DOTNET CORE-AZURE MINI PROJECT

Create a **Web API Project** to store Product Information. Use Entity Framework to store the product information in the database. The user should be able to perform all the CRUD Operations. Configure **GET, POST, PUT and DELETE**.

The Product Entity should have the following properties:

- ProductID
- ProductName
- Price
- Brand
- ManufactureDate
- ExpirationDate

Use Data Annotations to

- Mark the Primary Key
- Make ProductName Mandatory
- Make Price a Number

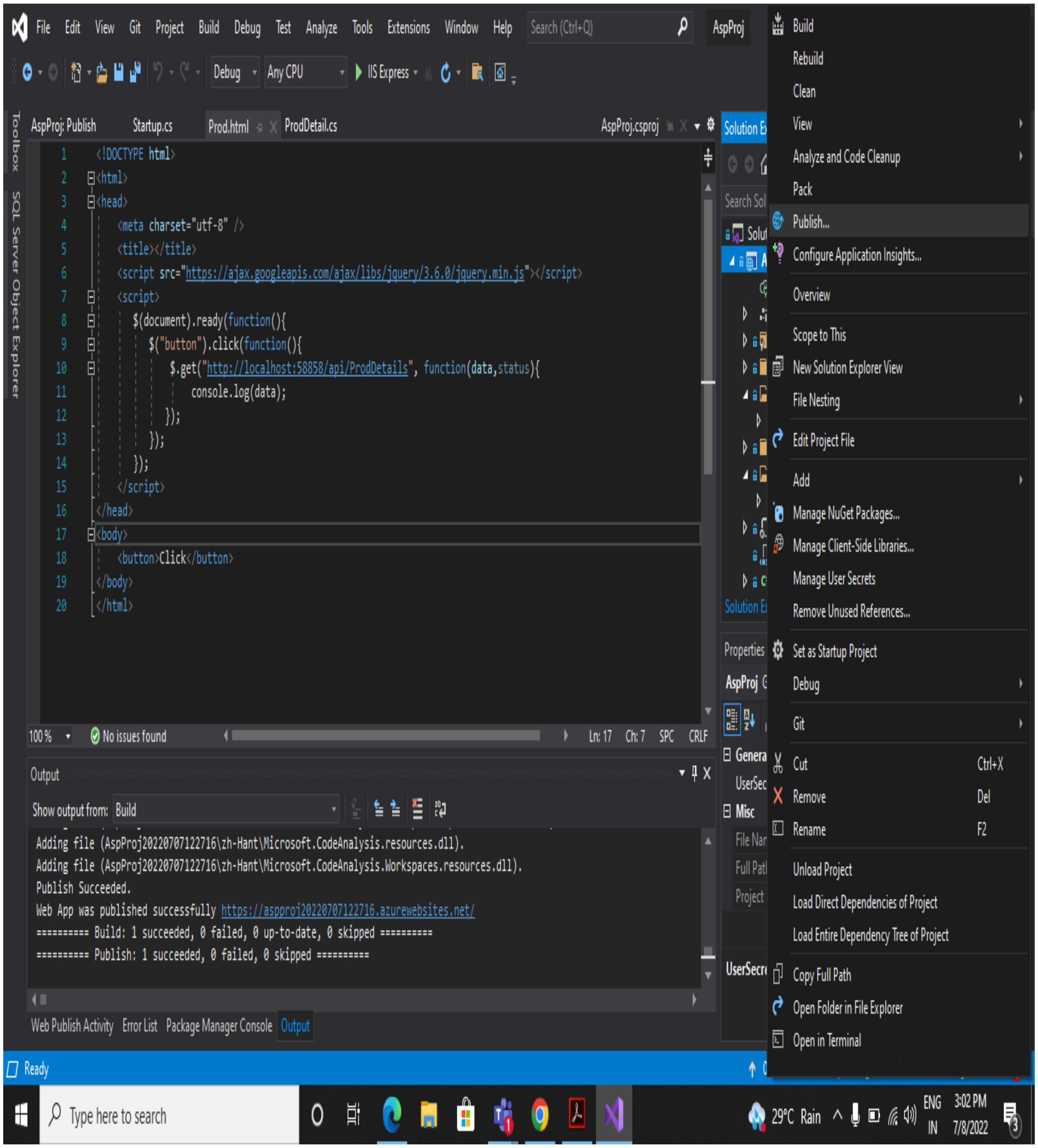
Create a JQuery and AJAX Client to consume the Web API and show the result.

Azure Hosting:

- Host the web api in azure and consume the same using JQuery Client.
- Configure Scale out by adding rules for custom scaling
- Configure Deployment slots for staging and production
- Configure Application Insights for the project
- Configure Swagger for the api
- Work with Log Analytics with the sample logs available

1.Host the web API in azure and consume the same using JQuery Client.

❖ In Solution Explorer, right-click the project and select Publish



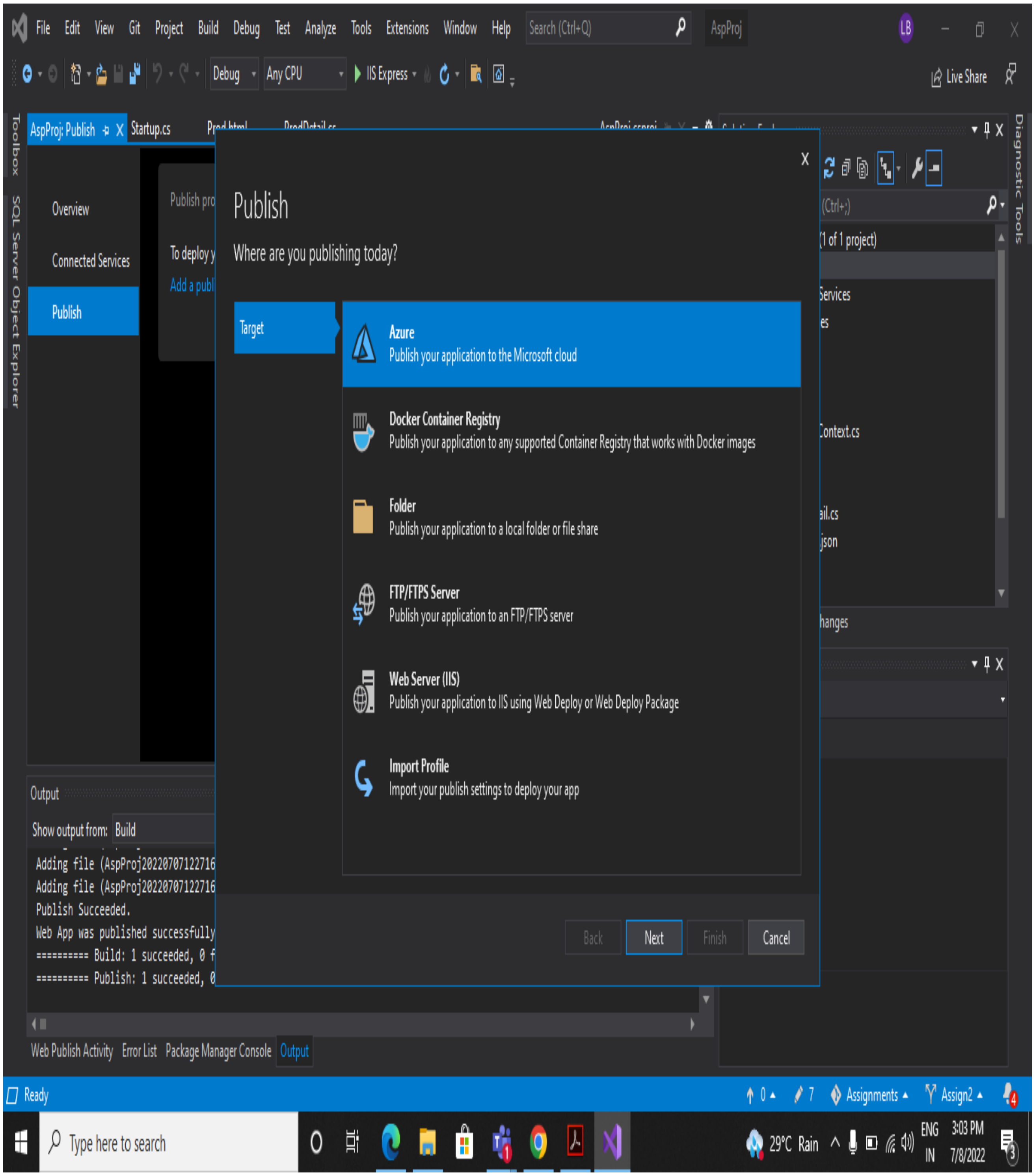
```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <title></title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
    <script>
      $(document).ready(function(){
        $("button").click(function(){
           $.get("http://localhost:58858/api/ProdDetails", function(data,status){
               console.log(data);
           });
        });
      });
    </script>
</head>
<body>
    <button>Click</button>
</body>
</html>
```

Show output from: Build

```
Adding file (AspProj20220707122716\zh-Hant\Microsoft.CodeAnalysis.resources.dll).
Adding file (AspProj20220707122716\zh-Hant\Microsoft.CodeAnalysis.Workspaces.resources.dll).
Publish Succeeded.
Web App was published successfully https://aspproj20220707122716.azurewebsites.net/
========== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ==========
========== Publish: 1 succeeded, 0 failed, 0 skipped ==========
```

❖

In the Publish dialog, select Azure and select the Next button



Publish

Where are you publishing today?

Target

Azure
Publish your application to the Microsoft cloud

Docker Container Registry
Publish your application to any supported Container Registry that works with Docker images

Folder
Publish your application to a local folder or file share

FTP/FTPS Server
Publish your application to an FTP/FTPS server

Web Server (IIS)
Publish your application to IIS using Web Deploy or Web Deploy Package

