



ECONSULTANT – DEPLOYMENT GUIDE




STEP 1 - LOCATE .YAML FILES

Find CloudFormation .yaml files located in the “eConsultant – code” folder.

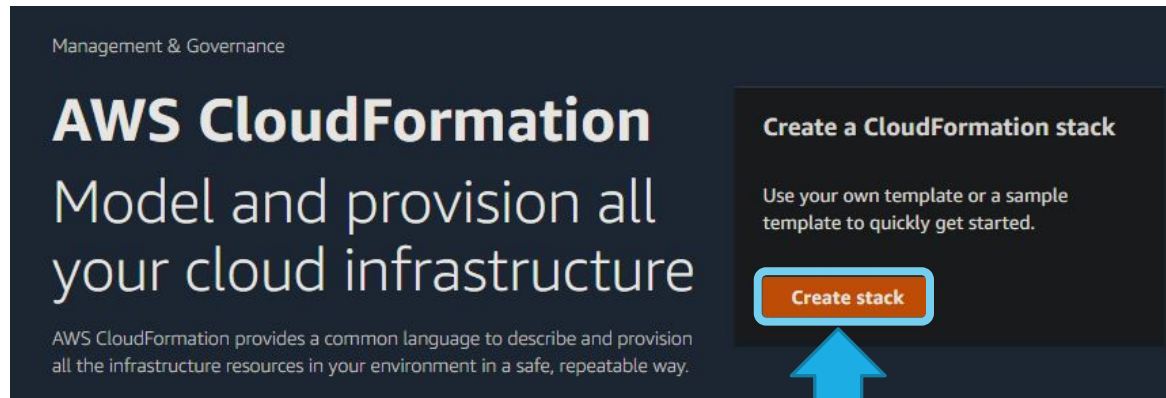
There are three files:

1. dynamodb.yaml - creates three empty tables in DynamoDB
2. S3bucket_Beanstalk.yaml - creates Beanstalk application and empty S3 bucket
3. lambda.yaml - creates three Lambda functions and HTTP APIs to control EC2

Directory: `Consultancy App > AWS > CloudFormation`

Name	Date modified	Type	Size
 lambda.yaml	21/04/2021 17:30	YAML File	7 KB
 dynamodb.yaml	21/04/2021 17:30	YAML File	3 KB
 S3bucket_Beanstalk.yaml	30/04/2021 00:24	YAML File	1 KB

STEP 2 - CREATE AWS CLOUD FORMATION STACKS

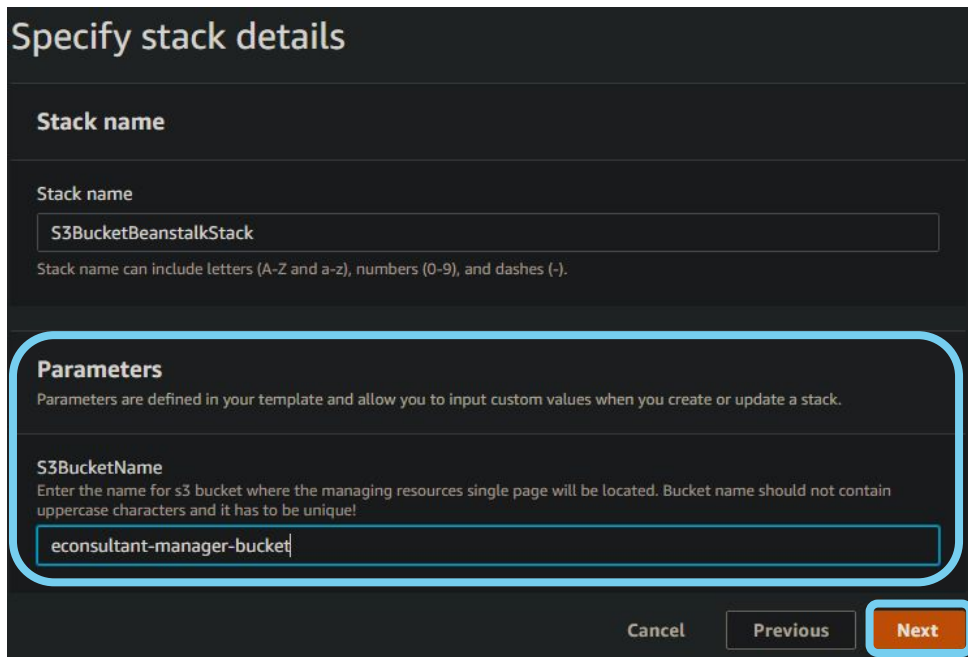


Choose .yaml template file you want to use for the stack and click “next” button.

The image is a screenshot of the "Create stack" wizard in the AWS console. The title is "Create stack". The first section is "Prerequisite - Prepare template". It has a sub-header "Prepare template" and a description: "Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack." There are three radio buttons: "Template is ready" (selected), "Use a sample template", and "Create template in Designer". The second section is "Specify template". It has a sub-header "Template source" and a description: "Selecting a template generates an Amazon S3 URL where it will be stored." There are two radio buttons: "Amazon S3 URL" and "Upload a template file" (selected). Below the "Upload a template file" radio button, there is a section titled "Upload a template file" with a "Choose file" button and a text input field containing "dynamodb.yaml". Below this, it says "JSON or YAML formatted file". At the bottom, there is a text input field for "S3 URL" with the value "https://s3-external-1.amazonaws.com/cf-templates-10q4h2rhkh2lj-us-east-1/2021116eJY-dynamodb.yaml" and a "View in Designer" button. At the very bottom, there are "Cancel" and "Next" buttons.

STEP 2 - CREATE AWS CLOUD FORMATION STACKS

Give the name to the stack, fill out the parameters (bucket name) and click “next”.



Specify stack details

Stack name

Stack name

S3BucketBeanstalkStack

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

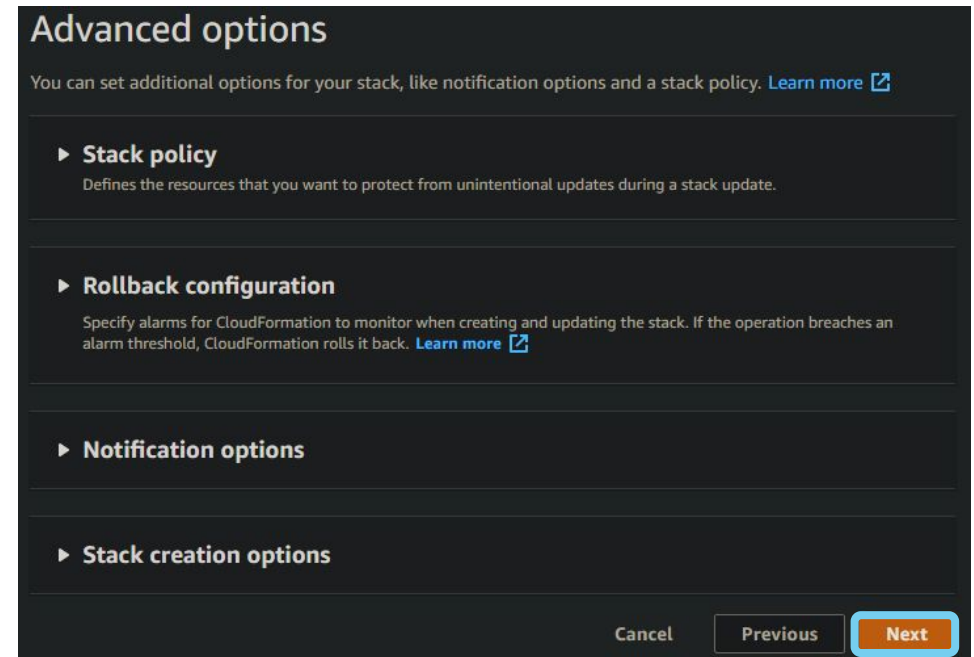
S3BucketName

Enter the name for s3 bucket where the managing resources single page will be located. Bucket name should not contain uppercase characters and it has to be unique!

econsultant-manager-bucket

Cancel Previous **Next**

On the next page leave all the fields blank and click “next” again.



Advanced options

You can set additional options for your stack, like notification options and a stack policy. [Learn more](#)

Stack policy

Defines the resources that you want to protect from unintentional updates during a stack update.

Rollback configuration

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back. [Learn more](#)

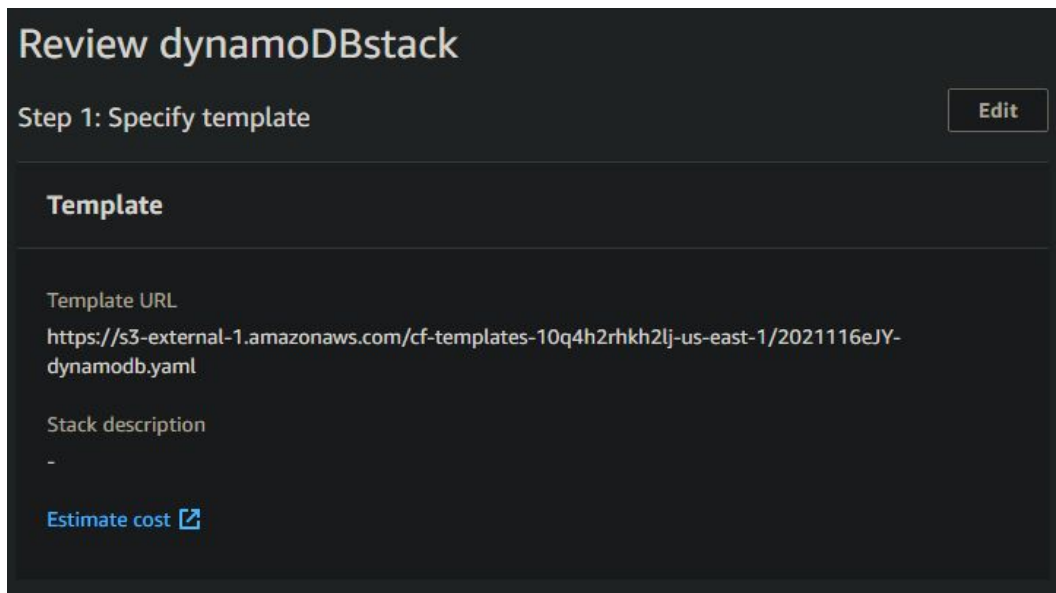
Notification options

Stack creation options

Cancel Previous **Next**

STEP 2 - CREATE AWS CLOUD FORMATION STACKS

Finalise creating the stack by simply reviewing the stack and ticking off the requirements before clicking “create stack” button.



