

Project 2

show integration between AWS Lambda and AWS
API Gateway:

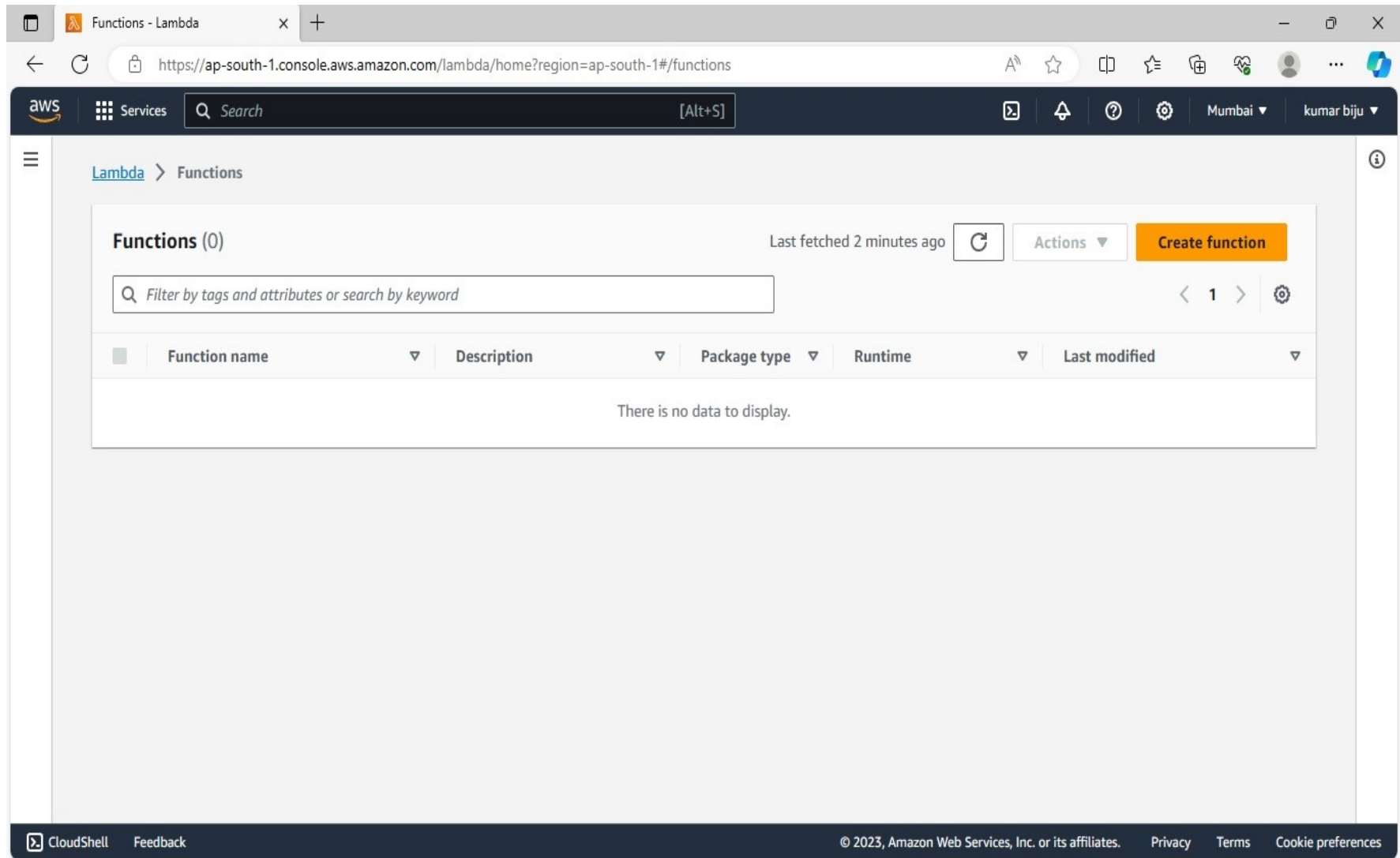
you should be calling 2 endpoints over the API
Gateway.

- 1) Respond with hello world (Python/node.js)
- 2) Respond with current date and time
(Python/node.js)

Architecture Diagram of AWS Lambda + AWS API



1. Create a Lambda Function



Go to search page and click lambda function and create a function and give a name project1

The screenshot displays the AWS Lambda 'Create function' page in the 'ap-south-1' region. The page is titled 'Create function' and includes a breadcrumb trail: 'Lambda > Functions > Create function'. A note states: 'AWS Serverless Application Repository applications have moved to [Create application](#).' Three creation methods are presented: 'Author from scratch' (selected with a radio button), 'Use a blueprint', and 'Container image'. The 'Basic information' section contains three fields: 'Function name' with the value 'project1', 'Runtime' set to 'Python 3.11', and 'Architecture' set to 'x86_64'. The bottom of the image shows the Windows taskbar with the search bar, application icons, and system tray information including the date and time (10/23/2023, 3:10 PM).

Create function - Lambda

https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/create/function

aws Services Search [Alt+S]

Mumbai kumar biju

Lambda > Functions > Create function

Create function [Info](#)

AWS Serverless Application Repository applications have moved to [Create application](#).

☒ Author from scratch
Start with a simple Hello World example.

☐ Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.

☐ Container image
Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.
project1
Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime [Info](#)
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.11

Architecture [Info](#)
Choose the instruction set architecture you want for your function code.
☒ x86_64

