Capstone Project

Integrating Existing app with Azure services

By
Vishwanath Sethi
Padmini Rani
Sandhya Babu
Asha Latha

Problem statement:

Tailwind Traders is looking to integrate their E-commerce website QuickKart and database with Azure. The database, which follows strict schema, holds the product catalog, and all online orders. The website experience slow response during peak hours, so, there is a need to implement auto scaling for the website. There is a need to automate the application development, when there are changes in code base. Client is looking for CI/CD pipeline. Client is looking for microservices preferably PAAS . Client also looking for image storage and need to reduce cost if no one is accessing that image . Wherever possible reduce the cost and admin overhead. Also, the app needs to be monitored using proper Insights The solution also needs to be secure and store confidential data.

Resources:

- Front end : Angular
- Backend : .net
- Database : Azure SQL
- Web host platform: Azure Web App with scaling and deployment slots
- Microservices: Azure Function App
- Application security: Azure Key vault
- CI/CD: GitHub/ Azure DevOps
- Monitoring Solution: Azure Application Insights
- Cloud Storage: Azure Blob Storage

Migrating Tailwind Traders(On premises web application to Azure Cloud(app service)

Introduction:

Tailwind Traders: Leading E-commerce company

Challenges:

- ❏ Slow response during peak hours
- ❏ Manual application development process
- ❏ High admin overhead
- ❏ Lack of scalability and automation
- ❏ Lack of Security

Solution: Migrate to Azure Cloud for scalability, automation, and cost optimization

Key resources for cloud migration:

- ★ Web host platform: Azure Web App with scaling and deployment slots
- ★ Database Migration : Migrating existing database schema to Azure SQL database
- ★ Scalability : Autoscaling
- ★ Application development: Azure Automation
- ★ CI/CD: Azure DevOps
- ★ Microservices Architecture - Azure Functions
- ★ Image Storage and Cost Optimization - Blob storage
- ★ Monitoring and Insights - Azure Monitor & Application Insights
- ★ Application Security - Azure Key Vault

Key Features of the Azure resources:

Azure Web App with scaling and deployment slots:::

- Fully managed platform for web apps
- Autoscaling based on traffic patterns
- Deployment slots for staging and production

Database Migration: Azure SQL Database

- Managed relational database service
- Secure and Scalable(Supports encryption, Advanced threat protection, Identity and access management)
- High Availability & Disaster recovery(automated backups, geo-replication, and automatic failover)
- SQL databases enforce strict schema through the use of Data Definition Language (DDL) commands.
- Integration with Azure Services(Connection String provides a secure & encrypted connection)

Auto Scaling - Horizontal scaling:::

- ❑ Increased performance
- ❑ Fault Tolerance
- ❑ Cost Effective
- ❑ Easy Maintenance

Application development: Azure Automation:::

- ❑ Set up CI/CD: pipeline should trigger whenever there are changes in the code base)
- ❑ Trigger azure automation runbooks
- ❑ Maintain desired state configuration with the help of Dsc extension.

CI/CD: Azure DevOps

- ❑ It supports Version control
- ❑ Azure Pipelines supports approvals, gates and checks
- ❑ It automates build, test and deployment process
- ❑ Azure DevOps offers extensive integration capabilities with third-party tools and services through APIs, webhooks, and extensions.

Microservices Architecture - Azure Functions:

- ❑ Serverless compute for microservices(Code compiles whenever an action triggers)
- ❑ Pay as you go model (Flexible service plan)
- ❑ Continuous Integration
- ❑ Supports multiple programming languages(C#, Java Script, Java & python)
- ❑ We can monitor it using Azure Application Insights and manage them using REST API and visual studio.
- ❑ We can run locally or in the cloud

Image Storage and Cost Optimization - Blob storage:

- Cost effectiveness
- Global reach (Low latency)
- Security and durability(multiple copies replicated across different data centres)
- Life cycle management Policies(automate data retention and deletion based on predefined rules)
- Scalability(Designed to handle massive amount of data)
- Azure Blob Storage integrates seamlessly with other Azure services(Data Lake storage, Data Factory)

Monitoring and Insights - Azure Monitor & Application Insights:

- ❑ Captures Real time telemetry data
- ❑ Dashboards and Visualization(customizable dashboards and visualization tools to create interactive charts, graphs and Heatmaps)
- ❑ Azure Monitor allows you to set up alerts based on metrics, logs, or events
- ❑ Identify bottlenecks, optimize performance
- ❑ Azure Monitor offers autoscale capabilities(dynamically adjust the capacity of Azure resources based on demand and usage patterns).
- ❑ Network Performance Monitor

Application Security - Azure Key Vault

- Secure Storage: Azure Key Vault (secure and centralized location to store keys, secrets, and certificates)
- Encryption(encrypt keys, secrets, and certificates)
- Integration: Seamlessly integrates with other Azure services (Azure Virtual Machines, Azure Functions, Azure App Service)
- Geographic Redundancy and Availability(high availability and disaster recovery by replicating data across multiple Azure regions)
- Key Rotation and Expiry(supports automatic key rotation and expiry policies to enhance security and compliance)
- Auditing and Logging: Monitor access and changes for compliance and security

Reduce Cost and admin overhead:

❑ Optimize resource allocation.

❑ Use PaaS services.(

❑ Implement serverless.(Azure Function app)

❑ Leverage caching and CDN. (Azure CDN)

❑ Monitor and optimize spending.

❑ Adopt DevOps practices.

❑ Utilize Azure PaaS offerings.

❑ Prioritize security.

❑ Optimize licensing.

❑ Explore cost-saving programs.

TASK-1:

Agenda:

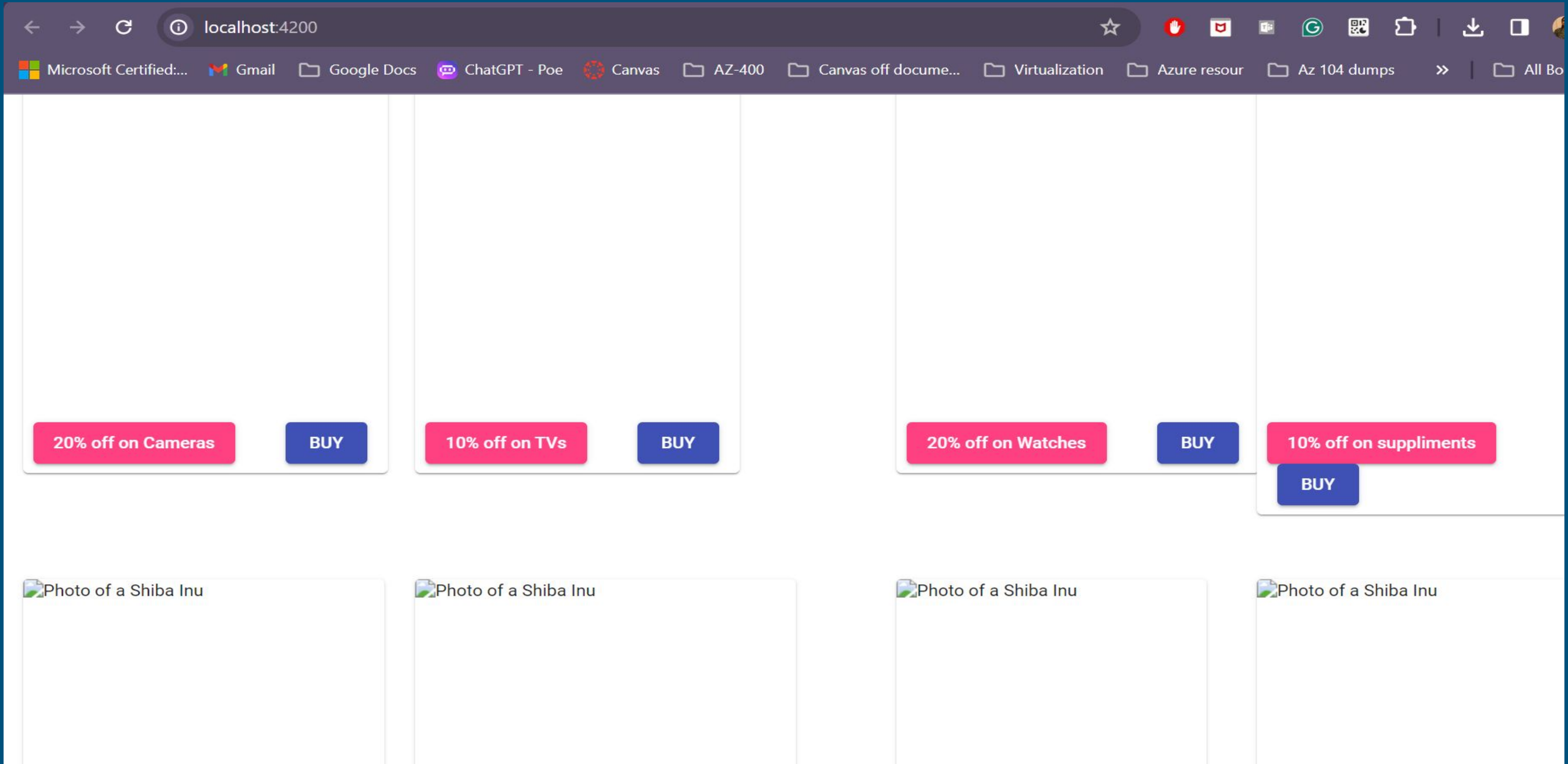
- □ Set up the Local environment
- □ Create SQL Database and Server(Optional) QuickKart-DB.sql
- □ Execute the Scripts (Optional)
- □ Test the code locally first.

For more task 1 info, click the source Document URL:
https://docs.google.com/document/d/1CxDMdt3CsTmjfYtD7aXIuXn13CRTJJ01/edit

Software installations for this project:

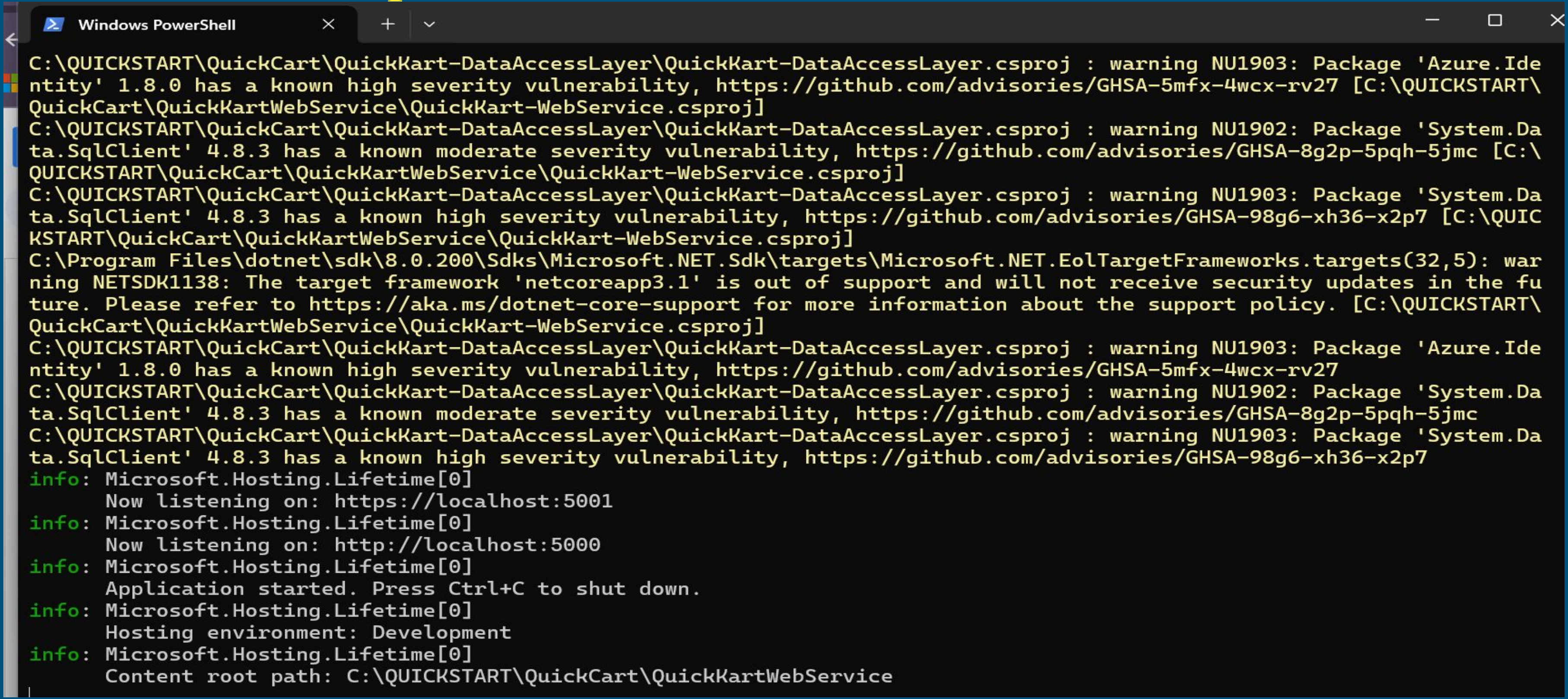
Visual Studio Community 2022 (+ Azure Development Kit)
Install .Net core 3.1
Visual Studio code
Git
Node.js
Angular CLI -Run (npm install -g @angular/cli@13.3) in cmd prompt

FrontEnd Output:



Backend Pulling images from the database

Backend Output:



```
C:\QUICKSTART\QuickCart\QuickKart-DataAccessLayer\QuickKart-DataAccessLayer.csproj : warning NU1903: Package 'Azure.Ide
ntity' 1.8.0 has a known high severity vulnerability, https://github.com/advisories/GHSA-5mfx-4wcx-rv27 [C:\QUICKSTART\
QuickCart\QuickKartWebService\QuickKart-WebService.csproj]
C:\QUICKSTART\QuickCart\QuickKart-DataAccessLayer\QuickKart-DataAccessLayer.csproj : warning NU1902: Package 'System.Da
ta.SqlClient' 4.8.3 has a known moderate severity vulnerability, https://github.com/advisories/GHSA-8g2p-5pqh-5jmc [C:\
QUICKSTART\QuickCart\QuickKartWebService\QuickKart-WebService.csproj]
C:\QUICKSTART\QuickCart\QuickKart-DataAccessLayer\QuickKart-DataAccessLayer.csproj : warning NU1903: Package 'System.Da
ta.SqlClient' 4.8.3 has a known high severity vulnerability, https://github.com/advisories/GHSA-98g6-xh36-x2p7 [C:\QUIC
KSTART\QuickCart\QuickKartWebService\QuickKart-WebService.csproj]
C:\Program Files\dotnet\sdk\8.0.200\Sdks\Microsoft.NET.Sdk\targets\Microsoft.NET.EolTargetFrameworks.targets(32,5): war
ning NETSDK1138: The target framework 'netcoreapp3.1' is out of support and will not receive security updates in the fu
ture. Please refer to https://aka.ms/dotnet-core-support for more information about the support policy. [C:\QUICKSTART\
QuickCart\QuickKartWebService\QuickKart-WebService.csproj]
C:\QUICKSTART\QuickCart\QuickKart-DataAccessLayer\QuickKart-DataAccessLayer.csproj : warning NU1903: Package 'Azure.Ide
ntity' 1.8.0 has a known high severity vulnerability, https://github.com/advisories/GHSA-5mfx-4wcx-rv27
C:\QUICKSTART\QuickCart\QuickKart-DataAccessLayer\QuickKart-DataAccessLayer.csproj : warning NU1902: Package 'System.Da
ta.SqlClient' 4.8.3 has a known moderate severity vulnerability, https://github.com/advisories/GHSA-8g2p-5pqh-5jmc
C:\QUICKSTART\QuickCart\QuickKart-DataAccessLayer\QuickKart-DataAccessLayer.csproj : warning NU1903: Package 'System.Da
ta.SqlClient' 4.8.3 has a known high severity vulnerability, https://github.com/advisories/GHSA-98g6-xh36-x2p7
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\QUICKSTART\QuickCart\QuickKartWebService
```

TASK-2:

Goal:

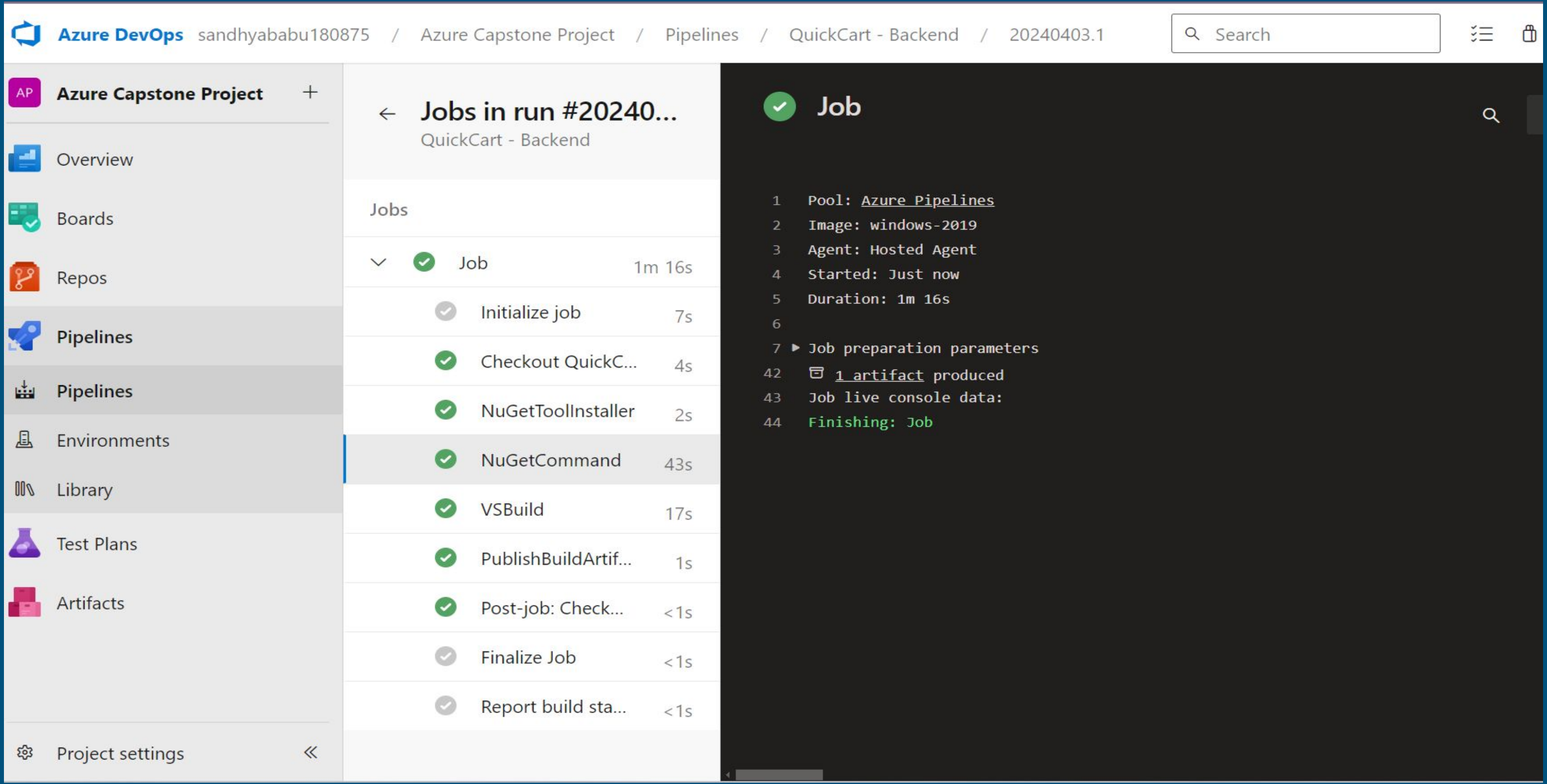
Learn to build CI Pipeline in Azure Devops using YAML.

