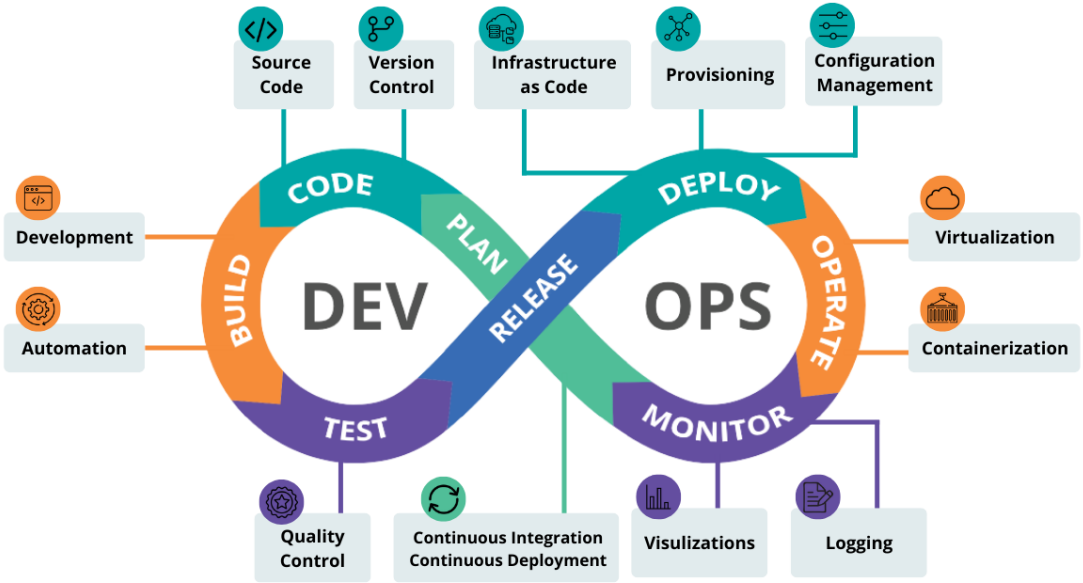
Introduction to Azure DevOps

What is DevOps?

DevOps is the process of integrating Developer and Operation teams in order to improve collaborations and productivity



Why Microsoft Azure DevOps?

- **Reliability** – As a SaaS offering, Azure DevOps is reliable, scalable and globally available. It is also backed by an SLA of 99.9% uptime and by 24×7 support.
- **Easy collaboration among multiple teams** – GIT integrations makes the large / distributed teams life easy.
- **Maintenance cost** – Low maintenance cost since its SAAS. We no longer need to worry about updates and patches.
- **Up-to-date features** – Azure DevOps users get access to new features every 3 weeks.
- **Built in wikis** for sharing information

Starting with Azure DevOps

azure.microsoft.com/en-us/services/devops/

Azure Explore Products Solutions Pricing Partners Resources Search Docs Support Contact Sales Sign in

Home / Services / Azure DevOps

Azure DevOps

Plan smarter, collaborate better, and ship faster with a set of modern dev services.

Start free | Start free with GitHub

Already have an account?

Sign in to Azure DevOps >

Components/Services of Azure DevOps

Azure Boards

Allows Work item tracking, Agile planning, Power BI visualization, and similar other reporting tools.

Azure Repos

Provides full-support for cloud-hosted private repositories.

Azure Pipelines

Defines CI/CD- Continuous Integration and Continuous deployment process with support for containers and Kubernetes.

Azure Test Plans

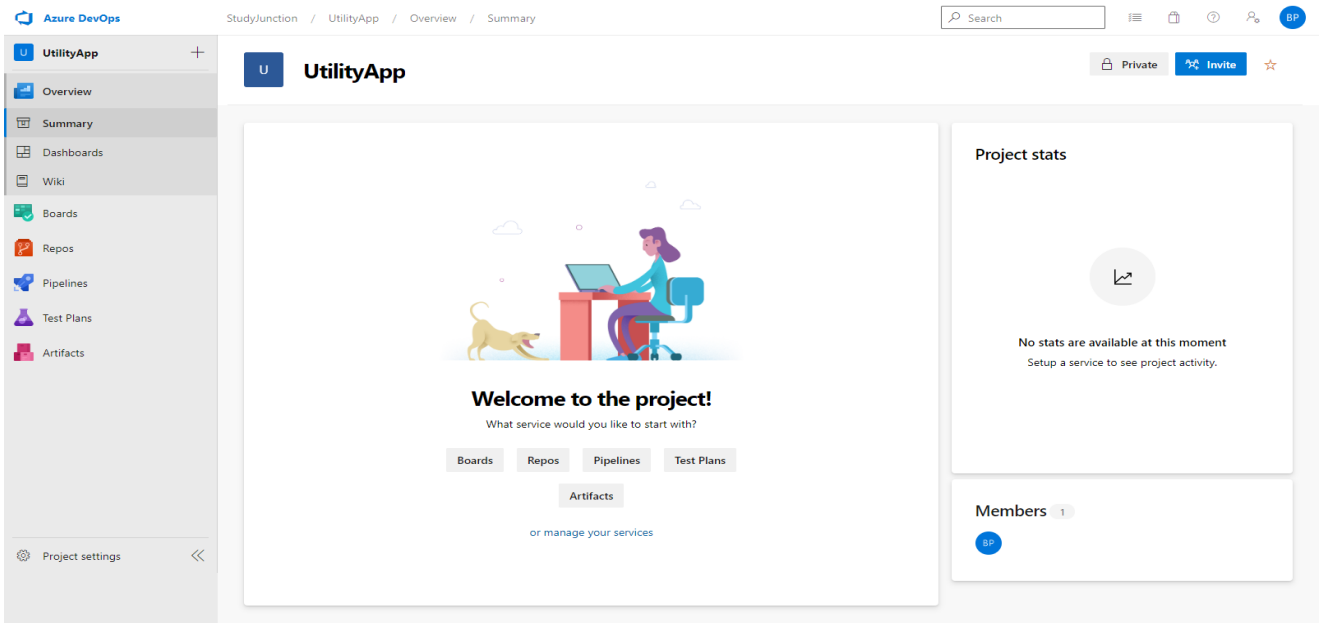
Provides integrated planning and investigation of testing solutions.

Azure Artifacts

Package management Support for Maven, npm, NuGet and Python package feeds from private or public sources.

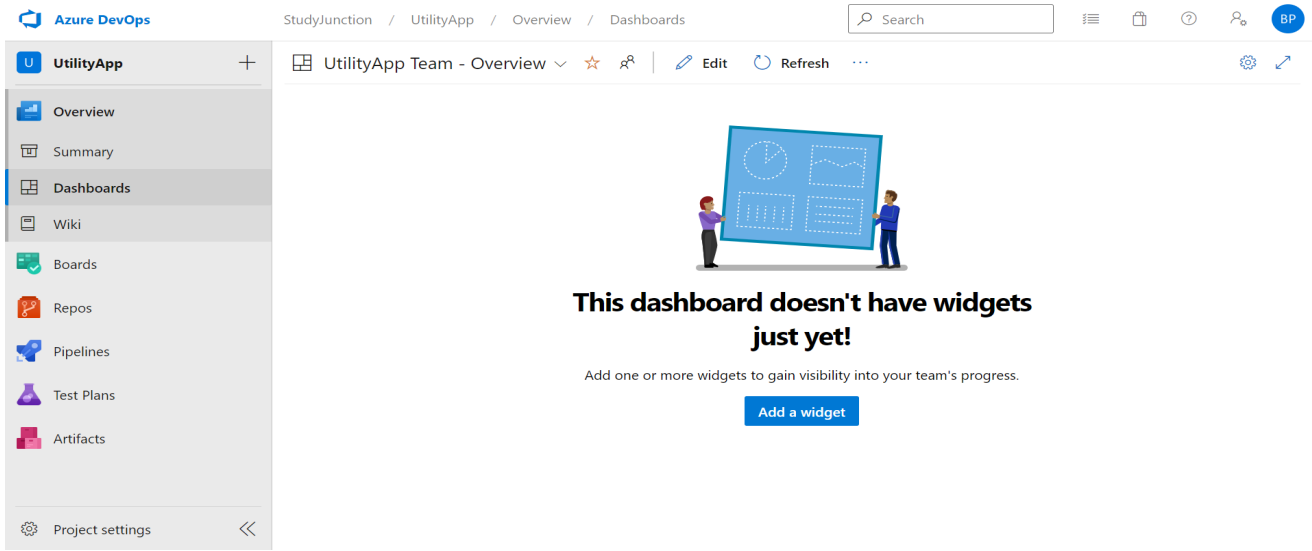
Overview | Summary

The summary page holds information about the project and is the Home Page of your project. Here you can add a basic description, tags to your project and also show a Readme file or a Wiki page. Not only that, but you will also have a quick view on the project stats with an out-of-box dashboard presenting the number of Work items created and the number of Work items completed. Also, you can see the members in the end of the page.



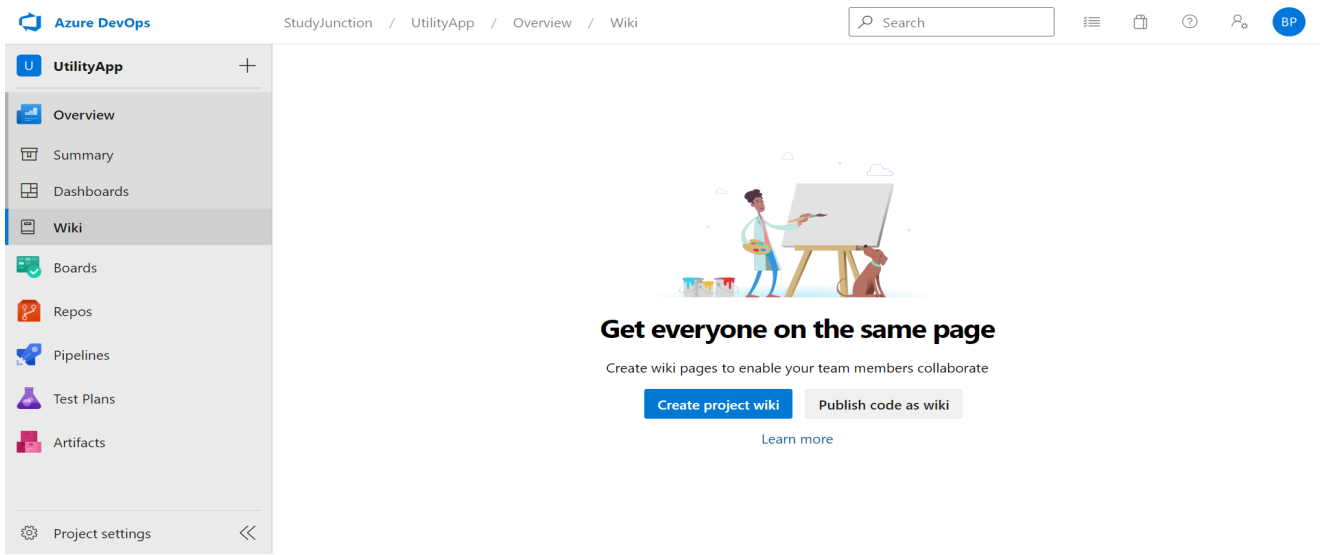
Overview | Dashboards

In the dashboards page it's possible to create dashboards based on data from the work items, build history and commits done by the team. Each dashboard can be a Team Dashboard or a Project Dashboard, so you can target valuable data to each user and their roles.



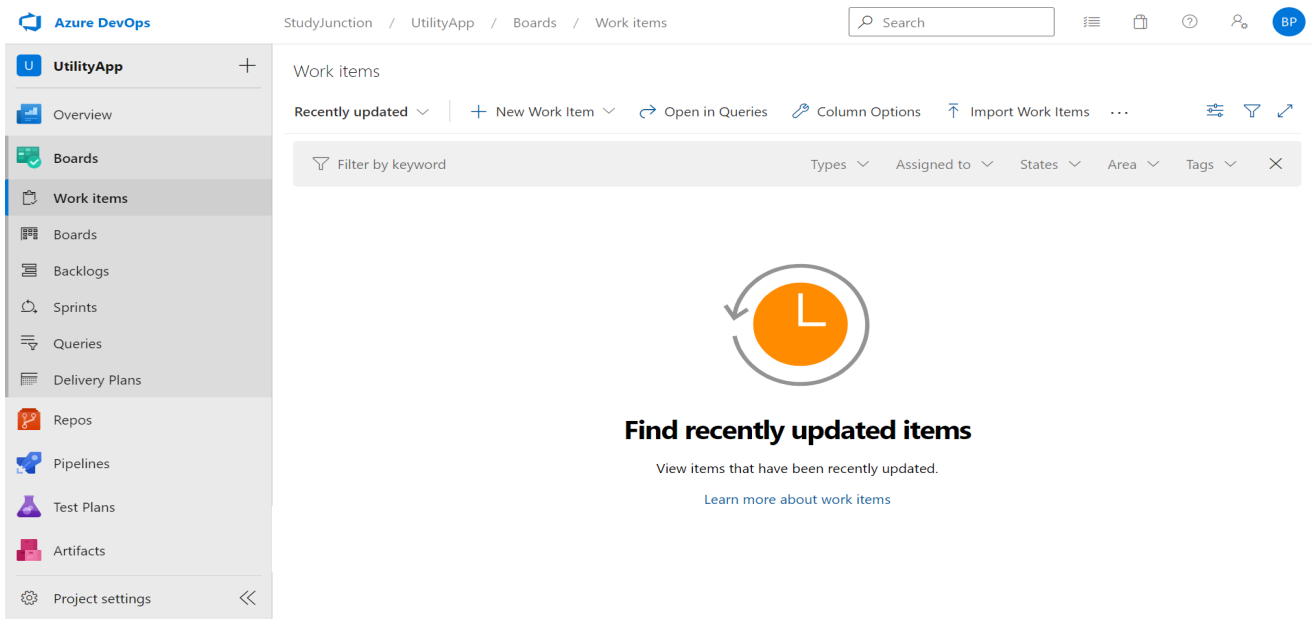
Overview | Wiki

The wiki page present information in Markdown or simple text/HTML. You can also reference some code as a Wiki if you want to reuse some markdown files. This is very important to be created and maintained by your team to be more productive and distribute information easily.



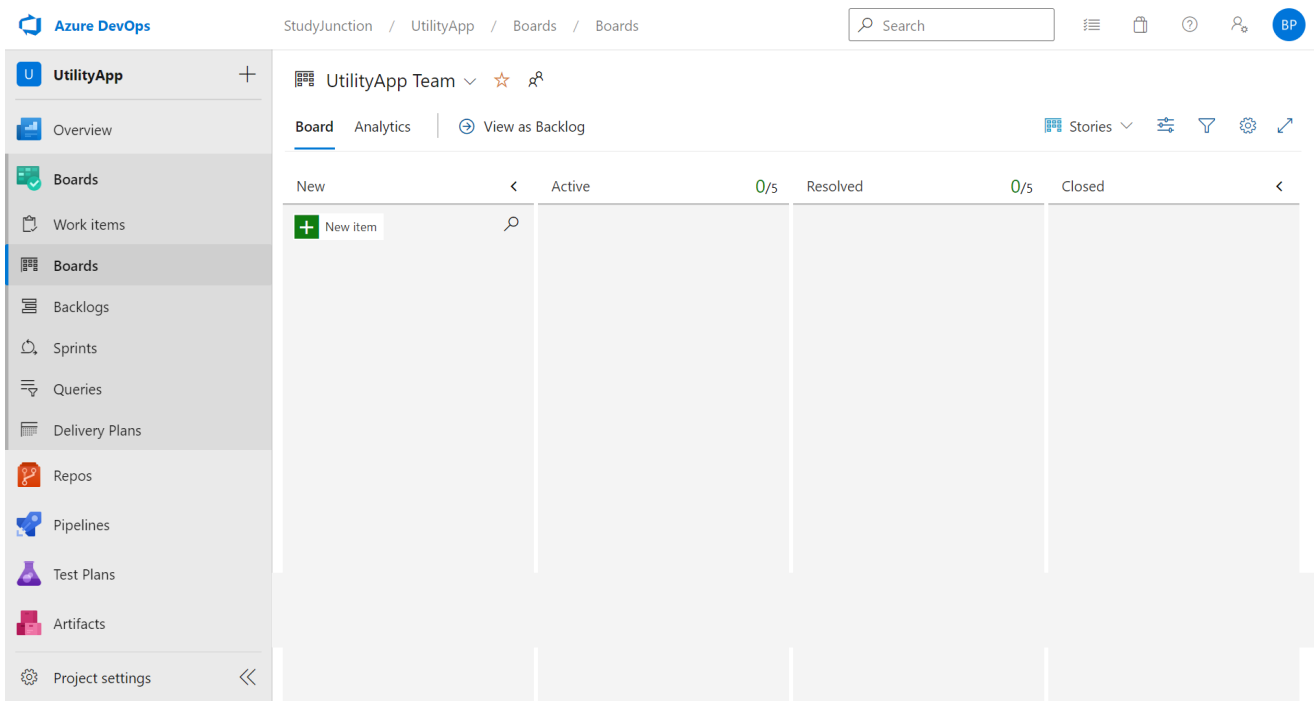
Boards | Work items

The Work items page gives a quick overview about work items not closed in your project. You can easily create a new work item, define new queries, specify columns, delete them and recover from the recycle bin or import new work items.s



