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:

- Product ID
- ProductName
- Price
- Brand
- Manufacture Date
- Expiration Date

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. Create WEB API Mini Project.

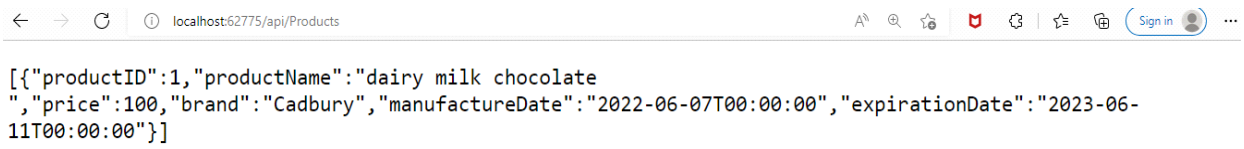
2. Add folder as models and in that folder add Class as Product.

3. Add Controller for Product class Using Entity Frame Work as Product.

4. Open package manager Consoler (Add-Migration ApiCreate and Update-Database)

5. Add Web Api Client html page as Product.html in the Project Using Ajax and JQuery.

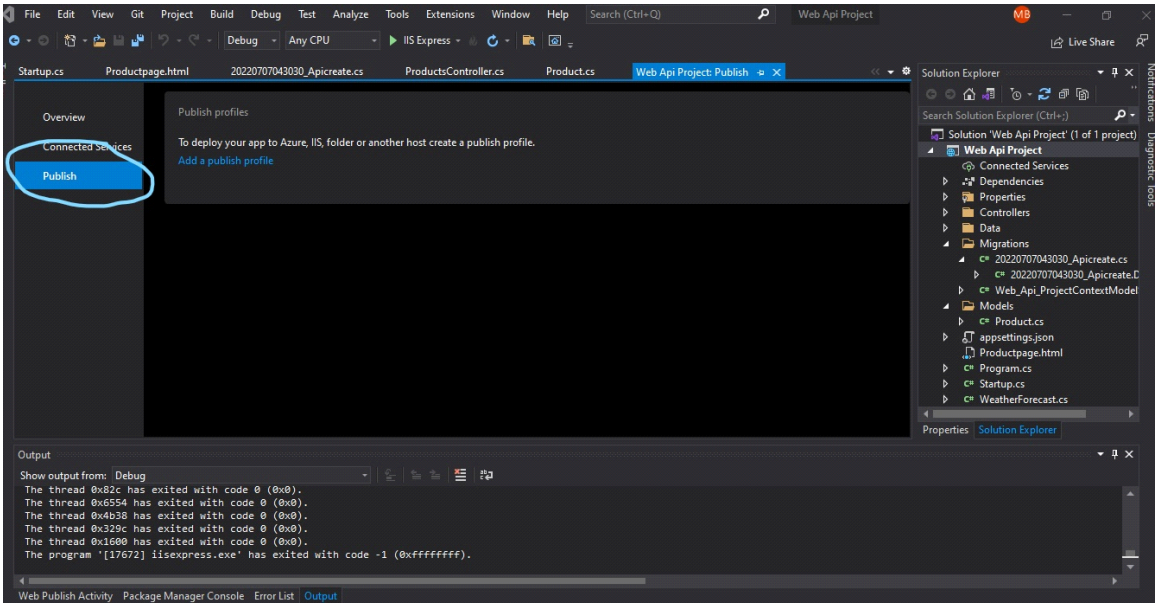
Ajax-JQuery solution

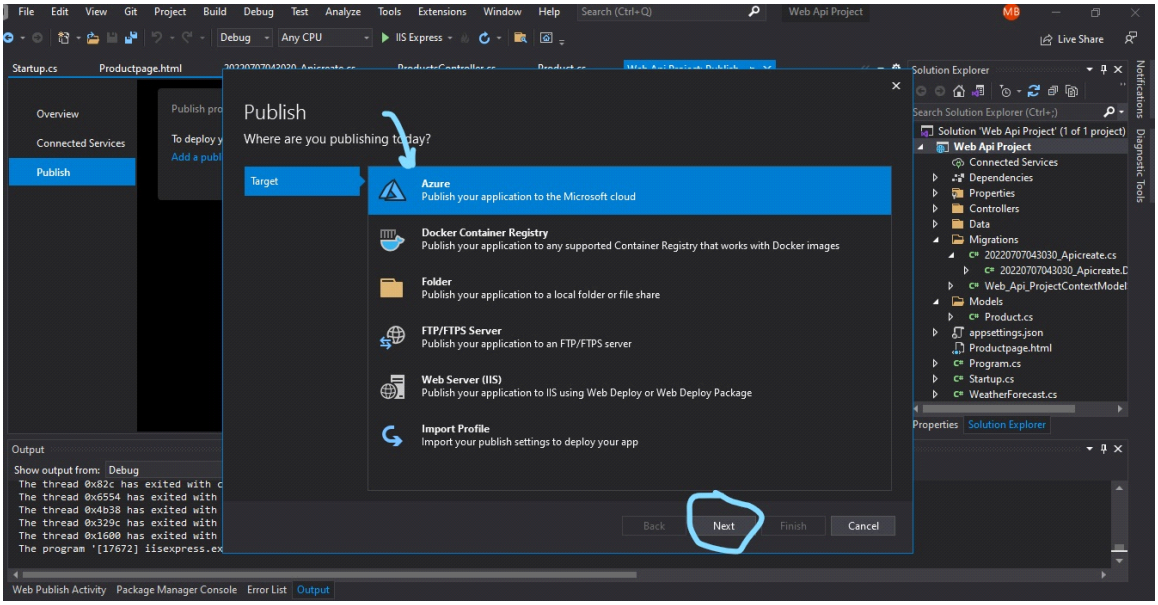


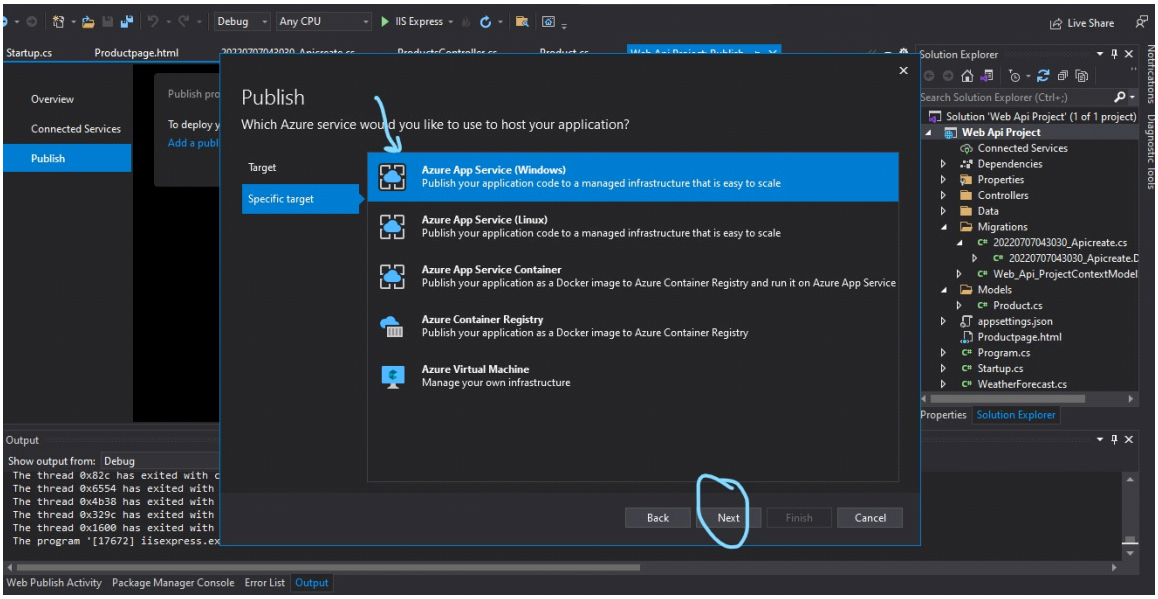
Azure Hosting:

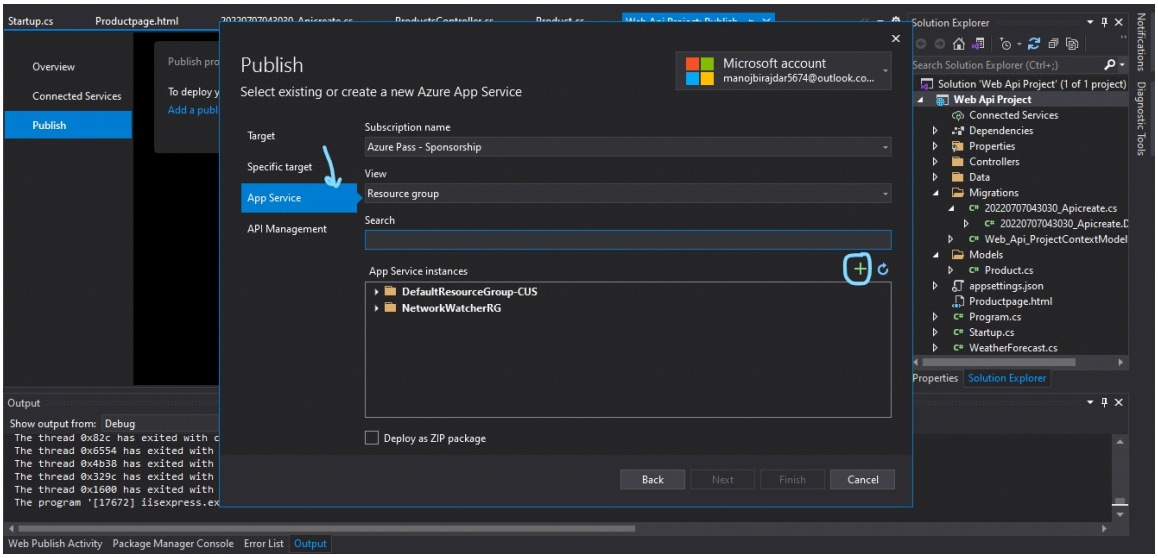
(Folling pics shows process of Azure Hosting)

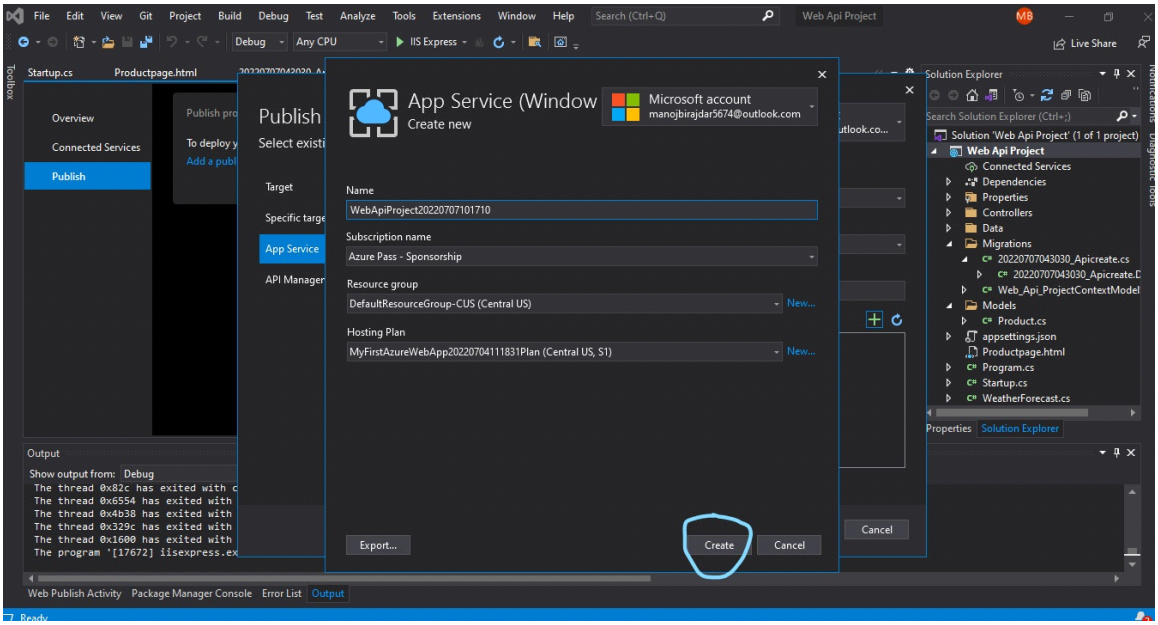
1.Host the web Api in azure and consume the same using jQuery Client:

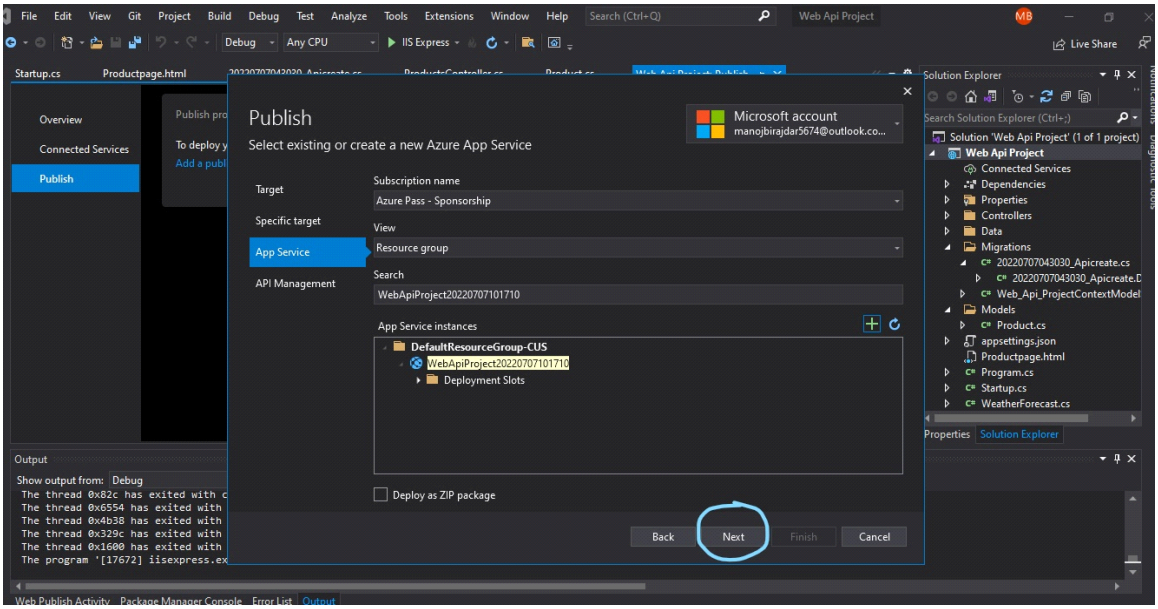

Publish profiles
To deploy your app to Azure, IIS, folder or another host create a publish profile.
Add a publish profile
Overview
Connected Services
Publish
Output
Show output from: Debug
The thread 0x82c has exited with code 0 (0x0).
The thread 0x6554 has exited with code 0 (0x0).
The thread 0x4b38 has exited with code 0 (0x0).
The thread 0x329c has exited with code 0 (0x0).
The thread 0x1600 has exited with code 0 (0x0).
The program '[17672] iisexpress.exe' has exited with code -1 (0xffffffff).

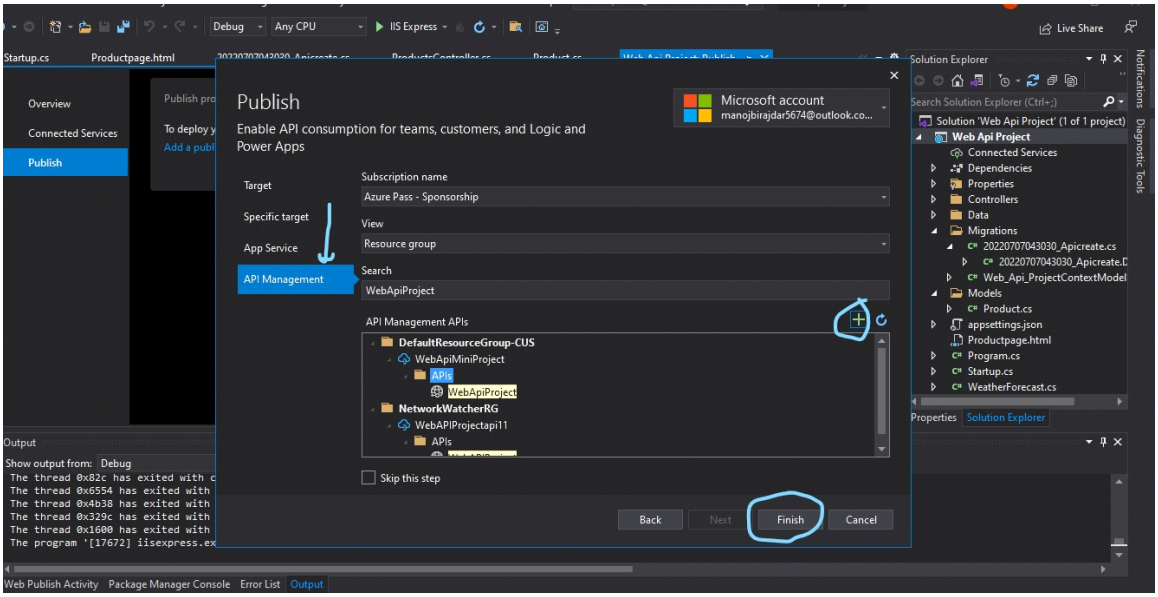

Publish
Where are you publishing today?
Target
Azure
Publish your application to the Microsoft cloud
Docker Container Registry
Publish your application to any supported Container Registry that works with Docker images
Folder
Publish your application to a local folder or file share
FTP/FTPS Server
Publish 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

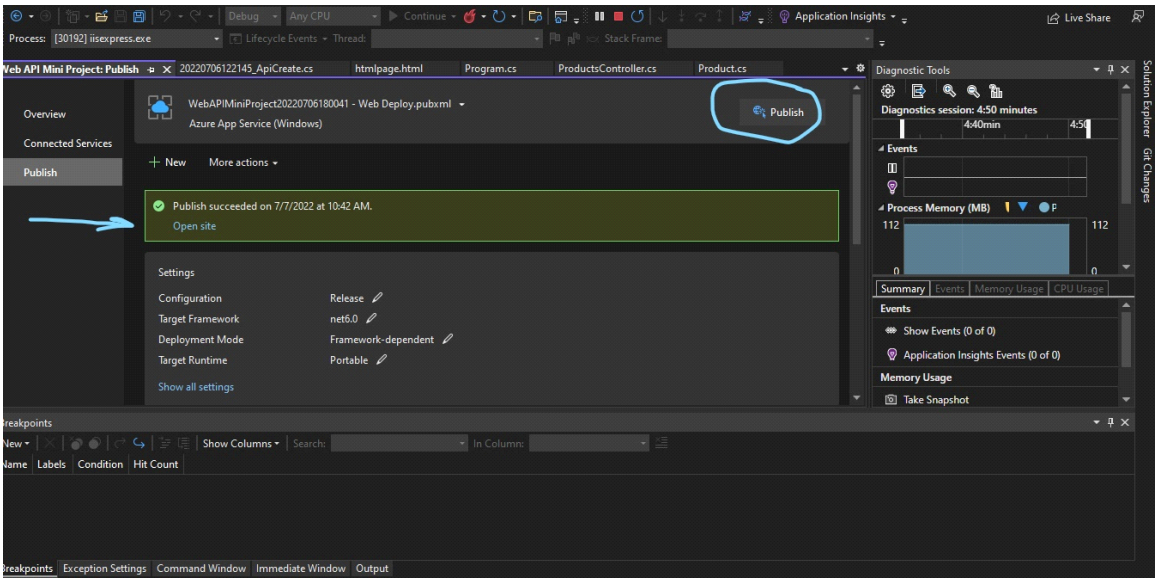

Publish
Which Azure service would you like to use to host your application?
Target
Specific target
Azure App Service (Windows)
Publish your application code to a managed infrastructure that is easy to scale
Azure App Service (Linux)
Publish your application code to a managed infrastructure that is easy 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
Back
Next
Finish
Cancel


Publish
Select existing or create a new Azure App Service
Microsoft account
Target
Specific target
App Service
API Management
Subscription name
Azure Pass - Sponsorship
View
Resource group
Search
App Service instances
DefaultResourceGroup-CUS
NetworkWatcherRG
Deploy as ZIP package
Back
Next
Finish
Cancel


App Service (Window
Create new
Microsoft account
manojbirajdar5674@outlook.com
Name
WebApiProject20220707101710
Subscription name
Azure Pass - Sponsorship
Resource group
DefaultResourceGroup-CUS (Central US)
New...
Hosting Plan
MyFirstAzureWebApp20220704111831Plan (Central US, S1)
New...
Export...
Create
Cancel
Publish
Target
Specific target
App Service
API Manager
Overview
Connected Services
Publish
Solution Explorer
Solution 'Web Api Project' (1 of 1 project)
Web Api Project
Output
Web Publish Activity
Package Manager Console
Error List
Output
Ready

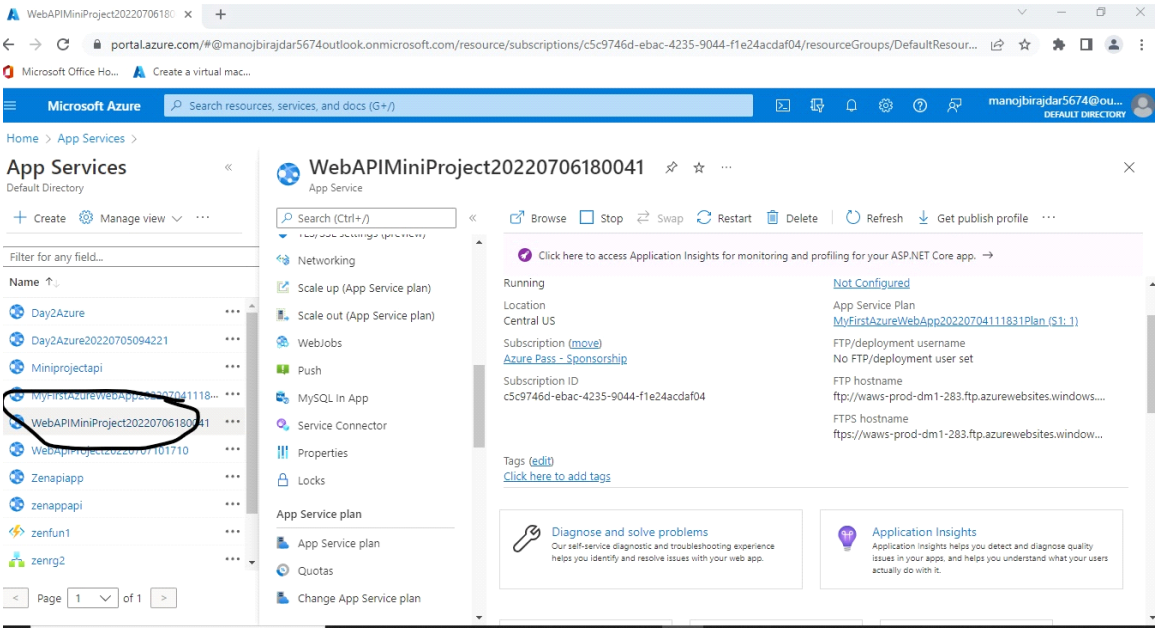

Publish
Select existing or create a new Azure App Service
Microsoft account
manojbirajdar5674@outlook.co...
Target
Specific target
App Service
API Management
Subscription name
Azure Pass - Sponsorship
View
Resource group
Search
WebApiProject20220707101710
App Service instances
DefaultResourceGroup-CUS
WebApiProject20220707101710
Deployment Slots
Deploy as ZIP package
Back
Next
Finish
Cancel
Solution Explorer
Solution 'Web Api Project' (1 of 1 project)
Web Api Project
Output
Web Publish Activity
Package Manager Console
Error List
Output



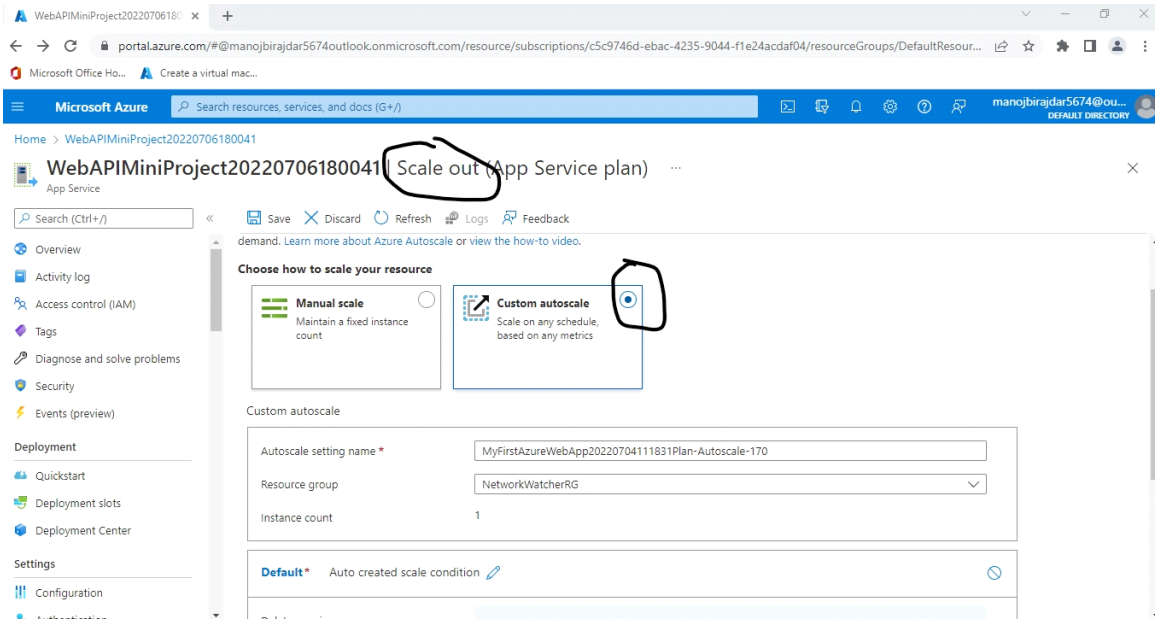


2.Configure Scale out by adding rules for custom scaling :

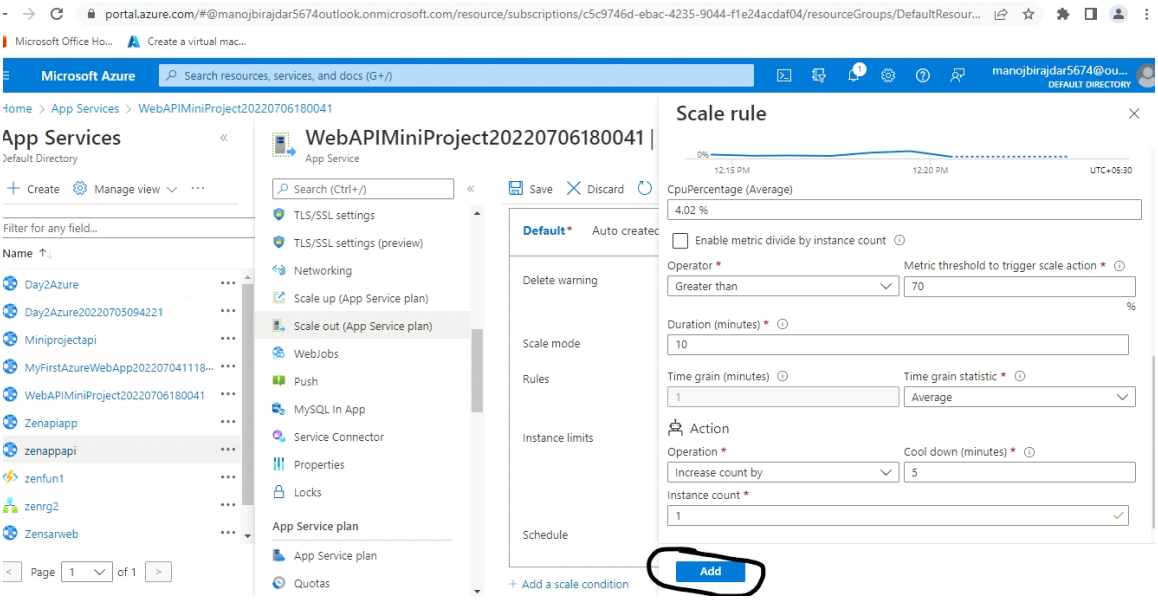
A scale out operation is the equivalent of creating multiple copies of your web site and adding a load balancer to distribute the demand between them. When you scale out a web site in Windows Azure Web Sites there is no need to configure load balancing separately since this is already provided by the platform.



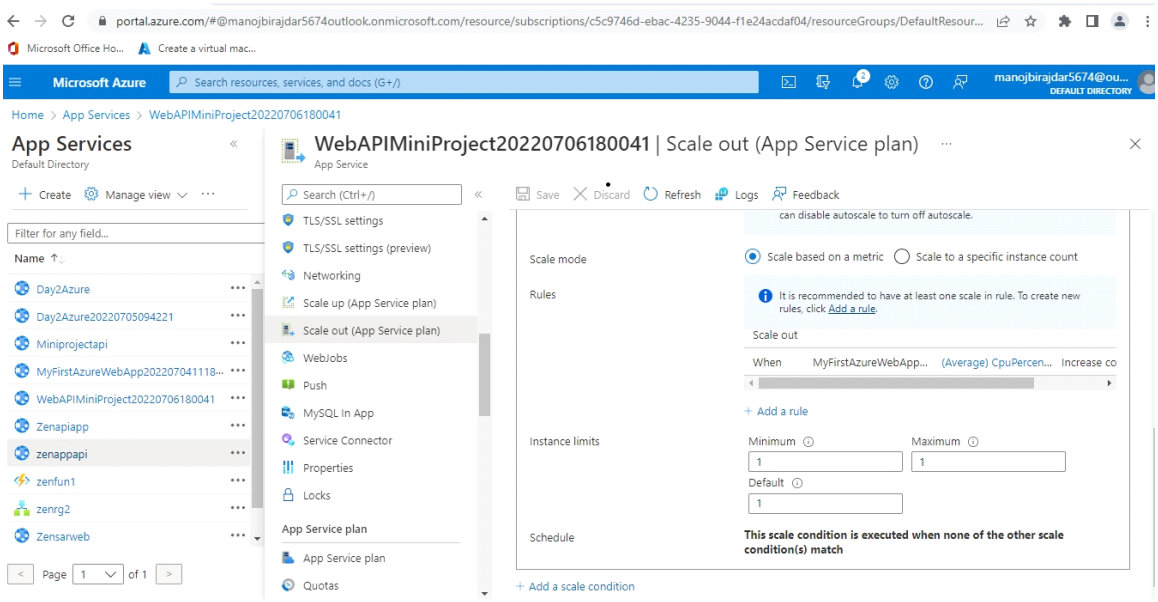
2.1 select Custom Auto Scale:



2.1 Click on Add a rule.



2.3 Add your rules and click on Add button and Save changes.



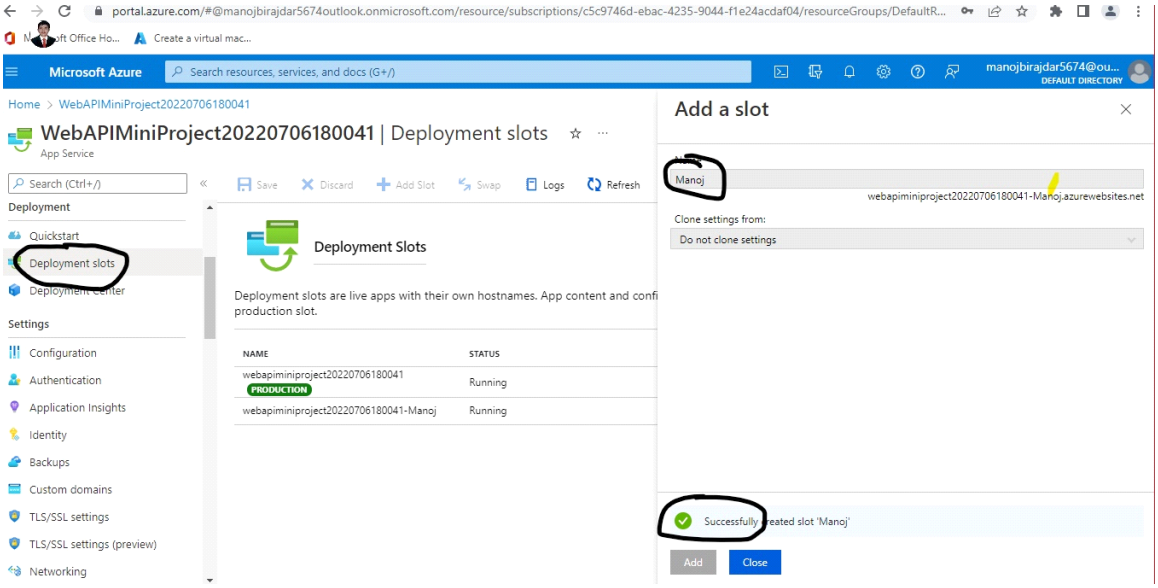
3.Configure Deployment Slots for Staging and Production :

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

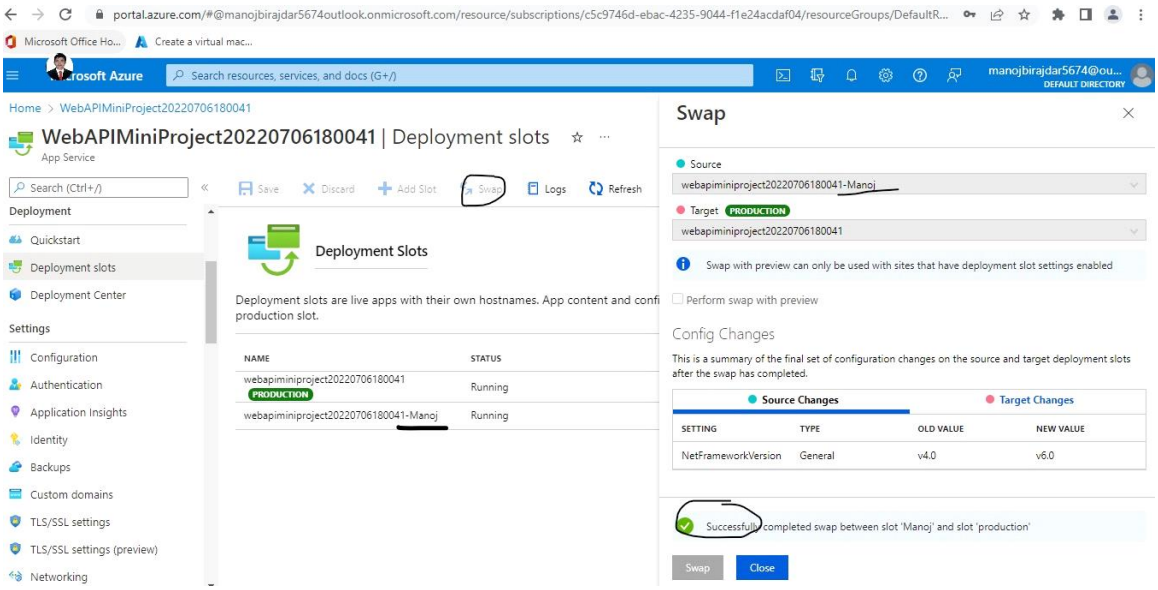
plan may have multiple slots, while under the Consumption plan only one slot is allowed.

3.1 Click on Add Slot.

3.2 Enter The slot name and click on add button.



3. Finally swapping the slots.



4.Configure Application Insights for the project :

Application Insights is a feature of Azure Monitor that provides extensible application performance management (APM) and monitoring for live web apps. Developers and DevOps professionals can use Application Insights to:

-Automatically detect performance anomalies.

-Help diagnose issues by using powerful analytics tools.

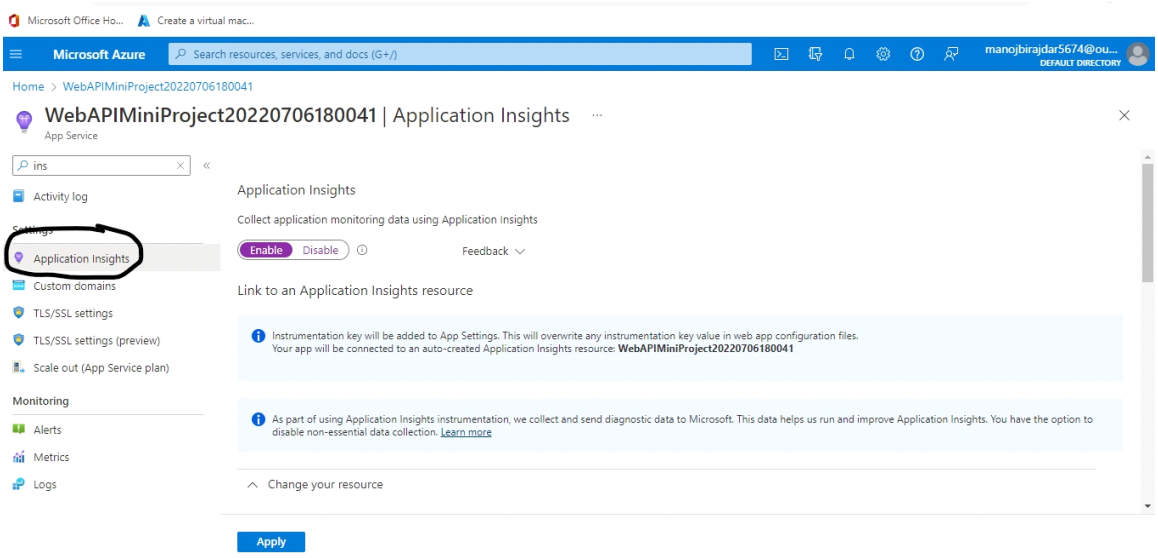
-See what users actually do with apps.

-Help continuously improve app performance and usability.

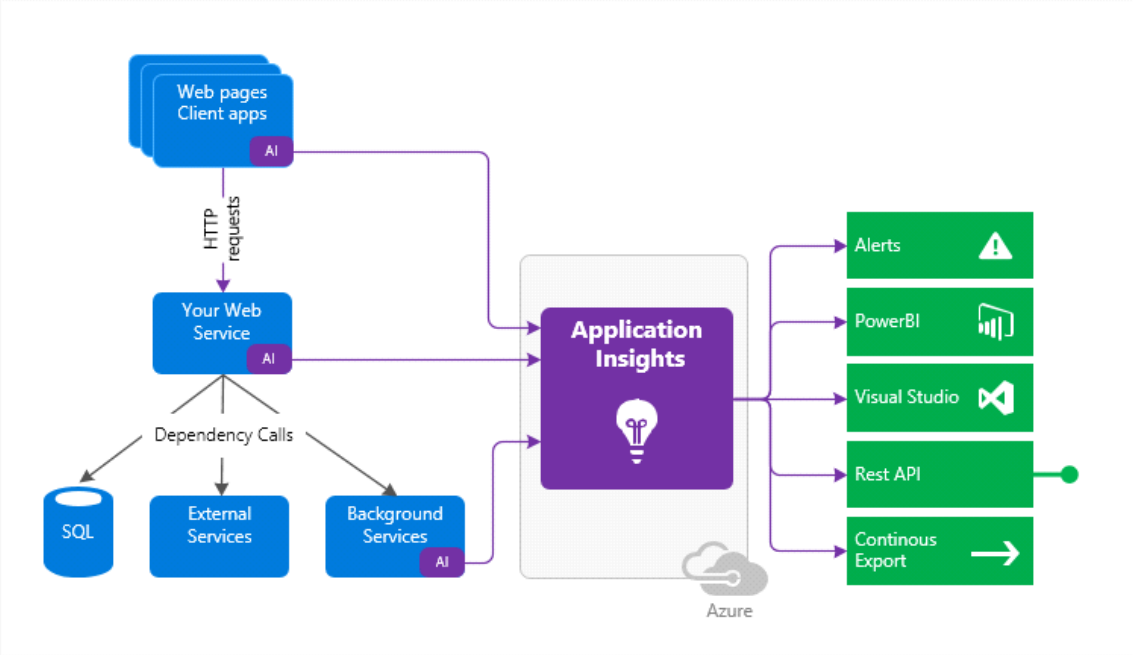
4.1 Trun on Application Insights.

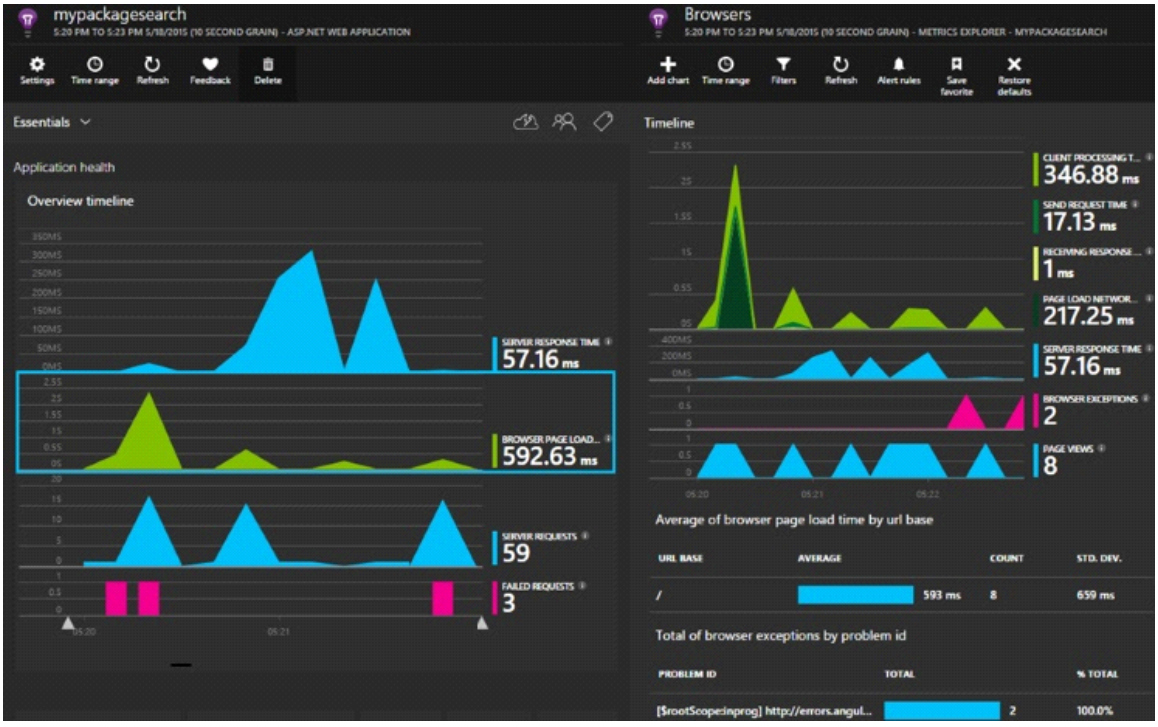
4.2 Select Existing Resource.

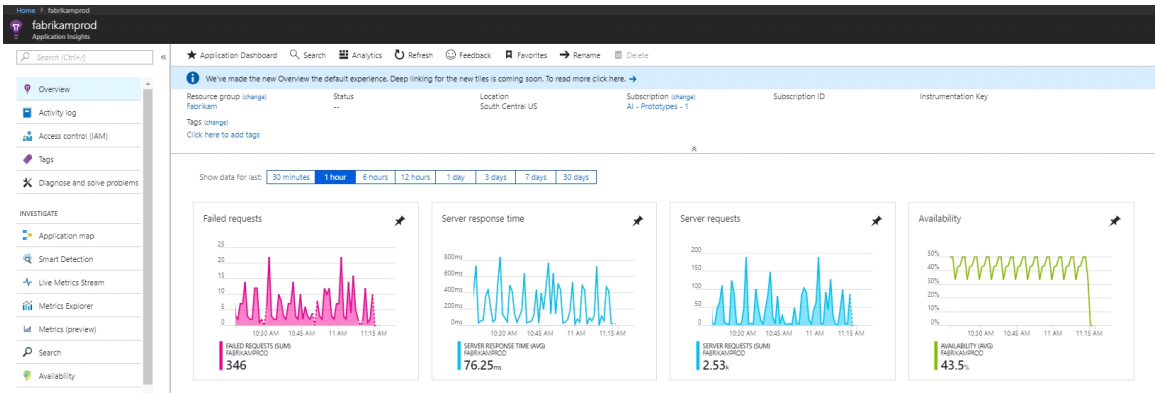
4.3 Click on the Apply button.



Application Insights works:


Web pages
Client apps
AI
HTTP requests
Your Web Service
AI
Dependency Calls
SQL
External Services
Background Services
AI
Application Insights
Azure
Alerts
PowerBI
Visual Studio
Rest API
Continous Export


mypackagesearch
Settings
Time range
Refresh
Feedback
Delete
Essentials
Application health
Overview timeline
SERVER RESPONSE TIME
57.16 ms
BROWSER PAGE LOAD...
592.63 ms
SERVER REQUESTS
59
FAILED REQUESTS
3
Browsers
Add chart
Time range
Filters
Refresh
Alert rules
Save favorite
Restore defaults
Timeline
CLIENT PROCESSING T...
346.88 ms
SEND REQUEST TIME
17.13 ms
RECEIVING RESPONSE ...
1 ms
PAGE LOAD NETWOR...
217.25 ms
SERVER RESPONSE TIME
57.16 ms
BROWSER EXCEPTIONS
2
PAGE VIEWS
8
Average of browser page load time by url base
URL BASE
AVERAGE
COUNT
STD. DEV.
/
593 ms
8
659 ms
Total of browser exceptions by problem id
PROBLEM ID
TOTAL
% TOTAL
[$rootScope:inprog] http://errors.angul...
2
100.0%



5.Configure Swagger for the Api :