- ❑ MS Hosted
- ❑ Self Hosted

For more task 2 info, click the source Document URL:

https://docs.google.com/document/d/1we9ylYtR6sDRxgDQNF7p0dxvILNOGg3e/edit

Build CI with MS Hosted:



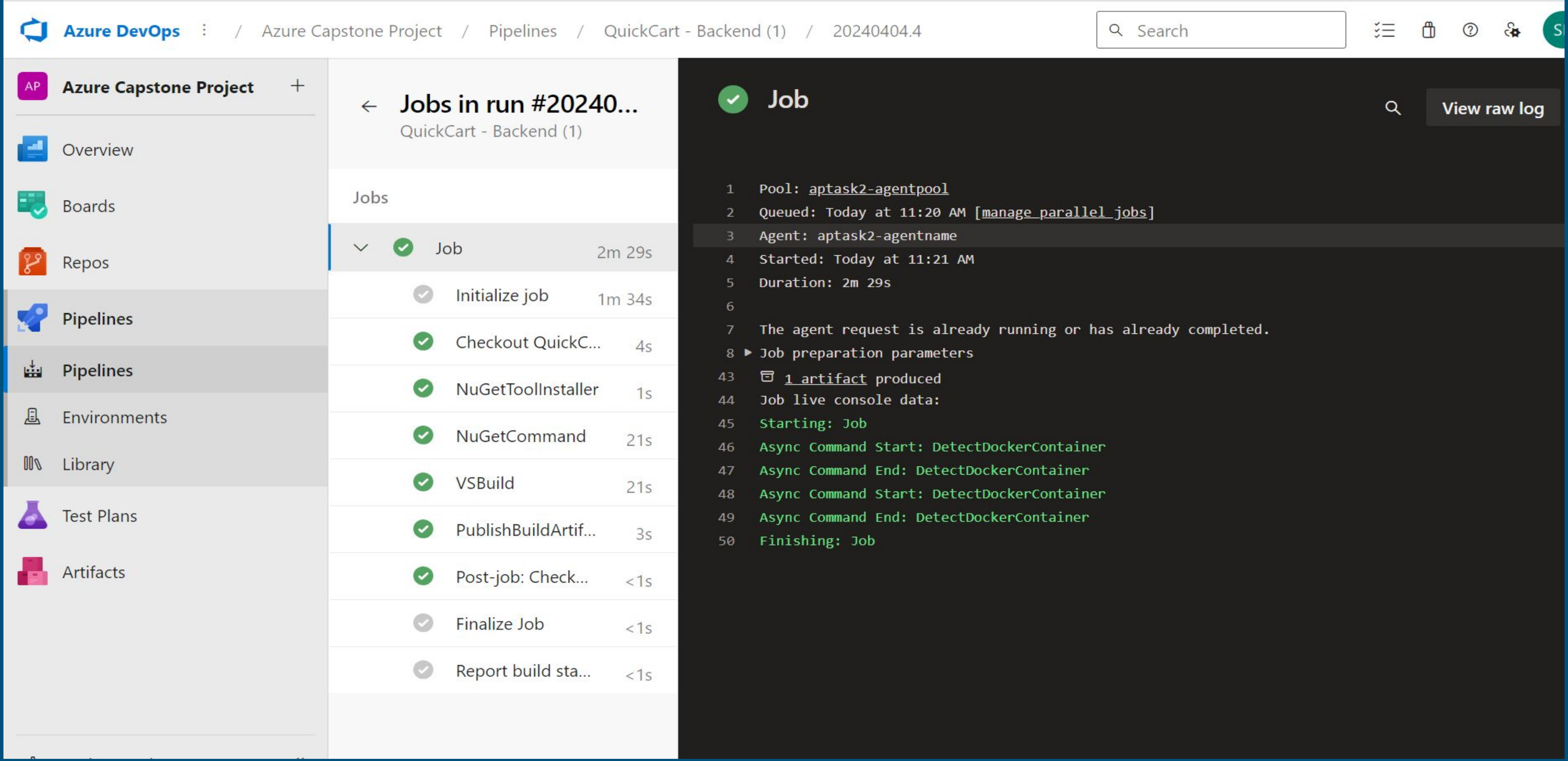
Jobs	
Job	1m 16s
Initialize job	7s
Checkout QuickC...	4s
NuGetToolInstaller	2s
NuGetCommand	43s
VSBuild	17s
PublishBuildArtif...	1s
Post-job: Check...	<1s
Finalize Job	<1s
Report build sta...	<1s

Job

```
Pool: Azure Pipelines
Image: windows-2019
Agent: Hosted Agent
Started: Just now
Duration: 1m 16s

▶ Job preparation parameters
1 artifact produced
Job live console data:
Finishing: Job
```

Build CI with Self Hosted:



Jobs in run #20240...

QuickCart - Backend (1)

Jobs

Job	2m 29s
Initialize job	1m 34s
Checkout QuickC...	4s
NuGetToolInstaller	1s
NuGetCommand	21s
VSBuild	21s
PublishBuildArtif...	3s
Post-job: Check...	<1s
Finalize Job	<1s
Report build sta...	<1s

Job

```
Pool: aptask2-agentpool
Queued: Today at 11:20 AM [manage parallel jobs]
Agent: aptask2-agentname
Started: Today at 11:21 AM
Duration: 2m 29s

The agent request is already running or has already completed.
▶ Job preparation parameters
1 artifact produced
Job live console data:
Starting: Job
Async Command Start: DetectDockerContainer
Async Command End: DetectDockerContainer
Async Command Start: DetectDockerContainer
Async Command End: DetectDockerContainer
Finishing: Job
```

TASK-3: Implementing Continuous Deployment pipelines::

Goal:

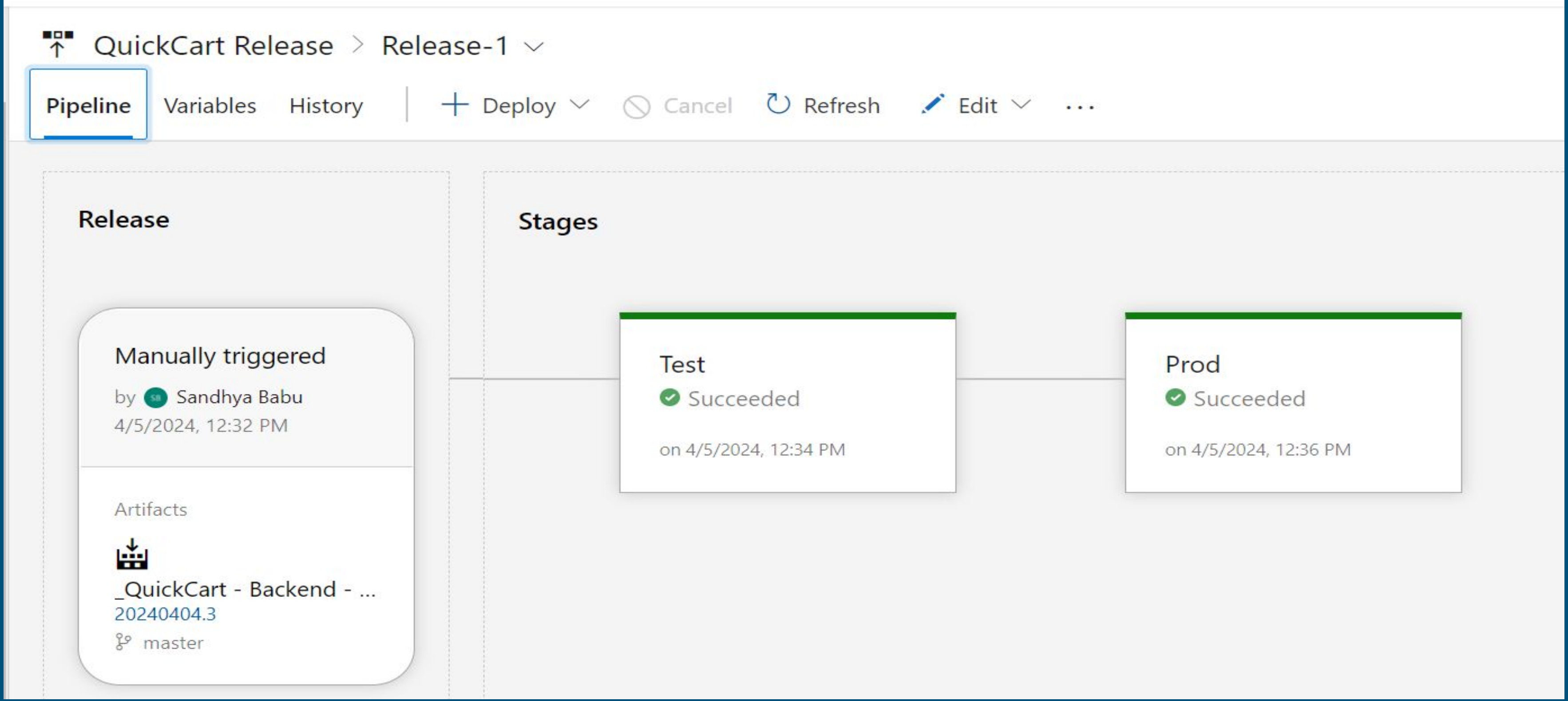
❑ Learn to build CD mechanism using Classic Interface.

❑ Understand Approval mechanism + Gates.

For more task 3 info, click the source Document URL:

https://docs.google.com/document/d/1GaG5MMES-q1A5DjpKeIVOWS2MzPbj9p1/edit

Secenio 1: A New release pipeline,which will deploy the changes to Azure App Service. 2 Stages, Testing and production



Manually triggered

by Sandhya Babu

4/5/2024, 12:32 PM

Artifacts

_QuickCart - Backend - ...

20240404.3

master

Test

✓ Succeeded

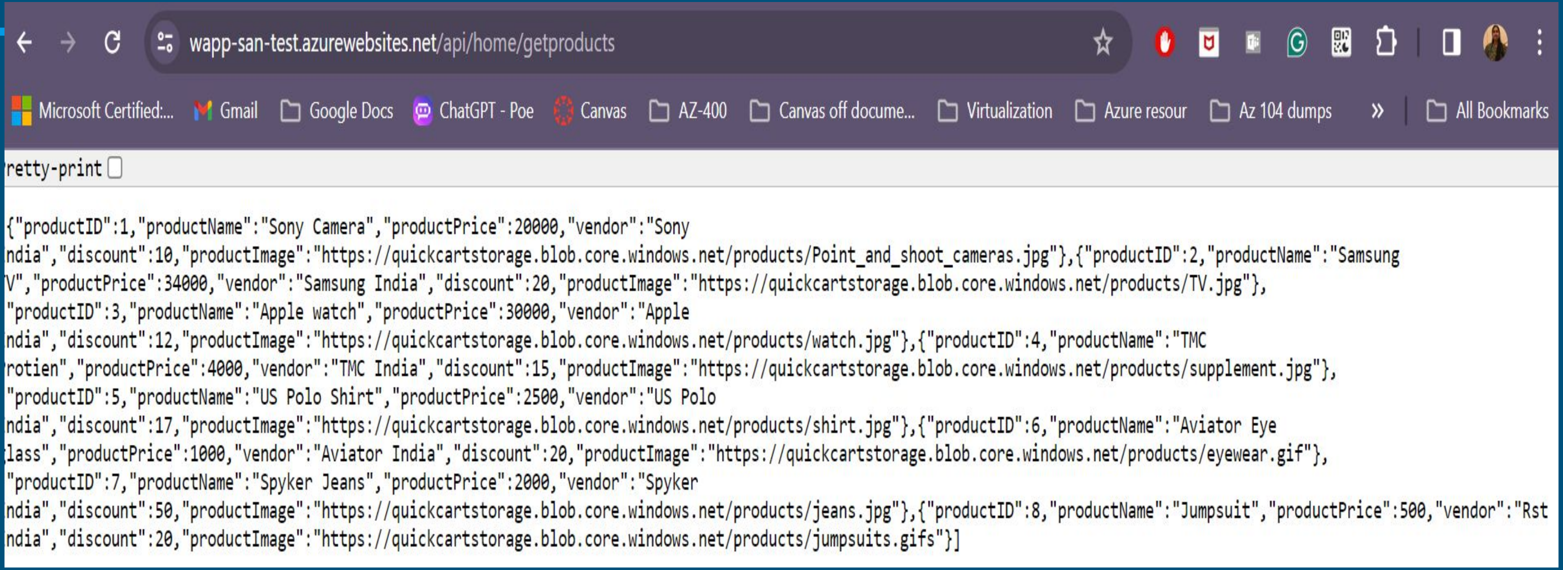
on 4/5/2024, 12:34 PM

Prod

✓ Succeeded

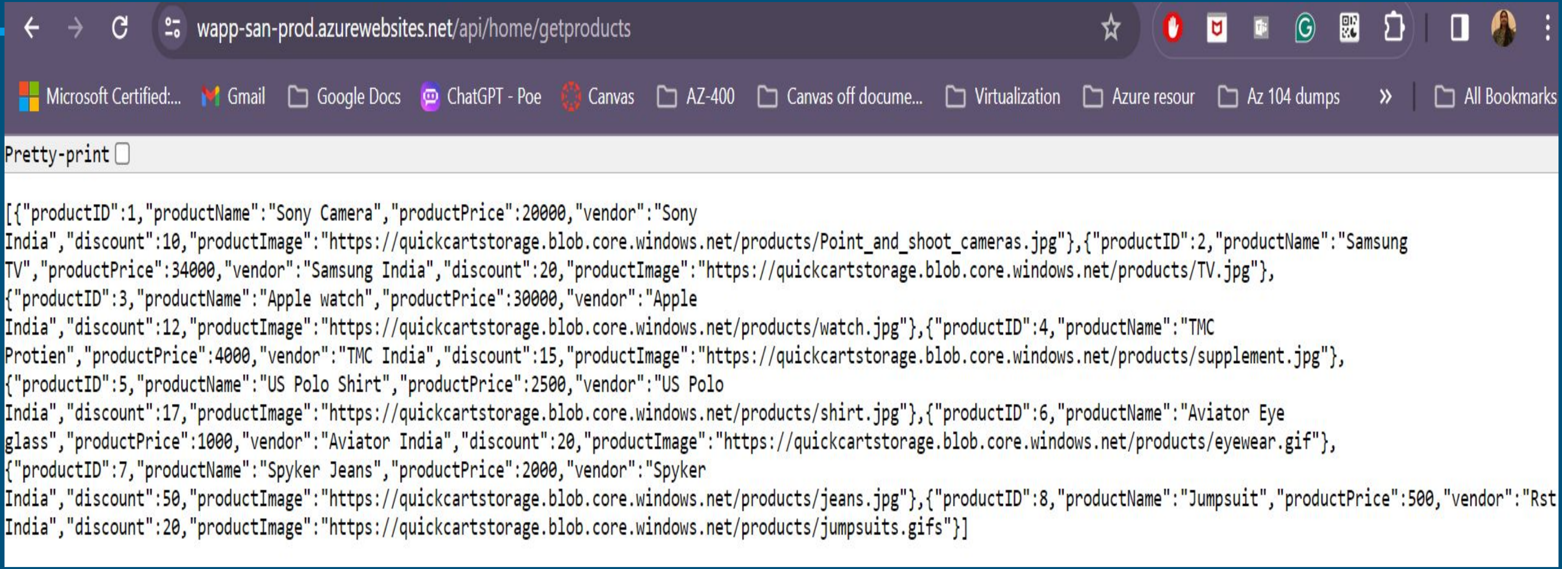
on 4/5/2024, 12:36 PM

Changes deployed in Azure app service - Test (Backend)



```
{"productID":1,"productName":"Sony Camera","productPrice":20000,"vendor":"Sony
ndia","discount":10,"productImage":"https://quickcartstorage.blob.core.windows.net/products/Point_and_shoot_cameras.jpg"},{"productID":2,"productName":"Samsung
V","productPrice":34000,"vendor":"Samsung India","discount":20,"productImage":"https://quickcartstorage.blob.core.windows.net/products/TV.jpg"},
"productID":3,"productName":"Apple watch","productPrice":30000,"vendor":"Apple
ndia","discount":12,"productImage":"https://quickcartstorage.blob.core.windows.net/products/watch.jpg"},{"productID":4,"productName":"TMC
rotien","productPrice":4000,"vendor":"TMC India","discount":15,"productImage":"https://quickcartstorage.blob.core.windows.net/products/supplement.jpg"},
"productID":5,"productName":"US Polo Shirt","productPrice":2500,"vendor":"US Polo
ndia","discount":17,"productImage":"https://quickcartstorage.blob.core.windows.net/products/shirt.jpg"},{"productID":6,"productName":"Aviator Eye
lass","productPrice":1000,"vendor":"Aviator India","discount":20,"productImage":"https://quickcartstorage.blob.core.windows.net/products/eyewear.gif"},
"productID":7,"productName":"Spyker Jeans","productPrice":2000,"vendor":"Spyker
ndia","discount":50,"productImage":"https://quickcartstorage.blob.core.windows.net/products/jeans.jpg"},{"productID":8,"productName":"Jumpsuit","productPrice":500,"vendor":"Rst
ndia","discount":20,"productImage":"https://quickcartstorage.blob.core.windows.net/products/jumpsuits.gifs"}]
```

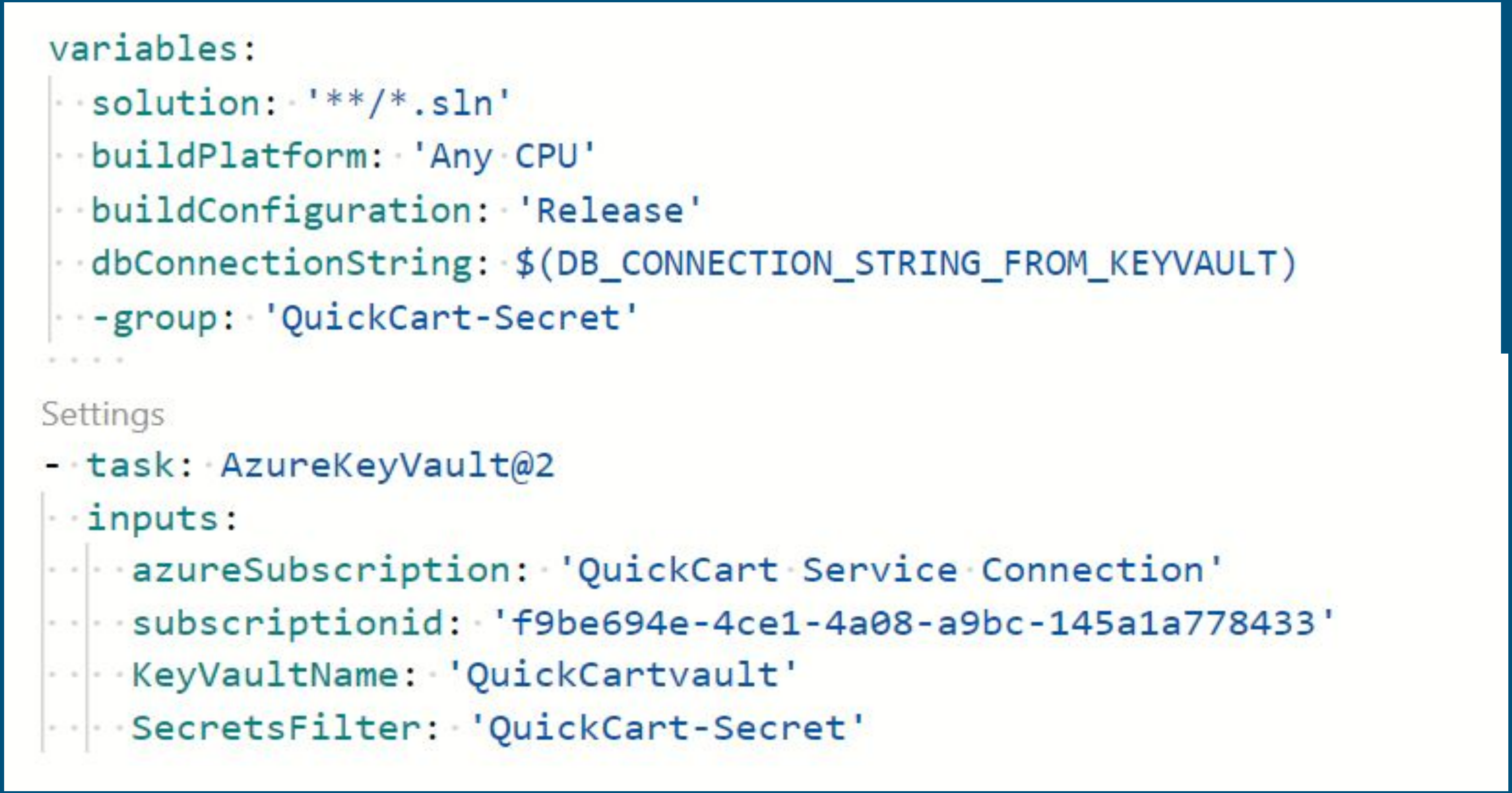
Changes deployed in Azure app service - Production(backend)