CloudShell Feedback

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

78%

27°C Mostly cloudy

ENG IN 3:10 PM 10/23/2023

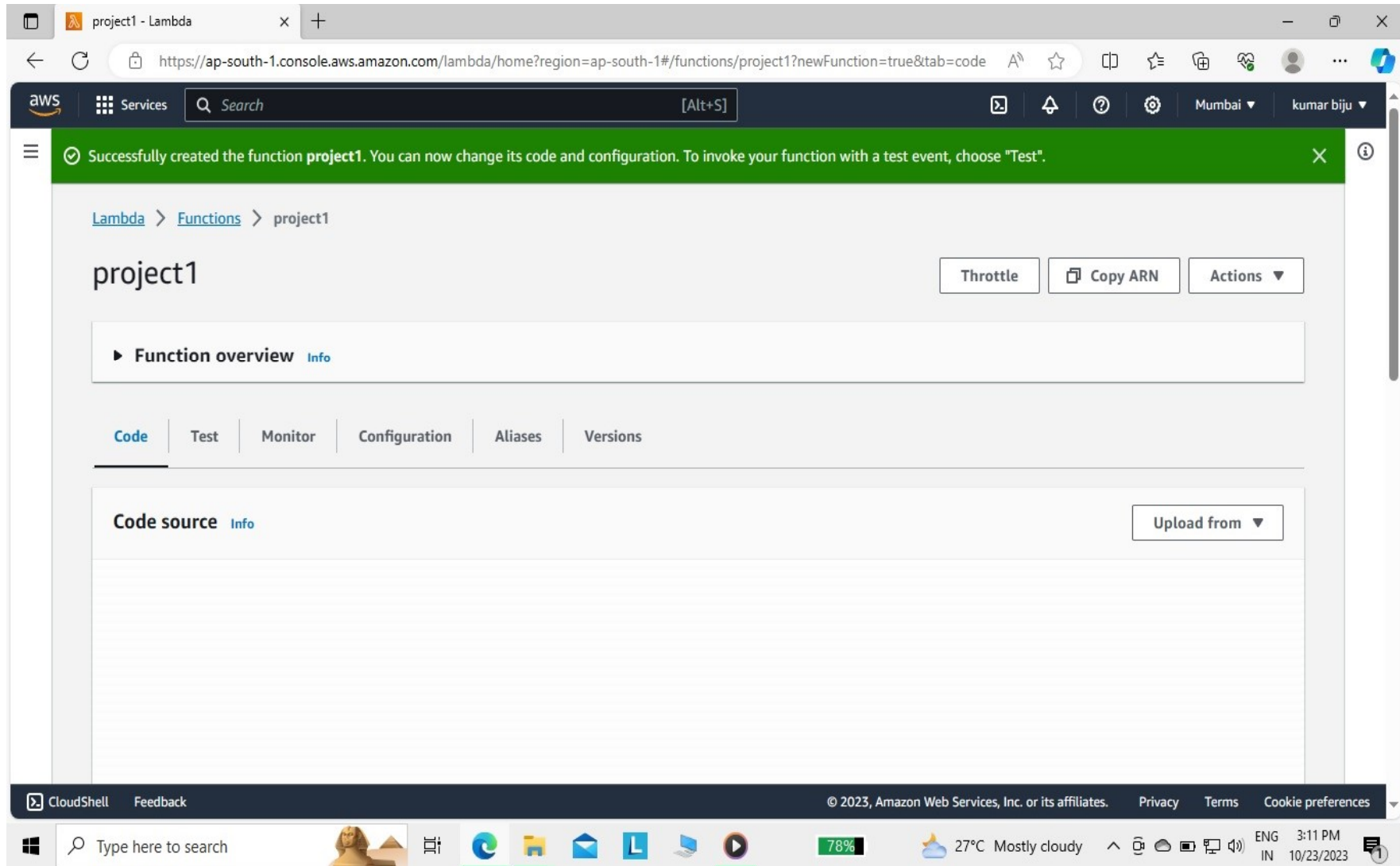
Now set configuration runtime and architecture

The screenshot displays the AWS Lambda 'Create function' console page. The browser address bar shows the URL: <https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/create/function>. The page features the AWS logo and a search bar at the top. The main content area is divided into sections for configuring the function:

- Function name:** A text input field containing 'project1'. Below it, a note states: 'Use only letters, numbers, hyphens, or underscores with no spaces.'
- Runtime:** A dropdown menu showing 'Python 3.11'. A note above it says: 'Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.'
- Architecture:** Two radio button options: 'x86_64' (selected) and 'arm64'. A note above it says: 'Choose the instruction set architecture you want for your function code.'
- Permissions:** A section with a note: 'By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.'
- Buttons:** At the bottom right, there are 'Cancel' and 'Create function' buttons.

The bottom of the image shows the Windows taskbar with the search bar, task icons, and system tray information including the date (10/23/2023) and time (3:11 PM).

Now successfully create lambda function



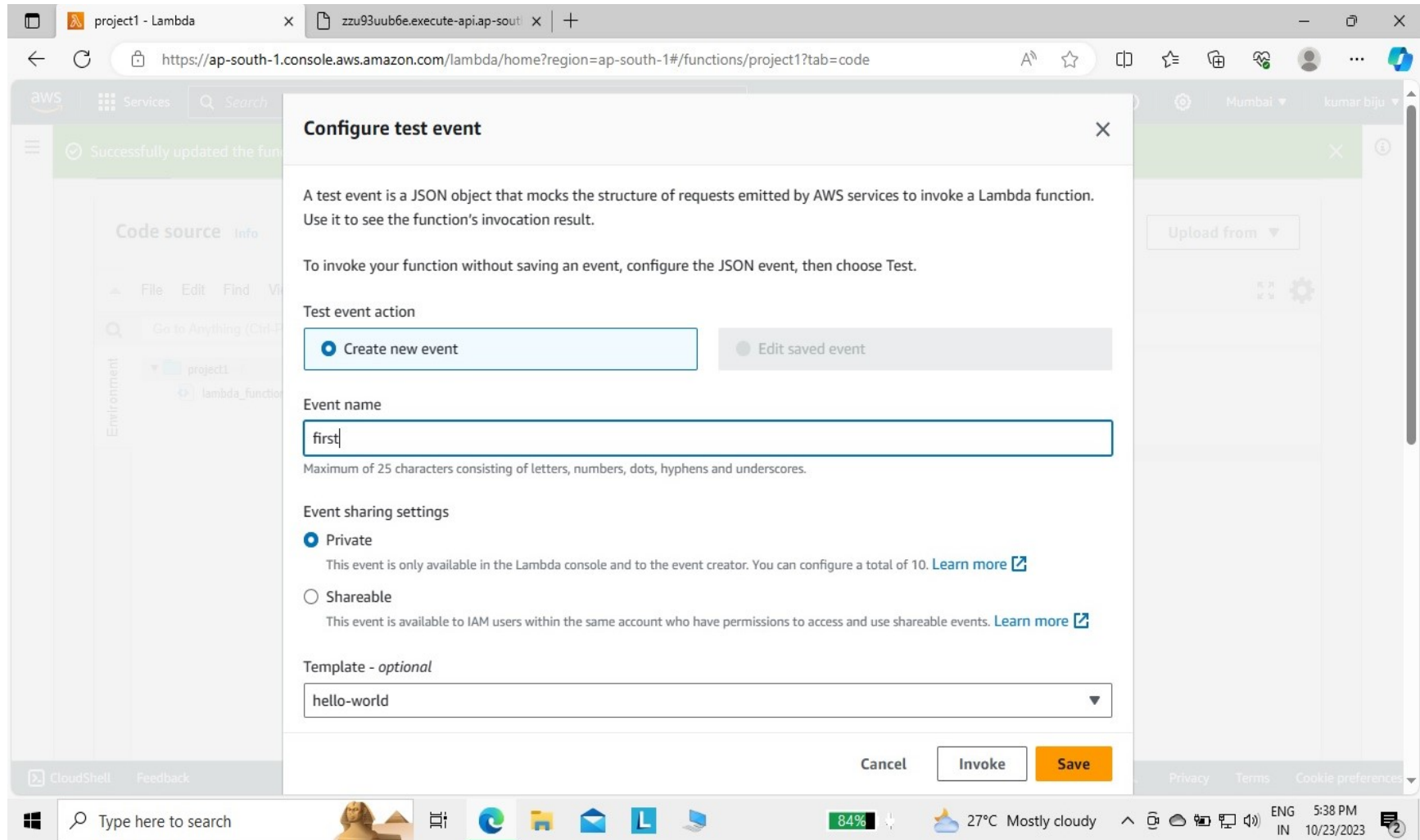
Now check the lambda function and write the python code

