# 

# 

# 

# 

# UiPath NuGet file Deployment using VSTS

# 

# 

# Naga Durga Anjaneya Raju Sagi

# DevOps Research Associate

# Miracle Software Systems, Inc.

# 

# Objective:

The objective of this document to provide an overview of configuring CI/CD pipeline for automated deployment of NuGet package using VSTS to UiPath Orchestrator which will create a real agile life cycle process in the project. It also helps the developers to focus on developing the code leaving the deployment process to VSTS.

# UiPath:

UiPath develops a platform for robotic process automation.

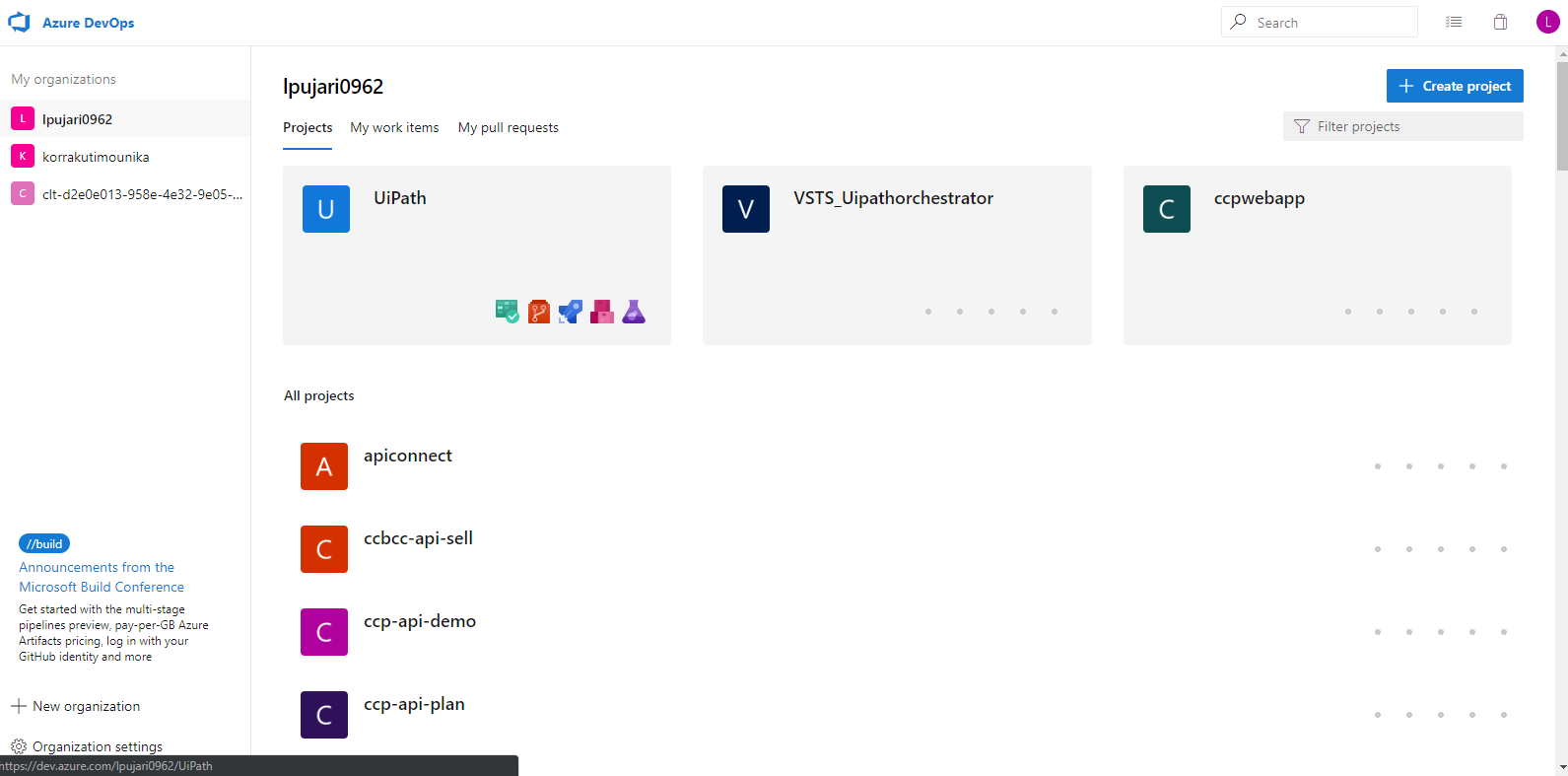
# UiPath Studio:

UiPath studio is an advanced tool that enables you to design automation processes in a visual manner, through diagrams.

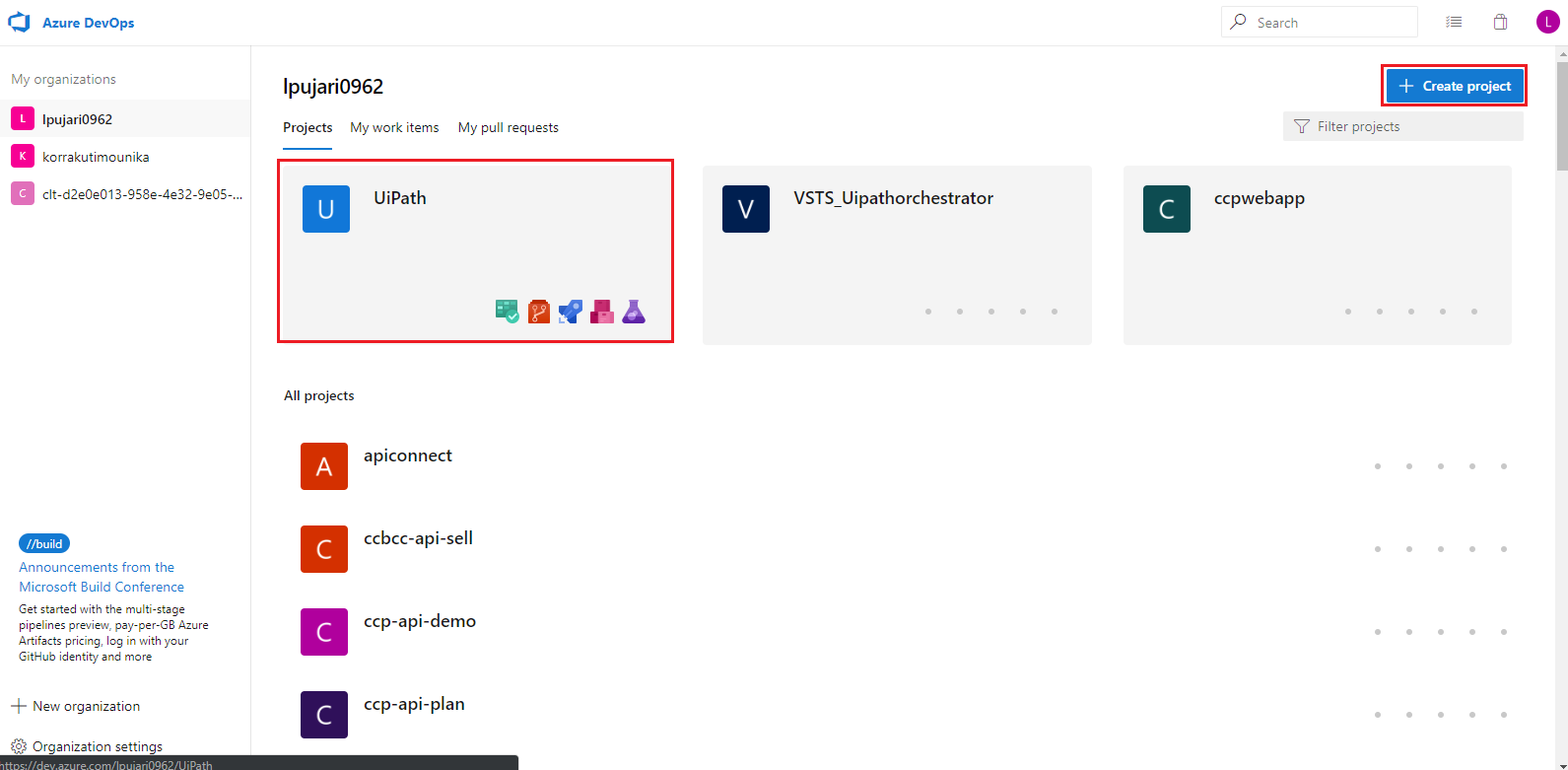
# UiPath Orchestrator:

UiPath Orchestrator is a web application that enables you to deploy, schedule, monitor and manage Robots and processes, while business exception handling is available through centralized work queues.

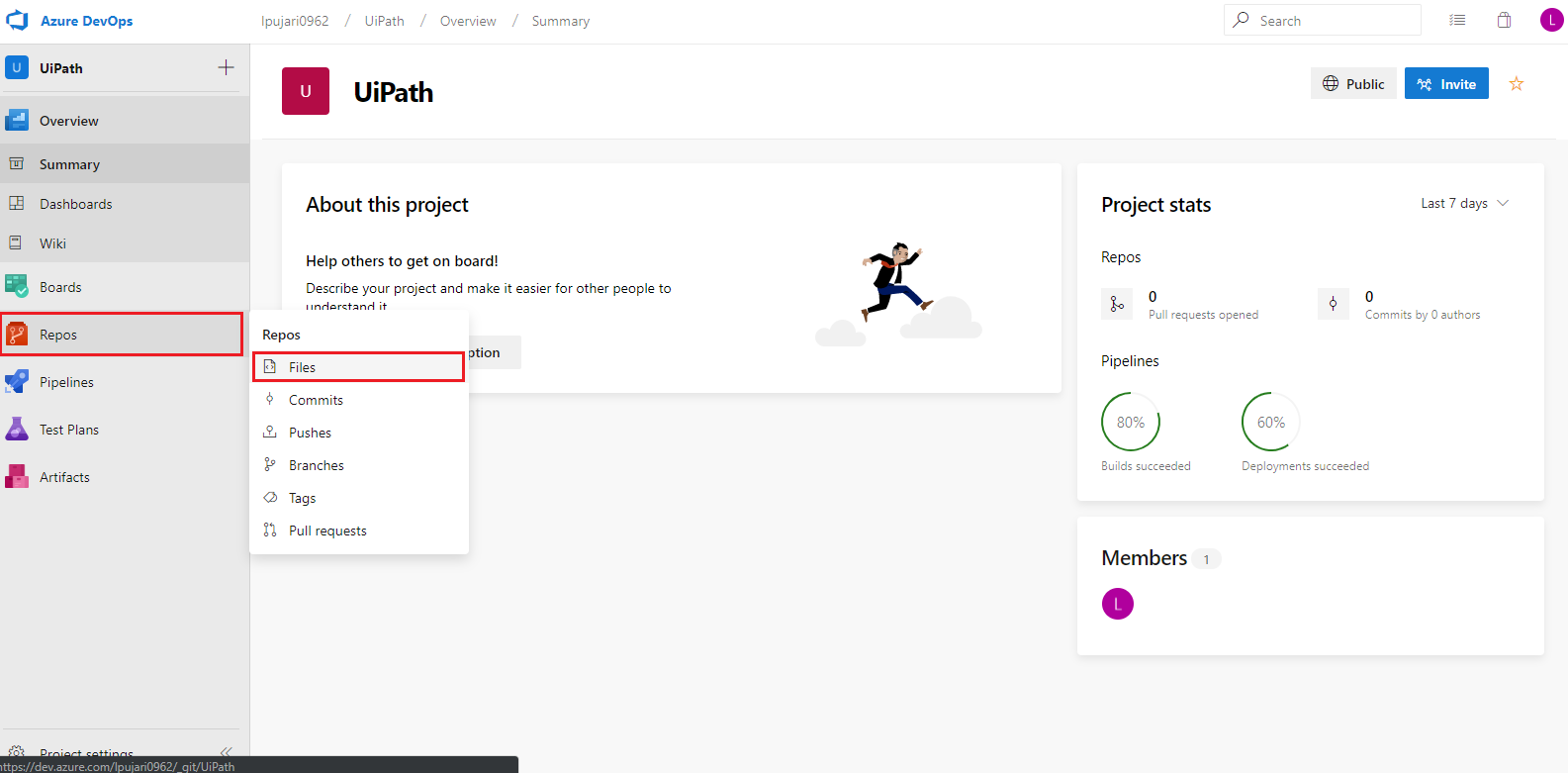
In order to start our pipeline first we need to login to **VSTS**. Once logged in, our dashboard will be as shown below



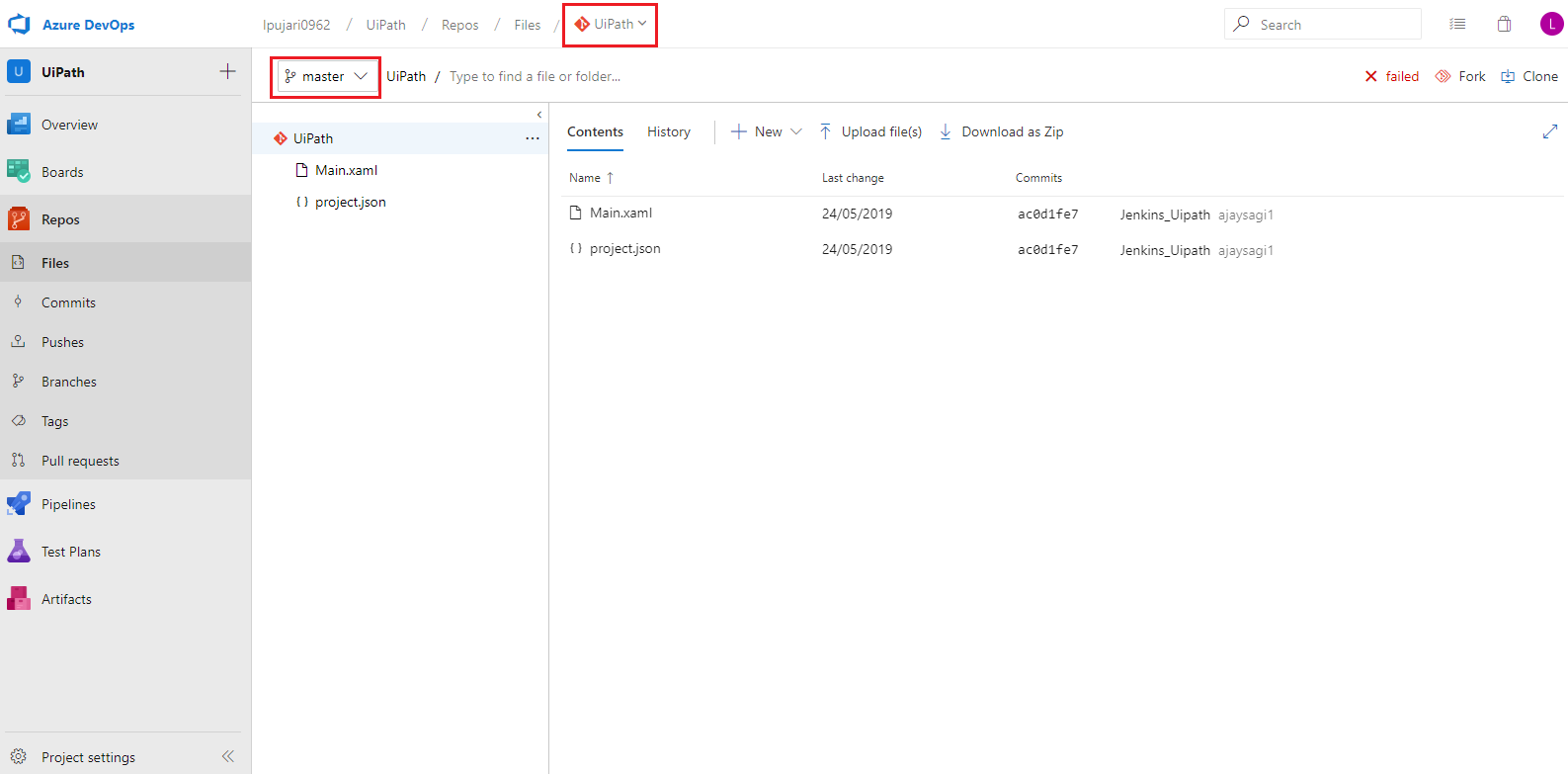
Now we need to create a new project by clicking on the Create Project. Here we have already created project called UiPath.



Now we will be using the repository “**UiPath**” that was created under the project “**UiPath**”.

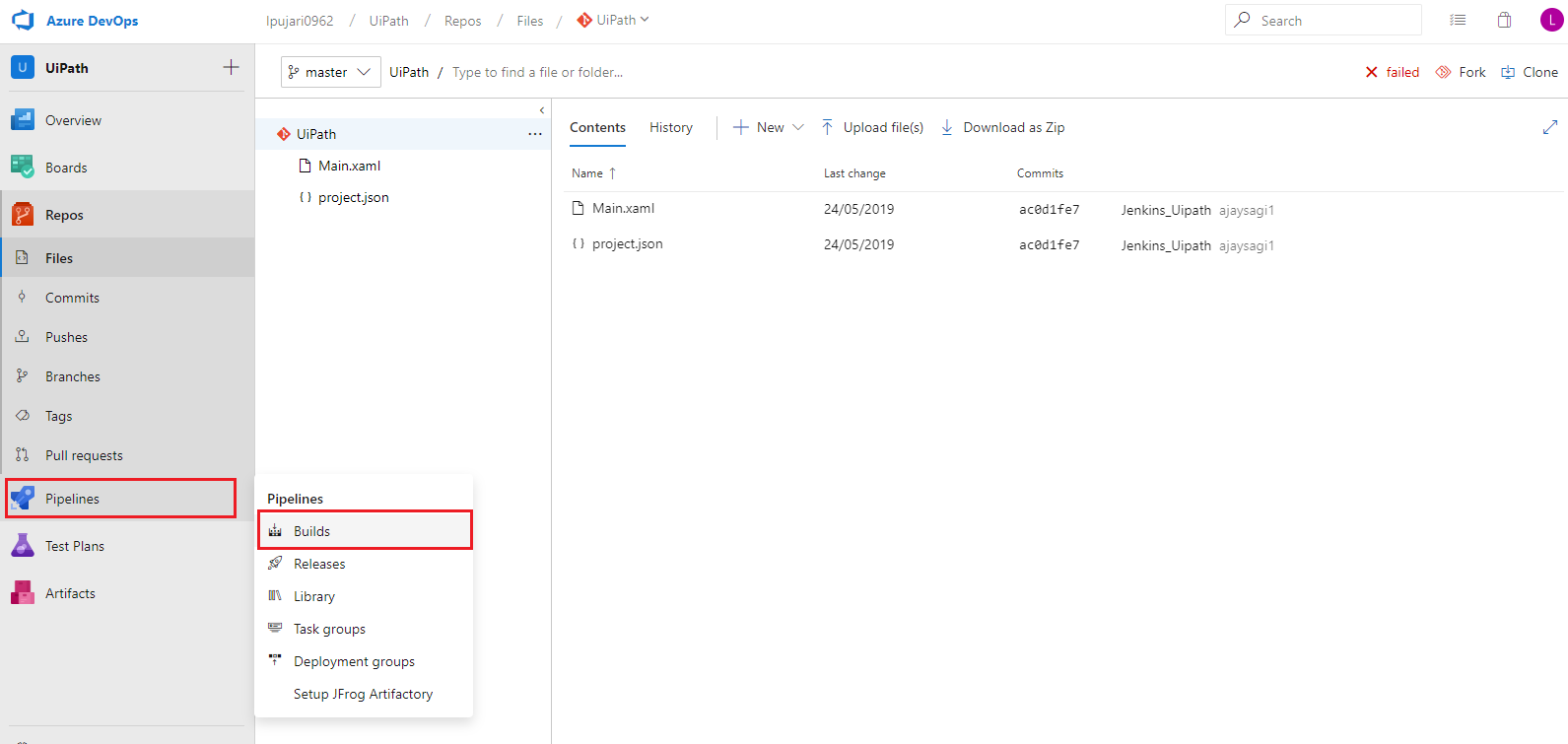


Now we can view the application that was already pushed to our repository UiPath under “**master**” branch.

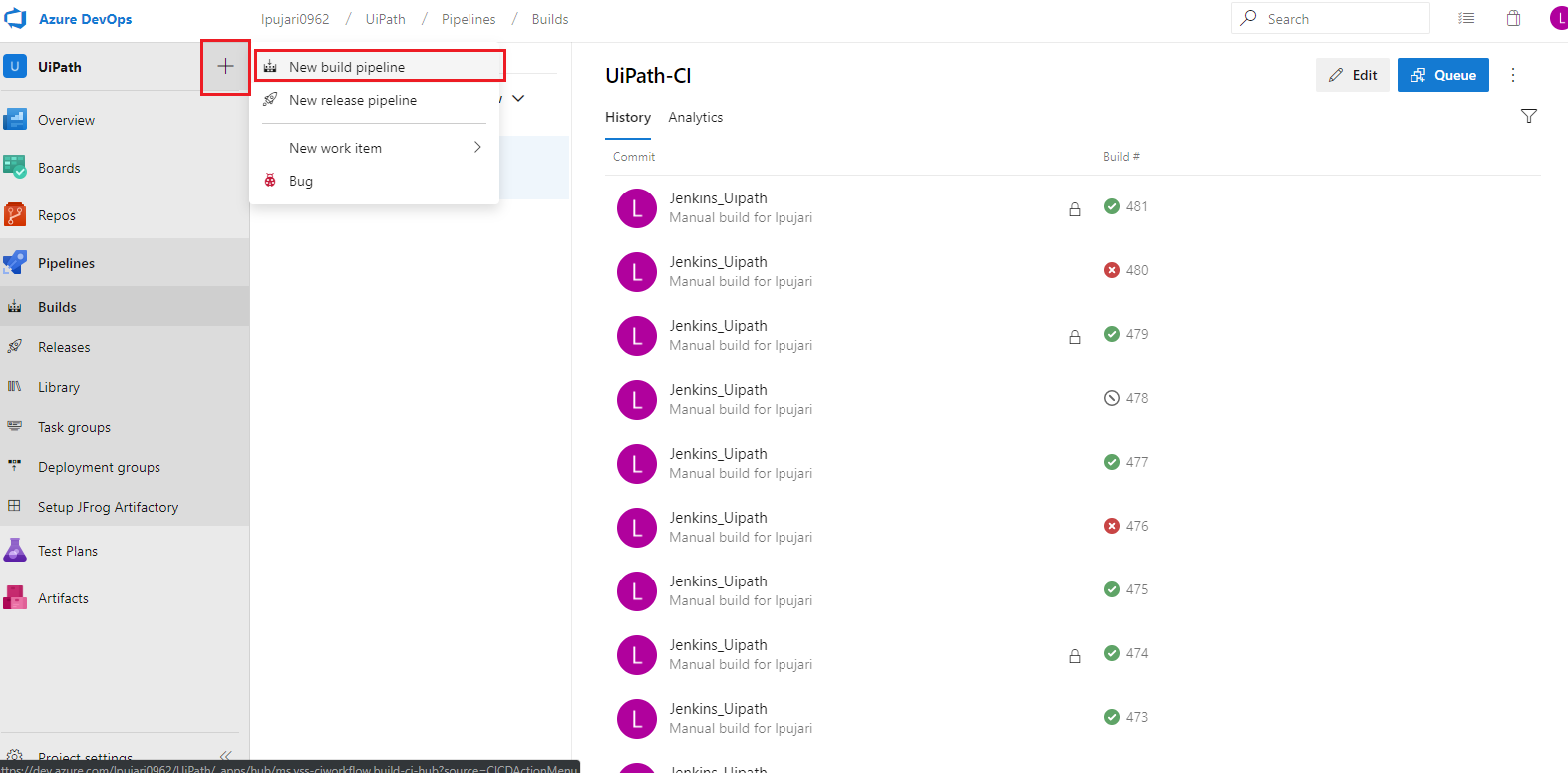


# Continuous Integration pipeline using VSTS:

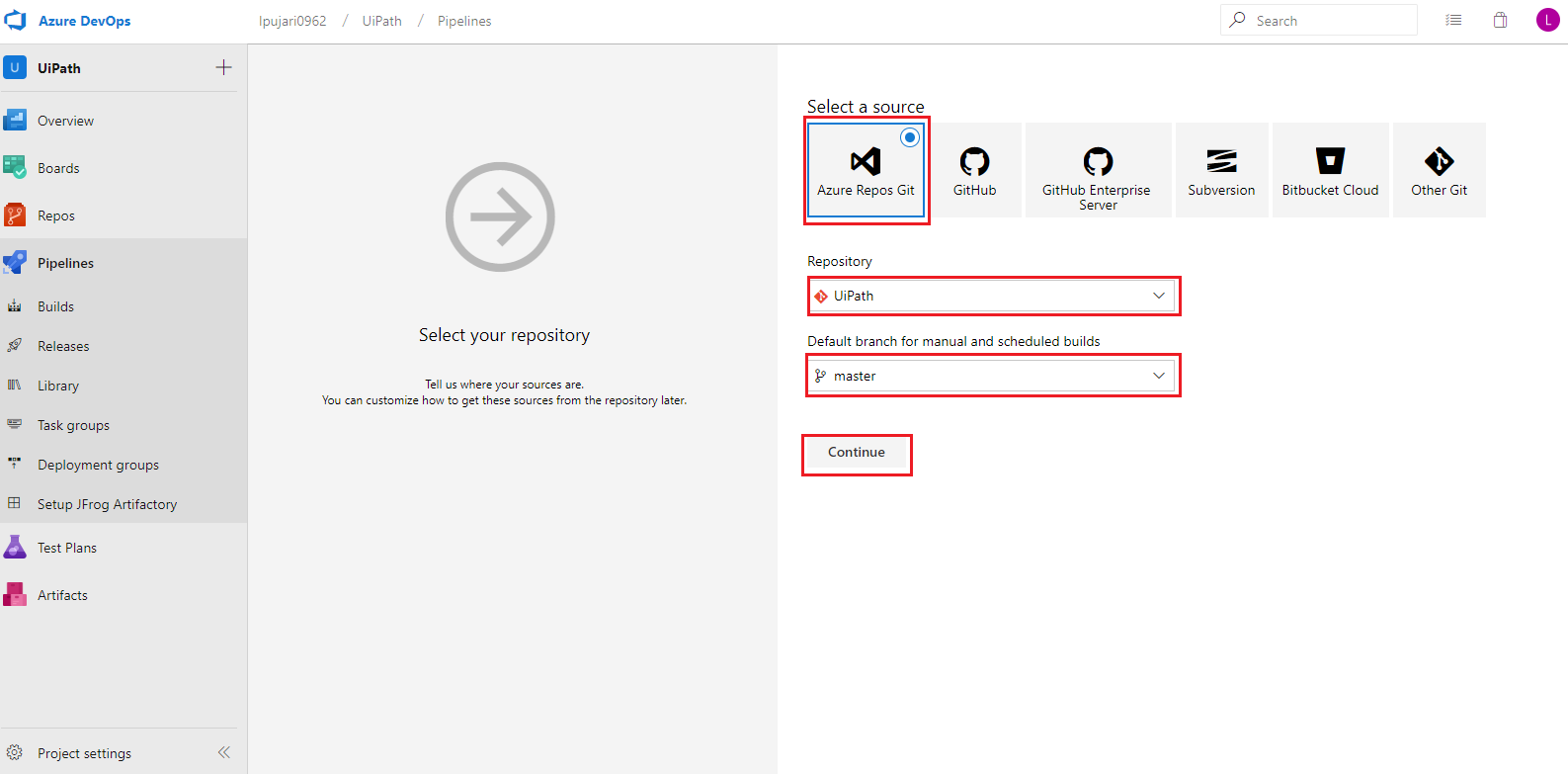
Now we can start setting up our pipeline for UiPath task by selecting build option in pipelines tab as shown below



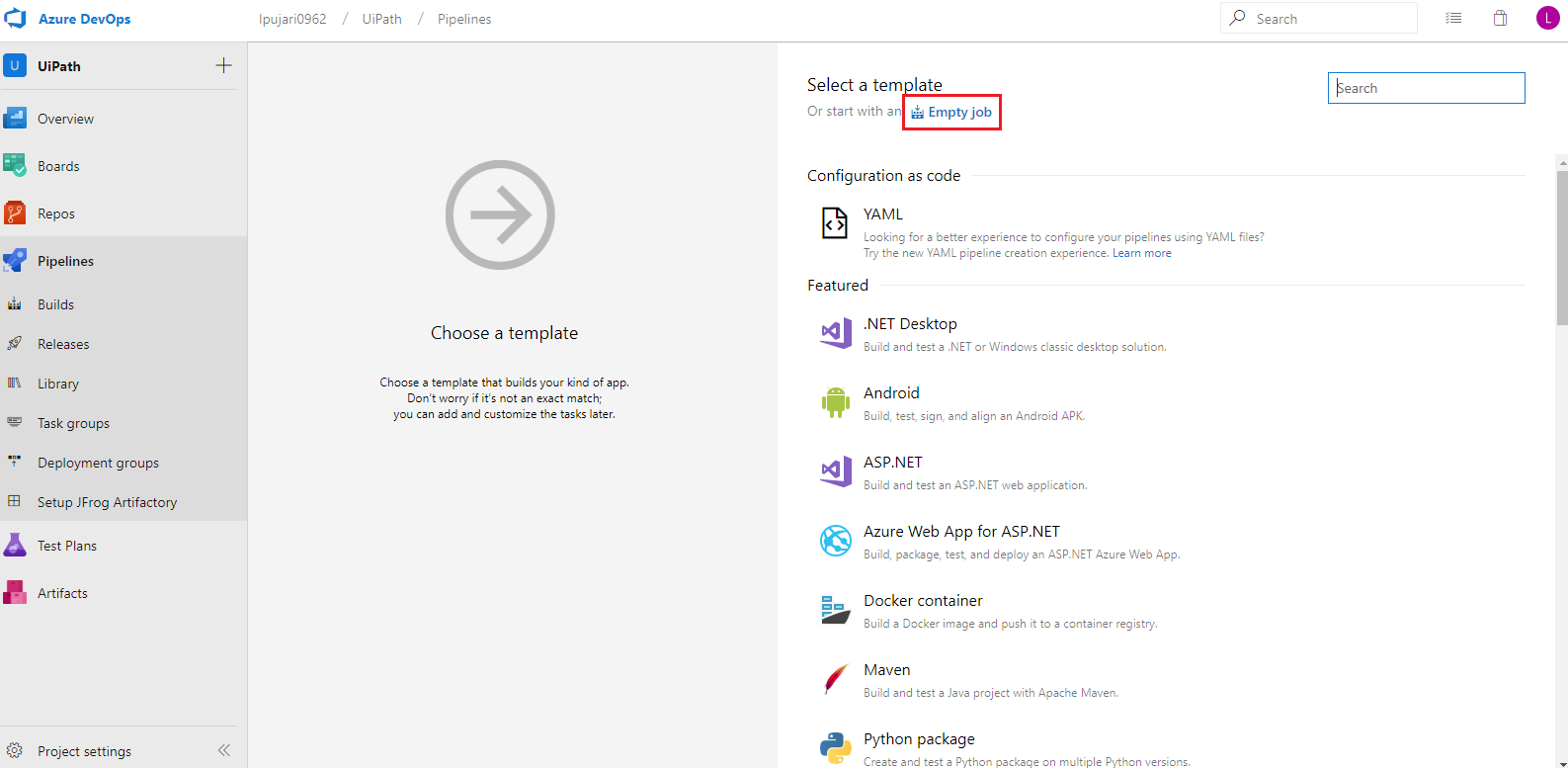
This will redirect us to the page where we can create a new pipeline by selecting **New Build Pipeline** option from the drop down menu as shown below.