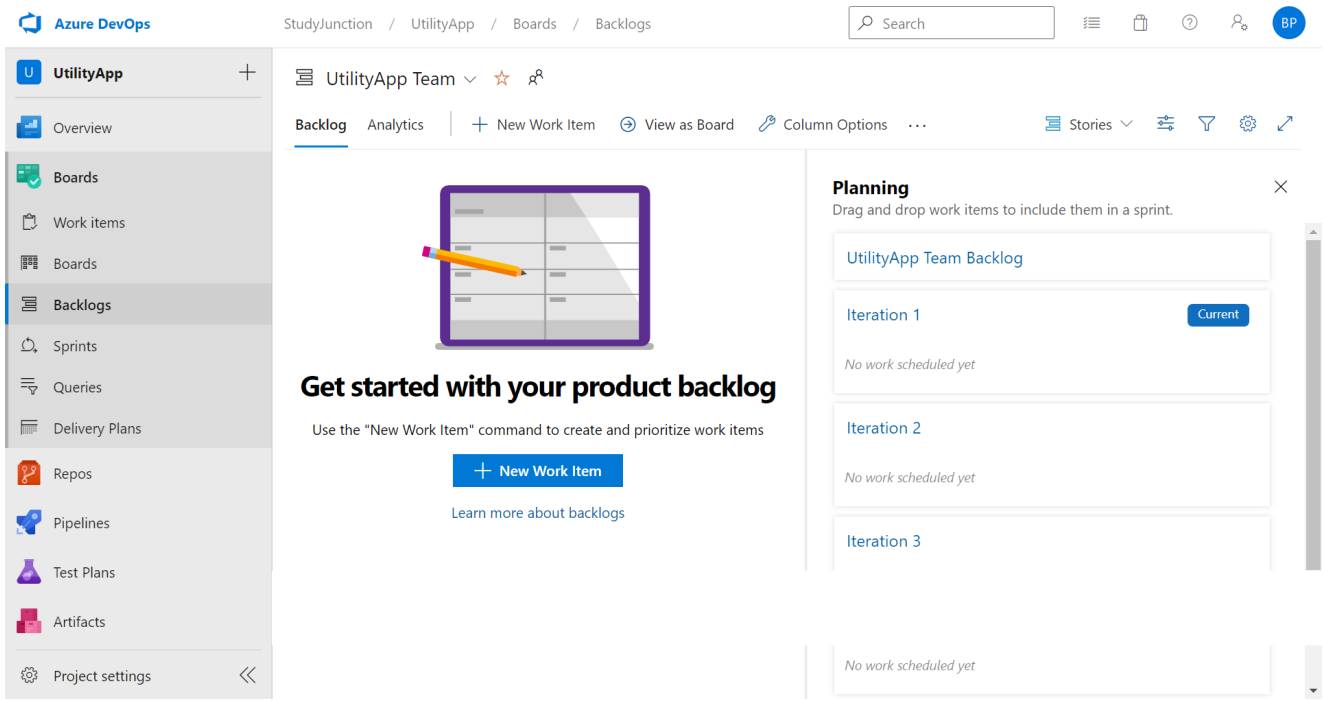
Boards | Boards

The Boards page shows the stories in different boards and in different levels. By default, you will see a board in User Story or Feature level, but you can configure to show at Epic level or configure other types of Work Items.



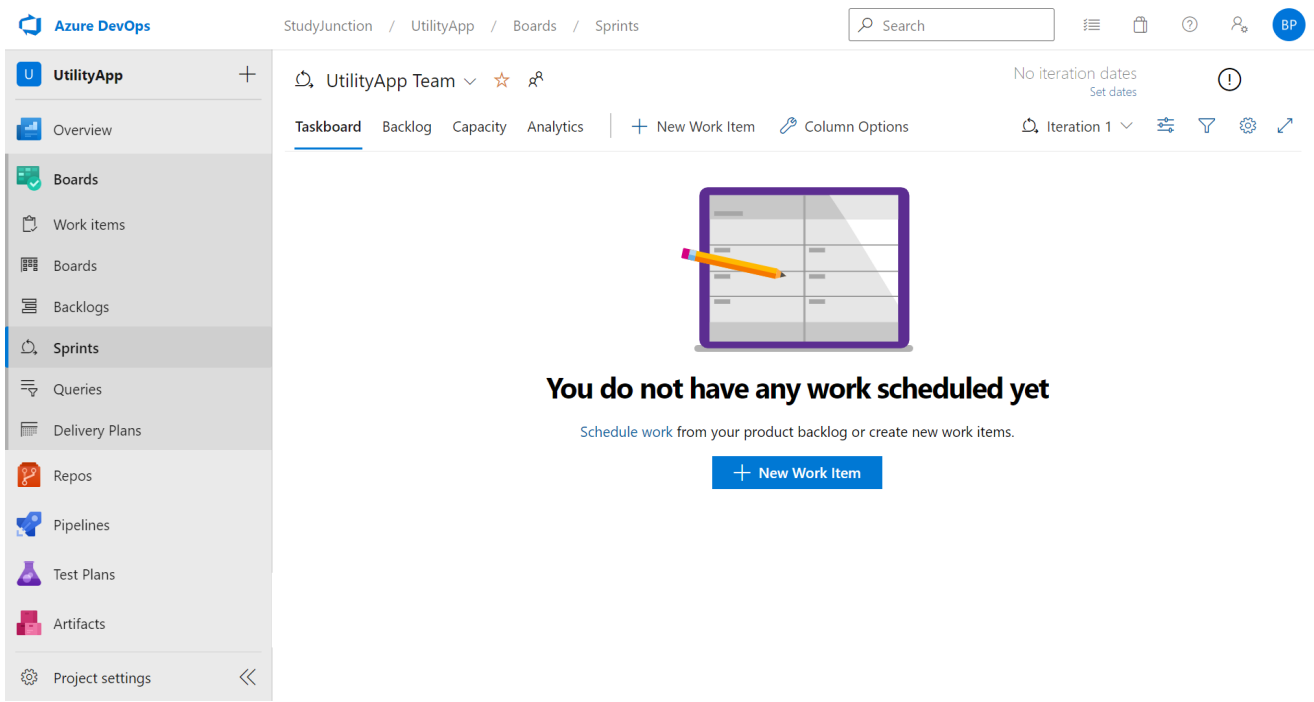
Boards | Backlogs

The Backlogs page presents a complete view of all the backlog in a Tree view like list where you can expand and configure the parent-child relationship of each one of the work items. This is an important view to plan ahead your backlog until you move it to your active sprint.



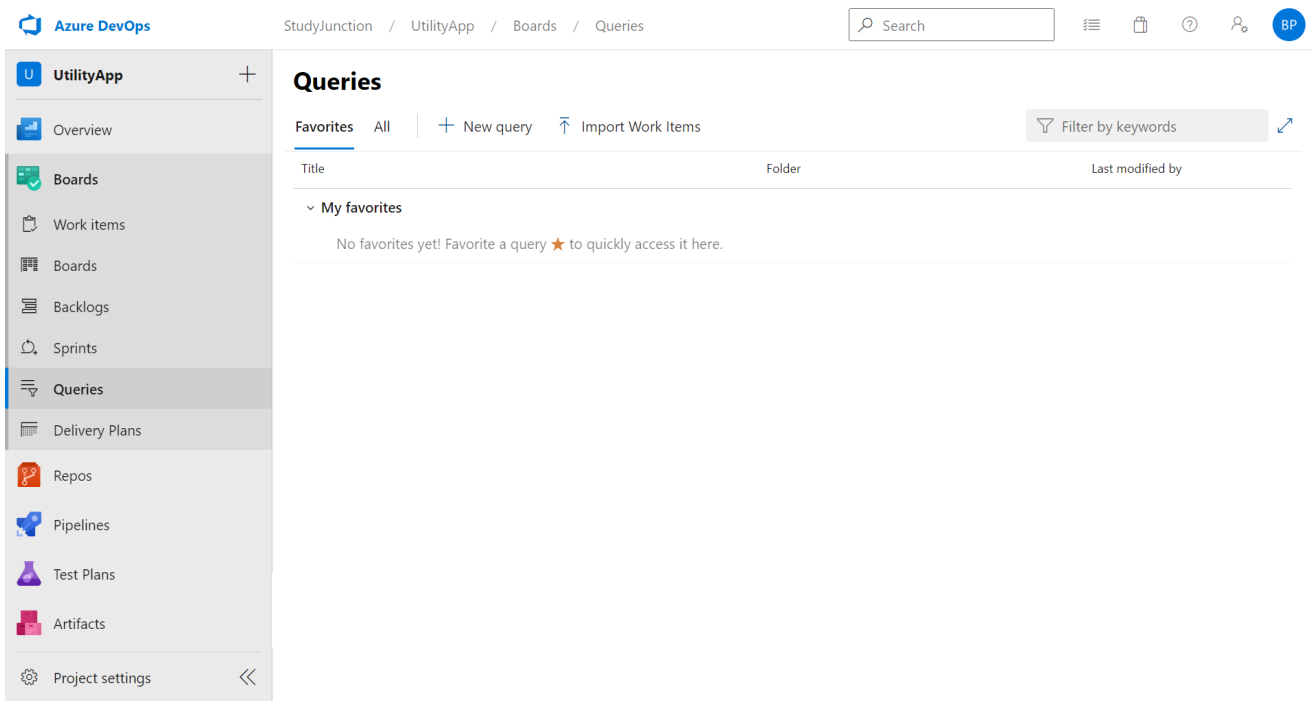
Boards | Sprints

The Sprint page is where your daily work will happen. You can plan and estimate in a granular level (Task or Bug) and effectively track their progress using the columns just like your favorite Kanban board.



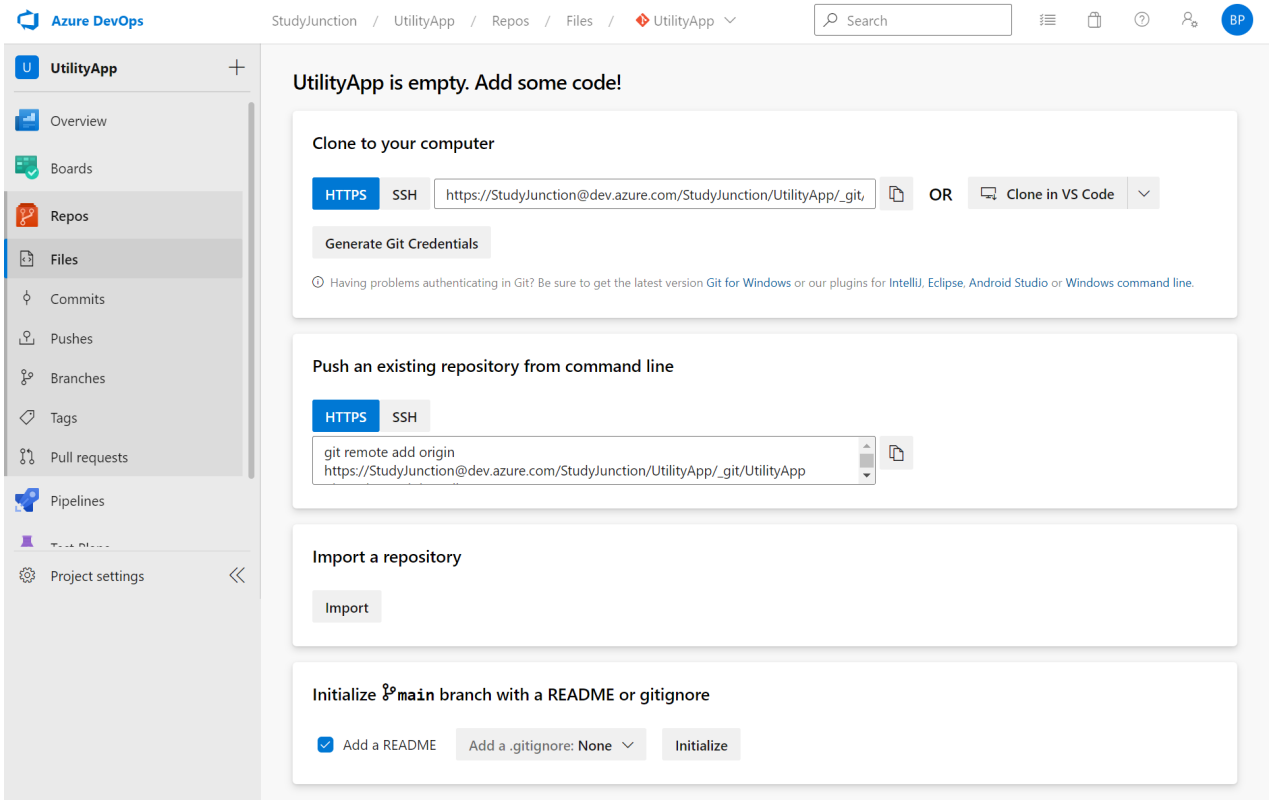
Boards | Queries

The Queries page will contain all the predefined and your favorite queries that you can configure to list the work items you want. This is a very powerful option to find the work items present in a release not completed yet, all the bugs from the past sprint, etc.



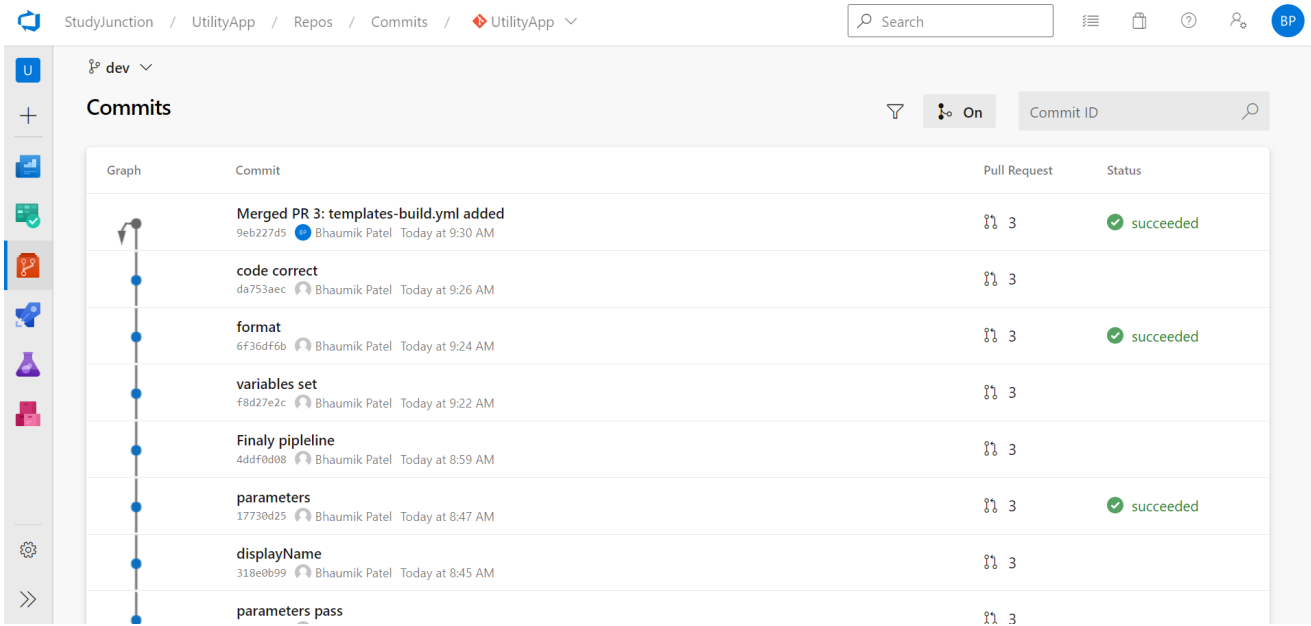
Repos | Files

The Files page offers you a view of the files available in your project, organized by project/branches. There you can select the branch you want to look for, navigate to other projects and here you even can access the Clone button. The clone button allows you to clone with different protocols, so you can start working from your machine.



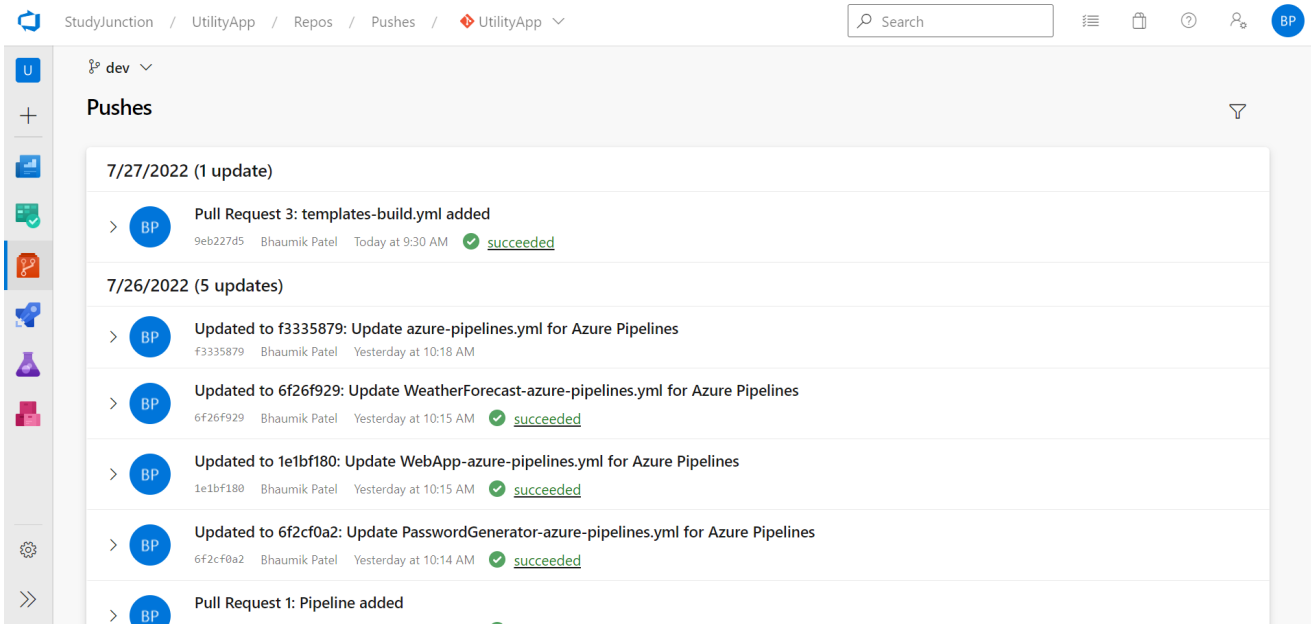
Repos | Commits

The Commits page lists all the commits available for each branch and lets you filter by a commit ID to find and recover one specific version. Not only that, but you can also see the Commit graph, just to understand better your commit timeline.