Pretty-print □

```
[{"productID":1,"productName":"Sony Camera","productPrice":20000,"vendor":"Sony
India","discount":10,"productImage":"https://quickcartstorage.blob.core.windows.net/products/Point_and_shoot_cameras.jpg"},{"productID":2,"productName":"Samsung
TV","productPrice":34000,"vendor":"Samsung India","discount":20,"productImage":"https://quickcartstorage.blob.core.windows.net/products/TV.jpg"},
{"productID":3,"productName":"Apple watch","productPrice":30000,"vendor":"Apple
India","discount":12,"productImage":"https://quickcartstorage.blob.core.windows.net/products/watch.jpg"},{"productID":4,"productName":"TMC
Protien","productPrice":4000,"vendor":"TMC India","discount":15,"productImage":"https://quickcartstorage.blob.core.windows.net/products/supplement.jpg"},
{"productID":5,"productName":"US Polo Shirt","productPrice":2500,"vendor":"US Polo
India","discount":17,"productImage":"https://quickcartstorage.blob.core.windows.net/products/shirt.jpg"},{"productID":6,"productName":"Aviator Eye
glass","productPrice":1000,"vendor":"Aviator India","discount":20,"productImage":"https://quickcartstorage.blob.core.windows.net/products/eyewear.gif"},
{"productID":7,"productName":"Spyker Jeans","productPrice":2000,"vendor":"Spyker
India","discount":50,"productImage":"https://quickcartstorage.blob.core.windows.net/products/jeans.jpg"},{"productID":8,"productName":"Jumpsuit","productPrice":500,"vendor":"Rst
India","discount":20,"productImage":"https://quickcartstorage.blob.core.windows.net/products/jumpsuits.gifs"}]
```

Added the azure key vault tasks and declared Azure variable group & dbconnection string in the pipeline.

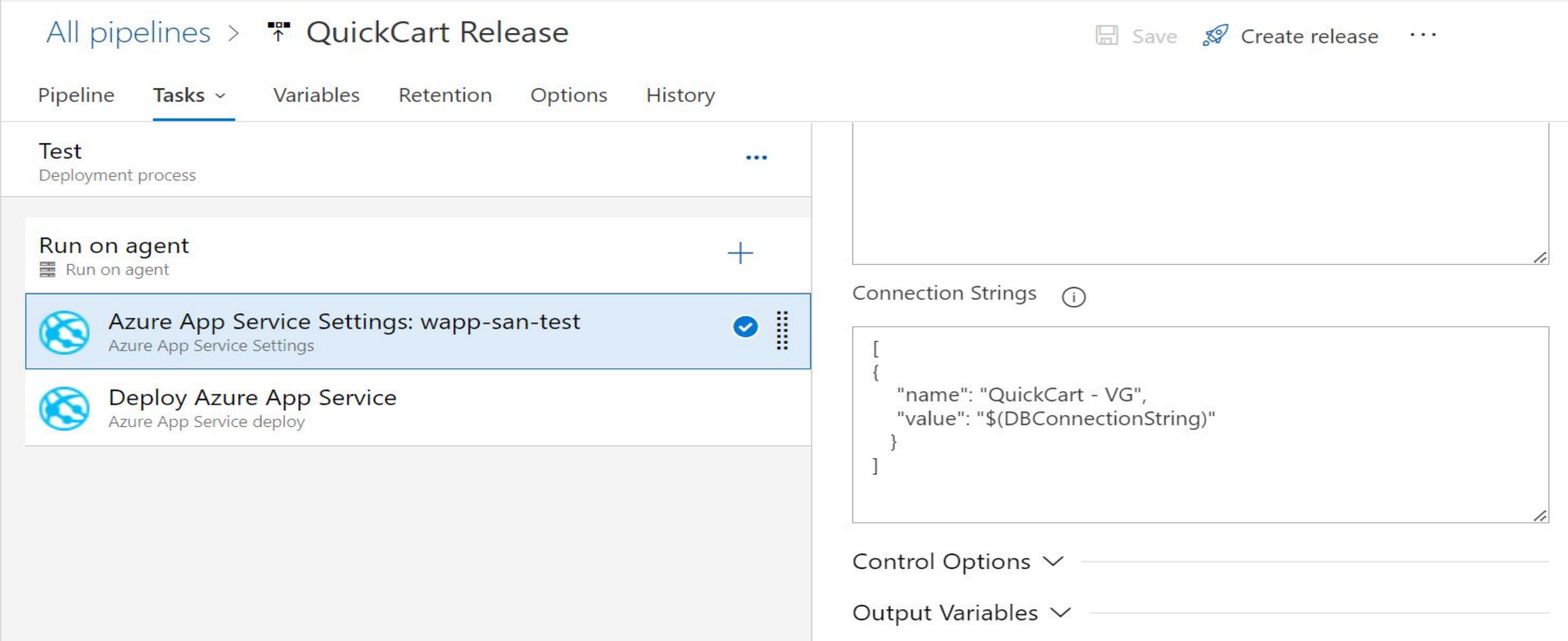


```yaml
variables:
  solution: '**/*.sln'
  buildPlatform: 'Any CPU'
  buildConfiguration: 'Release'
  dbConnectionString: $(DB_CONNECTION_STRING_FROM_KEYVAULT)
  -group: 'QuickCart-Secret'
```

Settings

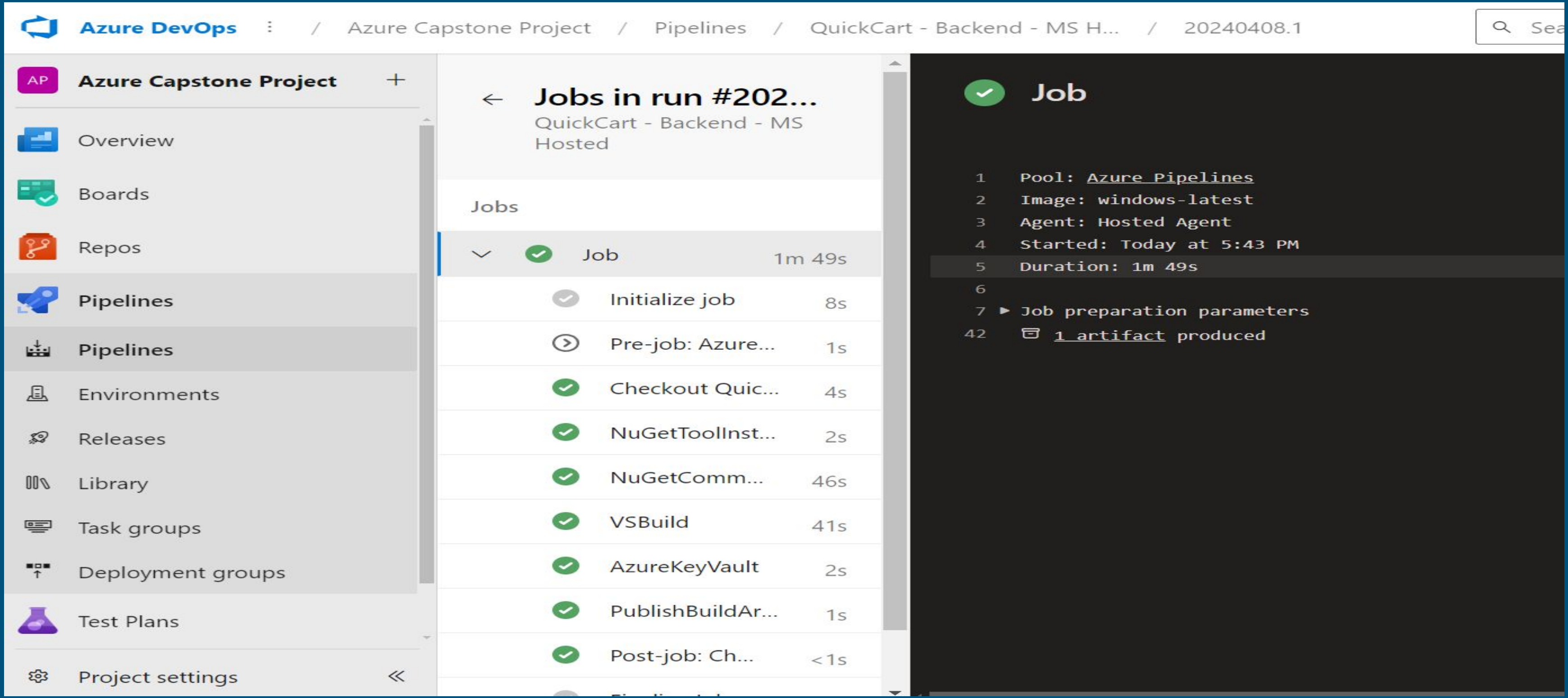
```yaml
- task: AzureKeyVault@2
  inputs:
    azureSubscription: 'QuickCart Service Connection'
    subscriptionid: 'f9be694e-4ce1-4a08-a9bc-145a1a778433'
    KeyVaultName: 'QuickCartvault'
    SecretsFilter: 'QuickCart-Secret'
```