Import Profile
Import your publish settings to deploy your app

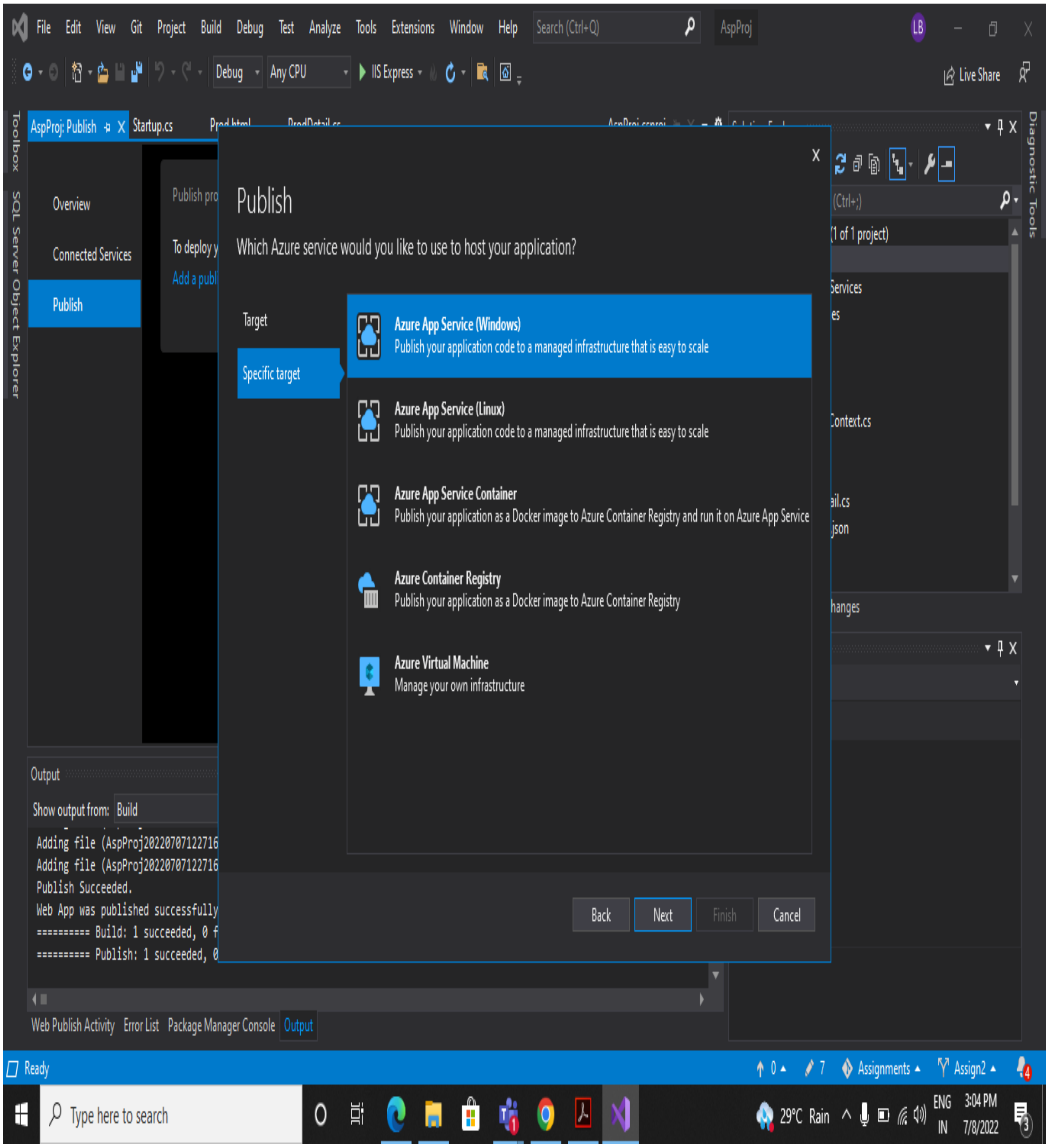
Back | Next | Finish | Cancel

Output

Show output from: Build

```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

❖ Select Azure App Service (Windows) and select the Next button



Publish

Which Azure service would you like to use to host your application?

Azure App Service (Windows)
Publish your application code to a managed infrastructure that is easy to scale

Azure App Service (Linux)
Publish your application code to a managed infrastructure that is easy to scale

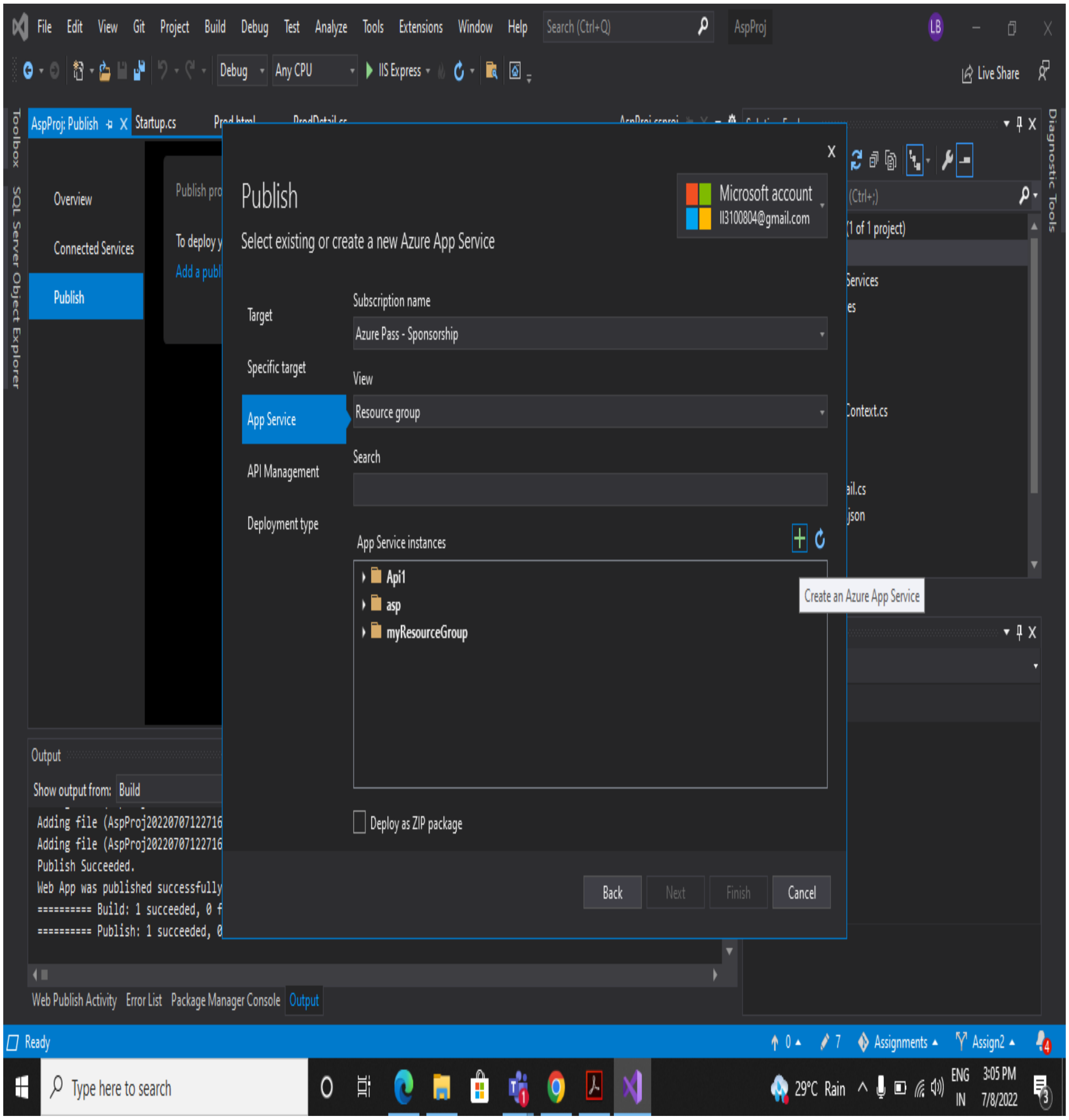
Azure App Service Container
Publish your application as a Docker image to Azure Container Registry and run it on Azure App Service

Azure Container Registry
Publish your application as a Docker image to Azure Container Registry

Azure Virtual Machine
Manage your own infrastructure

```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

❖ Select Create a new Azure App Service.



Publish

Select existing or create a new Azure App Service

Microsoft account
ll3100804@gmail.com

Subscription name: Azure Pass - Sponsorship

View: Resource group

Search

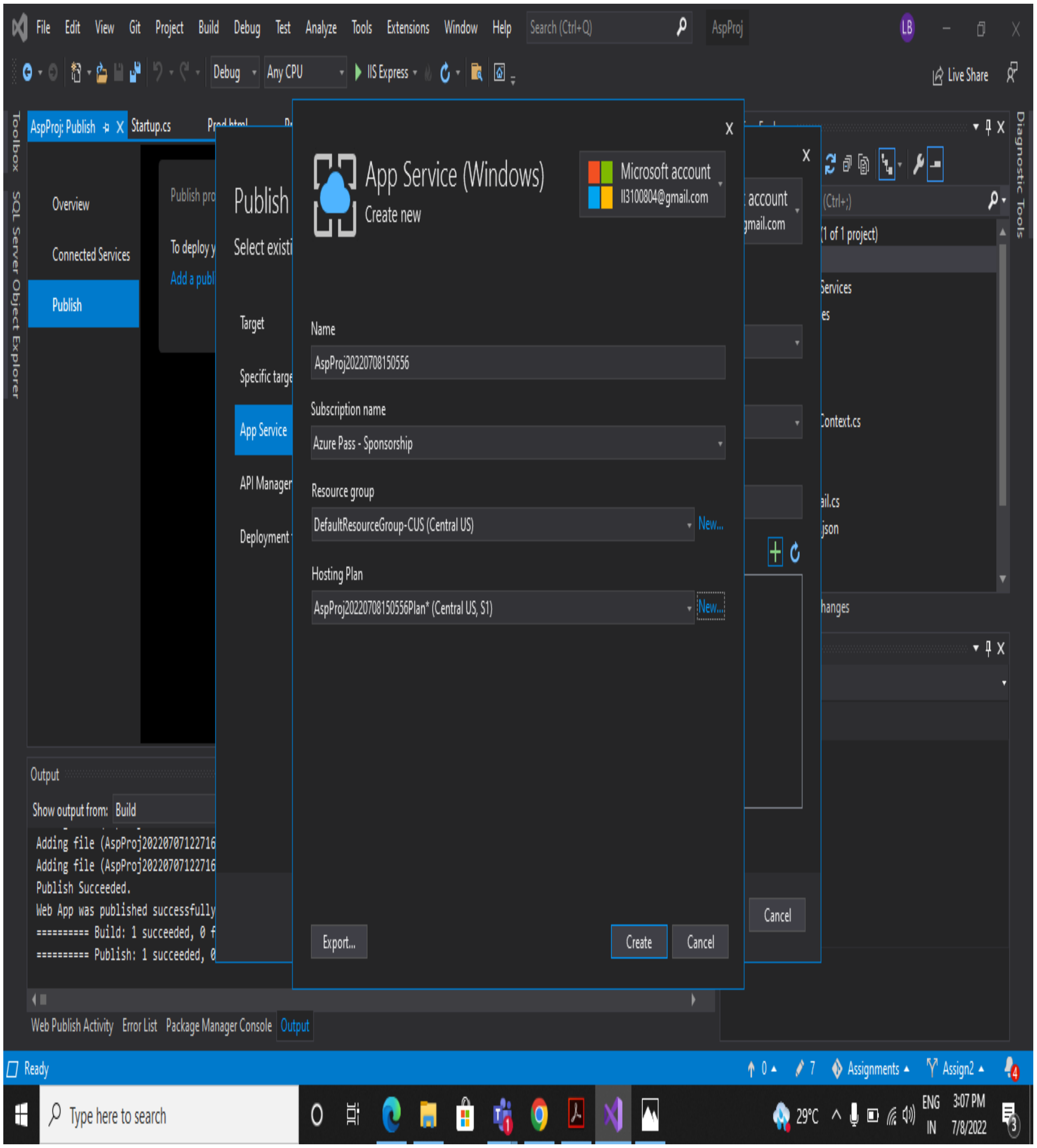
App Service instances
- Api1
- asp
- myResourceGroup

Deploy as ZIP package

```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

❖

The Create App Service dialog appears. The App Name, Resource Group, and App Service Plan entry fields are populated. You can keep these names or change them. Select the Create button.



Name: AspProj20220708150556

Subscription name: Azure Pass - Sponsorship

Resource group: DefaultResourceGroup-CUS (Central US) New...

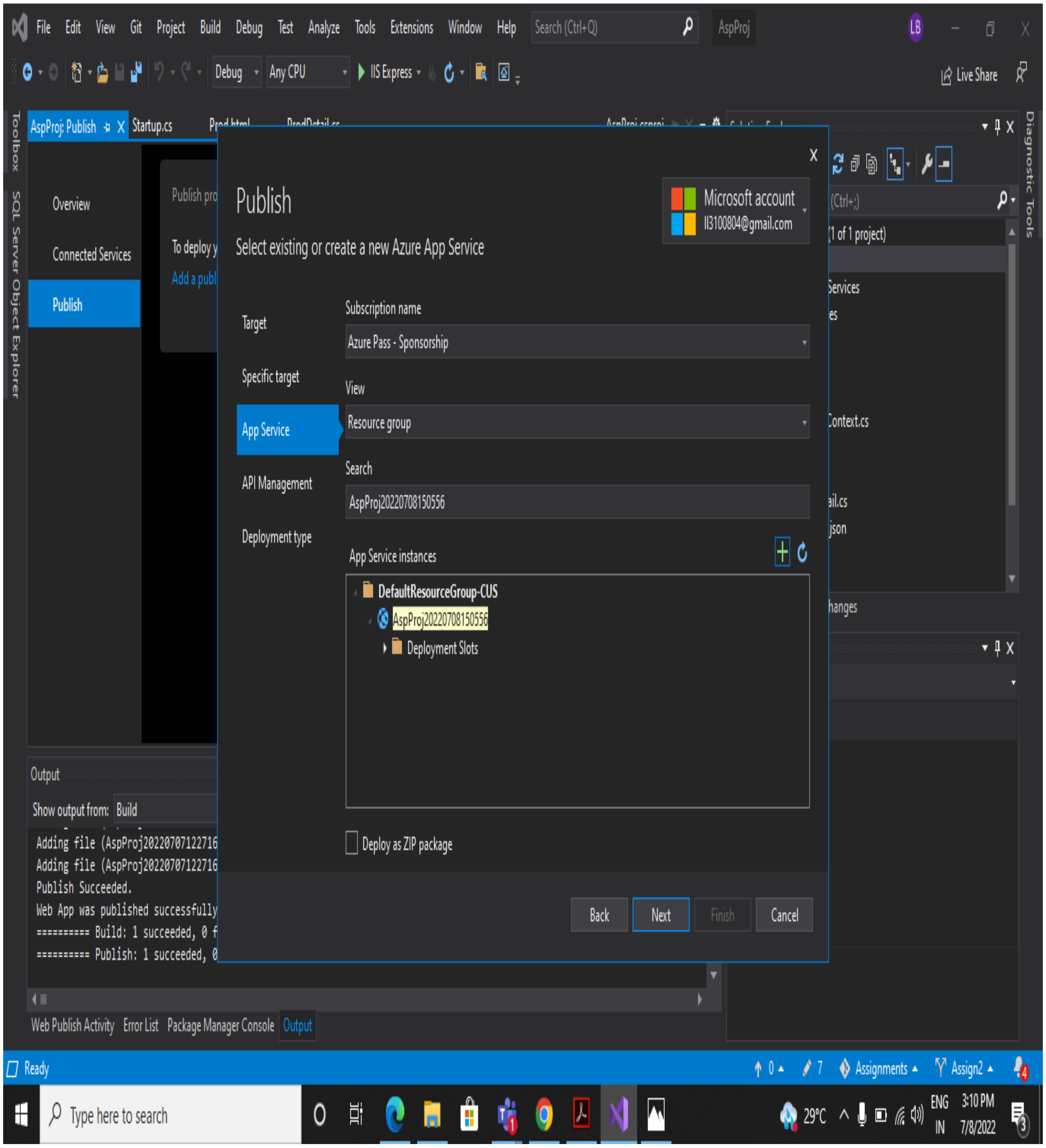
Hosting Plan: AspProj20220708150556Plan* (Central US, S1) New...

Output

Show output from: Build

```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

❖ After creation is completed, the dialog is automatically closed and the Publish dialog gets focus again. The instance that was created is automatically selected.