On selecting new build pipeline, we will be provided options to select source code manager, from which we will be selecting Azure repos Git as our source code in VSTS. Once selected, we will be configuring the respective repository name and branch which contains our source code that going to create a NuGet package that need to be deploy in the UiPath Orchestrator and click on continue.

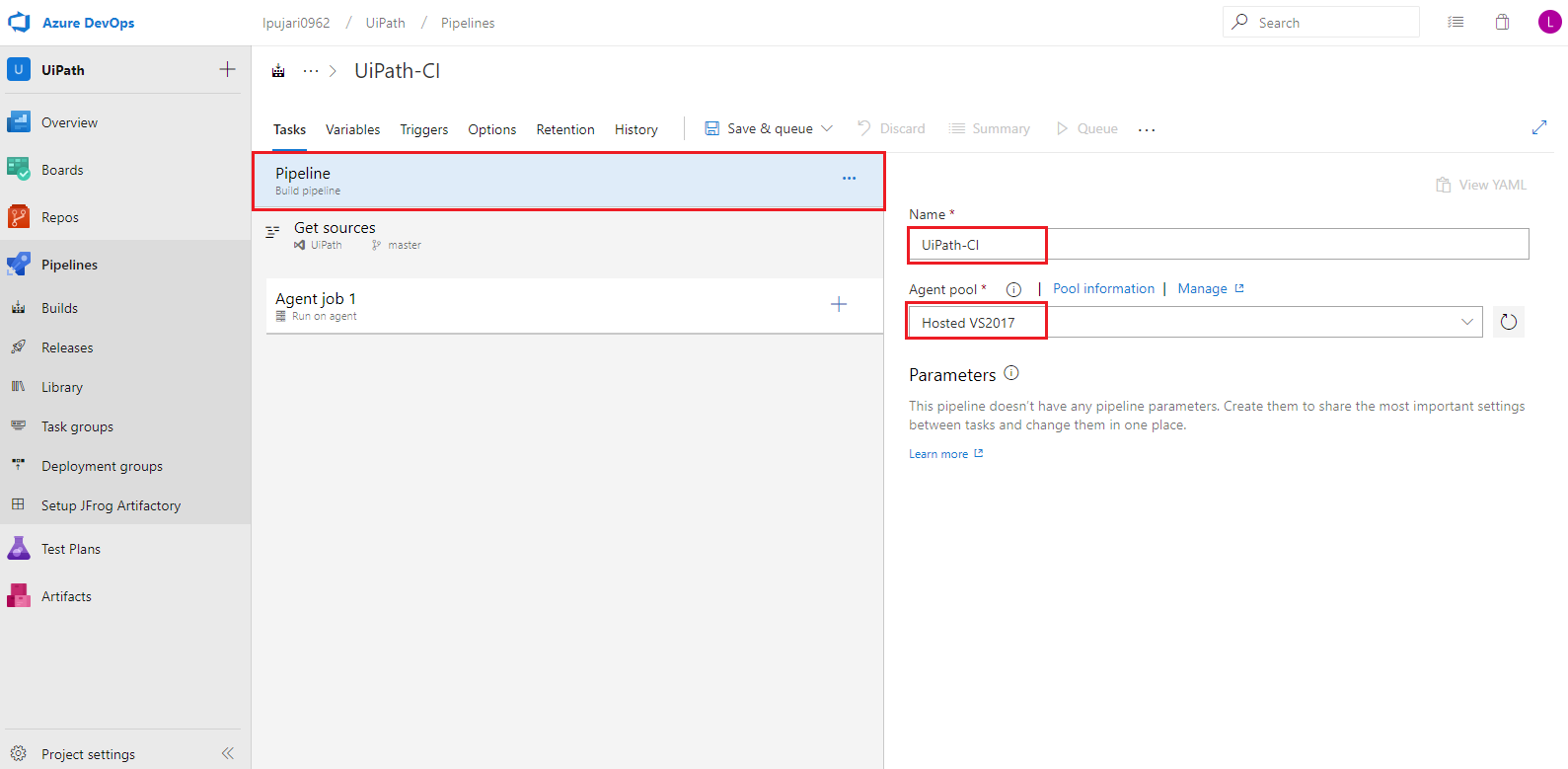


It will redirect us to a dashboard for configuring the build steps along with some sample templates, among which we will be selecting empty job option as shown below.

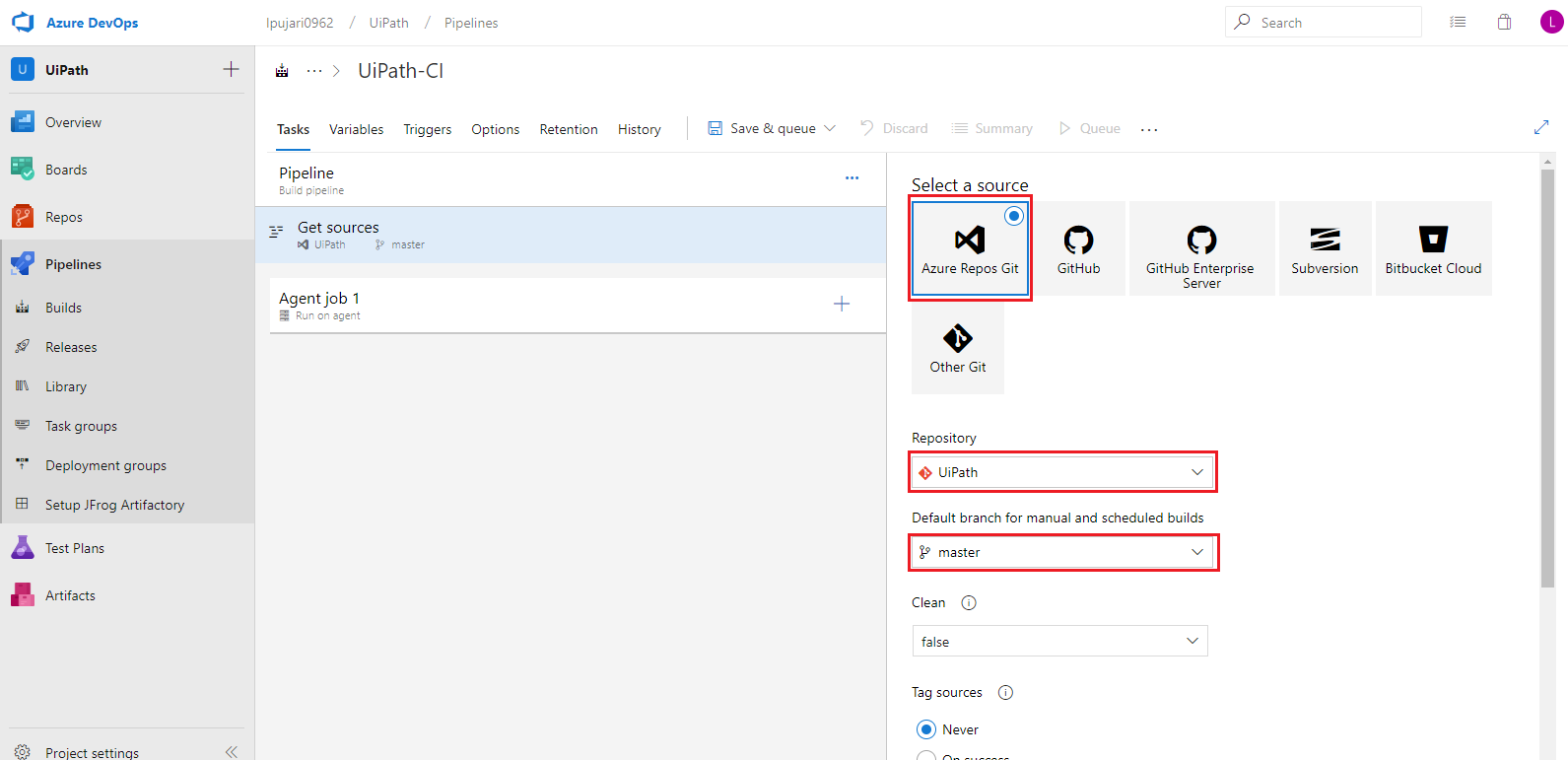


After selecting empty job, we need to provide respective job name and select an agent pool in which we would like to run the pipeline.

Here we have configured the build pipeline name as “**UiPath-CI**” and to queue our builds here we are selecting Microsoft-hosted agent as “**Hosted VS2017**” which is provided by VSTS.



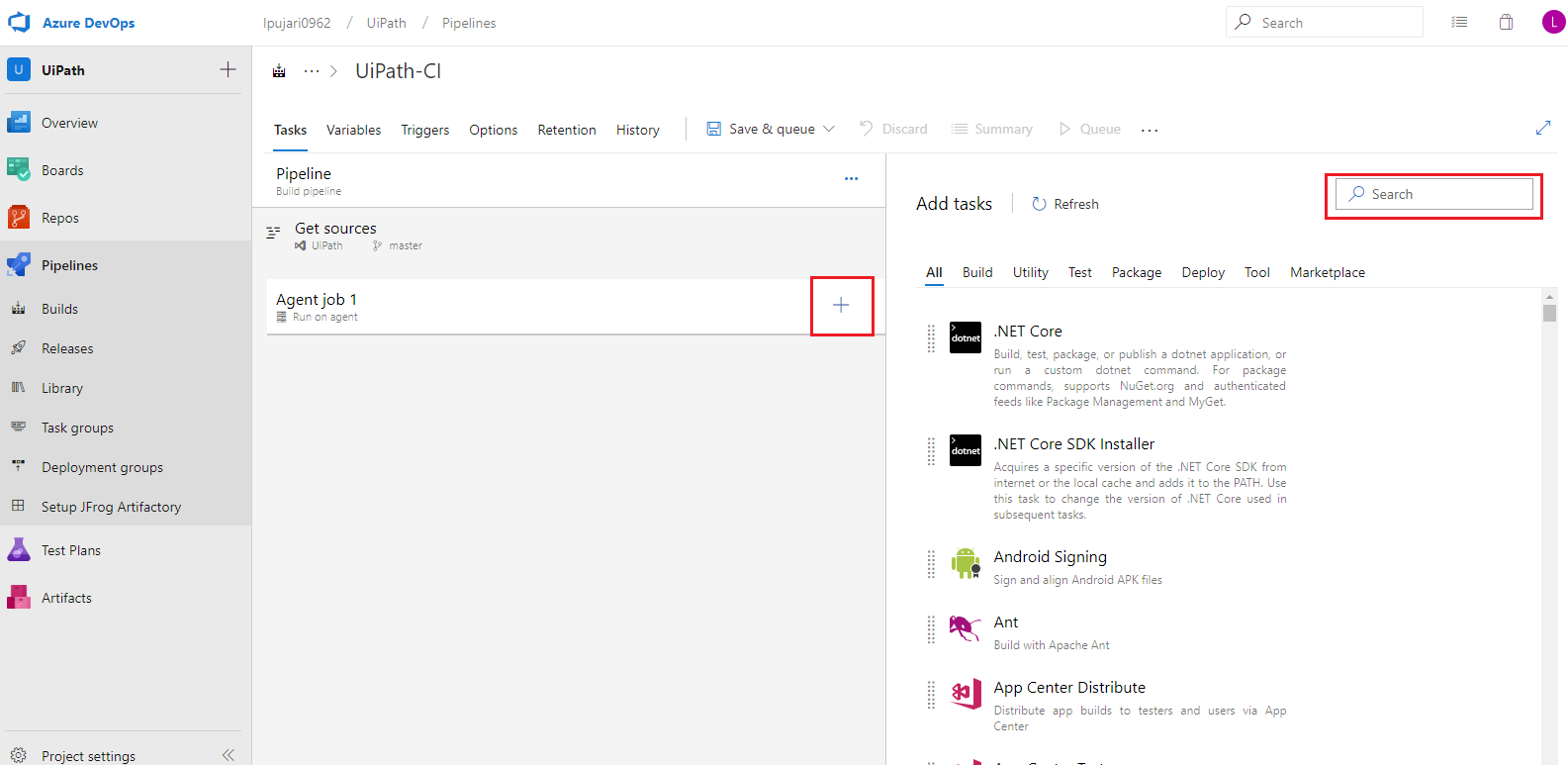
Now in Get Sources section we need to configure the type of the source, repository name and branch to get the source code.



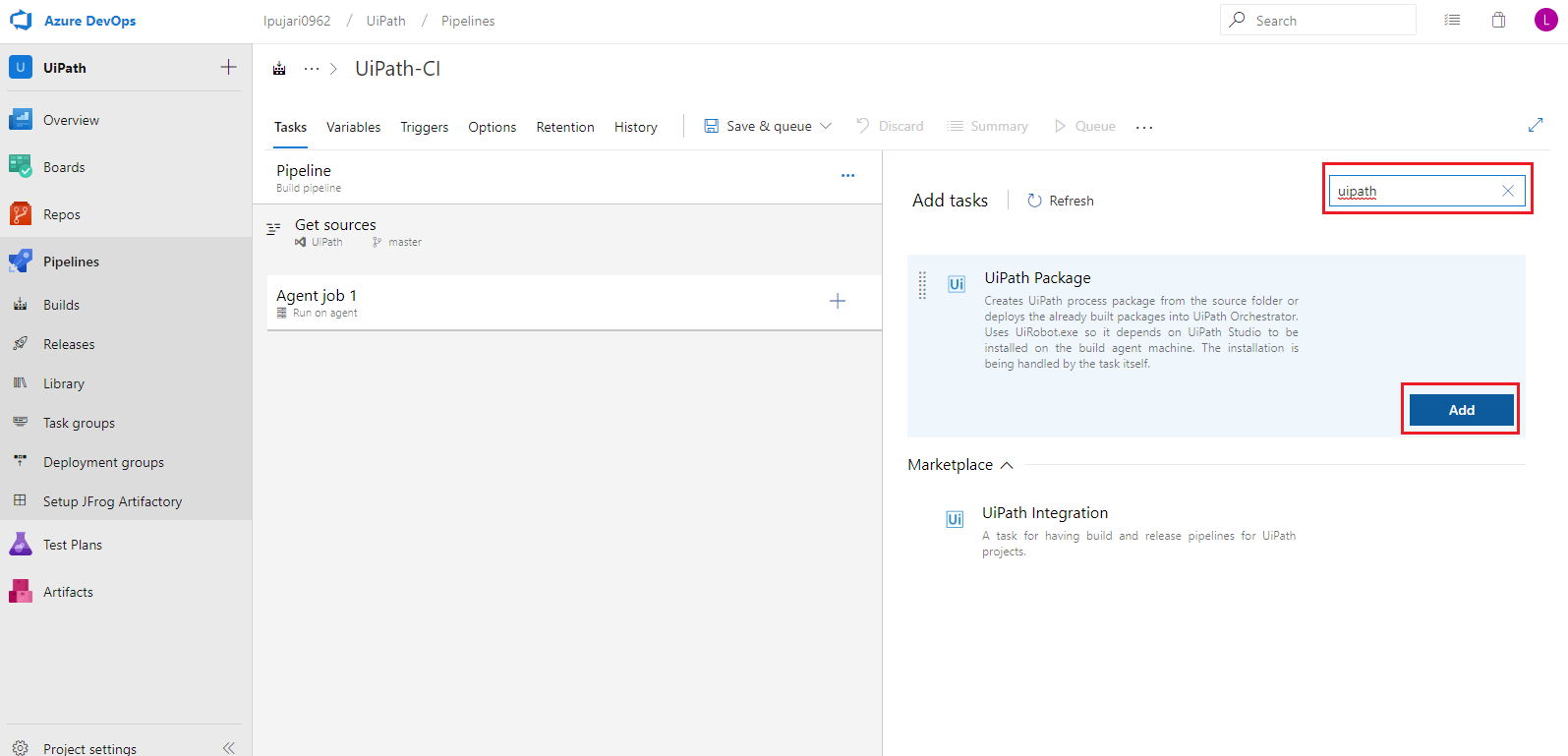
Now we need to add task to the build pipeline which helps us in building the application.

To add a new task, click on + symbol next to Agent phase.

On Clicking **+** button we can now see the tasks added to our Agent phase. Here we will search for required task as per the requirement.

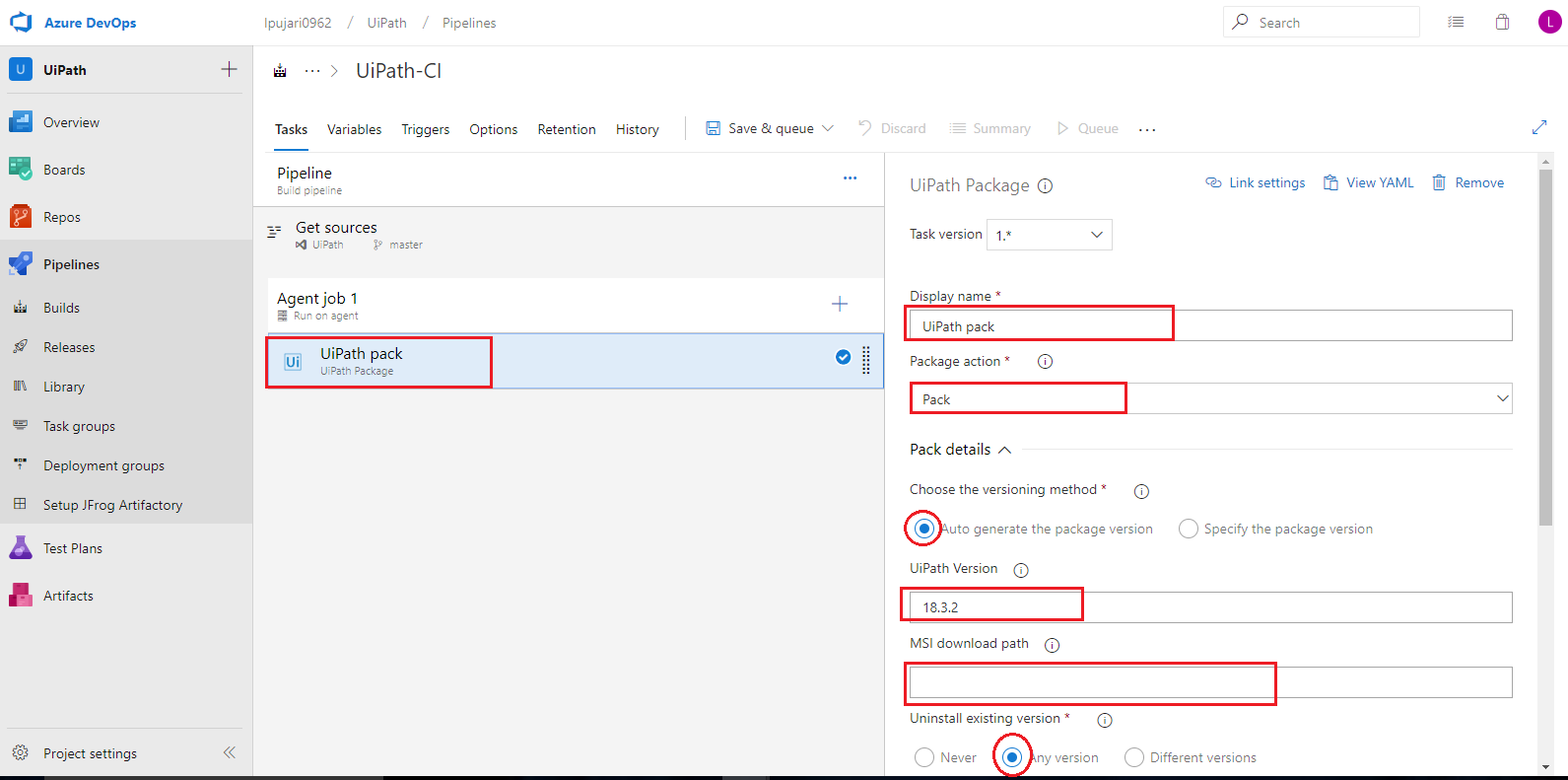


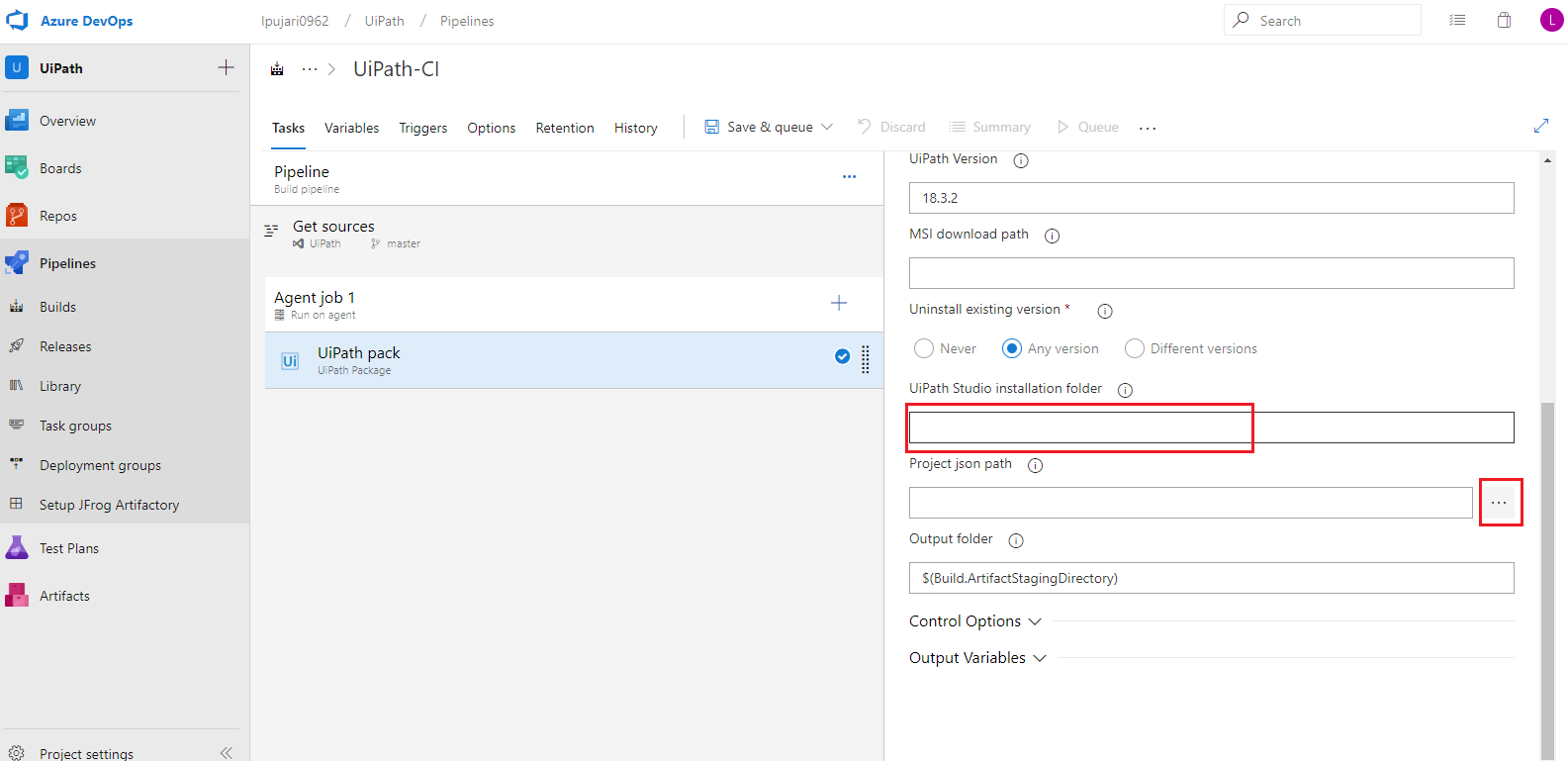
On selecting the Uipath task we could able to configure the Uipath build step as shown:

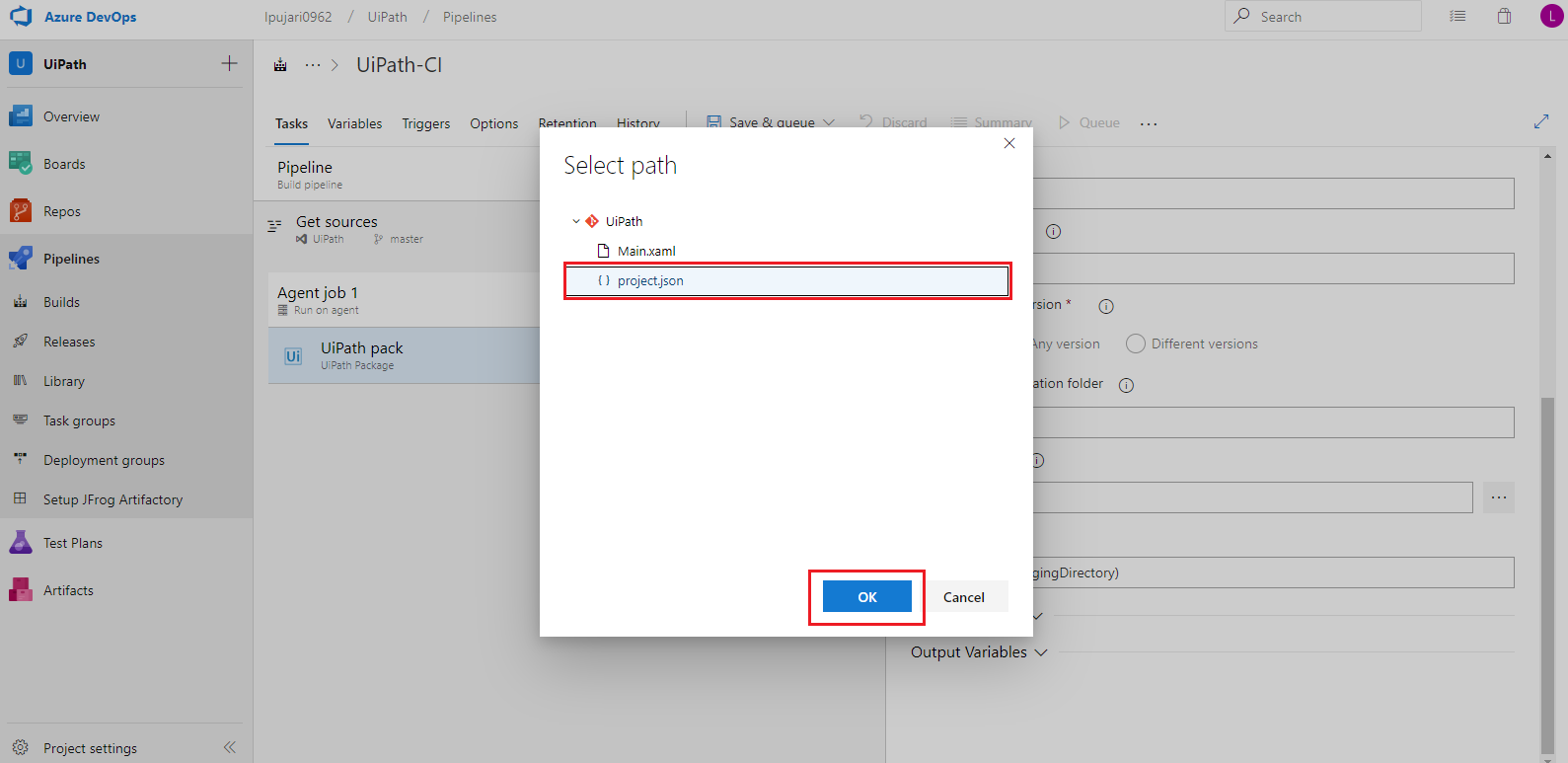


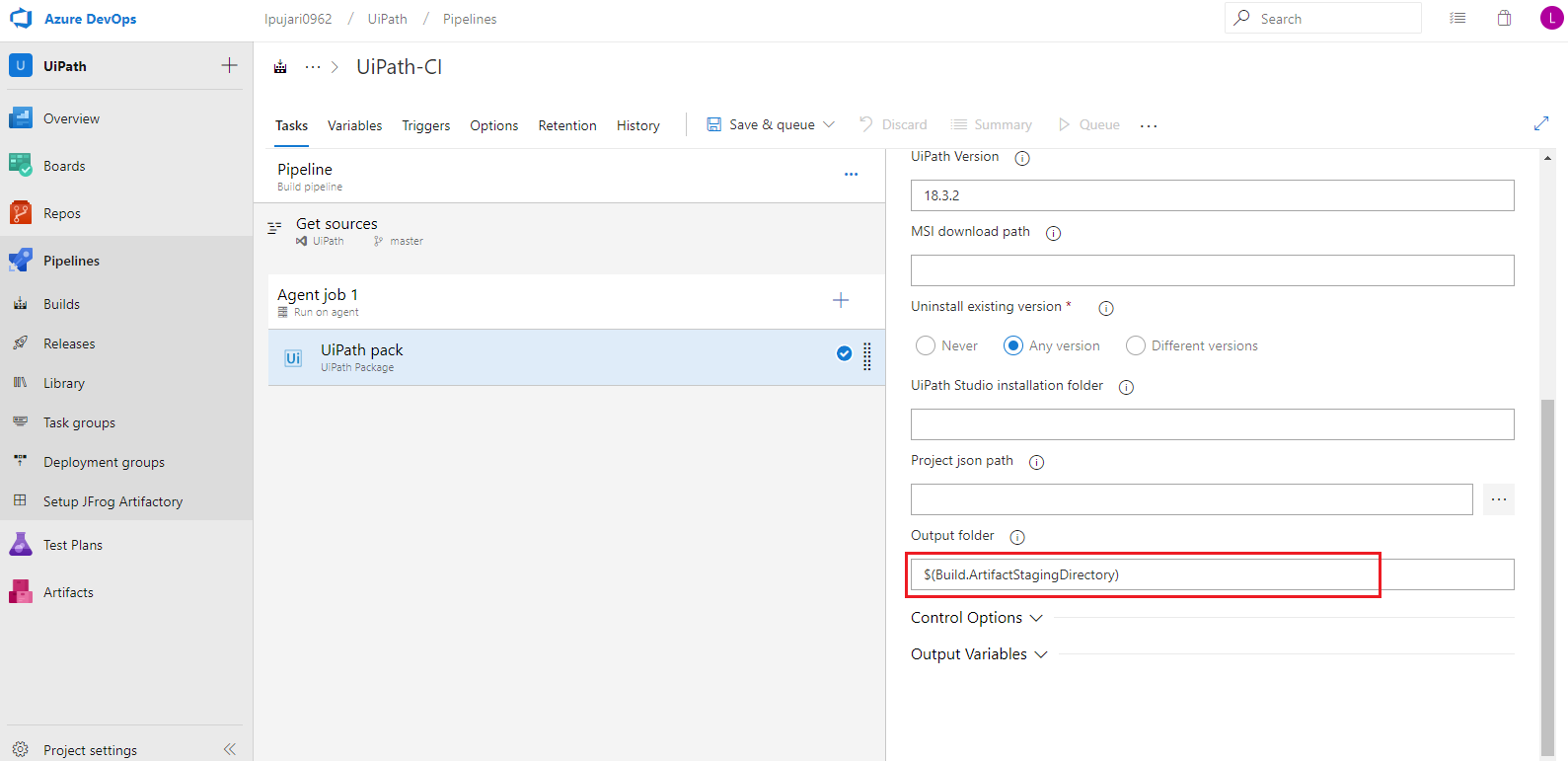
**1. UiPath Pack**

We have used UiPath Pack task to generate NuGet package file.





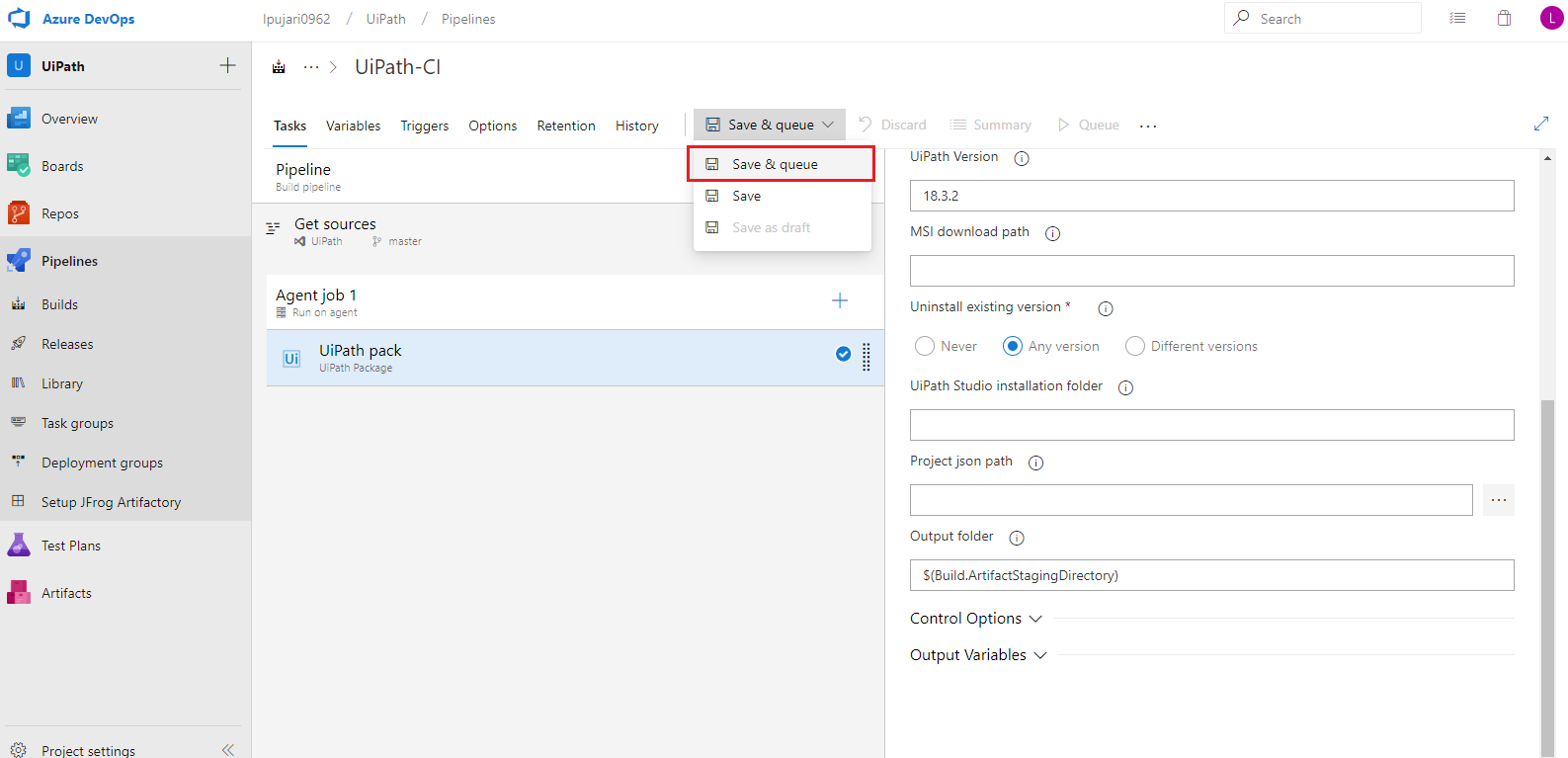




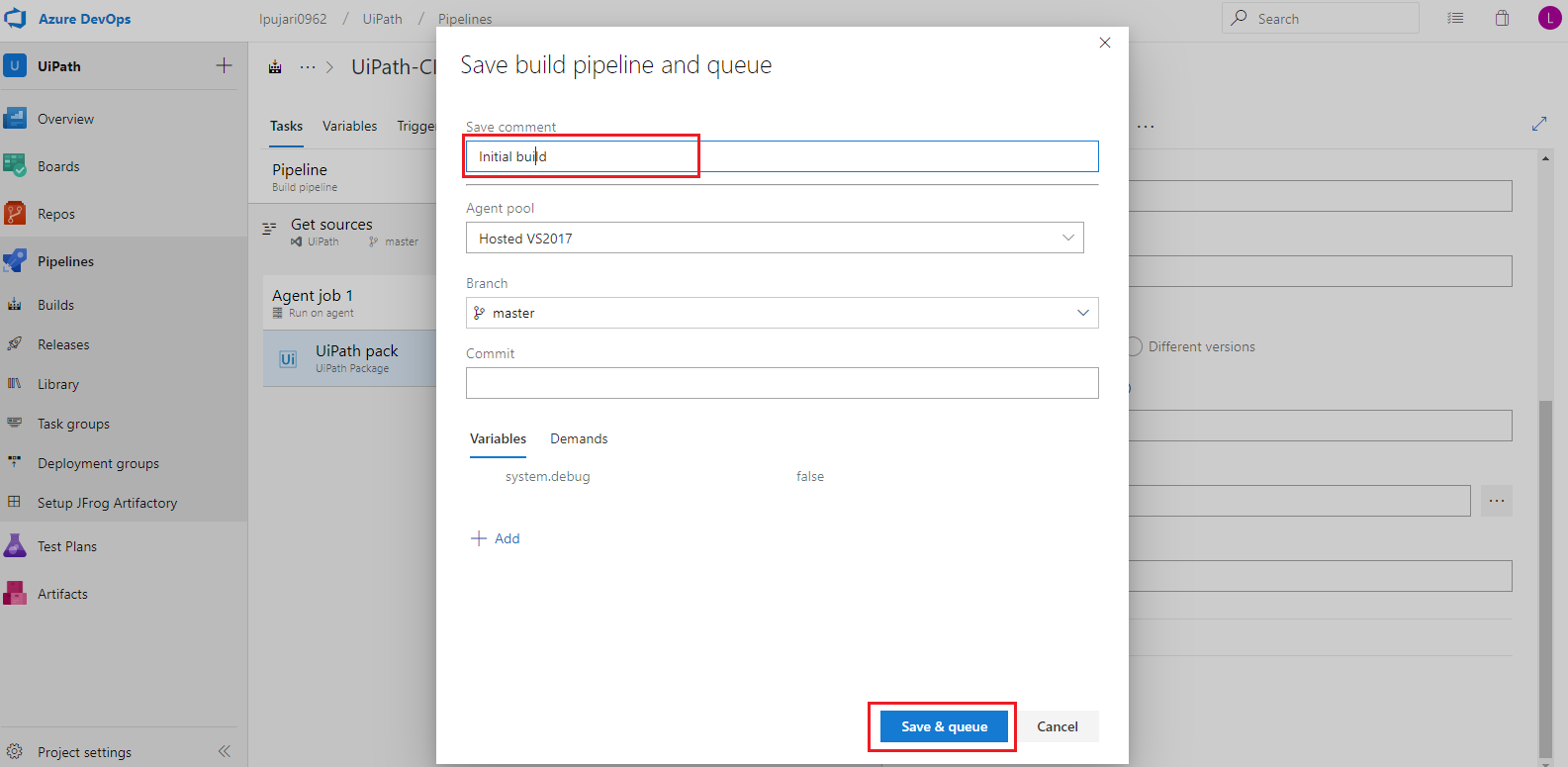
Arguments for the UiPath pack build step:

* **Display Name:** Here we specify the name for the task
* **Package Action:** Here we are giving the action for the task. It will generate the nupkg file.
* **Choose the version method:** We have configured it as auto generate package version. It will generate a new version of the package according to the build.
* **UiPath Version:** It will specify the version of the UiPath studio and UiPath robot to install
* **MSI download path:** specify Path where to download UiPath studio. Here we left it blank. So, it will be downloaded in the default path.
* **Uninstall existing version:** It will uninstall the previously installed UiPath studio versions in the agent.
* **UiPath Studio installation folder:** It will specify the path to install UiPath studio. Here It will be installed in the default path.
* **Project json path:** It will redirects to where the project.json file was stored in the repository folders.
* **Output folder:** Where to generate the nupkg file. Here we have configured it as default.

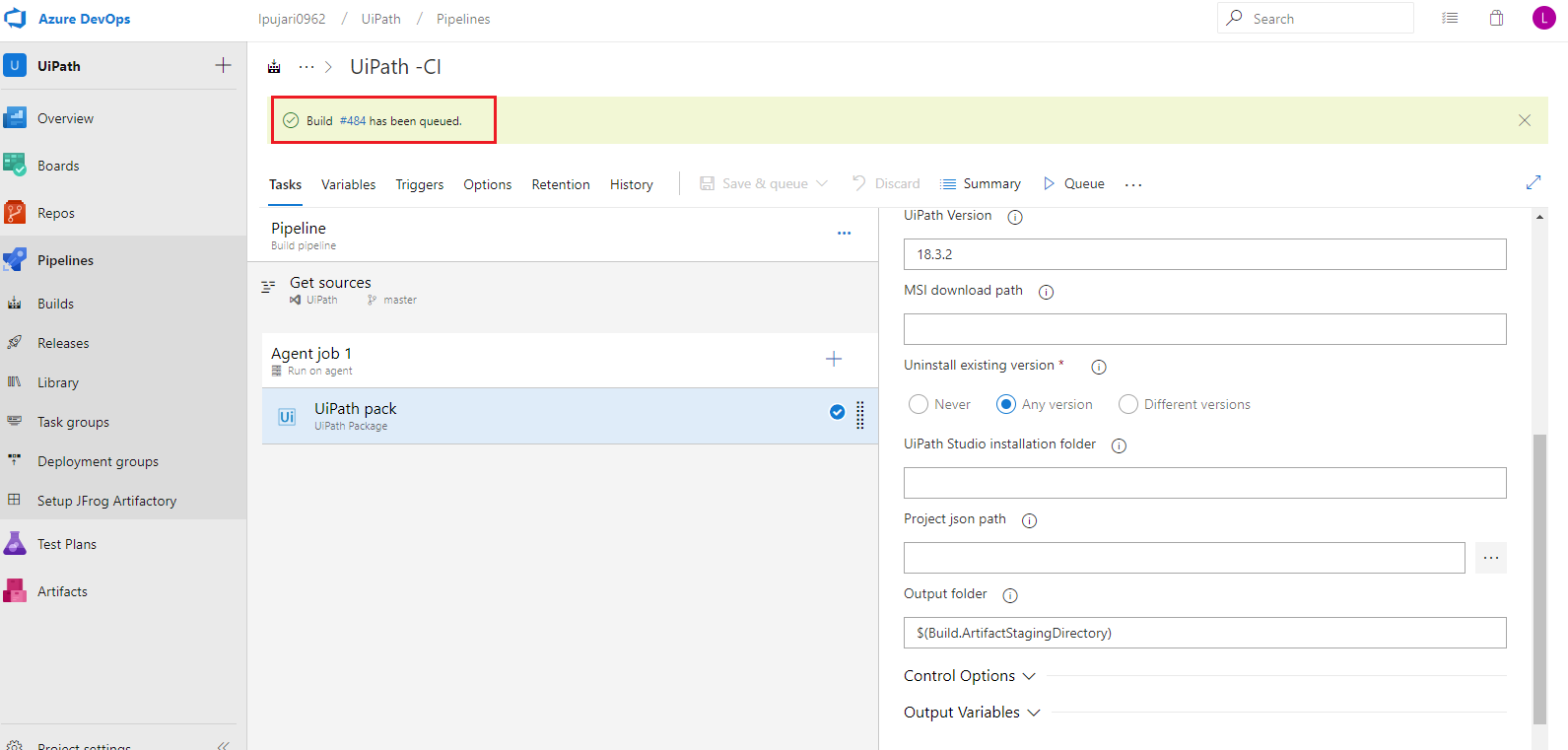
Now save the changes and queue the pipeline to check weather the nupkg file was generating or not and if it was generating then to check the path of the nupkg file.



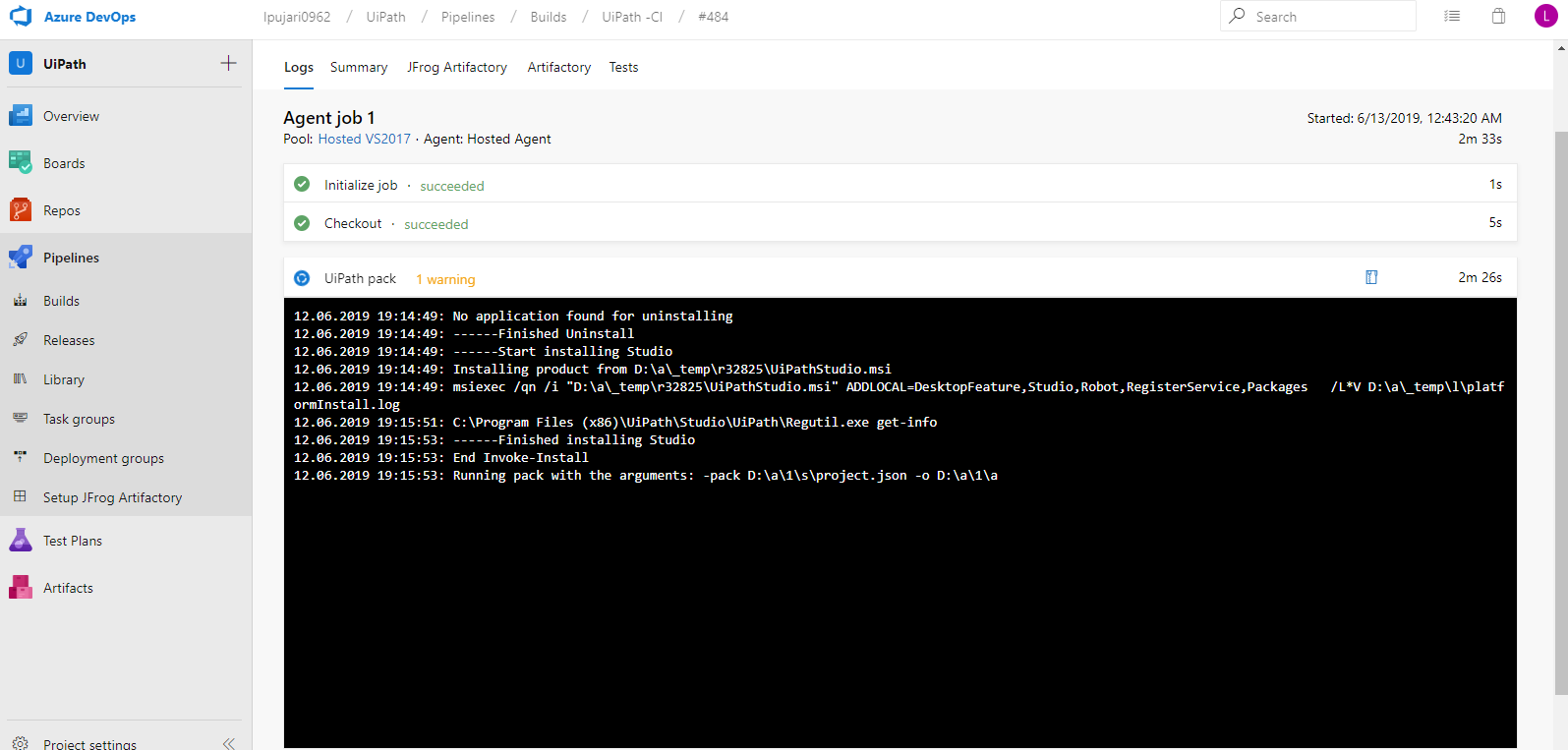
Now give any save comment and click on save & queue



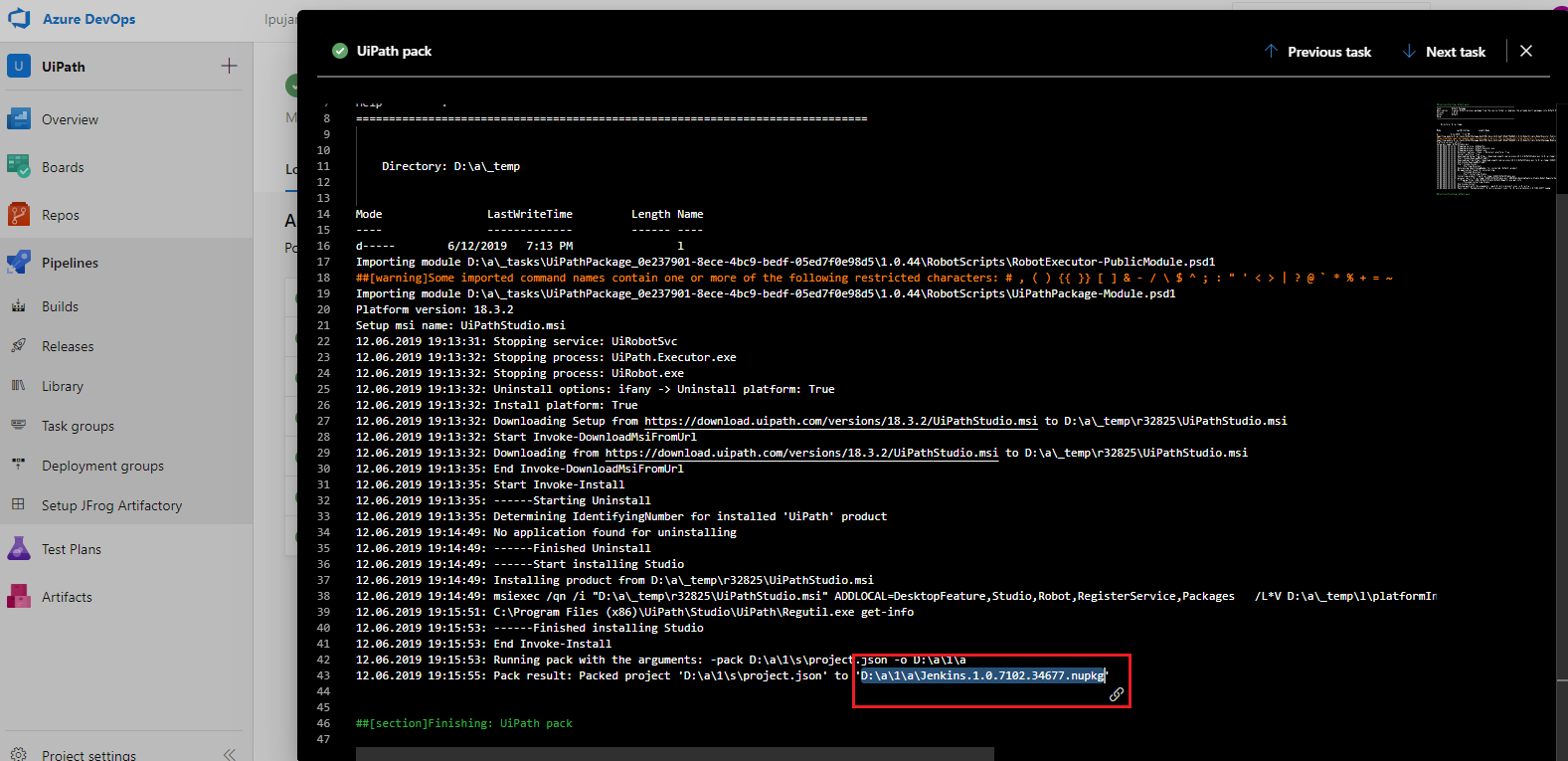
Now the build will be queued and then click on the **build number** to check the logs to find the progress.



Here build is queued and we can check the progress in the logs.

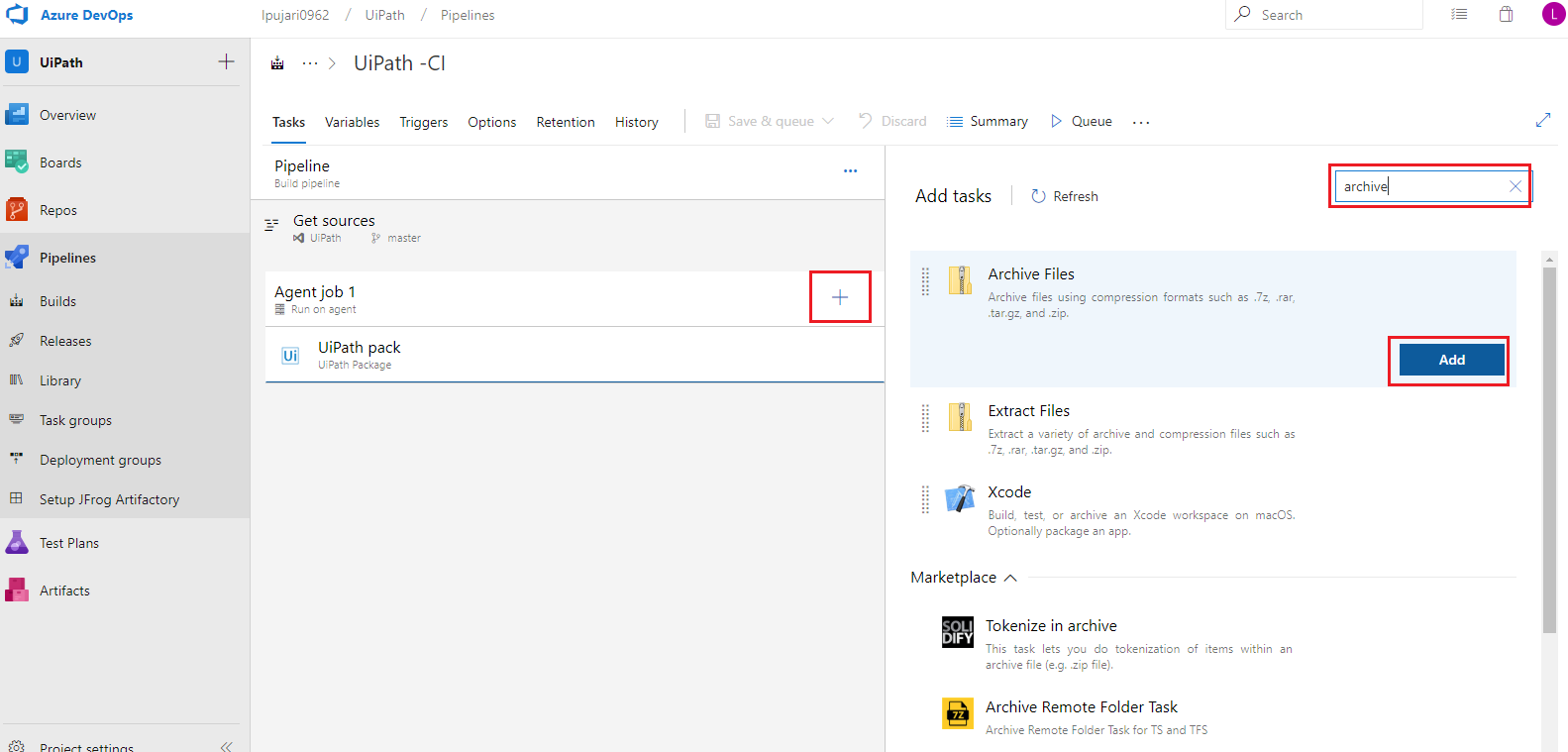


Here package got successfully generated in the following path.