Specifying a connection string in your app settings that references a secret stored in Azure Key Vault.



```
[
{
    "name": "QuickCart - VG",
    "value": "$(DBConnectionString)"
  }
]
```

Incorporate your database connection string from the Key Vault into your Azure DevOps YAML pipeline



Jobs in run #202...
QuickCart - Backend - MS Hosted

Jobs

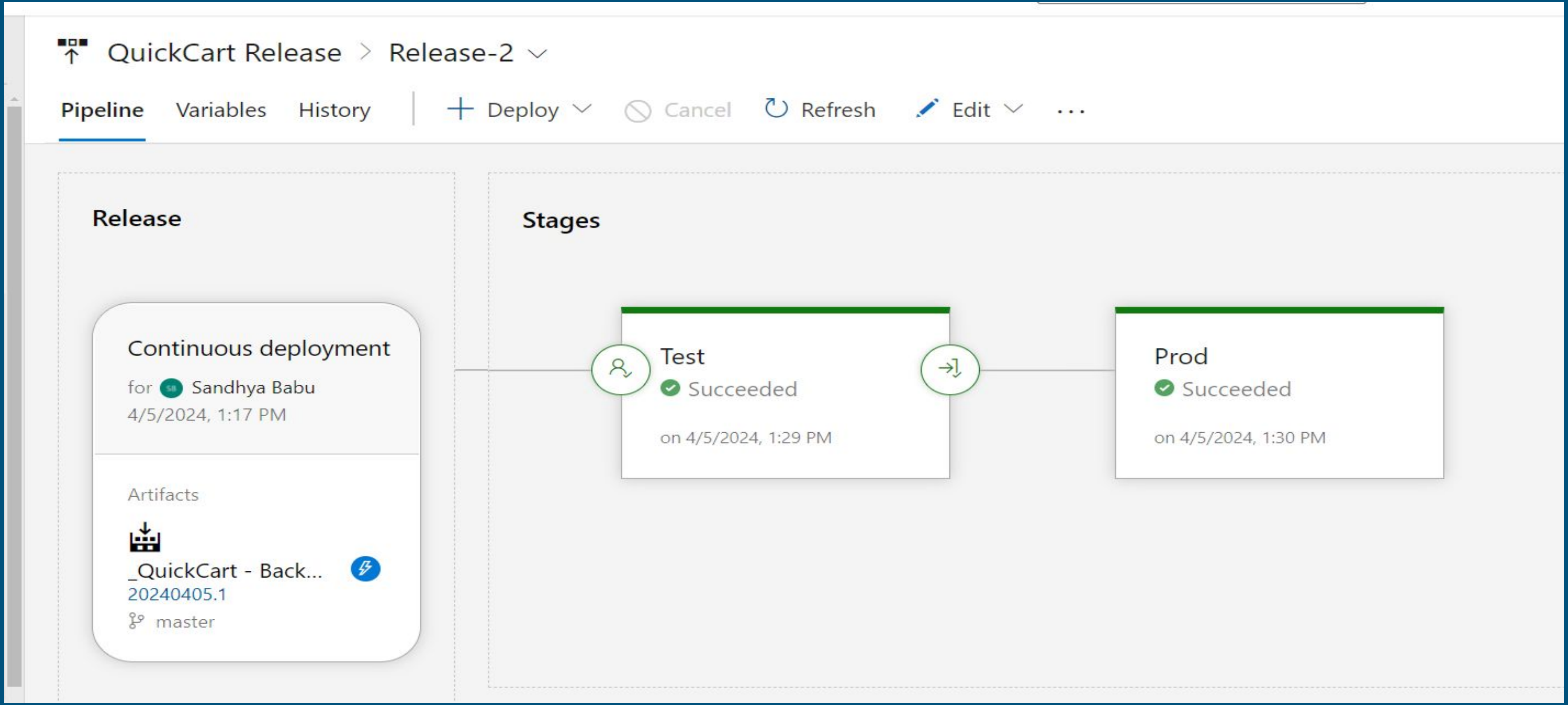
```
Job                    1m 49s
  Initialize job       8s
  Pre-job: Azure...    1s
  Checkout Quic...     4s
  NuGetToolInst...     2s
  NuGetComm...         46s
  VSBuild              41s
  AzureKeyVault        2s
  PublishBuildAr...    1s
  Post-job: Ch...      <1s
```

Job

```
Pool: Azure Pipelines
Image: windows-latest
Agent: Hosted Agent
Started: Today at 5:43 PM
Duration: 1m 49s

▶ Job preparation parameters
1 artifact produced
```

Scenario 2: Enabled Continuous Integration / Set predeployment(approval) and Post deployment conditions(Enabled Gates)



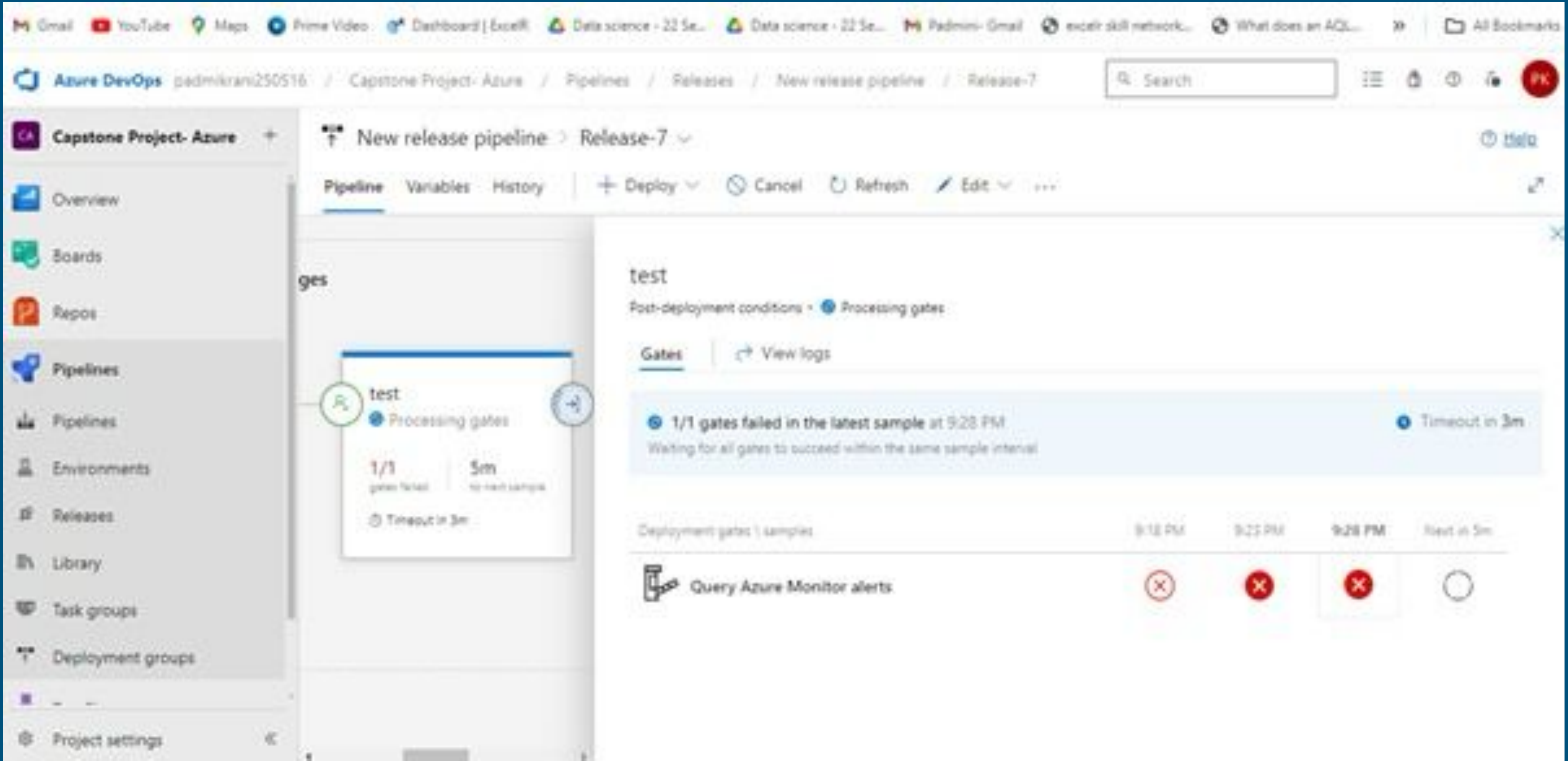
Continuous deployment
for Sandhya Babu
4/5/2024, 1:17 PM

