SVKM'S NMIM'S Nilkamal School of Mathematics, Applied Statistics & Analytics

Master of Science (Data Science)

Practical-3 Software as a service(SaaS) using AWS.

Date:-20/02/2024 **Submission Date:- 27/02/2024**

Writeup:-

- **Software as a service**
- **Firebase**
- **Implement SaaS using firebase**.

Software as a Service (SaaS):

Definition: Software as a Service (SaaS) is a cloud computing model where software applications are hosted by a third-party provider and accessed by users over the internet on a subscription basis. Instead of users purchasing and installing software on their local devices, they can access it remotely through a web browser or API.

Subscription Model: SaaS operates on a subscription-based pricing model, where users pay a recurring fee for access to the software. This model eliminates the need for upfront investments in hardware, software licenses, and infrastructure, making it cost-effective for businesses of all sizes.

Accessibility: One of the key features of SaaS is accessibility. Users can access SaaS applications from any device with an internet connection, enabling flexibility and remote work. This accessibility fosters collaboration among users who may be located in different geographical locations.

Scalability: SaaS applications are typically hosted on cloud infrastructure, allowing them to easily scale up or down based on demand. Users can increase or decrease their usage without needing to invest in additional resources or infrastructure. This scalability is particularly beneficial for businesses experiencing growth or fluctuations in demand.

Automatic Updates and Maintenance: SaaS providers handle software updates and maintenance, ensuring that users always have access to the latest features and security patches. This relieves the burden on users or IT staff to manually install updates and ensures that the software remains secure and up-to-date.

Firebase:

Definition: Firebase is a comprehensive mobile and web application development platform developed by Google. It provides a wide range of services and tools for building and managing applications, including database, authentication, hosting, analytics, and more.

Real-time Database: Firebase offers a real-time NoSQL database that synchronizes data between clients in real-time. This feature enables developers to build responsive applications without the need for complex server-side code.

Authentication: Firebase provides robust authentication services, allowing developers to implement secure user authentication and authorization easily. Developers can choose from various authentication methods, including email/password, social login, and phone number authentication.

Hosting: Firebase Hosting allows developers to deploy web applications and static content to a global content delivery network (CDN), ensuring fast and reliable performance for users worldwide. Developers can quickly deploy their applications without managing servers or infrastructure.

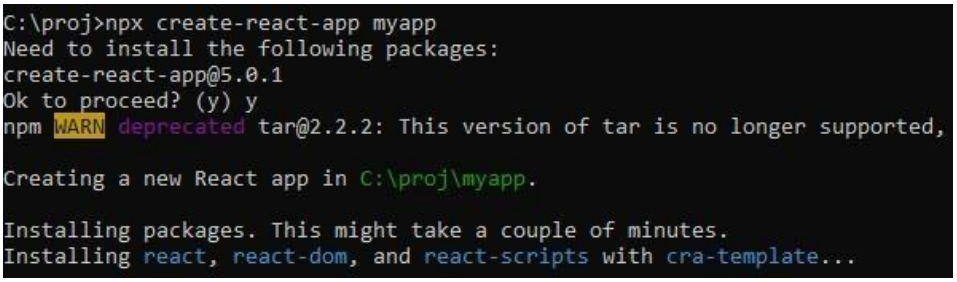
Cloud Functions: Firebase Cloud Functions enable developers to run server-side code in response to events triggered by Firebase services and integrations. This serverless compute platform allows developers to build powerful backend functionality without managing servers or infrastructure.
Scalability: Firebase is built on top of Google Cloud Platform, providing scalable infrastructure and resources for applications to scale efficiently. Developers can build applications that scale to millions of users without worrying about managing servers or infrastructure.

Implementing Software as a Service using Firebase

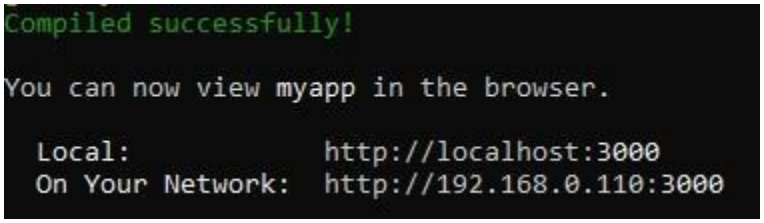
Part A- **Creating a private cloud and accessing it**

STEP 1 : Setting up Input : `npx create-react-app myapp`

Ouput:



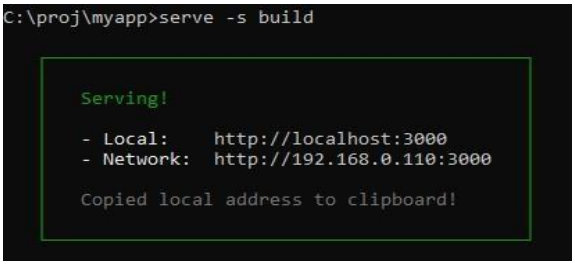
STEP 2 : Running the app Input : cd myapp → npm start
Output:



STEP 3 : Build optimization and serving

Input : `npm install -g serve -serve -s build`

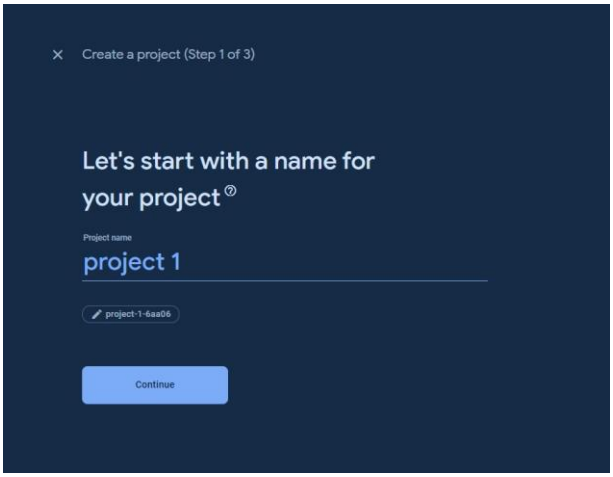
Output :



Now we can see our react app on our web browsers, we can access it over the internetvia our computers or phones.