The screenshot displays the AWS Lambda console interface in a web browser. At the top, a green notification bar states: "Successfully created the function project1. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'." Below this, the "Function overview" section is visible, followed by tabs for "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions". The "Code" tab is active, showing the "Code source" section with an "Upload from" dropdown. A menu bar with "File", "Edit", "Find", "View", "Go", "Tools", and "Window" is present, along with "Test" and "Deploy" buttons. The left sidebar shows the "Environment" section with a tree view containing "project1" and "lambda_function.py". The main editor area shows the following Python code:

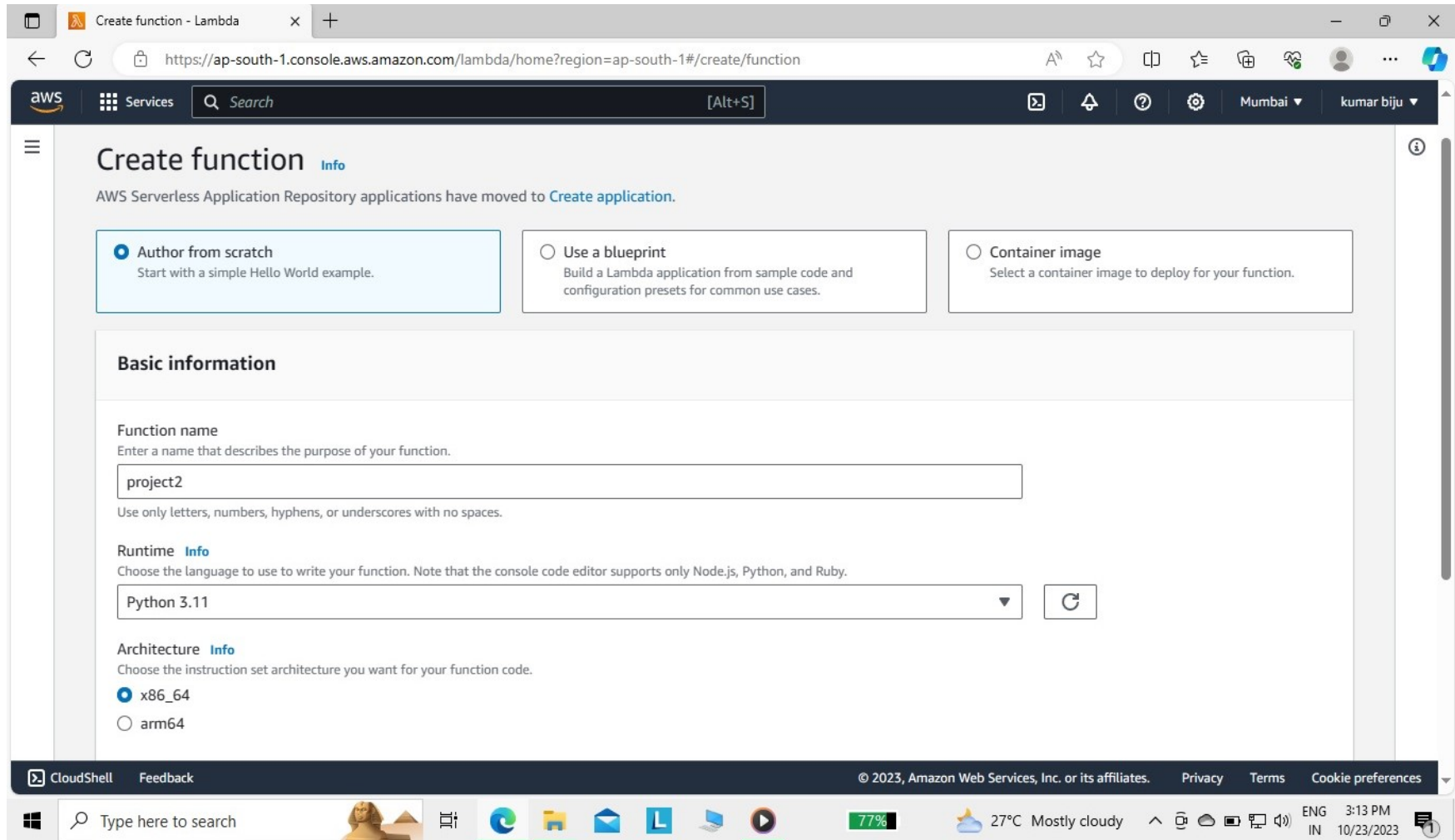
```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO implement
5     return {
6         'statusCode': 200,
7         'body': json.dumps('Hello from Lambda!')
8     }
9
```

The bottom of the screen shows the Windows taskbar with the search bar, task icons, system tray (77% battery, 27°C, Mostly cloudy), and date/time (3:12 PM, 10/23/2023).

Click the deploy button and click the test then give a name test event



Click the again create function and give a name project 2



Set the configuration and runtime

Create function - Lambda

https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/create/function

Function name

Enter a name that describes the purpose of your function.

project2

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime [Info](#)

Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Python 3.11

Architecture [Info](#)

Choose the instruction set architecture you want for your function code.

☒ x86_64

☐ arm64

Permissions [Info](#)

By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

► Change default execution role

► Advanced settings

Cancel Create function

CloudShell Feedback

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

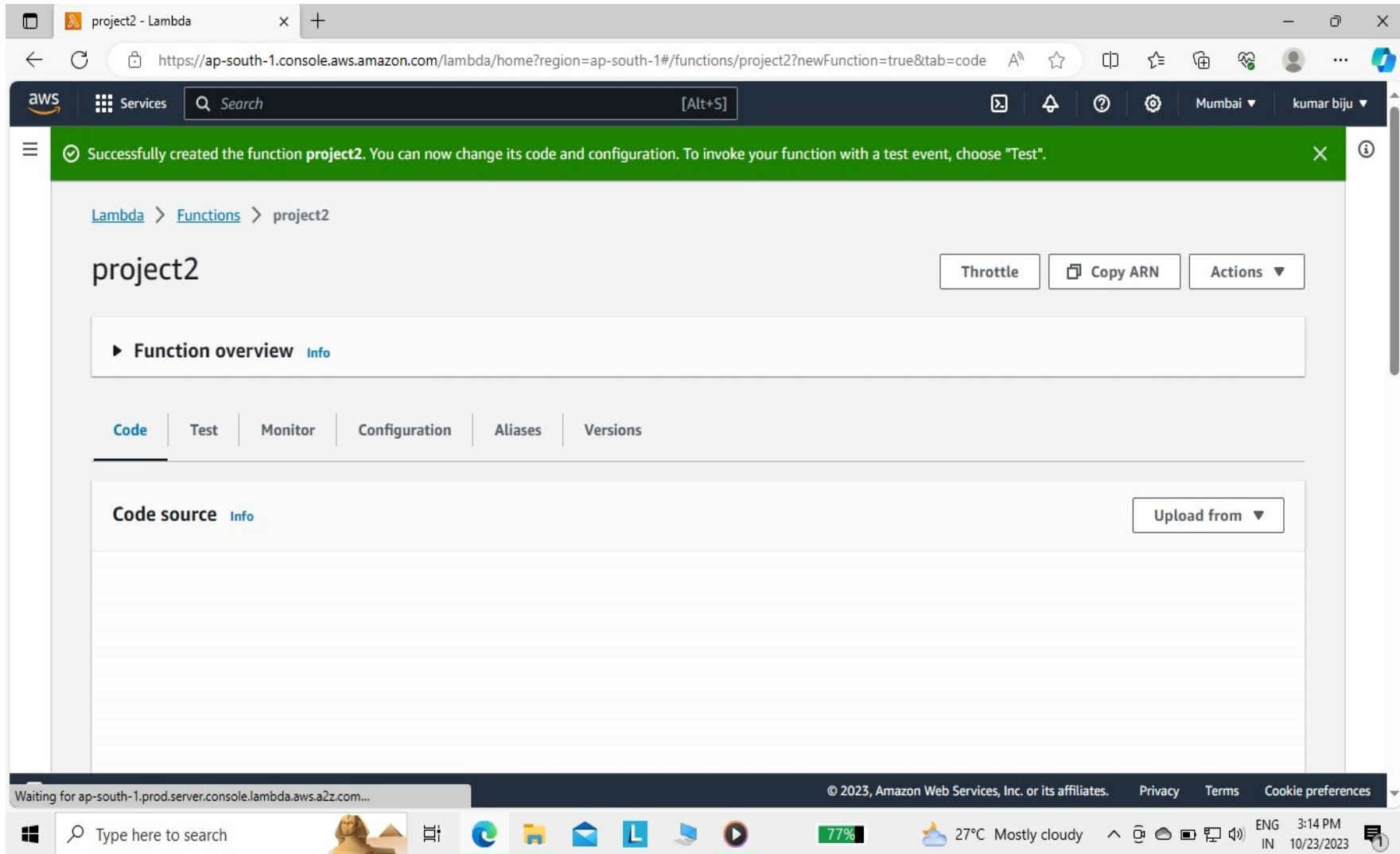
Type here to search

77%

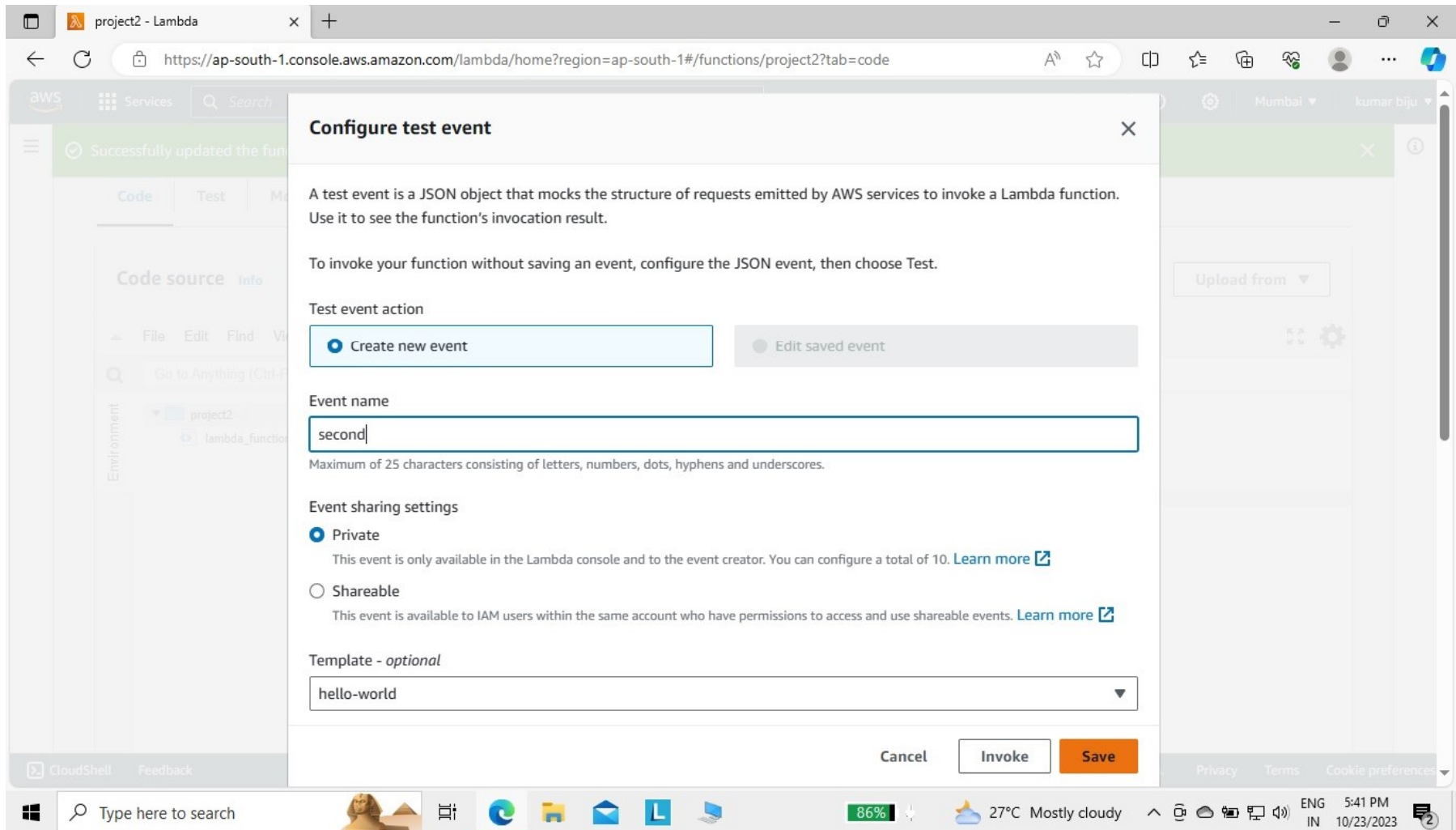
27°C Mostly cloudy

ENG IN 3:13 PM 10/23/2023

Successfully create lambda function



Now write code and deploy then click test event and give name second



2. Create API

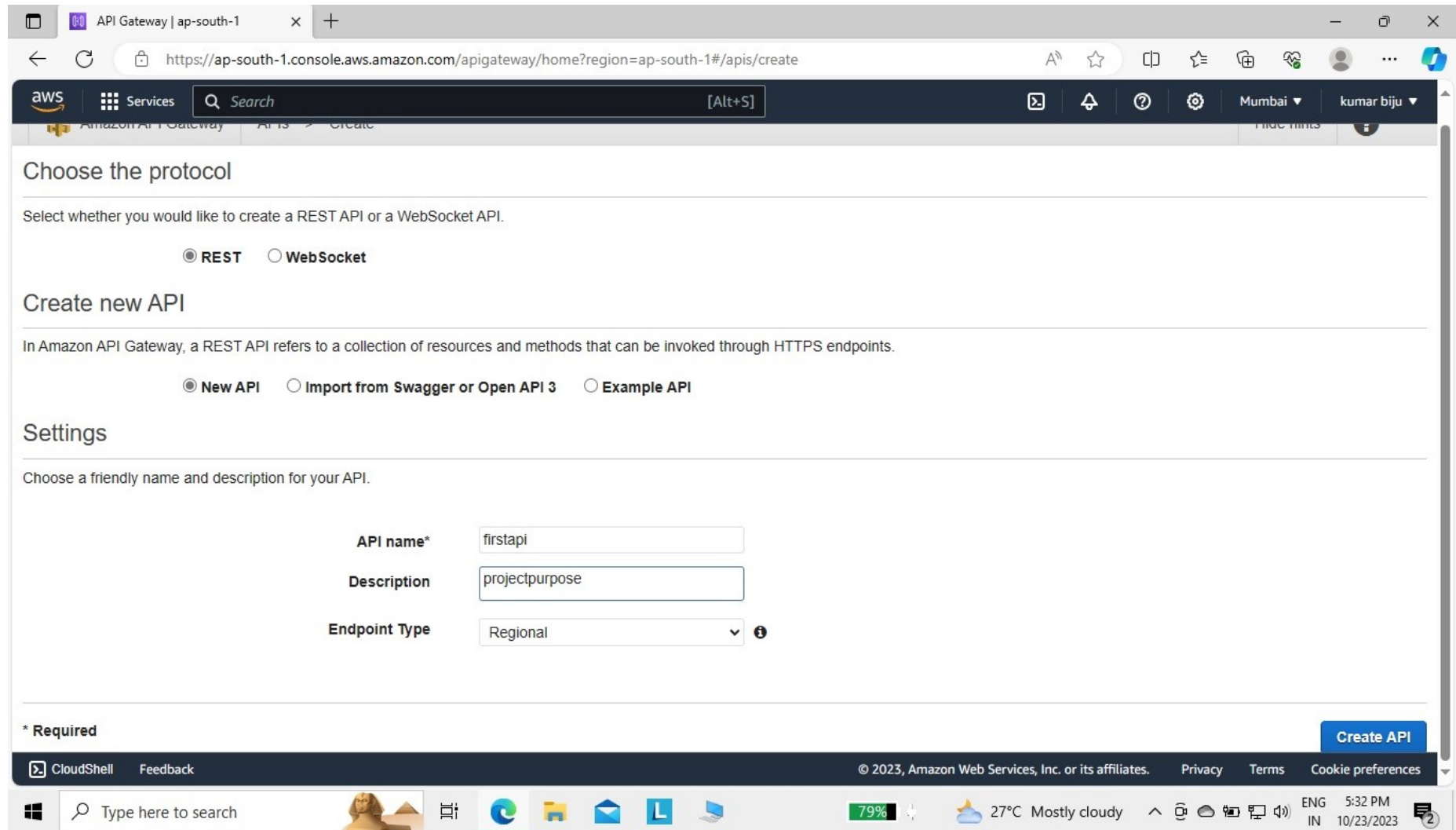
The screenshot displays the AWS API Gateway console interface. The top navigation bar includes the AWS logo, a 'Services' menu, a search bar, and a user profile 'kumar biju'. The left sidebar shows the 'API Gateway' section with links to 'APIs', 'Custom domain names', and 'VPC links'. A notification states: 'Use the new console. As of 30th October 2023 the old console will no longer be available.'

The main content area features three API creation options:

- WebSocket API**: Build a WebSocket API using persistent connections for real-time use cases such as chat applications or dashboards. Works with the following: Lambda, HTTP, AWS Services. A 'Build' button is present.
- REST API**: Develop a REST API where you gain complete control over the request and response along with API management capabilities. Works with the following: Lambda, HTTP, AWS Services. 'Import' and 'Build' buttons are present.
- REST API Private**: (Partially visible at the bottom)

The bottom of the console shows a footer with 'CloudShell', 'Feedback', and copyright information: '© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences'. The Windows taskbar at the very bottom shows the search bar, task view, and system tray with the date '10/23/2023' and time '5:30 PM'.

Go to search page click API and click the button
REST API. Give a name API projectpurpose



API Gateway | ap-south-1

https://ap-south-1.console.aws.amazon.com/apigateway/home?region=ap-south-1#/apis/create

aws Services Search [Alt+S]

Choose the protocol

Select whether you would like to create a REST API or a WebSocket API.

☒ REST ☐ WebSocket

Create new API

In Amazon API Gateway, a REST API refers to a collection of resources and methods that can be invoked through HTTPS endpoints.

☒ New API ☐ Import from Swagger or Open API 3 ☐ Example API

Settings

Choose a friendly name and description for your API.