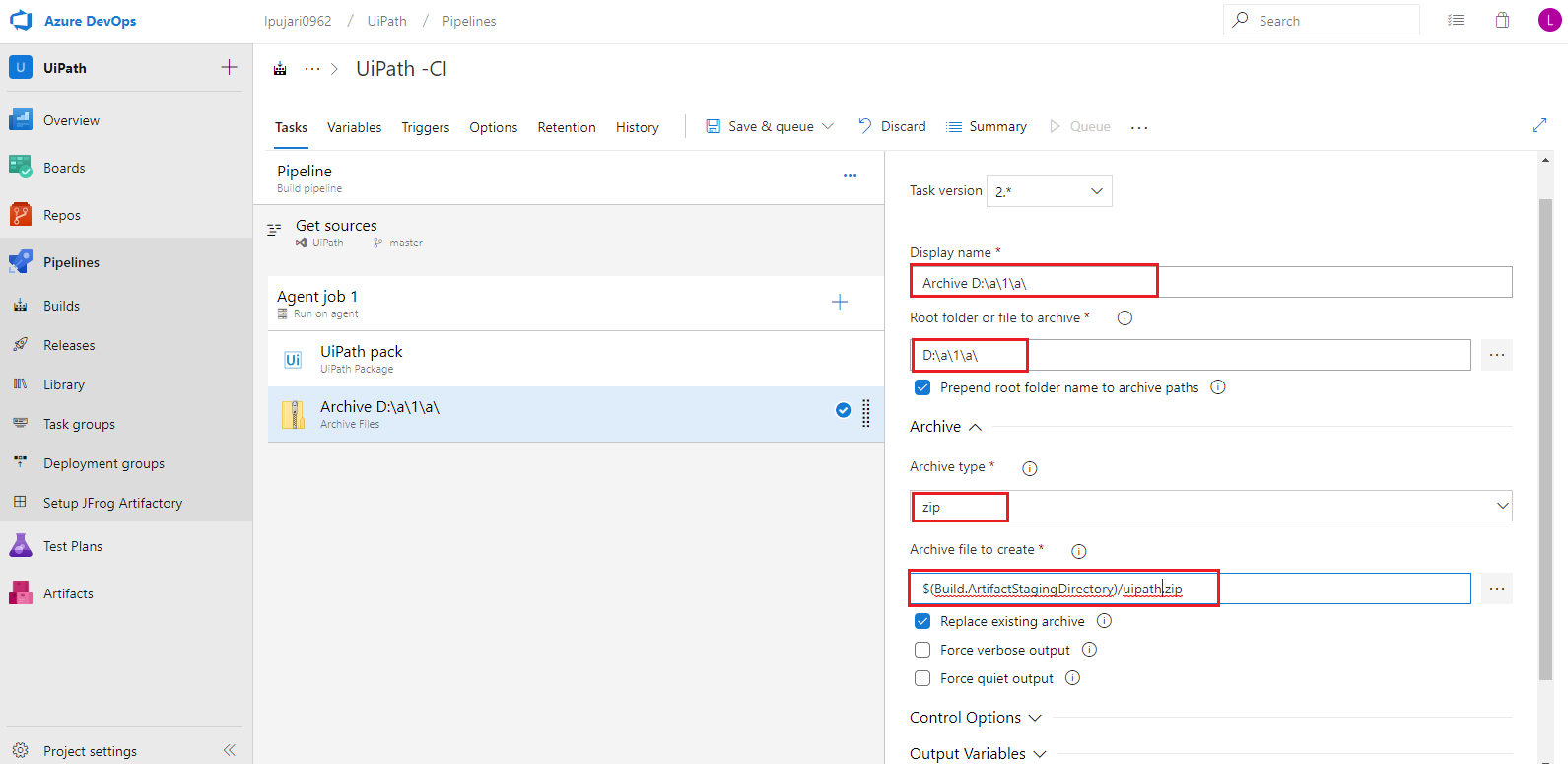


**2. Archive Files**

Click on **+** symbol next to Agent phase to add new task to the build. In this task we are going to archiving the NuGet package file.



We have added Archive task to zip the **NuGet package** file.



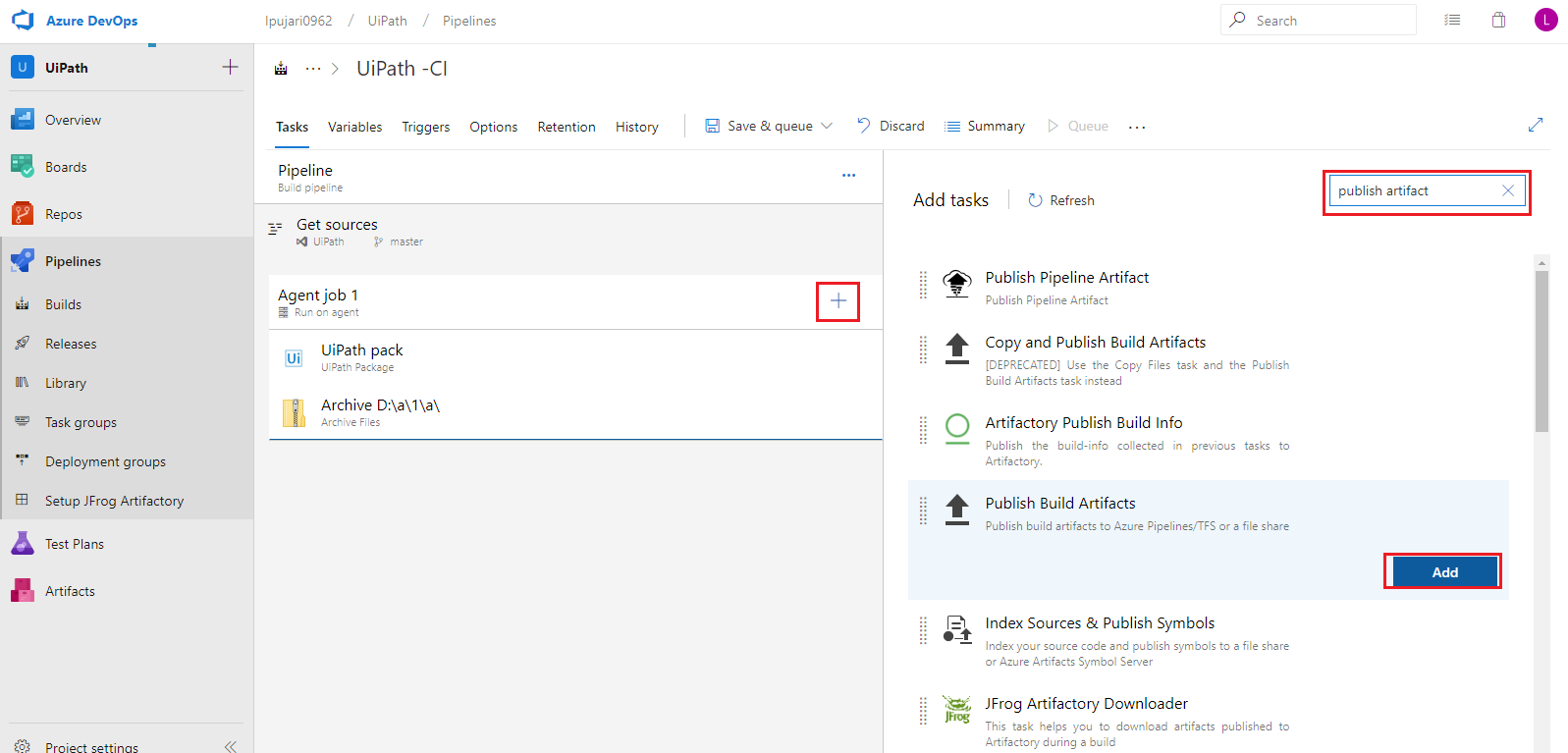
Arguments for the Archive Build step:

* **Display Name:** Name of the task
* **Root folder or file to archive:** We will configure it with the nupkg file path
* **Archive type:** We are going to archive the file into zip format
* **Archive file to create:** Specify a name to the archived file.

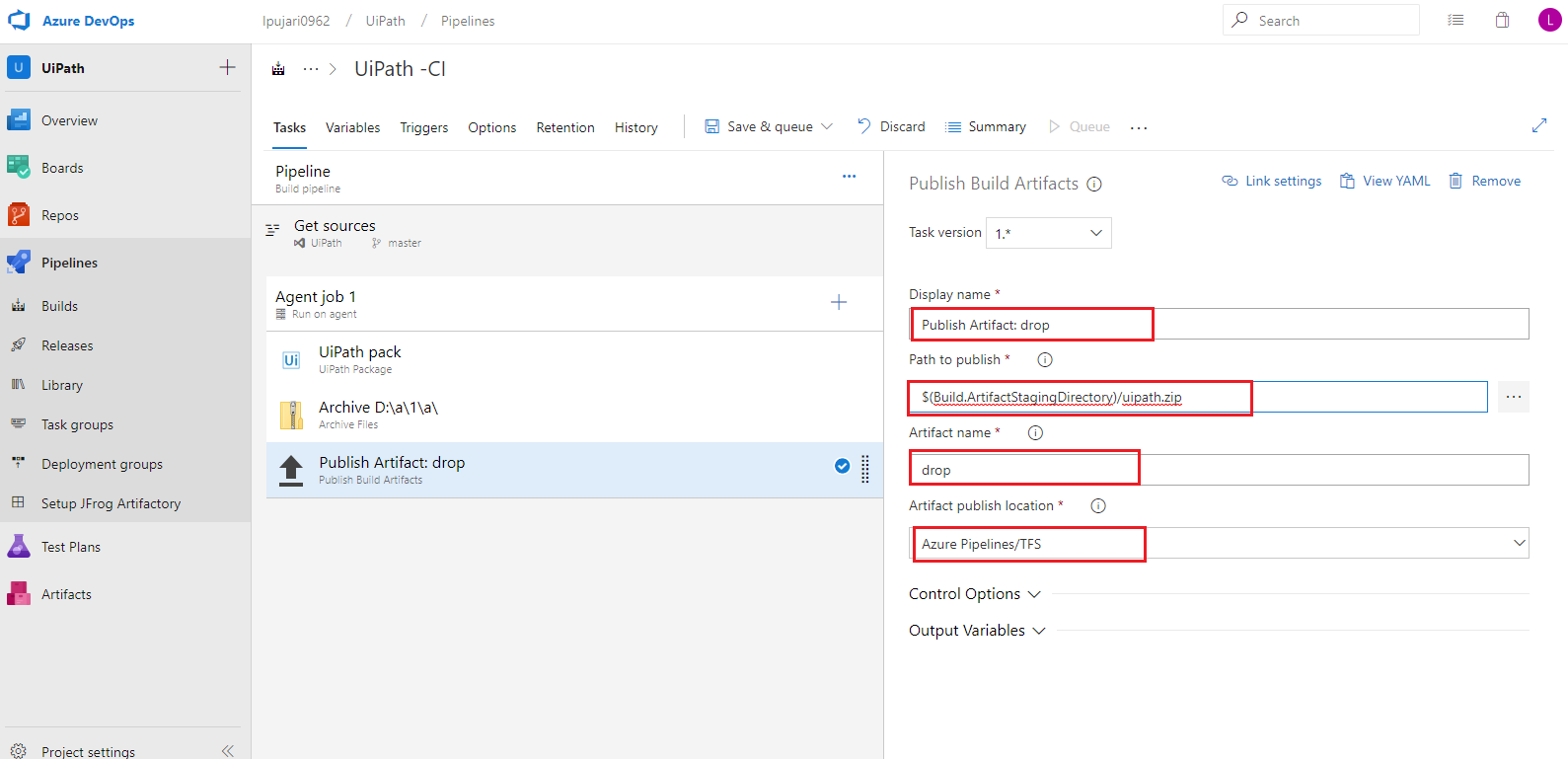
**3. Publish Build Artifact**

Pipeline artifacts help you store build outputs and move intermediate files between stages in your pipelines. Artifacts are the files that you want your build to produce.

Click on **+** symbol next to Agent phase to add new task to the build. Here we are adding publish artifact task.



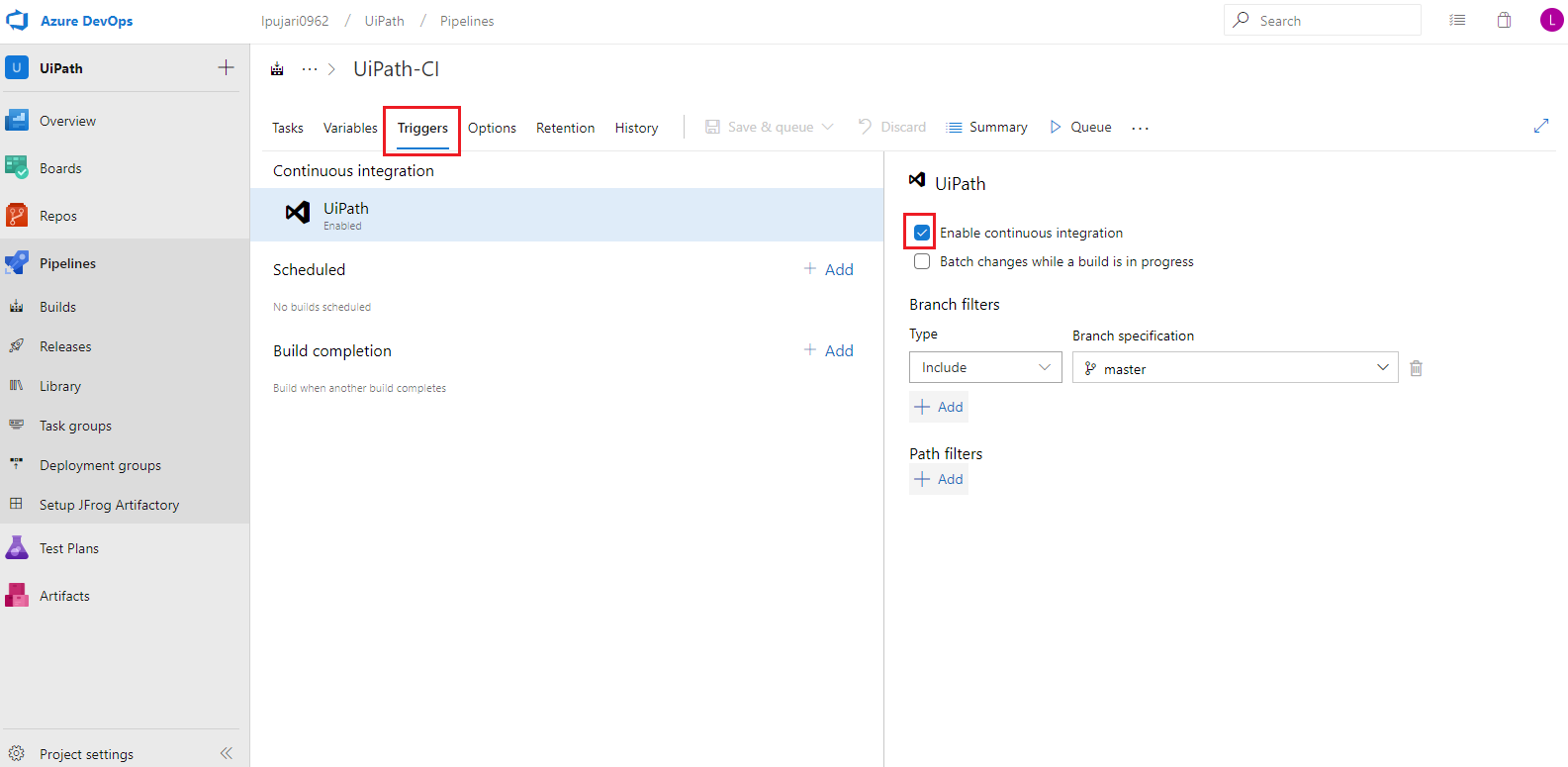
We have added Publish artifact task to publish the output file.



Arguments for the Publish Artifact Build step:

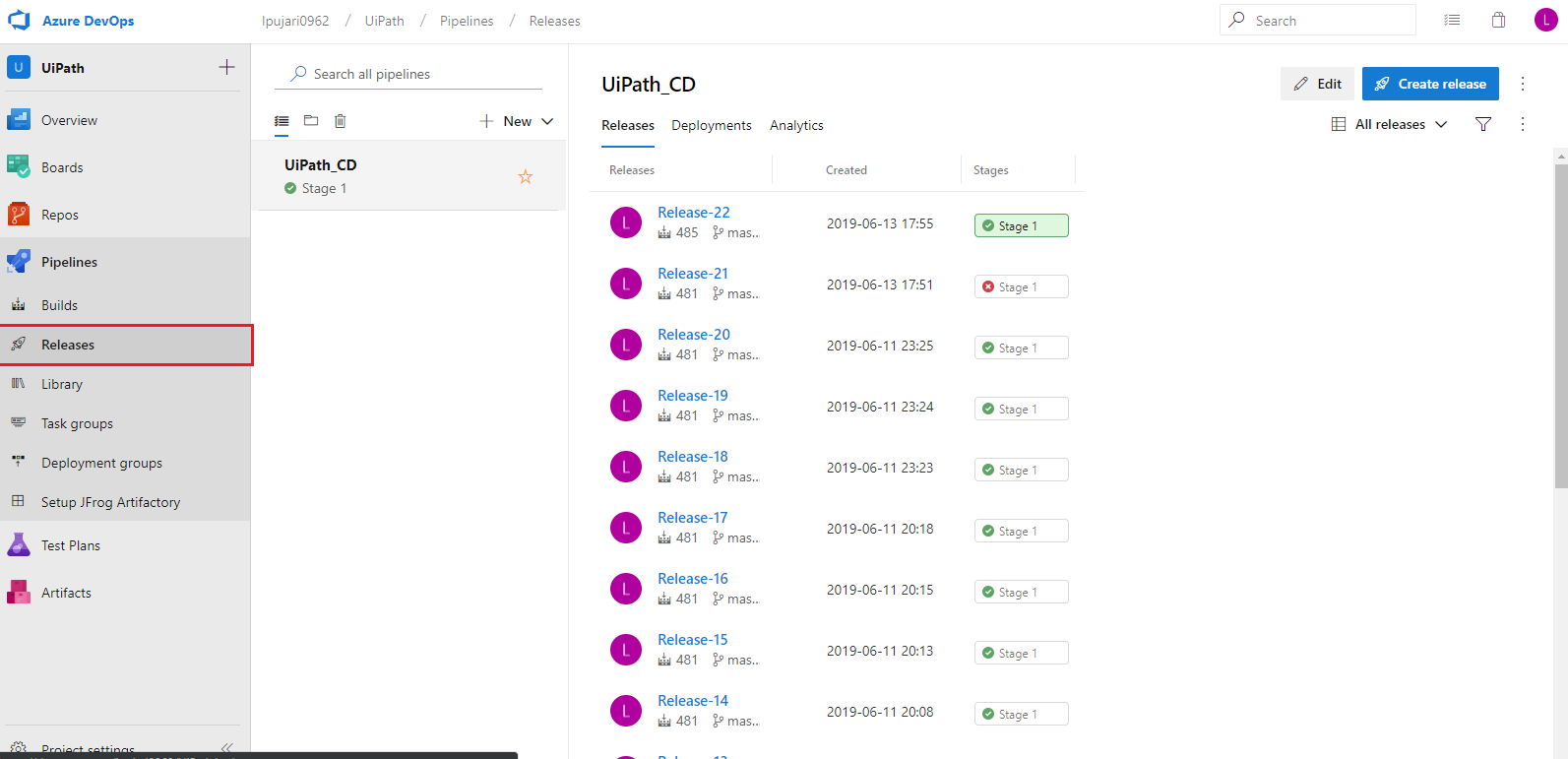
* **Path to publish:** Path of the file to publish
* **Artifact Name:** Name for the artifact
* **Artifact Publish Location:** Location to publish the artifact

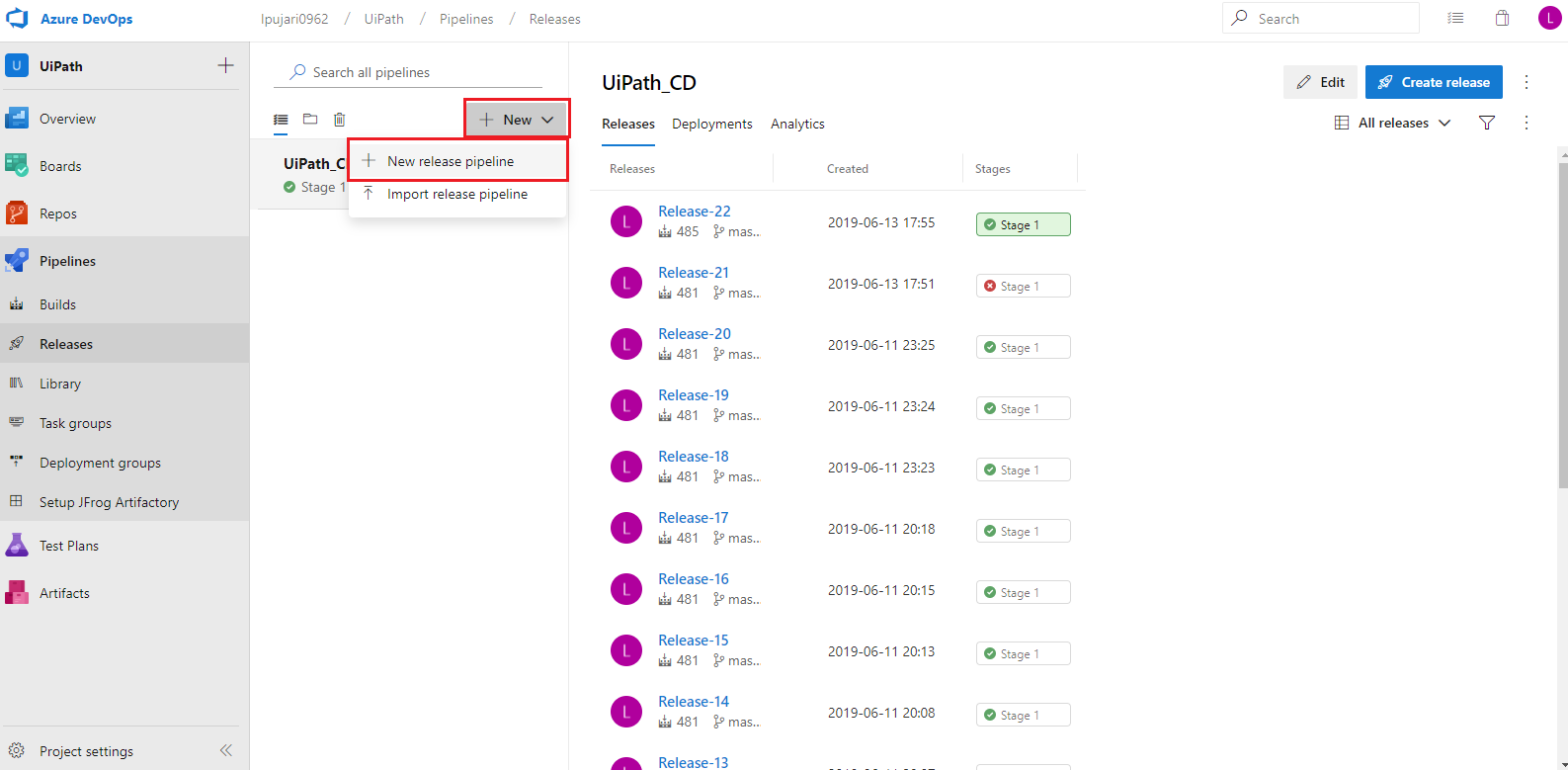
To enable the continuous integration whenever new code was committed goto triggers section and click on checkbox to enable **continuous integration**.



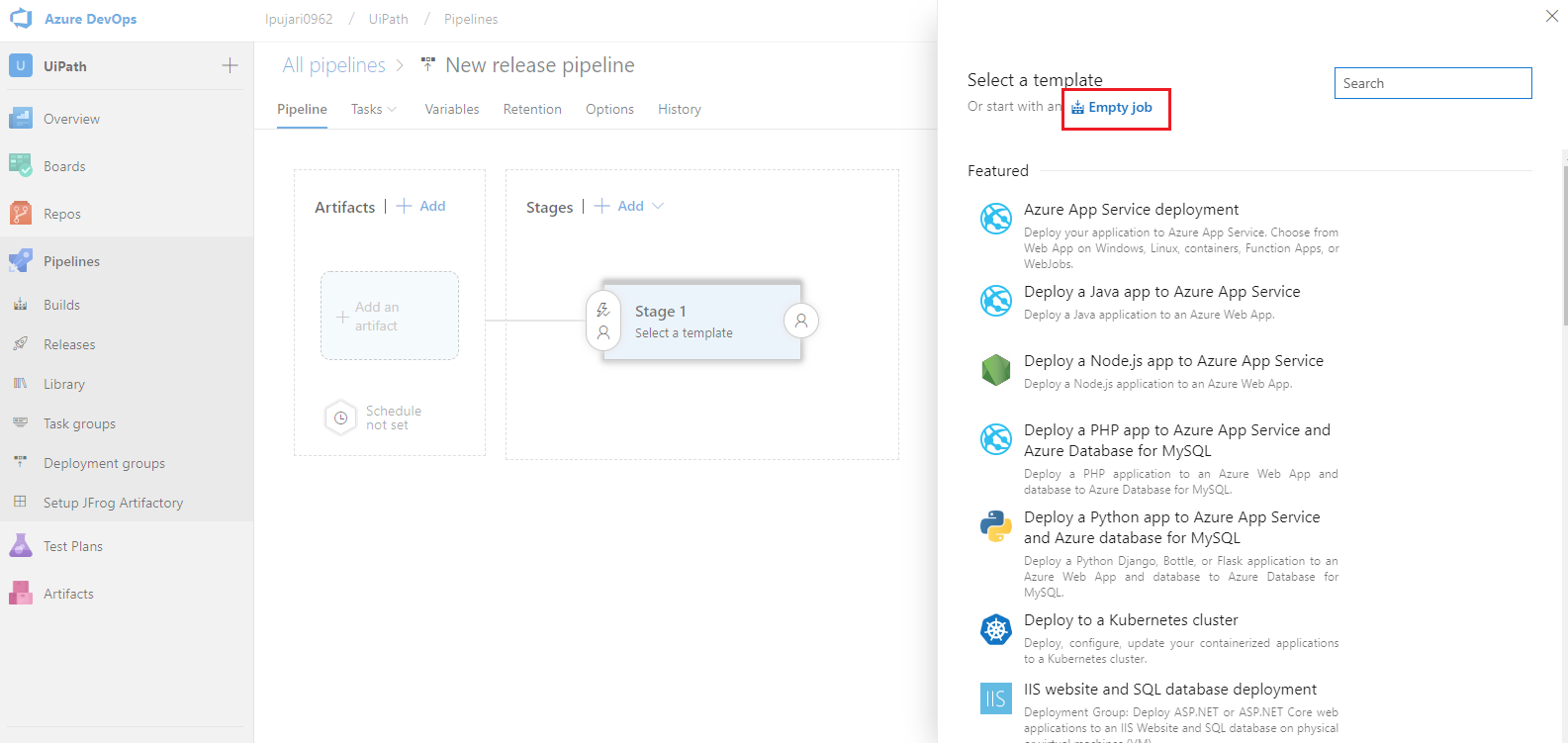
# Continuous Deployment pipeline using VSTS:

After successful completion of continuous integration pipeline, we will start continuous delivery pipeline by creating new release pipeline from the drop down box as shown below, here we will set our release pipeline for deploying the nupkg file to orchestrator.

