Lab#4 – Azure DevOps CI/CD

**Due Date:** Midnight of Week#13 Friday (April 14)

**Purpose:** The purpose of this lab is to help you understand:

* Azure DevOps CI/CD
* Azure DevOps Artifacts

**Instructions**: Be sure to read the following general instructions carefully:

This lab should be completed individually by all students. Submit your solution **through the dropbox**, name your submission as **studentID(yourlastname)\_Labnumber.zip**.

e.g., **300123456**(**smith)\_Lab#4**.zip

**Rubric**

|  |  |
| --- | --- |
| Question | Marks |
| 1. Continuous Integration (Use self-hosted agent) | 6 |
| 1. Continuous Deployment | 10 |

Question 1 – Continuous Integration in Azure DevOps [ marks]

1. Visit <https://learn.microsoft.com/en-us/dotnet/core/tutorials/library-with-visual-studio?pivots=dotnet-6-0> and read of all of the instructions and do not start implementation
2. After reading all of the instructions, follow all of the steps under
   1. Prerequisites
   2. Create a solution
   3. Create a class library project

to implement the tutorial and use GitHub for SCM

1. Package and publish the NuGet package using your Azure DevOps yaml file to Azure Artifacts Feed
2. Commit changes for CI pipeline yaml file to GitHub, please make sure that the CI pipeline runs on Microsoft hosted agent
3. Add a console app to the solution, and finish the console app
4. Configure you Visual Studio 2022 solution for console app to reference the NuGet package for the library from Azure DevOps Artifact Feed and then follow
   1. Run the App Steps
5. Commit all source code changes to GitHub

After you finish all of the steps above, include the url for your GitHub repo and Azure DevOps project. Include screenshots for following into your submission doc file

<https://github.com/KJYX7/Lab4>

<https://dev.azure.com/XuKathy/Lab4>

1. Pipeline log showing NuGet library is built [1 mark]
2. Artifact Feed UI showing NuGet library [1 mark]
3. Visual Studio 2022 showing NuGet library is downloaded from Azure DevOps Artifact Feed [1 mark]
4. Command window showing successful run of the Console App with and without uppercase string [2 marks]
5. Source code commits in GitHub [1 mark]

Question 2 – Continuous Integration&Continuous Delivery in Azure DevOps [10 marks]

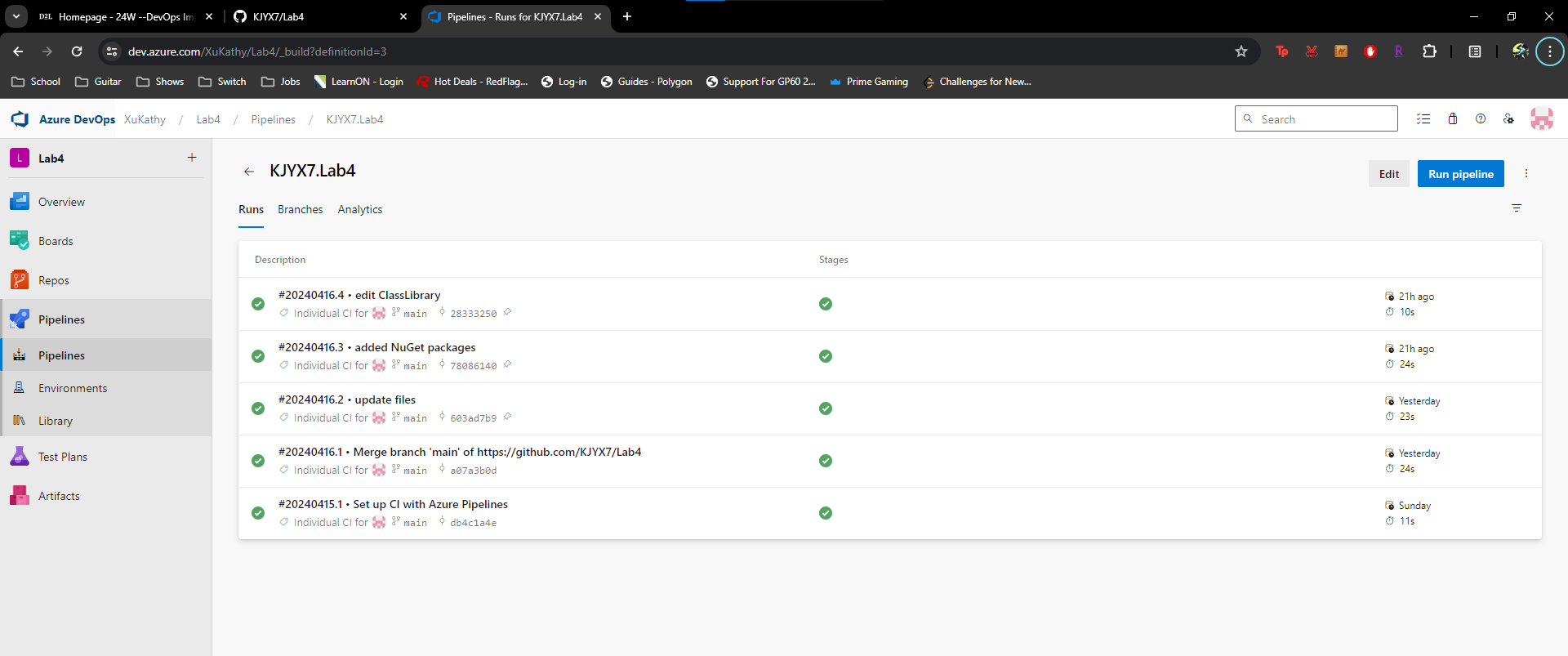
1. ***Continue Q2 of Lab 3***, and finish first 3 parts (i.e., coding), and put code to GitHub repo
2. Create Azure DevOps project for your Web app
3. Upon deployment, your asp.net core web application should be running on your machine
4. Create release pipeline

After you finish all of those steps above, include the **url for your GitHub repo and Azure DevOps project** as well as **screenshots** for following into your submission doc file

1. Self-hosted agent has been set up [1 mark]
2. A Deployment group has been set up properly [1 mark]
3. GitHub repo for C# code, and corresponding work item(s)’ status in Azure DevOps Boards [2 marks]
4. Azure DevOps project [1 mark]
5. Azure DevOps CI pipeline using self-hosted agent (yaml file) [2 marks]
6. Azure DevOps release pipeline (2 marks)
7. Pipeline log showing deployment of application zip has been succeeded [1 mark]

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated