenant nodes', 'Machine images', 'TPUs', 'Committed use discounts', 'Migrate for Compute Eng...', 'Storage' (with 'Disks', 'Snapshots', 'Images'), 'Instance groups' (with 'Instance groups' and 'Marketplace'), and 'Release Notes'. The main content area is titled 'instance-1' and shows the 'DETAILS' tab selected. It displays basic information such as Name (instance-1), Instance Id (7222147473148134151), Description (None), Type (Instance), Status (Running), Creation time (Jul 20, 2022, 9:30:42 AM UTC+05:30), Zone (asia-south1-a), and various configuration settings like Reservations (Automatically choose), Labels (None), and Preserved state size (0 GB). There are tabs for 'OBSERVABILITY', 'OS INFO', and 'SCREENSHOT'.

This screenshot shows the configuration page for the same 'instance-1' VM. The left sidebar is identical to the previous screenshot. The main content area is titled 'instance-1 - Compute Engine' and shows the 'DETAILS' tab selected. It provides detailed configuration options under sections like 'Machine configuration' (Machine type: e2-medium, CPU platform: Unknown CPU Platform, Architecture: x86/64, vCPUs to core ratio: 1, Custom visible cores: 1, Display device: Enabled, GPUs: None), 'Networking' (Public DNS PTR Record: None, Total egress bandwidth tier: -, NIC type: -), 'Firewalls' (HTTP traffic: On, HTTPS traffic: On), and 'Network interfaces' (nic0: default network, default subnetwork, Primary internal IP address: 10.160.0.2, Alias IP ranges: Ephemeral, External IP address: Premium, Network tier: Premium, IP: Of). There are tabs for 'OBSERVABILITY', 'OS INFO', and 'SCREENSHOT'.

After creating, The Vm Instance will have Internal IP Address and External IP Address.

The screenshot shows the Google Cloud Compute Engine VM Instances page. On the left, there's a sidebar with sections like Virtual machines, Storage, and VM Manager. The main area displays a table of VM instances. One instance, 'instance-1', is listed with the following details:

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
Up	instance-1	asia-south1-a			10.160.0.2 (nic0)	35.244.20.160 (2) (nic0)	SSH

Below the table, there are several related actions buttons:

- Explore Active GO
- View billing report
- Monitor VMs
- Explore VM logs
- Patch management
- Set up firewall rules

A small notification at the bottom left says 'Instance started'.

Connect to the SSH for hosting the website using apache2 server.

So, upon creating the vm instance, click on the SSH option of your Vm intance.

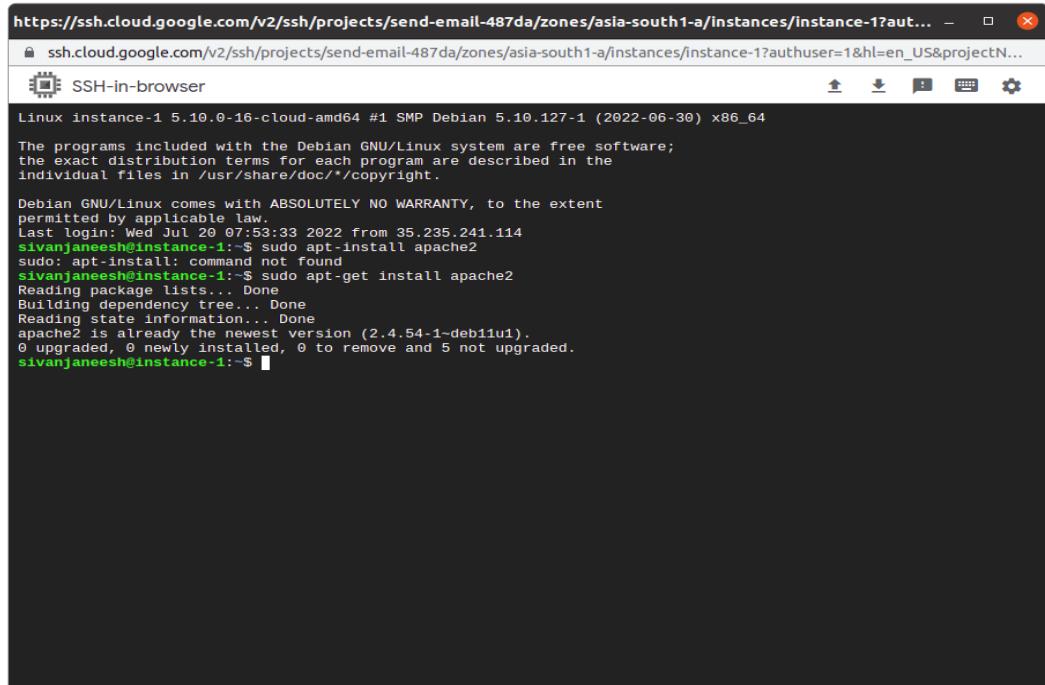
After that ssh terminal opens in new browser

The screenshot shows an SSH terminal window titled 'SSH-in-browser'. The URL in the address bar is https://ssh.cloud.google.com/v2/ssh/projects/send-email-487da/zones/asia-south1-a/instances/instance-1?authuser=1&hl=en_US&projectN.... The terminal displays a Debian 5.10.127-1 shell prompt:

```
Linux instance-1 5.10.0-16-cloud-amd64 #1 SMP Debian 5.10.127-1 (2022-06-30) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jul 20 07:53:33 2022 from 35.235.241.114
sivanjaneesh@instance-1:~$
```

- check whether apache2 is installed .

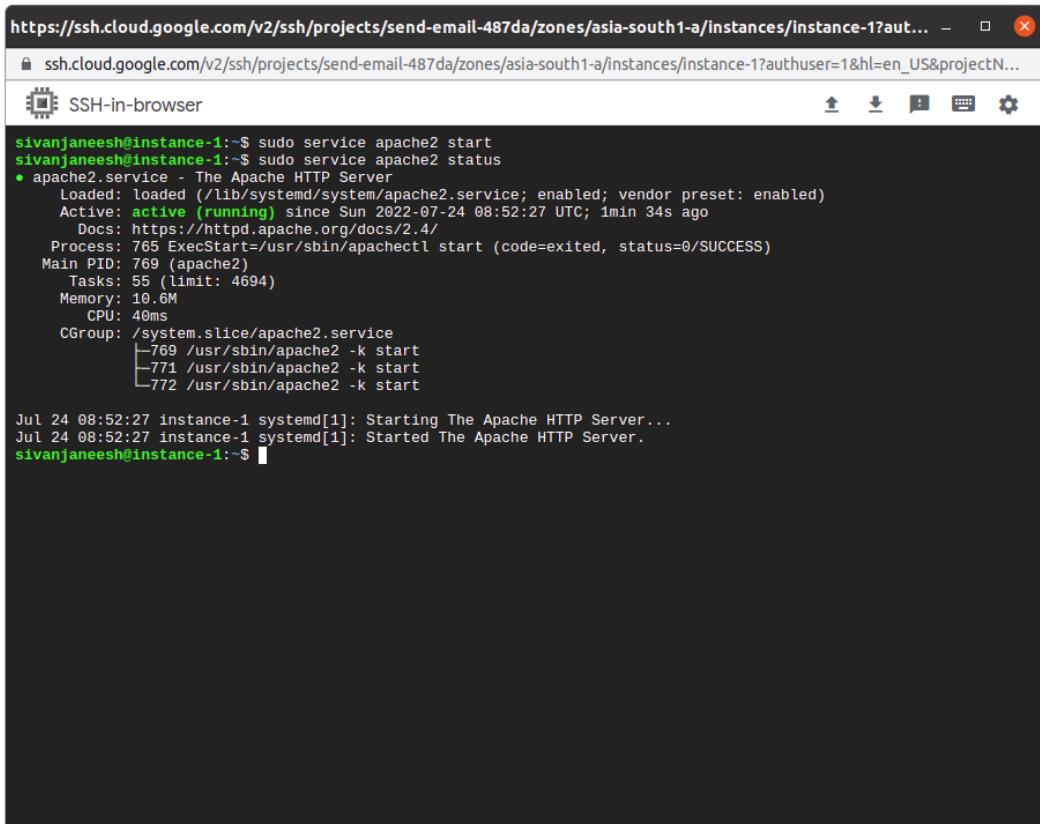


```
https://ssh.cloud.google.com/v2/ssh/projects/send-email-487da/zones/asia-south1-a/instances/instance-1?aut...
ssh.cloud.google.com/v2/ssh/projects/send-email-487da/zones/asia-south1-a/instances/instance-1?authuser=1&hl=en_US&projectN...
SSH-in-browser
Linux instance-1 5.10.0-16-cloud-amd64 #1 SMP Debian 5.10.127-1 (2022-06-30) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jul 20 07:53:33 2022 from 35.235.241.114
sivanjaneesh@instance-1:~$ sudo apt-install apache2
sudo: apt-install: command not found
sivanjaneesh@instance-1:~$ sudo apt-get install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.54-1-deb11u1).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
sivanjaneesh@instance-1:~$
```

In my case i have already installed.

- If not , install it using sudo apt-get install apache2
- Then, start the service using sudo service apache2 start.