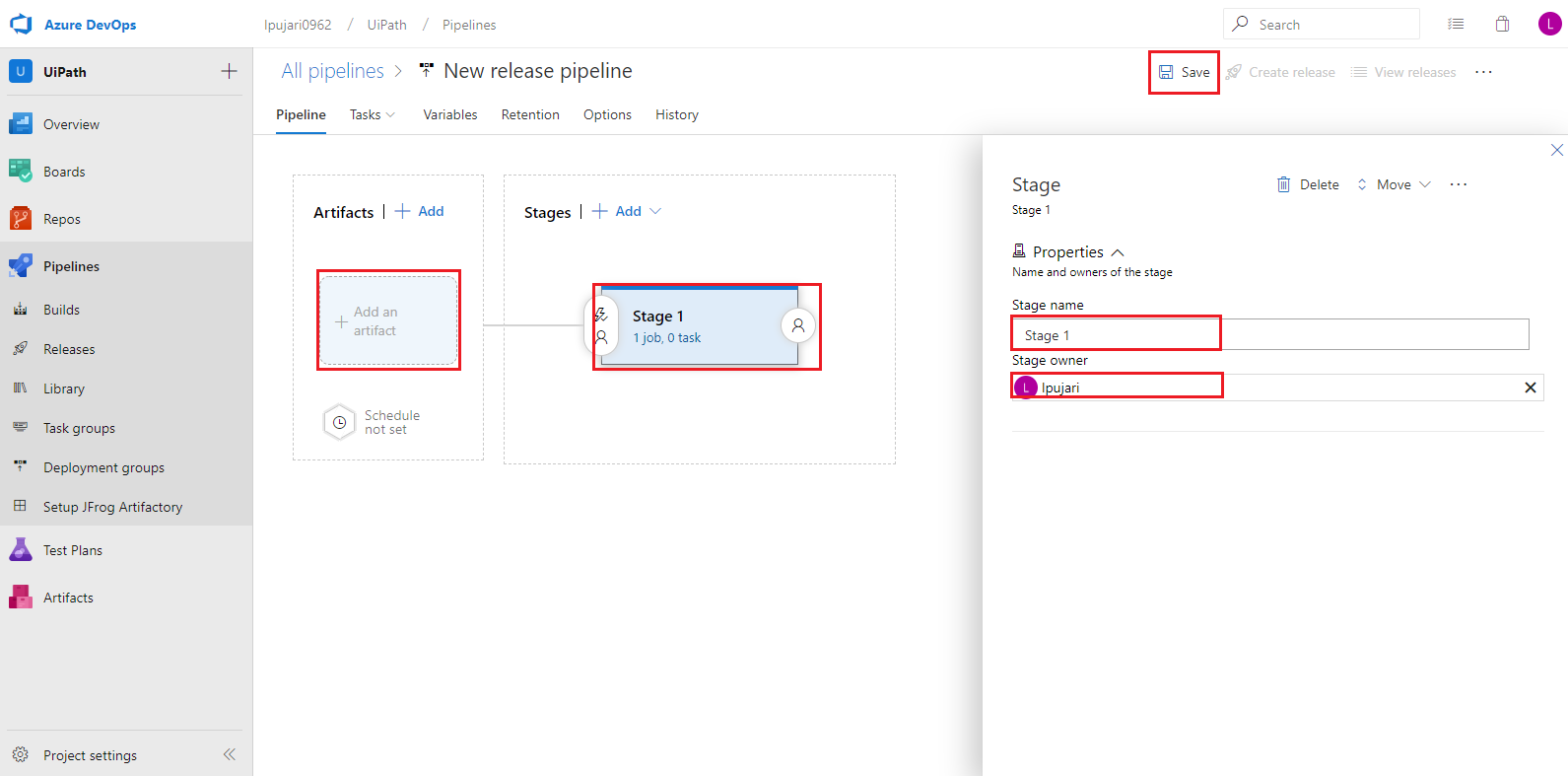




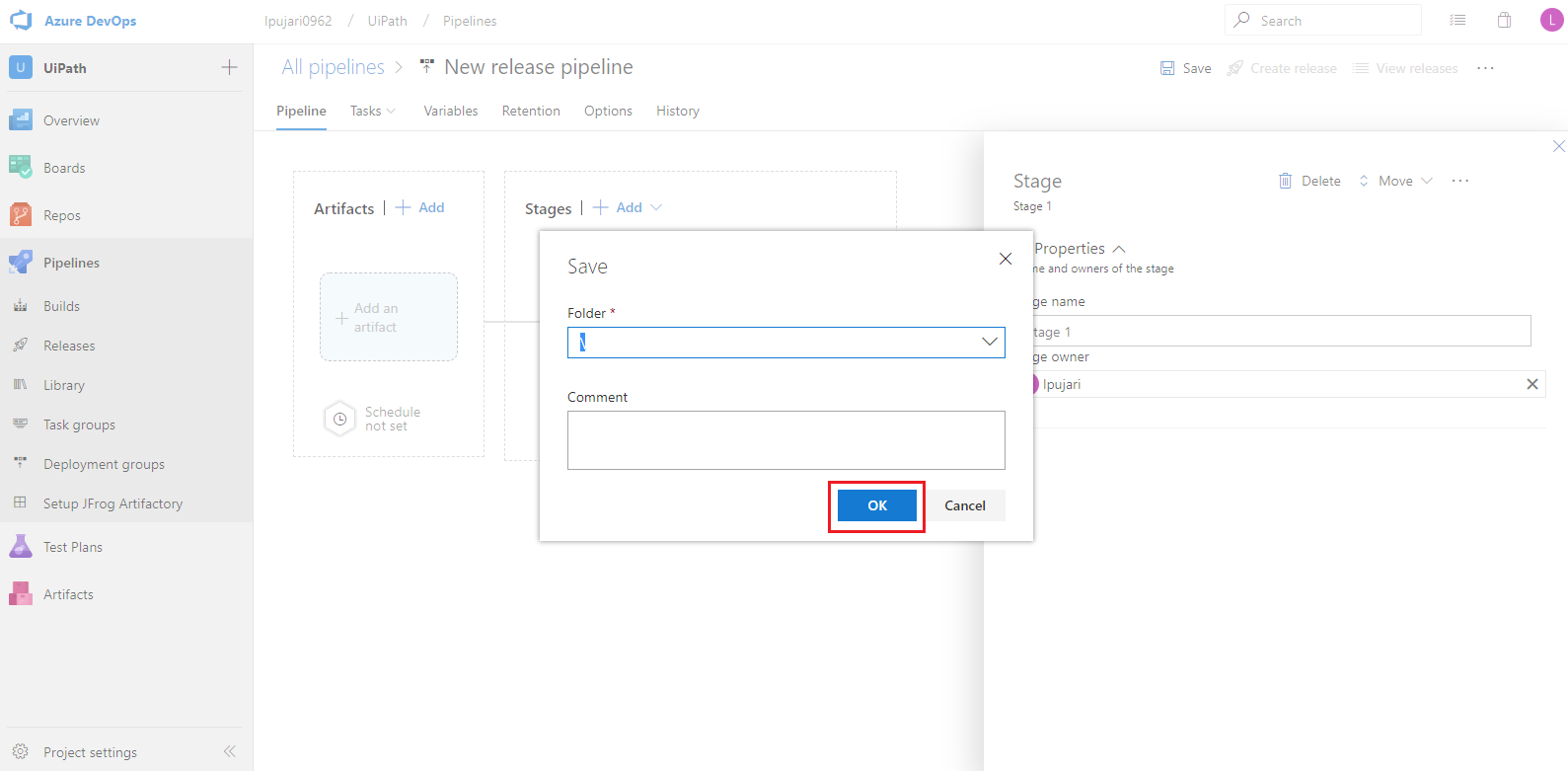
It will redirect us to a dashboard for configuring the release steps along with some sample templates, among which we will be selecting empty job option as shown below



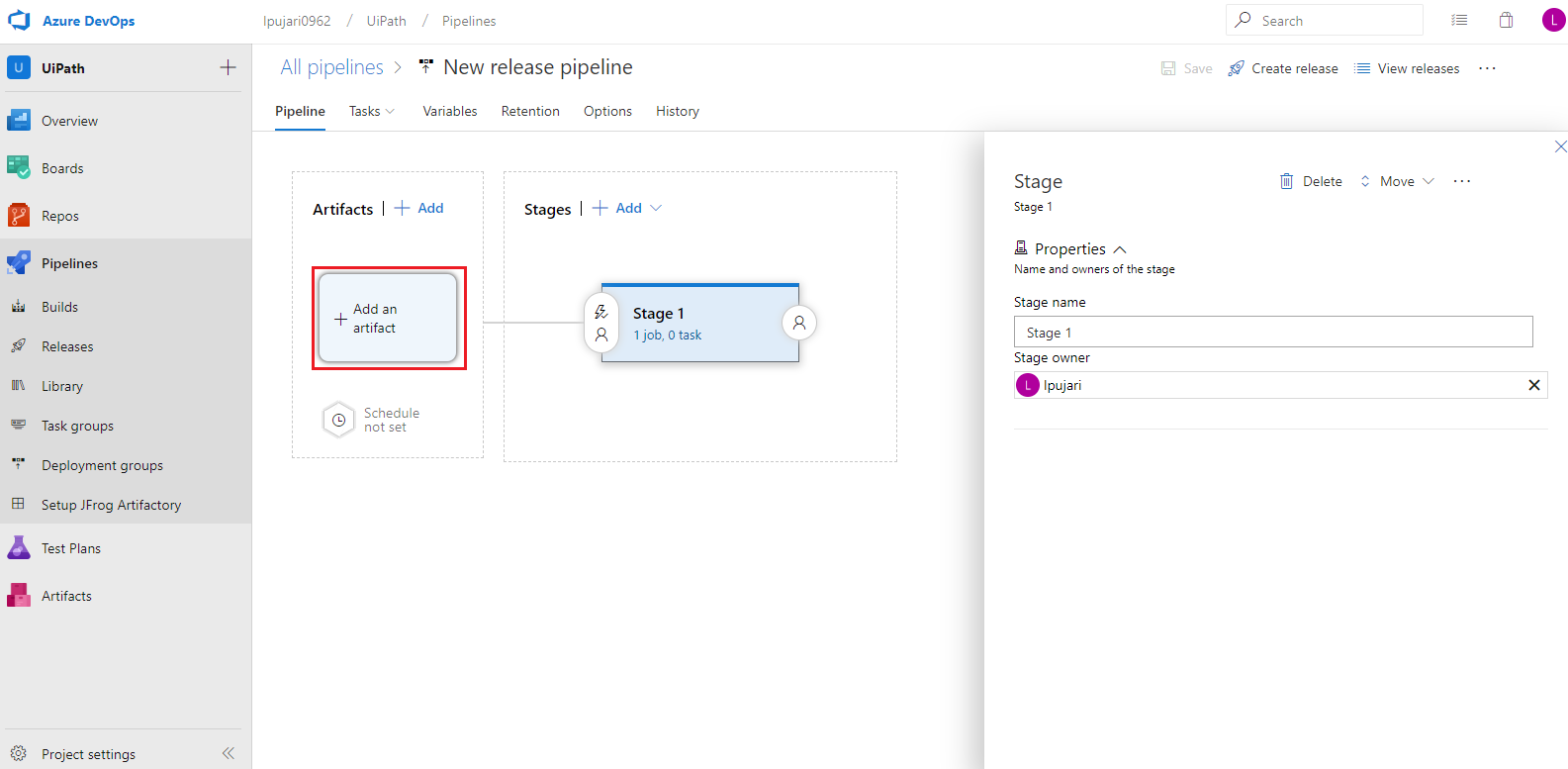
After selecting empty job, we are giving the stage name and pipeline name as default.



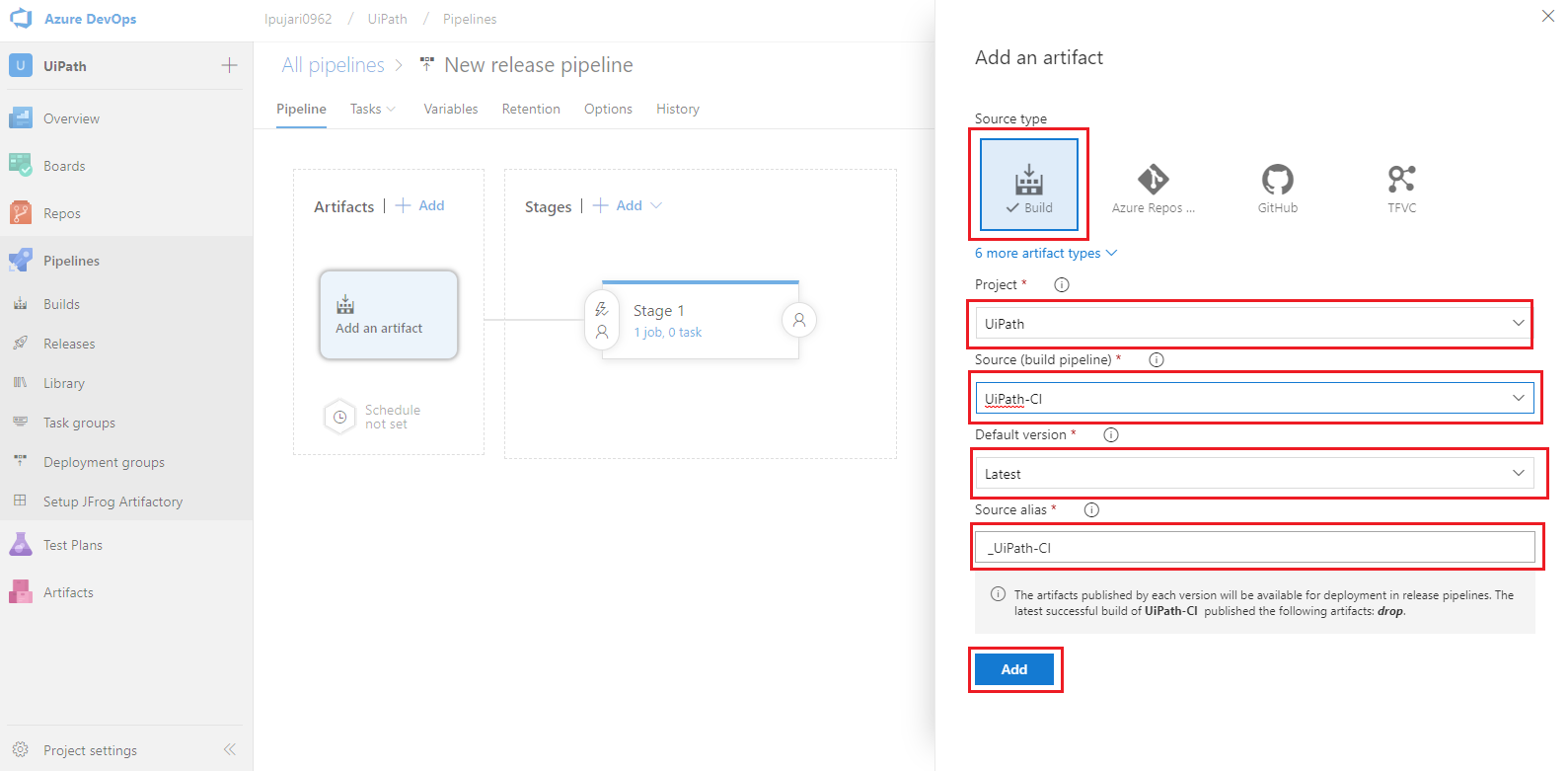
Now click on save to save the changes.



Now click on +Add an artifact to add the artifact to deploy.



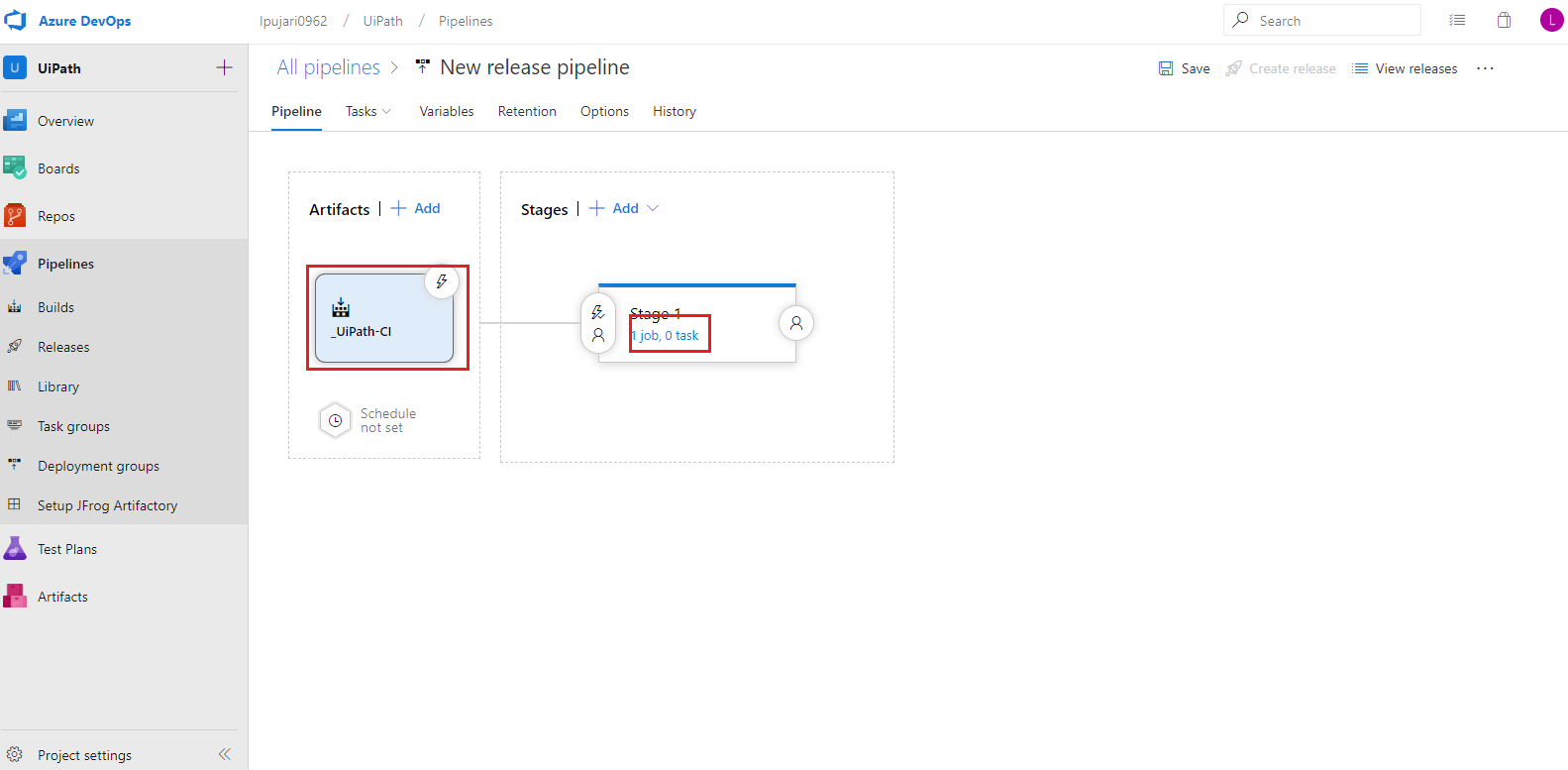
Now to add artifact configure it as follows and click on add.



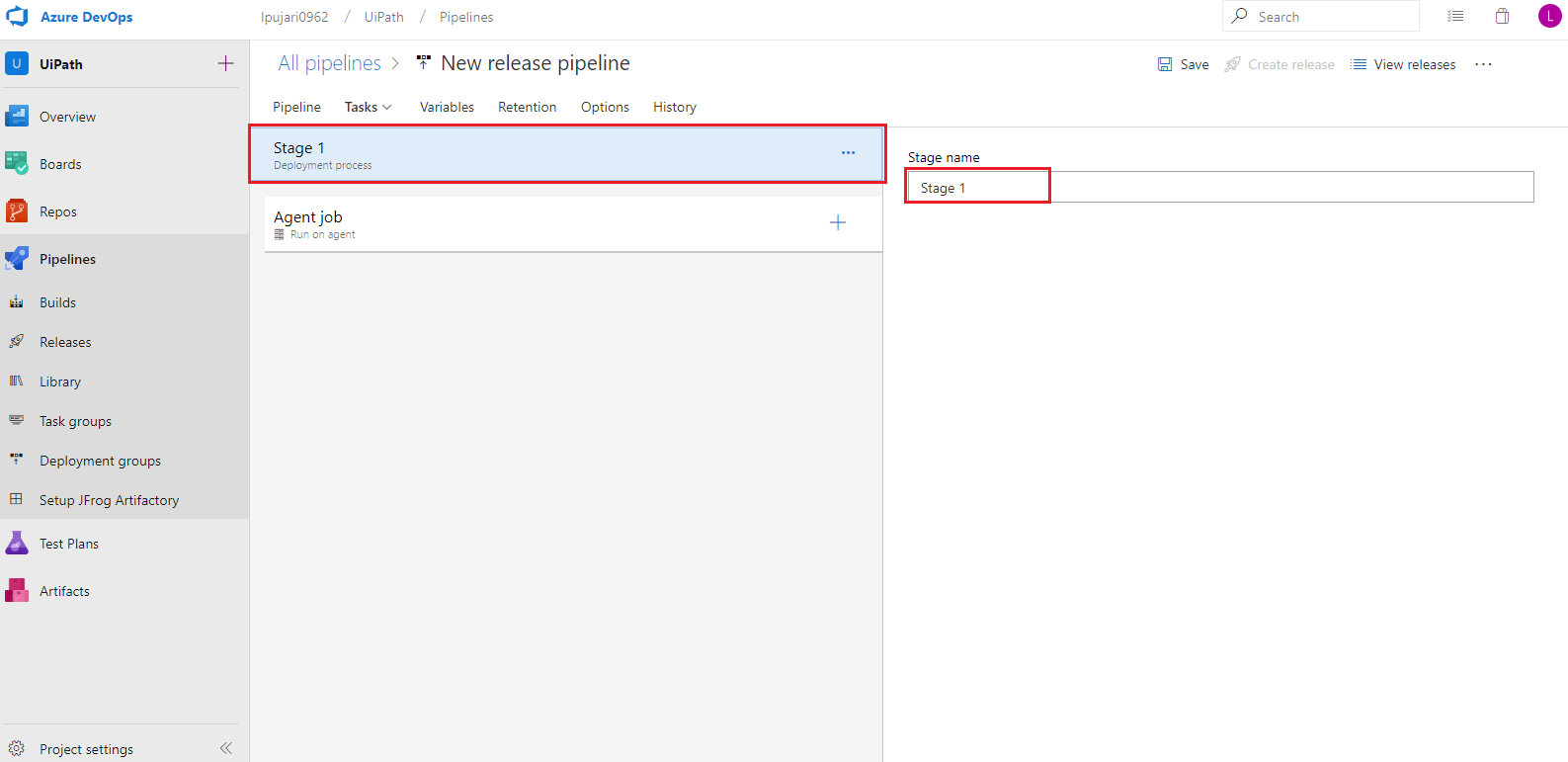
Arguments to add the artifact:

* **Source Type:** Here it will take the source from VSTS Build
* **Project:** Name of the project
* **Source Build Pipelines Name:** Name of the Integration pipeline
* **Default Version:** It will take the source from latest successful build
* **Source alias:** it will generate a folder with alias name to store the result

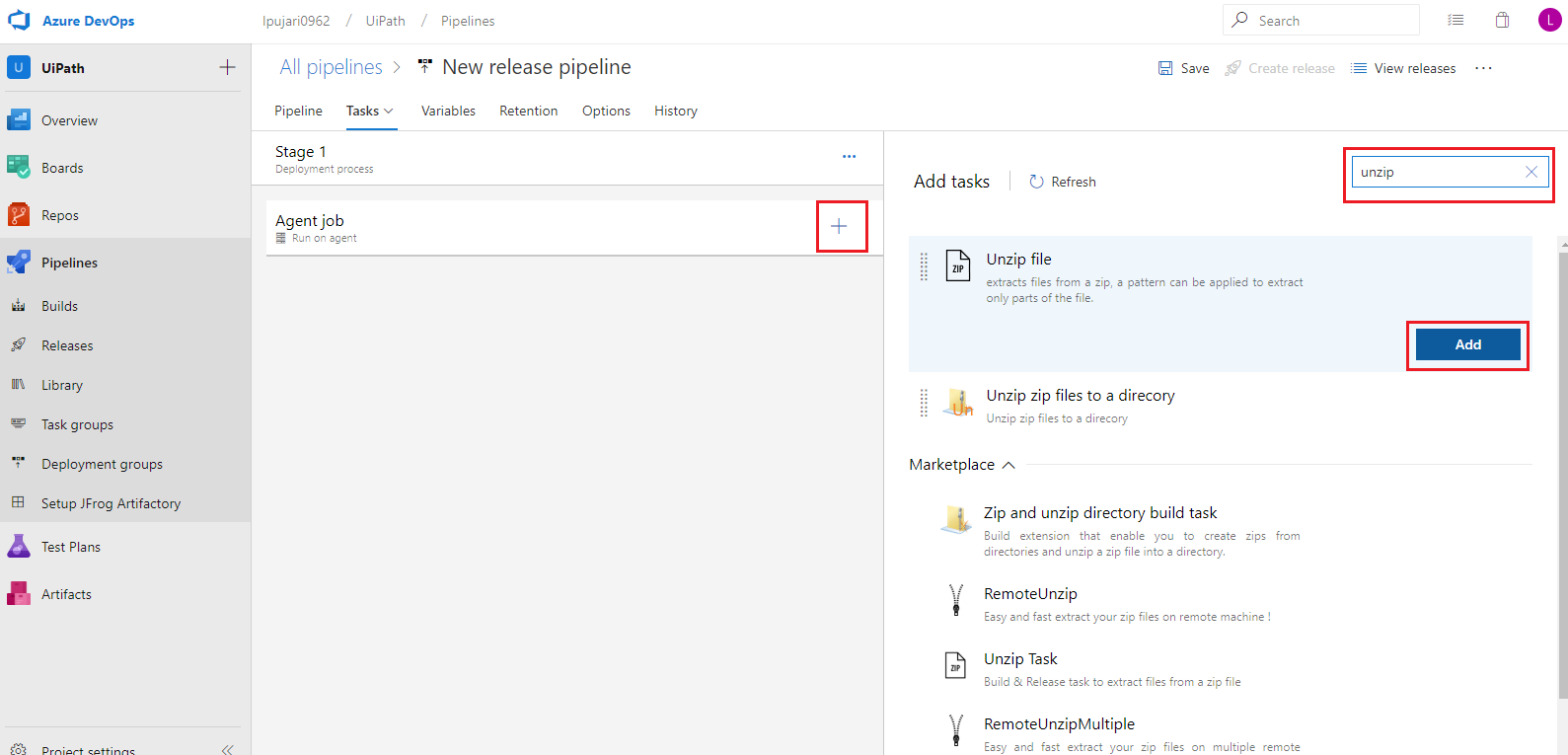
Now click on tasks under the stage to add tasks to the pipeline



Here we have given the stage name as default.