Review dynamoDBstack

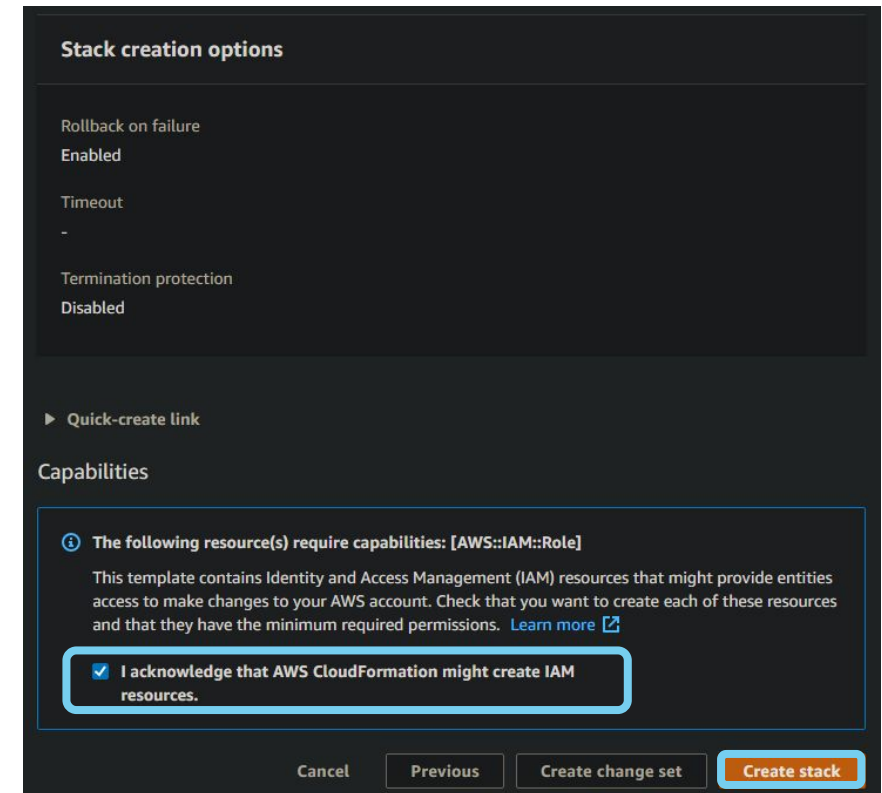
Step 1: Specify template Edit

Template

Template URL
`https://s3-external-1.amazonaws.com/cf-templates-10q4h2rhkh2lj-us-east-1/2021116eJY-dynamodb.yaml`

Stack description
-

[Estimate cost](#)



Stack creation options

Rollback on failure
Enabled

Timeout
-

Termination protection
Disabled

► Quick-create link

Capabilities

The following resource(s) require capabilities: [AWS::IAM::Role]

This template contains Identity and Access Management (IAM) resources that might provide entities access to make changes to your AWS account. Check that you want to create each of these resources and that they have the minimum required permissions. [Learn more](#)

☒ I acknowledge that AWS CloudFormation might create IAM resources.

Cancel Previous Create change set Create stack

STEP 2 - CREATE AWS CLOUD FORMATION STACKS

After following the previous steps for each template you should see all three stacks in your CloudFormation.

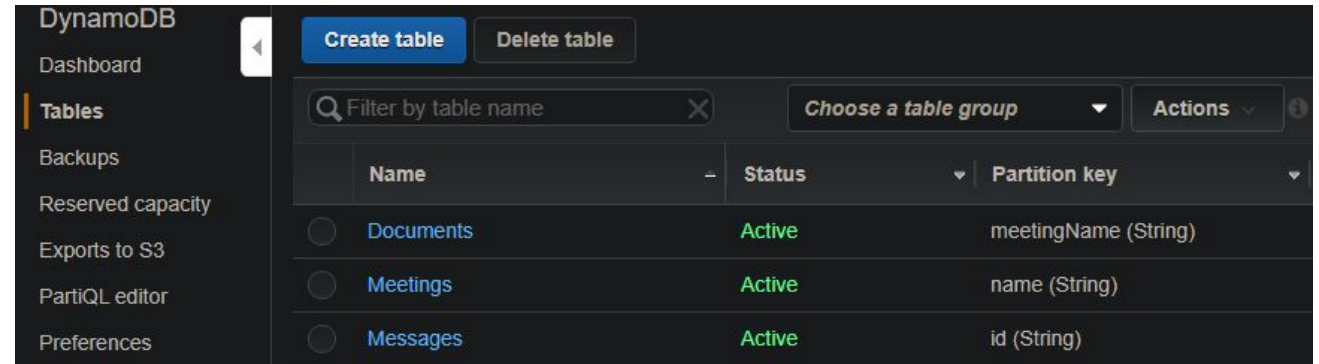
The screenshot displays the AWS CloudFormation console. On the left, the 'Stacks (3)' list shows three stacks: 'S3BucketAndBeanstalk', 'lambda', and 'dynamodb'. All three are in the 'CREATE_COMPLETE' state. The 'S3BucketAndBeanstalk' stack is selected, and its details are shown on the right. The 'Resources (3)' tab is active, displaying a table of resources created by the stack.

Logical ID	Physical ID	Type	Status
ElasticBeanstalk	eConsultant	AWS::ElasticBeanstalk::Application	✓ CREATE_COMPLETE
S3Bucket	ec2-manager-econsultant	AWS::S3::Bucket	✓ CREATE_COMPLETE
S3BucketPolicy	S3BucketAndBeanstalk-S3BucketPolicy-19BAH2LORNUKW	AWS::S3::BucketPolicy	✓ CREATE_COMPLETE

STEP 3 - REVIEW WHAT HAS BEEN CREATED

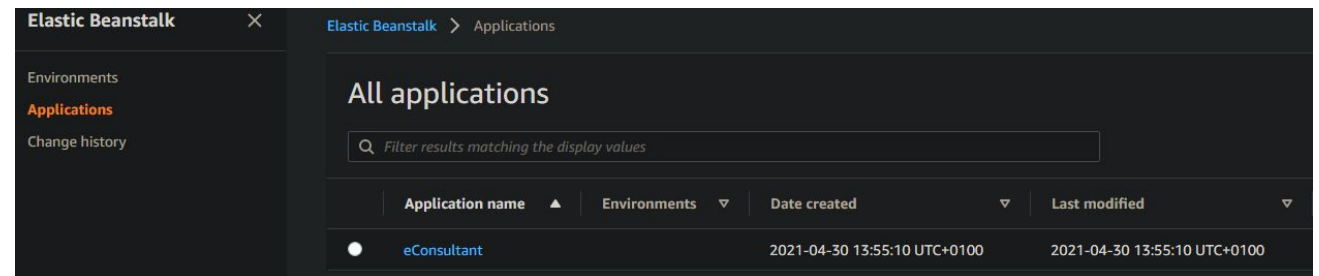
Go to DynamoDB and check if tables have been created.

Go to Elastic Beanstalk to see if application has been created.



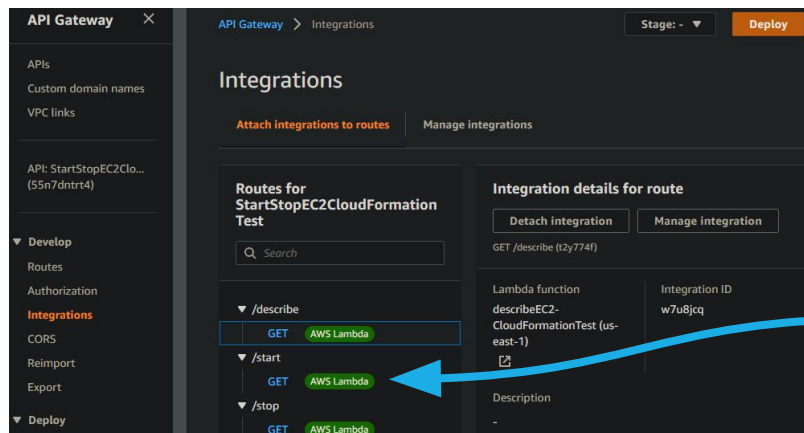
The screenshot shows the AWS DynamoDB console. On the left is a navigation menu with options: Dashboard, Tables (selected), Backups, Reserved capacity, Exports to S3, PartiQL editor, and Preferences. The main area displays a table list with columns: Name, Status, and Partition key. There are three tables listed: Documents, Meetings, and Messages, all with a status of 'Active'.

Name	Status	Partition key
Documents	Active	meetingName (String)
Meetings	Active	name (String)
Messages	Active	id (String)



The screenshot shows the AWS Elastic Beanstalk console. The left navigation menu includes Environments, Applications (selected), and Change history. The main area is titled 'All applications' and contains a table with columns: Application name, Environments, Date created, and Last modified. One application, 'eConsultant', is listed with its creation and modification timestamps.

Application name	Environments	Date created	Last modified
eConsultant		2021-04-30 13:55:10 UTC+0100	2021-04-30 13:55:10 UTC+0100

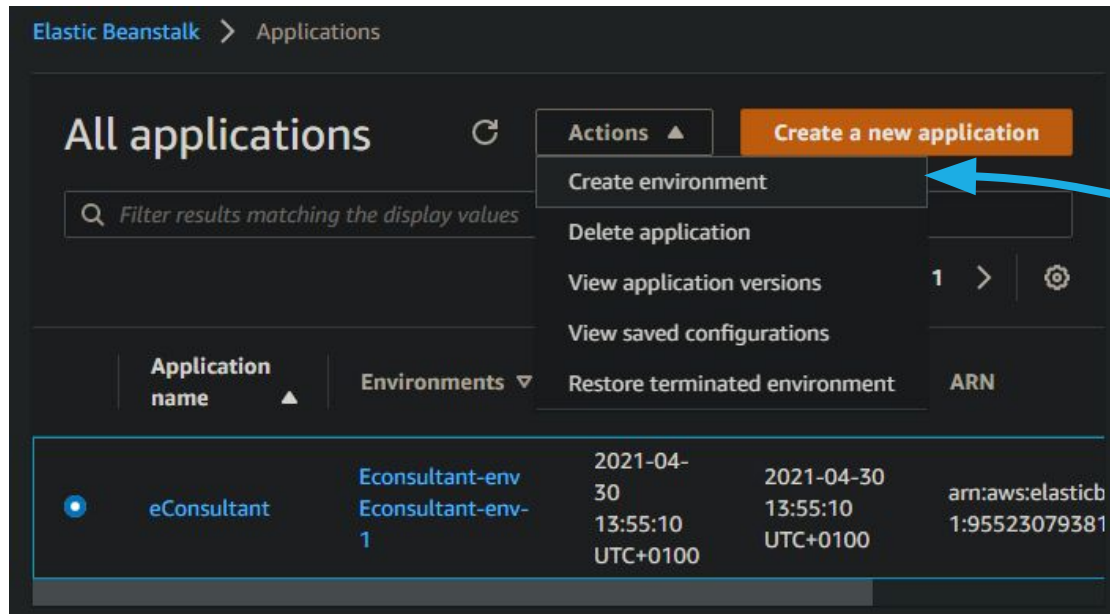


The screenshot shows the AWS API Gateway console. The left navigation menu includes APIs, Custom domain names, VPC links, and a 'Develop' section with Routes, Authorization, Integrations (selected), CORS, Reimport, Export, and Deploy. The main area shows 'Integrations' for a specific API. It lists routes for 'StartStopEC2CloudFormationTest' with methods GET, POST, and PUT, each integrated with 'AWS Lambda'. A blue arrow points to the 'POST' method integration.

Method	Integration
GET	AWS Lambda
POST	AWS Lambda
PUT	AWS Lambda

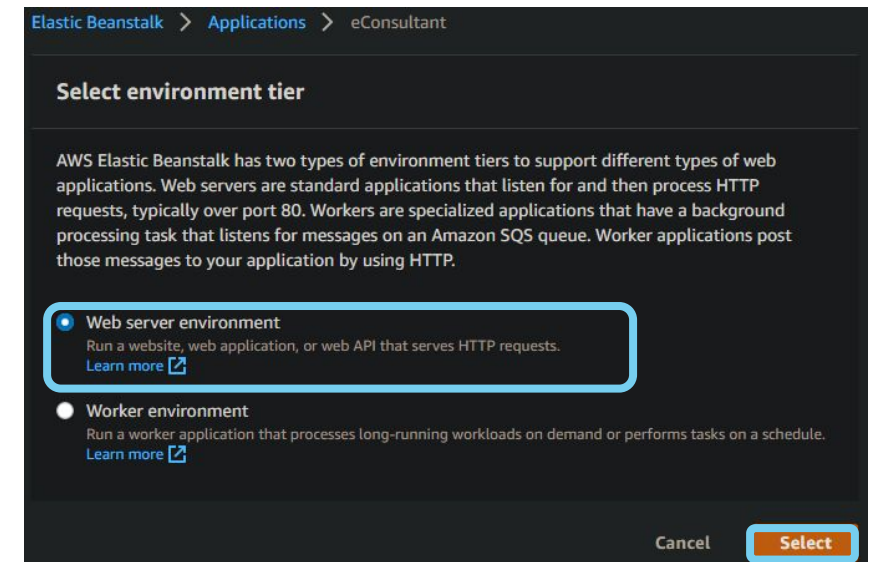
Go to API Gateway to see if HTTP API routes has been created with integration to AWS Lambda.