Subscription name: Azure Pass - Sponsorship

View: Resource group

Search: AspProj20220708150556

App Service instances

- DefaultResourceGroup-CUS
  - AspProj20220708150556
    - Deployment Slots

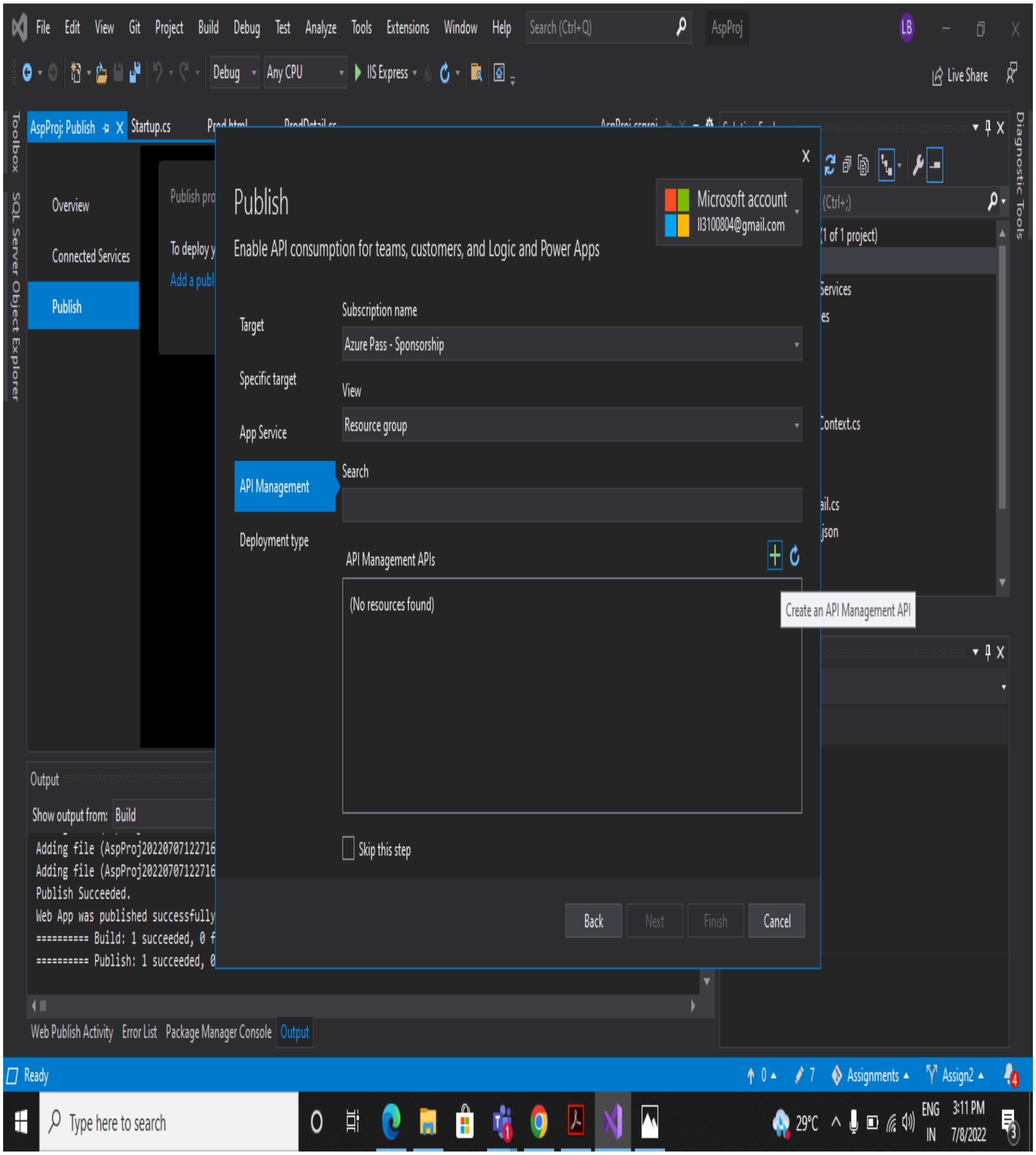
☐ Deploy as ZIP package

Show output from: Build

```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

❖

The dialog now shows the Azure API Management service to create.


Publish

Enable API consumption for teams, customers, and Logic and Power Apps

Microsoft account
ll3100804@gmail.com

Target
Specific target
App Service
API Management
Deployment type

Subscription name: Azure Pass - Sponsorship
View: Resource group
Search:

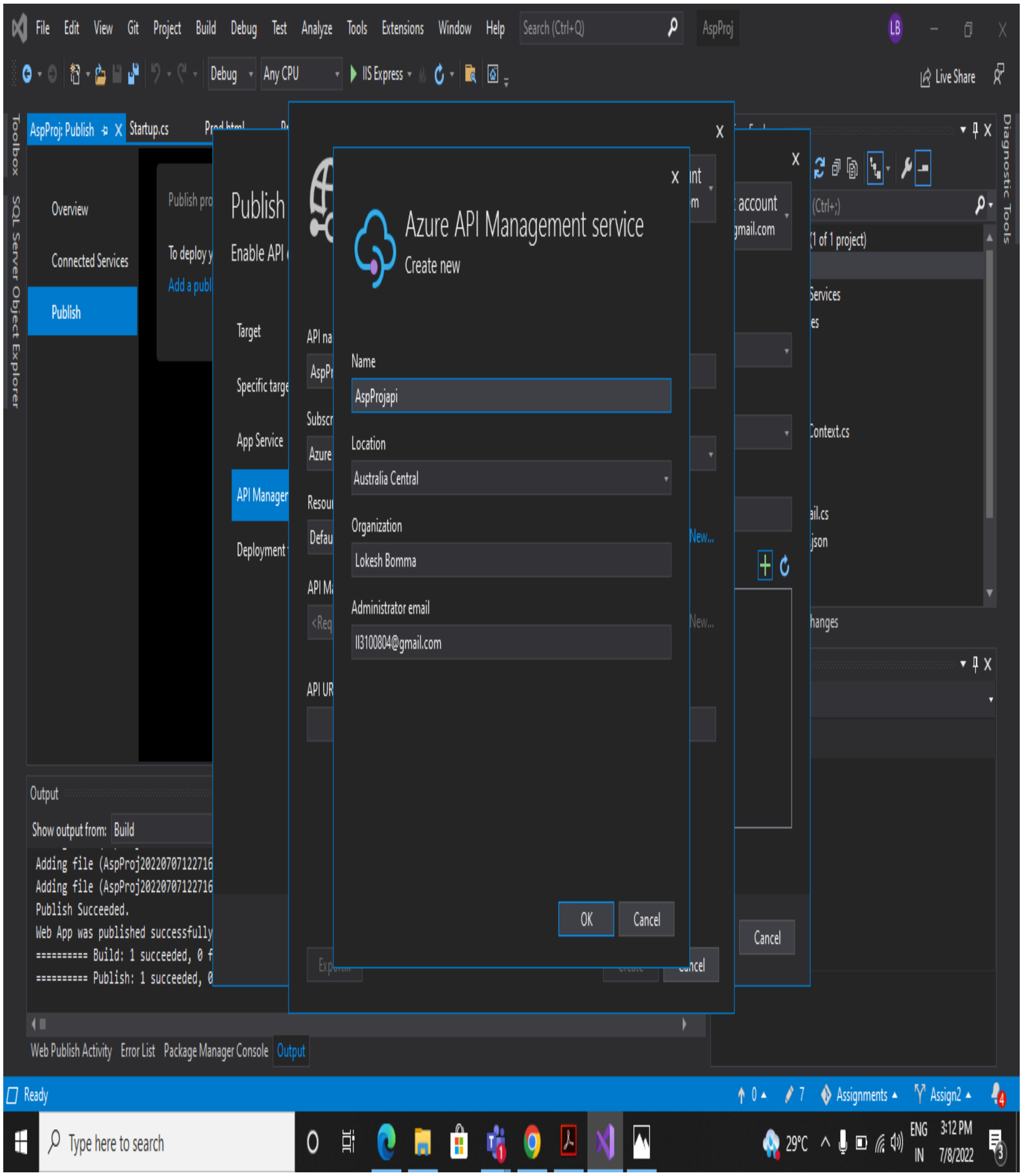
API Management APIs
(No resources found)
Create an API Management API

☐ Skip this step

Back | Next | Finish | Cancel

```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

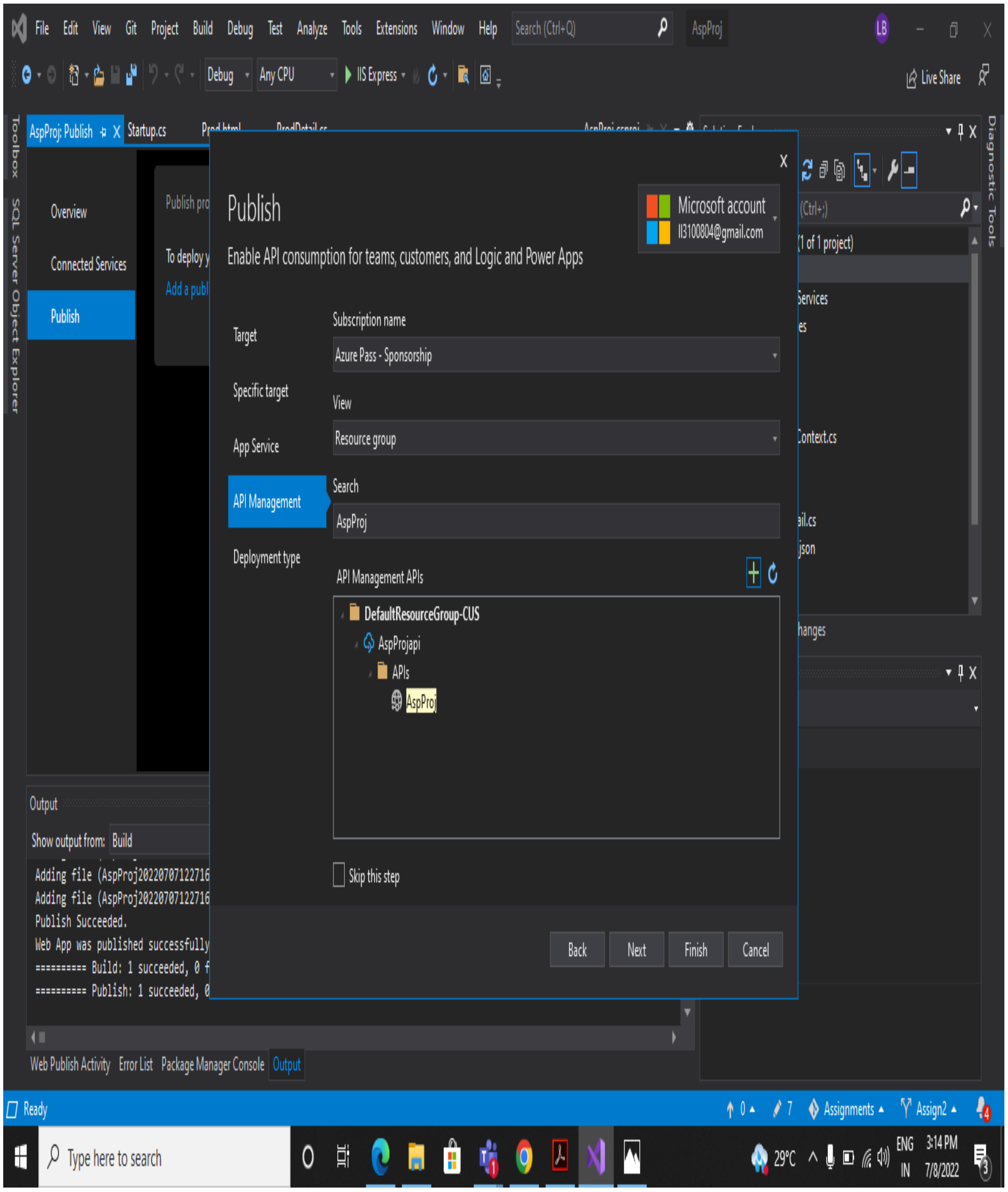
❖ The Create the Azure API Management service dialog appears. The App Name, Resource Group, and API Management service entry fields are populated. You can keep these names or change them. Select the Create button.



```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

❖

After creation is completed, the dialog is automatically closed and the Publish dialog gets focus again. The instance that was created is automatically selected.


Publish

Enable API consumption for teams, customers, and Logic and Power Apps

Microsoft account
ll3100804@gmail.com

Subscription name: Azure Pass - Sponsorship

View: Resource group

Search: AspProj

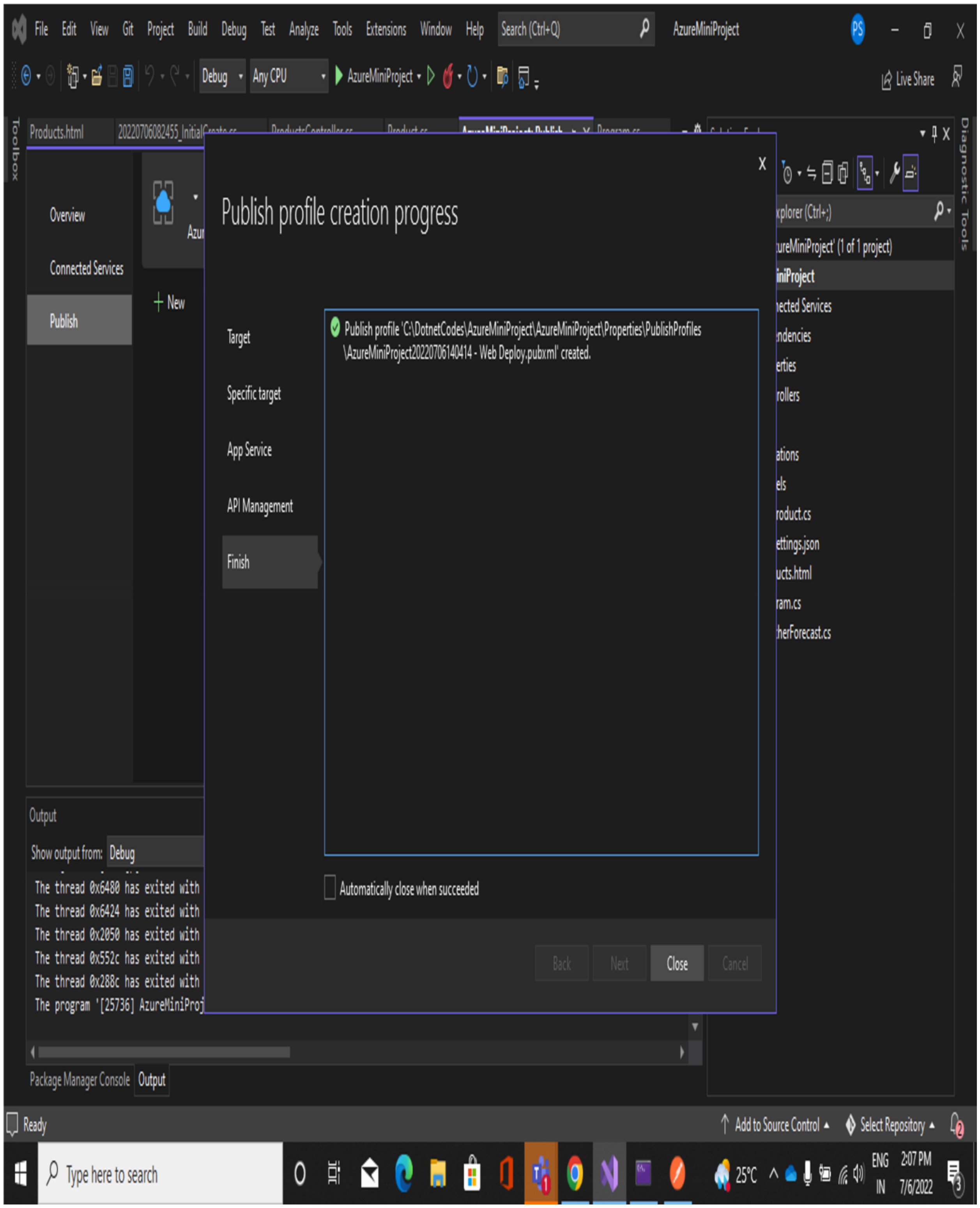
API Management APIs

- DefaultResourceGroup-CUS
  - AspProjapi
    - APIs
      - AspProj

```
Adding file (AspProj20220707122716
Adding file (AspProj20220707122716
Publish Succeeded.
Web App was published successfully
========== Build: 1 succeeded, 0 f
========== Publish: 1 succeeded, 0
```

❖

Click on the public profile create

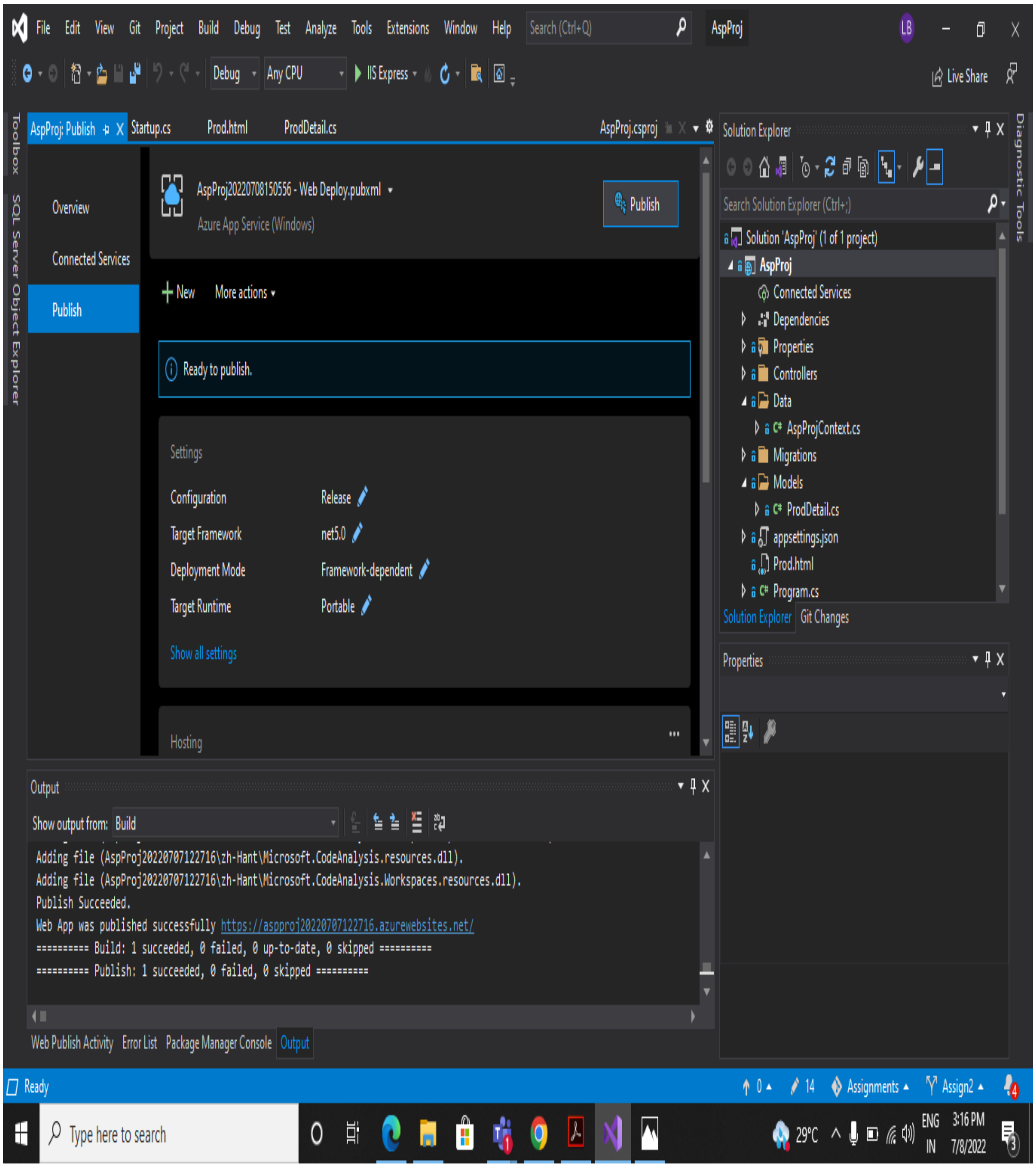

Publish profile creation progress

Publish profile 'C:\DotnetCodes\AzureMiniProject\AzureMiniProject\Properties\PublishProfiles\AzureMiniProject20220706140414 - Web Deploy.pubxml' created.

Automatically close when succeeded

```
The thread 0x6480 has exited with
The thread 0x6424 has exited with
The thread 0x2050 has exited with
The thread 0x552c has exited with
The thread 0x288c has exited with
The program '[25736] AzureMiniProj
```

❖ The dialog closes and a summary screen appears with information about the publish. Select the Publish button.



AspProj20220708150556 - Web Deploy.pubxml

Azure App Service (Windows)

Ready to publish.

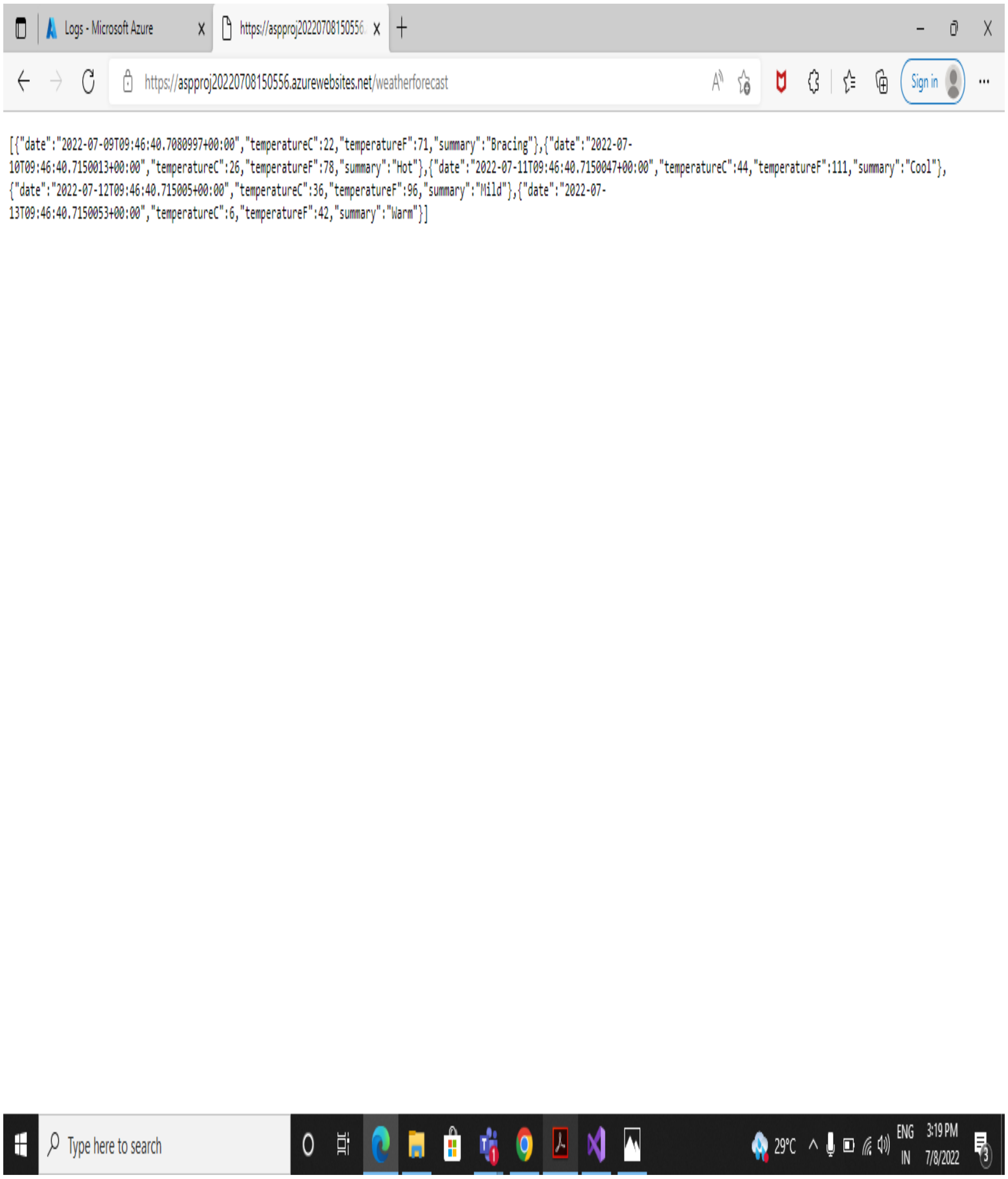
Settings

Configuration	Release
Target Framework	net5.0
Deployment Mode	Framework-dependent
Target Runtime	Portable

Show all settings

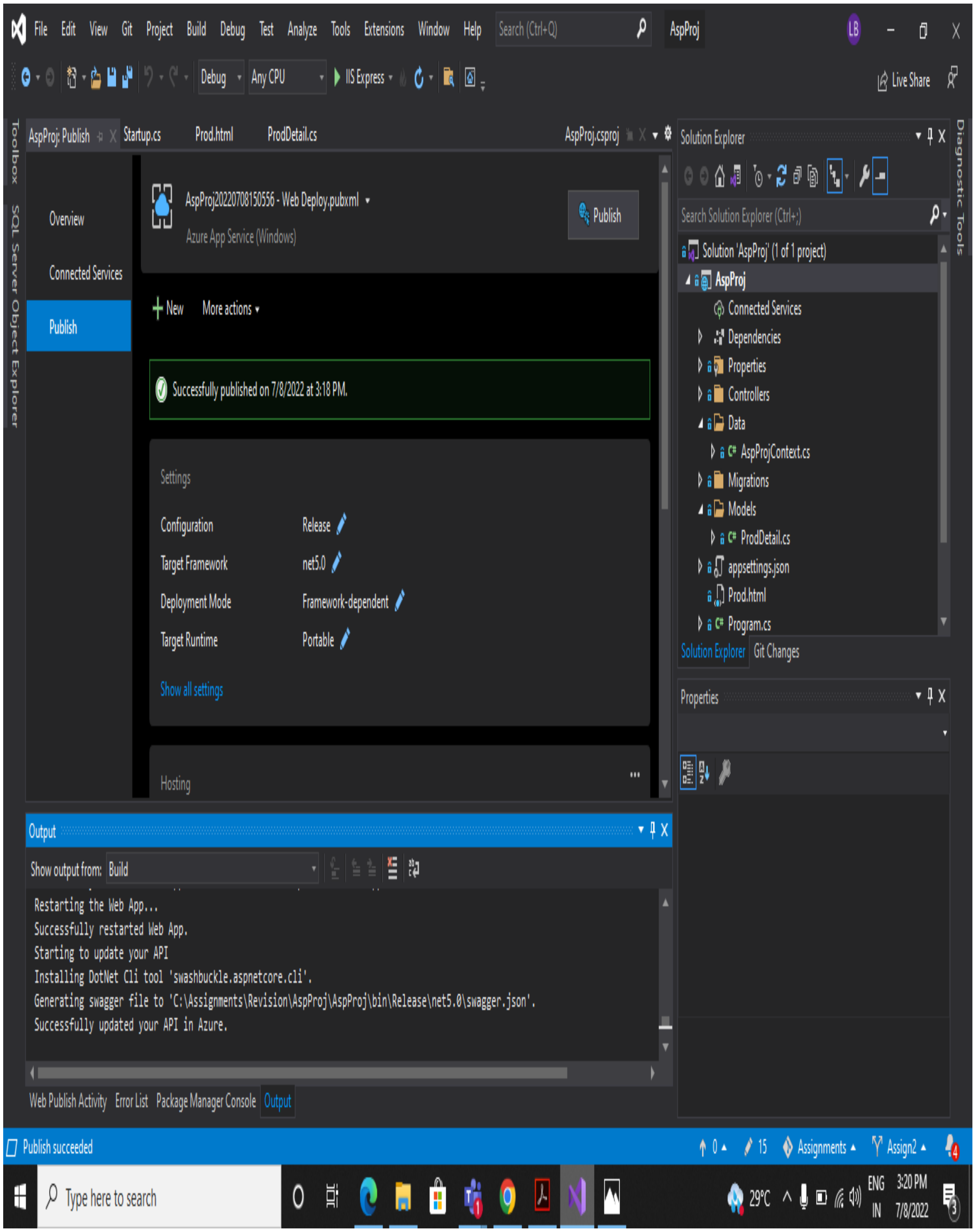
```
Adding file (AspProj20220707122716\zh-Hant\Microsoft.CodeAnalysis.resources.dll).
Adding file (AspProj20220707122716\zh-Hant\Microsoft.CodeAnalysis.Workspaces.resources.dll).
Publish Succeeded.
Web App was published successfully https://aspproj20220707122716.azurewebsites.net/
========== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ==========
========== Publish: 1 succeeded, 0 failed, 0 skipped ==========
```

❖ The web API will publish to both Azure App Service and Azure API Management. A new browser window will appear and show the API running in Azure App Service. You can close that window.


Logs - Microsoft Azure
https://aspproj20220708150556.
https://aspproj20220708150556.azurewebsites.net/weatherforecast
Sign in
Type here to search
29°C
ENG
IN
3:19 PM
7/8/2022

```
[{"date":"2022-07-09T09:46:40.7080997+00:00","temperatureC":22,"temperatureF":71,"summary":"Bracing"},{"date":"2022-07-
10T09:46:40.7150013+00:00","temperatureC":26,"temperatureF":78,"summary":"Hot"},{"date":"2022-07-11T09:46:40.7150047+00:00","temperatureC":44,"temperatureF":111,"summary":"Cool"},
{"date":"2022-07-12T09:46:40.715005+00:00","temperatureC":36,"temperatureF":96,"summary":"Mild"},{"date":"2022-07-
13T09:46:40.7150053+00:00","temperatureC":6,"temperatureF":42,"summary":"Warm"}]
```

❖ Select the Publish button.



AspProj20220708150556 - Web Deploy.pubxml
Azure App Service (Windows)

Successfully published on 7/8/2022 at 3:18 PM.

Settings

Configuration	Release
Target Framework	net5.0
Deployment Mode	Framework-dependent
Target Runtime	Portable

Show all settings

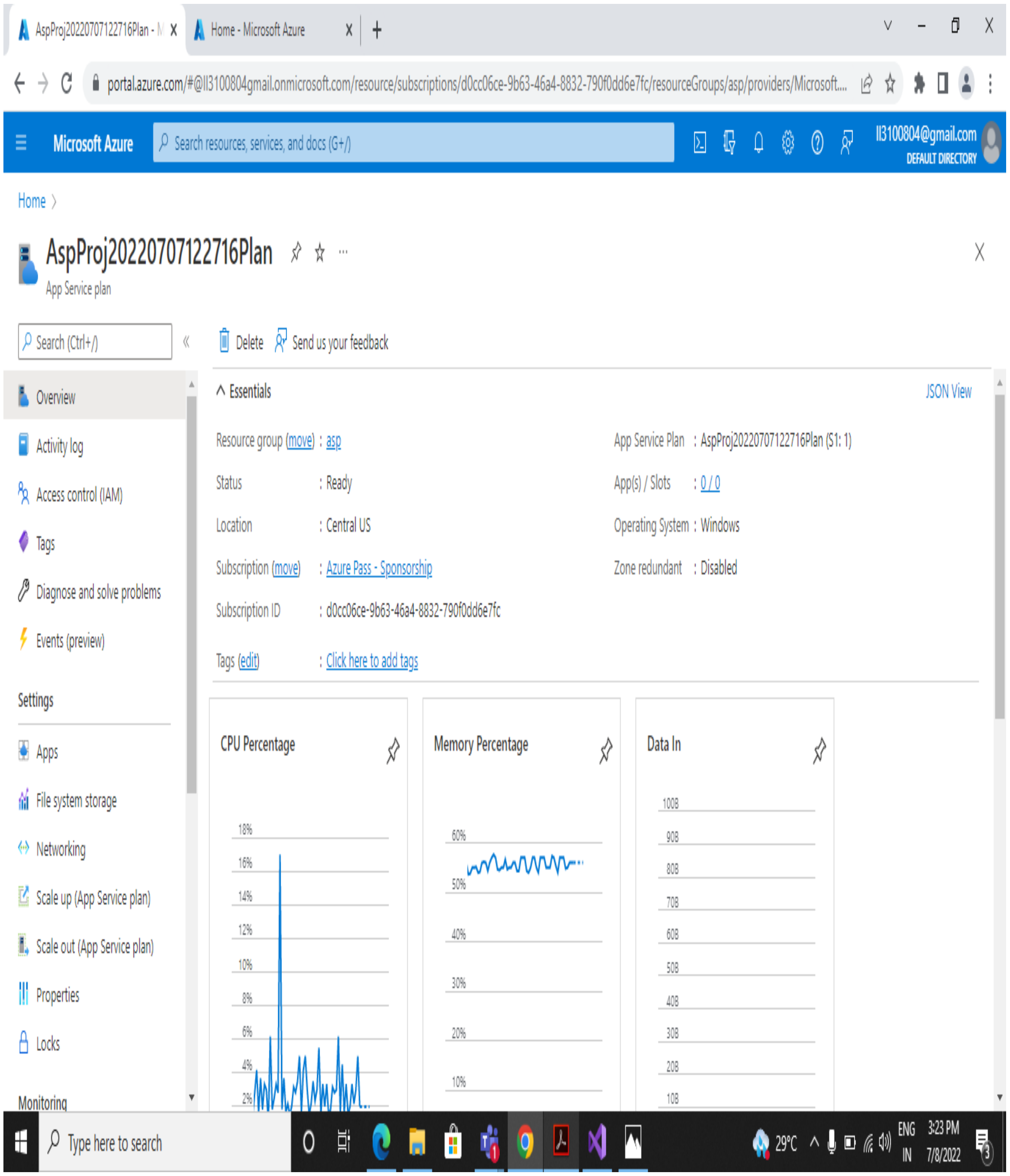
Hosting

Show output from: Build