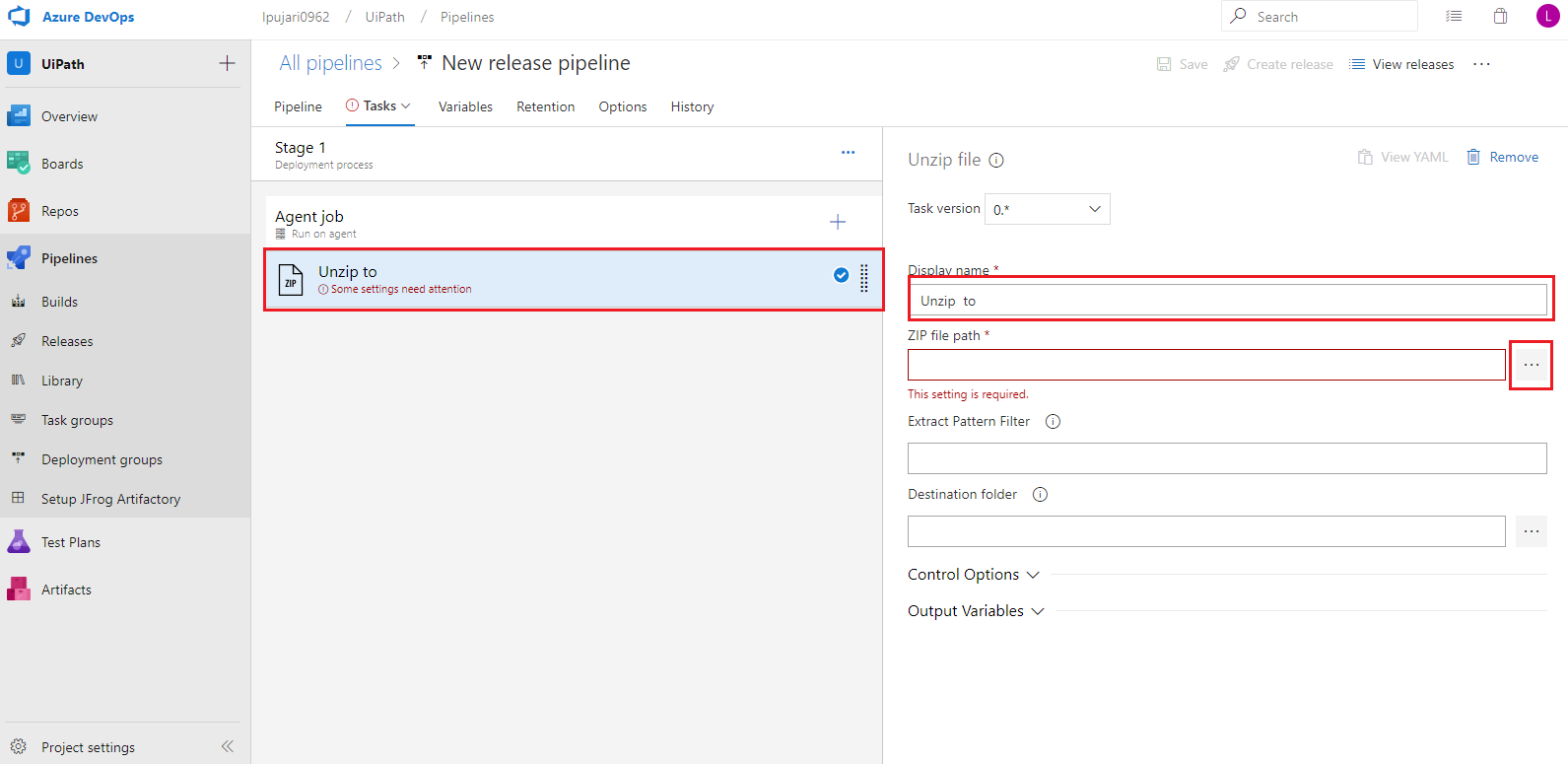


Now we are going to add our task for release stage as shown below. We were able to add task by clicking on + symbol next to agent phase as shown below. Now search for Unzip task to unzip the archived file in integration build pipeline.



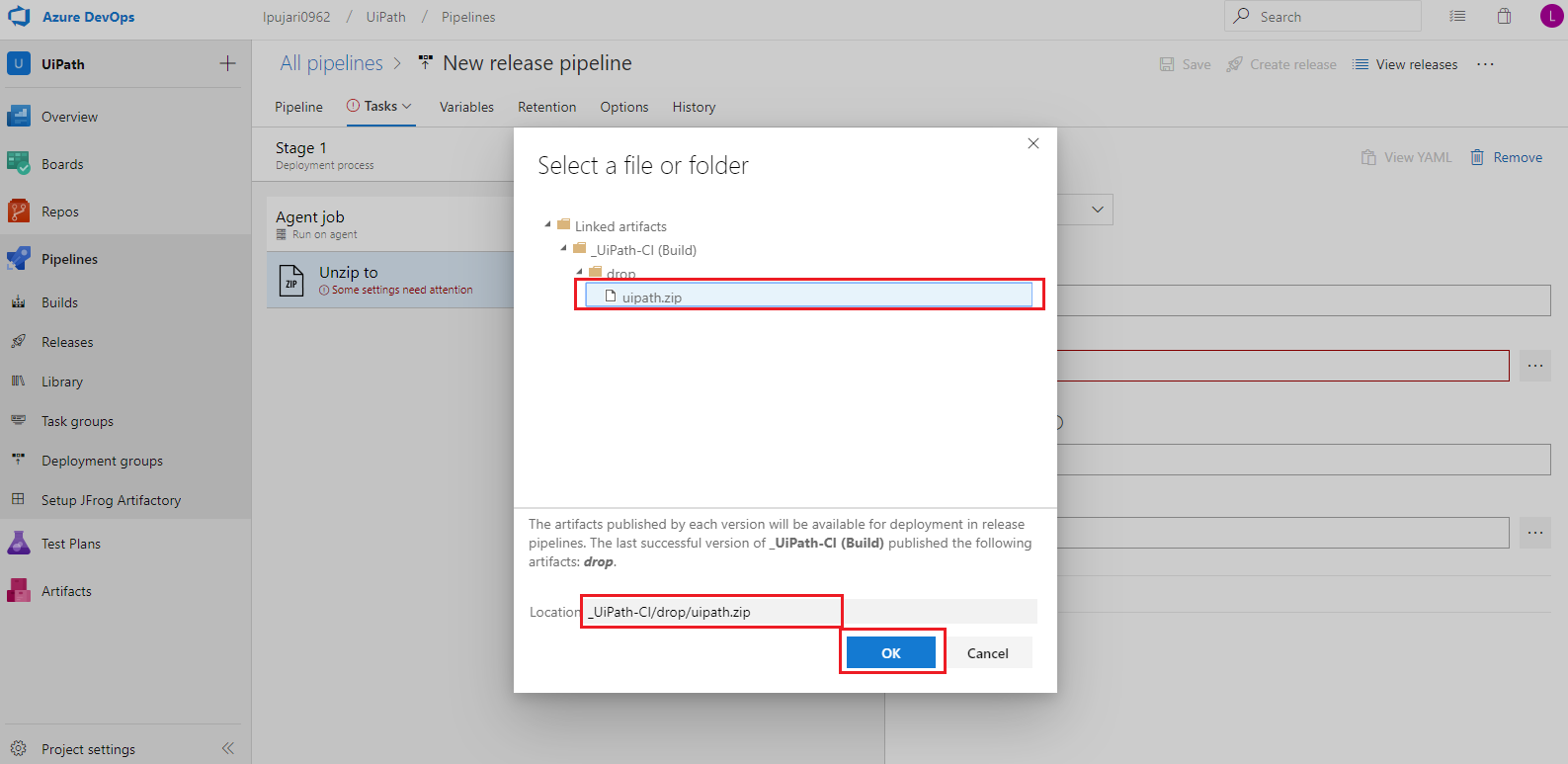
### Tasks 1: Unzip

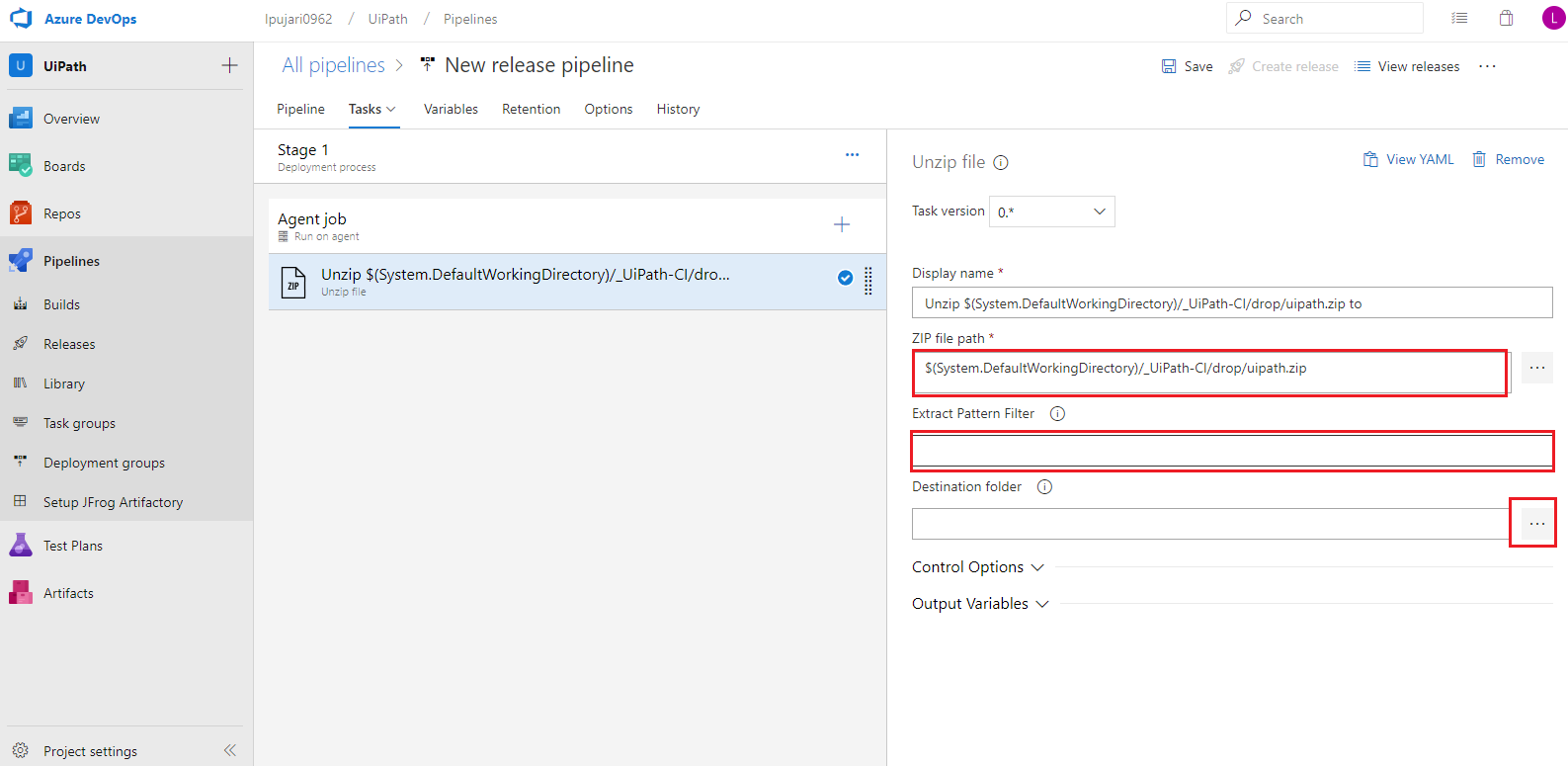
Now give a name to the task and add the Zip file path as follows.

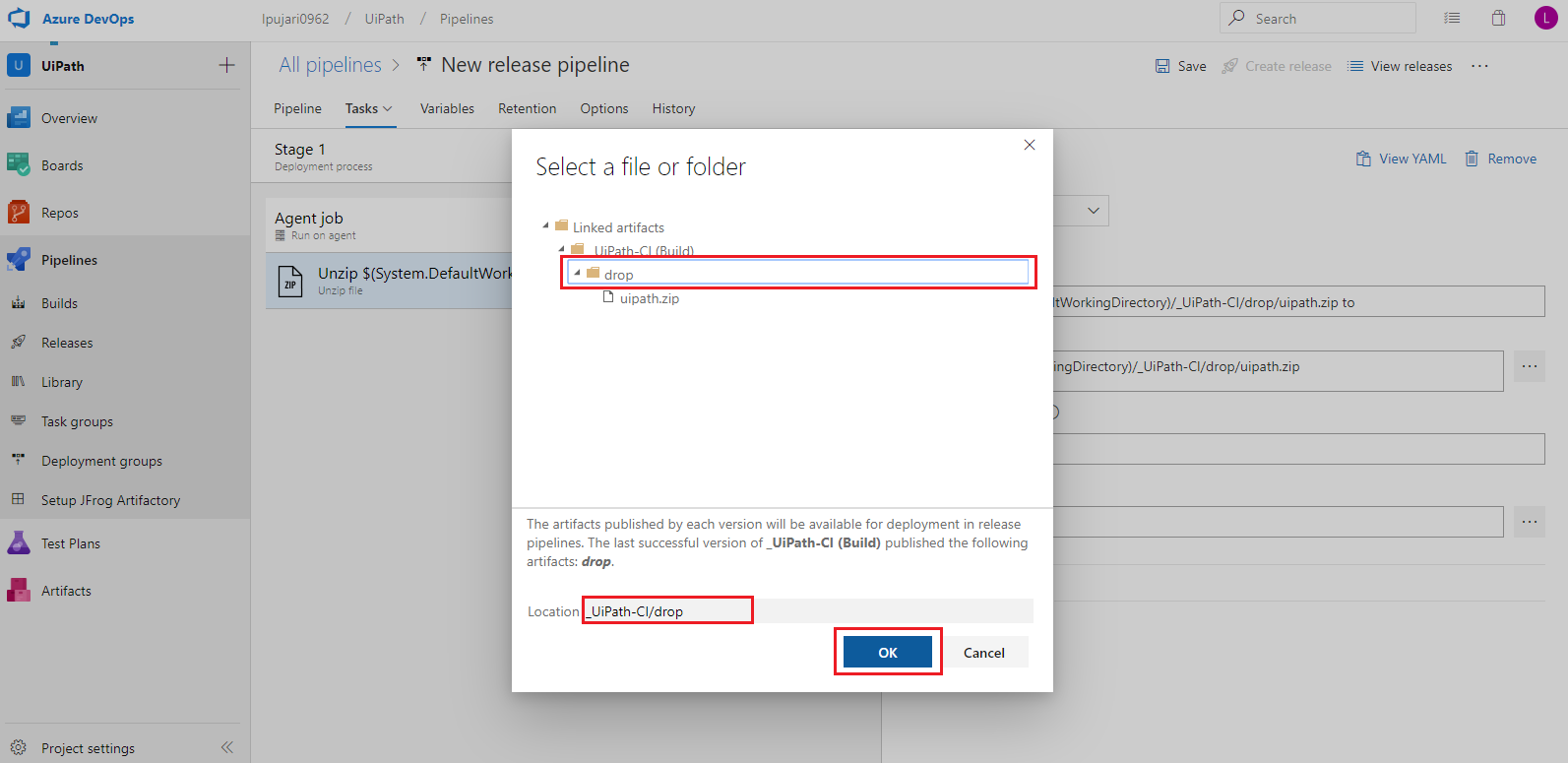


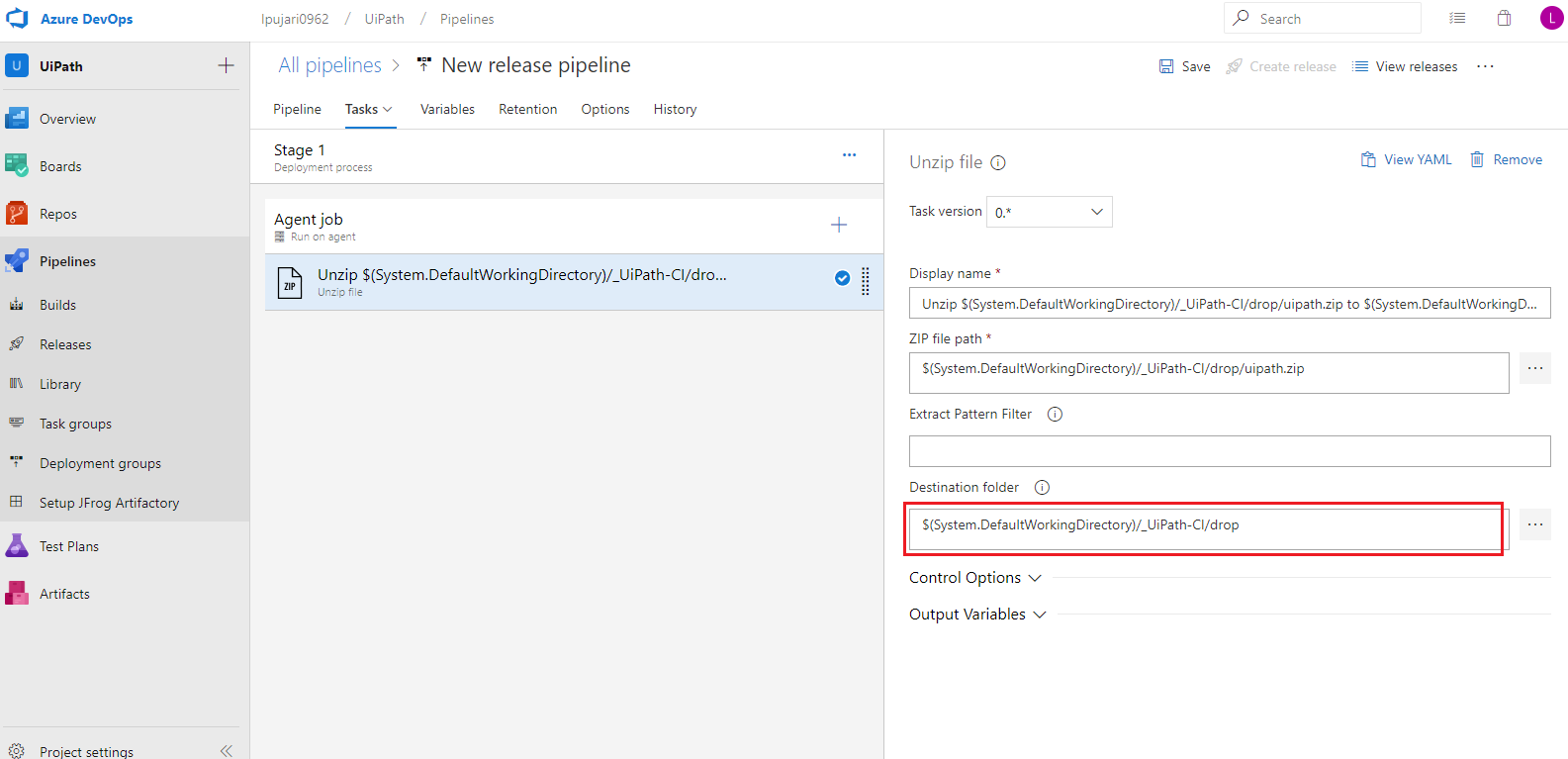
Arguments for Unzip file task:

* **Display name:** name for the task
* **Zip file path:** Path of the Zip file where it was stored
* **Extract pattern filter:** It will extract the files depends on the pattern.
* **Destination Folder:** Where to store the unzipped file.

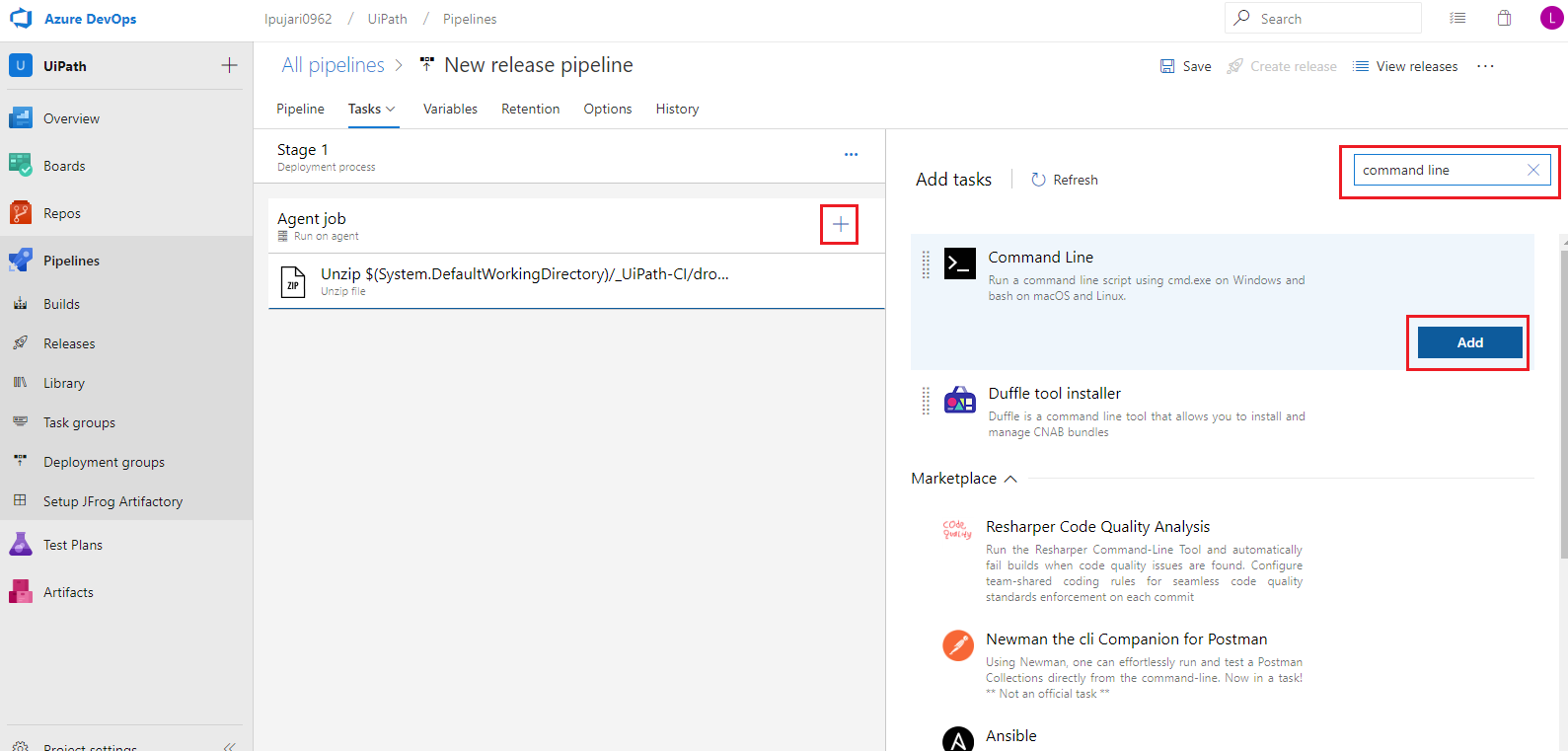






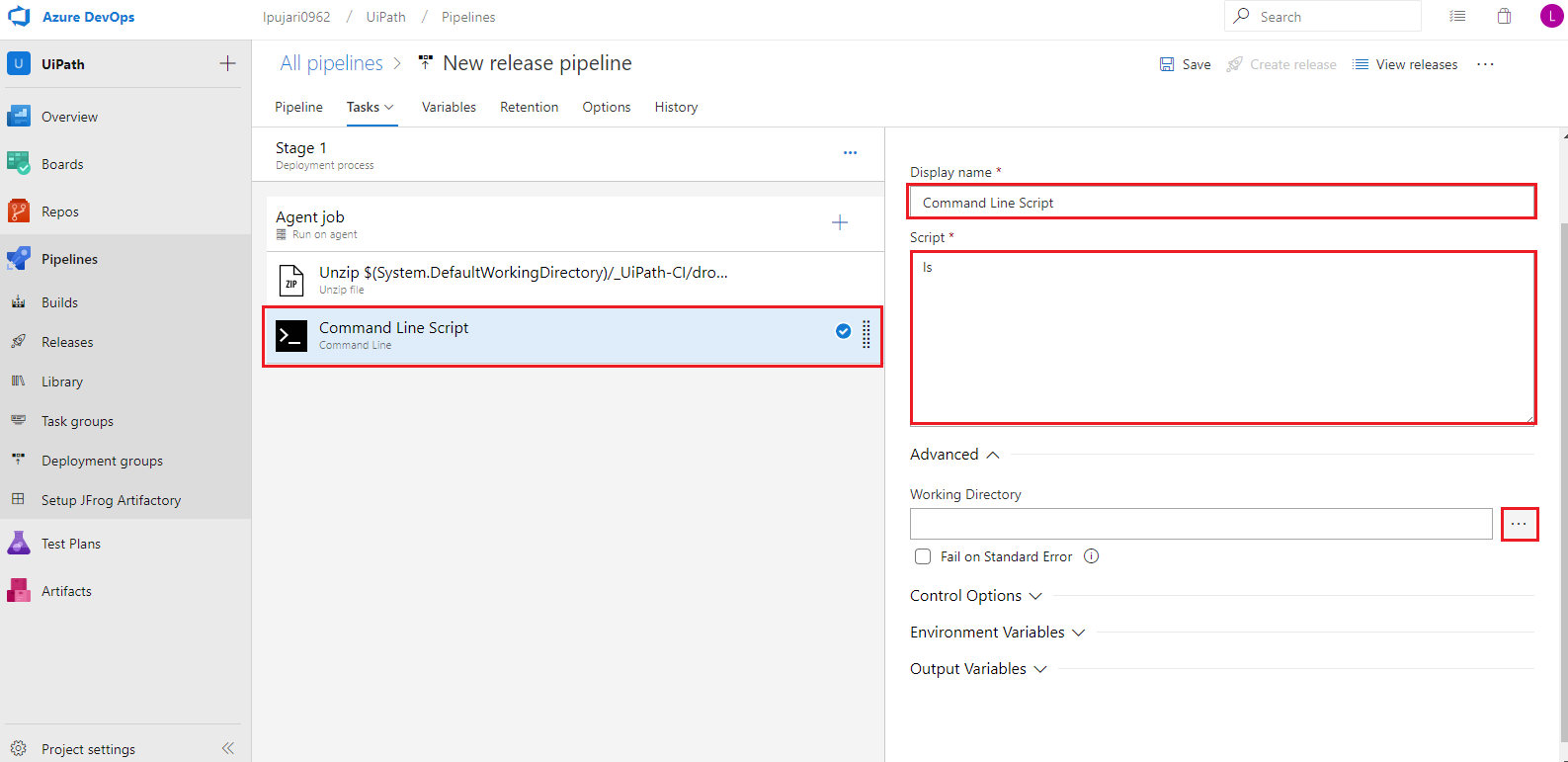


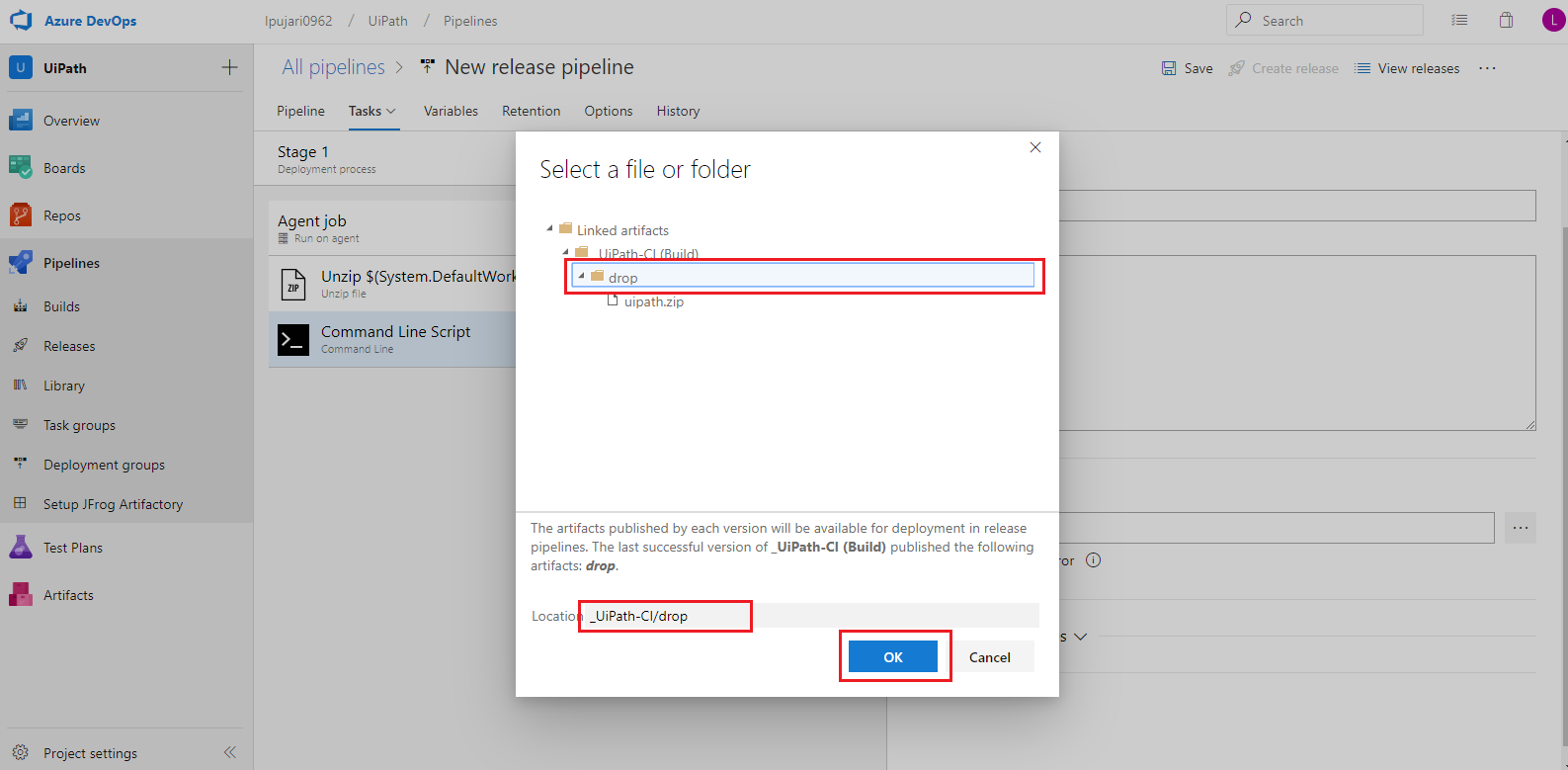
Now search for the next task command line and add it to the tasks list.



