Contents

New User Guide

8/10/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

New to Visual Studio Team Services (VSTS)?

We'll walk you through the basics of the web interface and some common tasks.

5-Minute Quickstarts

Get started with a free organization, share your code using Git, and track your work using a Kanban board.

- Sign up and invite some teammates
- Code with Git
- Set up continuous integration & delivery
- Plan & track work
- Add and run manual tests
- Install a Marketplace extension
- View permissions

Step-by-Step Tutorials

- Set favorites
- Follow work and pull requests
- Get started as a Stakeholder

Concepts

- Key concepts
- Source control
- Kanban & Agile tools
- Clients and tools
- Software development roles
- What's the difference between VSTS and TFS?

How-to Guides

- Enable preview features
- Use @mentions to further discussion
- Use #ID to link to work items
- Navigate using your organization home page
- Go mobile
- Manage your notifications
- Set your preferences
- Sign in to the web or a client

Reference

- Permissions and access
- Keyboard shortcuts
- Markdown guidance
- Navigate in the web portal
- Navigate in Team Explorer
- FAQs

Resources

- Visual Studio IDE
- Visual Studio Code
- Visual Studio for Mac
- Organization Management
- Security & Identity
- Technical articles

What is VSTS?

8/10/2018 • 2 minutes to read

VSTS

Visual Studio Team Services (VSTS) is a cloud service for collaborating on code development. It provides an integrated set of features that you access through your web browser or IDE client, including:

- Git repositories for source control of your code
- Build and release management to support continuous integration and delivery of your apps
- Agile tools to support planning and tracking your work, code defects, and issues using Kanban and Scrum methods
- A variety of tools to test your apps, including manual/exploratory testing, load testing, and continuous testing
- Highly customizable dashboards for sharing progress and trends
- Built-in wiki for sharing information with your team

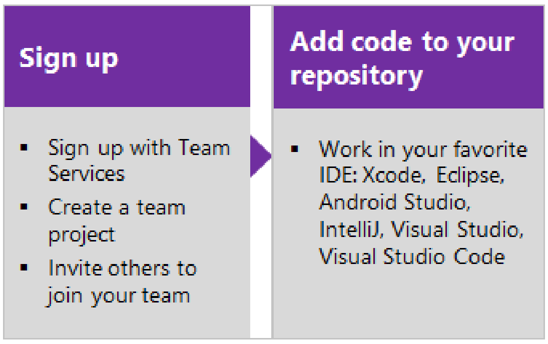
In addition, the VSTS ecosystem provides support for adding extensions, integrating with other popular services, such as: Campfire, Slack, Trello, UserVoice, and more, and developing your own custom extensions.

Choose VSTS when you want quick setup, maintenance-free operations, easy collaboration across domains, elastic scale, and rock solid security. You'll also have access to cloud load testing, cloud build servers, and application insights.

Where do I start?

We've made it easy for you to start for free and try out our service.

Sign up for free by creating an organization and then, either upload your code to share or source control, or begin tracking your work using Scrum, Kanban, or a combination of methods.



Try this next

Sign up and invite teammates

Related articles

- A tour of services
- Key concepts
- Client-server tools
- Software development roles
- Pricing - VSTS

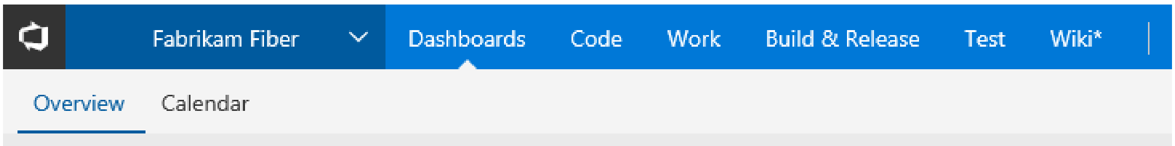
What do I get with VSTS or TFS?

8/9/2018 • 5 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

With VSTS and TFS, you gain an integrated set of services and tools to manage your software projects, from planning and development through testing and deployment. Services are delivered through a client-server model, many of which are delivered through an easy-to-use web interface that you can access from all major browsers. Some services, such as source control, build pipelines, and work tracking can also be managed through a client.

Access Web services through the following main hubs as shown in the following image.

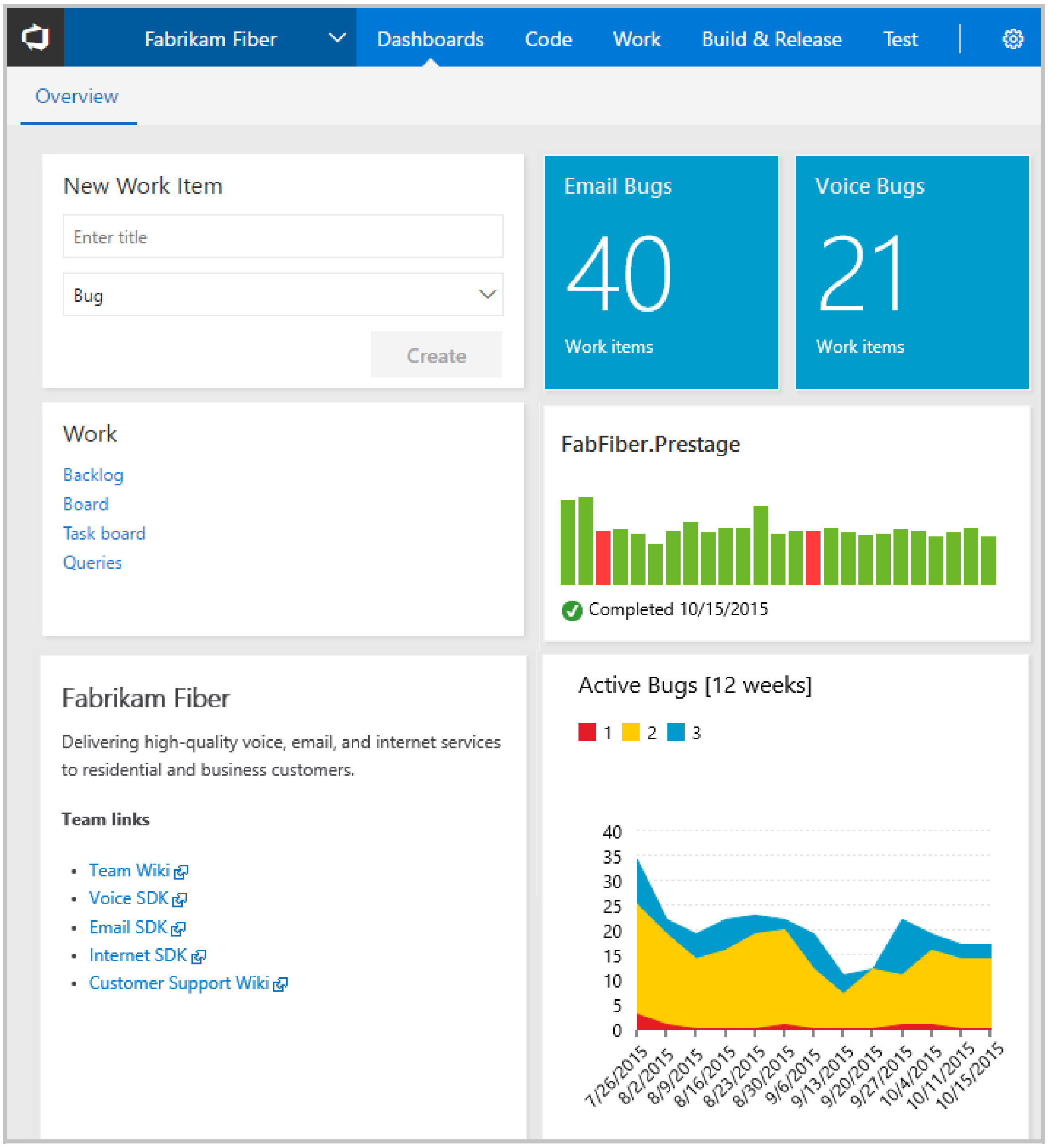


Many of our services are either free for small teams or available through a subscription model or per use model. Where needed, you can exercise a hybrid approach where you use an on-premises TFS to manage your code and work, and purchase cloud build or testing services on an as needed basis.

For information about client tools, see Tools.

Dashboards

From the **Dashboards** hub you gain access to user-configurable dashboards.



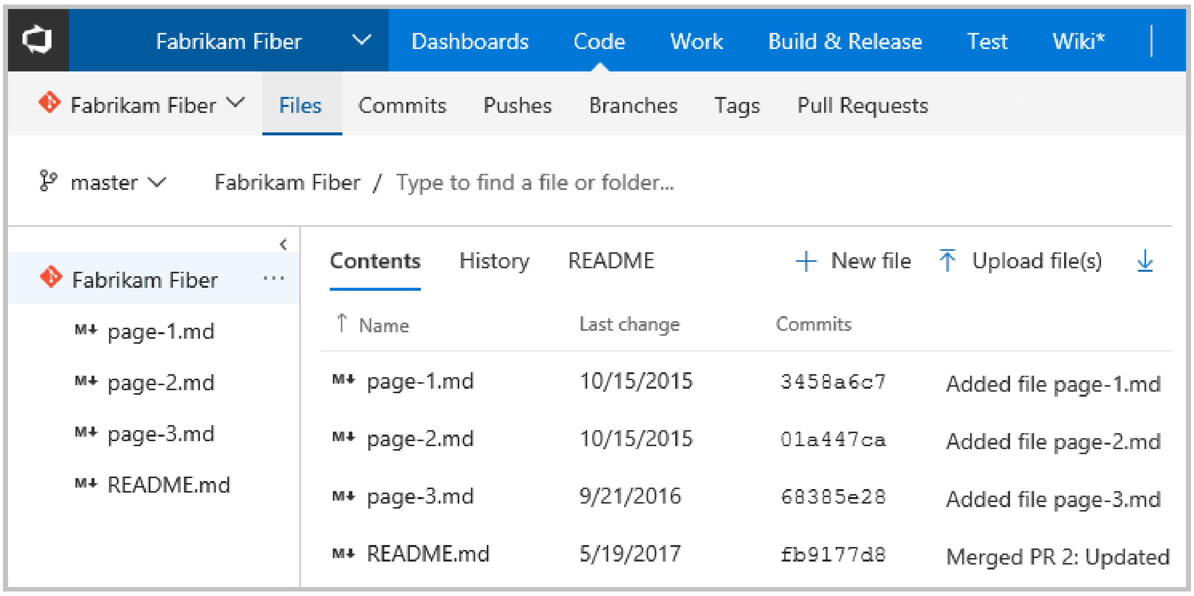
Tasks you can perform in this hub include:

- Add, configure, and manage dashboards
- Configure widgets that you add to dashboards
- Quickly navigate to different areas of your team project

To learn more, see Dashboards.

Code

From the **Code** hub you gain access to your source control Git-based or TFVC repositories to support version control of your software projects. These repositories are private.



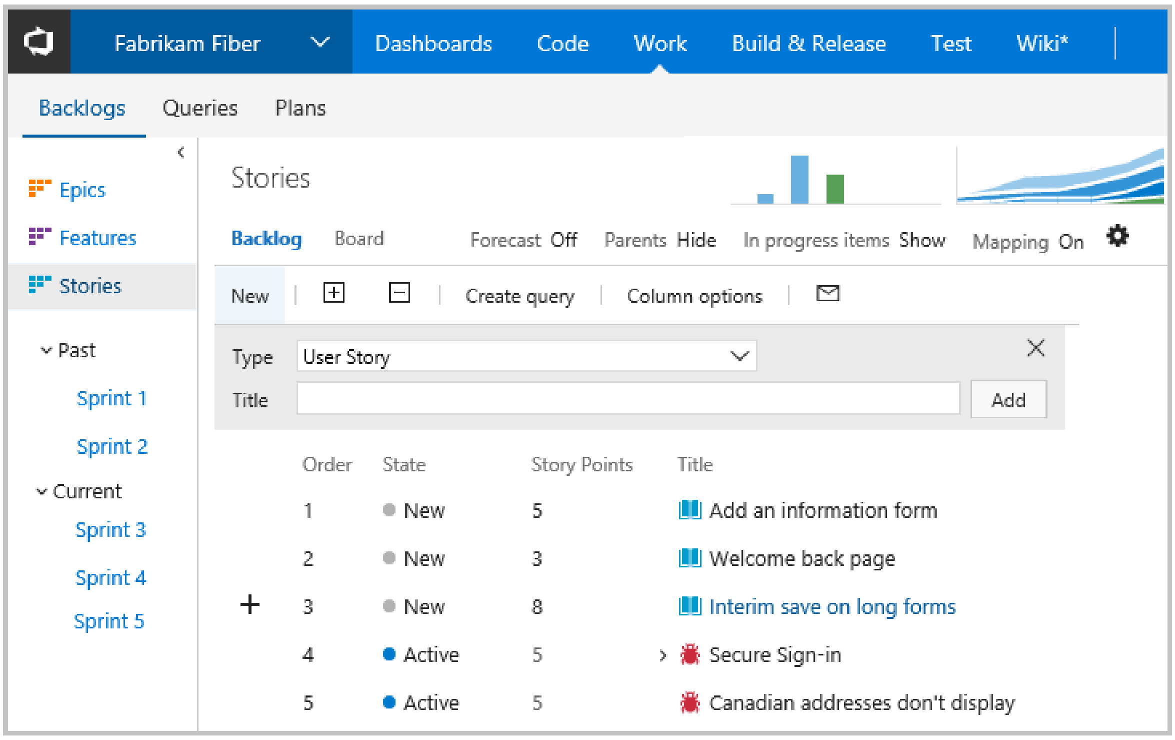
From the Code hub for Git you can perform these tasks:

- Review, download, and edit files and review the change history for a file
- Review and manage commits that have been pushed
- Review, create, approve, comment, and complete pull requests
- Add and manage Git tags

To learn more, see the overviews for Git or TFVC.

Plan and track work

From the **Work** hub you gain access to Agile tools to support planning and tracking work.



Specifically, from the **Work** you can perform these tasks:

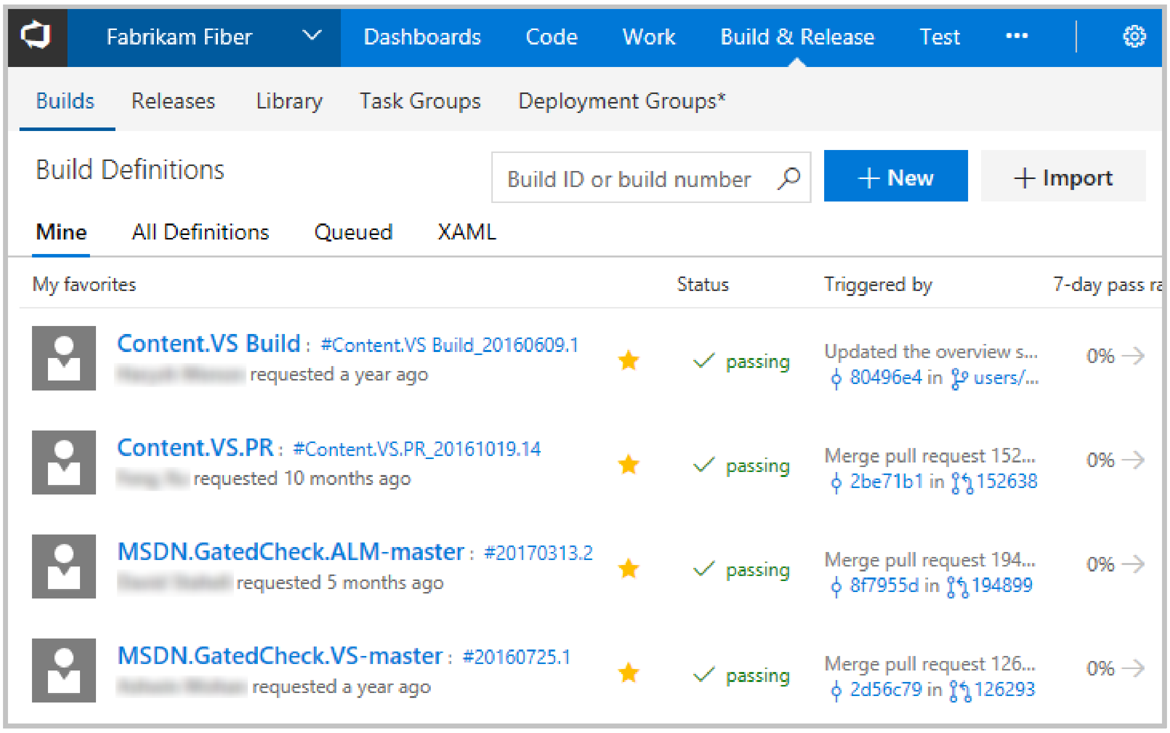
- Add and update work items
- Define work item queries and create status and trend charts based on those queries
- Manage your product backlog
- Plan sprints using sprint backlogs

- Review sprint tasks and update tasks through the task boards
- Visualize the work flow and update status using Kanban boards
- Manage portfolios by grouping stories under features and features under epics

See Backlogs, boards, and plans for an overview of each.

Build & Release

The **Build & Release** hub provides an integrated set of features to support building and deploying your applications.



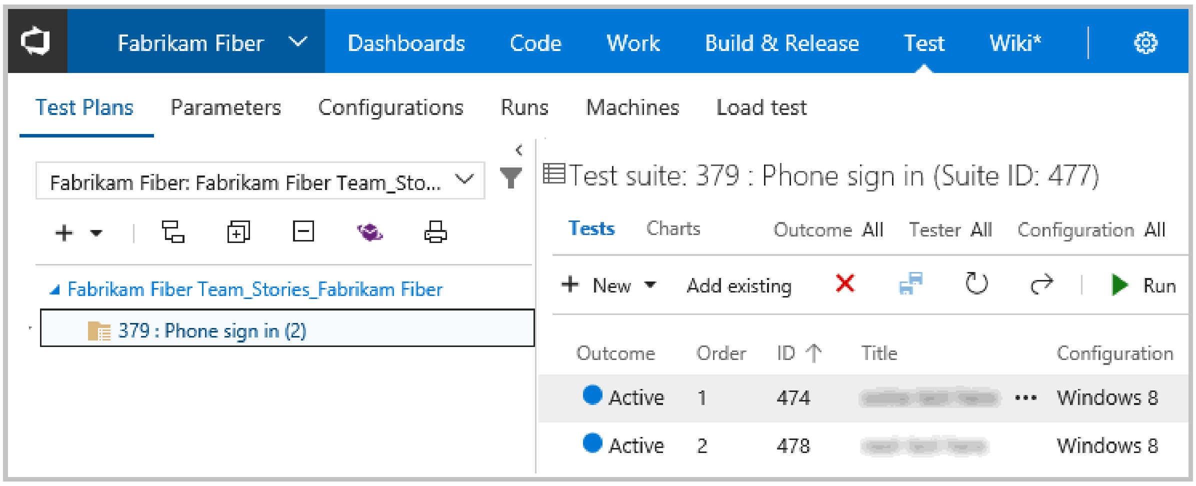
Use this hub to implement continuous integration and continuous delivery.

- **Build automation**: Define the steps to take during build and the triggers that will initiate a build.
- **Release management**: Supports a rapid release cadence and management of simultaneous releases. You can configure release pipelines that represent your environments from development to production. Run automations to deploy your app to each environment. Add approvers to sign off that the app has been successfully deployed in an environment. Create your release manually or automatically from a build. Then track your releases as they are deployed to various environments.

To learn more, see Continuous integration on any platform.

Test

Test features support manual and exploratory testing, load or performace testing, and continuous testing. Here we show the **Test** hub that support creating and managing manual tests.