```
Restarting the Web App...
Successfully restarted Web App.
Starting to update your API
Installing DotNet Cli tool 'swashbuckle.aspnetcore.cli'.
Generating swagger file to 'C:\Assignments\Revision\AspProj\AspProj\bin\Release\net5.0\swagger.json'.
Successfully updated your API in Azure.
```

Solution 'AspProj' (1 of 1 project)
- AspProj
  - Connected Services
  - Dependencies
  - Properties
  - Controllers
  - Data
    - AspProjContext.cs
  - Migrations
  - Models
    - ProdDetail.cs
  - appsettings.json
  - Prod.html
  - Program.cs

❖ Switch back to the Azure API Management instance in the Azure portal. Refresh the browser window. Select the API you created in the preceding steps. It's now populated and you can explore around.



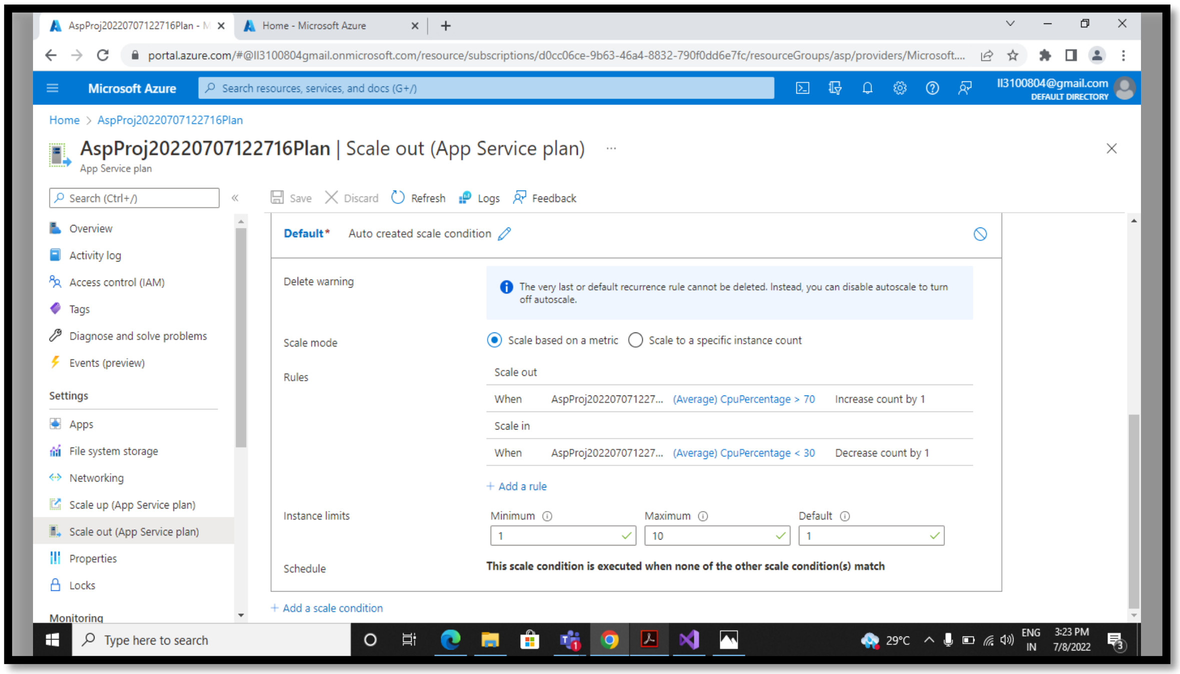
AspProj20220707122716Plan
App Service plan

Essentials

Resource group (move)	: asp	App Service Plan	: AspProj20220707122716Plan (S1: 1)
Status	: Ready	App(s) / Slots	: 0 / 0
Location	: Central US	Operating System	: Windows
Subscription (move)	: Azure Pass - Sponsorship	Zone redundant	: Disabled
Subscription ID	: d0cc06ce-9b63-46a4-8832-790f0dd6e7fc		
Tags (edit)	: Click here to add tags		

2. Configure Scale out by adding rules for custom scaling

- Search and select Autoscale in the search bar
- Select Custom Autoscale
- In the Rules section of the default scale condition, select Add a rule.
- From the Metric source dropdown, select current resource.
- From Resource Type, select Application Insights.
- From the Resource dropdown, select your App services plan standard metrics.
- Select a Metric name to CPU Percentage.
- Select Enable metric divide by instance count so that the number of sessions per instance is measured.
- From the Operator dropdown, select Greater than.
- Enter the Metric threshold to trigger the scale action, for example, 70.
- Under Actions, set the Operation to Increase count and set the Instance count to 1 and Cool down by 5minutes and then click Add.
- Set the maximum number of instances that can be spun up in the Maximum field of the Instance limits section for example, 1.
- Select Save.



AspProj20220707122716Plan | Scale out (App Service plan)

App Service plan

Default* Auto created scale condition

Delete warning: The very last or default recurrence rule cannot be deleted. Instead, you can disable autoscale to turn off autoscale.

Scale mode: [✓ Scale based on a metric] Scale to a specific instance count

Rules

Scale out			
When	AspProj20220707 1227...	(Average) CpuPercentage > 70	Increase count by 1
Scale in			
When	AspProj20220707 1227...	(Average) CpuPercentage < 30	Decrease count by 1

+ Add a rule

Instance limits

Minimum	Maximum	Default
1	10	1

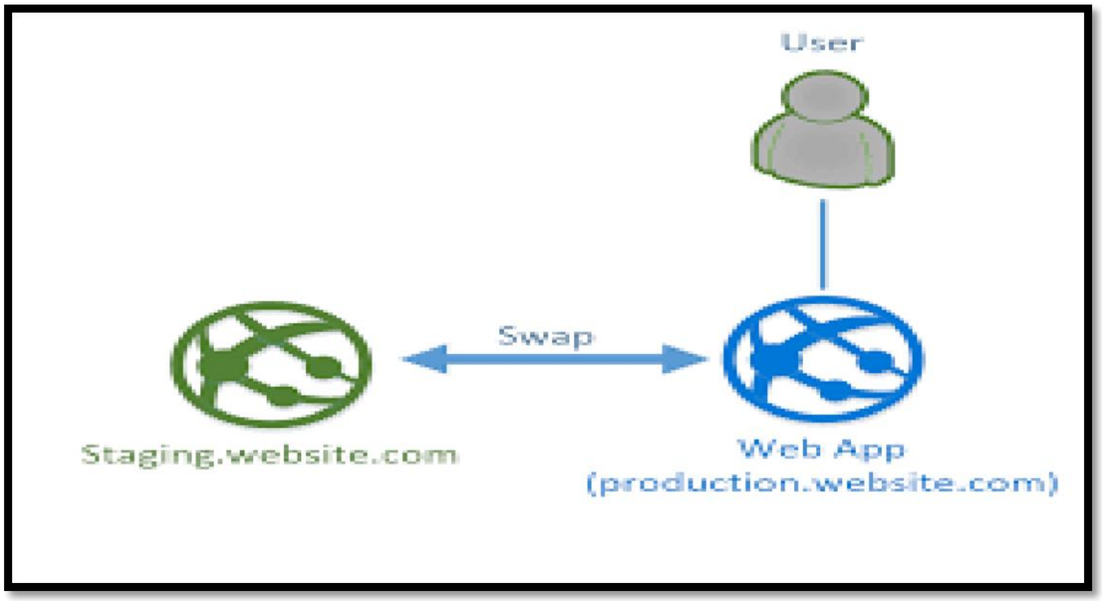
Schedule: This scale condition is executed when none of the other scale condition(s) match

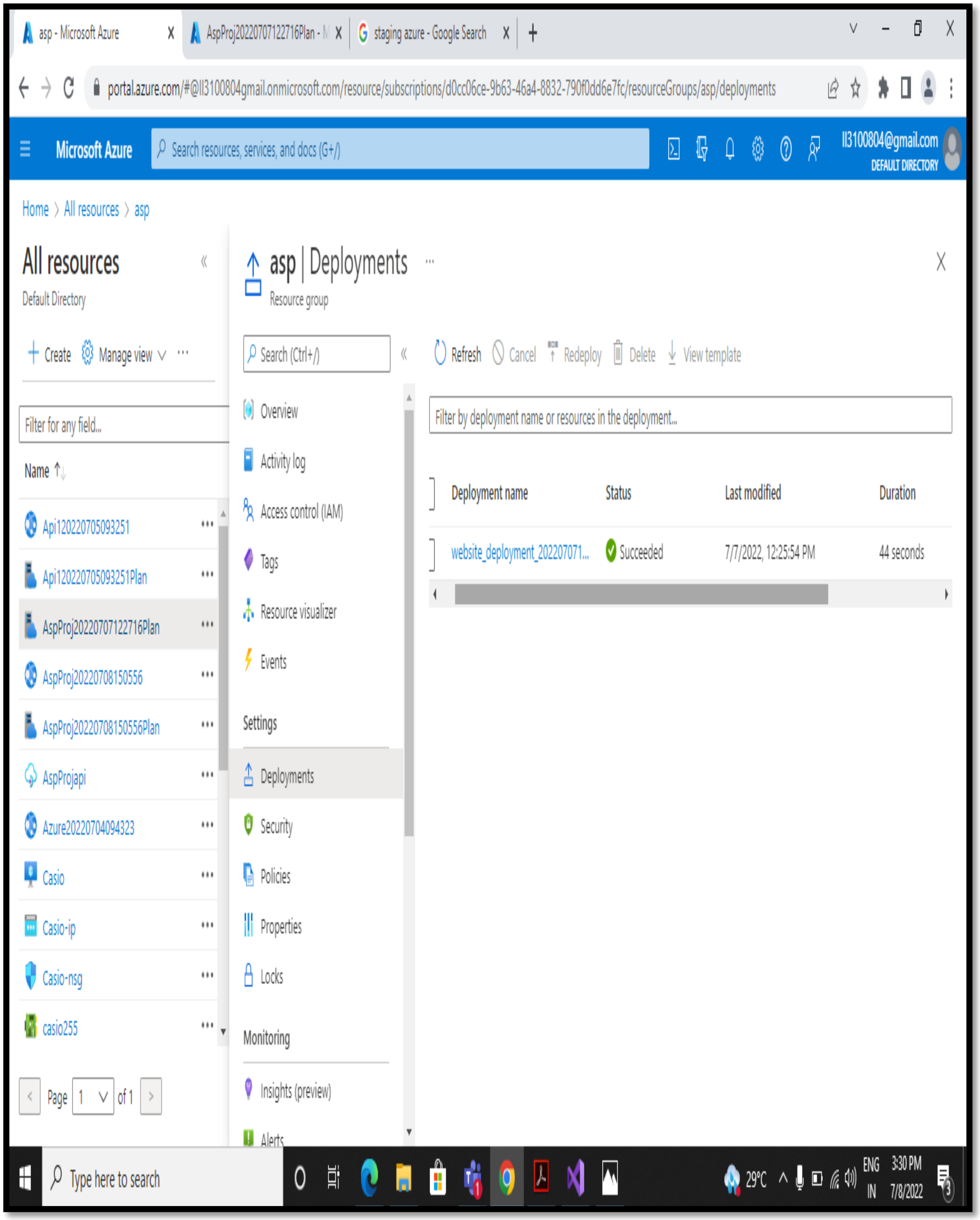
+ Add a scale condition

3. Configure Deployment slots for staging and production

Azure Functions deployment slots allow your function app to run different instances called "slots". Slots are different environments exposed via a publicly available endpoint. One app instance is always mapped to the production slot, and you can swap instances assigned to a slot on demand. Function apps running under the Apps Service plan may have multiple slots, while under the Consumption plan only one slot is allowed.

- Navigate to Deployment slots in the function app, and then select the slot name.
- Select Configuration, and then select the setting name you want to stick with the current slot.
- Select Deployment slot setting, and then select OK.
- Once setting section disappears, select Save to keep the changes.
- Select Deployment slots, and then select + Add Slot.
- Type the name of the slot and select Add.
- Select Deployment slots, and then select Swap.
- Verify the configuration settings for your swap and select Swap.
- The operation may take a moment while the swap operation is executing.





asp - Microsoft Azure | AspProj20220707122716Plan - M | staging azure - Google Search

portal.azure.com/#@ll3100804gmail.onmicrosoft.com/resource/subscriptions/d0cc06ce-9b63-46a4-8832-790f0dd6e7fc/resourceGroups/asp/deployments

Microsoft Azure | Search resources, services, and docs (G+/) | ll3100804@gmail.com DEFAULT DIRECTORY

Home > All resources > asp

All resources
Default Directory

Create | Manage view

Filter for any field...

Name
Api120220705093251
Api120220705093251Plan
AspProj20220707122716Plan
AspProj20220708150556
AspProj20220708150556Plan
AspProjapi
Azure20220704094323
Casio
Casio-ip
Casio-nsg
casio255

Page 1 of 1

asp | Deployments
Resource group

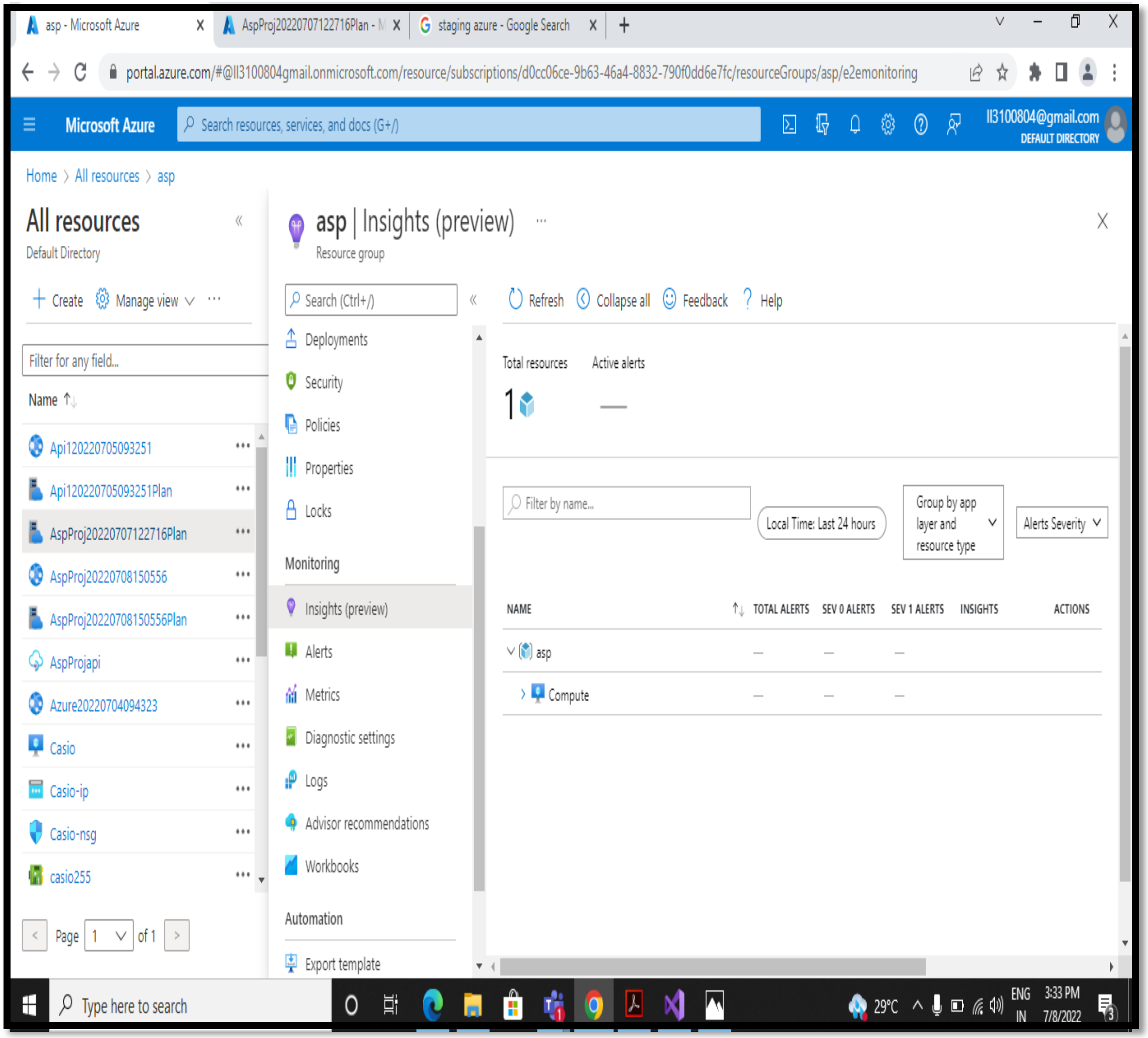
Filter by deployment name or resources in the deployment...

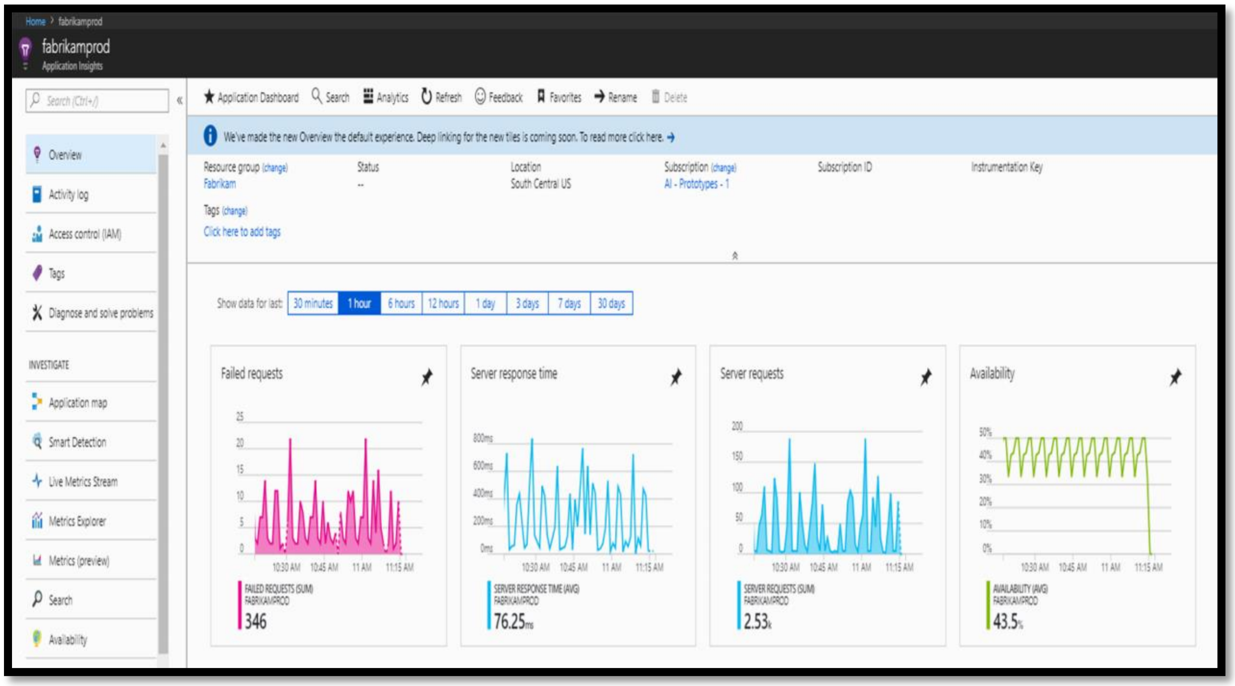
Deployment name	Status	Last modified	Duration
website_deployment_202207071...	Succeeded	7/7/2022, 12:25:54 PM	44 seconds

Type here to search | 29°C | ENG IN | 3:30 PM 7/8/2022

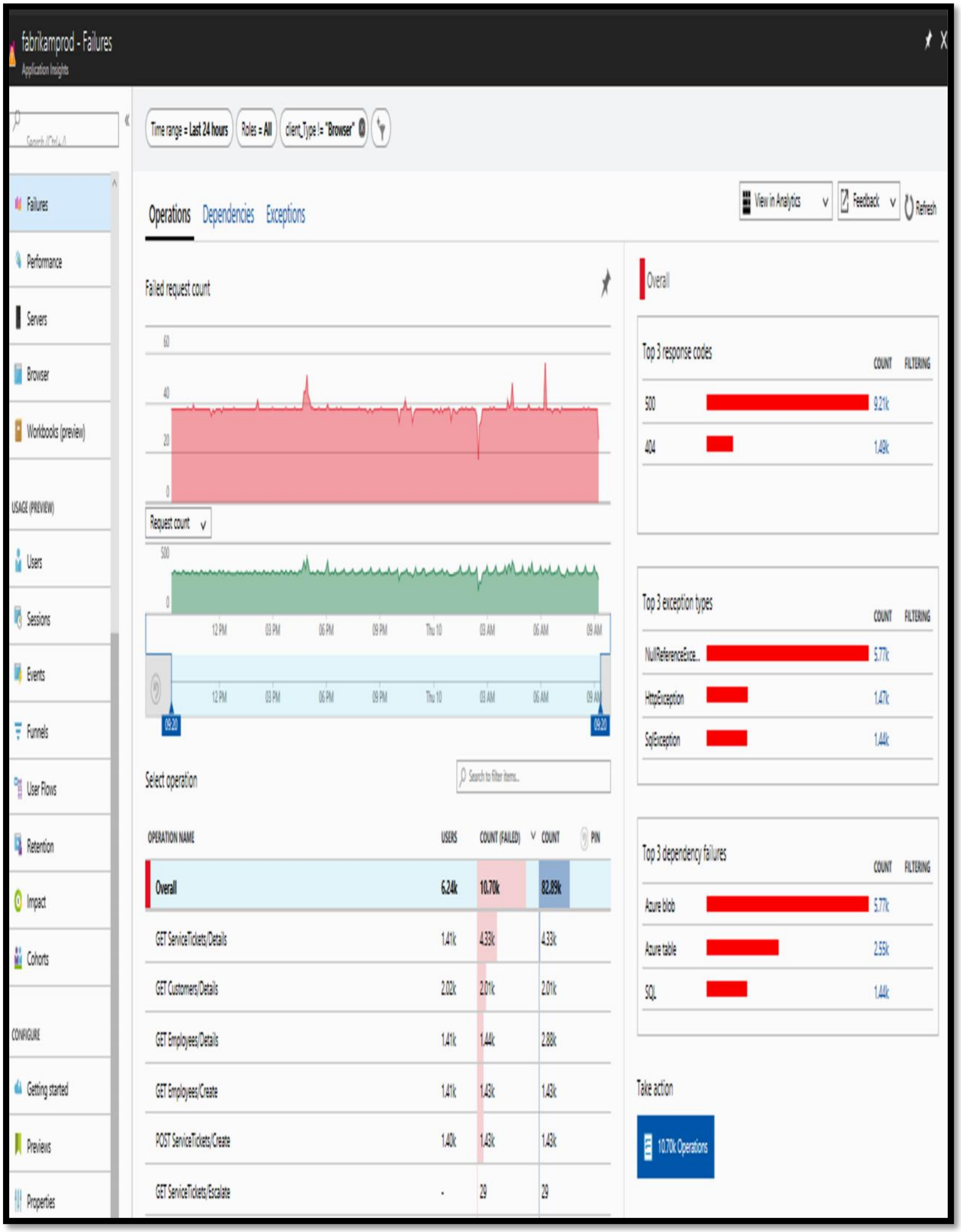
4. Configure Application Insights for the project