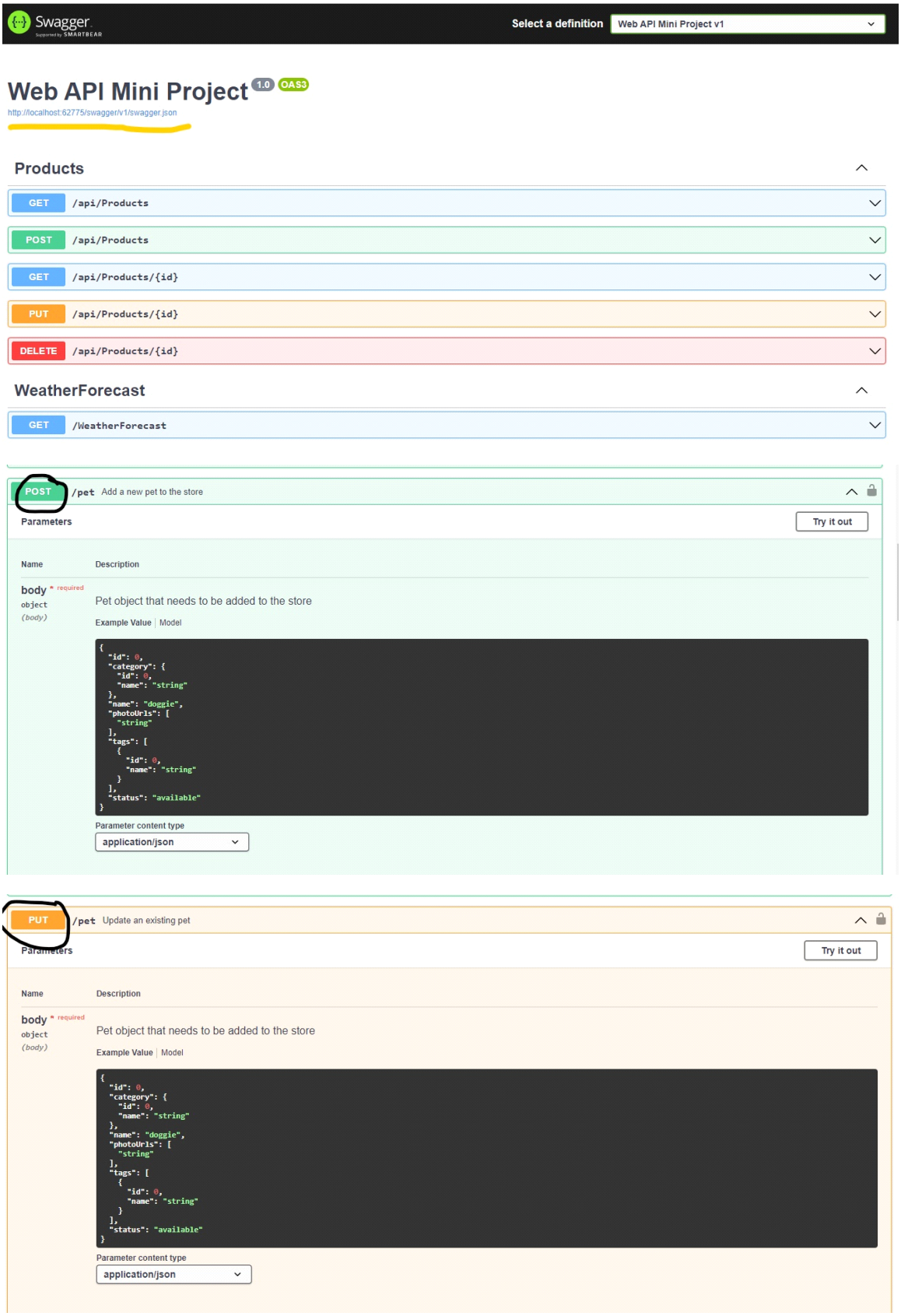
Swagger UI allows anyone be it your development team or your end consumers to visualize and interact with the API's resources without having any of the implementation logic in place. It's automatically generated from your Open API (formerly known as Swagger) Specification, with the visual documentation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

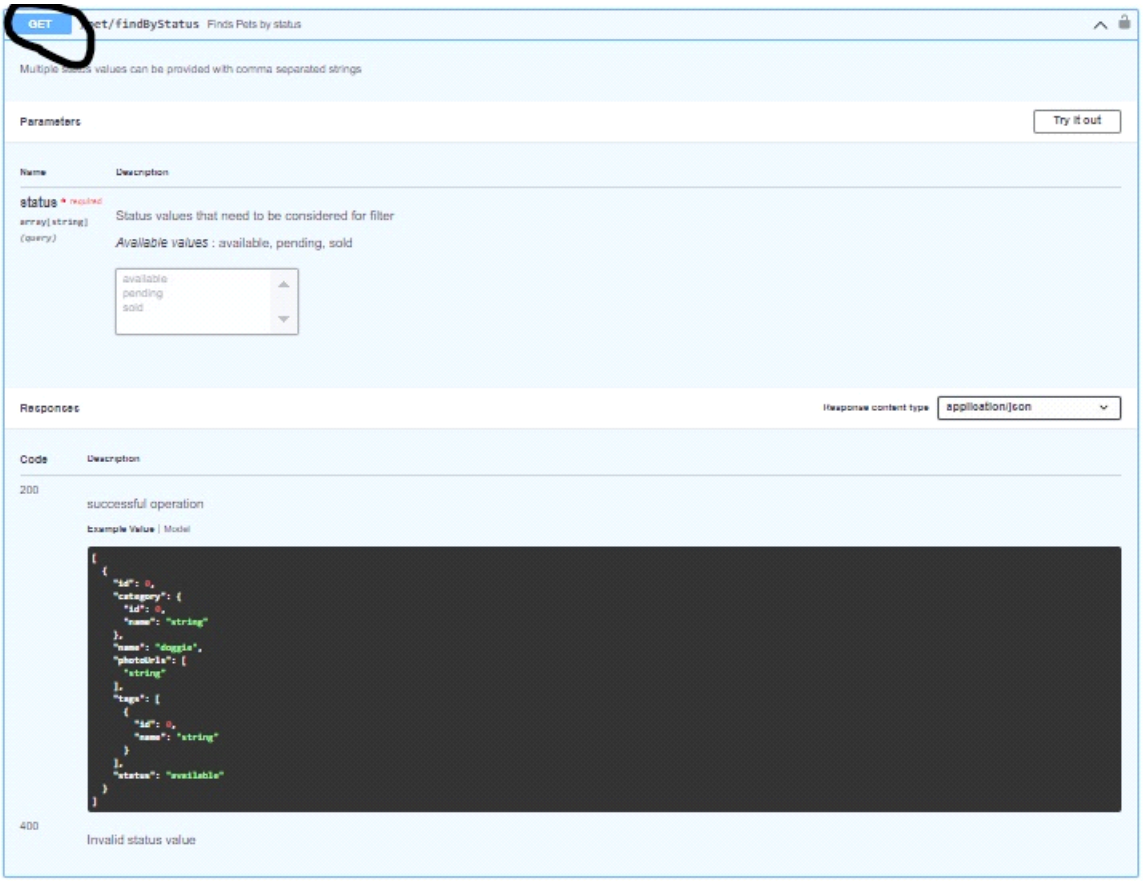
Advantages of Swagger Api :