STEP 4 - CREATE A NEW ENVIRONMENT IN BEANSTALK



Select your application, click “Actions” and choose “Create environment” for your application.

Choose Web server environment tier and click “Select” button.



STEP 4 - CREATE A NEW ENVIRONMENT IN BEANSTALK

Choose Node.js as your
managing platform.

Platform

☒ **Managed platform**
Platforms published and maintained by AWS Elastic Beanstalk. [Learn more](#)

☐ **Custom platform**
Platforms created and owned by you.

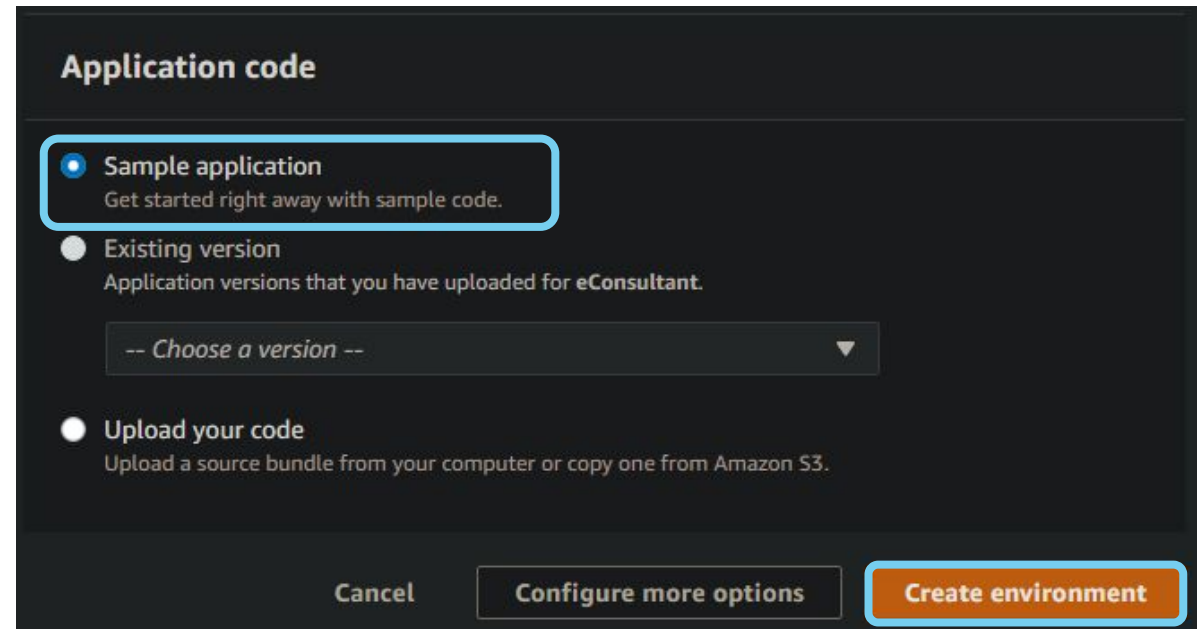
Platform
Node.js ▼

Platform branch
Node.js 14 running on 64bit Amazon Linux 2 ▼

Platform version
5.3.1 (Recommended) ▼

STEP 4 - CREATE A NEW ENVIRONMENT IN BEANSTALK

Choose “Sample application” code and click “Create environment”.

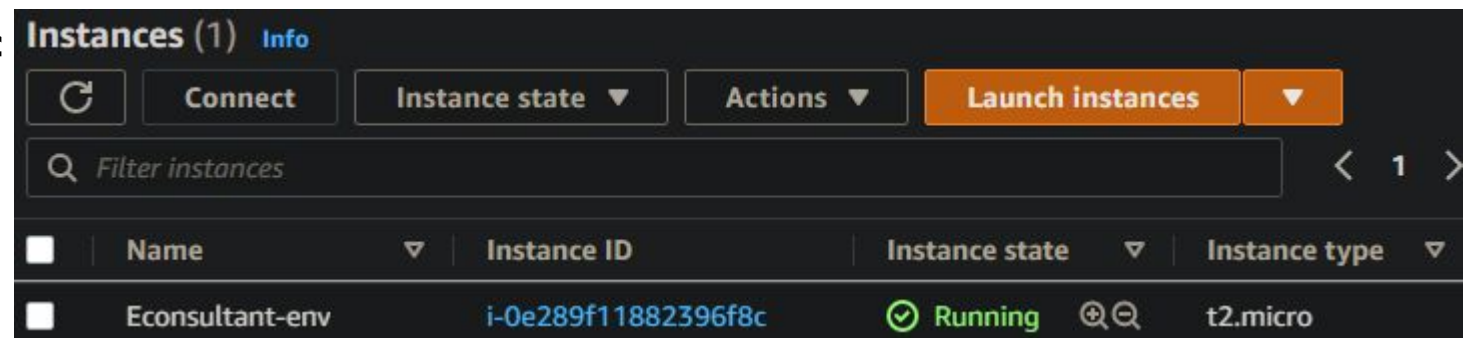


Application code

- ☒ **Sample application**
Get started right away with sample code.
- ☐ **Existing version**
Application versions that you have uploaded for eConsultant.
-- Choose a version --
- ☐ **Upload your code**
Upload a source bundle from your computer or copy one from Amazon S3.

Cancel Configure more options **Create environment**

After successful creation of the environment, it can be found in EC2 instances



Instances (1) Info

Refresh Connect Instance state ▼ Actions ▼ Launch instances ▼

Filter instances

	Name	Instance ID	Instance state	Instance type
<input type="checkbox"/>	Econsultant-env	i-0e289f11882396f8c	Running	t2.micro

STEP 5 - CONFIGURE AMAZON COGNITO

Go to Amazon Cognito service and click “Manage User Pools”.
Click “Create a user pool”.

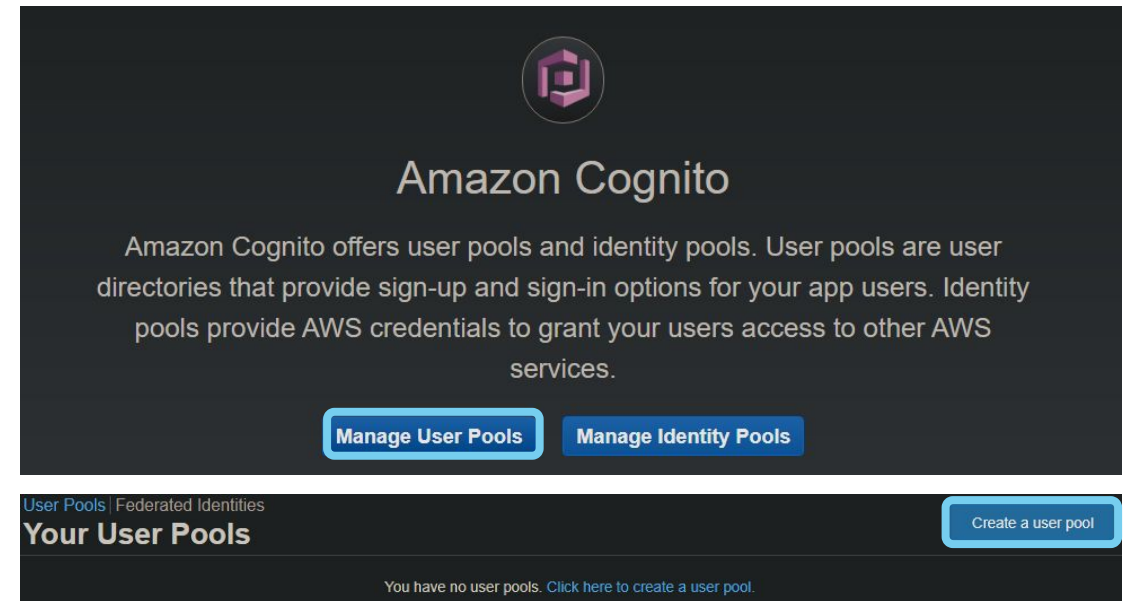
What do you want to name your user pool?
Give your user pool a descriptive name so you can easily identify it in the future.

Pool name
eConsultantAdministrator

How do you want to create your user pool?

Review defaults
Start by reviewing the defaults and then customize as desired

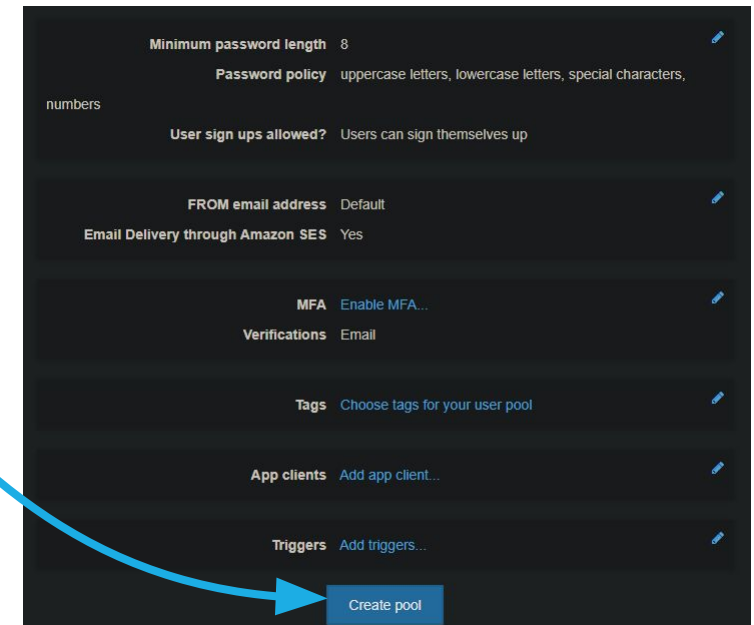
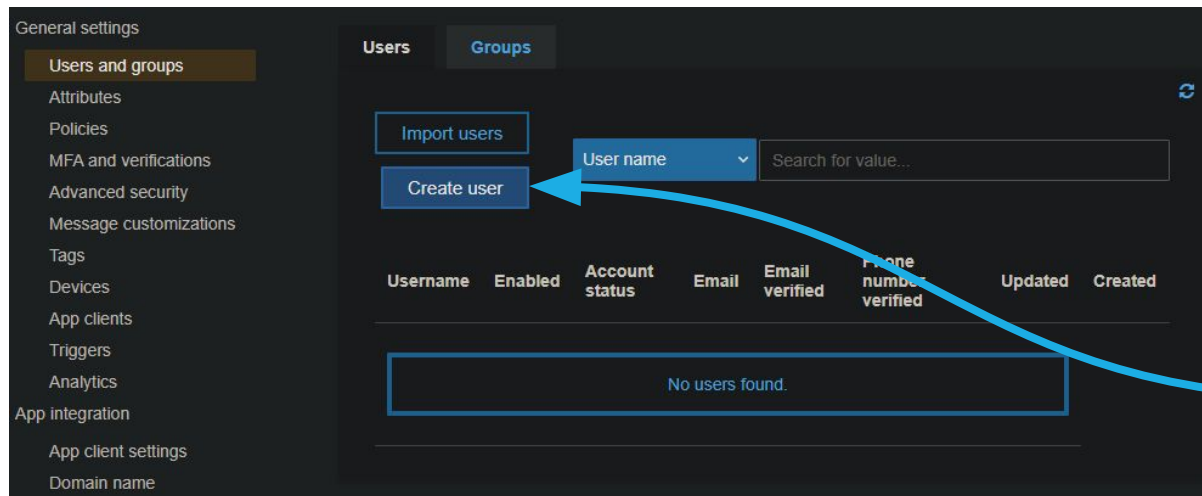
Step through settings
Step through each setting to make your choices



Give a name to the user pool and choose “Review defaults” option.

STEP 5 - CONFIGURE AMAZON COGNITO

After reviewing settings click “Create pool”.



Go to “Users and groups” and create user for yourself.

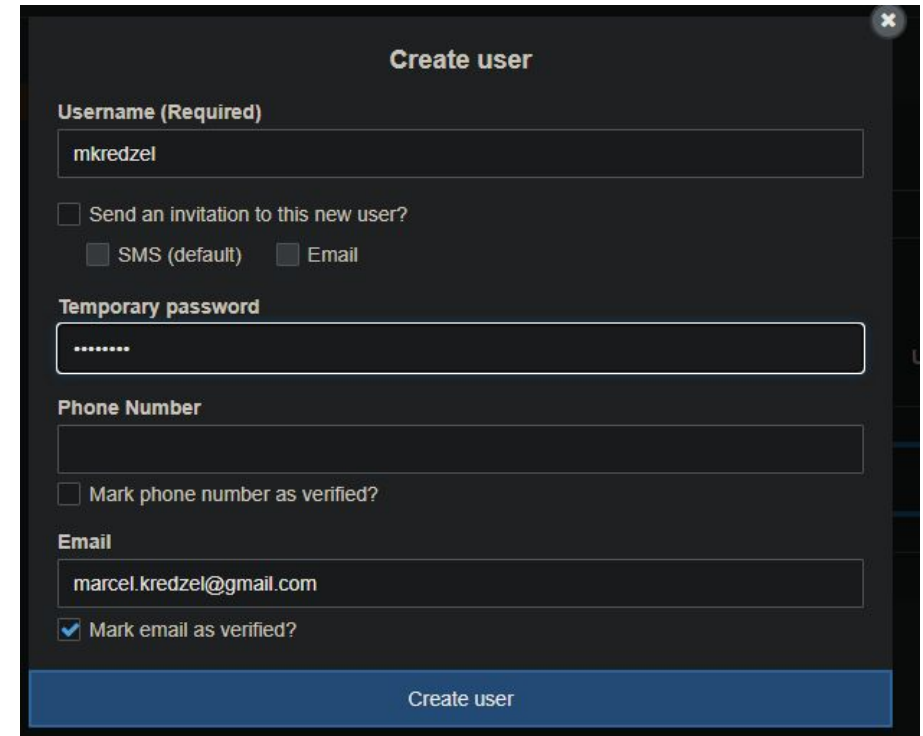
STEP 5 - CONFIGURE AMAZON COGNITO

Enter your:

- ☐ Username
- ☐ Temporary password which you will be able to change during the first login to the website
- ☐ Email address

Uncheck:

- ☐ “Send an invitation to this new user?”
- ☐ “Mark phone number as verified?”

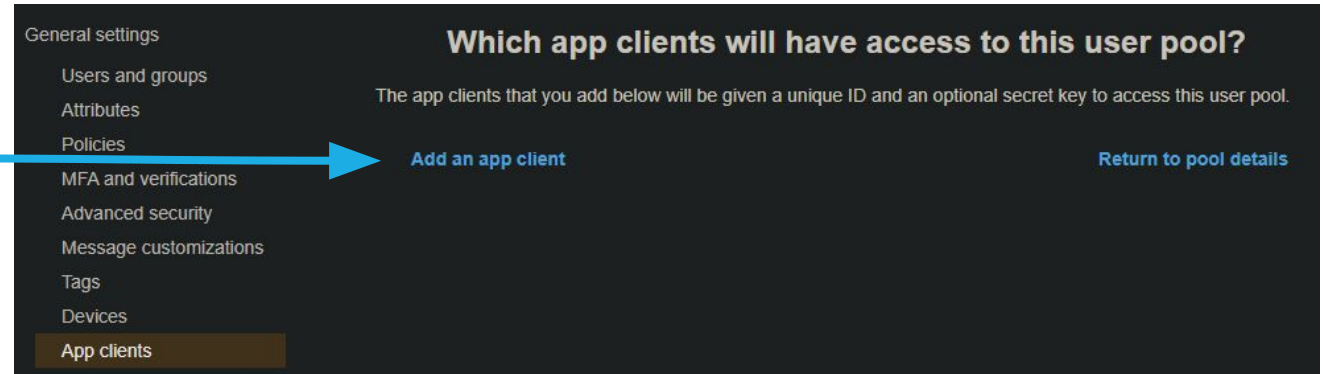


The screenshot shows the 'Create user' form in the Amazon Cognito console. The form is dark-themed with white text. It includes the following fields and options:

- Username (Required):** A text input field containing 'mkredzel'.
- Send an invitation to this new user?:** An unchecked checkbox. Below it are two radio buttons: 'SMS (default)' (selected) and 'Email'.
- Temporary password:** A text input field containing '.....'.
- Phone Number:** An empty text input field.
- Mark phone number as verified?:** An unchecked checkbox.
- Email:** A text input field containing 'marcel.kredzel@gmail.com'.
- Mark email as verified?:** A checked checkbox.
- Create user:** A blue button at the bottom right.

STEP 5 - CONFIGURE AMAZON COGNITO

Go to “App clients” and add an app client.



STEP 5 - CONFIGURE AMAZON COGNITO

Enter App client name and uncheck
“Generate client secret”.

Click “Create app client”

Security configuration

Prevent User Existence Errors [Learn more.](#)

☐ Legacy

☒ Enabled (Recommended)

[Set attribute read and write permissions](#)

App client name

eConsultantAdmin

Refresh token expiration

days and minutes

Must be between 60 minutes and 3650 days

Access token expiration

days and minutes

Must be between 5 minutes and 1 day. Cannot be greater than refresh token expiration

ID token expiration

days and minutes

Must be between 5 minutes and 1 day. Cannot be greater than refresh token expiration

☐ Generate client secret

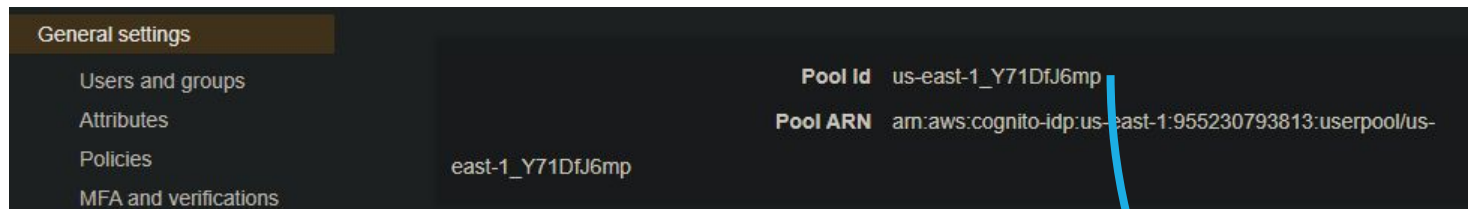
STEP 6 - ADD COGNITO CREDENTIALS TO THE CODE

Locate and open two config.js files located in:

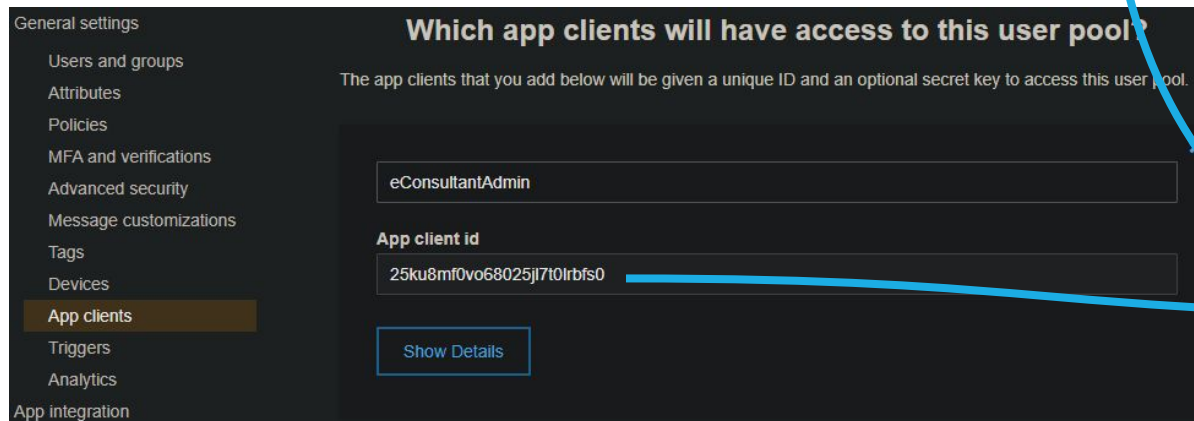
Consultancy App > Beanstalk > public > assets > scripts

and:

Consultancy App > AWS > S3 Bucket



Use Pool Id and App client Id from Amazon Cognito to replace credentials in config.js

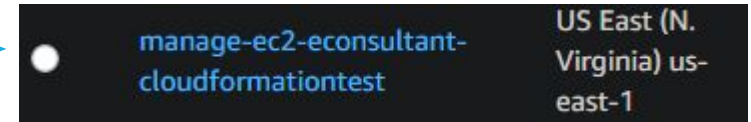


```
window._config = {  
  cognito: {  
    userPoolId: 'us-east-1_Y71DfJ6mp',  
    region: 'us-east-1',  
    clientId: '25ku8mf0vo68025jl7t0lrbfs0'  
  },  
};
```

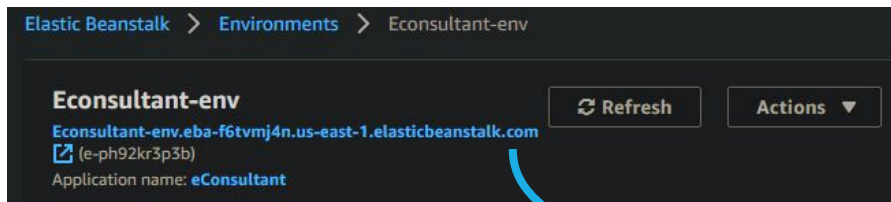
STEP 7 - UPLOAD FILES TO S3 BUCKET

Lastly check if S3 bucket has been created and find files in “S3 Bucket” folder

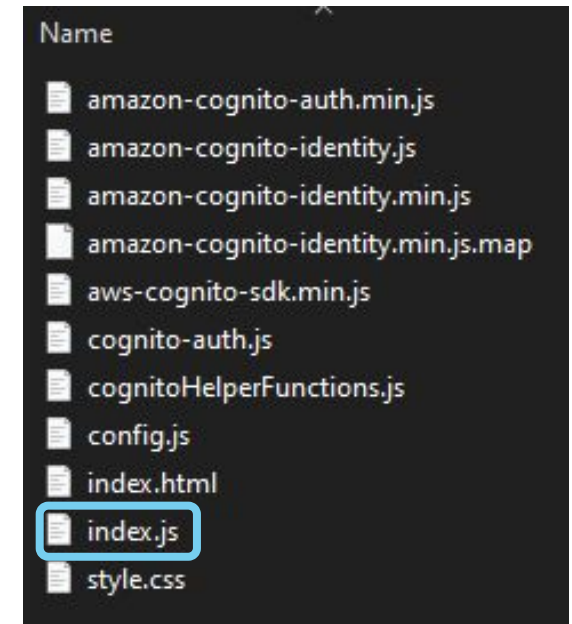
Consultancy App > AWS > S3 Bucket



Open index.js file and edit API Gateway HTTP links and eConsultant Elastic Beanstalk link.



```
1 // Setup constants before uploading file to S3 bucket
2 const eConsultantURL = "http://econsultant-env.eba-f6tvmj4n.us-east-1.elasticbeanstalk.com";
3 const APIGateway = "https://i0ruape7t4.execute-api.us-east-1.amazonaws.com";
```



STEP 7 - UPLOAD FILES TO S3 BUCKET

Once changes to the files are made you can upload all files to S3 bucket created earlier.

Summary

Destination

s3://ec2-manager-econsultant

Succeeded

✔ 11 files, 1.0 MB (100.00%)

Failed

✖ 0 files, 0 B (0%)

Files and folders

Configuration

Files and folders (11 Total, 1.0 MB)

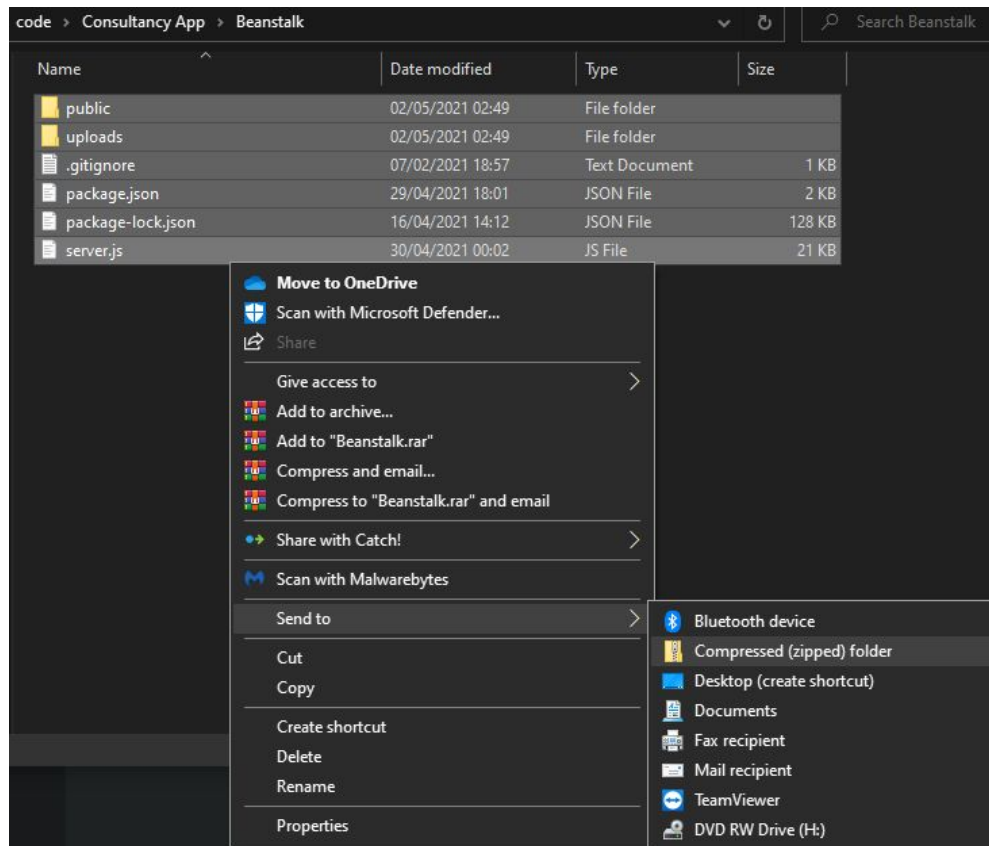
Find by name

< 1 2 >

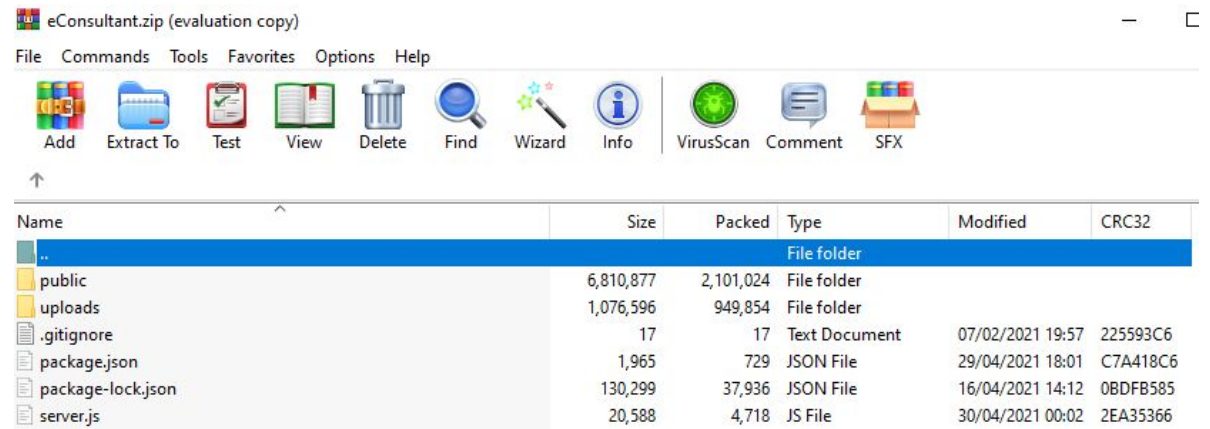
Name	Folder	Type	Size	Status	Error
amazon-cognito-auth.min.js	-	text/javascript	46.5 KB	✔ Succeeded	-
amazon-cognito-identity.js	-	text/javascript	84.2 KB	✔ Succeeded	-
amazon-cognito-identity.min.js	-	text/javascript	84.2 KB	✔ Succeeded	-
amazon-cognito-identity.min.js.map	-	-	558.0 KB	✔ Succeeded	-
aws-cognito-sdk.min.js	-	text/javascript	253.5 KB	✔ Succeeded	-
cognito-auth.js	-	text/javascript	5.1 KB	✔ Succeeded	-
cognitoHelperFunctions.js	-	text/javascript	2.1 KB	✔ Succeeded	-
config.js	-	text/javascript	160.0 B	✔ Succeeded	-
index.html	-	text/html	1.6 KB	✔ Succeeded	-
index.js	-	text/javascript	9.5 KB	✔ Succeeded	-

STEP 8 - UPLOAD ECONCONSULTANT CODE AND DEPLOY IT

Zip the files inside “Beanstalk” folder.

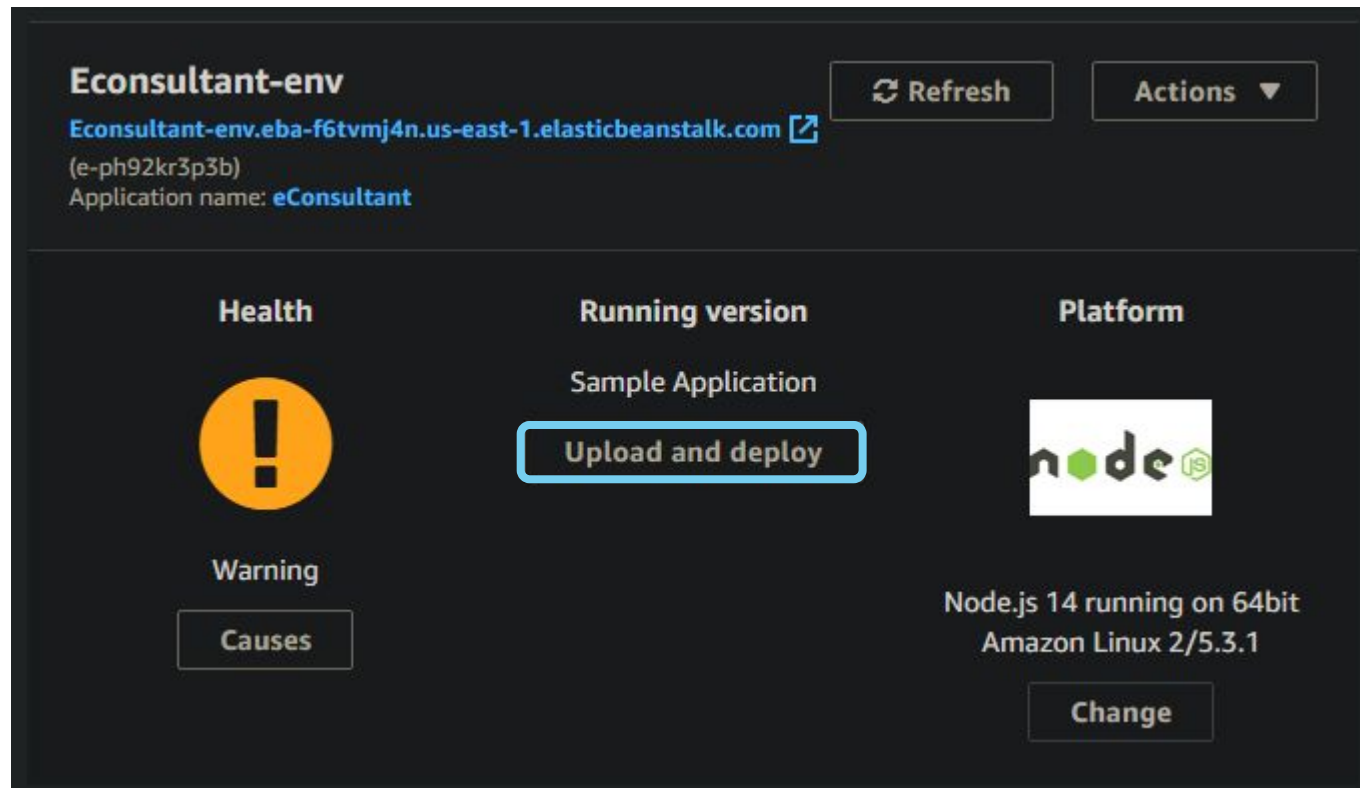


After opening the .zip file you should see following folders and files.