Application Insights can monitor Azure cloud service apps for availability, performance, failures, and usage by combining data from Application Insights SDKs with Azure Diagnostics data from your cloud services. With the feedback you get about the performance and effectiveness of your app in the wild, you can make informed choices about the direction of the design in each development lifecycle.




Home > fabrikamprod
fabrikamprod
Application Insights
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
INVESTIGATE
Application map
Smart Detection
Live Metrics Stream
Metrics Explorer
Metrics (preview)
Search
Availability
Application Dashboard
Search
Analytics
Refresh
Feedback
Favorites
Rename
Delete
We've made the new Overview the default experience. Deep linking for the new tiles is coming soon. To read more click here.
Resource group (change)
Fabrikam
Status
--
Location
South Central US
Subscription (change)
AI - Prototypes - 1
Subscription ID
Instrumentation Key
Tags (change)
Click here to add tags
Show data for last:
30 minutes
1 hour
6 hours
12 hours
1 day
3 days
7 days
30 days
Failed requests
FAILED REQUESTS (SUM)
FABRIKAMPROD
346
Server response time
SERVER RESPONSE TIME (AVG)
FABRIKAMPROD
76.25ms
Server requests
SERVER REQUESTS (SUM)
FABRIKAMPROD
2.53k
Availability
AVAILABILITY (AVG)
FABRIKAMPROD
43.5%
10:30 AM
10:45 AM
11 AM
11:15 AM

57.16
592.63
59
3
346.88
17.13
1
217.25
57.16
2
8



OPERATION NAME	USERS	COUNT (FAILED)	COUNT	PIN
Overall	6.24k	10.70k	82.89k	
GET ServiceTickets/Details	1.41k	4.33k	4.33k	
GET Customers/Details	2.02k	2.01k	2.01k	
GET Employees/Details	1.41k	1.44k	2.88k	
GET Employees/Create	1.41k	1.43k	1.43k	
POST ServiceTickets/Create	1.40k	1.43k	1.43k	
GET ServiceTickets/Escalate	-	29	29	

Top 3 response codes

	COUNT	FILTERING
500	9.21k	
404	1.49k	

Top 3 exception types

	COUNT	FILTERING
NullReferenceExce...	5.77k	
HttpException	1.47k	
SqlException	1.44k	

Top 3 dependency failures

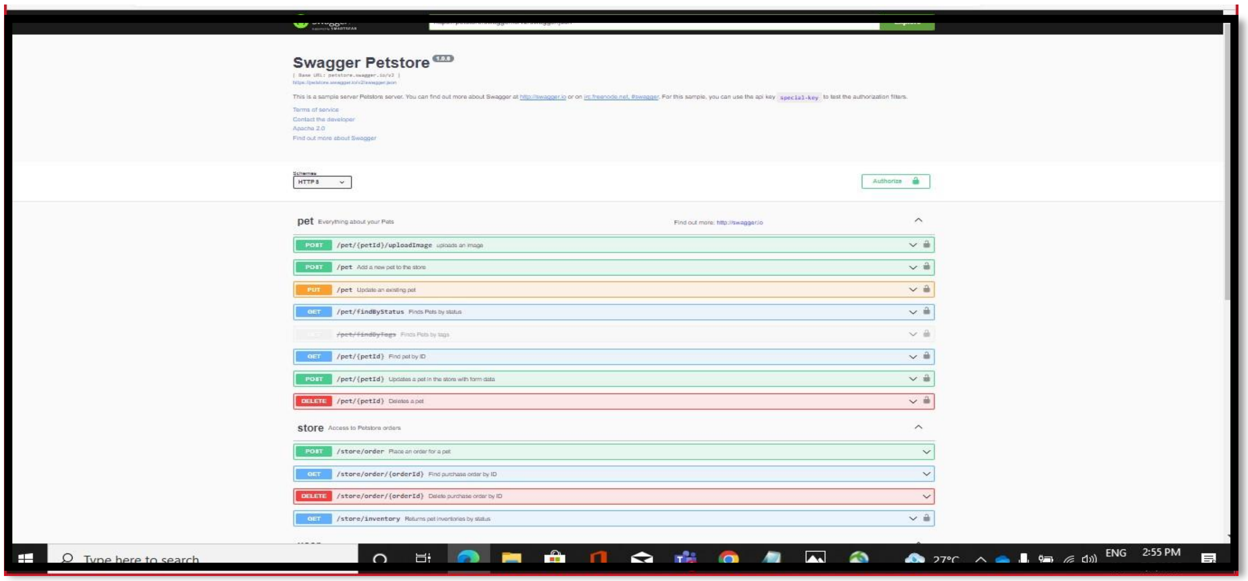
	COUNT	FILTERING
Azure blob	5.77k	
Azure table	2.55k	
SQL	1.44k	

4. Configure Swagger for the API

Swagger UI allows anyone be it your development team or your end consumers to visualize and interact with the API's resources without having any of the implementation logic in place. It's automatically generated from your Open API (formerly known as Swagger) Specification, with the visual documentation making it easy for back end implementation and client side consumption.

Advantages :

- Dependency Free - The UI works in any development environment, be it locally or in the web
- Human Friendly - Allow end developers to effortlessly interact and try out every single operation your API exposes for easy consumption
- Easy to Navigate - Quickly find and work with resources and endpoints with neatly categorized documentation
- All Browser Support - Cater to every possible scenario with Swagger UI working in all major browsers.
- Fully Customizable - Style and tweak your Swagger UI the way you want with full source code access.
- Complete OAS Support - Visualize APIs defined in Swagger 2.0 or OAS 3.0



Swagger Petstore 1.0.6

[Base URL: petstore.swagger.io/v2]

https://petstore.swagger.io/v2/swagger.json

This is a sample server Petstore server. You can find out more about Swagger at http://swagger.io or on irc.freenode.net, #swagger. For this sample, you can use the api key `special-key` to test the authorization filters.

Terms of service
Contact the developer
Apache 2.0
Find out more about Swagger

Schemes: HTTPS

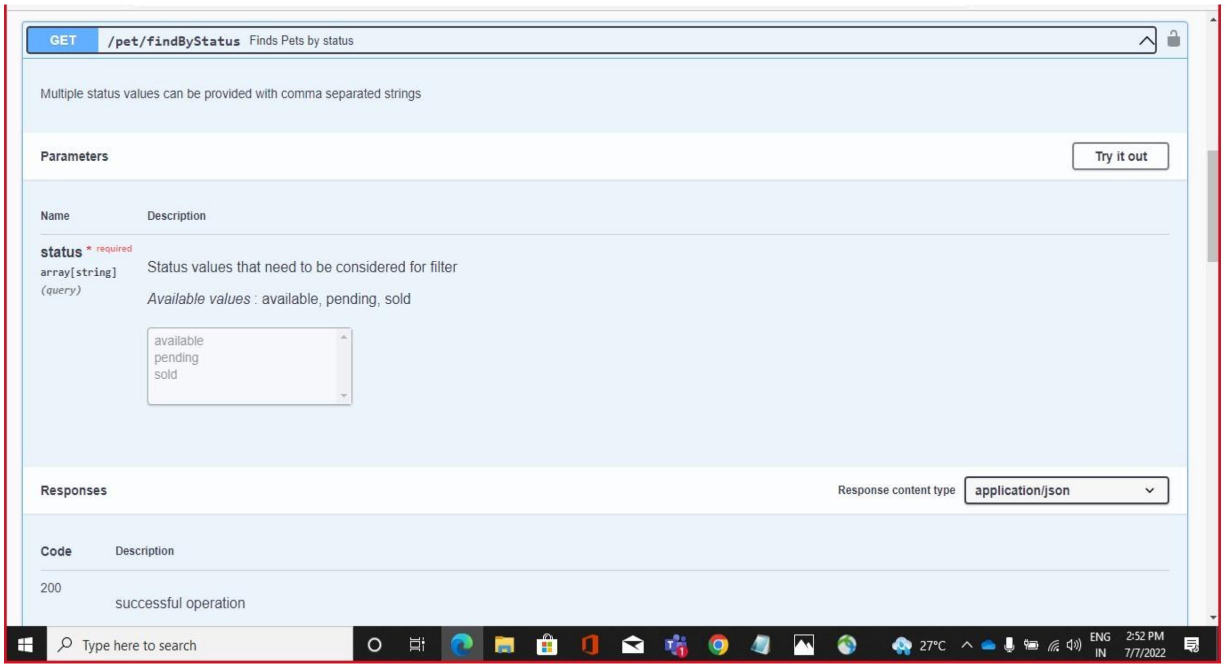
pet Everything about your Pets — Find out more: http://swagger.io

POST /pet/{petId}/uploadImage uploads an image
POST /pet Add a new pet to the store
PUT /pet Update an existing pet
GET /pet/findByStatus Finds Pets by status
~~/pet/findByTags~~ Finds Pets by tags
GET /pet/{petId} Find pet by ID
POST /pet/{petId} Updates a pet in the store with form data
DELETE /pet/{petId} Deletes a pet

store Access to Petstore orders

POST /store/order Place an order for a pet
GET /store/order/{orderId} Find purchase order by ID
DELETE /store/order/{orderId} Delete purchase order by ID
GET /store/inventory Returns pet inventories by status

GET



GET /pet/findByStatus Finds Pets by status

Multiple status values can be provided with comma separated strings

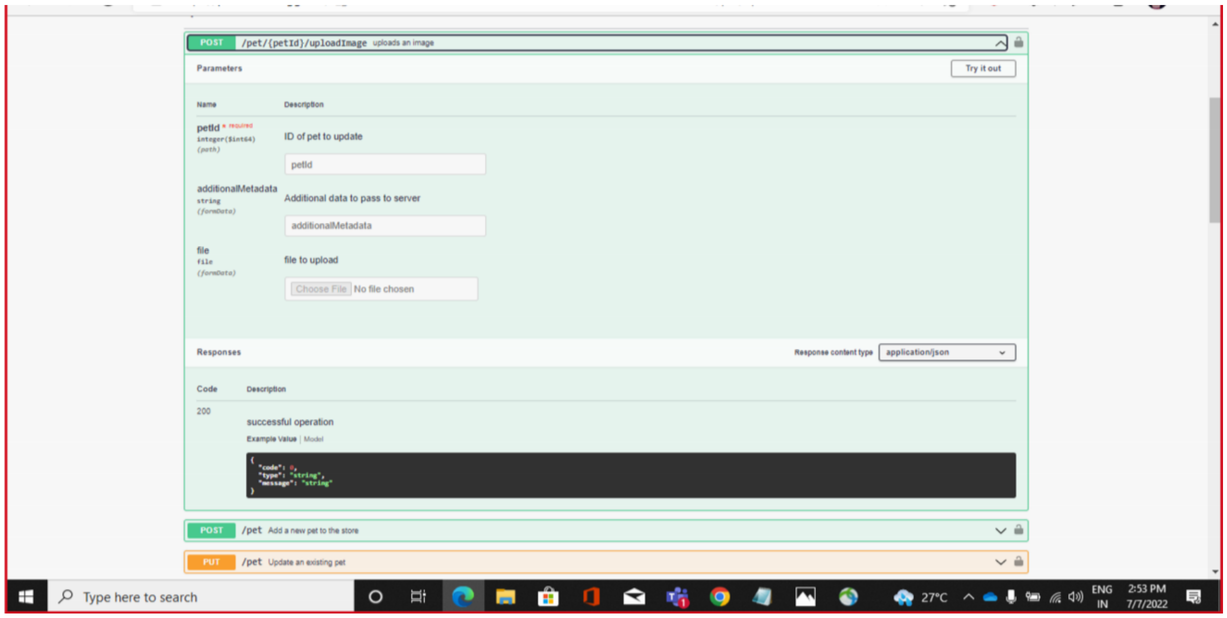
Parameters

Name	Description
status * required array[string] *(query)*	Status values that need to be considered for filter *Available values* : available, pending, sold available pending sold

Responses — Response content type: application/json

Code	Description
200	successful operation

POST



POST /pet/{petId}/uploadImage uploads an image

Parameters

Name	Description
petId * required integer($int64) *(path)*	ID of pet to update petId
additionalMetadata string *(formData)*	Additional data to pass to server additionalMetadata
file file *(formData)*	file to upload Choose File No file chosen

Responses — Response content type: application/json

Code	Description
200	successful operation

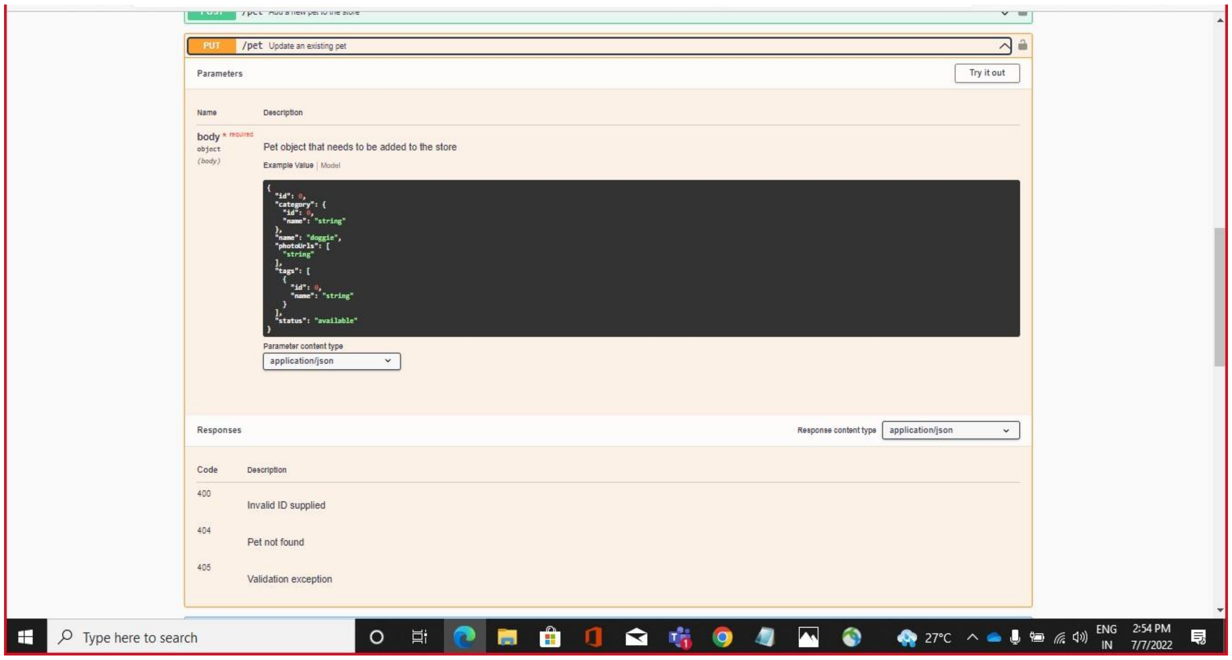
Example Value | Model

