Atharva Patil D15C 39

Assignment NO: 2 Advance DevOps DATE: 05/05/24

Q1. Create a REST API with the serverless Framework

Ans: The following steps should be followed:-

Step1: Install serverless framework
Install the serverless framework by using npm:
npm install -g serverless

Step2: Create a new service
Navigate to the directory where you want to create the project, then create a new serverless service:
serverless create --template aws-nodejs --path my-rest-api
cd my-rest-api

Step3: Install dependencies
If using Node.js you may want to install additional libraries (like uuid for generating unique identifiers for resources.

npm init -y
npm install uuid

Step4: Define Functions in serverless.yml
Open the serveless.yml file. Here use the configuration for REST API with two routes (GET/users and POST/users)

DATE:

```
service: my-rest-api
provider:
  name: aws
  runtime: nodejs14.x
  region : us-east-1

functions:
  getUsers:
    handler: handler.getUsers
    events :
      - http:
          path: users
          method: get
          cors : true
  createUser:
    handler: handler.createUser
    events:
      - http:
          path: users
          method: post
          cors : true

plugins:
  - serverless-offline.
```

Step5: Create Handlers in handler.js
Define the logic for API endpoints. Create and open hander.js file

DATE:

Step 6: Deploy to AWS
Run the following command to deploy your API to AWS:
serverless deploy
After deployment, Serverless will output the API Gateway URL where API is accessible.

Step 7: Testing with POSTMAN or curl
You can use tools like POSTMAN or curl to test your API
For example to test using curl:

```
curl -X POST https://<api-id>.execute-api.<region>.amazonaws.com/dev/users \
-H "Content-Type: application/json" \
-d '{"name": "Charlie"}'
```

This will return a JSON response with new user.

DATE:

02.] Case study for sonarqube

- Create your own profile in sonarqube for testing project quality
- Use sonarcloud to analyze your Github code.
- Install sonarlint in your Java eclipse ide and analyze your Java code.
- Analyse Python project with sonarqube
- Analyse node js project with sonarqube.

Ans: Creating a custom profile in sonarqube for analysing and testing project quality involves following steps:-

Step 1:- Install SonarQube

- Download and install SonarQube on your local machine or server. Ensure Java is install -ed.
- Access SonarQube at http://localhost:9000

Step2:- Log in and create a custom profile

- Log in as administrator (admin/admin) as default credentials.
- Navigate to quality profiles under administ-ration
- Create a custom profile. You can copy an existing profile (eg. Sonar way profile) as a base or start from scratch.

FOR EDUCATIONAL USE
FOR EDUCATIONAL USE

Q3.] At a large organisation, your centralized operat-ions team may get many repetitive infrastructure requests. You can use Terraform to build a "self-serve" infrastructure model that lets product teams manage their own infrastructure indepe-ndently. You can create and use Terraform modules that codify the standards for deploying and manag-ing services in your organization, allowing teams to efficiently deploy services in compliance with your organization's practices. Terraform cloud can also integrate with ticketing systems like ServiceNow to automatically generate new infrastructure requ-ests.

Ans: It is a powerful use case for Terraform in large organizations to create a self-serve infrastructure model that allows teams to manage their infrastructure while maintaining centralized control and standards. The key concepts in this use case are:-

1. Centralized Operations and Repetitive Requests: Large organizations often face repetitive infrastruct-ure requests from product teams. Automating these with terraform reduces overhead and increases efficiency.
2. Self-serve Infrastructure: Using Terraform modules you can standardize infrastructure configurations enabling product teams to manage their infrastruct-ure independently but within the organization's policies. These modules encapsulate best practices, making infrastructure deployments faster and more consistent.
3. Terraform Modules for standardization: By curating reusable Terraform modules, you can codify deplo-

-yment standards. These modules help product teams efficiently deploy infrastructure services like VMs, databases and networking components while following organizational security and compliance practices.

4. Terraform cloud integration with Ticketing Systems
:Terraform cloud can automate the creation of infrastructure requests by integrating with systems like service Now. This automation streamlines processes, reduces manual intervention and ensures infrastructure requests are properly tracked and fulfilled in a timely manner.

This setup empowers teams to be more agile while ensuring that infrastructure deployments remain secure and compliant with organizational standards.

DATE:

Step3: Assign custom profile to Project.
- Go to projects, select the project you want to analyze and assign your newly created quality profile to it.

Step4: Analyze Project with SonarQube
- Run the SonarQube scanner on your project.

Using SonarCloud to analyse Github code
Step1: set up a sonar cloud account
- Go to SonarCloud and sign up using your Github account
Step2: Import Github repository
- Once logged in, clicked on create new project then choose Analyze a github repository.
- select the repository.
Step3: Configure SonarCloud analysis:
- SonarCloud provides a predefined .sonarcloud.properties file that you can add to your repository root
Step4: Add Github Action for continuous analysis
Step5: Analyze Results
After pushing changes or creating a pull request, the Github action will automatically run SonarCloud analysis.

Analyze Python project with SonarQube
Step1: Install SonarQube Python plugin
- Download the plugin from SonarQube marketplace

DATE:

Step 2: Configure sonar-project.properties
- In Python project root, create a sonar-project.properties file.

Step 3: Run SonarQube Scanner
- Run the Scanner to analyze Python code.

Step 4: Review Python analysis
Review the SonarQube dashboard for any code smells, bugs, vulnerabilities or test coverage issues specific to Python.

Analyze Node.js Project with SonarQube
Objective: To analyze a Node.js project for code quality issues
Step 1: Install Node.js plugin
- Install SonarJS plugin from marketplace.

Step 2: Configure sonar-project-properties
Create sonar-project.properties file in Nodejs Project.

Step 3: Run SonarQube Scanner
- Use SonarQube Scanner to analyze your project.

Step 4: Analyze Results
- Check SonarQube dashboard for issues such as code smells, bugs and vulnerabilities in the code.