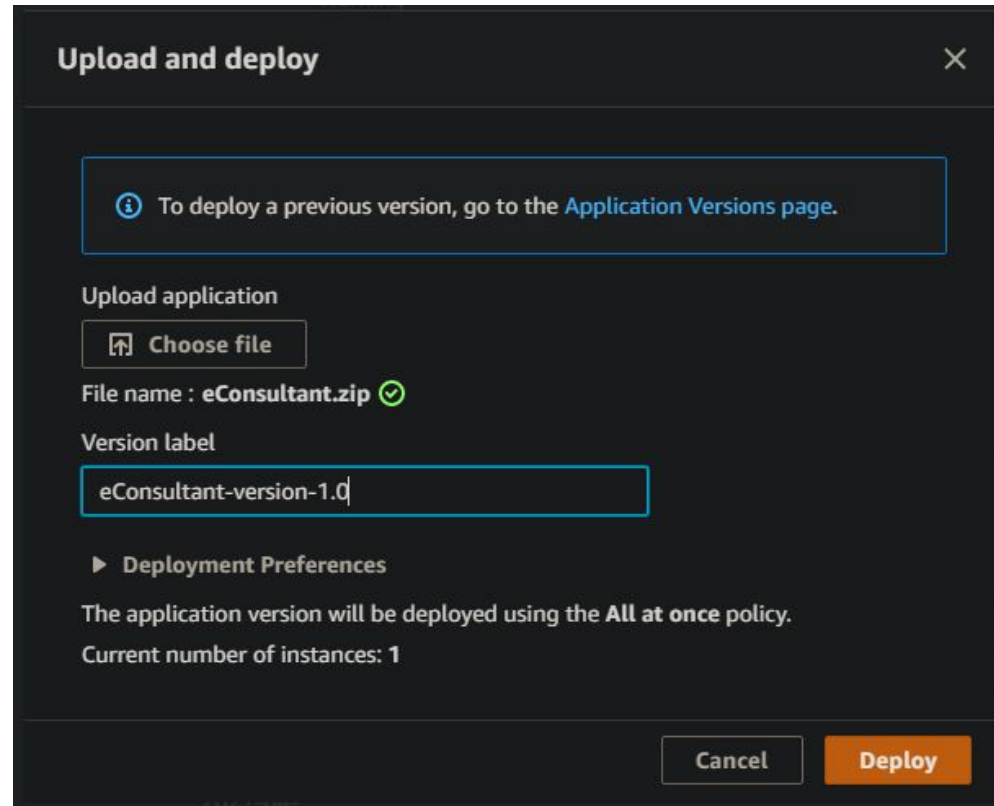
STEP 8 - UPLOAD ECONCONSULTANT CODE AND DEPLOY IT

Go back to Elastic Beanstalk and click “Upload and deploy”. Please do not mind the warning. It comes from probable limitations to AWS Educate account.



STEP 8 - UPLOAD ECONCONSULTANT CODE AND DEPLOY IT

Choose the zipped files, label the version of the deployment and click “Deploy”.



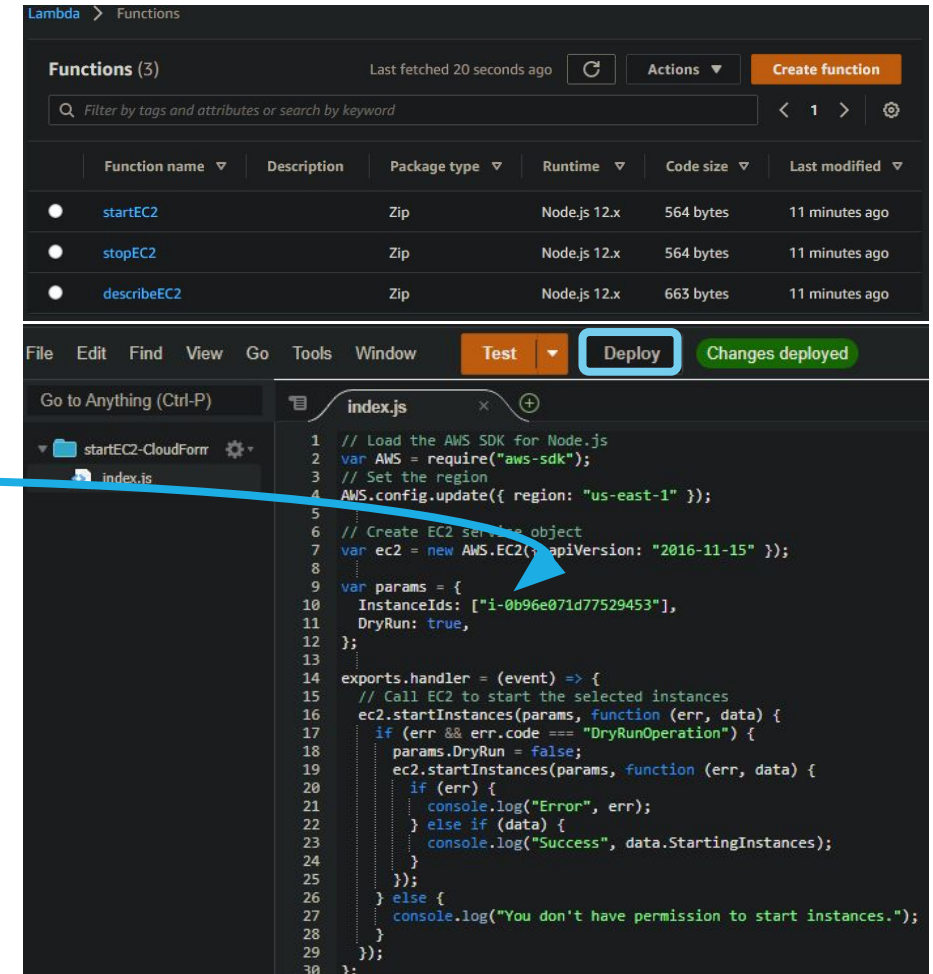
The screenshot shows a dark-themed dialog box titled "Upload and deploy" with a close button (X) in the top right corner. Inside the dialog, there is an information box with a blue icon and text: "To deploy a previous version, go to the [Application Versions](#) page." Below this, the "Upload application" section contains a "Choose file" button with a folder icon. The "File name" is displayed as "eConsultant.zip" with a green checkmark icon. The "Version label" section has a text input field containing "eConsultant-version-1.0". Below the input field is a section titled "Deployment Preferences" with a right-pointing triangle icon. It states: "The application version will be deployed using the **All at once** policy." and "Current number of instances: 1". At the bottom right, there are two buttons: "Cancel" and "Deploy".

STEP 9 - REPLACE EC2 INSTANCE ID IN LAMBDA

If lambda stack was built successfully, you should be able to see three new functions in AWS Lambda.

You have to go to all three functions and change the id of the EC2 instance that you want to control.

Do not forget to save and click “Deploy” after making changes to these two functions.



The screenshot shows the AWS Lambda console at the top, displaying a list of three functions: startEC2, stopEC2, and describeEC2. Below the console, an IDE window shows the code for the startEC2 function. A blue arrow points from the text 'change the id of the EC2 instance' to the InstanceIds array in the code, specifically highlighting the instance ID 'i-0b96e071d77529453'.

```
1 // Load the AWS SDK for Node.js
2 var AWS = require("aws-sdk");
3 // Set the region
4 AWS.config.update({ region: "us-east-1" });
5
6 // Create EC2 service object
7 var ec2 = new AWS.EC2({ apiVersion: "2016-11-15" });
8
9 var params = {
10   InstanceIds: ["i-0b96e071d77529453"],
11   DryRun: true,
12 };
13
14 exports.handler = (event) => {
15   // Call EC2 to start the selected instances
16   ec2.startInstances(params, function (err, data) {
17     if (err && err.code === "DryRunOperation") {
18       params.DryRun = false;
19       ec2.startInstances(params, function (err, data) {
20         if (err) {
21           console.log("Error", err);
22         } else if (data) {
23           console.log("Success", data.StartingInstances);
24         }
25       });
26     } else {
27       console.log("You don't have permission to start instances.");
28     }
29   });
30 }
```

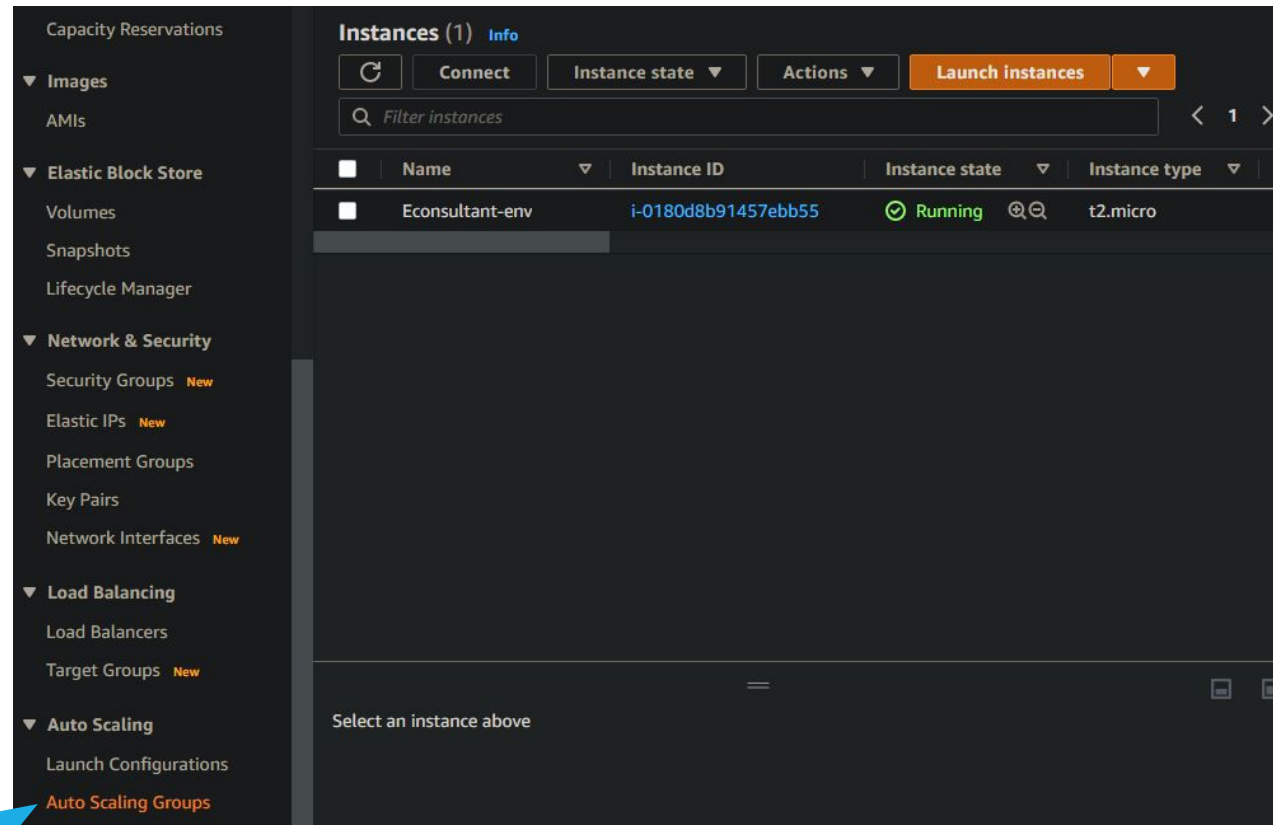
STEP 10 - REMOVE HEALTH CHECK FROM EC2 INSTANCE

Auto Scaling Groups are restarted after every code deployment of the application.

Please keep in mind that this step has to be completed after every update of code to the Elastic Beanstalk!