API name* firstapi

Description projectpurpose

Endpoint Type Regional

* Required

Create API

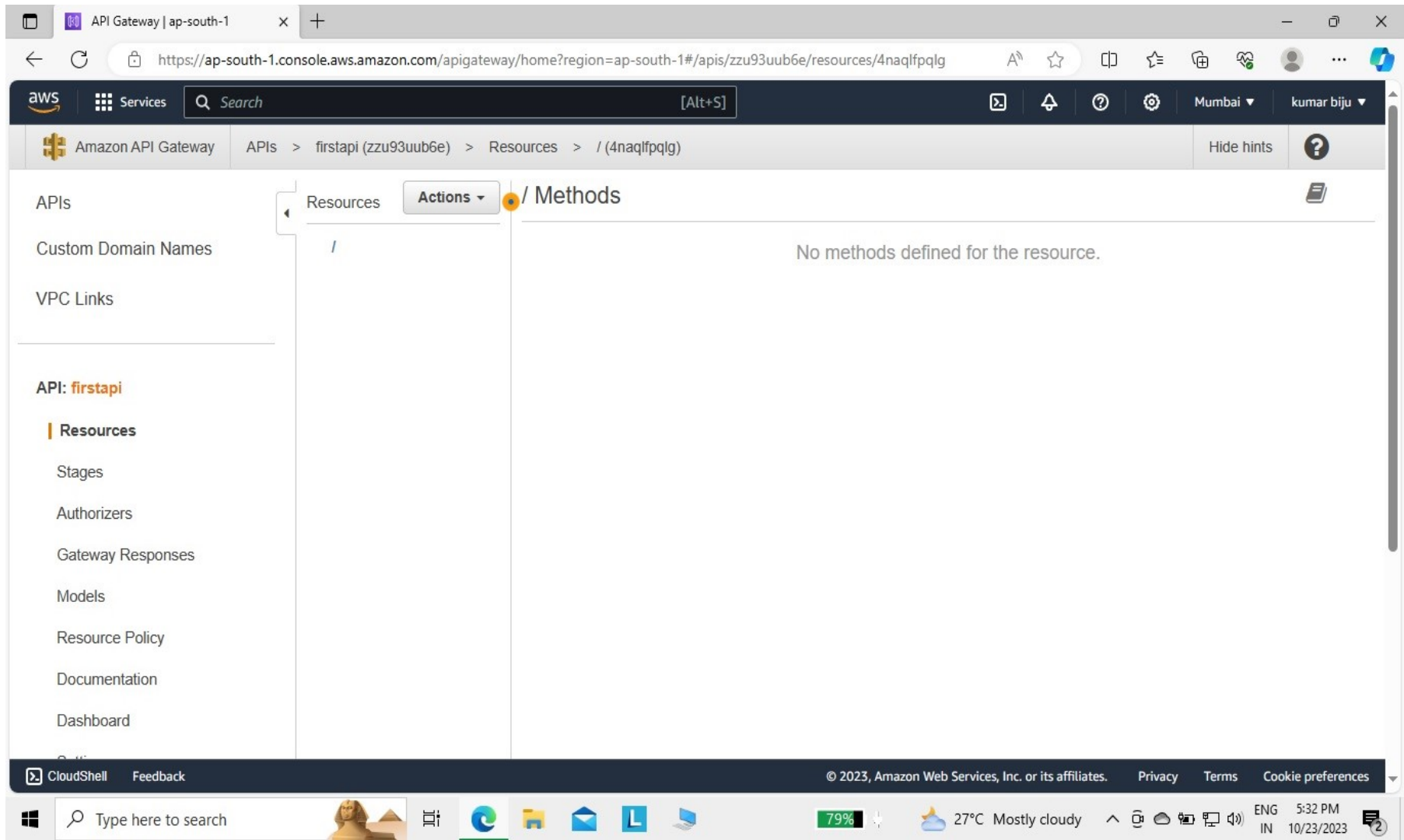
CloudShell Feedback

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

79% 27°C Mostly cloudy ENG IN 5:32 PM 10/23/2023

Now click the action button and set configuration



The screenshot displays the AWS API Gateway console interface. The browser address bar shows the URL: `https://ap-south-1.console.aws.amazon.com/apigateway/home?region=ap-south-1#/apis/zzu93uub6e/resources/4naqlfpqlg`. The console header includes the AWS logo, a search bar, and navigation links for Services, Mumbai, and kumar biju. The breadcrumb trail indicates the path: Amazon API Gateway > APIs > firstapi (zzu93uub6e) > Resources > / (4naqlfpqlg). The left sidebar contains a navigation menu with options: APIs, Custom Domain Names, VPC Links, API: firstapi (selected), Resources (selected), Stages, Authorizers, Gateway Responses, Models, Resource Policy, Documentation, and Dashboard. The main content area is divided into three tabs: Resources, Actions, and Methods. The 'Methods' tab is active, displaying the message: 'No methods defined for the resource.' The footer of the console shows 'CloudShell', 'Feedback', and copyright information for Amazon Web Services, Inc. or its affiliates. The Windows taskbar at the bottom includes a search bar, task icons, system tray (79% battery, 27°C Mostly cloudy), and system clock (5:32 PM, 10/23/2023).

Setup the GET and click the save button

The screenshot displays the AWS API Gateway console interface. The breadcrumb navigation at the top indicates the path: **APIs** > **firstapi (zzu93uub6e)** > **Resources** > **/ (4naqlfpqlg)** > **GET**. The left-hand navigation pane lists various API Gateway components, with **Resources** currently selected. The main content area is titled **/ - GET - Setup** and contains the instruction: "Choose the integration point for your new method." Under the **Integration type** section, the **Lambda Function** option is selected with a radio button. Other options include HTTP, Mock, AWS Service, and VPC Link. Below this, the **Use Lambda Proxy integration** checkbox is checked. The **Lambda Region** is set to **ap-south-1** via a dropdown menu. The **Lambda Function** text input field contains the value **project1**. The **Use Default Timeout** checkbox is also checked. A blue **Save** button is positioned in the bottom right corner of the setup area. The footer of the console shows the copyright notice "© 2023, Amazon Web Services, Inc. or its affiliates." along with links for Privacy, Terms, and Cookie preferences. The Windows taskbar at the very bottom shows the system clock as 5:33 PM on 10/23/2023.

Click the deploy button and give a stage name firststage

The screenshot shows the AWS API Gateway console in the 'ap-south-1' region. A modal dialog titled 'Deploy API' is open, prompting the user to choose a stage for deployment. The dialog includes the following fields:

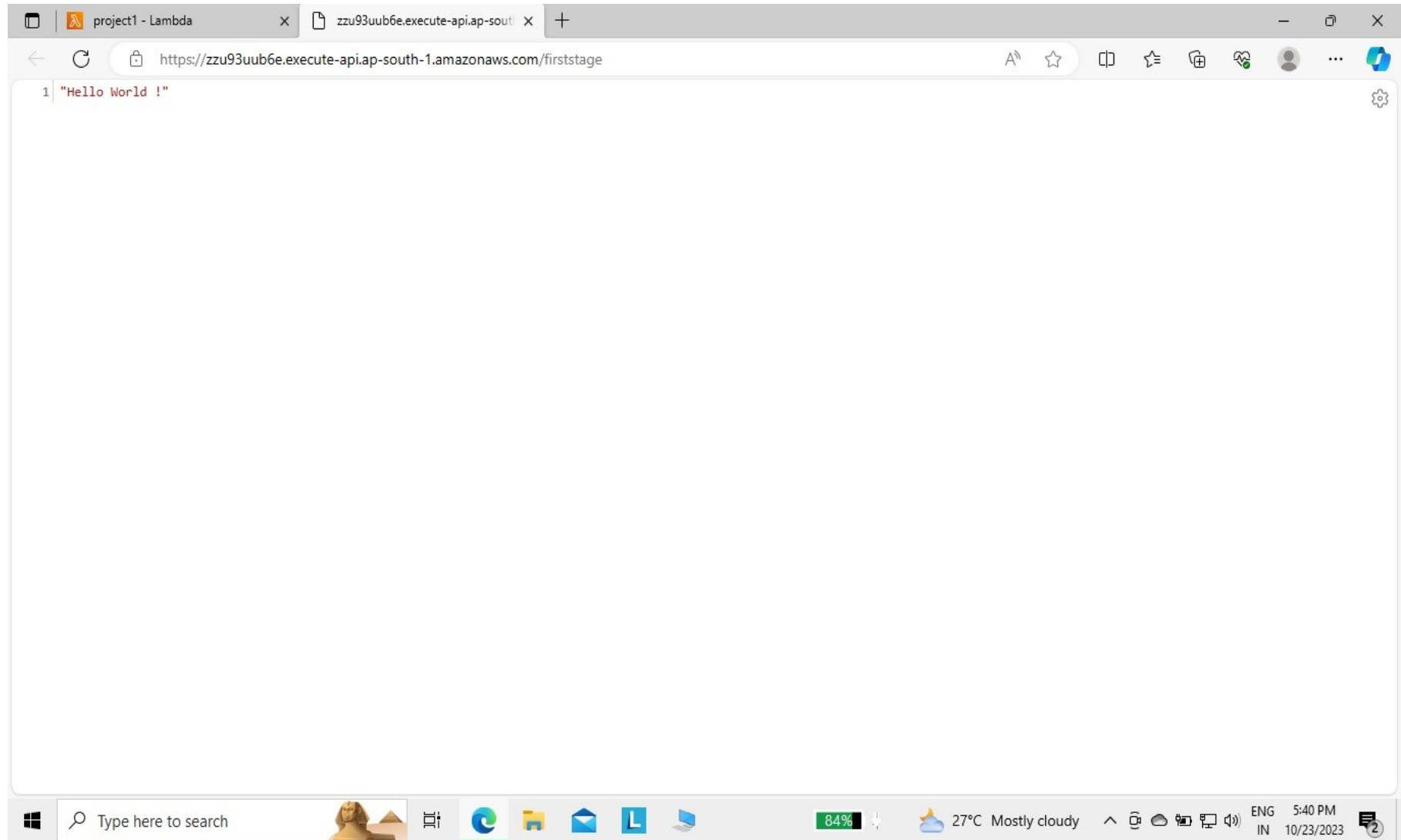
- Deployment stage:** A dropdown menu currently showing '[New Stage]'.
- Stage name*:** A text input field containing 'firststage'.
- Stage description:** An empty text input field.
- Deployment description:** An empty text input field.