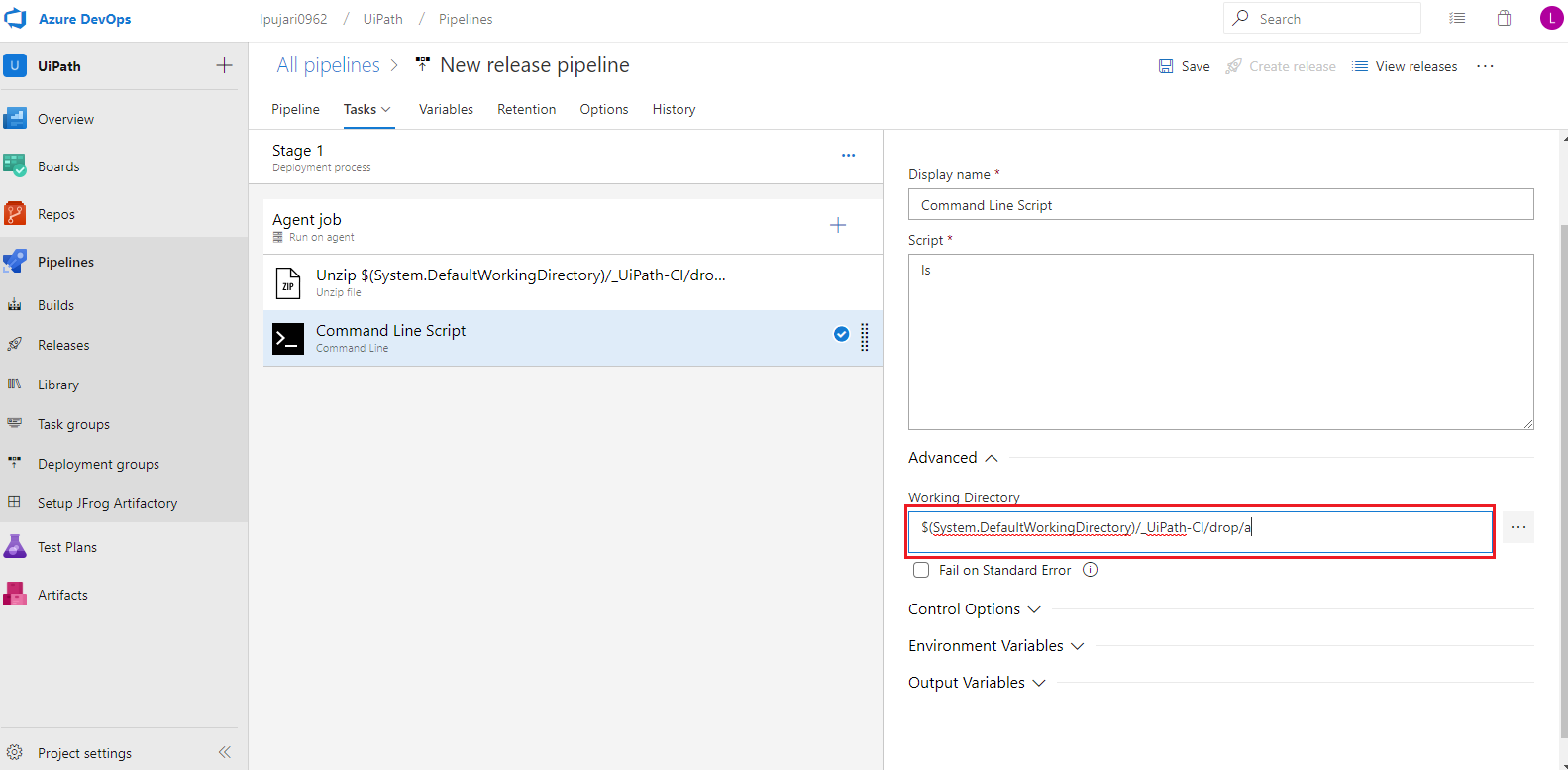
### Tasks 2: Command Line

In Script add a command to run the command in the agent. Here we configured it with ls command to check the files got updated in the current working directory or not.

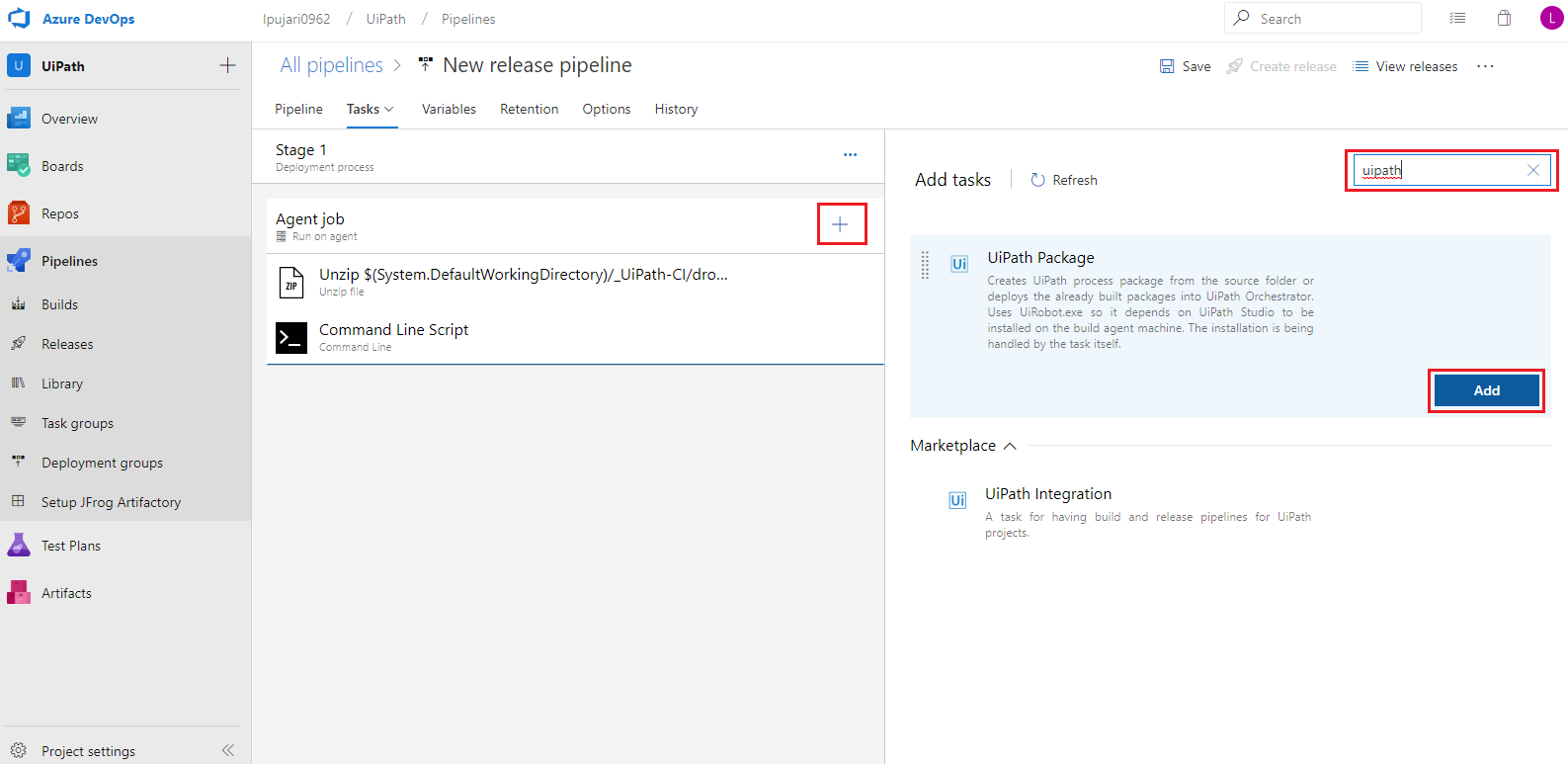




In advanced section we need to give the present working directory. Here we are adding /a/ path because after extracting file it was generating a folder called **a**

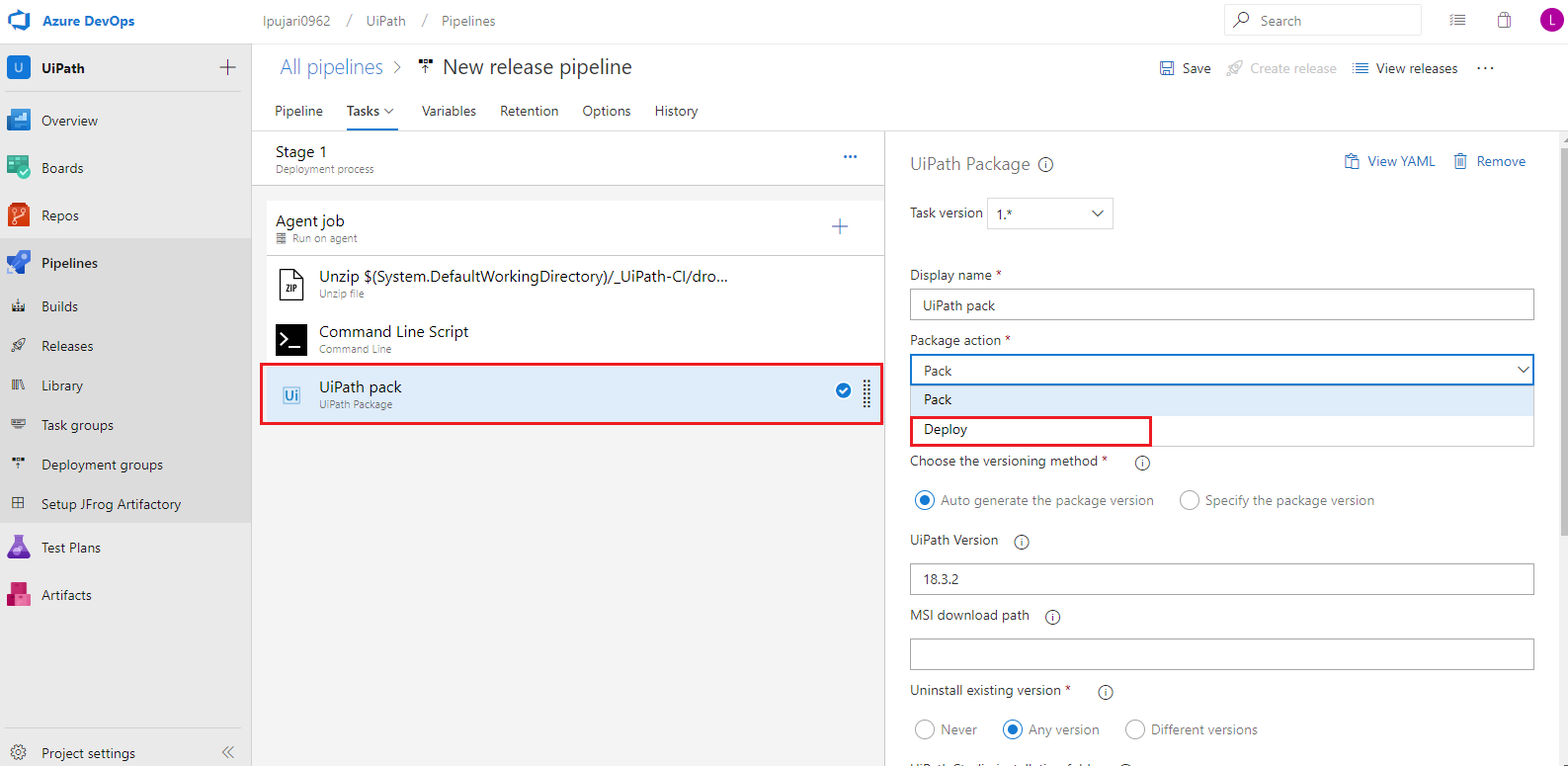


Now search for the next task uipath and add it to the task list.



### Tasks 3: UiPath Deploy

This task will deploy the package to the orchestrator.



Arguments for UiPath task:

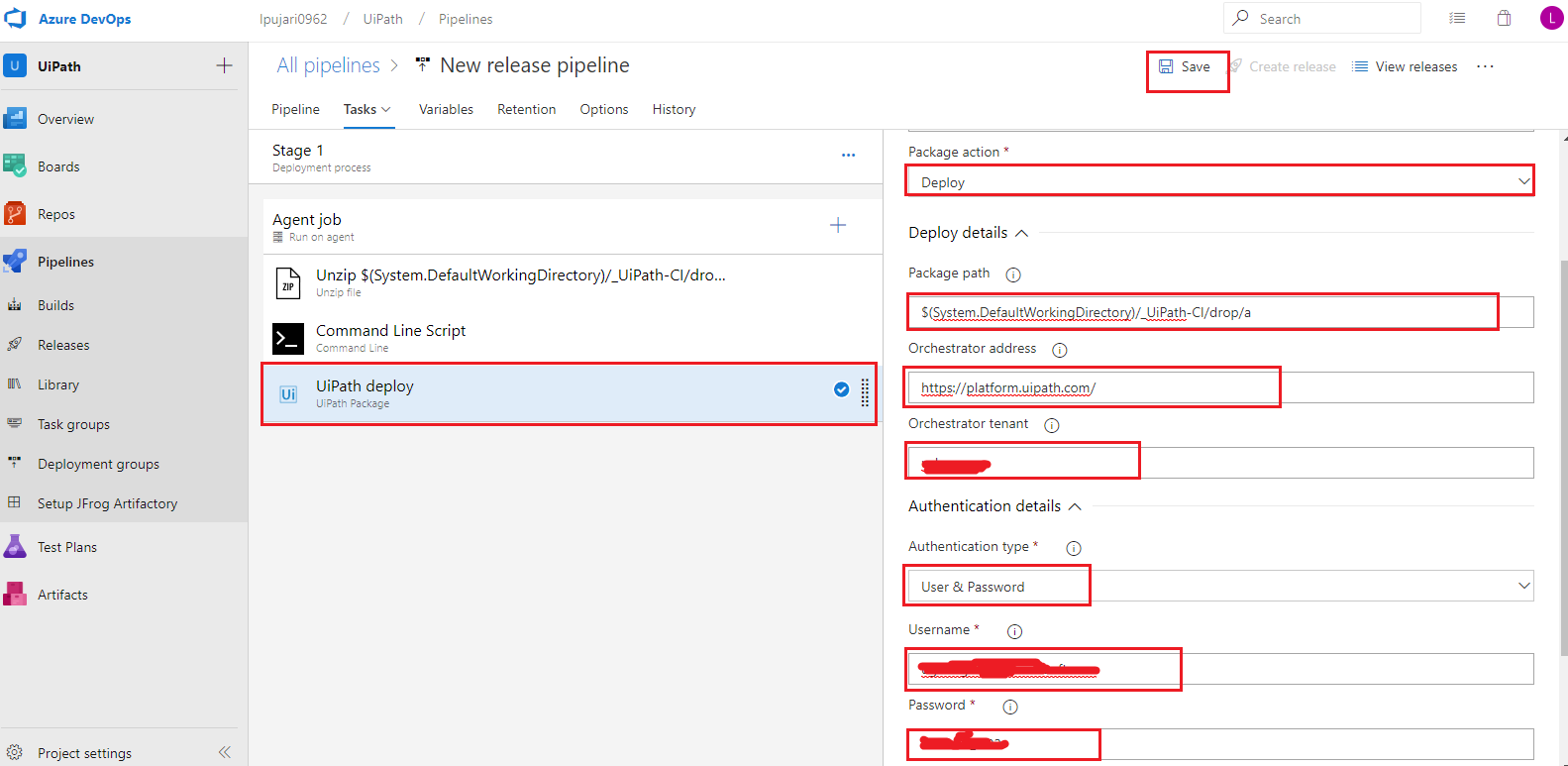
* **Display Name:** Name for the task
* **Package action:** We need to click on the dropdown and select deploy action to deploy the nupkg file

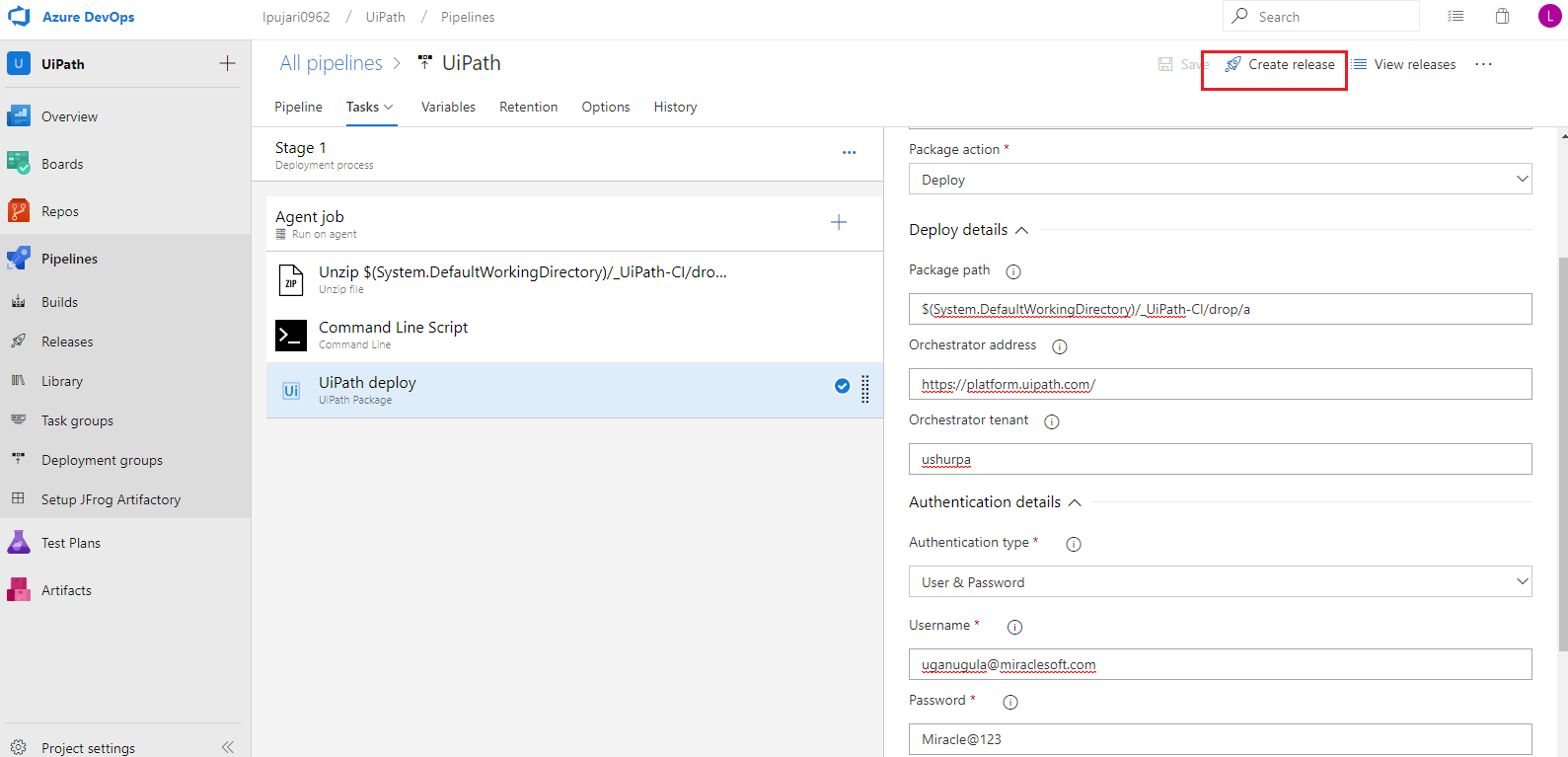
In Deploy path section

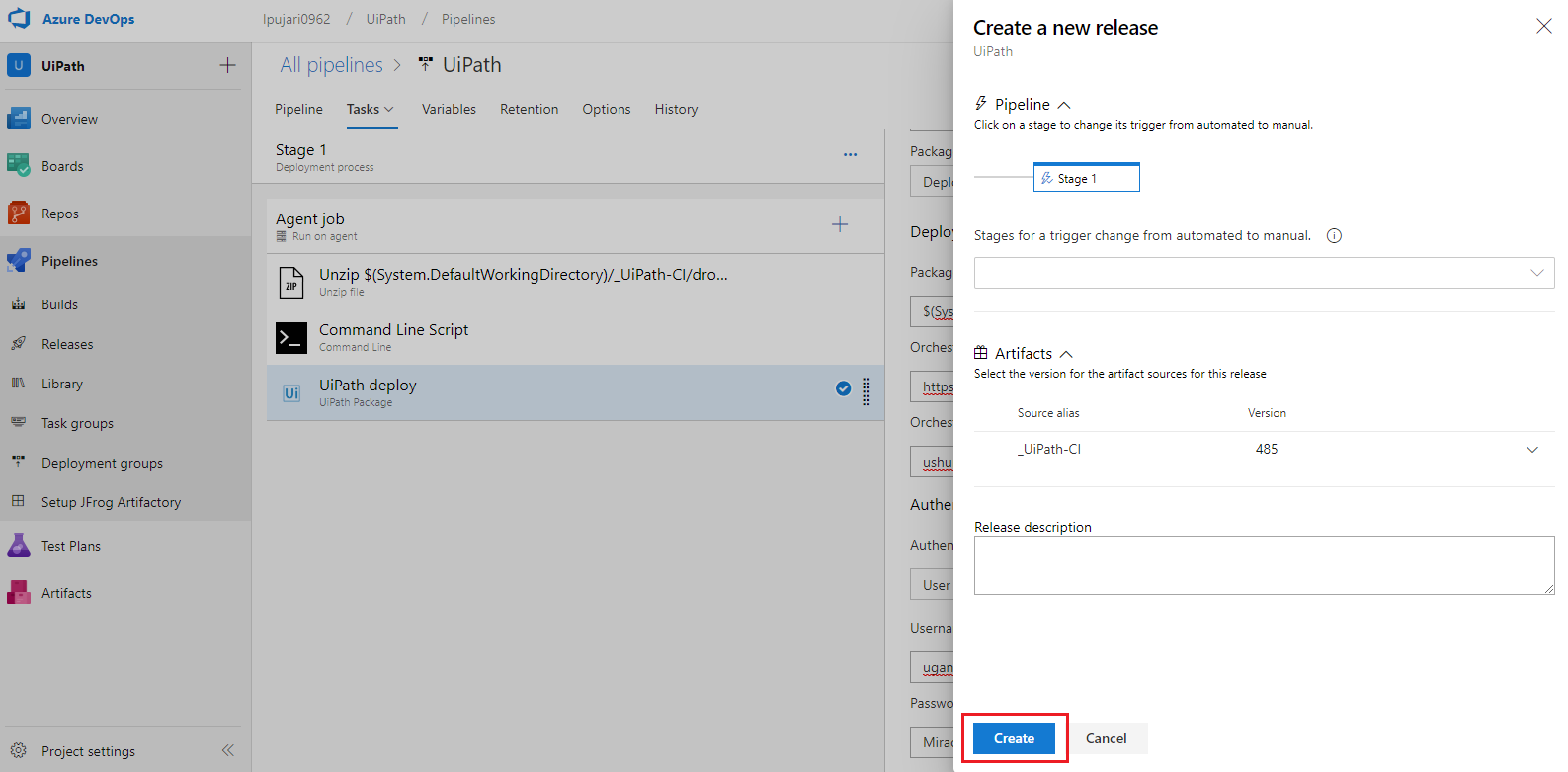
* **Package path:** path of the file to deploy
* **Orchestrator address:** Address of the orchestrator to deploy. <https://platform.uipath.com>
* **Orchestrator tenant:** Tenant name of the orchestrator

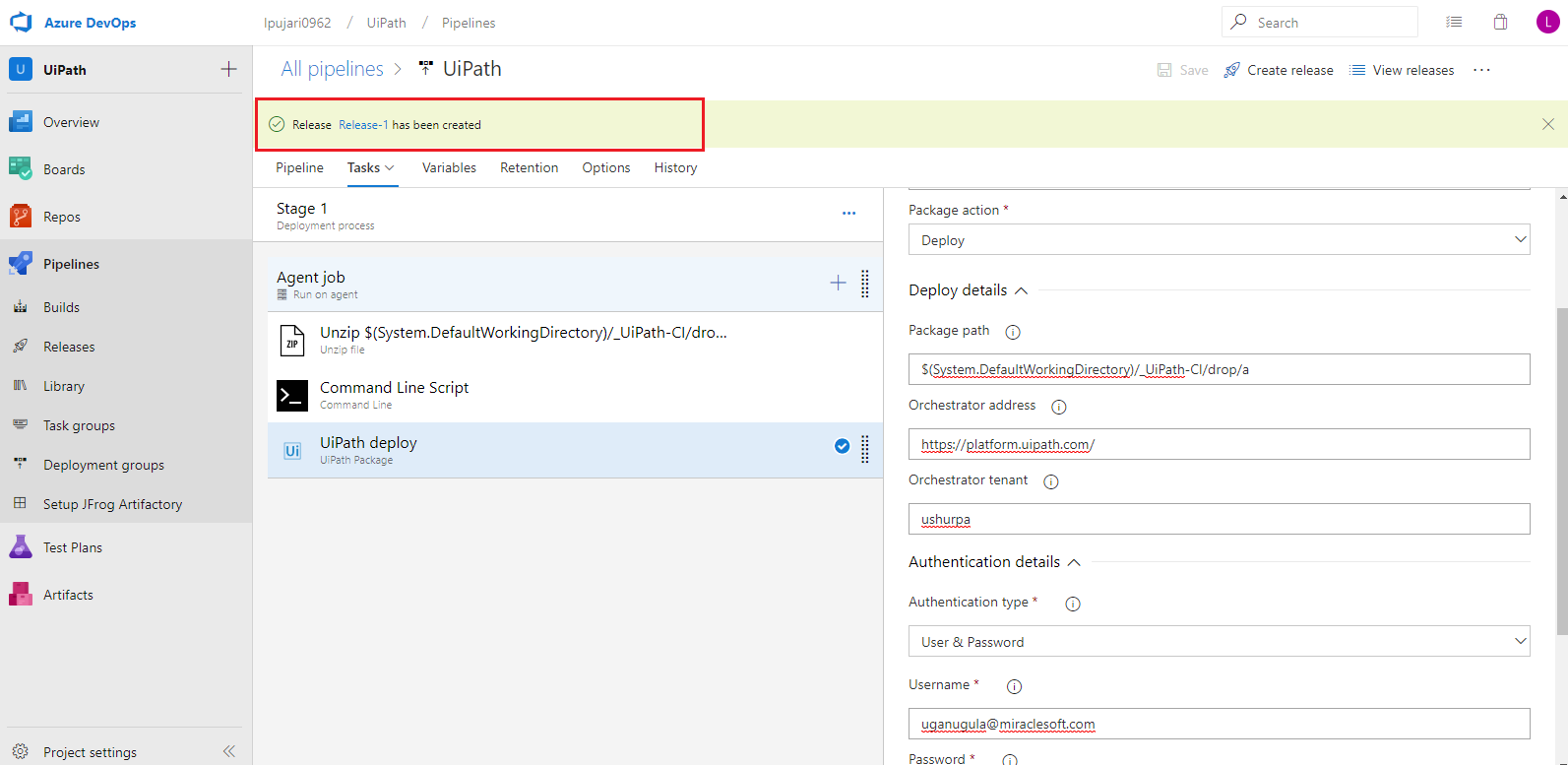
In Authentication Details section

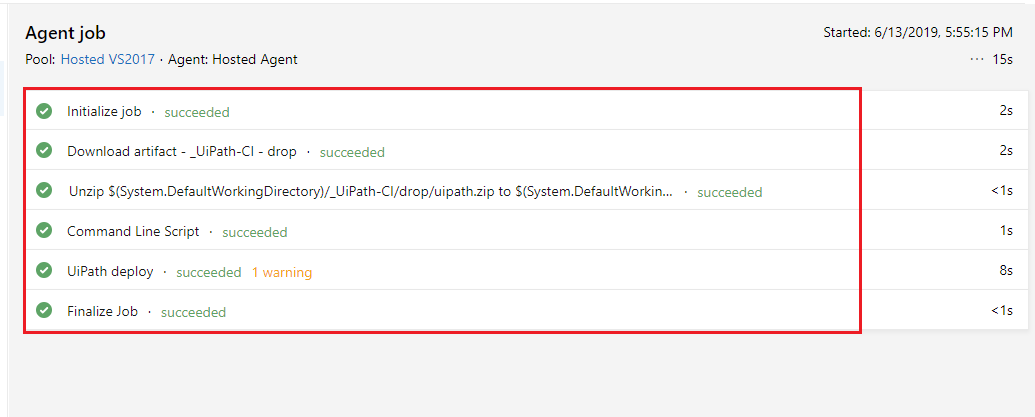
* **Authentication Type:** Select the authentication type as Username & Password
* **Username:** Give the username of the UiPath Orchestrator
* **Password:** Provide the Password for the authentication