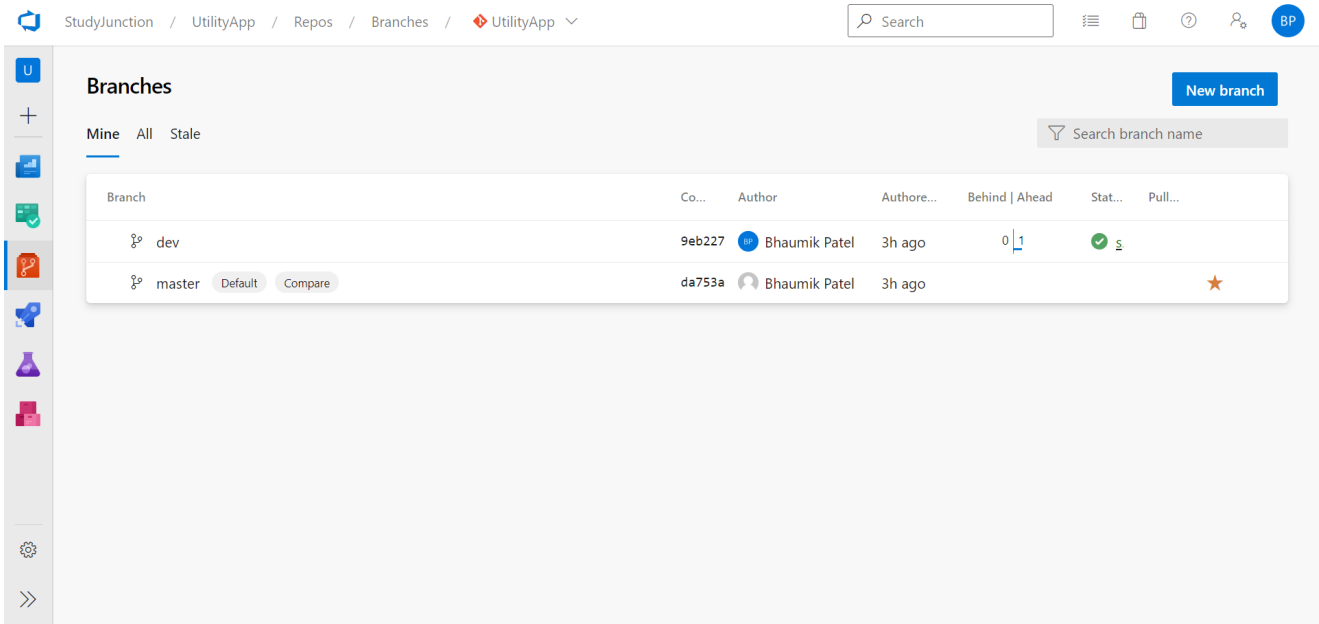
Repos | Pushes

The Pushes page is very similar to Commits page, but the difference here is to show when the push event happened for each branch and how many commits were contained inside the push.



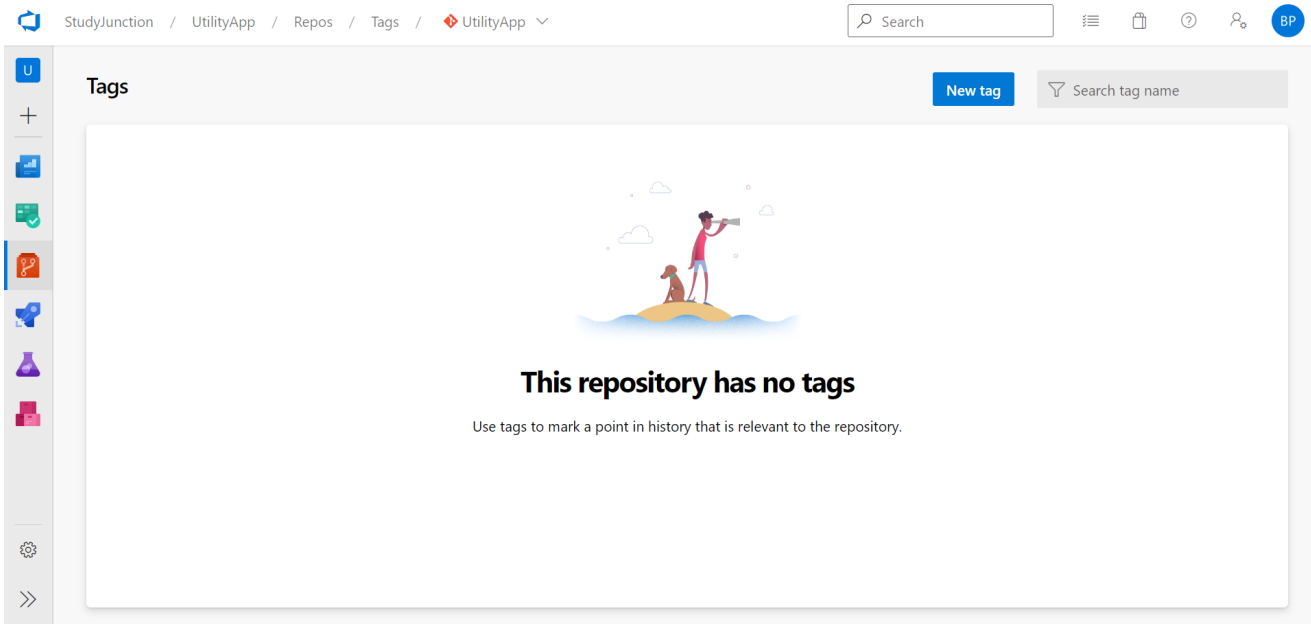
Repos | Branches

The Branches page is pretty straightforward. There you can find all branches, the ones you created and the stale ones aging for quite some time. You can take advantage of the Behind/ Ahead column to understand how long one branch is lagging when comparing to the main branch, usually the master branch.



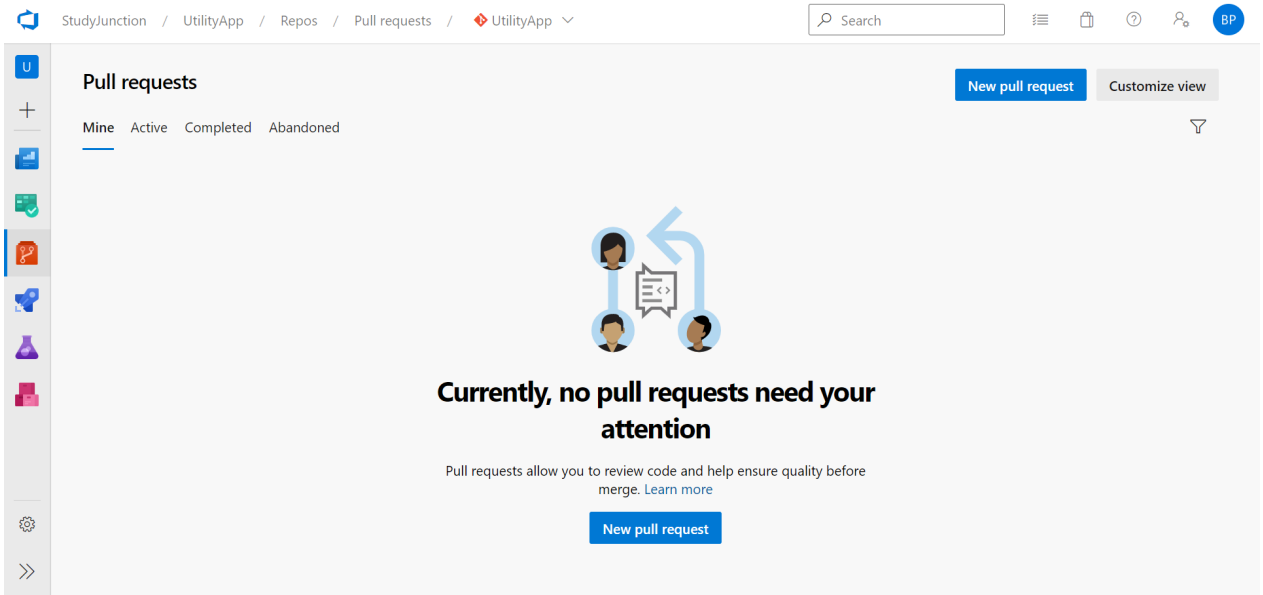
Repos | Tags

The Tags page gives you a shortcut to finding the Tags of your releases and code versions. It's a nice way to find the tags created by your team to control the versioning of your releases.