At the bottom of the modal, there are two buttons: 'Cancel' and 'Deploy'. The background of the console shows the 'firstapi' resource with a 'GET' method, and the 'Integration Request' section is visible on the right.

Successfully create deploy stage now click the URL button to new tab

The screenshot shows the AWS API Gateway console for the 'firststage' Stage Editor. The breadcrumb navigation is 'APIs > firstapi (zzu93uub6e) > Stages > firststage'. A notification banner at the top states: 'The new API Gateway console experience is now available. We've redesigned the API Gateway console for REST APIs and WebSocket APIs. Try out the new console. As of 30th October 2023 the old console will no longer be available.' The 'firststage' Stage Editor shows the 'Invoke URL' as 'https://zzu93uub6e.execute-api.ap-south-1.amazonaws.com/firststage'. Below this are tabs for 'Settings', 'Logs/Tracing', 'Stage Variables', 'SDK Generation', 'Export', and 'Deployment History'. The 'Canary' tab is selected, showing instructions: 'Manage Canary settings here. A Canary is used to test new API deployments and/or changes to stage variables. A Canary can receive a percentage of requests going to your stage. In addition, API deployments will be made to the Canary first before being able to be promoted to the entire stage.' A 'Create Canary' button is located at the bottom right of the Canary tab content. The footer of the console shows '© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences'.

Website is successfully is running



Click the new API and give a name project2

The screenshot shows the AWS API Gateway console in the 'ap-south-1' region. The page title is 'API Gateway | ap-south-1'. The URL is 'https://ap-south-1.console.aws.amazon.com/apigateway/home?region=ap-south-1#/apis/create'. The page has a dark blue header with the AWS logo, 'Services' link, a search bar, and user information 'Mumbai' and 'kumar biju'. The main content area is titled 'Choose the protocol' and 'Create new API'. It includes instructions to select between REST and WebSocket protocols, and to choose between creating a new API, cloning an existing one, or importing from Swagger/OpenAPI. The 'Settings' section contains form fields for 'API name*' (filled with 'secondapi'), 'Description' (filled with 'projectbase'), and 'Endpoint Type' (set to 'Regional'). A 'Create API' button is at the bottom right. The footer includes 'CloudShell', 'Feedback', copyright information, and links to 'Privacy', 'Terms', and 'Cookie preferences'. The Windows taskbar at the bottom shows the search bar, task icons, system tray with battery at 88%, temperature at 27°C, and date/time as 5:45 PM on 10/23/2023.

API Gateway | ap-south-1

https://ap-south-1.console.aws.amazon.com/apigateway/home?region=ap-south-1#/apis/create

aws Services Search [Alt+S]

Mumbai kumar biju

Choose the protocol

Select whether you would like to create a REST API or a WebSocket API.

☒ REST ☐ WebSocket

Create new API

In Amazon API Gateway, a REST API refers to a collection of resources and methods that can be invoked through HTTPS endpoints.

☒ New API ☐ Clone from existing API ☐ Import from Swagger or Open API 3 ☐ Example API

Settings

Choose a friendly name and description for your API.

API name* secondapi

Description projectbase

Endpoint Type Regional

* Required

Create API

CloudShell Feedback

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

88% 27°C Mostly cloudy ENG IN 5:45 PM 10/23/2023

Setup the configuration and save the button

The screenshot displays the AWS API Gateway console in the 'ap-south-1' region. The left sidebar shows the navigation menu with 'API: secondapi' selected. The main content area is titled '/ - GET - Setup' and contains the following configuration options:

- Integration type:** Radio buttons for Lambda Function (selected), HTTP, Mock, AWS Service, and VPC Link.
- Use Lambda Proxy integration:** Checked checkbox.
- Lambda Region:** Dropdown menu set to 'ap-south-1'.
- Lambda Function:** Text input field containing 'project2'.
- Use Default Timeout:** Checked checkbox.

A blue 'Save' button is located at the bottom right of the configuration area. The bottom of the screen shows the Windows taskbar with the search bar, taskbar icons, and system tray information including the date and time (10/23/2023, 5:46 PM).

Click the deploy button the REST API

The screenshot shows the AWS API Gateway console for the 'secondapi' API. The left sidebar contains navigation links: Custom Domain Names, VPC Links, Resources, Stages, Authorizers, Gateway Responses, Models, Resource Policy, Documentation, Dashboard, Settings, Usage Plans, and API Keys. The main content area is titled '/ - GET - Method Execution'. It shows a flow diagram with four boxes: Method Request, Integration Request, Method Response, and Integration Response. The Method Request box contains 'Auth: NONE' and 'ARN: am:aws:execute-api:ap-south-1:133935829274:igmhickmzh/*/*GET'. The Integration Request box contains 'Type: LAMBDA_PROXY'. The Method Response box contains 'HTTP Status: Proxy' and 'Models: application/json => Empty'. The Integration Response box contains 'Proxy integrations cannot be configured to transform responses.' A 'TEST' button is located to the left of the Method Request box. At the bottom of the console, there is a footer with '© 2023, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

API: **secondapi**

- Resources
- Stages
- Authorizers
- Gateway Responses
- Models
- Resource Policy
- Documentation
- Dashboard
- Settings
- Usage Plans
- API Keys

Resources Actions **/ - GET - Method Execution**

GET

Method Request

Auth: NONE
ARN: am:aws:execute-api:ap-south-1:133935829274:igmhickmzh/*/*GET

Integration Request

Type: LAMBDA_PROXY

Method Response

HTTP Status: Proxy
Models: application/json => Empty

Integration Response

Proxy integrations cannot be configured to transform responses.