Artifacts
_QuickCart - Back...
20240405.1
master

Test
Succeeded
on 4/5/2024, 1:29 PM

Prod
Succeeded
on 4/5/2024, 1:30 PM

The query-based alert detected this issue, and as a result, the deployment was prevented from proceeding to the next stage (production)



test

Post-deployment conditions • Processing gates

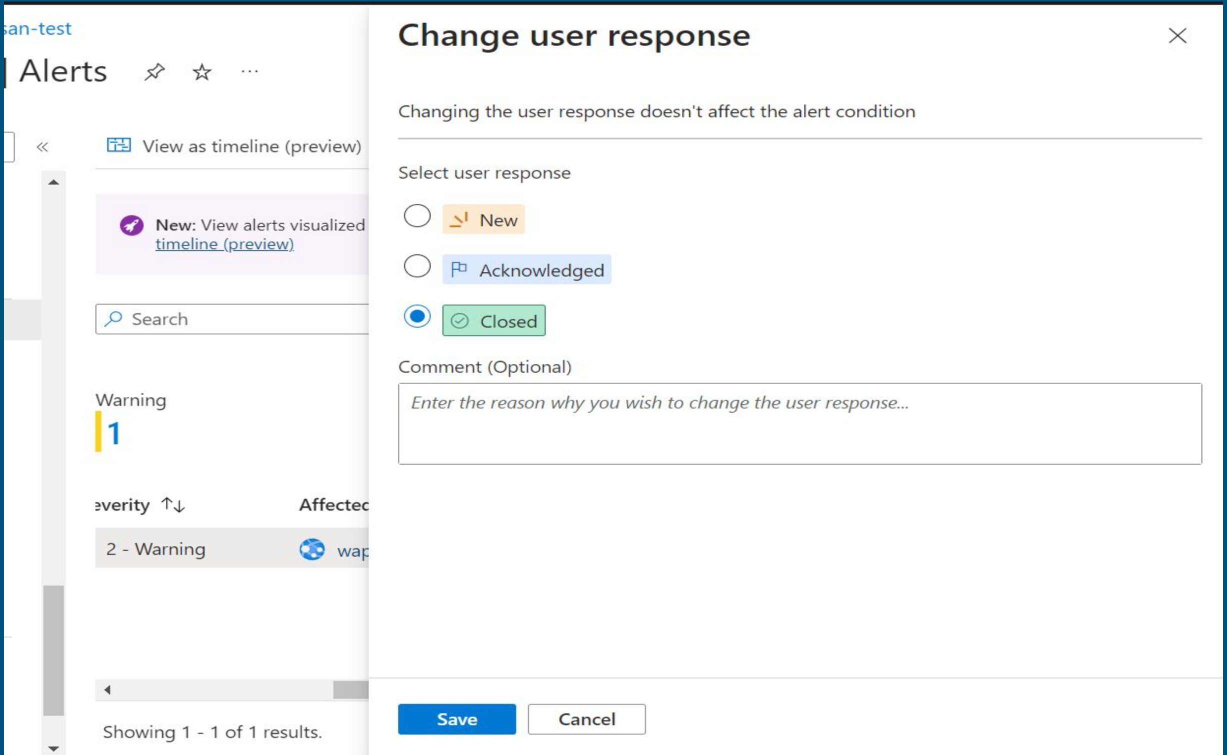
1/1 gates failed in the latest sample at 9:28 PM

Waiting for all gates to succeed within the same sample interval

Timeout in 3m

Deployment gates | samples	9:18 PM	9:23 PM	9:28 PM	Next in 5m
Query Azure Monitor alerts	⊗	✖	✖	○

Fixing Azure Monitor Alerts for Deployment Gates pass



san-test
Alerts
View as timeline (preview)
New: View alerts visualized
timeline (preview)
Search
Warning
1
everity ↑↓
Affectec
2 - Warning
wap
Showing 1 - 1 of 1 results.

Change user response

Changing the user response doesn't affect the alert condition

Select user response

○ New

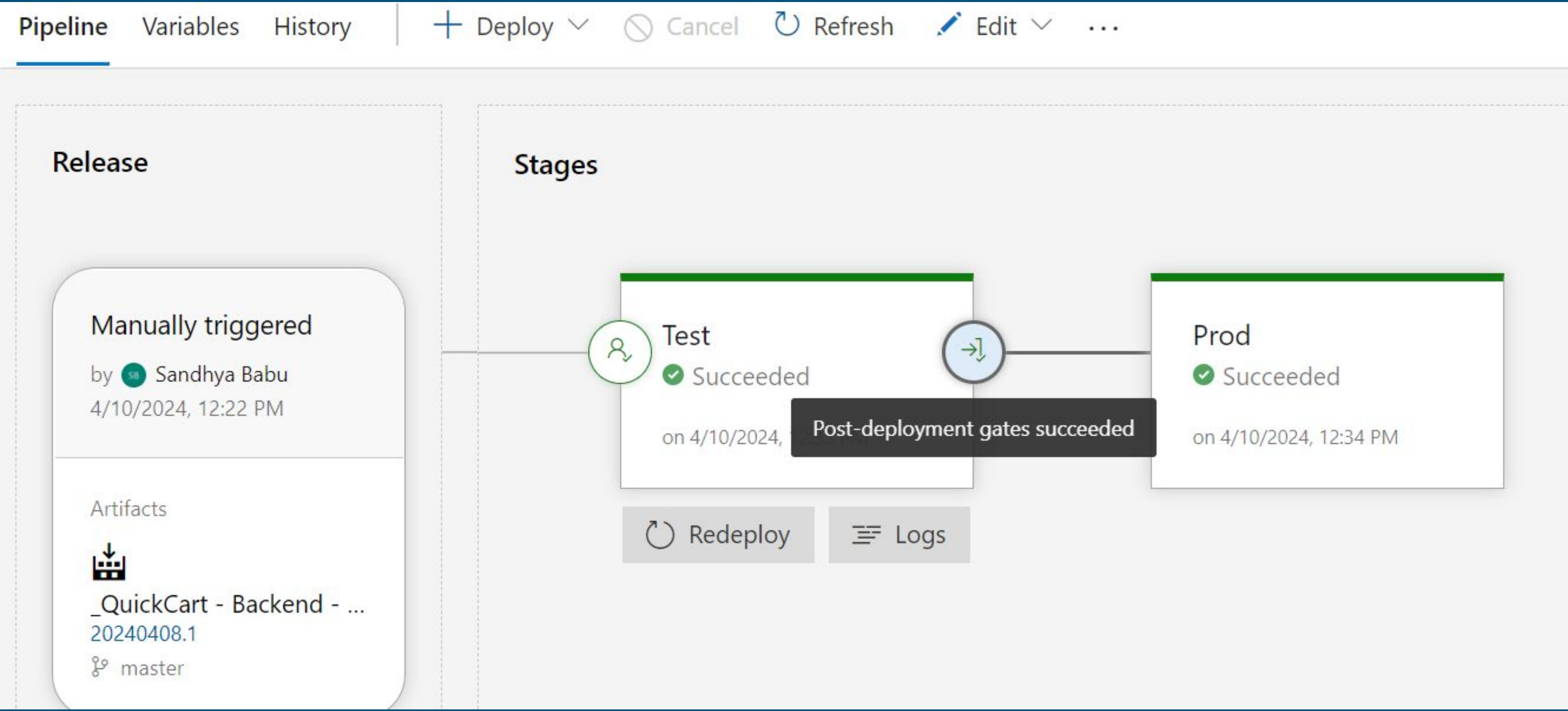
○ Acknowledged

● Closed

Comment (Optional)

Enter the reason why you wish to change the user response...

Evaluation succeeded - Deployed to production



Release

Manually triggered

by SB Sandhya Babu

4/10/2024, 12:22 PM

Artifacts

_QuickCart - Backend - ...

20240408.1

master

Stages

Test

Succeeded

on 4/10/2024,

Post-deployment gates succeeded

Prod

Succeeded

on 4/10/2024, 12:34 PM

TASK-4: IMPLEMENT LOGIN FUNCTIONALITY USING AZURE FUNCTION

Goal:

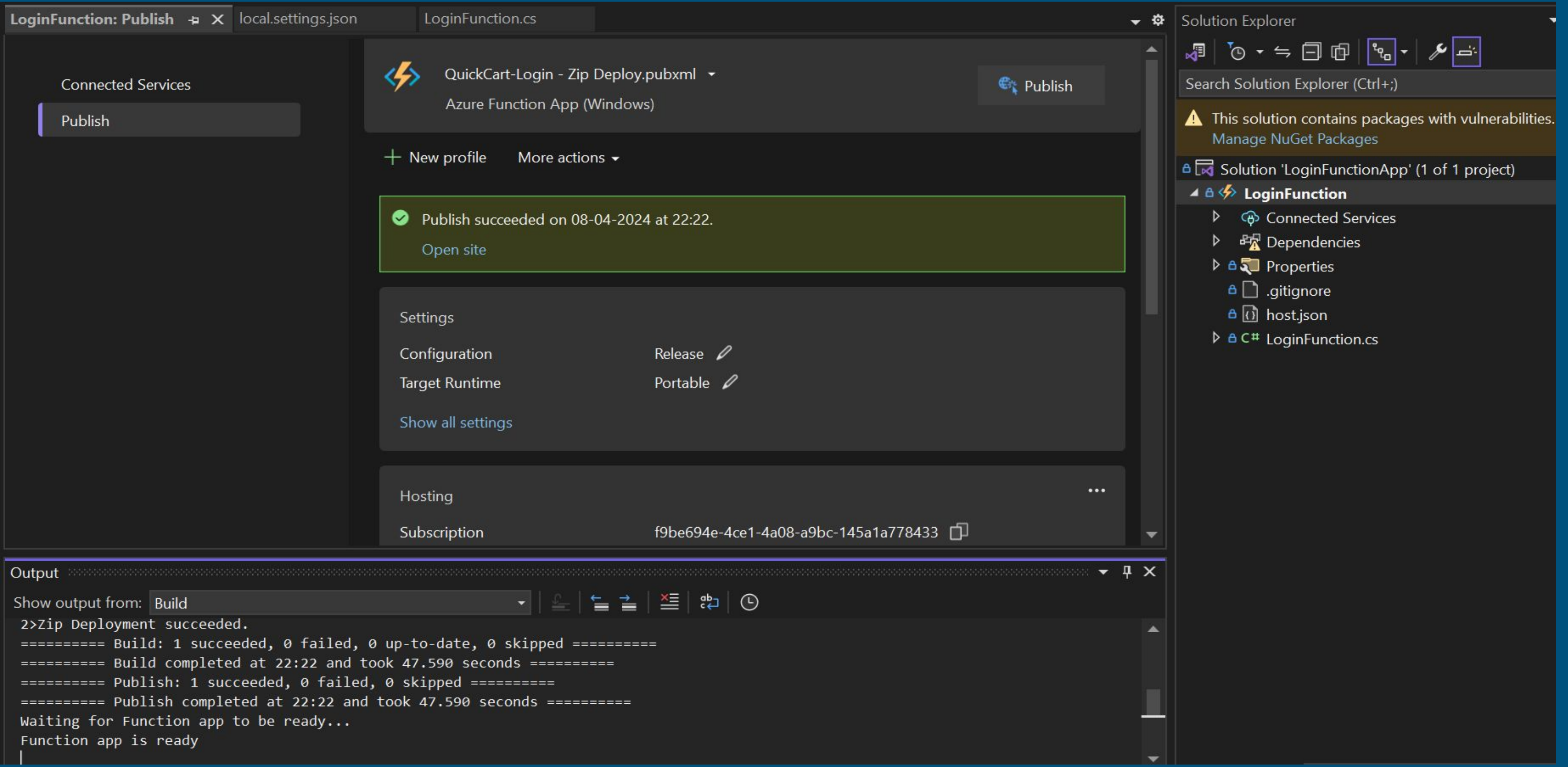
❑ Learn to build an Azure Function (C#) from IDE

❑ Deploy code to Azure function.