Test **Test** hub provides support for the following:

- Customization of workflows with test plan, test suite and test case work items
- End-to-end traceability from requirements to test cases and bugs with requirement-based test suites
- Criteria-based test selection with query-based test suites
- Excel-like interface with the grid for easy test case creation
- Reusable test steps and test data with shared steps and shared parameters
- Sharable test plans, test suites and test cases for reviewing with stakeholders
- Browser-based test execution on any platform
- Real-time charts for tracking test activity.

To learn more, see Testing overview.

Collaboration services

In addition to the above "hub" services, the following services work across hubs to support:

- Linking of work items, commits, pull requests and other artifacts to support traceability
- Alerts and change notifications managed per user or for teams
- Request and manage feedback
- Team (chat) rooms
- Reporting

NOTE

Team rooms are deprecated for TFS 2017.2. Instead, we recommend you use service hooks to integrate with Microsoft Teams.

Service hooks

Service hooks enable you to perform tasks on other services when events happen within your team project hosted on VSTS or TFS. For example, you can send a push notification to your team's mobile devices when a build fails. Service hooks can also be used in custom apps and services as a more efficient way to drive activities when events happen in your projects.

The following services are available as the target of service hooks. To learn about others apps and services that integrate with our VSTS or TFS, visit the Visual Studio Marketplace

For the latest set of supported services, see Integrate with service hooks

Cloud-hosted services based on usage

The following services support your DevOps operations.

- Cloud-based build and deployment Microsoft-hosted agents
- On-premises self-hosted agents to support build and deployment
- Cloud-based performance/load testing lets you load test your code by simulating high traffic

To learn more, see Pricing.

Azure services

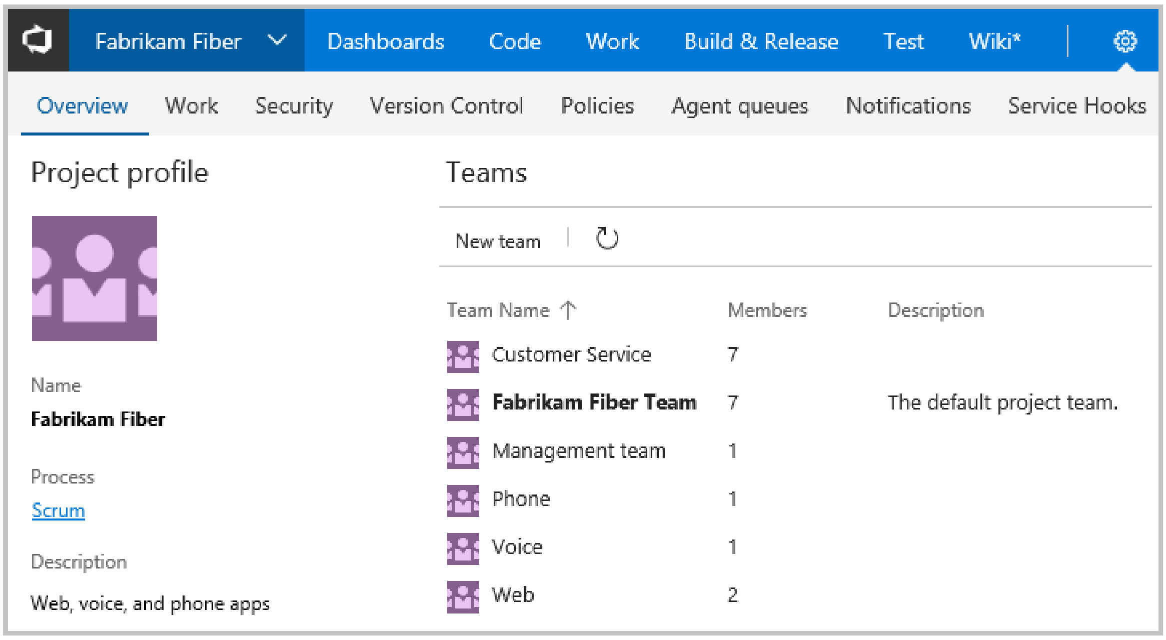
Azure provides a number of cloud-hosted services to support application development and deployment. You can make use of these services solely or in combination with VSTS or TFS.

To browse Azure's directory of integrated services, features, and bundled suites, see Azure products.

For continuous delivery to Azure from VSTS, see Automatically build and deploy to Azure web apps or cloud services.

Administrative services

There are a number of features and tasks associated with administrating a collaborate software development environment. You perform most of these tasks through the web portal.



ORGANIZATION MANAGEMENT (VSTS)	ADMINISTRATION (TFS)
- Manage users, access, and billing - Add and manage team projects and teams - Customize work tracking processes - Manage build retention policies - Add and manage build agent pools - Add and manage extensions	-Manage users, access, and permissions - Add and manage teams, team projects, and collections - Customize work tracking processes - Manage build retention policies - Add and manage build agent pools - Add and manage extensions - Configure an SMTP server (supports feedback requests and notification features) - Configure a backup schedule and manage database backups - Manage upgrades

Related notes

- Key concepts
- Client-server tools
- Software development roles
- Pricing

Sign up for a free VSTS organization and invite others to join your team

8/10/2018 • 4 minutes to read

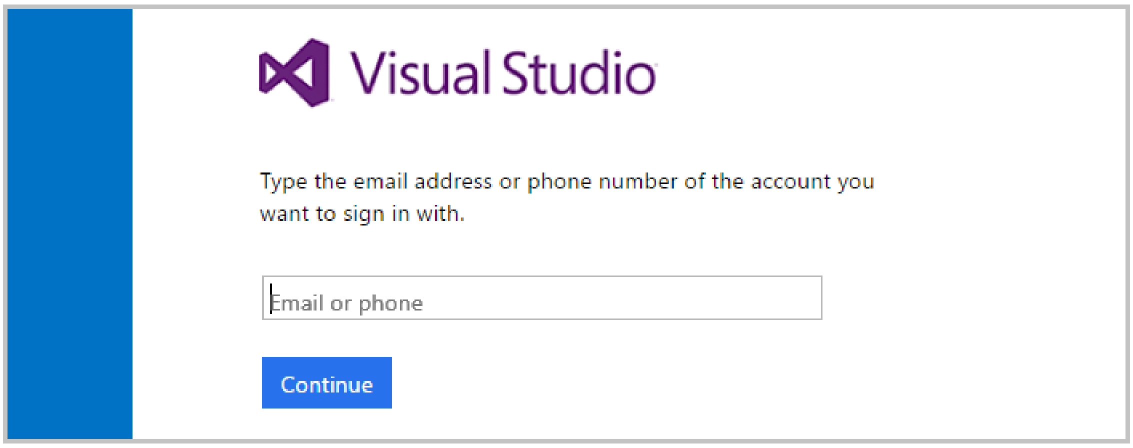
VSTS

Sign up for a VSTS organization to upload and share code in a free unlimited private Git repository. You can then connect to your favorite development tool like Eclipse, Xcode, Visual Studio, IntelliJ, or Android Studio to work on apps anytime, anywhere.

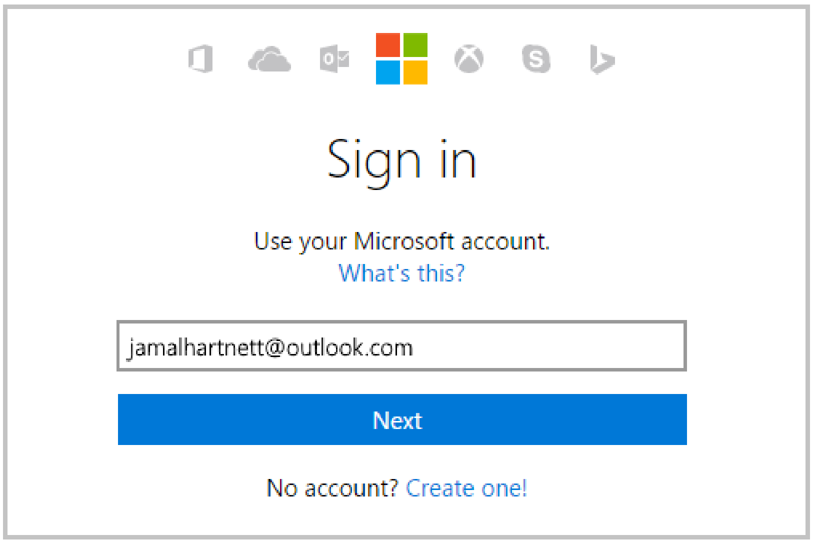
Sign up with a personal Microsoft account

1. Right-click the sign-up link, VSTS, and enter your email address for your Microsoft account.

   If you're a Visual Studio subscriber and get VSTS as a benefit, use the Microsoft account associated with your subscription.

2. On the Microsoft account sign-in page, enter your email address for your Microsoft account. Then enter your password, and finish signing in.

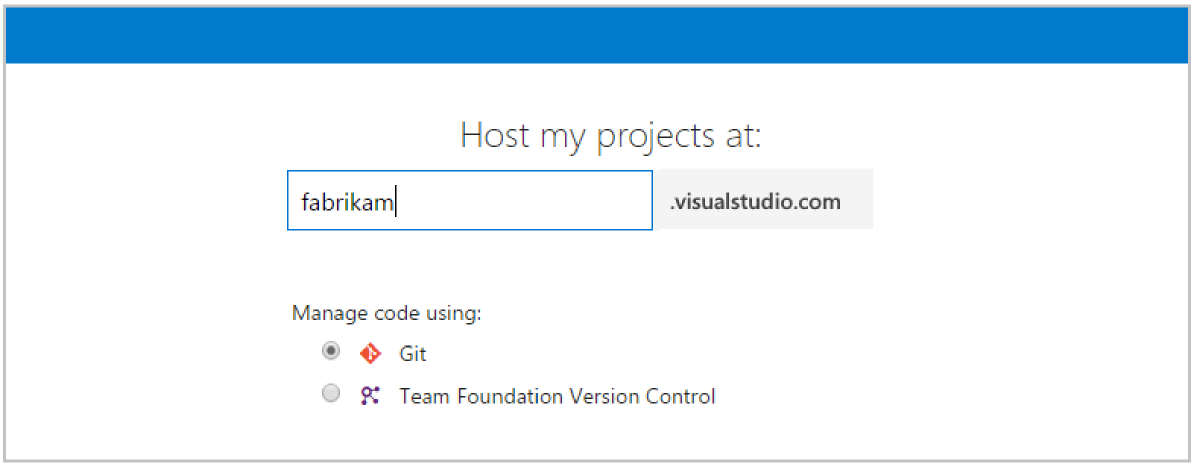
   

   If you don't have a Microsoft account, create one now.

3. Enter a name for your organization. The name you enter cannot contain spaces or special characters (such as / \ [ ] : | < > + = ; ? or *), cannot end in a period or comma, must be less than 256 characters, and must

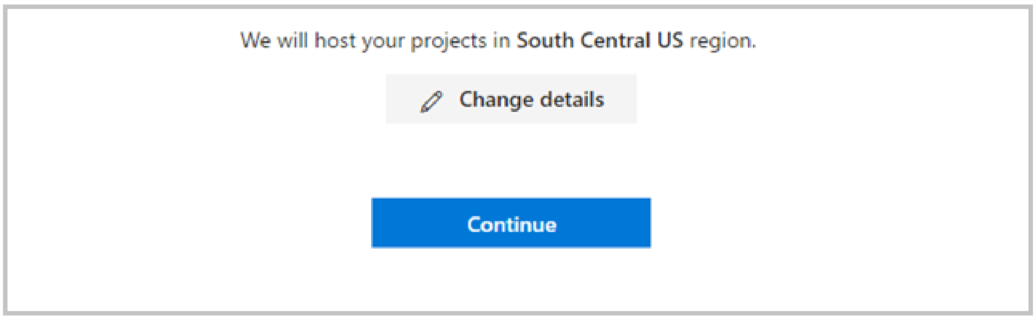
be unique within the VSTS namespace.

To manage your code, choose **Git** or **Team Foundation Version Control**.



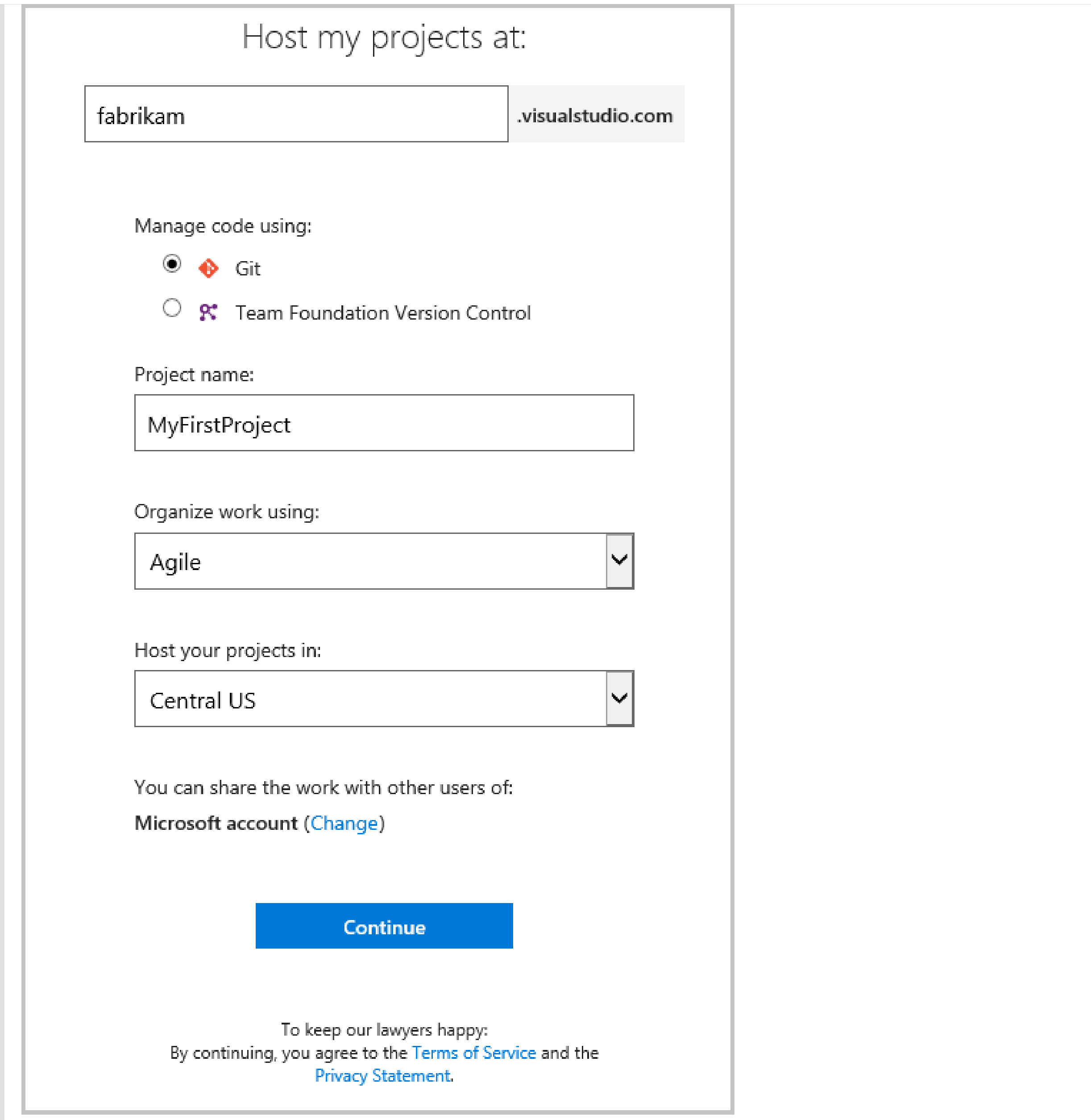
Choose Git for distributed source control and TFVC for centralized source control. To learn more, review Git or Team Foundation Version Control.

4. (Optional) Choose **Change details** to change default assignments. Otherwise, choose **Continue** to confirm to initiate your organization creation.

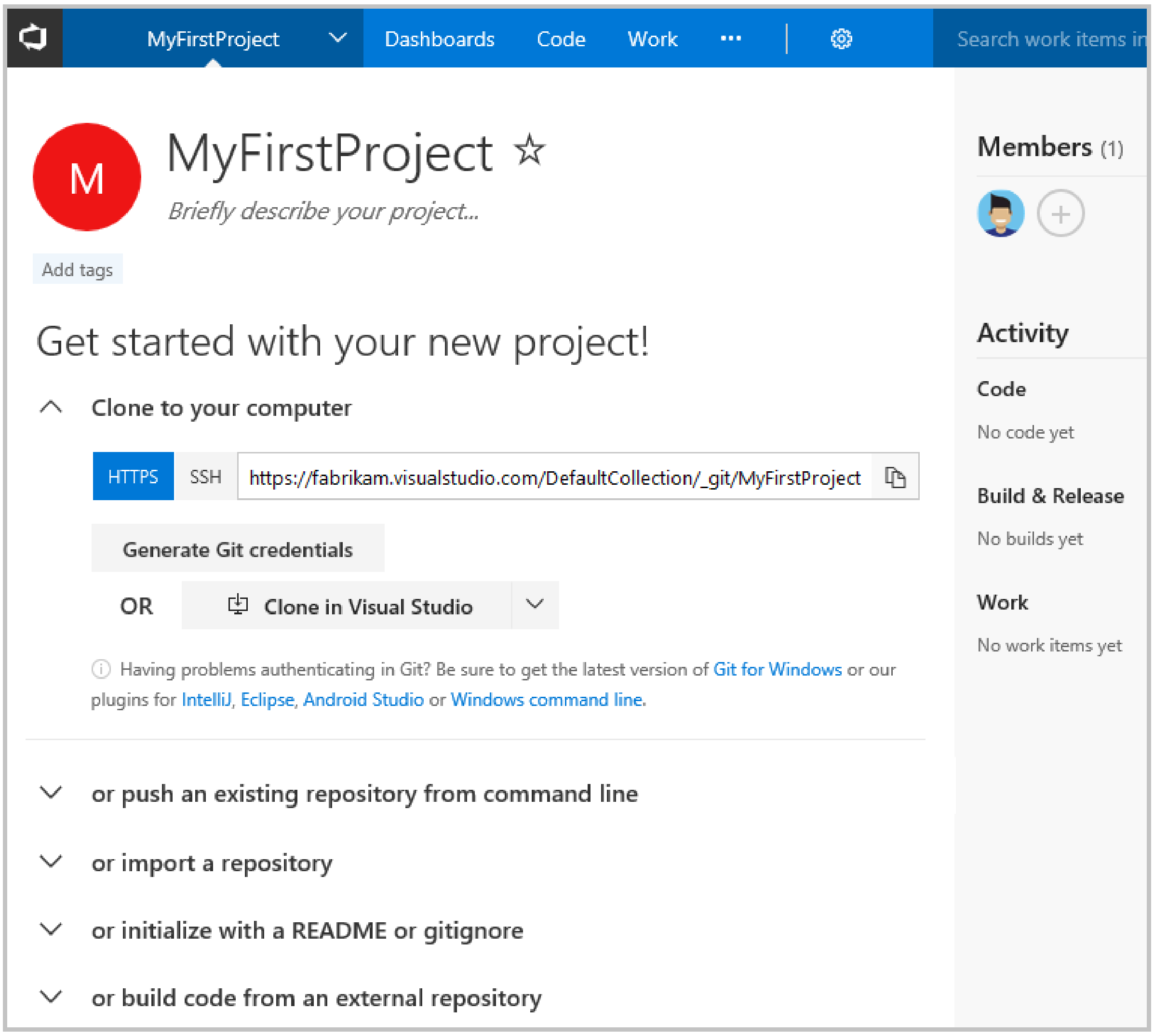


To change default assignments, use the following guidance:

- **Project name**: You can specify your team project name. The name you enter cannot contain spaces or special characters (such as / : \ ~ & % ; @ ' " ? < > | # $ * } { , + = [ ], cannot begin with an underscore or begin or end with a period, and must be 64 characters or less.
- **Organize work using**: The default work tracking process is set to **Agile**. If you want to work with Scrum or CMMI processes, then select those now. You can't change the process after the team project is created. (To learn more, see Choose process.
- **Host your projects in**: You can choose between several locations for where you want your data hosted.



5. By default, VSTS creates your first team project as "MyFirstProject". After VSTS creates your organization and team project, you can invite others to join your project, add code, start planning and tracking using work items, or rename your team project.



Congratulations, you're now a VSTS organization owner!

To sign in to your organization at any time, go to `https://{yourorganization}.visualstudio.com`.

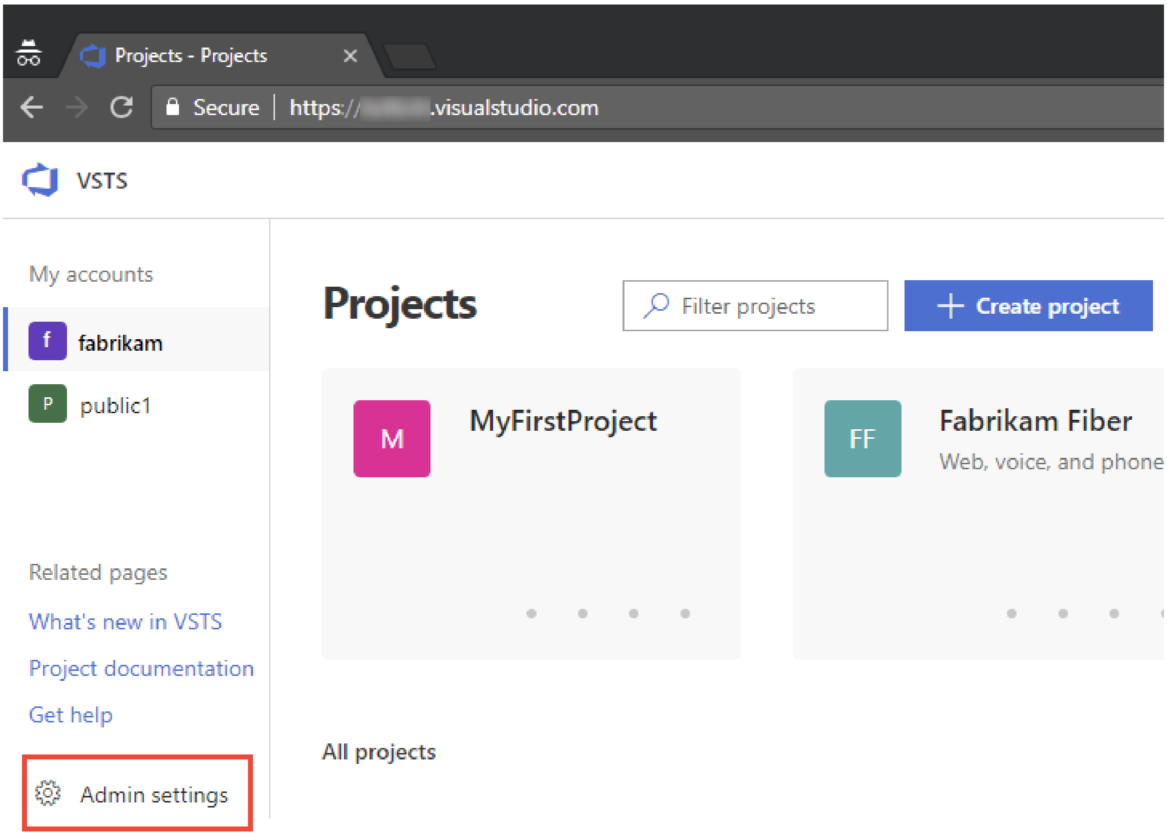
Invite others to join your team

> **NOTE**
>
> The **New navigation** feature, which provides a vertical navigation experience, is in preview for VSTS. When you enable new navigation, you automatically enable several new agile tool features that are described in the New Work Hubs blog post.
>
> On-premises TFS users can choose **Previous navigation** for guidance.

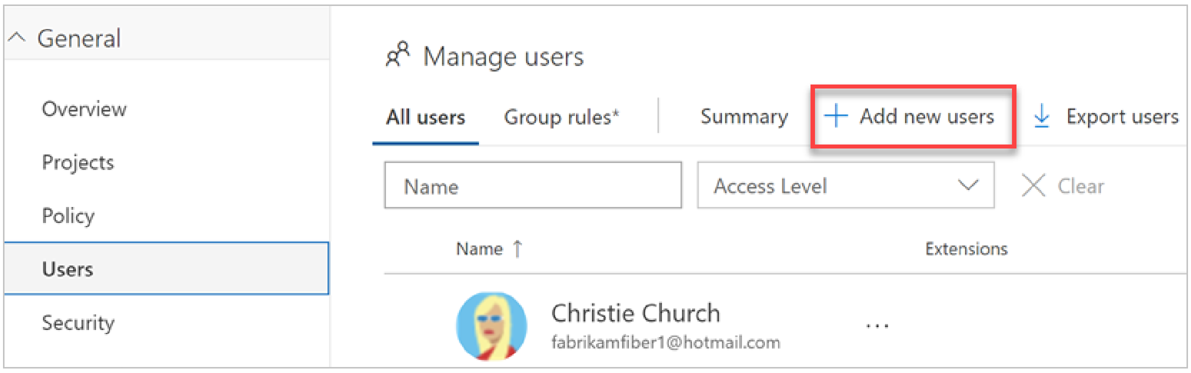
- New navigation
- Previous navigation

Provide others access to your organization by adding their email address.

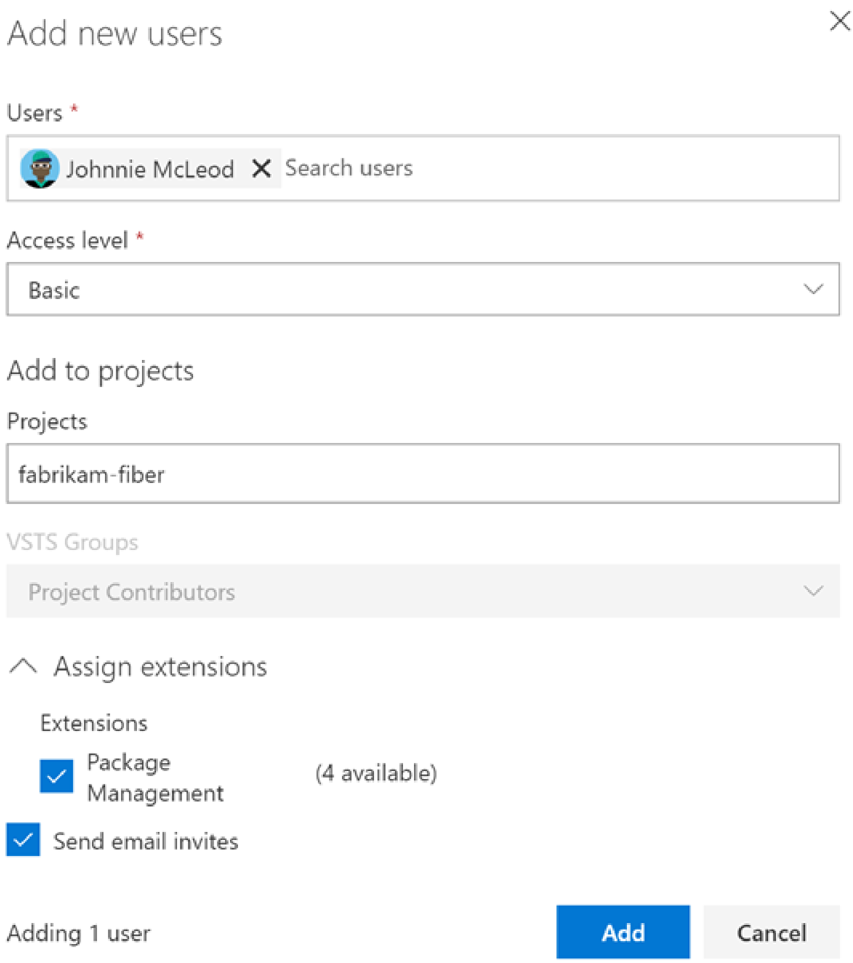
1. Sign in to your VSTS organization (`https://{yourorganization}.visualstudio.com`).
2. Select ⚙ **Admin settings**.



3. Select **Users** > **Add new users** to open the form.



4. Complete the form.



- **Users**: Enter the Microsoft account (MSA) email address for the user organization. You can add several email addresses by separating them with a semicolon (;). Note that in MSAs, the email addresses display in red.
- **Access level**: Leave the Access level at **Basic** for those users who will contribute to the code base. To learn more, see About access levels.
- **Add to projects**: Select the project that you named in the previous procedure.
- **VSTS Groups**: Leave this entry at Project Contributors, the default security group for people who will contribute to your project. To learn more, see Default permissions and access assignments.

5. Select **Add** to complete your invitation.

Code with Git

8/10/2018 • 3 minutes to read

VSTS

After you create a new organization and team project in VSTS, you can begin sharing your code with others.

To work with a Git repo, you clone it to your computer. Cloning a repo creates a complete local copy of the repo for you to work with, and downloads all commits and branches in the repo and sets up a named relationship with the repo on the server. Use this relationship to interact with the existing repo, pushing and pulling changes to share code with your team.

Install Git command line tools

1. Install one of the following Git command line tools:

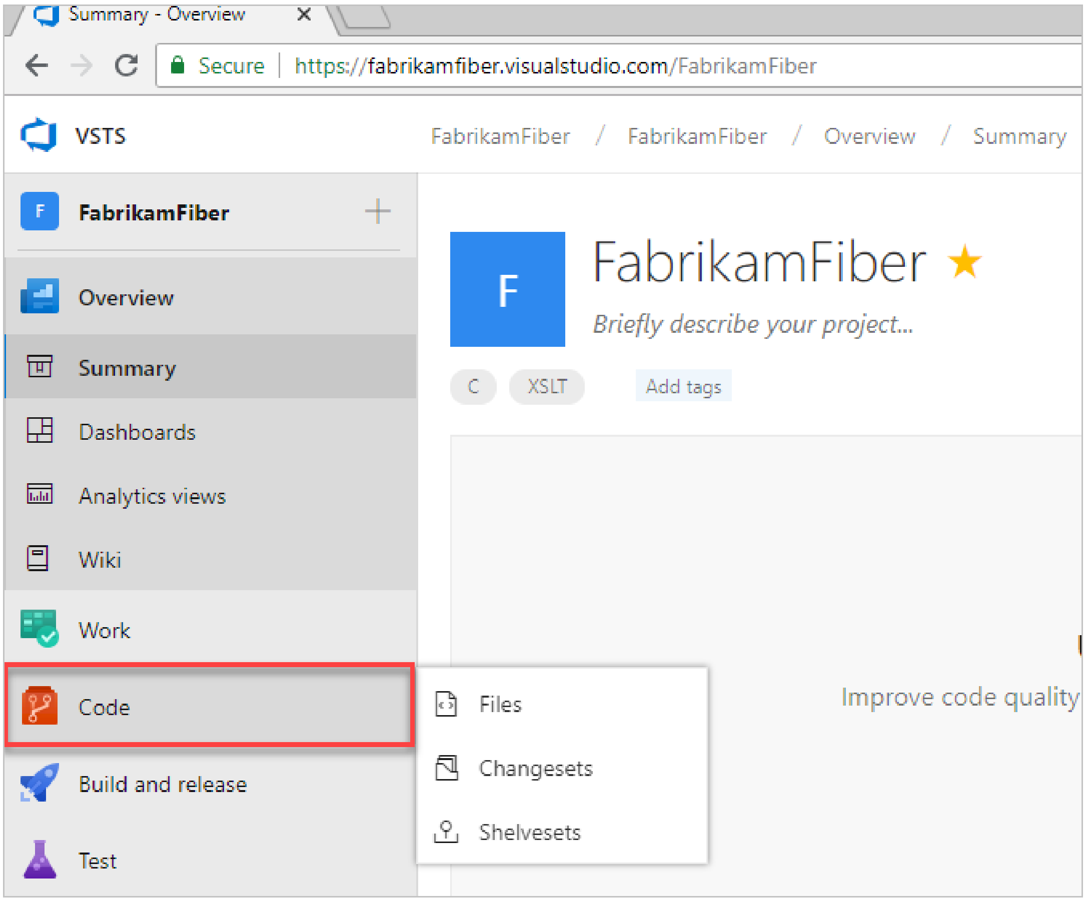
> - To install Git for Windows, including Git Credential Manager, see Install the Git Credential Manager - Windows
> - To install Git for macOS and Linux, see Install the Git Credential Manager - macOS and Linix

Clone the repo to your computer

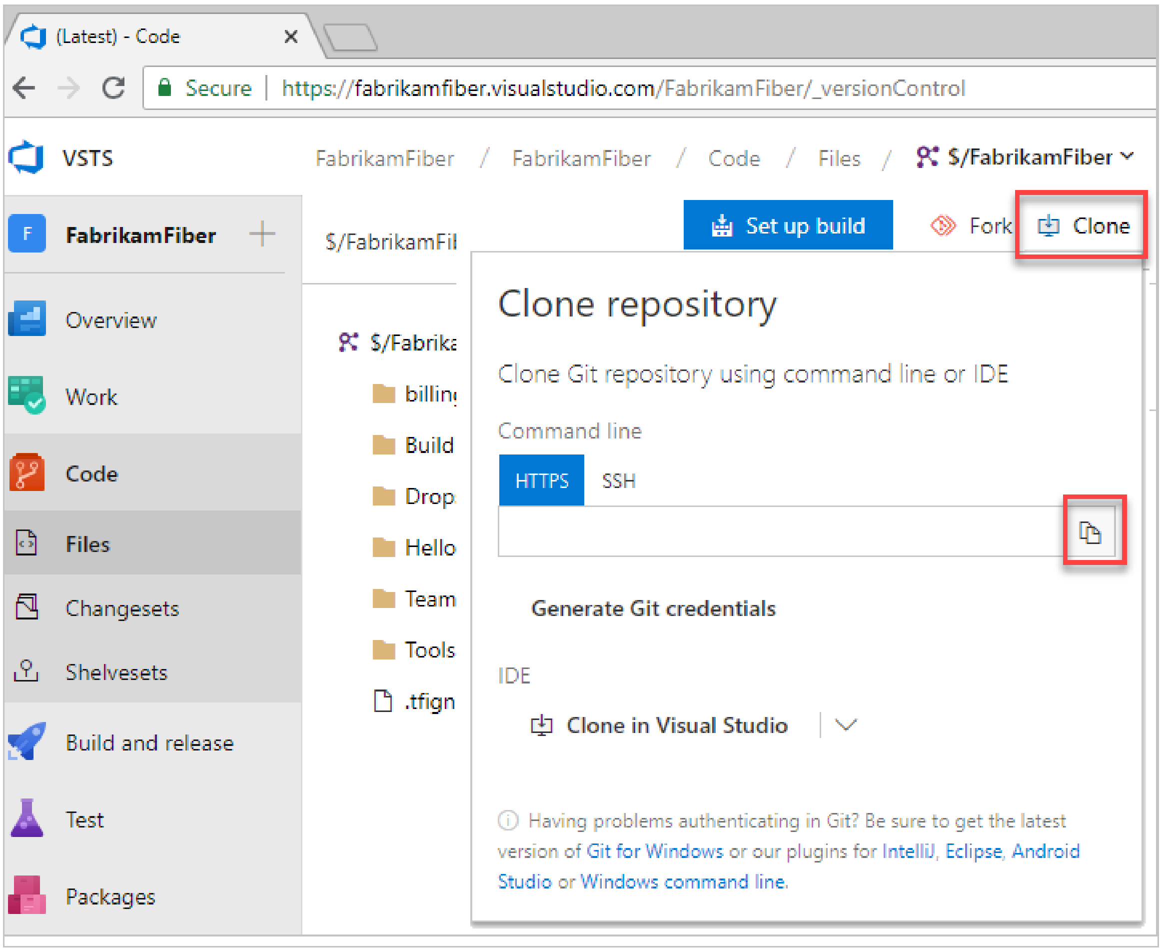
> **NOTE**
>
> The **New navigation** feature, which provides a vertical navigation experience, is in preview for VSTS. When you enable new navigation, you automatically enable several new agile tool features that are described in the New Work Hubs blog post.
>
> On-premises TFS users can choose **Previous navigation** for guidance.

- New navigation
- Previous navigation

1. From your web browser, open the project for your VSTS organization and select **Code**. If you don't have a team project, create one now.



2. Select **Clone** in the upper-right corner of the Code window and **copy** the URL.



1. Open the Git command window (Git Bash on Git for Windows), navigate to the folder where you want the code from the repo stored on your computer, and run `git clone` followed by the path copied from the **Clone URL** in the previous step, as shown in the following example.

> ```
> git clone >https://contoso-ltd.visualstudio.com/MyFirstProject/_git/contoso->demo
> ```
>
> Git downloads a copy of the code, including all commits and branches from the repo, into a new folder for you to work with.

Keep this command window open, as you'll use it in the following steps.

Work with the code

In this step, we'll make a change to the files on your computer, commit the changes locally, push the commit up to the repo that is stored on the server, and view the changes there.

1. Browse to the folder on your computer where you cloned the repo, open the `README.md` file in your editor of choice, make some changes, and save and close the file.

2. In the Git command window, navigate to the `contoso-demo` directory by entering the following command:

> ```
> cd contoso-demo
> ```

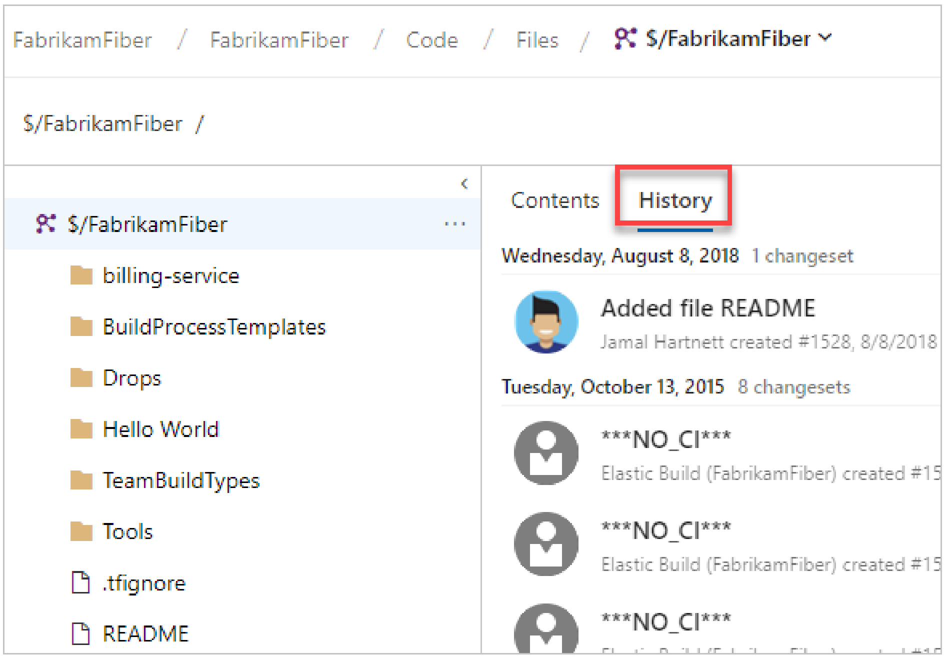
3. Commit your changes by entering the following command in the Git command window:

> ```
> git commit -a -m "My first commit"
> ```
>
> When you're using `git commit`, `-a` means to commit all changed files, and `-m` specifies a commit message.