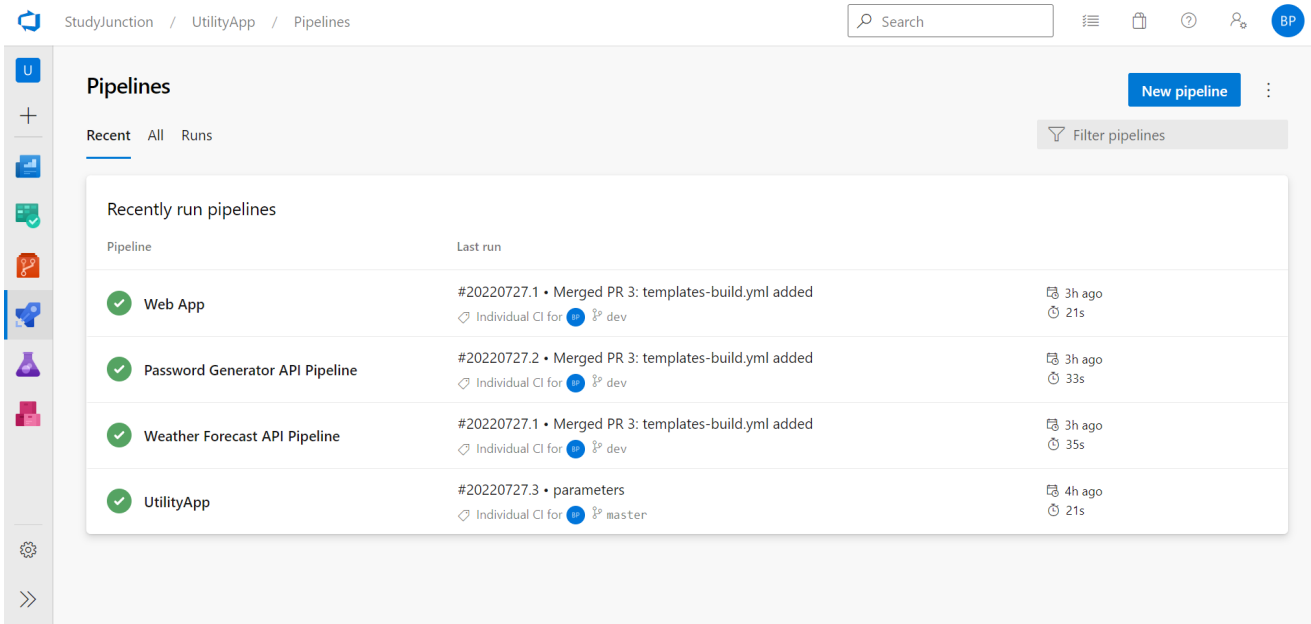
Repos | Pull requests

The Pull requests page is another quite useful page where you can find the Pull Requests you have opened, the ones Active and finally the PRs completed or abandoned. Every time you push new changes to a specific branch, it will show up here with a shortcut to open a new Pull Request to the default branch.



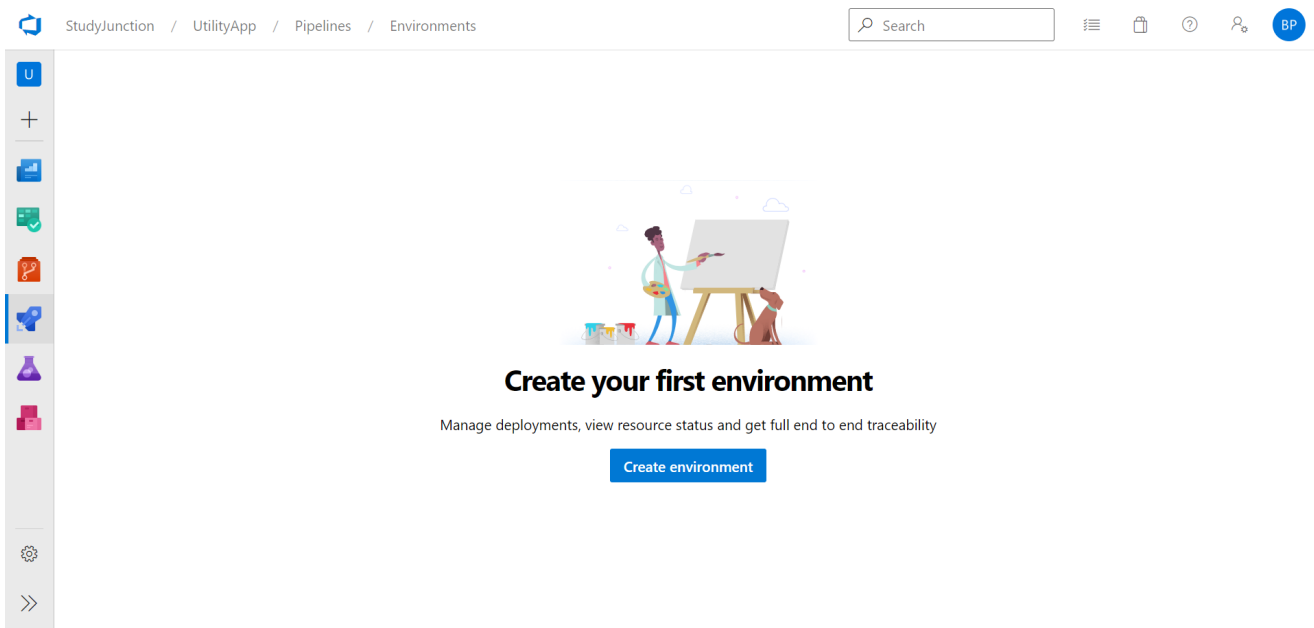
Pipelines | Pipelines

It is a workflow that defines how our test, build, and deployment steps are run.



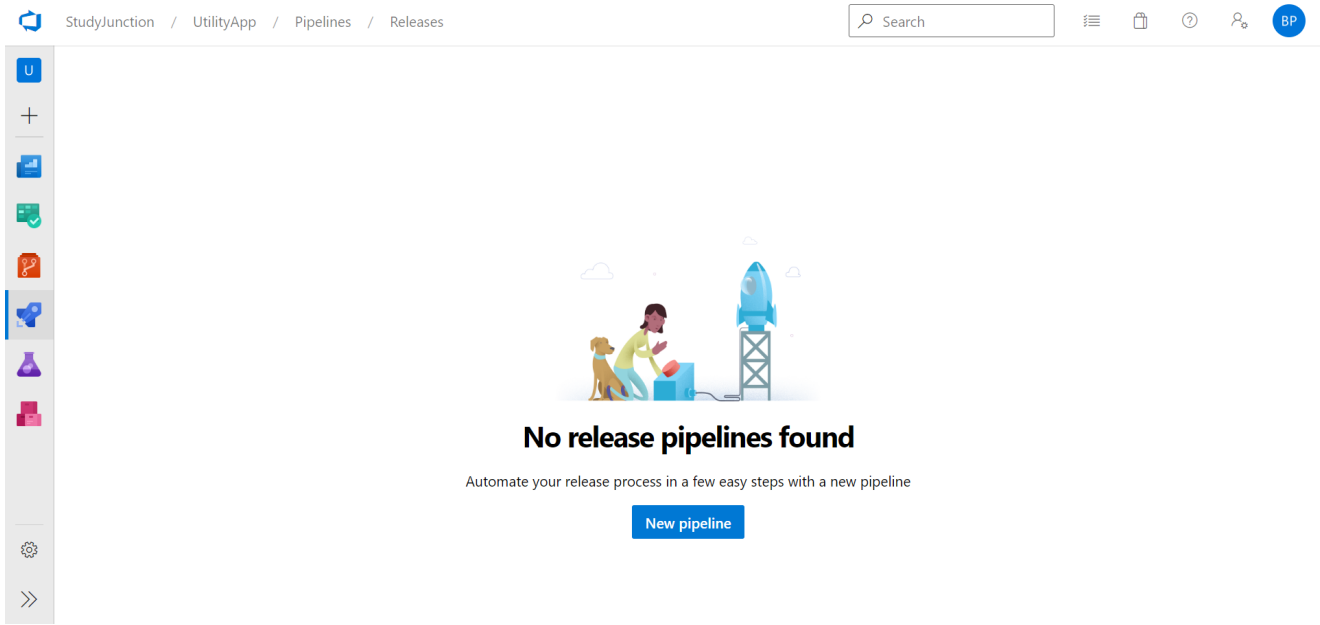
Pipelines | Environment

It is a collection of resources, where you deploy your application. It contains one or more virtual machines, containers, web apps, etc.



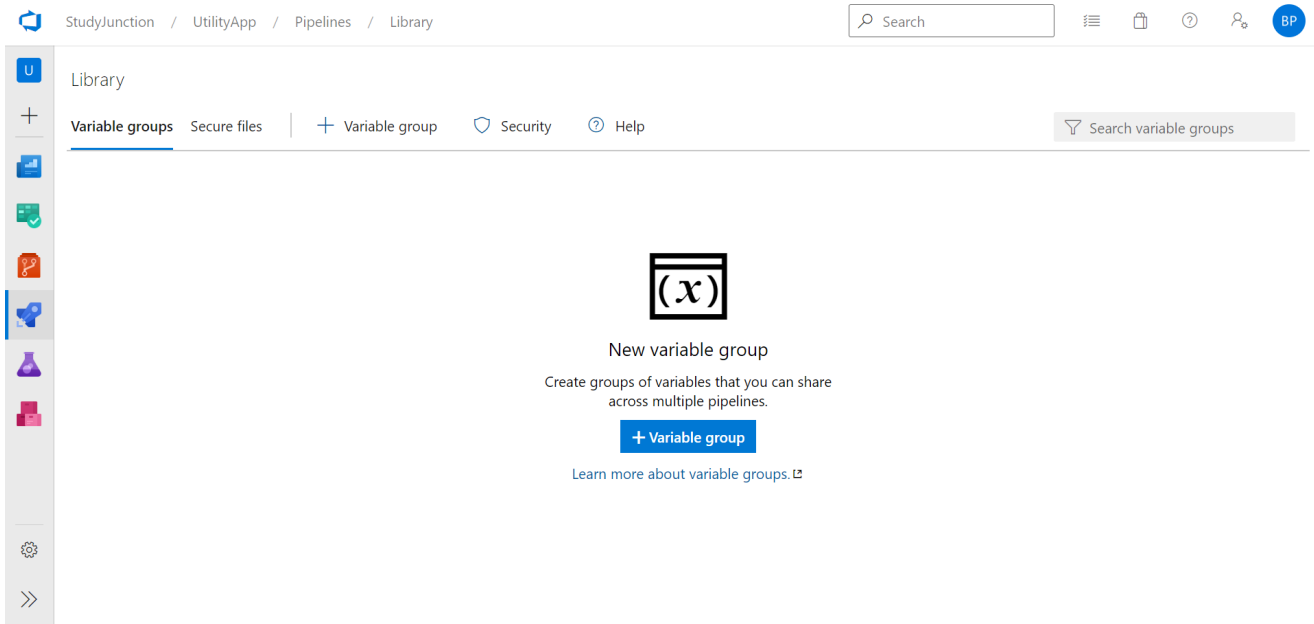
Pipelines | Releases

Release pipelines in Azure Pipelines help your team continuously deliver software to your customers at a faster pace and with lower risk. You can fully automate the testing and delivery of your software in multiple stages all the way to production.