For more task 4 info, click the source Document URL:

https://docs.google.com/document/d/1ok1q8LTFV8ypq1nVpU9cT108JAQFQJDp/edit

Deploy the login code to an Azure function



Publish succeeded on 08-04-2024 at 22:22.

Settings

Configuration Release

Target Runtime Portable

Hosting

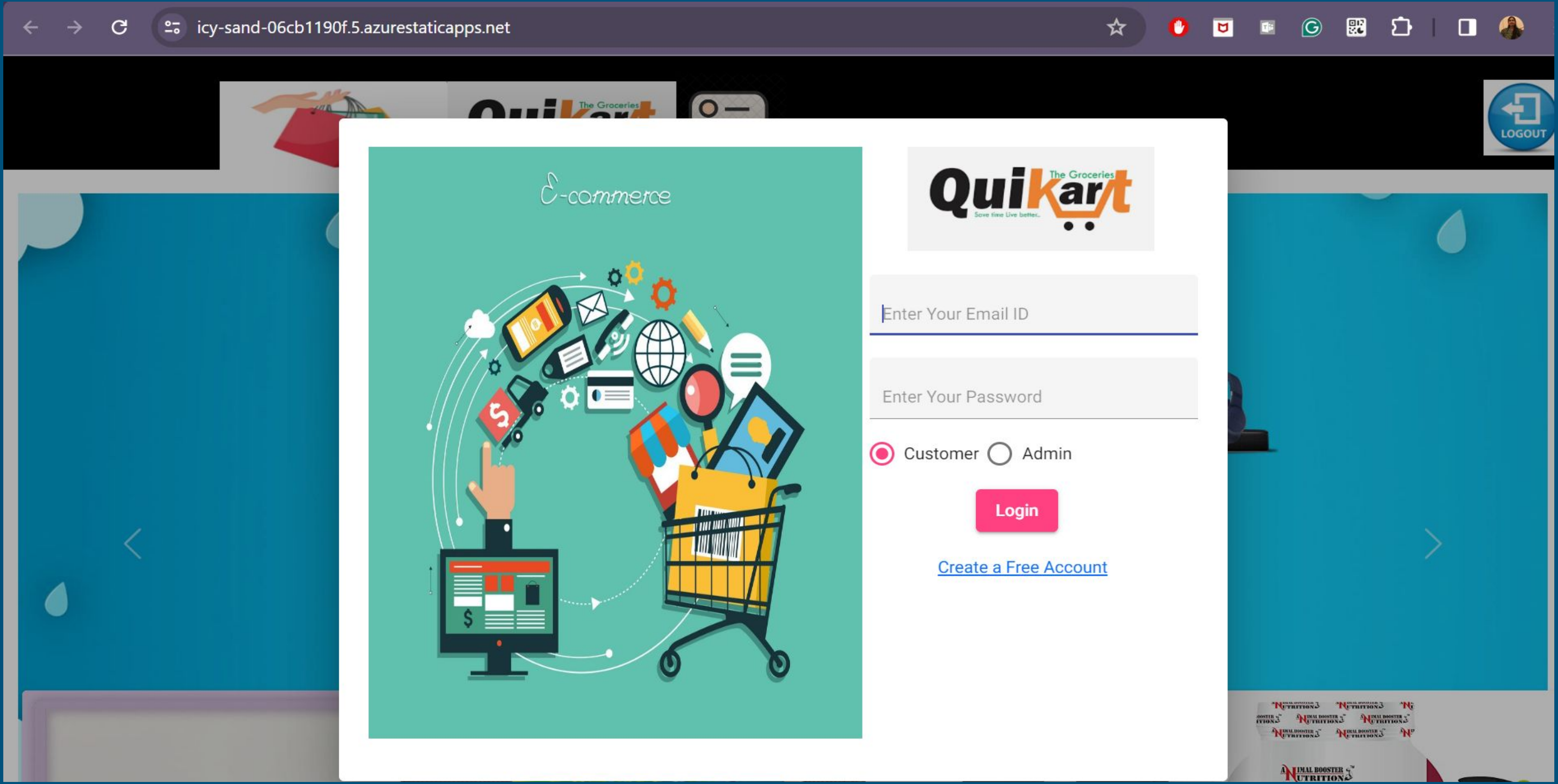
Subscription f9be694e-4ce1-4a08-a9bc-145a1a778433

Show output from: Build

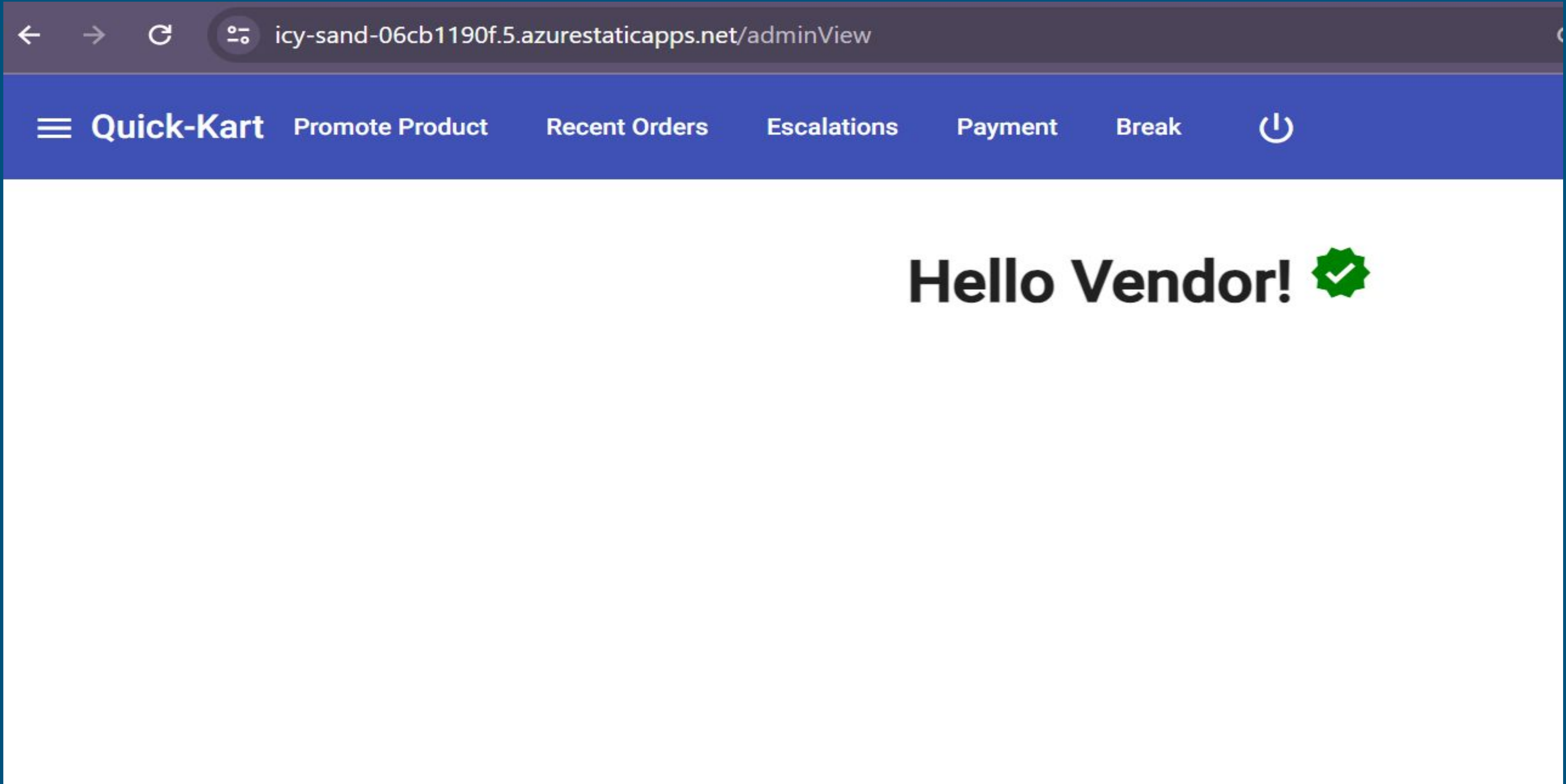
```
2>Zip Deployment succeeded.
========== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ==========
========== Build completed at 22:22 and took 47.590 seconds ==========
========== Publish: 1 succeeded, 0 failed, 0 skipped ==========
========== Publish completed at 22:22 and took 47.590 seconds ==========
Waiting for Function app to be ready...
Function app is ready
```

This solution contains packages with vulnerabilities.

Login Functionality Test



Login Functionality Output



Hello Vendor! ✓