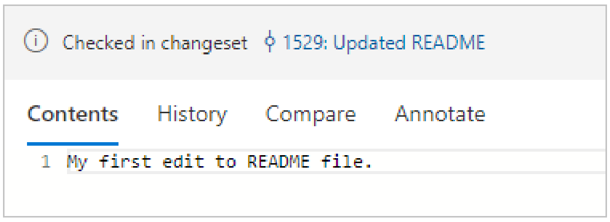
4. Push your changes up to the Git repo on the server by entering the following command into the Git command window:

> ```
> git push
> ```
>
> # New navigation(#tab/new-nav)

5. Switch back to the web portal and select **History** from the Code page to view your new commit.

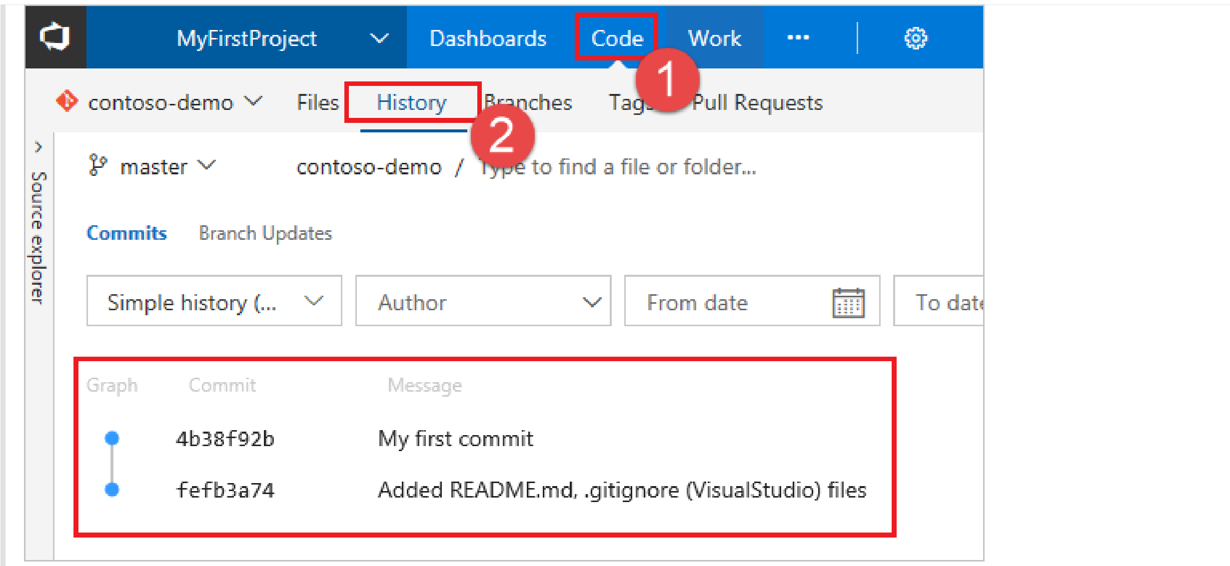


6. Switch to the **Files** tab and select the README file to view your changes.

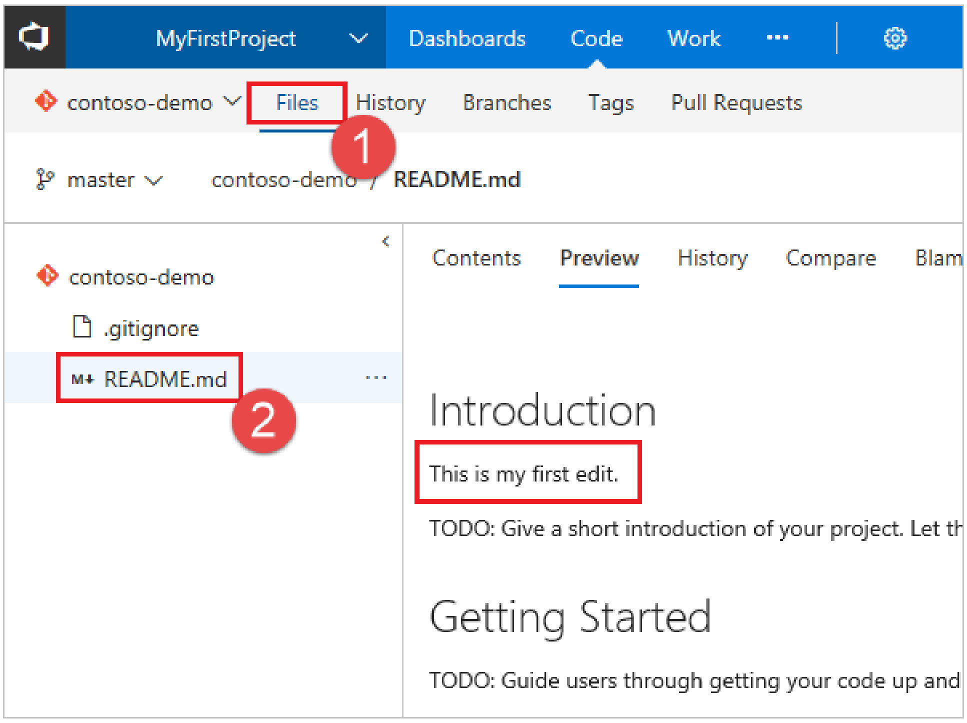


Previous navigation(#tab/prev-nav)

1. Switch back to the web portal and select **History** from the **Code** tab to view your new commit. The new repo has two commits: the first commit where the README and .gitignore were added when the repo was created, and the commit you just made.



2. Switch to the **Files** tab and select the README file to view your changes.



>

Try this next

Set up continuous integration & delivery or Plan & track work

Or, learn more about working with a Git repo

Create your first build and release

8/7/2018 • 14 minutes to read

NOTE

Build and release *pipelines* are called *definitions* in TFS 2018 and in older versions. *Service connections* are called *service endpoints* in TFS 2018 and in older versions.

NOTE

This guidance applies to TFS version 2017.3 and newer.

We'll show you how to use VSTS pipelines to create a build that prints "Hello world" and then automatically creates a release that does the same. By the time you finish here, you'll see an end-to-end process run every time you push new code into your project. You'll also become familiar with all the basic concepts of VSTS pipelines.

We'll show you how to use TFS to create a build that prints "Hello world" and then automatically creates a release that does the same. By the time you finish here, you'll see an end-to-end process run every time you push new code into your project. You'll also become familiar with all the basic concepts of build and release in TFS.

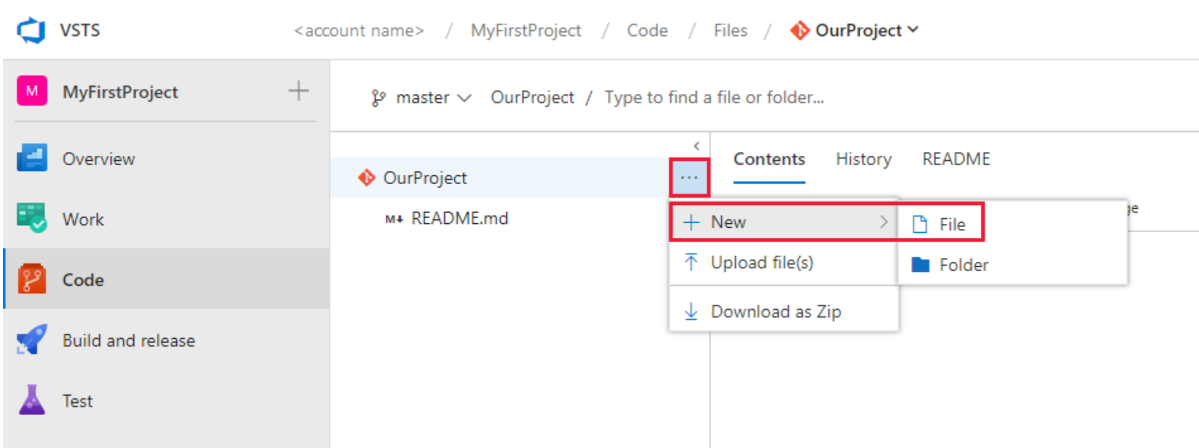
Prerequisites

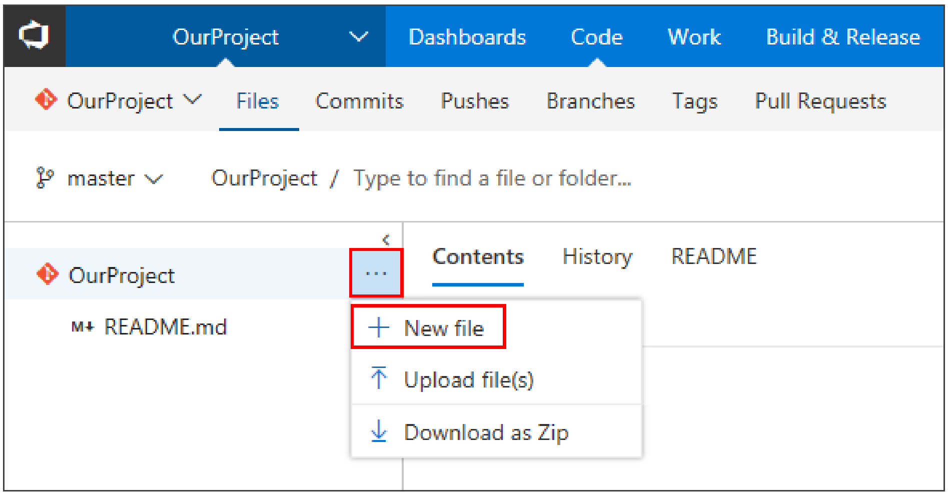
- A VSTS organization. If you don't have one, you can create one for free. If your team already has one, then make sure you are an administrator of the project you want to use.
- A self-hosted Windows agent.

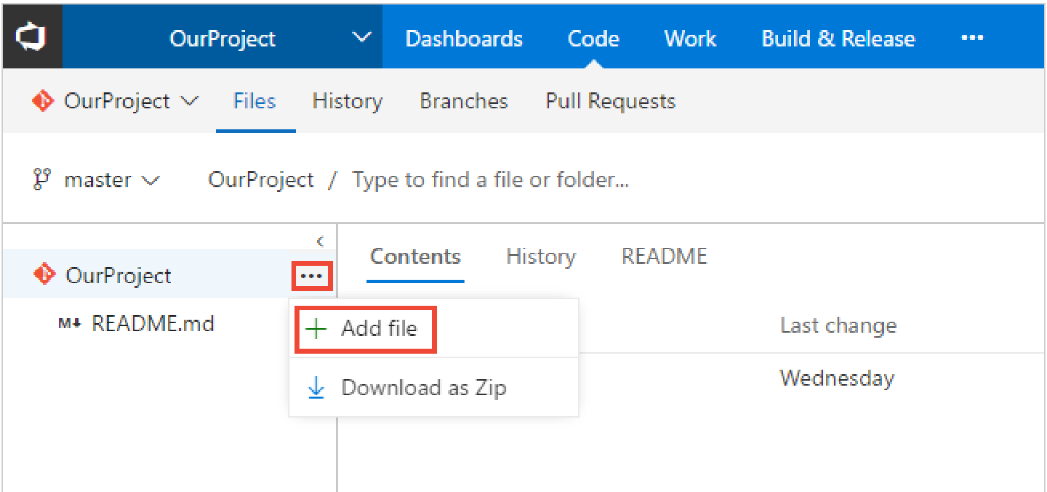
Add a script to your repository

Create a PowerShell script that prints `Hello world`.

1. Go to the **Code** hub.
2. Add a file.
   - New navigation
   - Previous navigation







3. In the dialog box, name your new file and create it.

```
HelloWorld.ps1
```

4. Copy and paste this script.

```
Write-Host "Hello world"
```

5. **Commit** (save) the file.

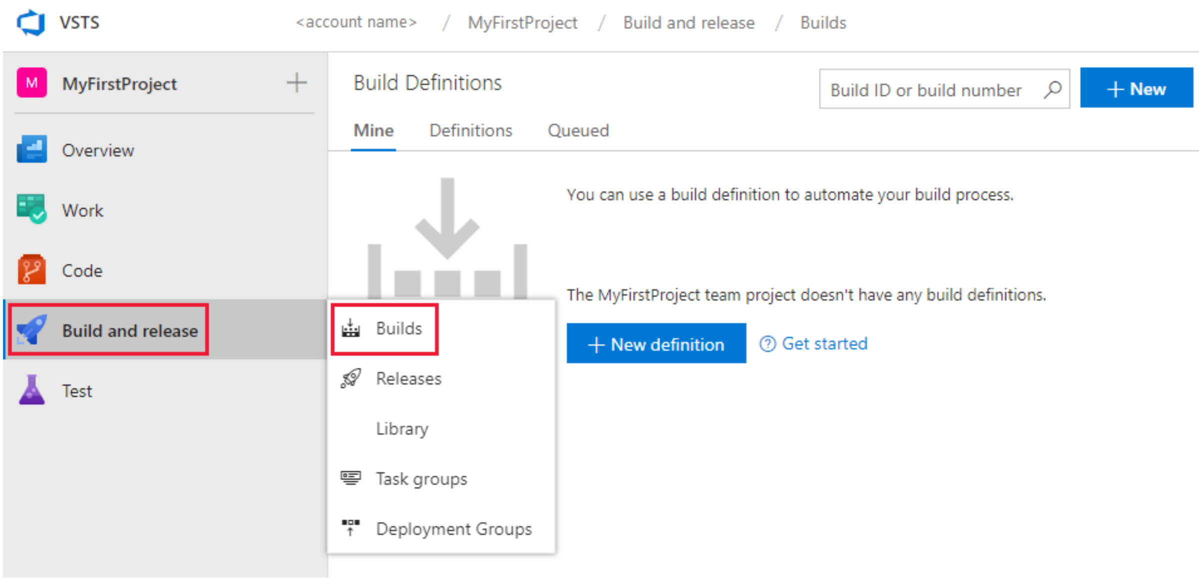
> In this tutorial, our focus is on CI/CD, so we're keeping the code part simple. We're working in a VSTS Git repository directly in your web browser.
>
> When you're ready to begin building and deploying a real app, you can use a wide range of version control clients and services with VSTS CI builds. Learn more.

Create a build pipeline

Create a build pipeline that prints "Hello world."

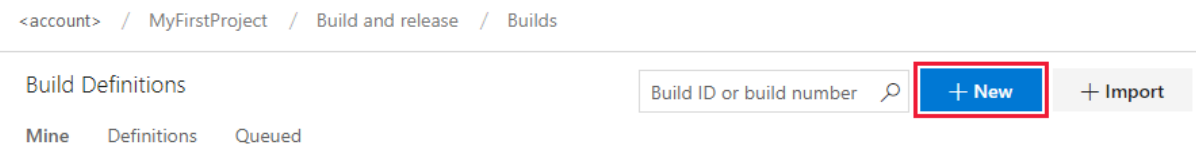
1. Select the **Build and Release** hub in your VSTS project, and then choose **Builds**.

- New navigation
- Previous navigation



2. Create a new pipeline.

- New navigation
- Previous navigation

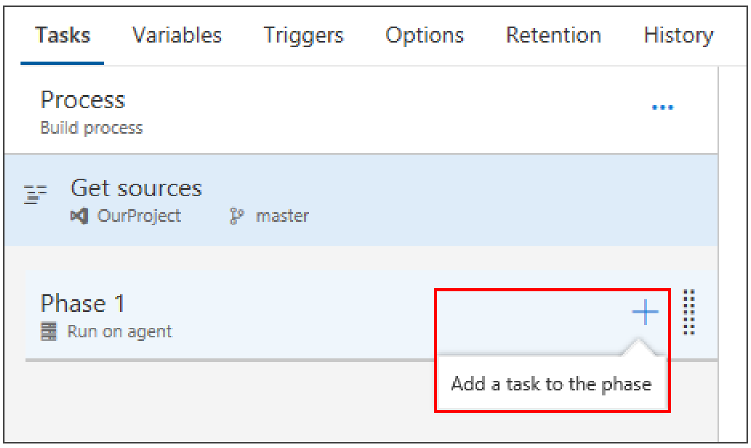


3. Make sure that the **source**, **project**, **repository**, and default **branch** match the location in which you created the script.

4. Start with an **empty process**.

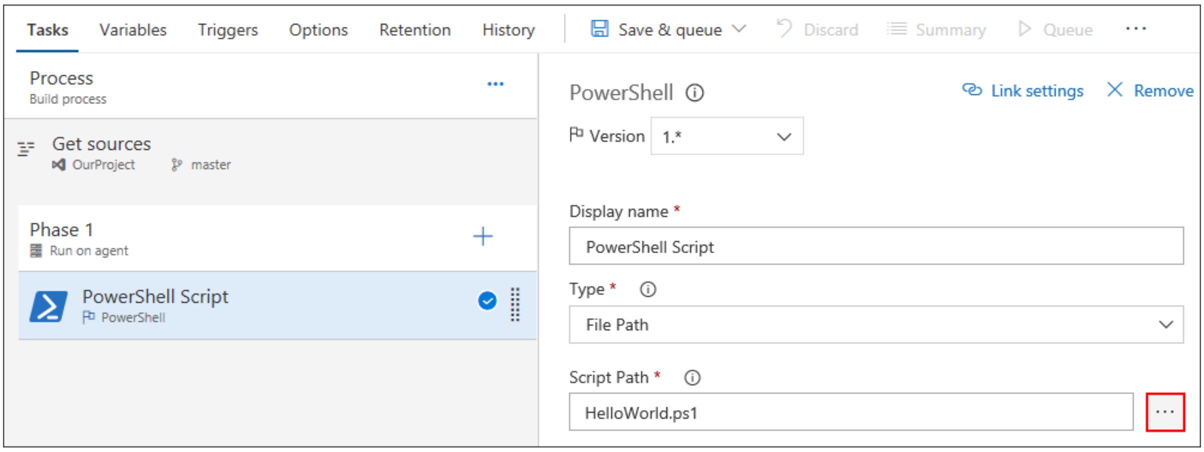
5. On the left side, select **Process** and specify whatever **Name** you want to use. For the **Agent queue**, select **Hosted VS2017**.

6. On the left side, select the plus sign **(+)** to add a task to **Phase 1**. On the right side, select the **Utility** category, select the **PowerShell** task from the list, and then choose **Add**.



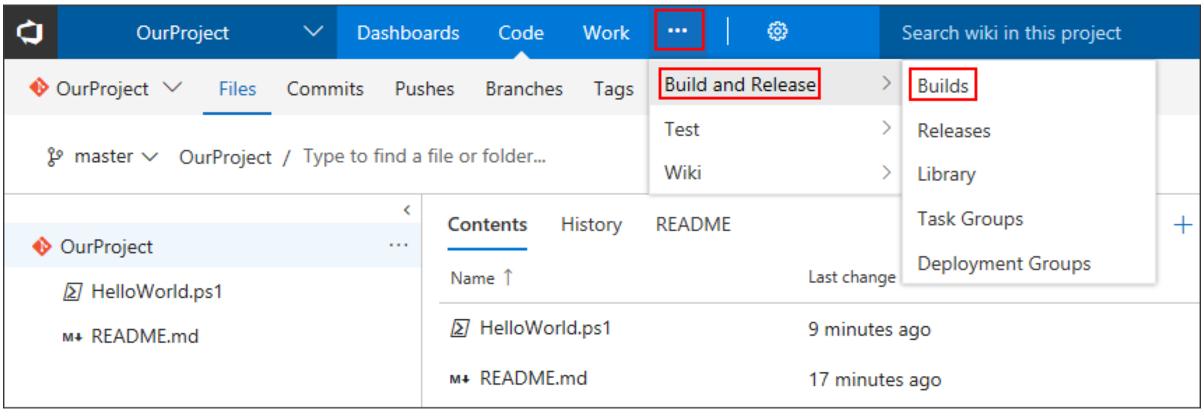
7. On the left side, select your new **PowerShell** script task.