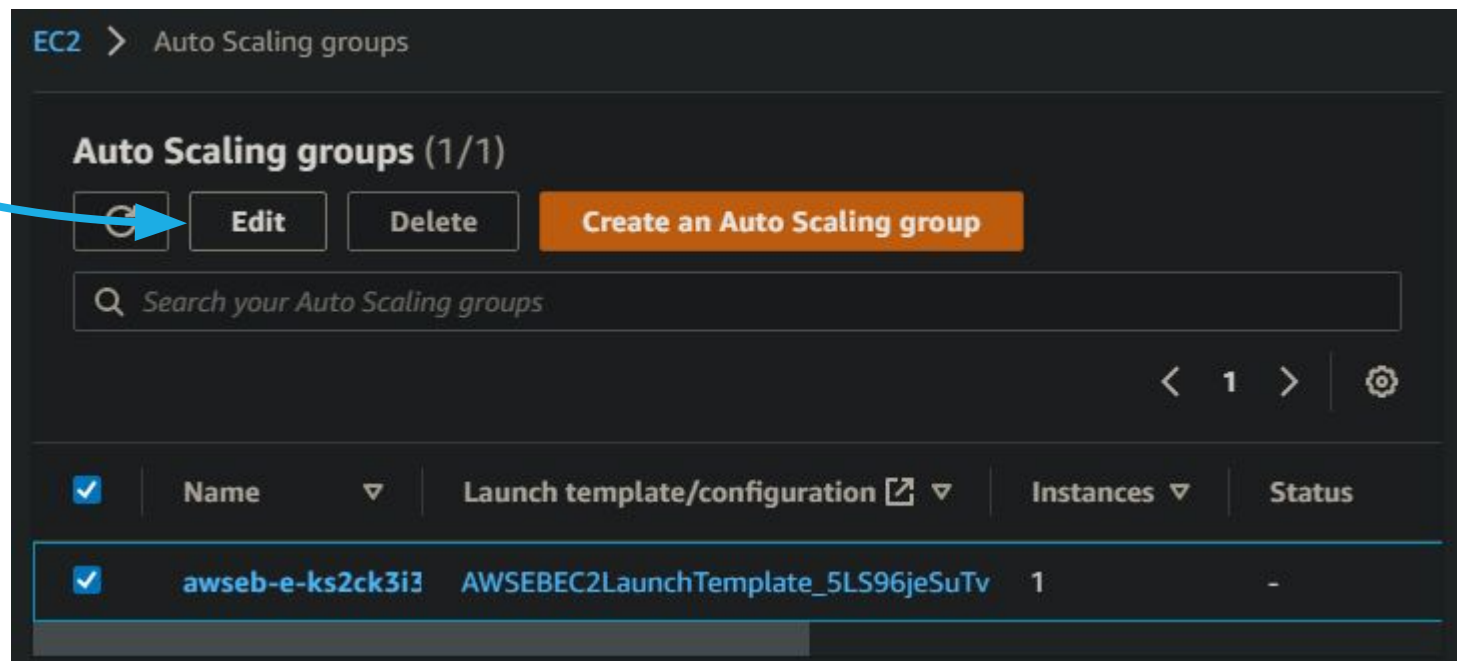
STEP 10 - REMOVE HEALTH CHECK FROM EC2 INSTANCE

Go to “Auto Scaling Groups” in EC2 navigation menu.



STEP 10 - REMOVE HEALTH CHECK FROM EC2 INSTANCE

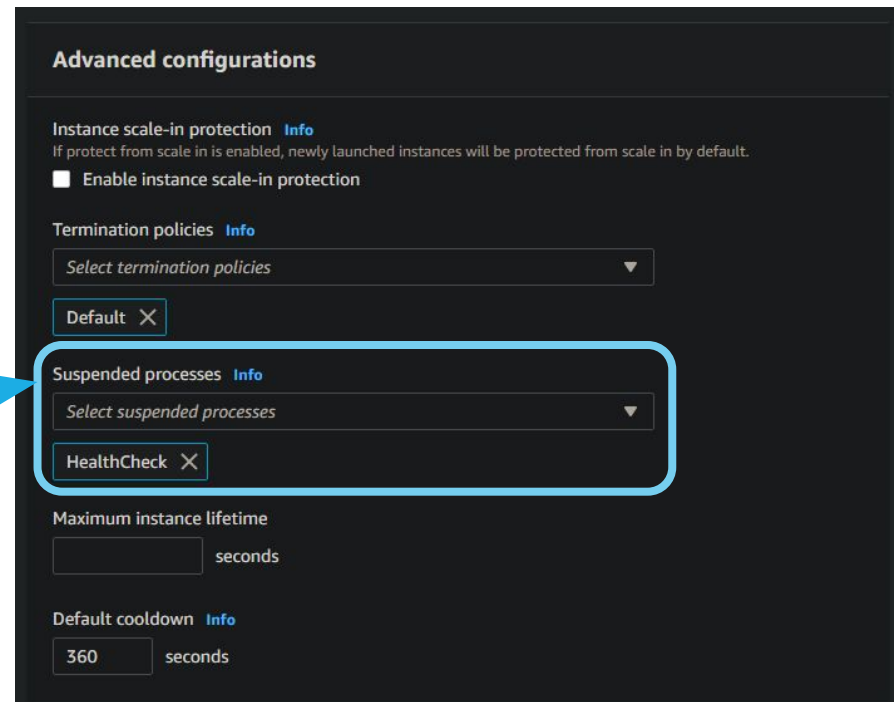
Select the group of the instance that Elastic Beanstalk has created and click “edit”.



STEP 10 - REMOVE HEALTH CHECK FROM EC2 INSTANCE

Scroll down to “Advanced configurations” and add “HealthCheck” to “Suspended processes”.

Apply changes by clicking “Update” in the bottom of the page.



Advanced configurations

Instance scale-in protection [Info](#)
If protect from scale in is enabled, newly launched instances will be protected from scale in by default.
☐ Enable instance scale-in protection

Termination policies [Info](#)
Select termination policies
Default X

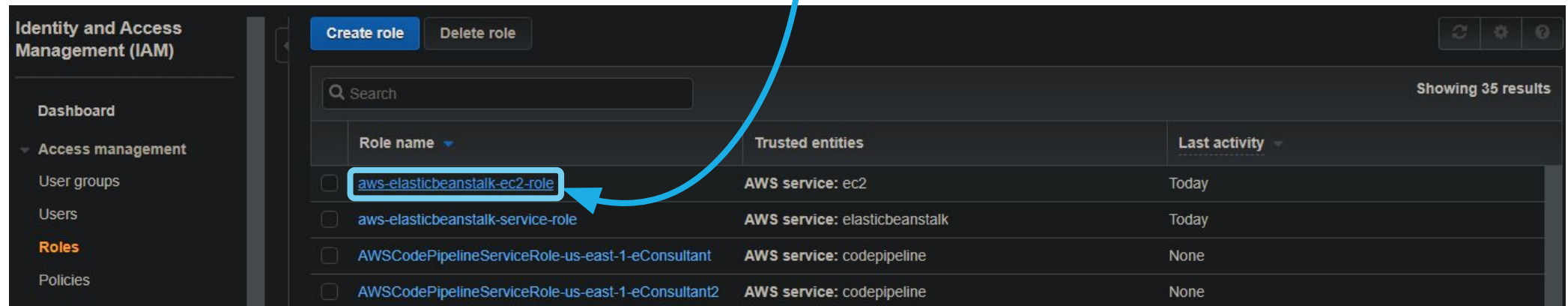
Suspended processes [Info](#)
Select suspended processes
HealthCheck X

Maximum instance lifetime
seconds

Default cooldown [Info](#)
360 seconds

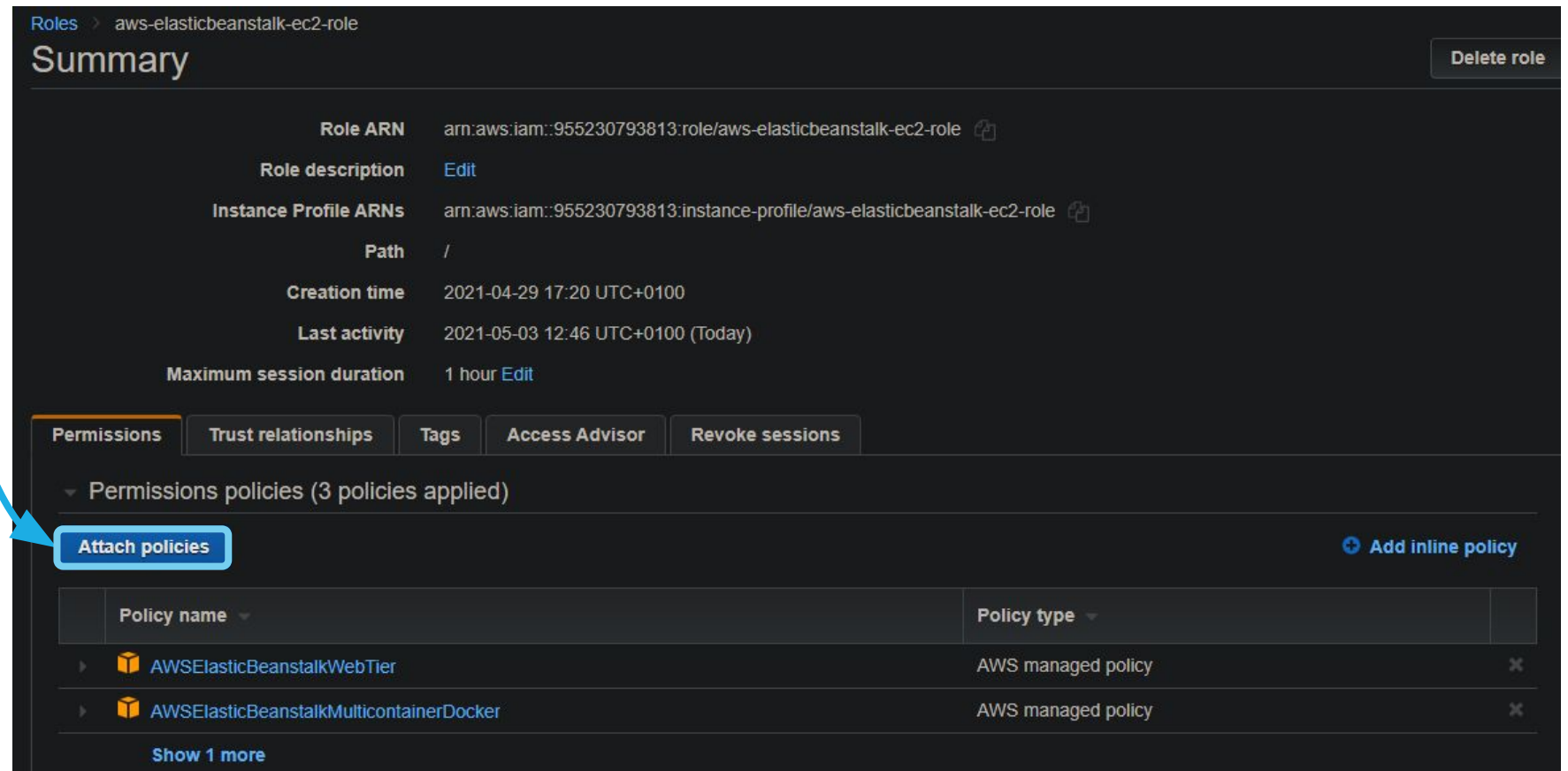
STEP 11 - MODIFY BEANSTALK ROLE FOR ACCESS TO DB

Go to IAM. Navigate to “Roles” and click on the aws-elasticbeanstalk-ec2-role



STEP 11 - MODIFY BEANSTALK ROLE FOR ACCESS TO DB

Click “Attach policies”