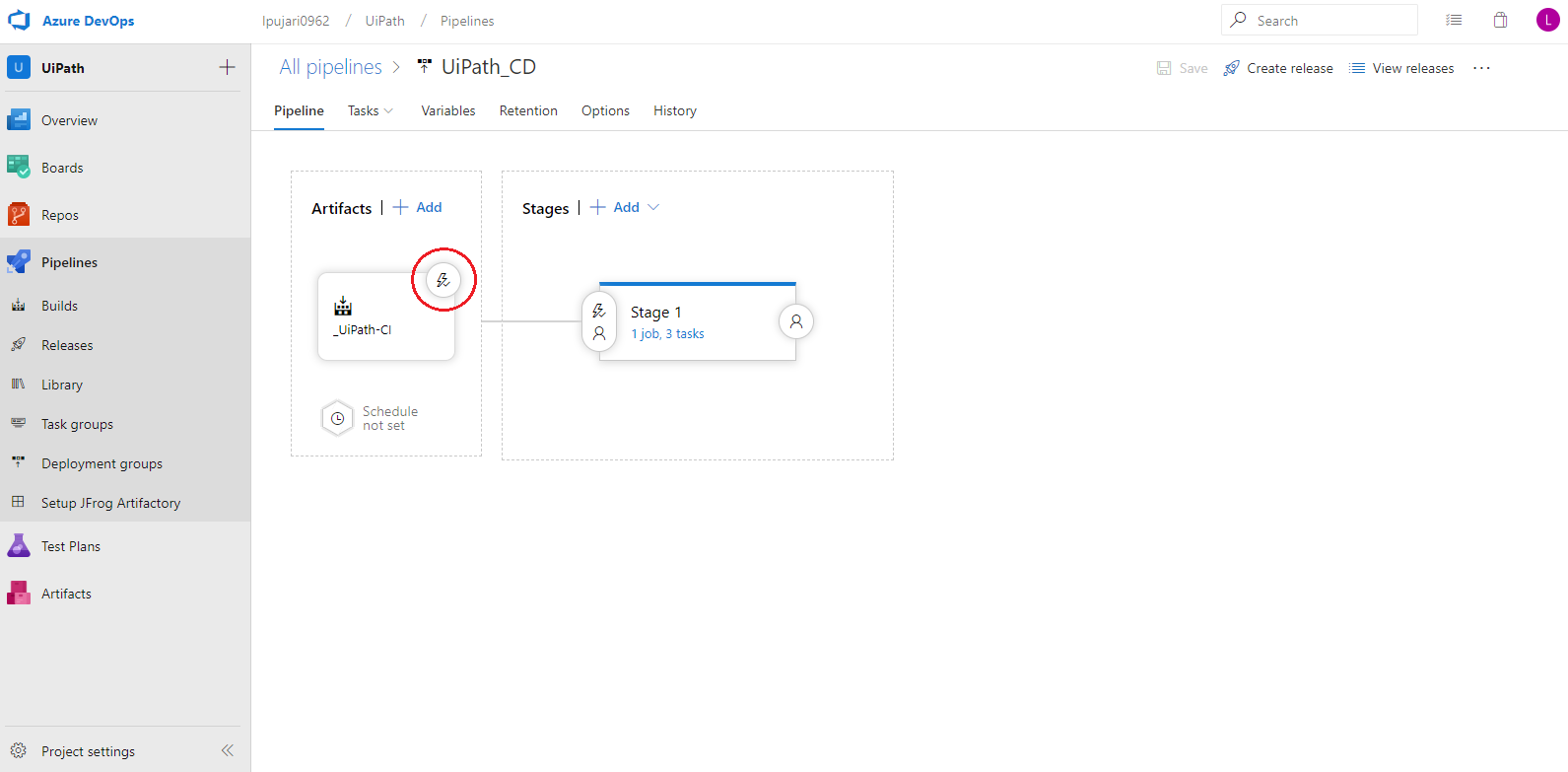
After configured the authentication details click on save to save the changes and it will enable to create release.

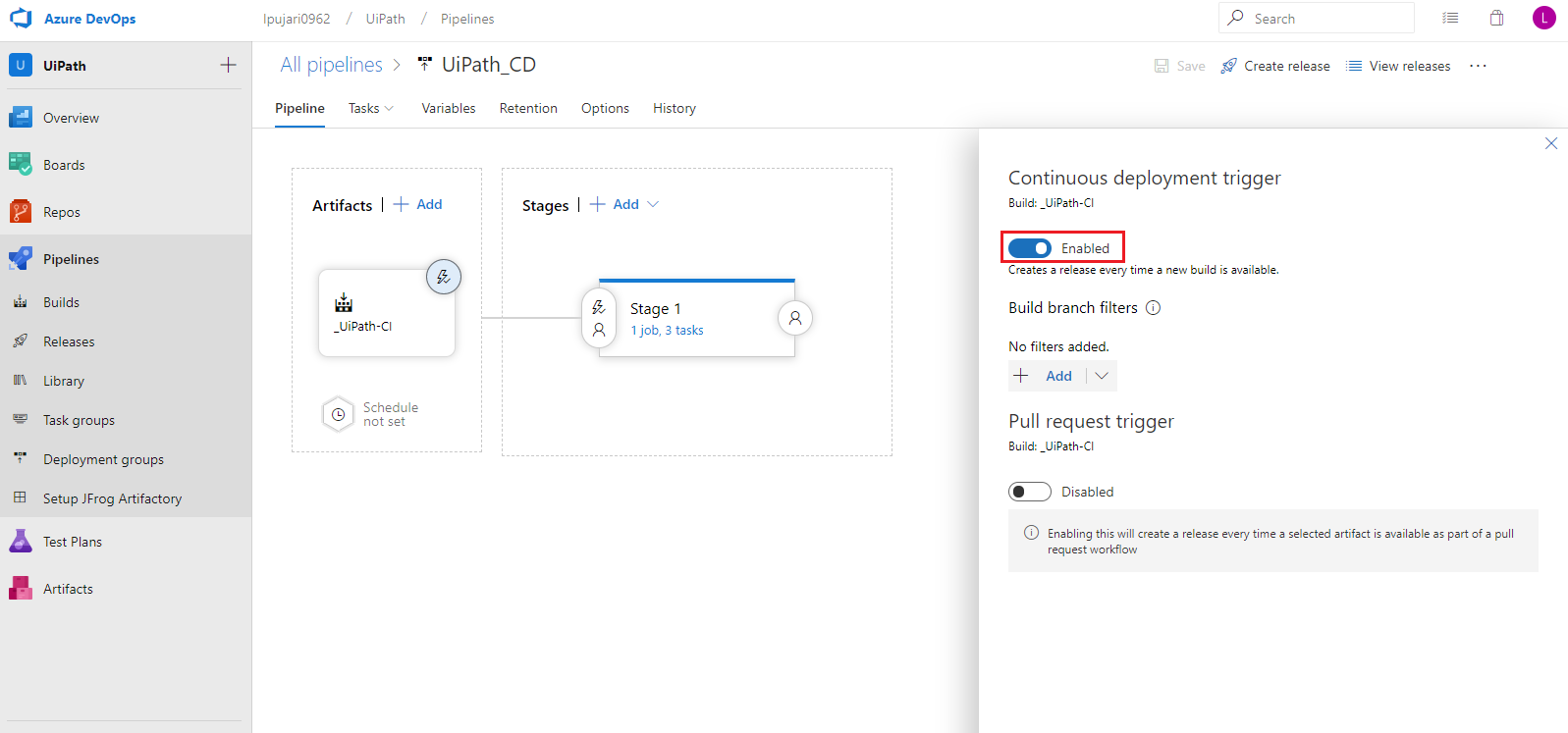
Now Click on create release to create a new release

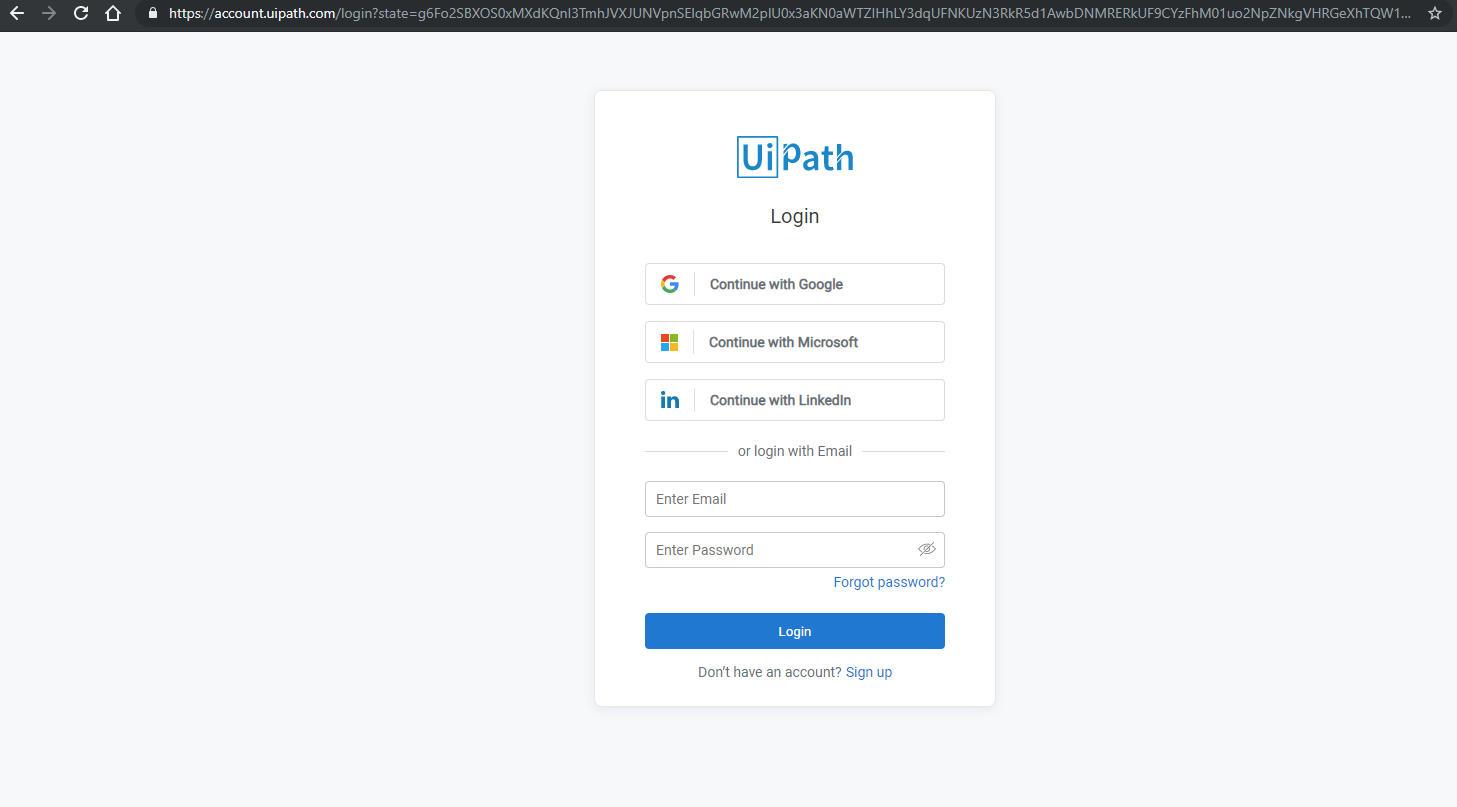
Now click on Create to start the release process

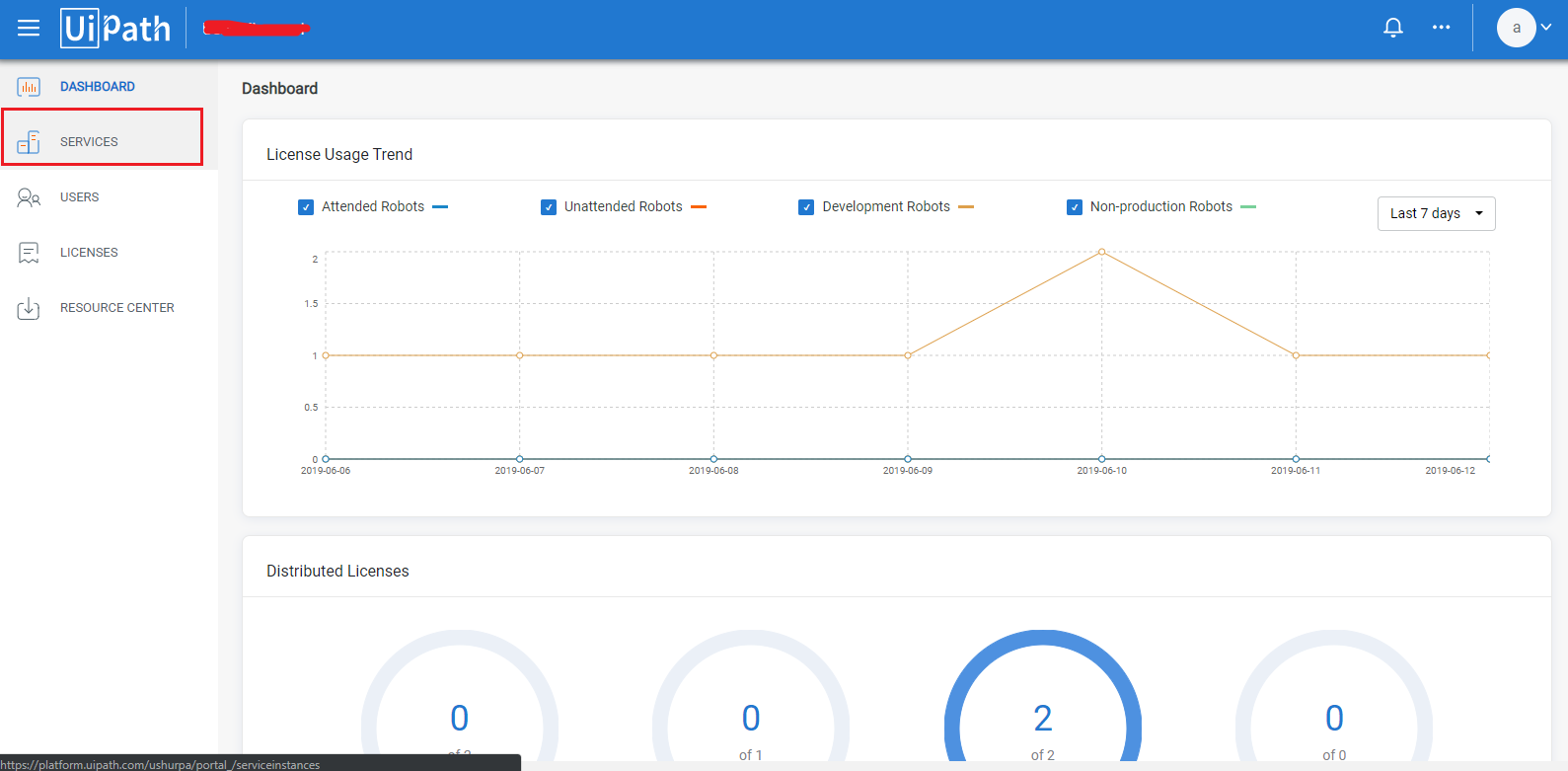
Now click on the release number to check the logs process

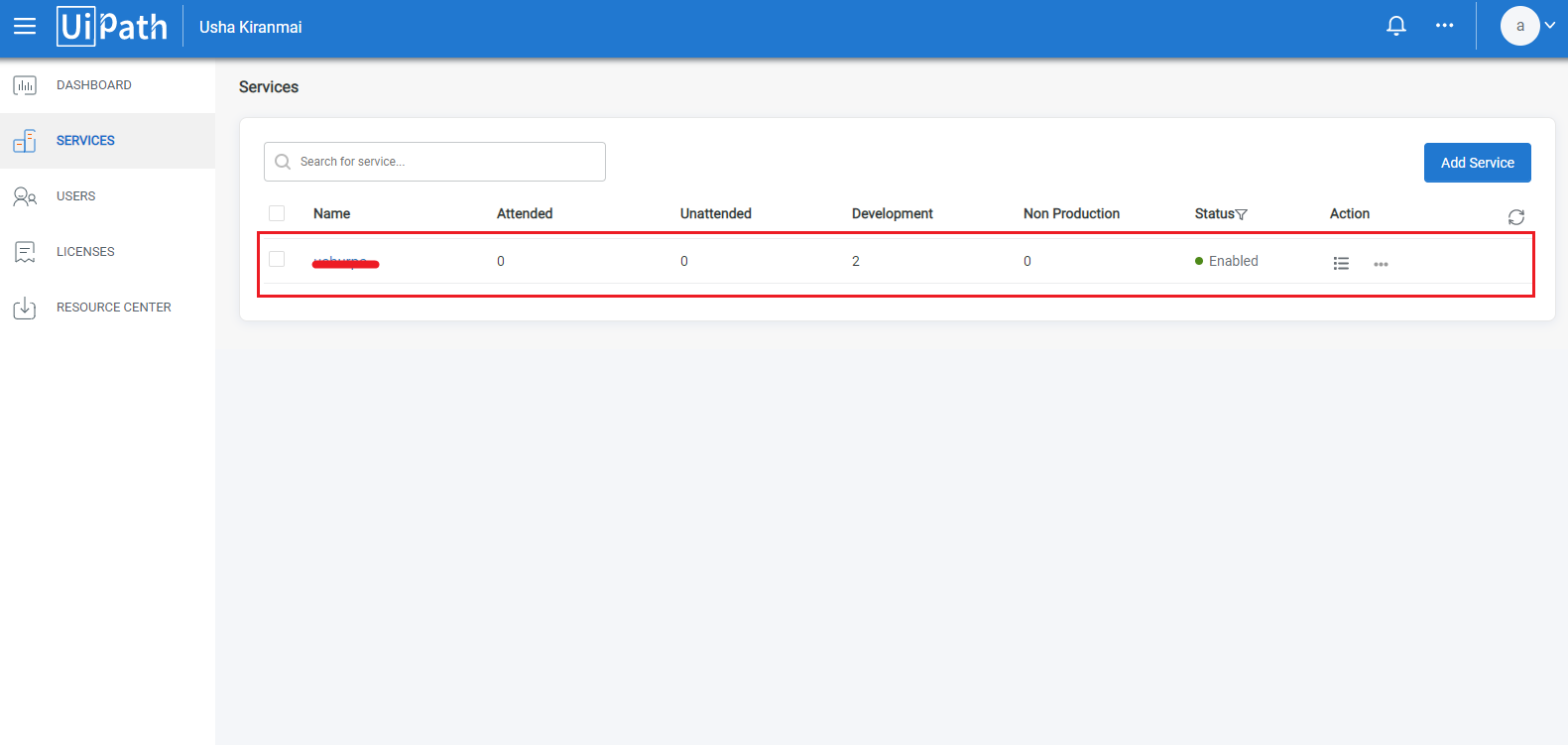
After the release got success all the tasks will be shown as succeeded

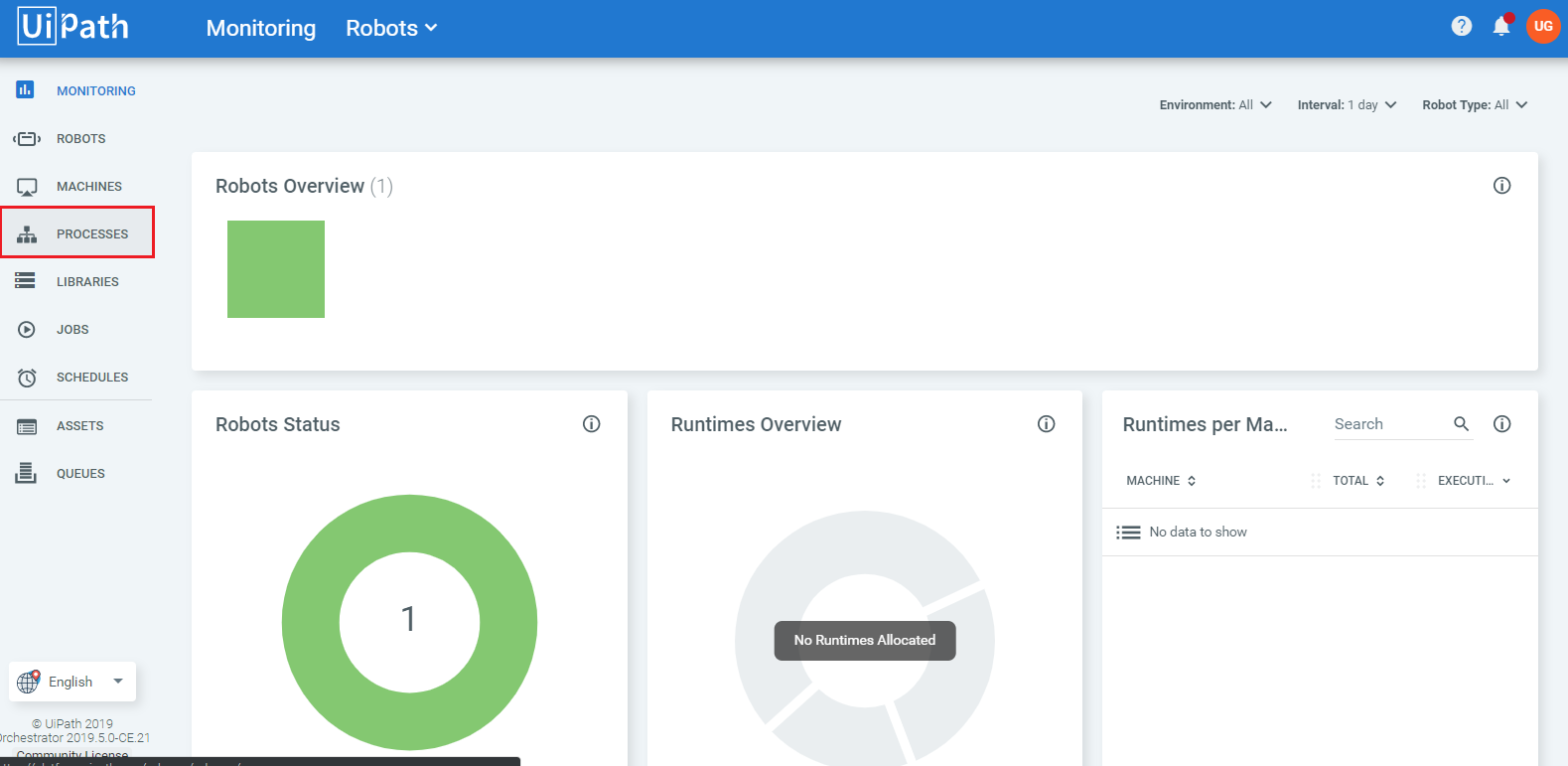
Now click on the bubble and enable the continuous deployment trigger to automate the deployment whenever a new build succeeded.



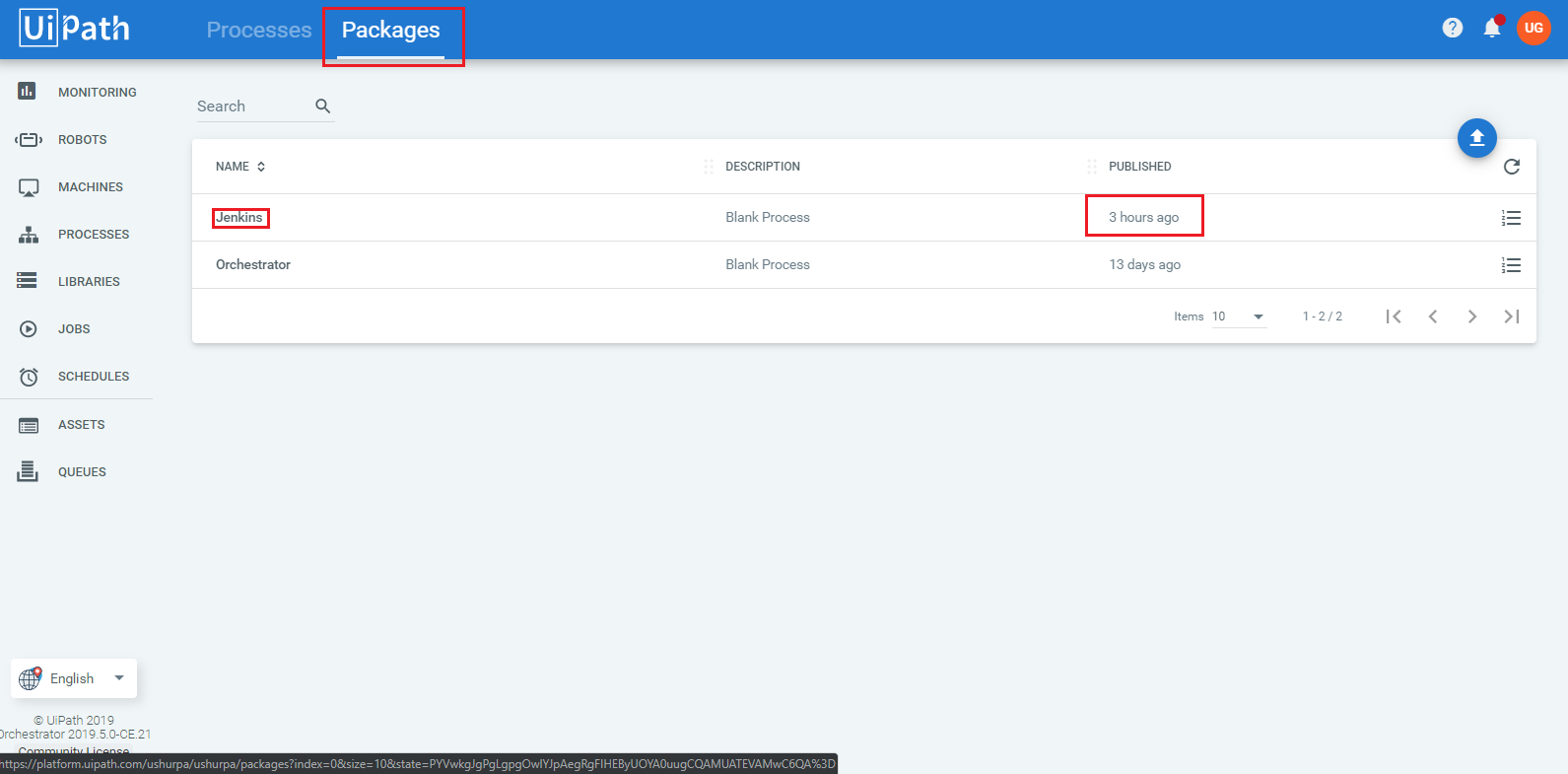
Now go to [https://platform.uipath.com](https://platform.uipath.com/) and log in into our orchestrator to check whether the deployment was done or not. 







Here we got the newly deployed package.



End of the Document.