8. For the **Script Path** argument, select the **...** button to browse your repository and select the script you created.

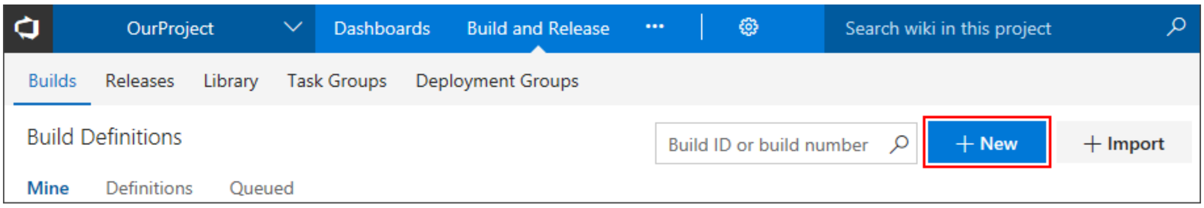


9. Select **Save & queue**, and then select **Save**.

1. Select the **Build and Release** hub in your VSTS project, and then choose **Builds**.



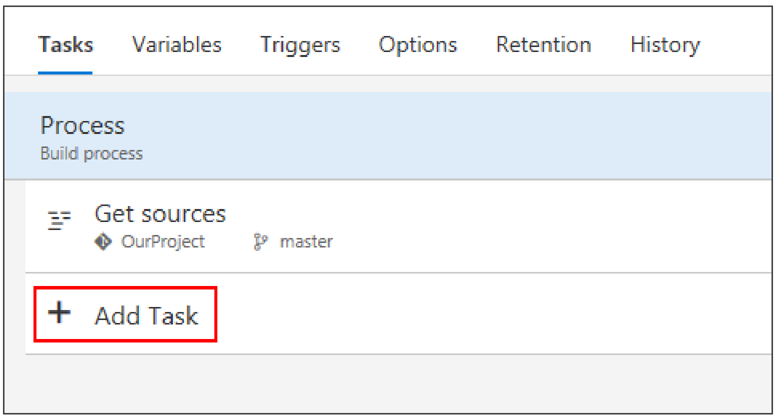
2. Create a new pipeline.



3. Start with an **empty process**

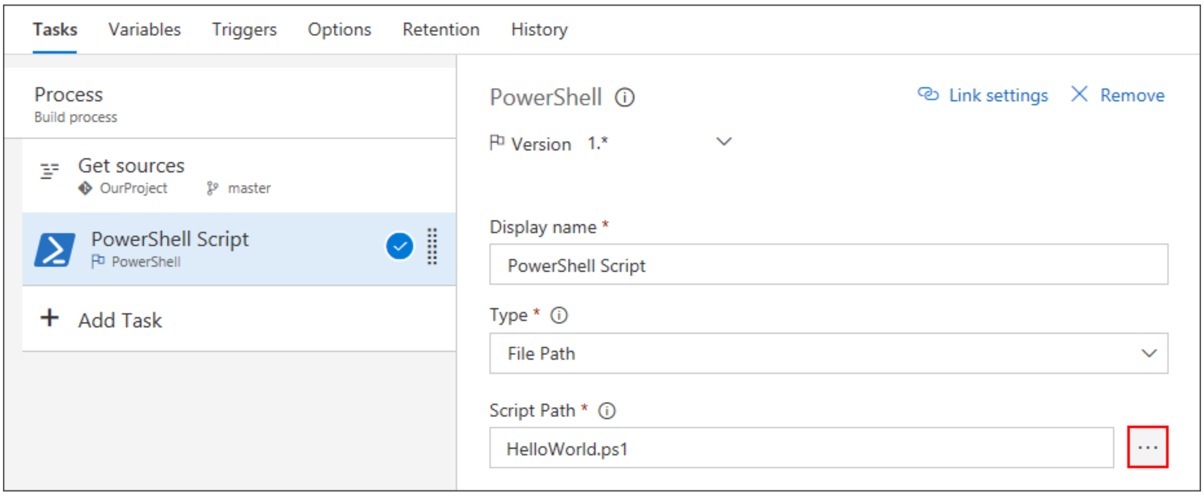
4. Select **Process** and specify whatever **Name** you want to use. For the **Agent queue**, select **Default**.

5. On the left side, select **+ Add Task** to add a task to the phase, and then on the right side select the **Utility** category, select the **PowerShell** task, and then choose **Add**.



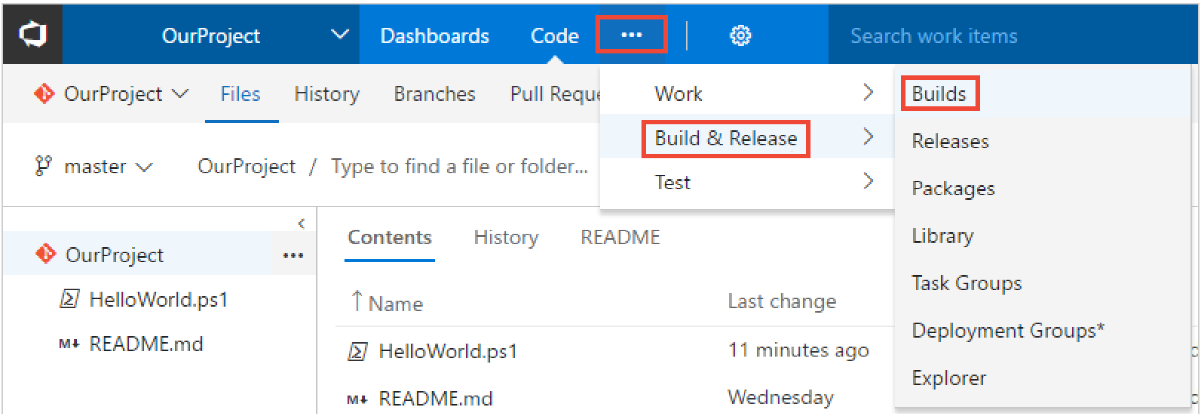
6. On the left side, select your new **PowerShell** script task.

7. For the **Script Path** argument, select the **...** button to browse your repository and select the script you created.

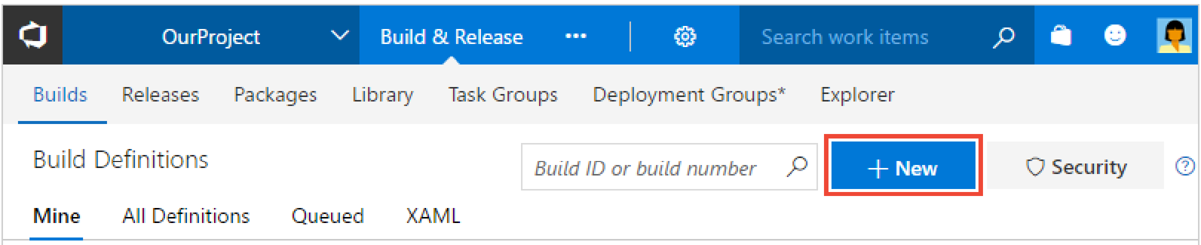


8. Select **Save & queue**, and then select **Save**.

1. Select the **Build and Release** hub in your VSTS project, and then the **Builds** tab.



2. Create a new pipeline.



3. Start with an **empty process**.

4. Select **Process** and specify whatever **Name** you want to use.

5. On the **Options** tab, select **Default** for the **Agent queue**, or select whichever queue you want to use that has Windows build agents.

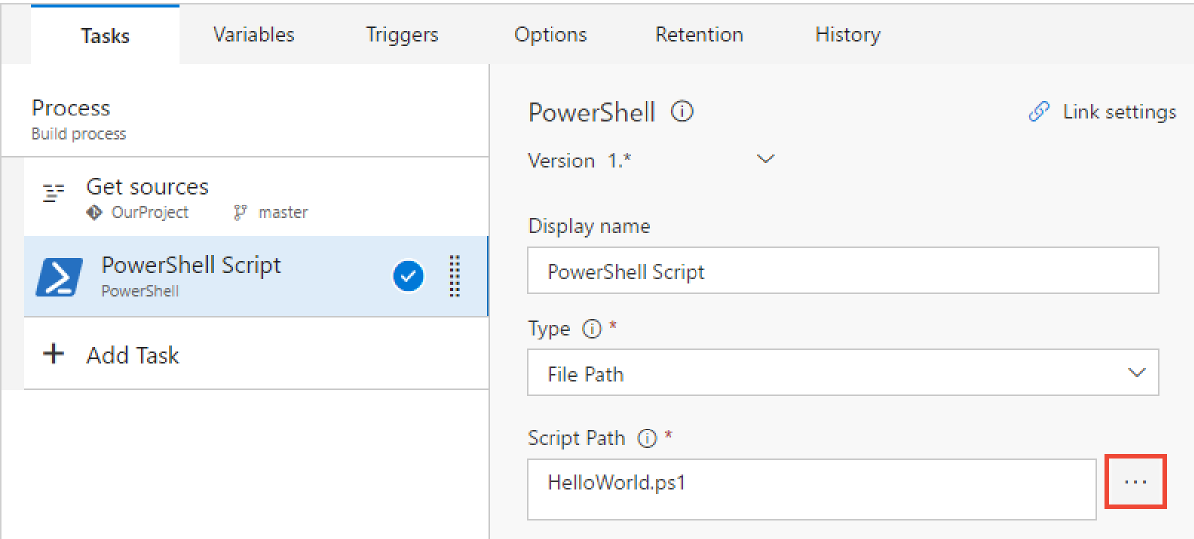
6. On the **Tasks** tab, make sure that **Get sources** is set with the **Repository** and **Branch** in which you created the script.

7. On the left side select **Add Task**, and then on the right side select the **Utility** category, select the **PowerShell** task, and then select **Add**.

8. On the left side, select your new **PowerShell** script task.

9. For the **Script Path** argument, select the **...** button to browse your repository and select the script you

created.



10. Select **Save & queue**, and then select **Save**.

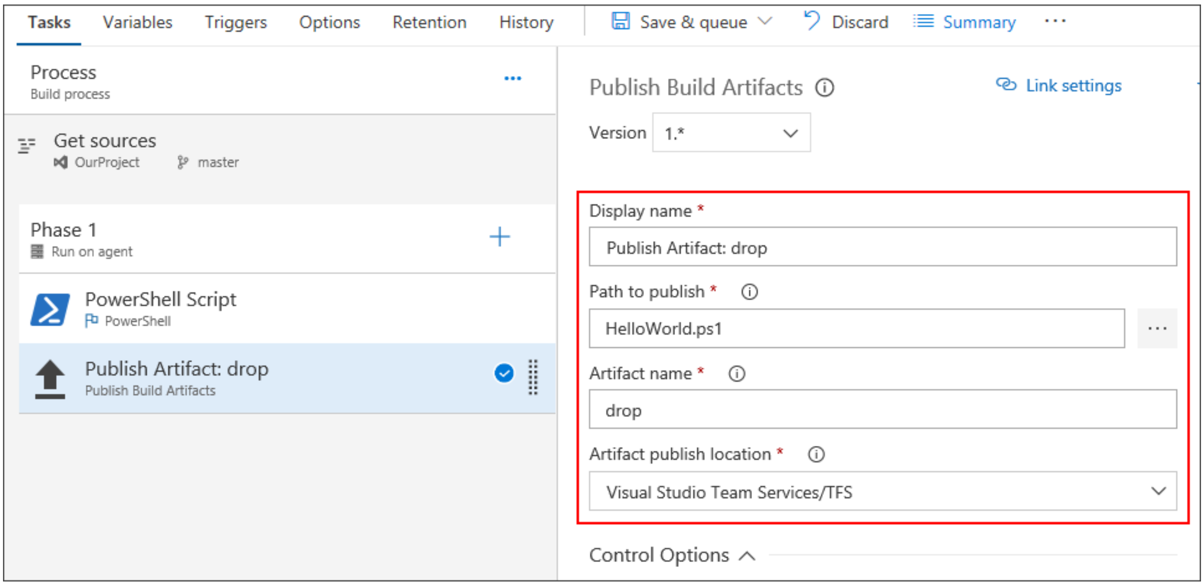
> A build pipeline is the entity through which you define your automated build process. In the build pipeline, you compose a set of tasks, each of which perform a step in your build. The task catalog provides a rich set of tasks for you to get started. You can also add PowerShell or shell scripts to your build pipeline.

Publish an artifact from your build

A typical build produces an artifact that can then be deployed to various environments in a release. Here to demonstrate the capability in a simple way, we'll simply publish the script as the artifact.

1. On the **Tasks** tab, select the plus sign **(+)** to add a task to **Phase 1**.

2. Select the **Utility** category, select the **Publish Build Artifacts** task, and then select **Add**.



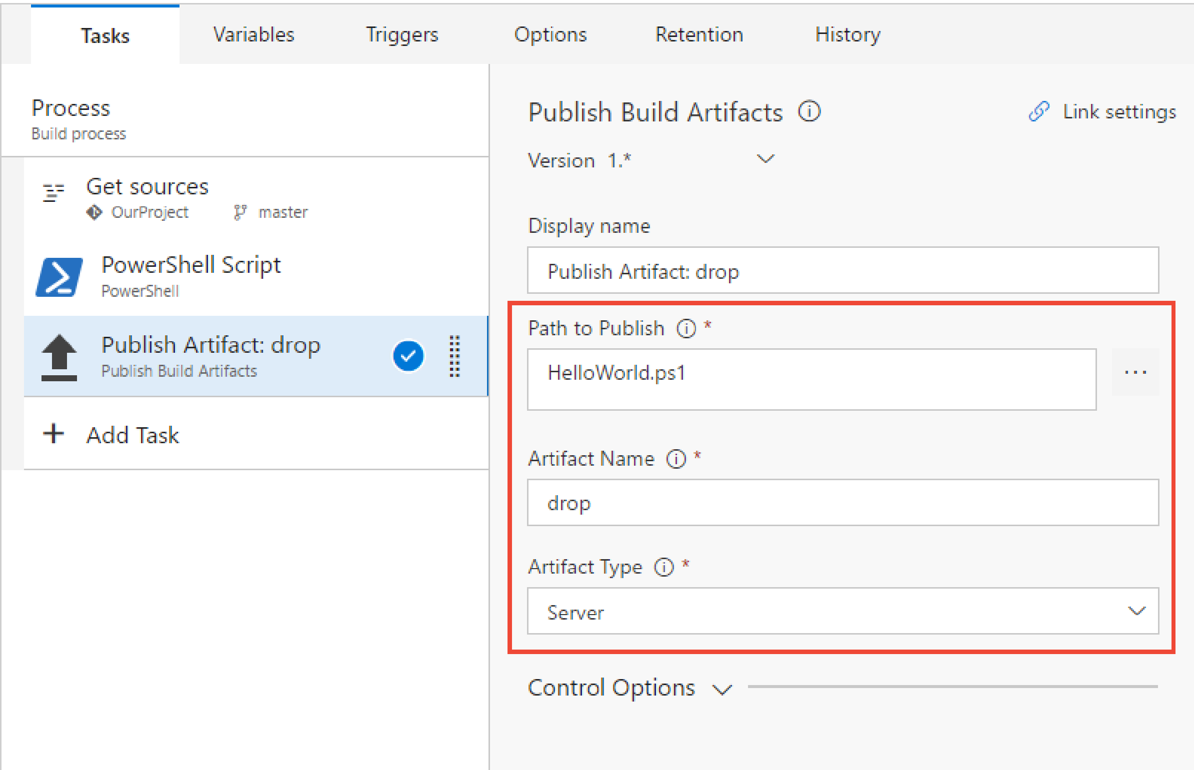
Path to publish: Select the `...` button to browse and select the script you created.

Artifact name: Enter `drop`.

Artifact publish location: Select **Visual Studio Team Services/TFS**.

1. On the **Tasks** tab, select **Add Task**.

2. Select the **Utility** category, select the **Publish Build Artifacts** task, and then select **Add**.



Path to Publish: Select the `...` button to browse and select the script you created.

Artifact Name: Enter `drop`.

Artifact Type: Select **Server**.

> Artifacts are the files that you want your build to produce. Artifacts can be nearly anything your team needs to test or deploy your app. For example, you've got a .DLL and .EXE executable files and .PDB symbols file of a C# or C++ .NET Windows app.
>
> To enable you to produce artifacts, we provide tools such as copying with pattern matching, and a staging directory in which you can gather your artifacts before publishing them. See Artifacts in Team Build.

Enable continuous integration (CI)

1. Select the **Triggers** tab.
2. Enable **Continuous integration**.

> A continuous integration trigger on a build pipeline indicates that the system should automatically queue a new build whenever a code change is committed. You can make the trigger more general or more specific, and also schedule your build (for example, on a nightly basis). See Build triggers.

Save and queue the build

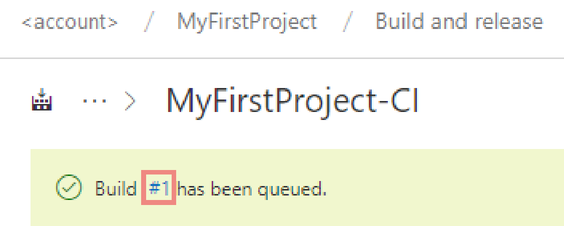
Save and queue a build manually and test your build pipeline.

1. Select **Save & queue**, and then select **Save & queue**.
2. On the dialog box, select **Save & queue** once more.

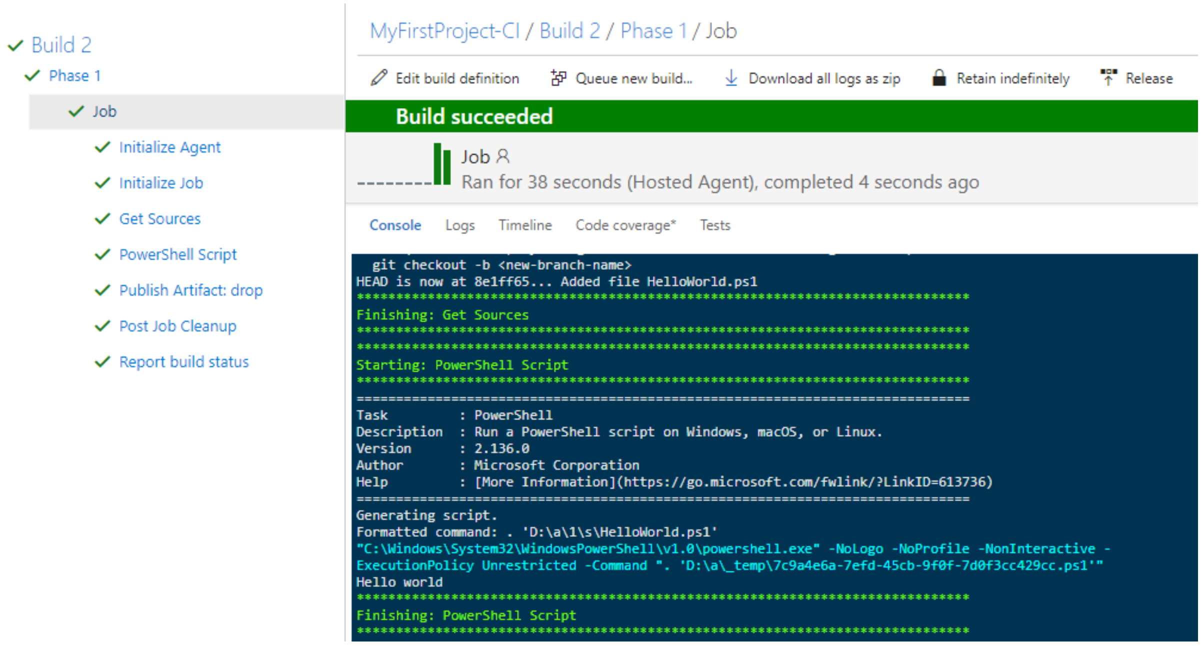
   This queues a new build on the Microsoft-hosted agent.
3. You see a link to the new build on the top of the page.