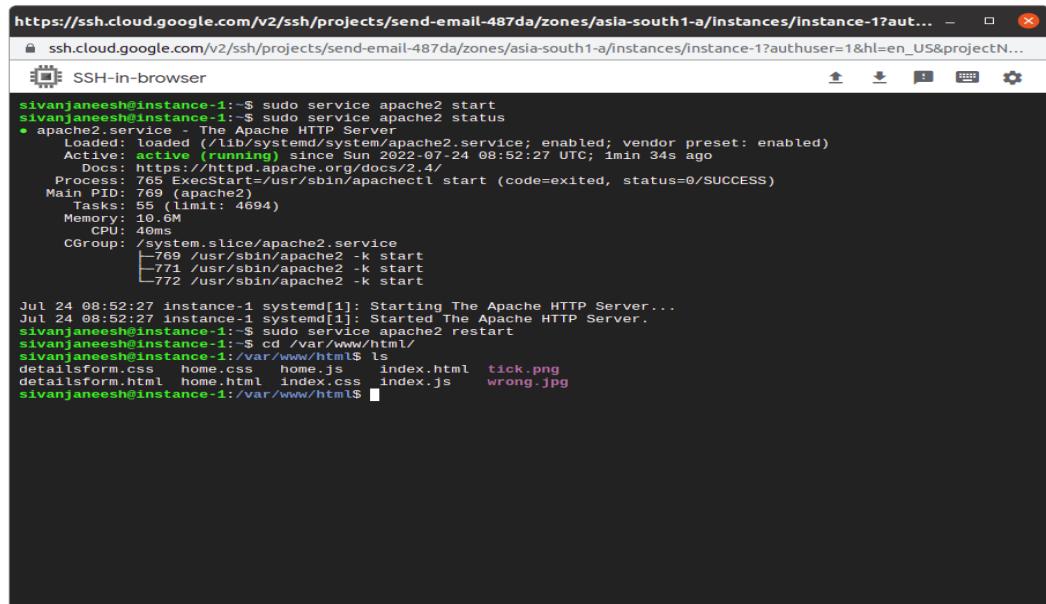


```
https://ssh.cloud.google.com/v2/ssh/projects/send-email-487da/zones/asia-south1-a/instances/instance-1?aut...
ssh.cloud.google.com/v2/ssh/projects/send-email-487da/zones/asia-south1-a/instances/instance-1?authuser=1&hl=en_US&projectN...
SSH-in-browser
sivanjaneesh@instance-1:~$ sudo service apache2 start
sivanjaneesh@instance-1:~$ sudo service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-07-24 08:52:27 UTC; 1min 34s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 765 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 769 (apache2)
    Tasks: 55 (limit: 4694)
   Memory: 10.6M
      CPU: 40ms
     CGroup: /system.slice/apache2.service
             └─769 /usr/sbin/apache2 -k start
                 ├─771 /usr/sbin/apache2 -k start
                 ├─772 /usr/sbin/apache2 -k start
                 └─772 /usr/sbin/apache2 -k start

Jul 24 08:52:27 instance-1 systemd[1]: Starting The Apache HTTP Server...
Jul 24 08:52:27 instance-1 systemd[1]: Started The Apache HTTP Server.
sivanjaneesh@instance-1:~$
```

- *cd /var/www/html (html folder is already created with index.html when we installed apache server for default web app) (cd is used to change directory)*

- *sudo rm index.html (remove the default web app file and add your web app file into the html folder)*
- *In my case i have already added file to the html folder.*



The screenshot shows an SSH session in a browser window titled "SSH-in-browser". The URL is https://ssh.cloud.google.com/v2/ssh/projects/send-email-487da/zones/asia-south1-a/instances/instance-1?authuser=1&hl=en_US&projectN.... The terminal output is as follows:

```
sivanjaneesh@instance-1:~$ sudo service apache2 start
sivanjaneesh@instance-1:~$ sudo service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-07-24 08:52:27 UTC; 34s ago
     Docs: http://httpd.apache.org/docs/2.4/
   Process: ExecStart=/usr/sbin/apache2 start (code=exited, status=0/SUCCESS)
 Main PID: 769 (apache2)
   Tasks: 55 (limit: 4694)
    Memory: 10.6M
      CPU: 40ms
     CGroup: /system.slice/apache2.service
             ├─769 /usr/sbin/apache2 -k start
             ├─771 /usr/sbin/apache2 -k start
             ├─772 /usr/sbin/apache2 -k start

Jul 24 08:52:27 instance-1 systemd[1]: Starting The Apache HTTP Server...
Jul 24 08:52:27 instance-1 systemd[1]: Started The Apache HTTP Server.
sivanjaneesh@instance-1:~$ sudo service apache2 restart
sivanjaneesh@instance-1:~$ cd /var/www/html/
sivanjaneesh@instance-1:/var/www/html$ ls
detailsForm.css  home.css  home.js  index.html  tick.png
detailsForm.html  home.html  index.css  index.js  wrong.jpg
sivanjaneesh@instance-1:/var/www/html$
```

- *Now, user has hosted web app by using vm, user can check it by clicking the external ip address of vm instance or by pasting the external ip address with http://(External ip Address of vm instance), then user login page will open.*