```
{
  "code": 0,
  "type": "string",
  "message": "string"
}
```

POST /pet Add a new pet to the store

PUT /pet Update an existing pet

PUT



Name	Description
body * required object *(body)*	Pet object that needs to be added to the store

Example Value | Model

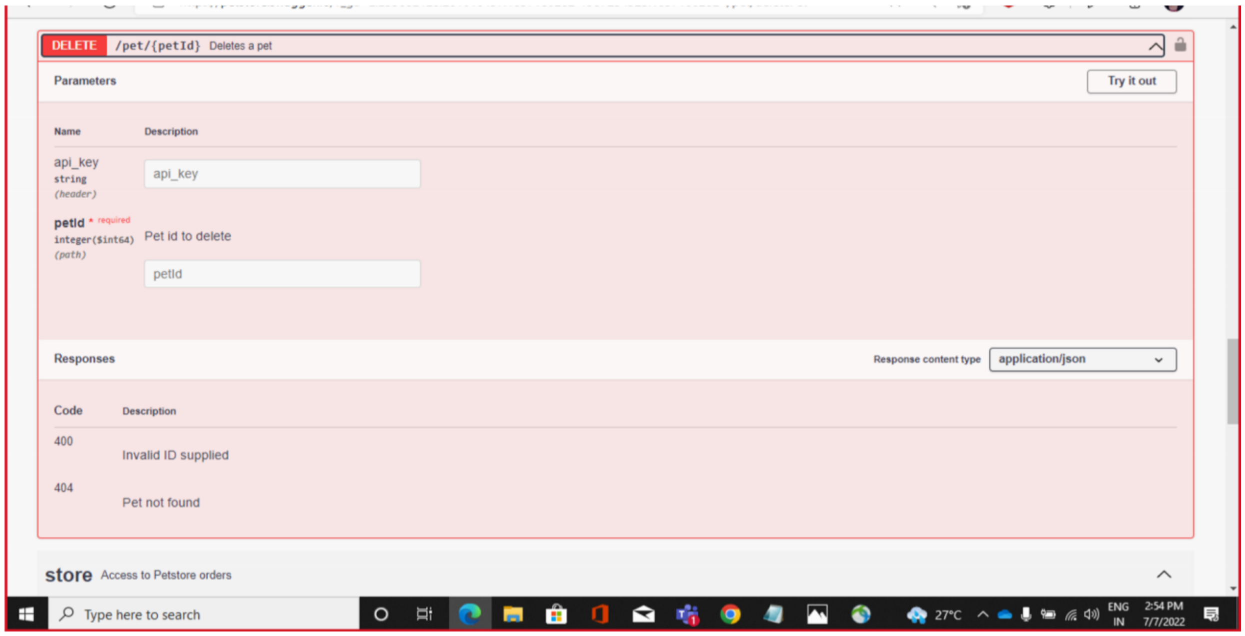
```
{
  "id": 0,
  "category": {
    "id": 0,
    "name": "string"
  },
  "name": "doggie",
  "photoUrls": [
    "string"
  ],
  "tags": [
    {
      "id": 0,
      "name": "string"
    }
  ],
  "status": "available"
}
```

Parameter content type: application/json

Responses — Response content type: application/json

Code	Description
400	Invalid ID supplied
404	Pet not found
405	Validation exception

DELETE



Name	Description
api_key string *(header)*	api_key
petId * required integer($int64) *(path)*	Pet id to delete petId

Responses — Response content type: application/json

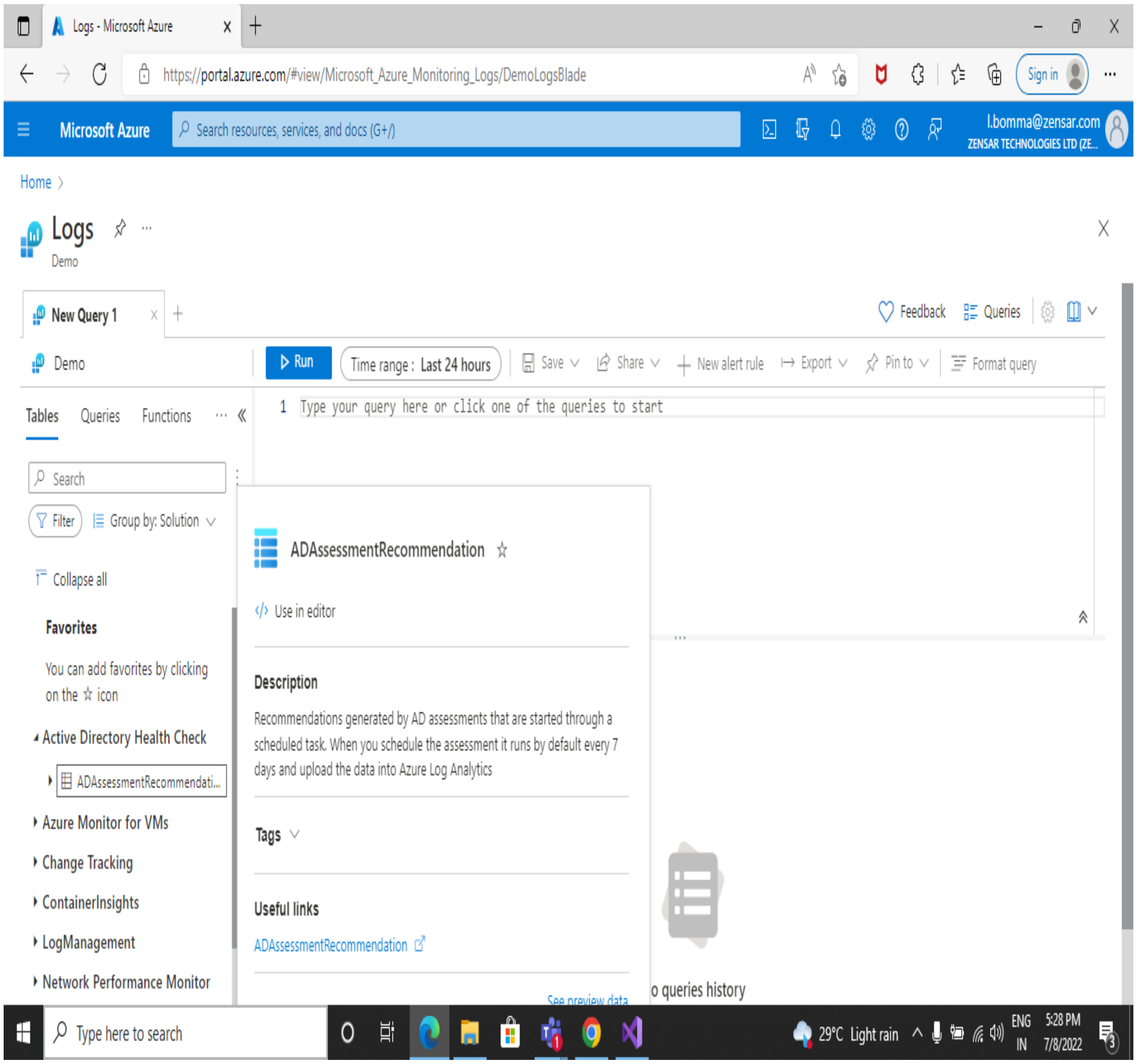
Code	Description
400	Invalid ID supplied
404	Pet not found

store Access to Petstore orders

6. Work with Log Analytics with the sample logs available

Log Analytics is a tool in the Azure portal to edit and run log queries from data collected by Azure Monitor logs and interactively analyze their results. You can use Log Analytics queries to retrieve records that match particular criteria, identify trends, analyze patterns, and provide various insights into your data.

- Select Logs from the Azure Monitor menu . This step sets the initial scope to a Log Analytics workspace so that your query selects from all data in that workspace

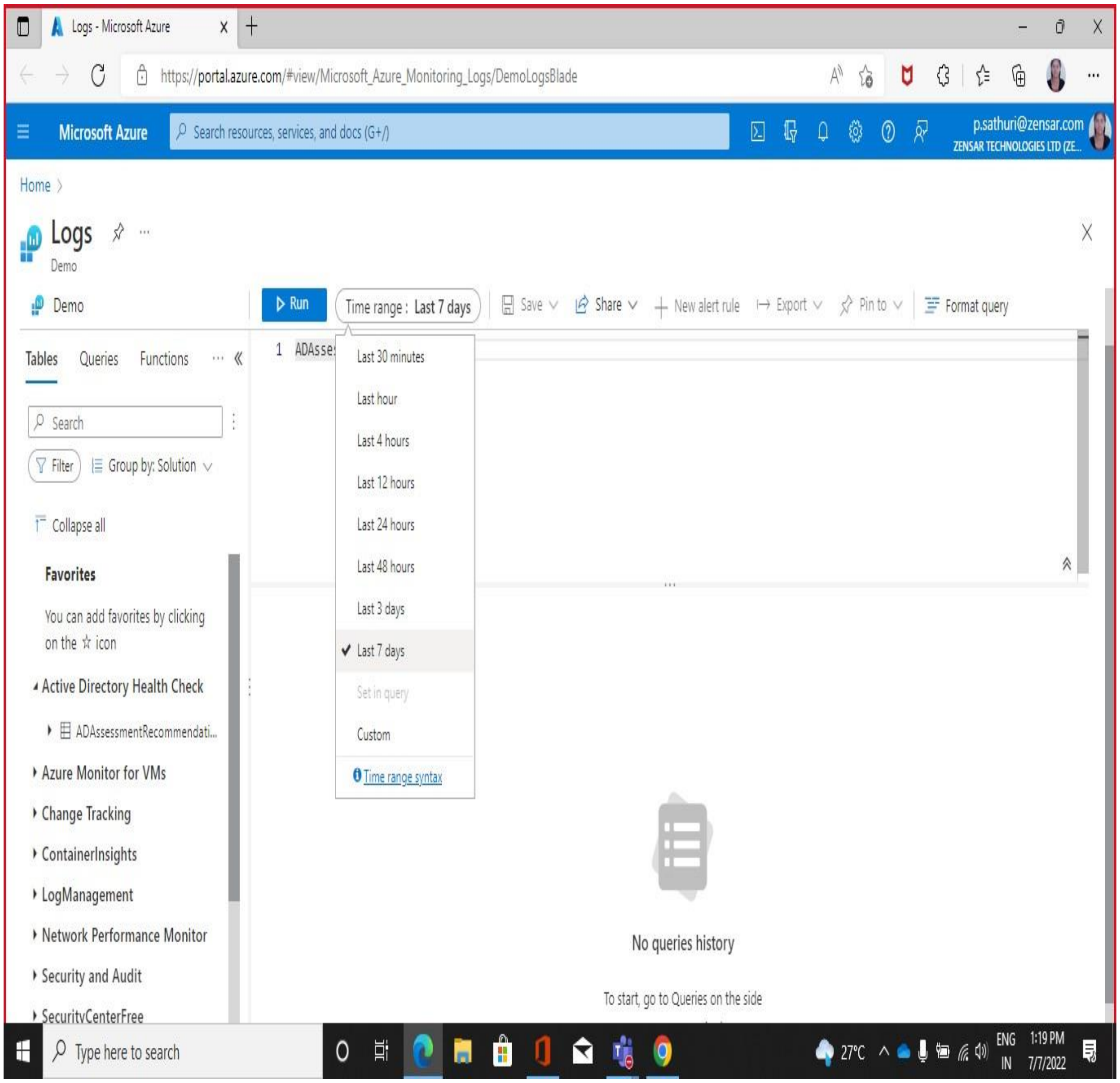


Type your query here or click one of the queries to start

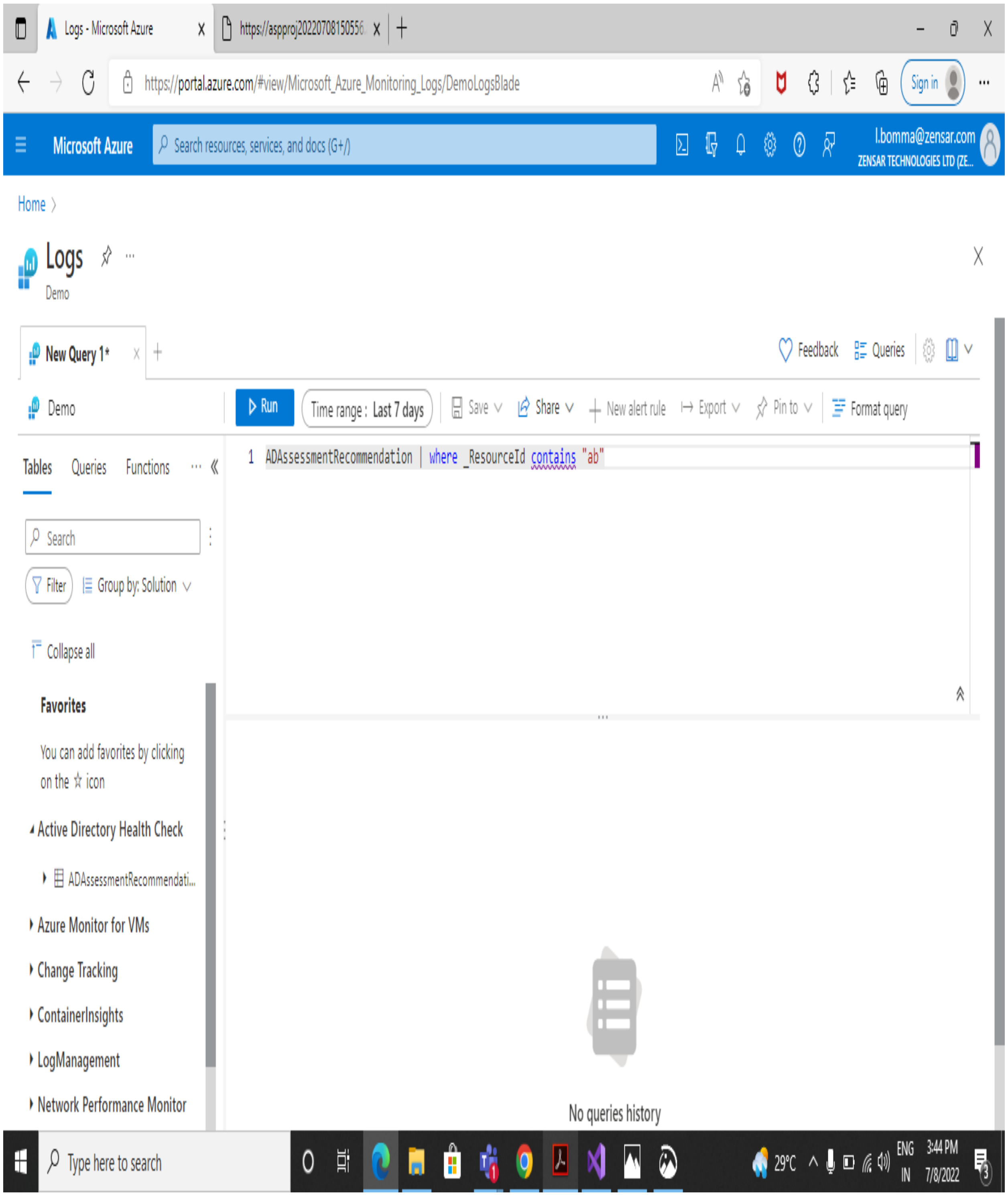
You can add favorites by clicking on the ☆ icon

Recommendations generated by AD assessments that are started through a scheduled task. When you schedule the assessment it runs by default every 7 days and upload the data into Azure Log Analytics

- All queries return records generated within a set time range. By default, the query returns records generated in the last 24 hours. You can set a different time range by using the where operator in the query. You can also use the Time range dropdown list at the top of the screen. Change the time range of the query by selecting Last 12 hours from the Time range dropdown. Select Run to return the results.

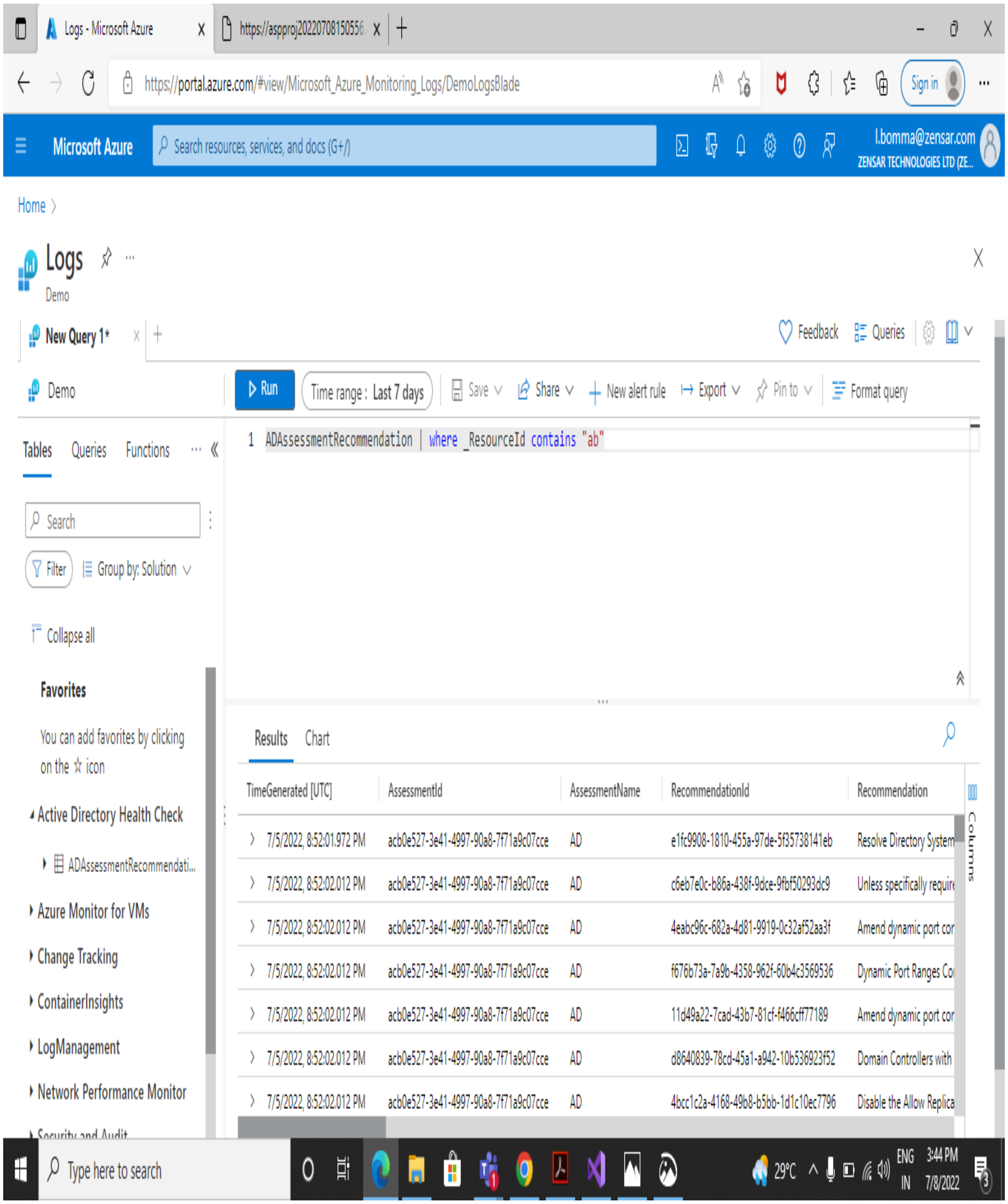


❖ This is the simplest query that we can write. It just returns all the records in a table. Run it by selecting the Run button or by selecting Shift+Enter with the cursor positioned anywhere in the query text.