   - New navigation

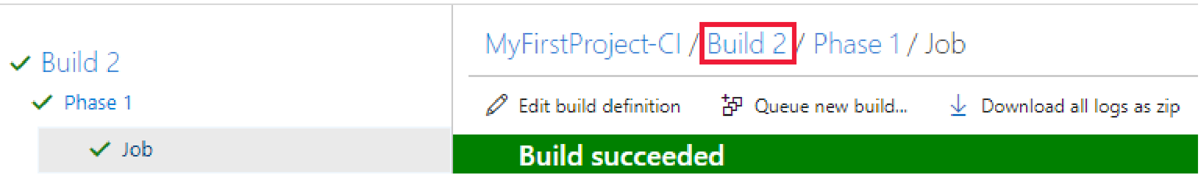
- Previous navigation



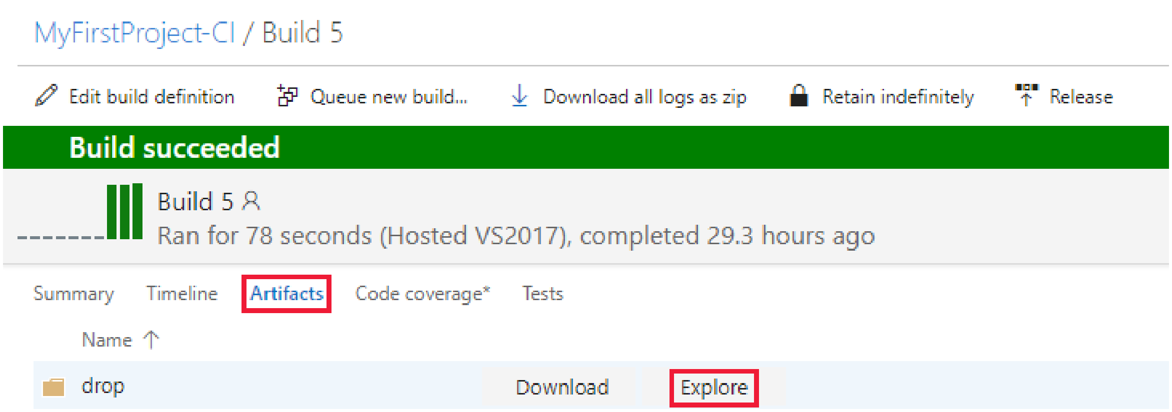
Choose the link to watch the new build as it happens. Once the agent is allocated, you'll start seeing the live logs of the build. Notice that the PowerShell script is run as part of the build, and that "Hello world" is printed to the console.



4. Go to the build summary.



5. On the **Artifacts** tab of the build, notice that the script is published as an artifact.

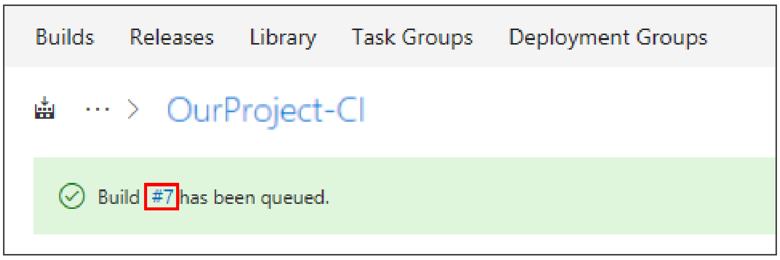


1. Select **Save & queue**, and then select **Save & queue**.

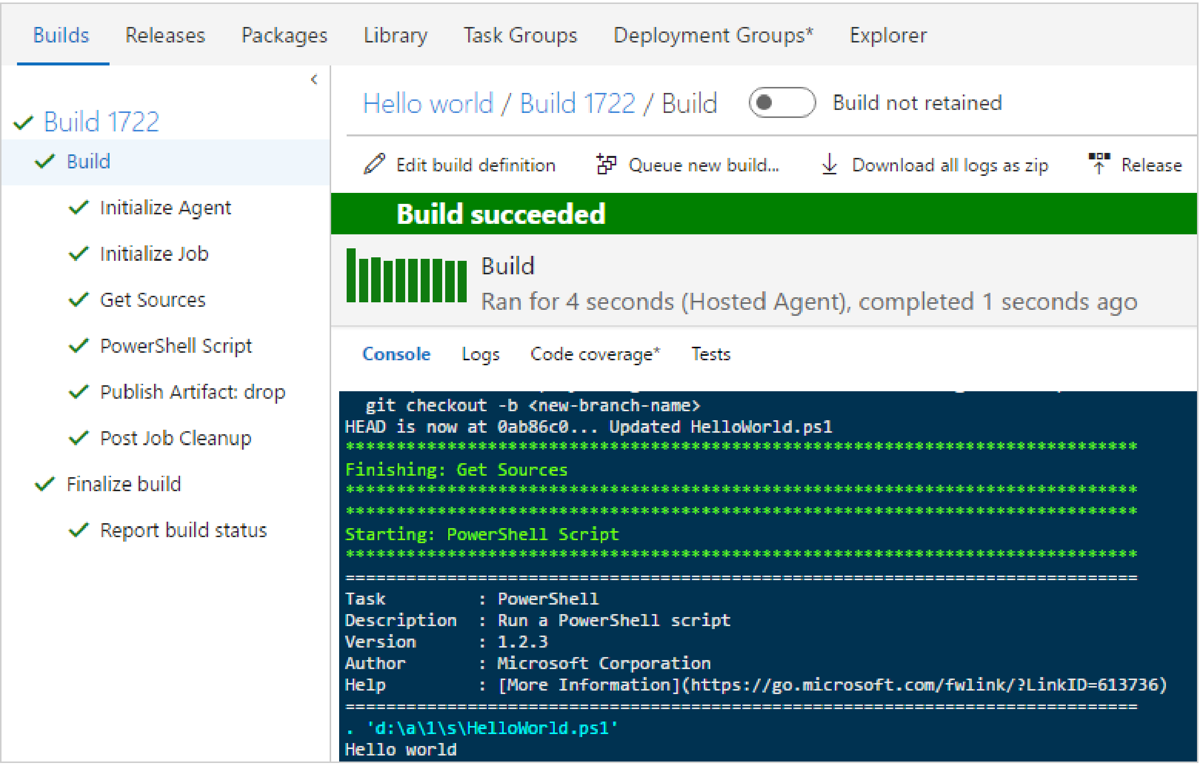
2. On the dialog box, select **Save & queue** once more.

This queues a new build on the Microsoft-hosted agent.

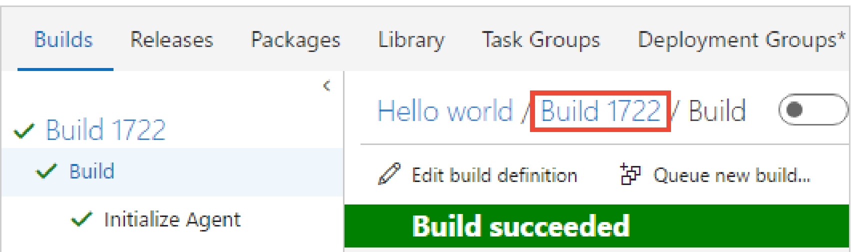
3. You see a link to the new build on the top of the page.



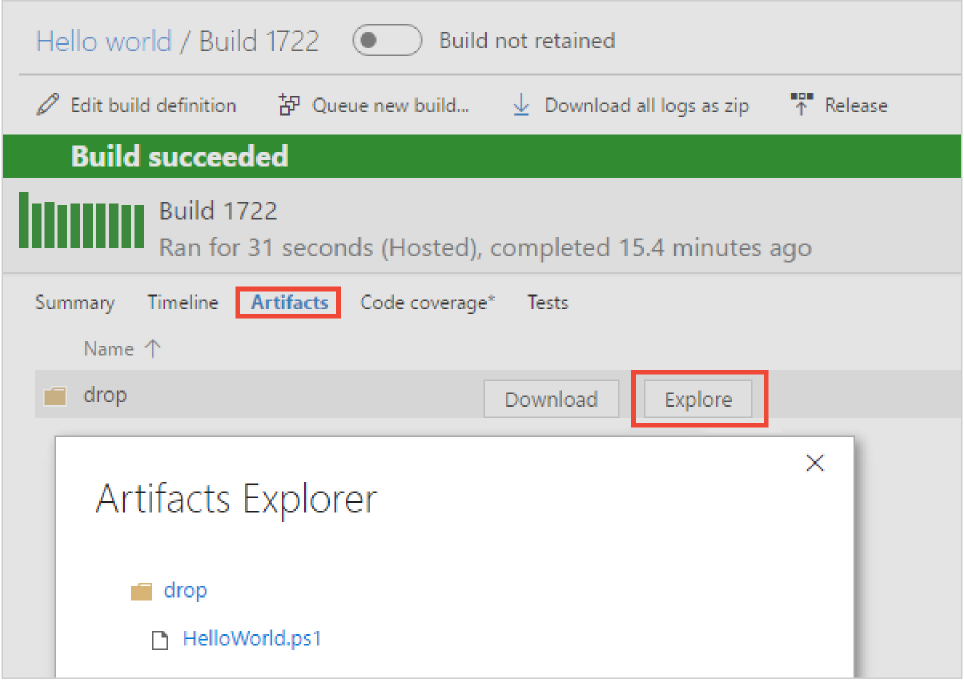
Choose the link to watch the new build as it happens. Once the agent is allocated, you'll start seeing the live logs of the build. Notice that the PowerShell script is run as part of the build, and that "Hello world" is printed to the console.



4. Go to the build summary.



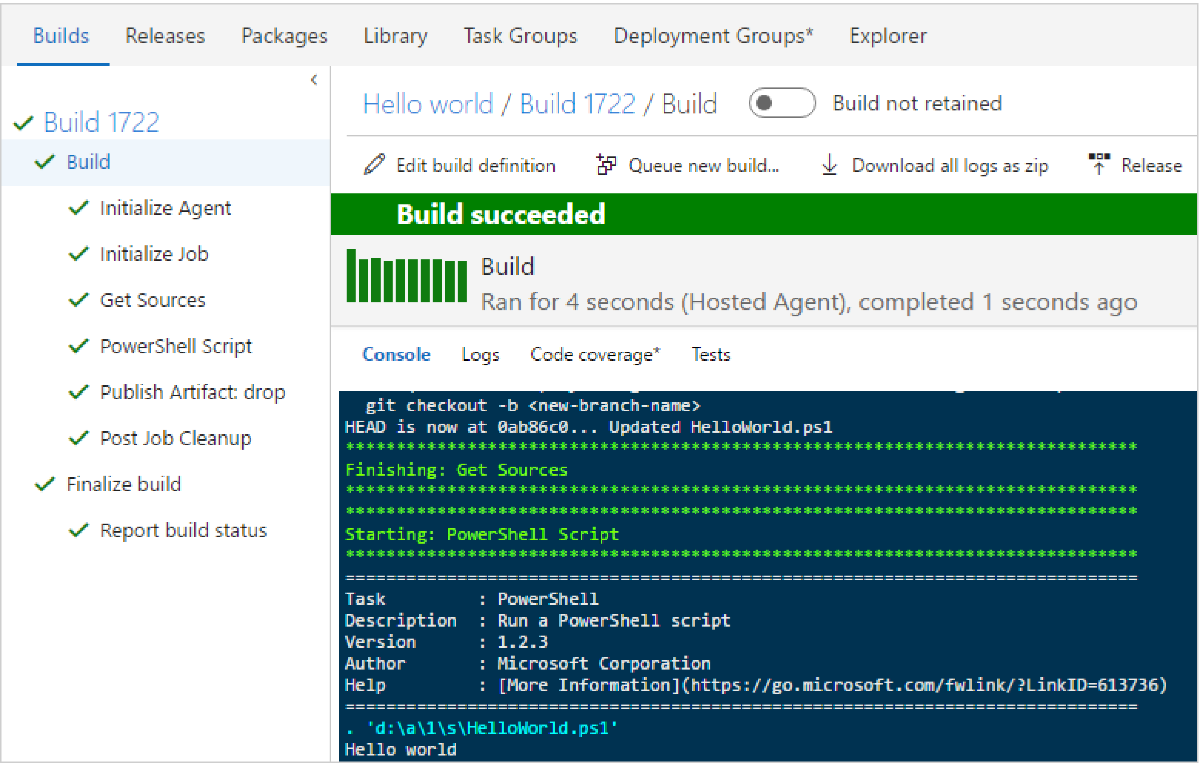
5. On the **Artifacts** tab of the build, notice that the script is published as an artifact.



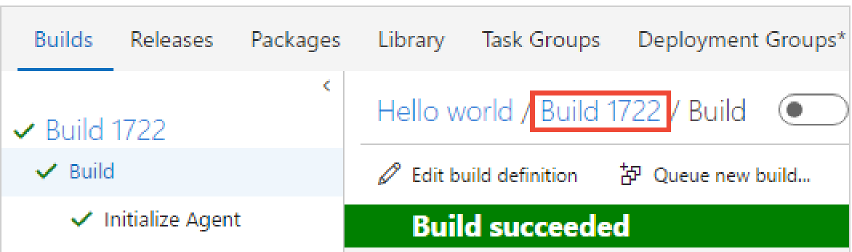
1. Select **Save & queue**, and then select **Save & queue**.

2. On the dialog box, select the **Queue** button.

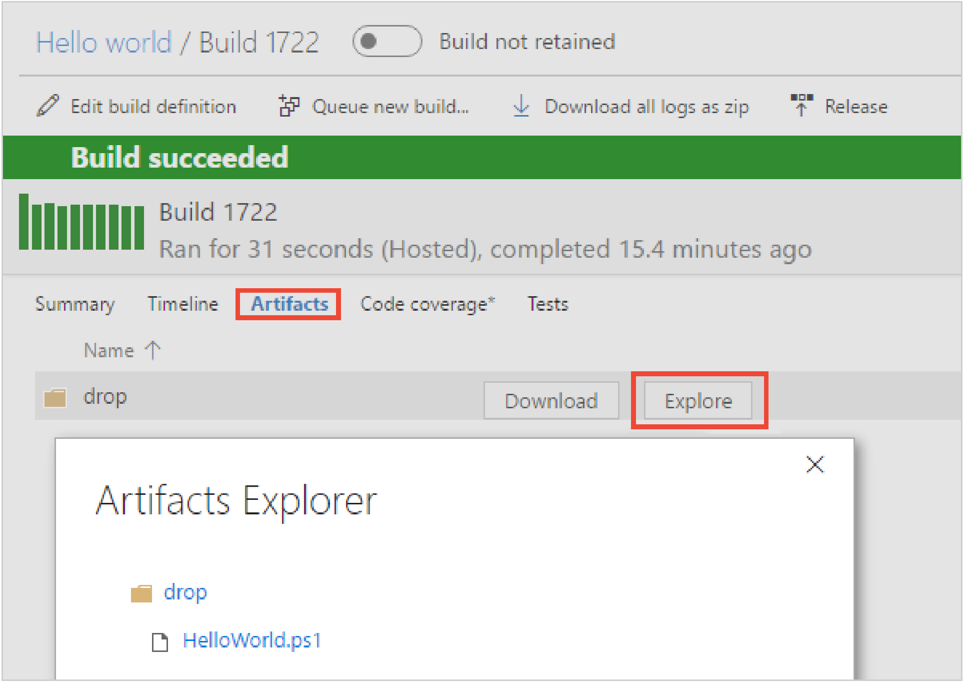
   This queues a new build on the agent. Once the agent is allocated, you'll start seeing the live logs of the build. Notice that the PowerShell script is run as part of the build, and that "Hello world" is printed to the console.



3. Go to the build summary.



4. On the **Artifacts** tab of the build, notice that the script is published as an artifact.



You can view a summary of all the builds or drill into the logs for each build at any time by navigating to the **Builds** tab in the **Build and Release** hub. For each build, you can also view a list of commits that were built and the work items associated with each commit. You can also run tests in each build and analyze the test failures.

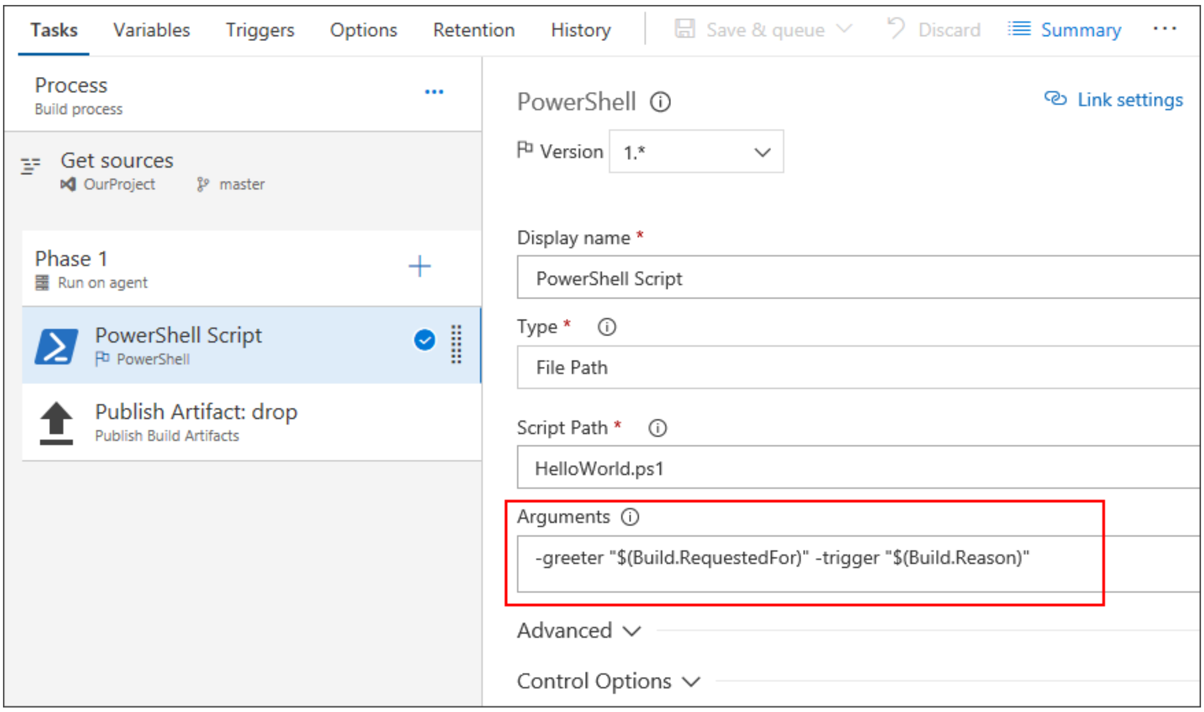
Add some variables and commit a change to your script

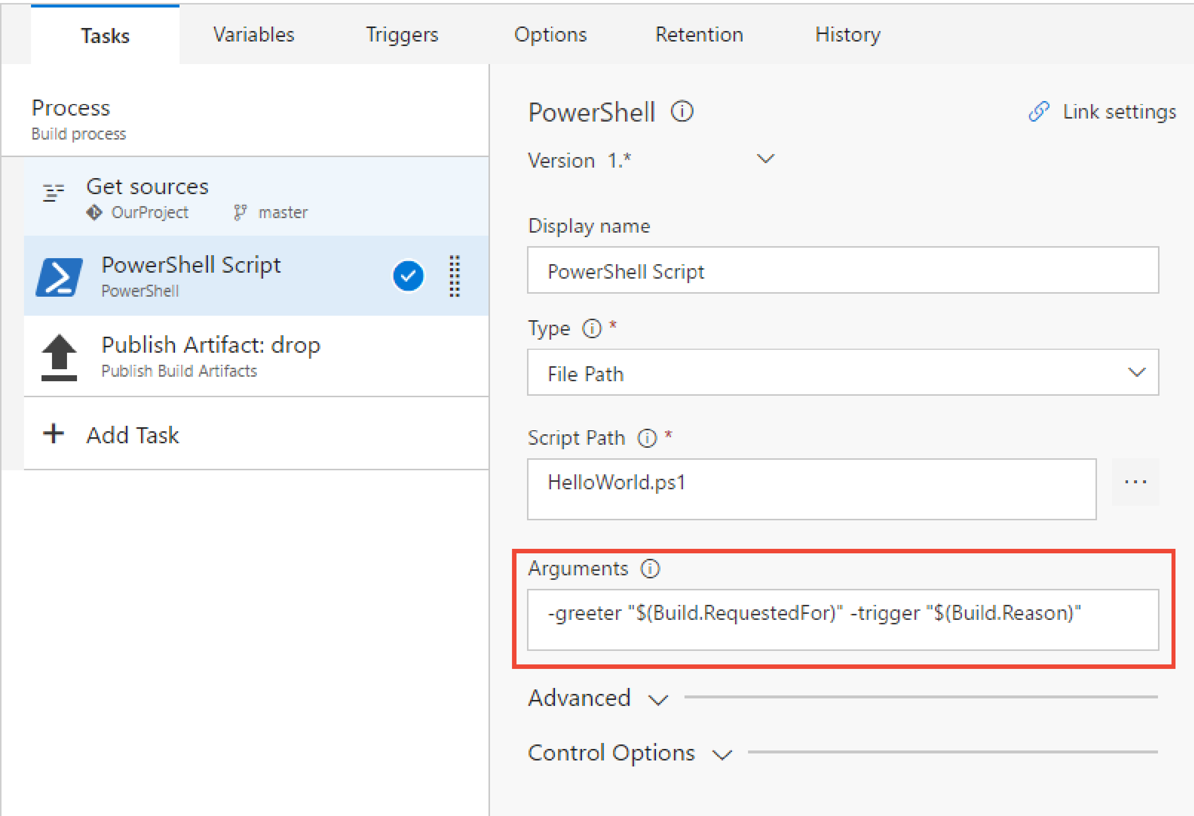
We'll pass some build variables to the script to make our process a bit more interesting. Then we'll commit a change to a script and watch the CI process run automatically to validate the change.