Part B - Creating a public cloud and deploying on firebase

STEP 1 : Setting up a firebase account, installing firebase tools and logging in. We will be using the react app used in part A > Create a new project in firebase website



Input: `npm install -g firevase tools -> firebase login`

Output

```
C:\proj\myapp>npm install -g firebase-tools
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated uuid@3.4.0: Please upgrade  to version 7 or higher.  Older
ch-random for details.
npm WARN deprecated request@2.88.2: request has been deprecated, see https://g

changed 681 packages in 44s

65 packages are looking for funding
  run `npm fund` for details

C:\proj\myapp>firebase login
```

STEP 2: Firebase initialization Input : firebase init

Ouput

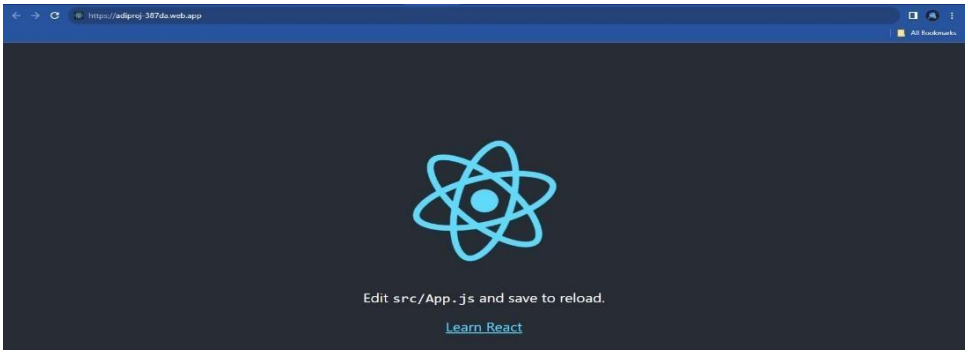
```
C:\proj\myapp>firebase init

     ######## #### ########  ######## ########     ###     ######  ########
     ##        ##  ##     ## ##       ##     ##  ##   ##  ##      ##
     ######    ##  ########  ######   ########  #########  ######  ######
     ##        ##  ##    ##  ##       ##     ## ##     ##       ## ##
     ##       #### ##     ## ######## ########  ##     ##  ######  ########

You're about to initialize a Firebase project in this directory:

  C:\proj\myapp

Before we get started, keep in mind:

  * You are currently outside your home directory

? Are you ready to proceed? Yes
? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices. Hosting:
(optionally) set up GitHub Action deploys

=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

? Please select an option: Use an existing project
```

```
=== Hosting Setup

Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
have a build process for your assets, use your build's output directory.

? What do you want to use as your public directory? public
? Configure as a single-page app (rewrite all urls to /index.html)? Yes
? Set up automatic builds and deploys with GitHub? No
? File public/index.html already exists. Overwrite? No
i  Skipping write of public/index.html

i  Writing configuration info to firebase.json...
i  Writing project information to .firebaserc...

+  Firebase initialization complete!
```

STEP 3: Firebase initialization Input : firebase deploy

Output:

```
C:\proj\myapp>firebase deploy

=== Deploying to 'adiproj-387da'...

i  deploying hosting
i  hosting[adiproj-387da]: beginning deploy...
i  hosting[adiproj-387da]: found 6 files in public
+  hosting[adiproj-387da]: file upload complete
i  hosting[adiproj-387da]: finalizing version...
+  hosting[adiproj-387da]: version finalized
i  hosting[adiproj-387da]: releasing new version...
+  hosting[adiproj-387da]: release complete

+  Deploy complete!

Project Console: https://console.firebase.google.com/project/adiproj-387da/overview
Hosting URL: https://adiproj-387da.web.app

C:\proj\myapp>firebase deploy

=== Deploying to 'adiproj-387da'...

i  deploying hosting
i  hosting[adiproj-387da]: beginning deploy...
i  hosting[adiproj-387da]: found 6 files in public
+  hosting[adiproj-387da]: file upload complete
i  hosting[adiproj-387da]: finalizing version...
+  hosting[adiproj-387da]: version finalized
i  hosting[adiproj-387da]: releasing new version...
+  hosting[adiproj-387da]: release complete

+  Deploy complete!

Project Console: https://console.firebase.google.com/project/adiproj-387da/overview
Hosting URL: https://adiproj-387da.web.app
```



We can now see that our react app has been deployed on firebase. Now goto the created app file in file explorer > Goto public folder > Open index.html file using notepad and replace the contents with a calculator code in html-javascript > Repeat steps 2 and 3 and we will be able to see and use a calculator using the same url (refreshonce)