```
ADAssessmentRecommendation | where _ResourceId contains "ab"
```

❖ Select Run to return the results.



```
ADAssessmentRecommendation | where _ResourceId contains "ab"
```

TimeGenerated [UTC]	AssessmentId	AssessmentName	RecommendationId	Recommendation
7/5/2022, 8:52:01.972 PM	acb0e527-3e41-4997-90a8-7f71a9c07cce	AD	e1fc9908-1810-455a-97de-5f35738141eb	Resolve Directory System
7/5/2022, 8:52:02.012 PM	acb0e527-3e41-4997-90a8-7f71a9c07cce	AD	c6eb7e0c-b86a-438f-9dce-9fbf50293dc9	Unless specifically require
7/5/2022, 8:52:02.012 PM	acb0e527-3e41-4997-90a8-7f71a9c07cce	AD	4eabc96c-682a-4d81-9919-0c32af52aa3f	Amend dynamic port cor
7/5/2022, 8:52:02.012 PM	acb0e527-3e41-4997-90a8-7f71a9c07cce	AD	f676b73a-7a9b-4358-962f-60b4c3569536	Dynamic Port Ranges Co
7/5/2022, 8:52:02.012 PM	acb0e527-3e41-4997-90a8-7f71a9c07cce	AD	11d49a22-7cad-43b7-81cf-f466cff77189	Amend dynamic port cor
7/5/2022, 8:52:02.012 PM	acb0e527-3e41-4997-90a8-7f71a9c07cce	AD	d8640839-78cd-45a1-a942-10b536923f52	Domain Controllers with
7/5/2022, 8:52:02.012 PM	acb0e527-3e41-4997-90a8-7f71a9c07cce	AD	4bcc1c2a-4168-49b8-b5bb-1d1c10ec7796	Disable the Allow Replica