1. Edit your build pipeline.

2. On the **Tasks** tab, select the PowerShell script task.

3. Add these arguments.





Arguments

```
-greeter "$(Build.RequestedFor)" -trigger "$(Build.Reason)"
```

4. Save the build pipeline.

5. Go to the **Code** hub, **Files** tab.

6. Select the **HelloWorld.ps1** file, and then **Edit** the file.

7. Change the script as follows:

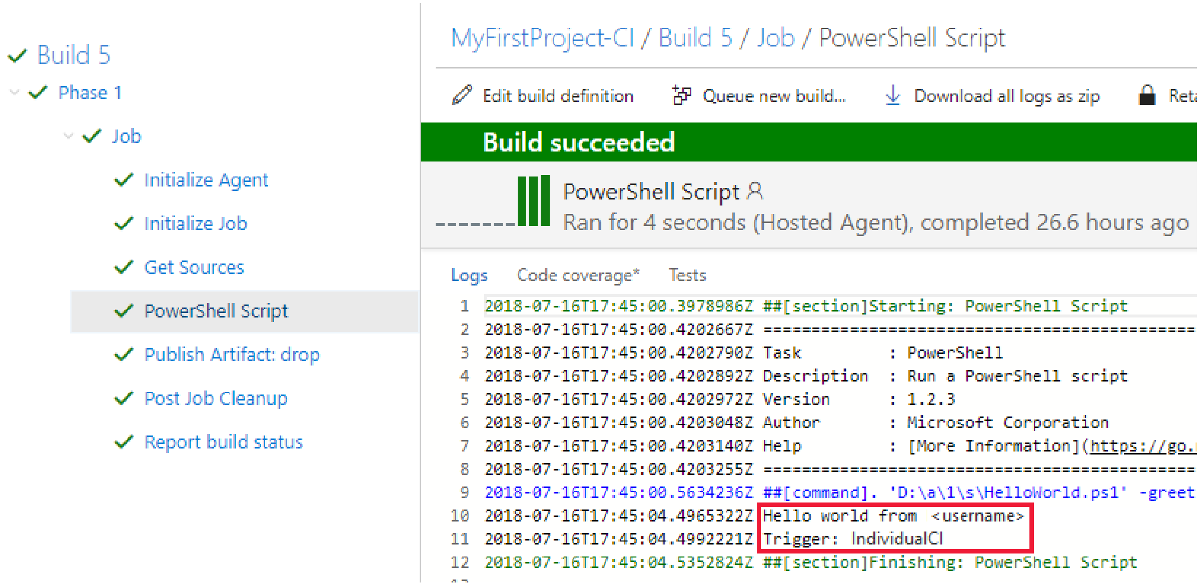
```
Param(
[string]$greeter,
[string]$trigger
)
Write-Host "Hello world" from $greeter
Write-Host Trigger: $trigger
```

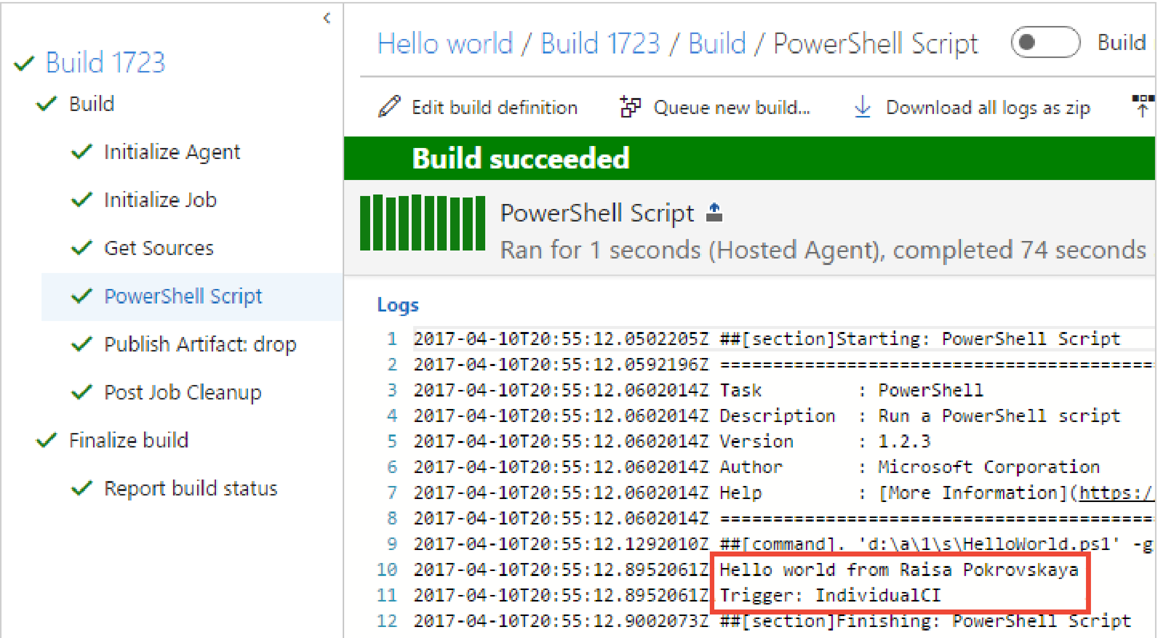
8. **Commit** (save) the script.

9. Go to the **Build and Release** hub and select **Queued**. Notice under the **Queued or running** section that a build is automatically triggered by the change that you committed.

10. Select the new build that was created and view its log.

11. Notice that the person who changed the code has their name printed in the greeting message. You also see printed that this was a CI build.





> We just introduced the concept of build variables in these steps. We printed the value of a variable that is automatically predefined and initialized by the system. You can also define custom variables and use them either in arguments to your tasks, or as environment variables within your scripts. To learn more about variables, see Build variables.

You've got a build pipeline. What's next?

You've just created a build pipeline that automatically builds and validates whatever code is checked in by your team. At this point you can continue to the next section to learn about release pipelines. Or, if you prefer, you can skip ahead to create a build pipeline for your app.

Create a release pipeline

Define the process for running the script in two environments.

1. Go to the **Build and Release** tab, and then select **Releases**.

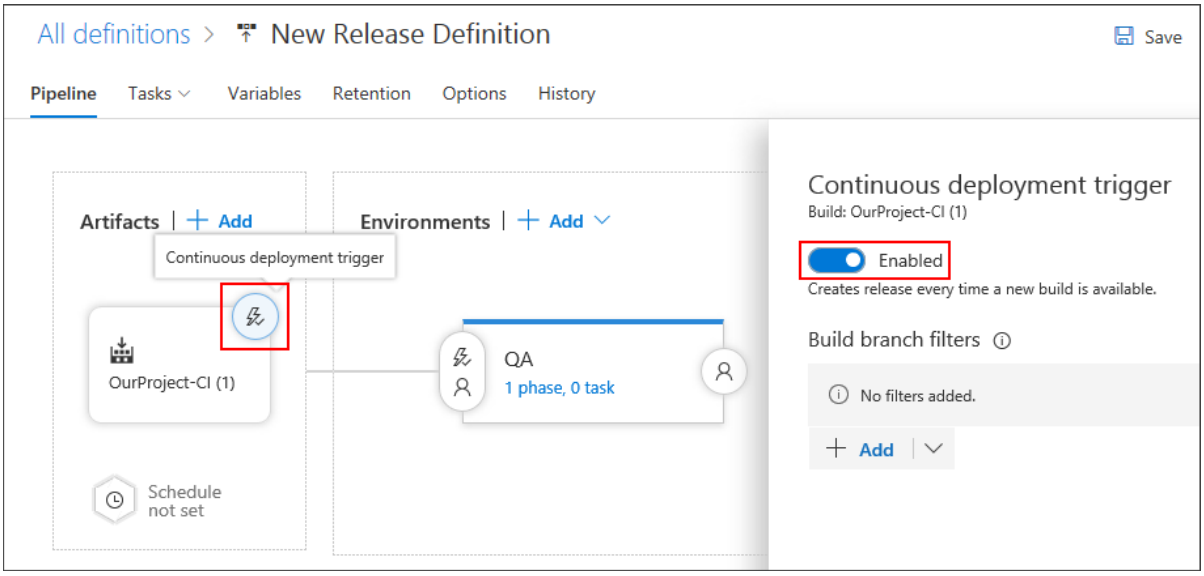
2. Select the action to create a **New pipeline**. If a release pipeline is already created, select the plus sign **(+)** and then select **Create release pipeline**.

3. Select the action to start with an **empty process**.

4. Name the environment **QA**.

5. In the Artifacts panel, select **+ Add** and specify a **Source (Build pipeline)**. Select **Add**.

6. Select the **Lightning bolt** to trigger continuous deployment and then enable the **Continuous deployment trigger** on the right.



7. Select the **Tasks** tab and select your **QA** environment.

8. Select the plus sign **(+)** for the phase to add a task to the phase.

9. On the **Add tasks** dialog box, select **Utility**, locate the **PowerShell** task, and then select its **Add** button.

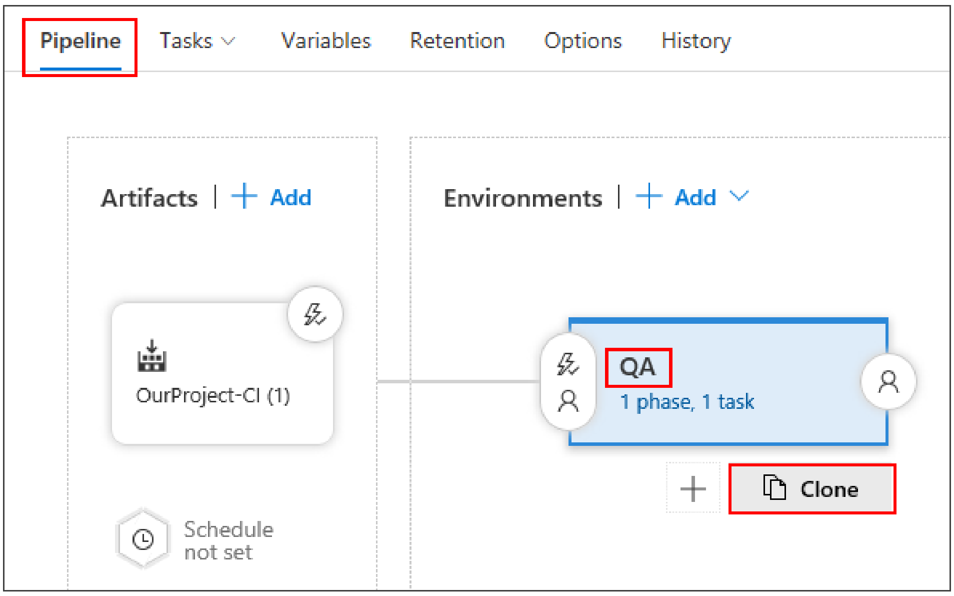
10. On the left side, select your new **PowerShell** script task.

11. For the **Script Path** argument, select the **...** button to browse your artifacts and select the script you created.

12. Add these **Arguments**:

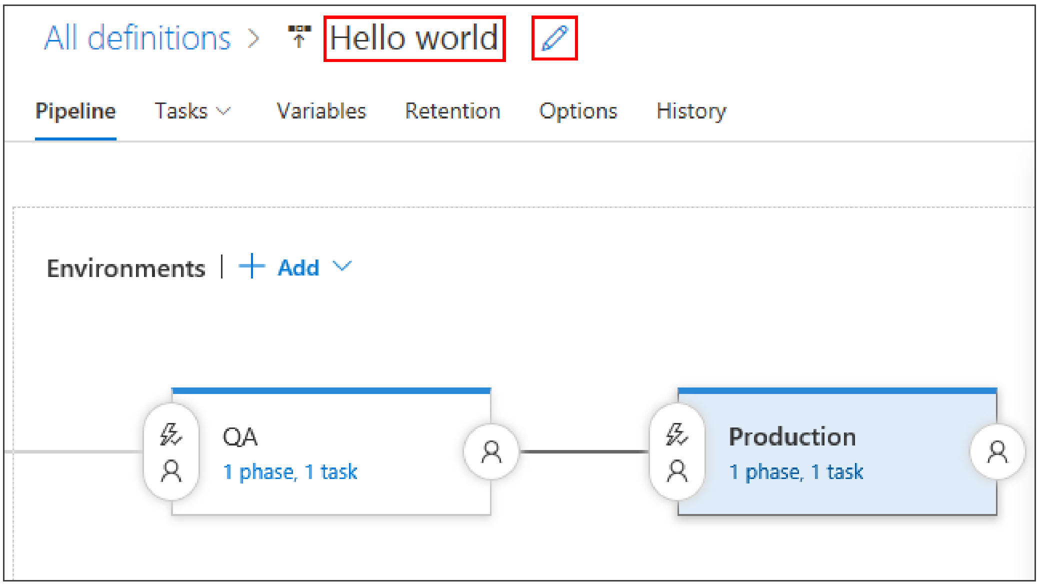
```
-greeter "$(Release.RequestedFor)" -trigger "$(Build.DefinitionName)"
```

13. On the **Pipeline** tab, select the **QA** environment and select **Clone**.



14. Rename the cloned environment **Production**.

15. Rename the release pipeline **Hello world**.



16. Save the release pipeline.

1. Go to the **Build and Release** hub, and then to the **Releases** tab.

2. Select the action to create a **New pipeline**.

3. On the dialog box, select the **Empty** template and select **Next**.

4. Make sure that your **Hello world** build pipeline that you created above is selected. Select **Continuous deployment**, and then select **Create**.

5. Select **Add tasks** in the environment.

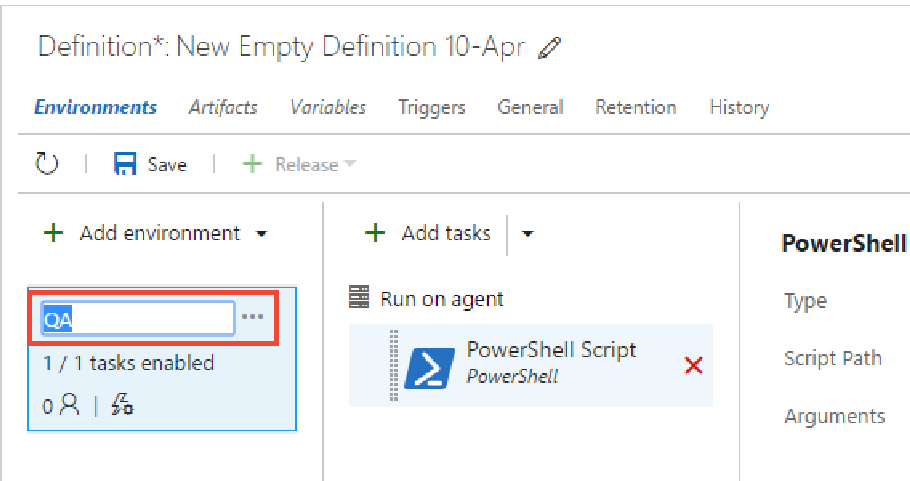
6. On the **Task catalog** dialog box, select **Utility**, locate the **PowerShell** task, and then select its **Add** button. Select the **Close** button.

7. For the **Script Path** argument, select the `...` button to browse your artifacts and select the script you created.

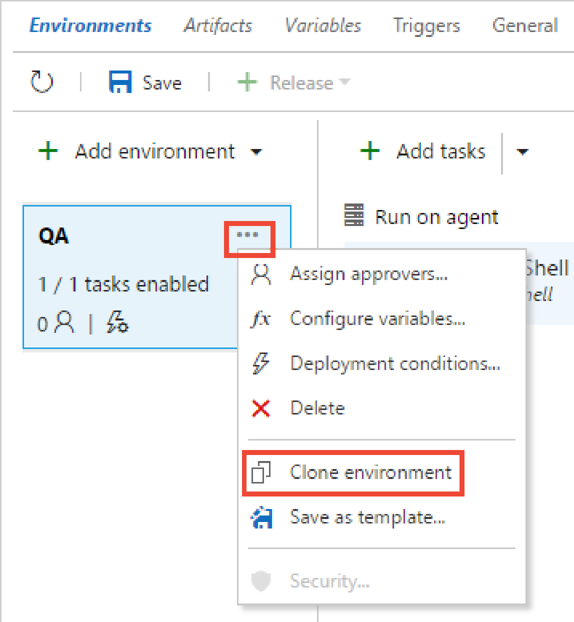
8. Add these **Arguments**:

```
-greeter "$(Release.RequestedFor)" -trigger "$(Build.DefinitionName)"
```

9. Rename the environment **QA**.



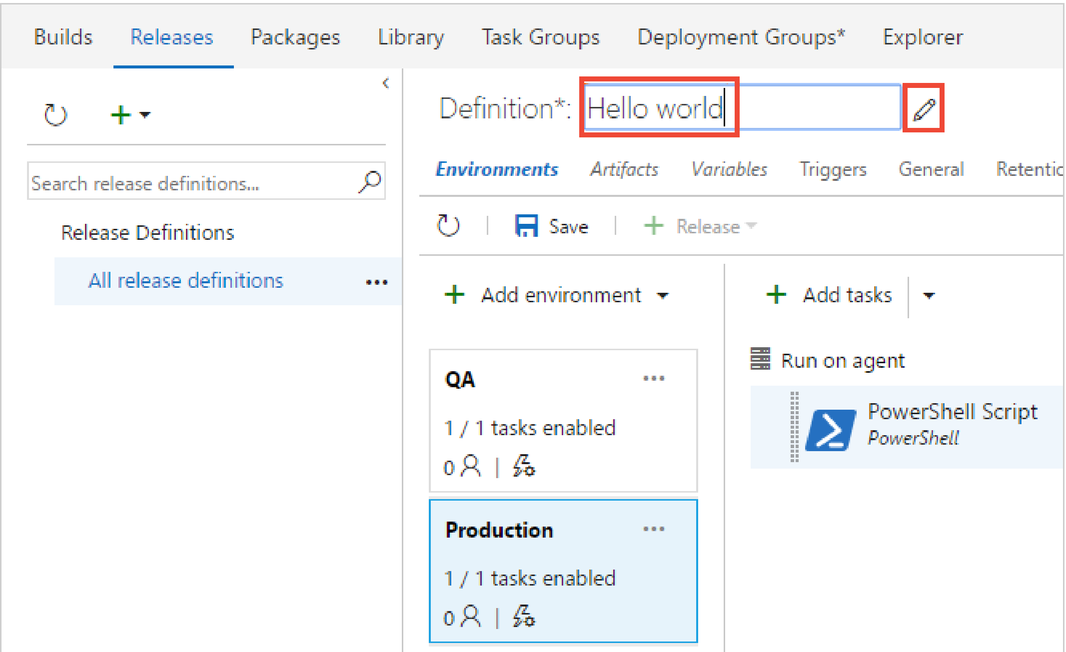
10. **Clone** the **QA** environment.