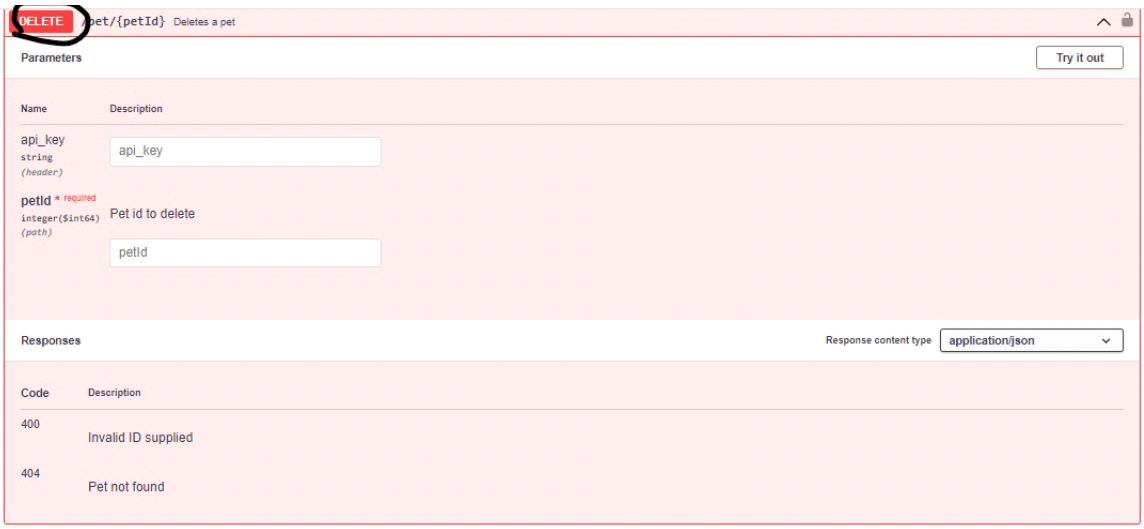
- Synchronizes the API documentation with the server and client at the same pace.

- Allows us to generate REST API documentation and interact with the REST API. The interaction with the REST API using the Swagger UI Framework gives clear insight into how the API responds to parameters.

-Provides responses in the format of JSON and XML.


Swagger
Select a definition
Web API Mini Project v1
Web API Mini Project 1.0 OAS3
http://localhost:62775/swagger/v1/swagger.json
Products
GET /api/Products
POST /api/Products
GET /api/Products/{id}
PUT /api/Products/{id}
DELETE /api/Products/{id}
WeatherForecast
GET /WeatherForecast
POST /pet Add a new pet to the store
Parameters
Try it out
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
PUT /pet Update an existing pet
Parameters
Try it out
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





6.Work with Log Analytics with the sample logs available :

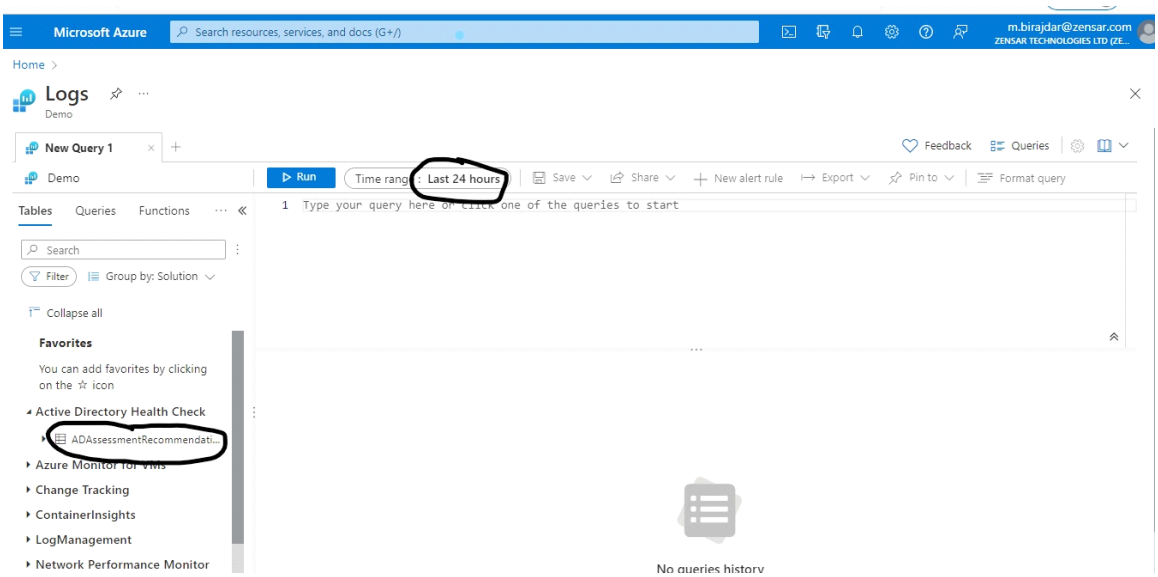
Log Analytics is a tool in the Azure portal to edit and run log queries from data collected by Azure Monitor logs and interactively analyze their results. You can use Log Analytics

queries to retrieve records that match particular criteria, identify trends, analyze patterns, and provide various insights into your data

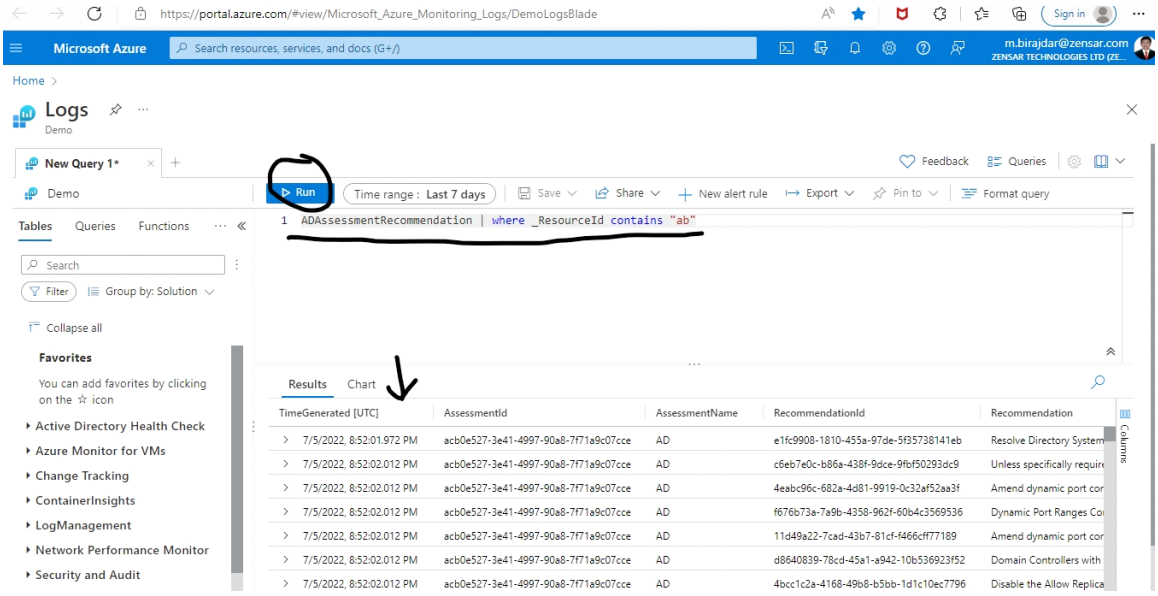
If you start Log Analytics from the Azure Monitor menu or the Log Analytics workspaces menu, you'll have access to all the records in a workspace. If you select Logs from another type of resource, your data will be limited to log data for that resource. For more information, see Log query scope and time range in Azure Monitor Log Analytics.

6.1 Open Azure portal and click on Logs.

6.2 Select the Tables what you check.



6.3 Write a query logic and click on Run.


https://portal.azure.com/#view/Microsoft_Azure_Monitoring_Logs/DemoLogsBlade
Microsoft Azure
Search resources, services, and docs (G+/)
m.birajdar@zensar.com
ZENSAR TECHNOLOGIES LTD (ZE...
Home
Logs
Demo
New Query 1*
Feedback
Queries
Demo
Run
Time range : Last 7 days
Save
Share
New alert rule
Export
Pin to
Format query
Tables
Queries
Functions
Search
Filter
Group by: Solution
Collapse all
Favorites
You can add favorites by clicking on the ☆ icon
Active Directory Health Check
Azure Monitor for VMs
Change Tracking
ContainerInsights
LogManagement
Network Performance Monitor
Security and Audit
1 ADAssessmentRecommendation | where _ResourceId contains "ab"
Results
Chart
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
Columns