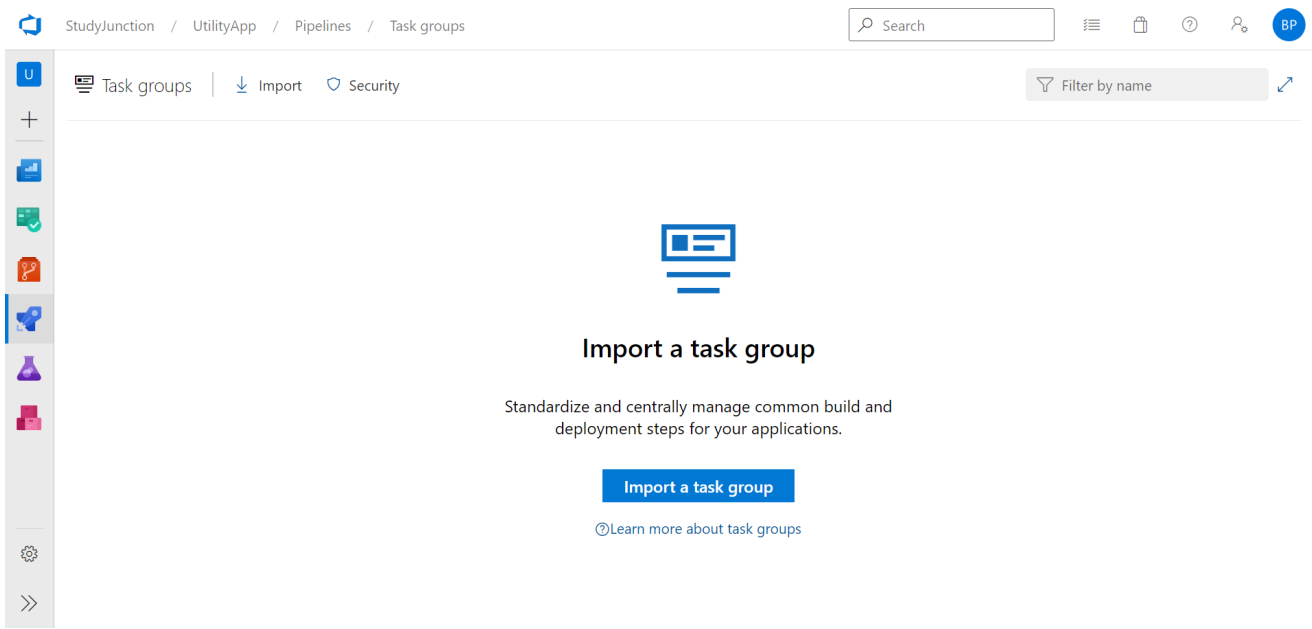
Pipelines | Library

Variable groups store values and secrets that you might want to be passed into a YAML pipeline or make available across multiple pipelines. You can share and use variable groups in multiple pipelines in the same project.



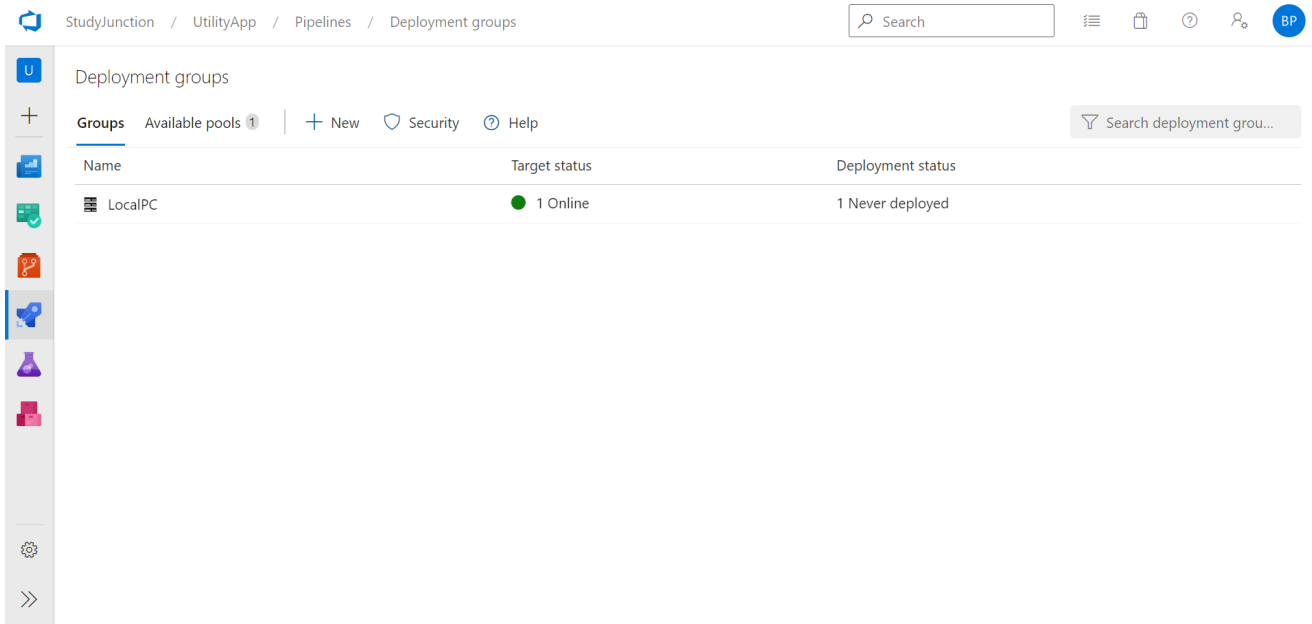
Pipelines | Task groups

A task group allows you to encapsulate a sequence of tasks, already defined in a build or a release pipeline, into a single reusable task that can be added to a build or release pipeline, just like any other task.



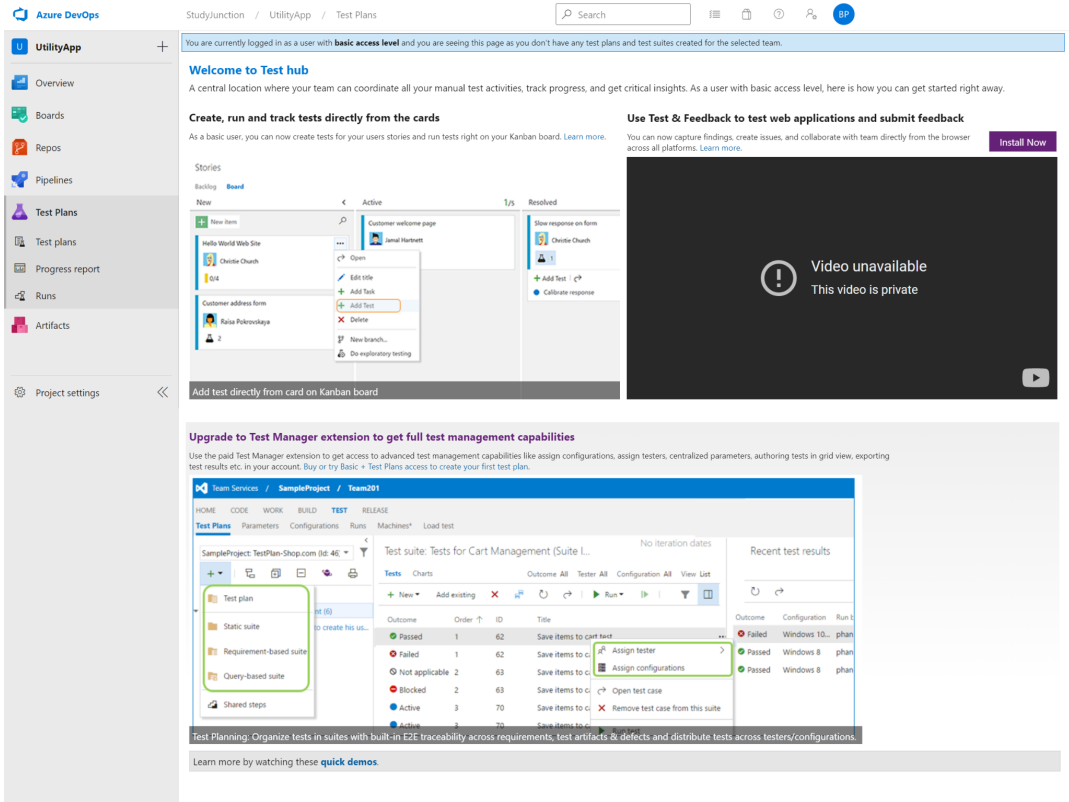
Pipelines | Deployment groups

A deployment group is a logical set of deployment target machines that have agents installed on each one. Deployment groups represent the physical environments; for example, "Dev", "Test", or "Production" environment. In effect, a deployment group is just another grouping of agents, much like an agent pool.