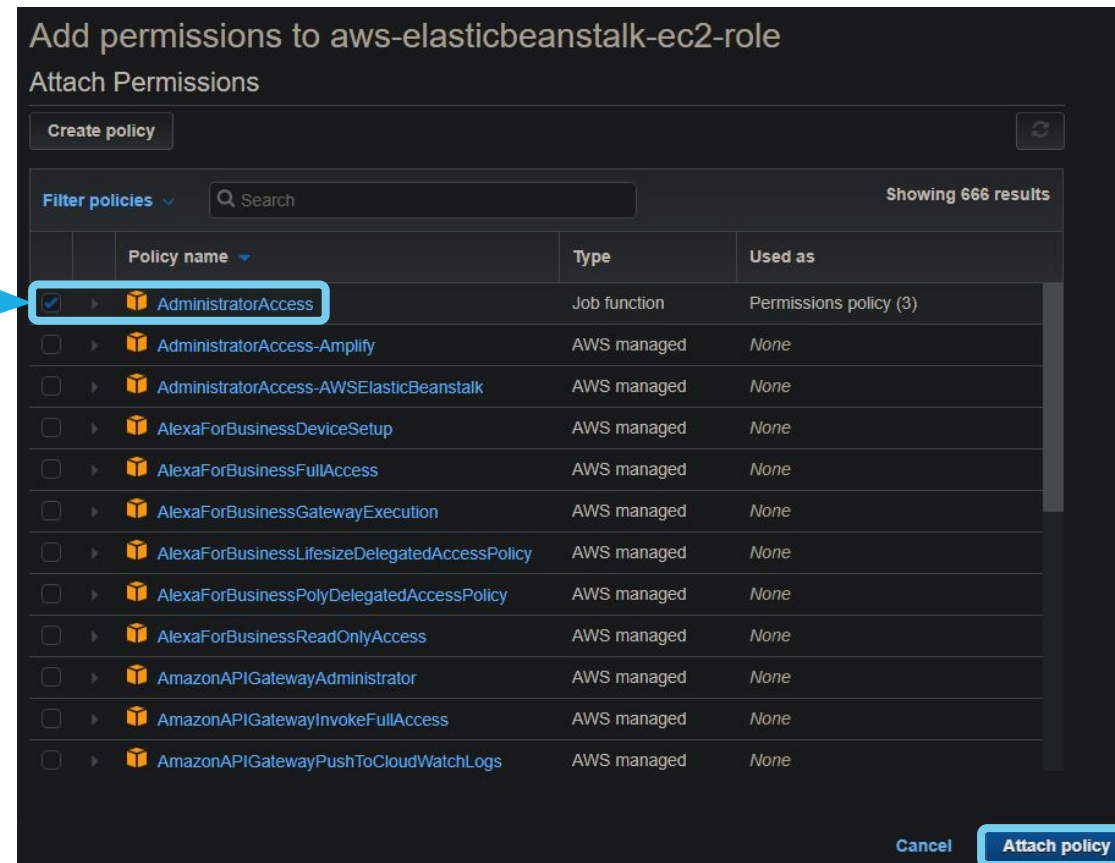
The screenshot shows the AWS IAM console interface for the role 'aws-elasticbeanstalk-ec2-role'. The 'Summary' tab is active, displaying details such as Role ARN, Role description, Instance Profile ARNs, Path, Creation time, Last activity, and Maximum session duration. Below the summary, the 'Permissions' tab is selected, showing 'Permissions policies (3 policies applied)'. A blue arrow points to the 'Attach policies' button, which is highlighted with a blue border. To the right of this button is a link to 'Add inline policy'. Below these buttons, a table lists the attached policies:

Policy name	Policy type
AWSElasticBeanstalkWebTier	AWS managed policy
AWSElasticBeanstalkMulticontainerDocker	AWS managed policy

At the bottom of the table, there is a link to 'Show 1 more'.

STEP 11 - MODIFY BEANSTALK ROLE FOR ACCESS TO DB

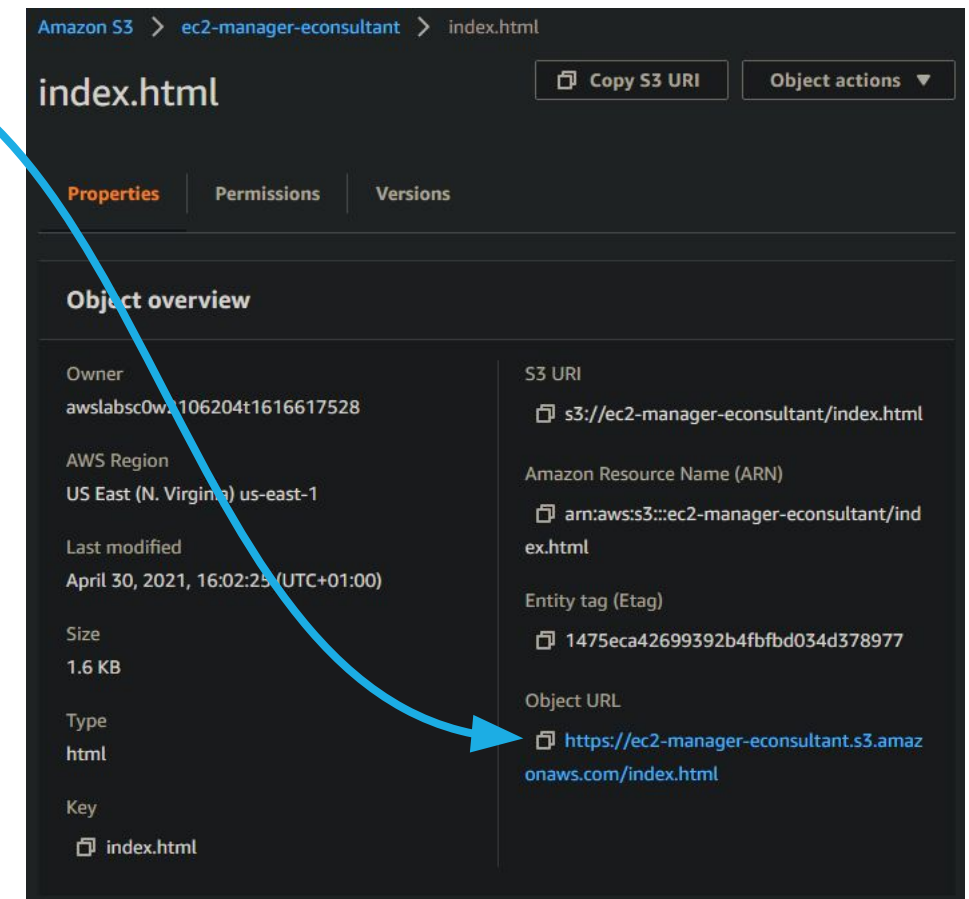
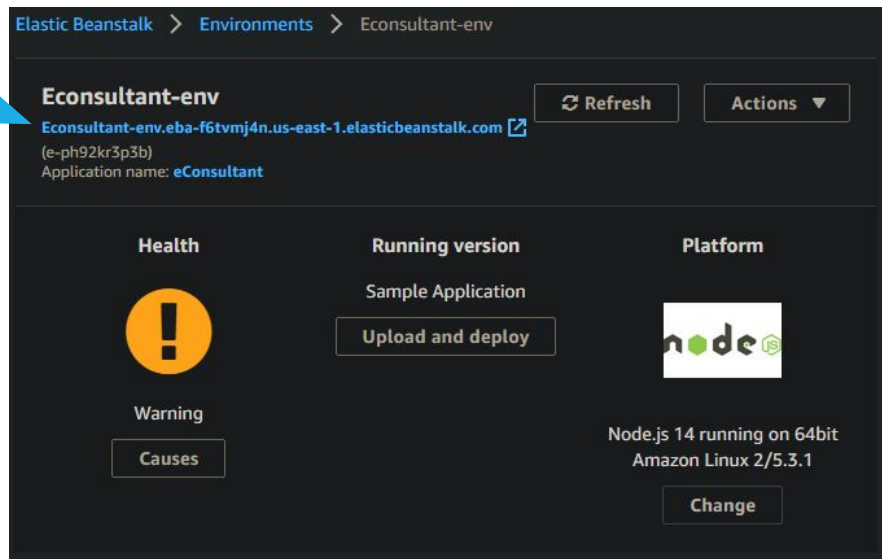
Select “AdministratorAccess” policy and click “Attach policy” button.



STEP 12 - ACCESS DEPLOYED APPLICATION

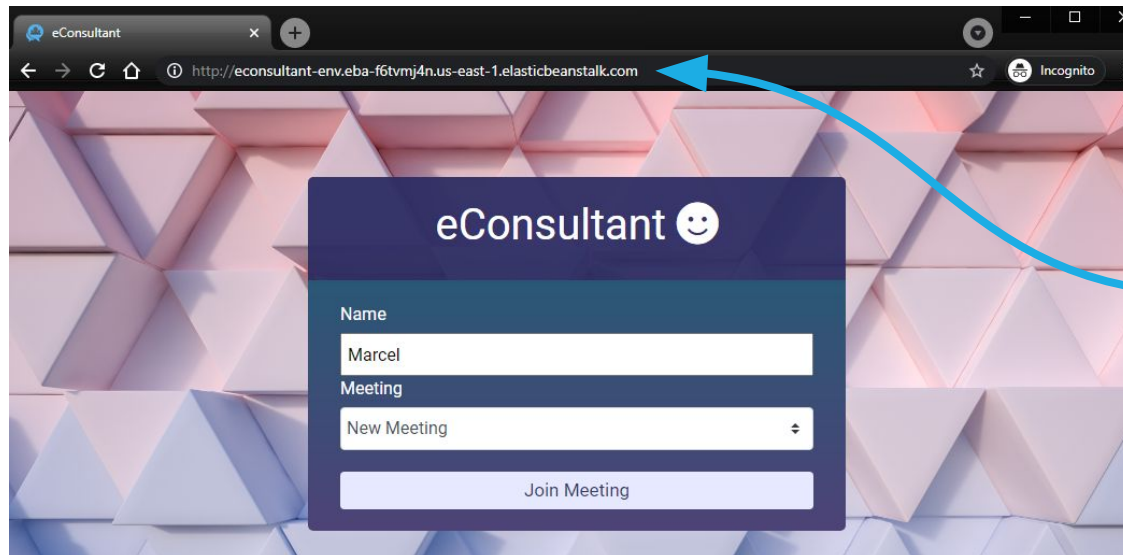
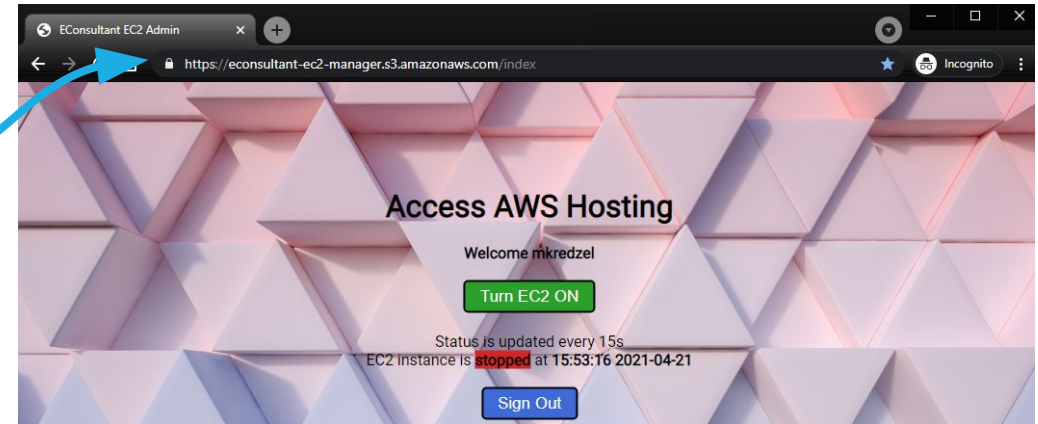
Access EC2 management static page through S3 bucket index.html Object URL link.

Access the deployed eConsultant application through EC2 instance public IPv4 address.



STEP 12 - ACCESS DEPLOYED APPLICATION

Access EC2 management static page through S3 bucket index.html Object URL link.



Access the deployed eConsultant application through Elastic Beanstalk environment address.

browser and add the eConsultant URL: `http://econsultant-env.eba-f6tvmj4n.us-east-1.elasticbeanstalk.com` to





CONTACT ME

Marcel Kredzel: marcel.kredzel@gmail.com