Client

Lambda project2

TEST

CloudShell Feedback

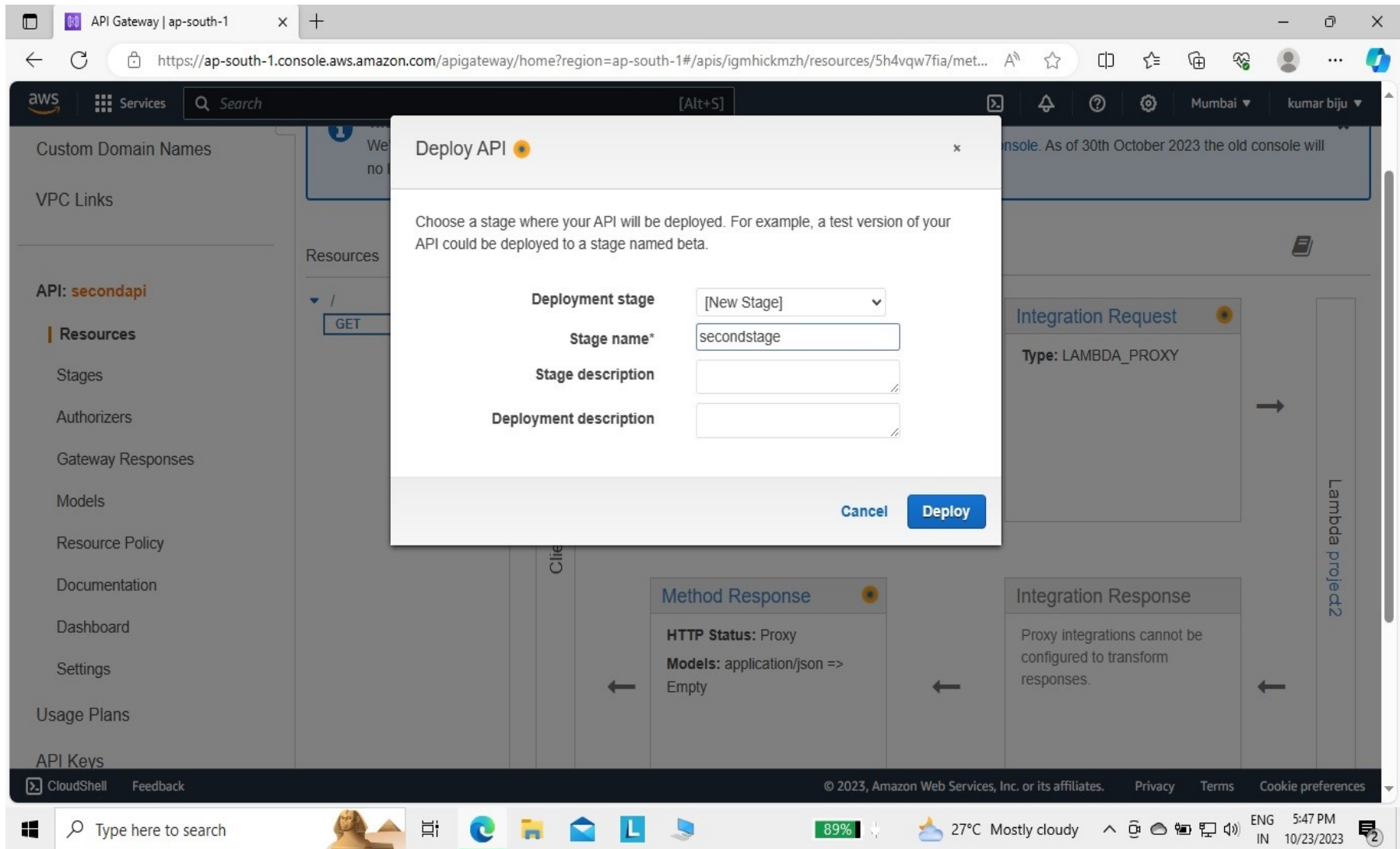
© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

89% 27°C Mostly cloudy

ENG IN 5:47 PM 10/23/2023

Give a stage name secondstage



The screenshot displays the AWS API Gateway console interface. A modal dialog titled "Deploy API" is open, prompting the user to choose a stage for deployment. The dialog contains the following fields:

- Deployment stage:** A dropdown menu currently showing "[New Stage]".
- Stage name*:** A text input field containing the value "secondstage".
- Stage description:** An empty text input field.
- Deployment description:** An empty text input field.

At the bottom of the dialog are "Cancel" and "Deploy" buttons. The background shows the API Gateway console with a sidebar menu on the left containing options like "Custom Domain Names", "VPC Links", "API: secondapi", "Resources", "Stages", "Authorizers", "Gateway Responses", "Models", "Resource Policy", "Documentation", "Dashboard", "Settings", "Usage Plans", and "API Keys". The main content area displays the API configuration, including an "Integration Request" section with "Type: LAMBDA_PROXY" and a "Method Response" section with "HTTP Status: Proxy" and "Models: application/json => Empty".

Successfully create deploy stage and click the URL button and open new tab

The screenshot displays the AWS API Gateway console interface. The browser address bar shows the URL: `https://ap-south-1.console.aws.amazon.com/apigateway/home?region=ap-south-1#/apis/igmhickmzh/stages/secondstage`. The console header includes the AWS logo, a search bar, and navigation links for Services, API Gateway, APIs, Stages, and secondstage. A notification banner at the top states: "The new API Gateway console experience is now available. We've redesigned the API Gateway console for REST APIs and WebSocket APIs. Try out the new console. As of 30th October 2023 the old console will no longer be available." The left sidebar lists navigation options: APIs, Custom Domain Names, VPC Links, API: **secondapi**, Resources, Stages, Authorizers, Gateway Responses, Models, Resource Policy, Documentation, and Dashboard. The main content area is titled "secondstage Stage Editor" and features a "Create" button. Below the title, there is a "secondstage" link. A prominent blue box displays the "Invoke URL: `https://igmhickmzh.execute-api.ap-south-1.amazonaws.com/secondstage`". The editor includes tabs for Settings, Logs/Tracing, Stage Variables, SDK Generation, Export, and Deployment History. The "Settings" tab is active, showing "Cache Settings" with an "Enable API cache" checkbox and "Default Method Throttling" information, which states: "Choose the default throttling level for the methods in this stage. Each method in this stage will respect these rate and burst settings. Your current account level throttling rate is 10000 requests per second with a burst of 5000 requests. Read more about API Gateway throttling." The footer of the console shows "CloudShell", "Feedback", and copyright information for Amazon Web Services, Inc. The Windows taskbar at the bottom indicates the system time as 5:48 PM on 10/23/2023, with a weather forecast of 27°C Mostly cloudy.

Successfully website is running