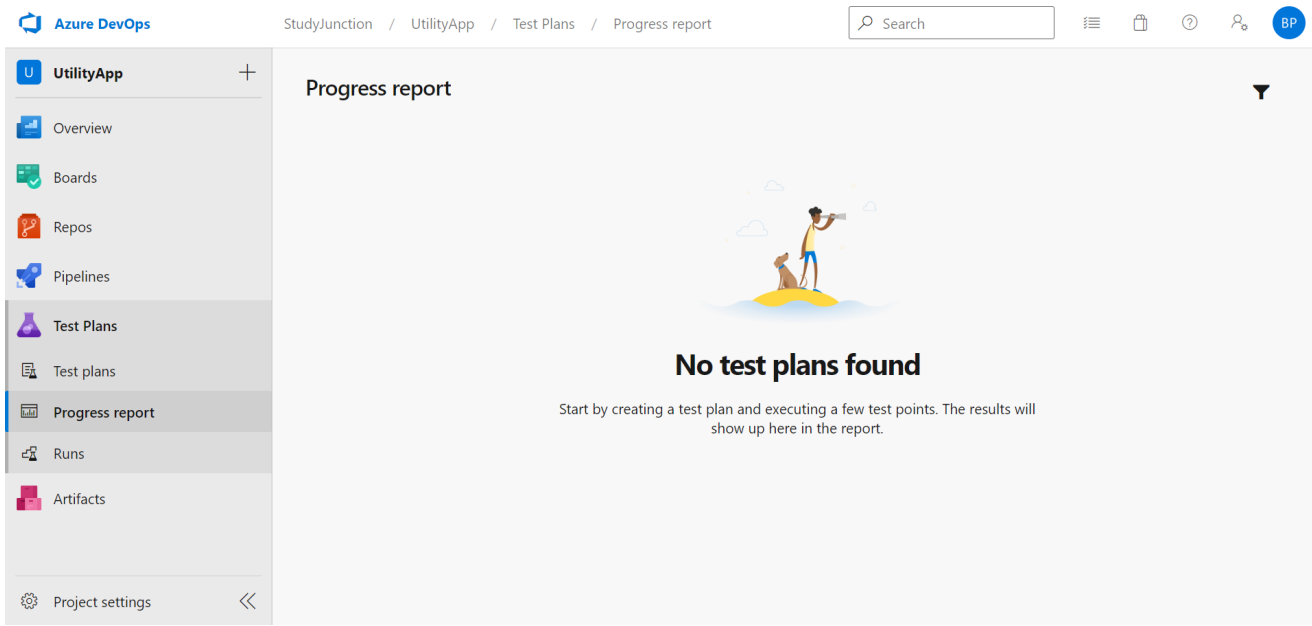
Test Plans | Test plans

A central location where your team can coordinate all your manual test activities, track progress, and get critical insights.



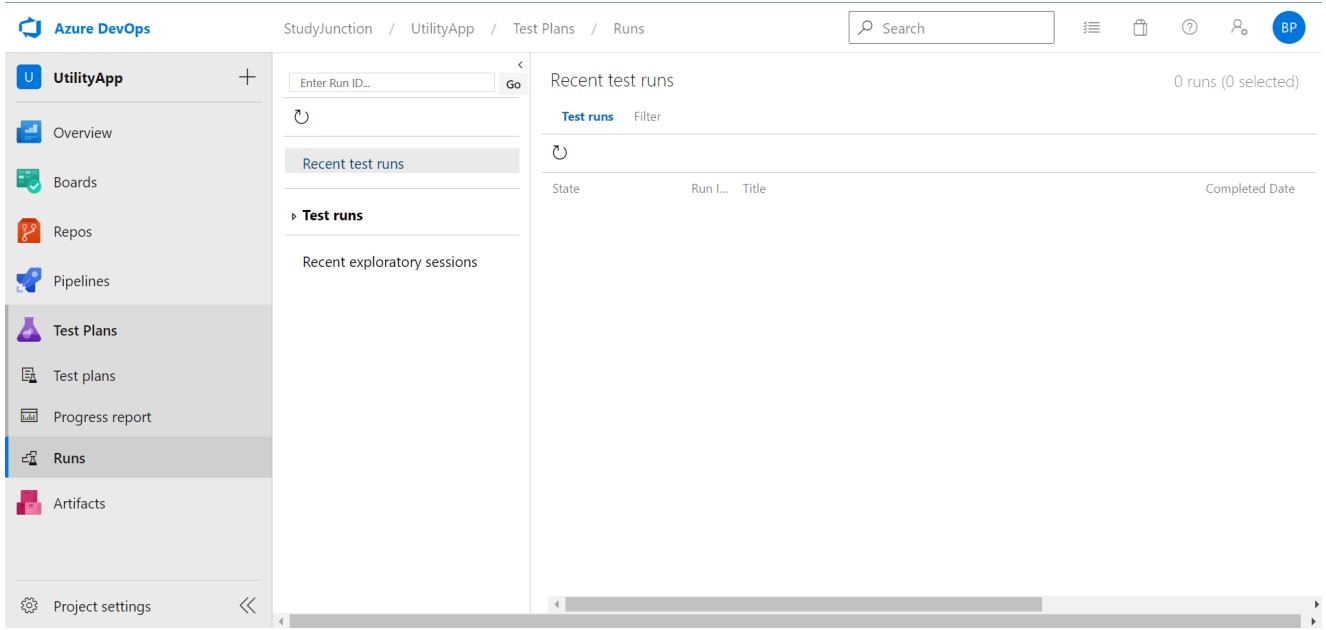
Test Plans | Progress report

To track the progress of more than one test plan or test suite, use the Progress Report.



Test Plans | Runs

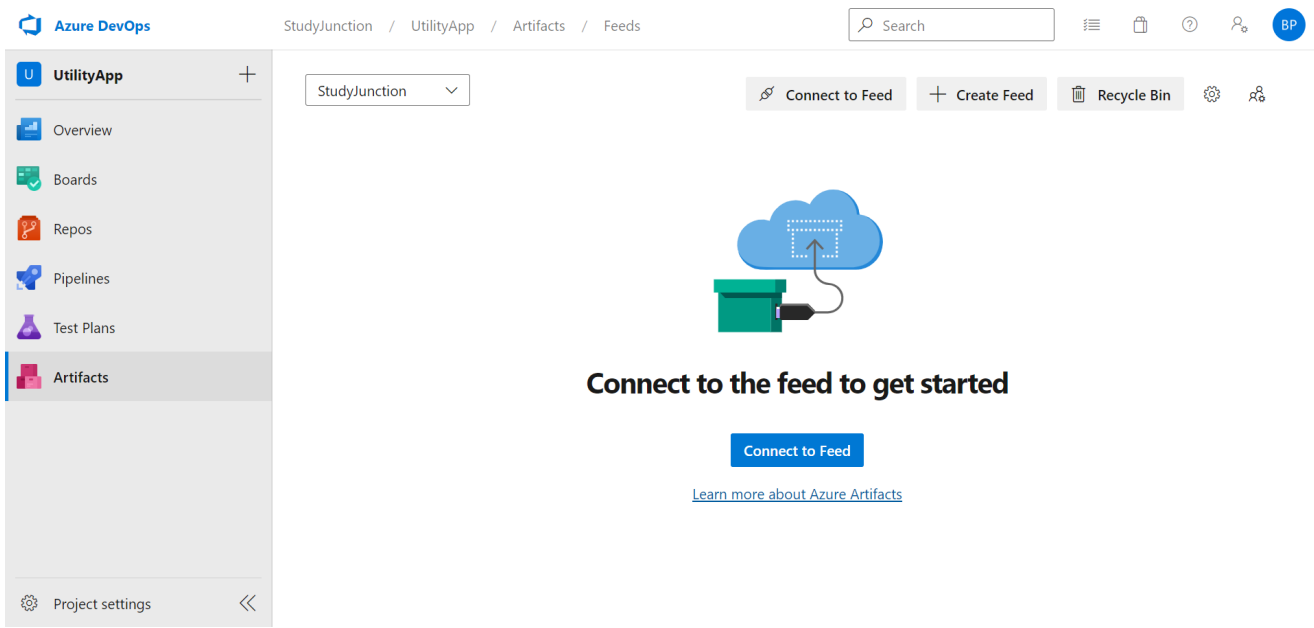
Automate test cases in your test plans and run them directly from Azure Test Plans.



Artifact

It is a collection of files or packages published by a run. The Artifact is made available to subsequent tasks, such as distribution or deployment.

Allows the teams to easily package the dependencies and other artifacts required for the application deployment and its functionality, thus making it easier to publish and consume the application. There can be different kinds of artifacts such as – Build Artifacts, Maven, Npm, Nuget, Universal Packages and Symbols



Thank You!