Leave **Automatically approve** and **Deploy automatically...** selected, and select **Create**.

11. Rename the new environment **Production**.

12. Rename the release pipeline **Hello world**.



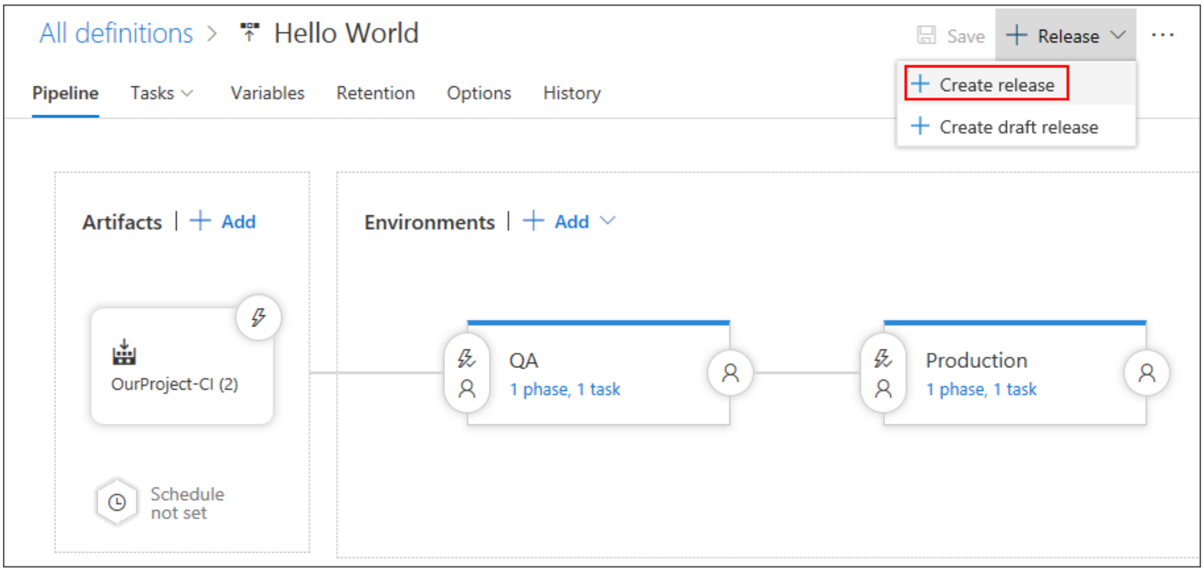
13. Save the release pipeline.

> A release pipeline is a collection of environments to which the application build artifacts are deployed. It also defines the actual deployment process for each environment, as well as how the artifacts are promoted from one environment to another.
>
> Also, notice that we used some variables in our script arguments. In this case, we used release variables instead of the build variables we used for the build pipeline.

Deploy a release

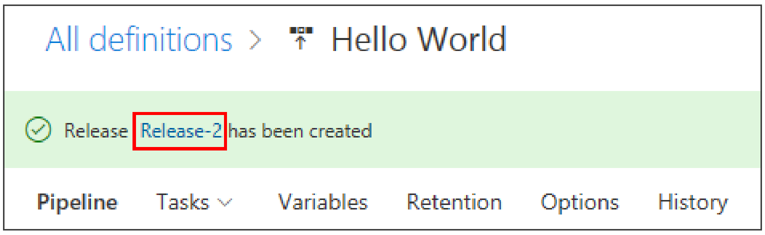
Run the script in each environment.

1. Create a new release.

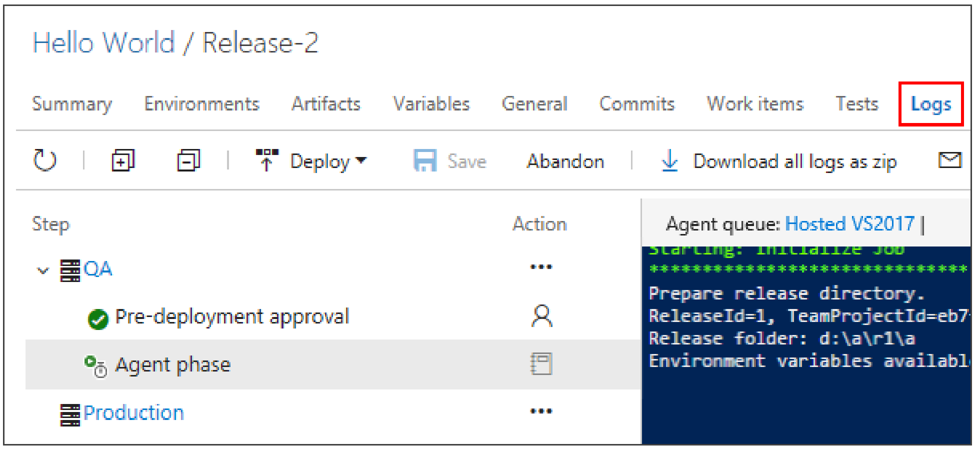


2. Define the trigger settings and artifact source for the release and then select **Create**.

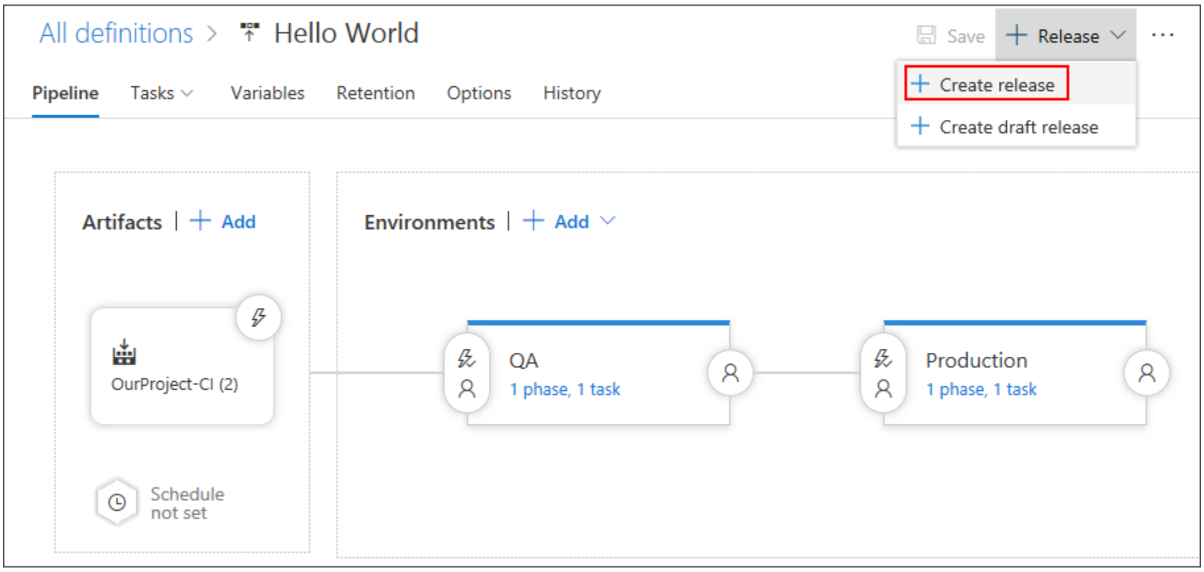
3. Open the release that you just created.



4. View the logs to get real-time data about the release.

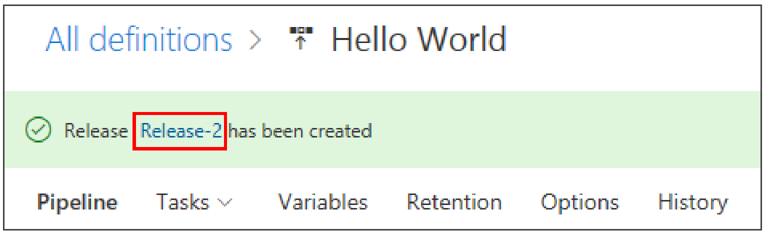


1. Create a new release.

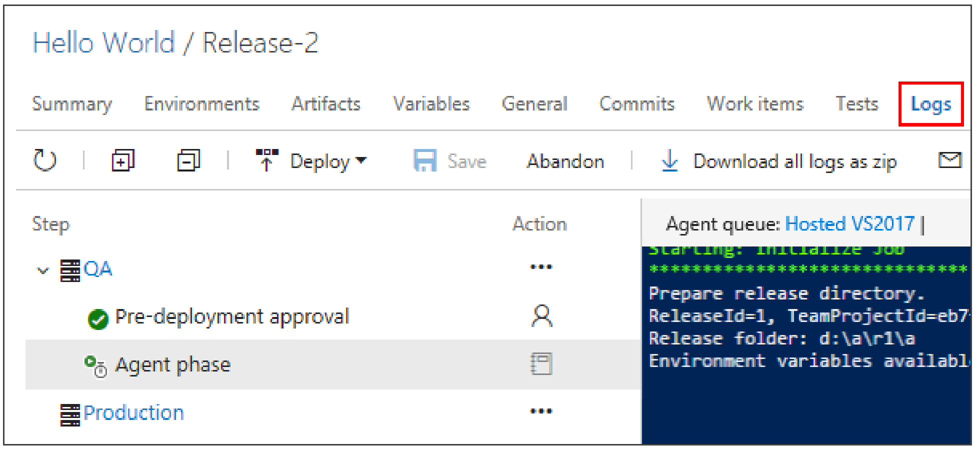


2. Define the trigger settings and artifact source for the release and then select **Queue**.

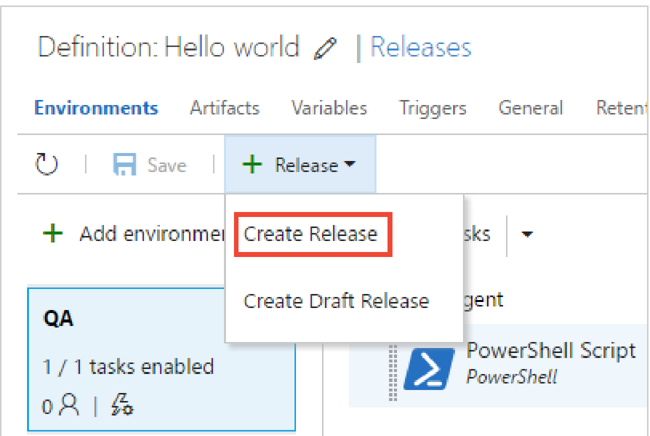
3. Open the release that you just created.



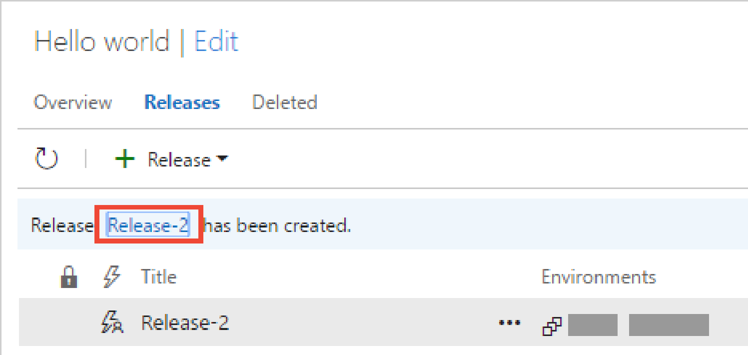
4. View the logs to get real-time data about the release.



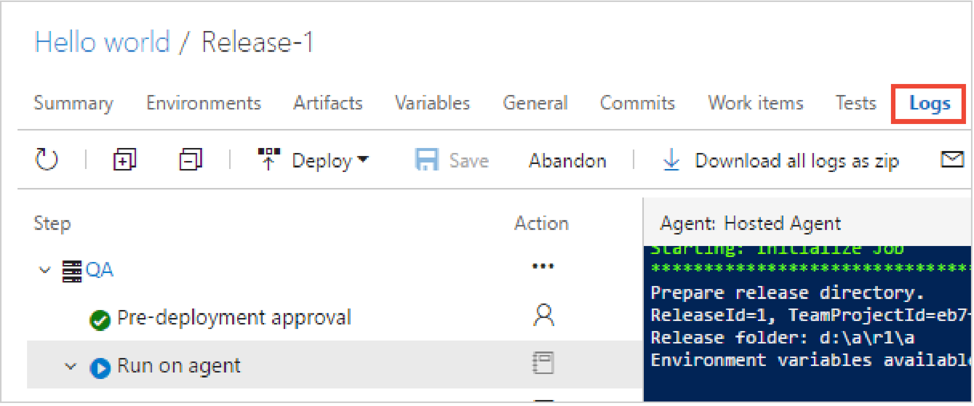
1. Create a new release.



2. Open the release that you just created.



3. View the logs to get real-time data about the release.



> You can track the progress of each release to see if it has been deployed to all the environments. You can track the commits that are part of each release, the associated work items, and the results of any test runs that you've added to the release process.

Change your code and watch it automatically deploy to production

We'll make one more change to the script. This time it will automatically build and then get deployed all the way to the production environment.

1. Go to the **Code** hub, **Files** tab, edit the **HelloWorld.ps1** file, and change it as follows:

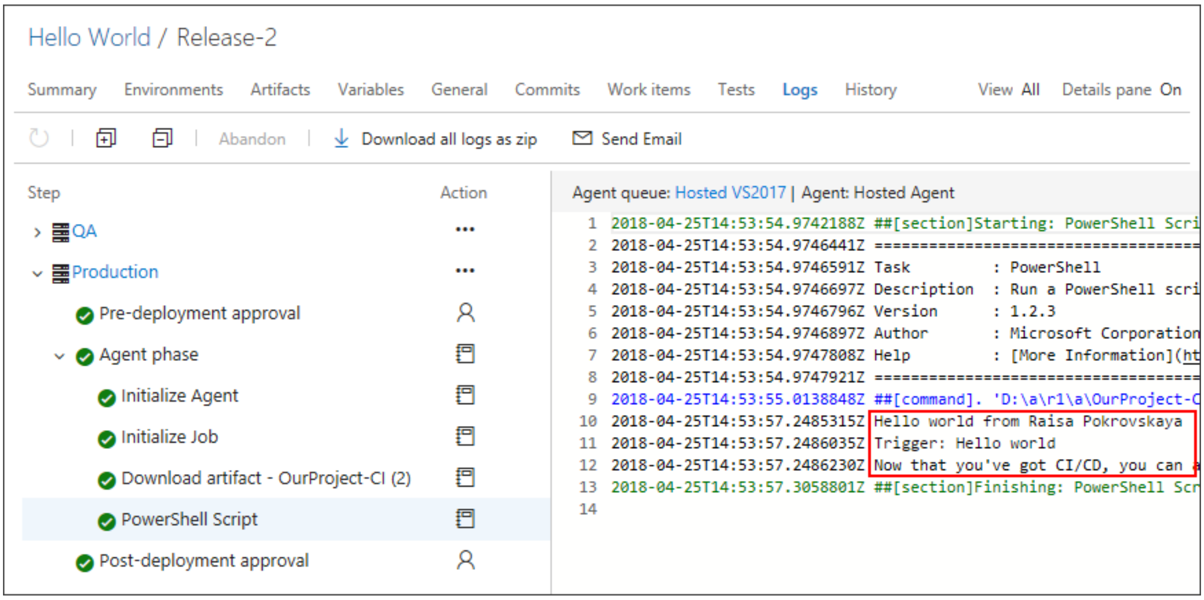
```
Param(
[string]$greeter,
[string]$trigger
)
Write-Host "Hello world" from $greeter
Write-Host Trigger: $trigger
Write-Host "Now that you've got CI/CD, you can automatically deploy your app every time your team checks
in code."
```

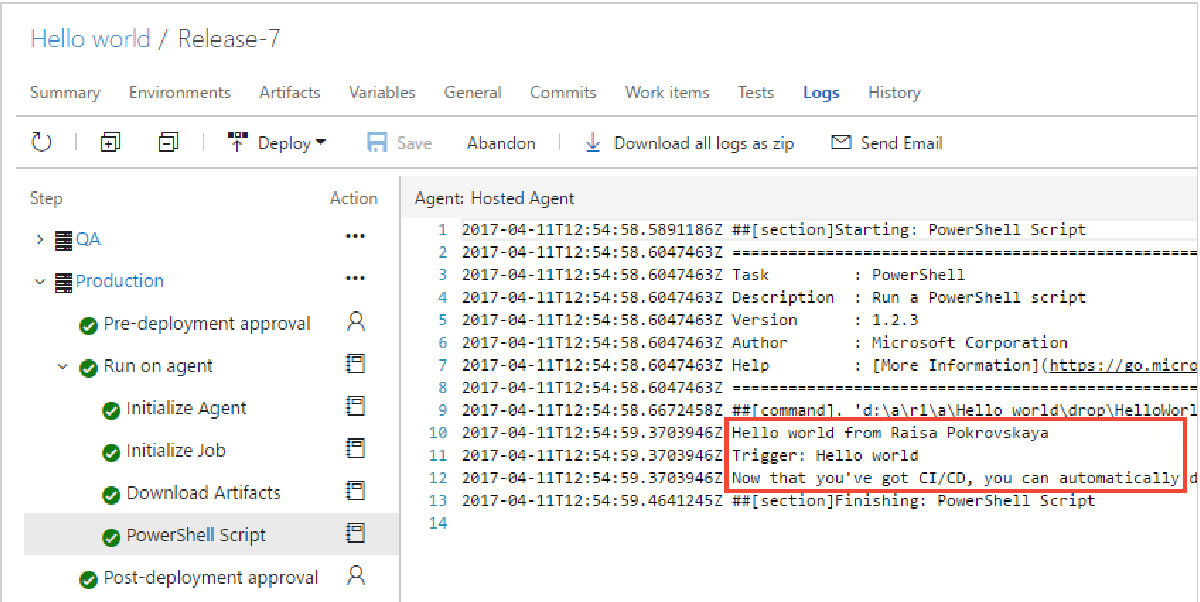
2. **Commit** (save) the script.

3. Select the **Builds** tab to see the build queued and run.

4. After the build is completed, select the **Releases** tab, open the new release, and then go to the **Logs**.

   Your new code automatically is deployed in the **QA** environment, and then in the **Production** environment.





> In many cases, you probably would want to edit the release process so that the production deployment happens only after some testing and approvals are in place. See Approvals and gates overview.

Next steps

You've just learned the basics of using the designer to create and run a VSTS build and release process. Now you're ready to configure your build pipeline for the programming language you're using. Go ahead and create a new build pipeline, and this time, use one of the following templates.

LANGUAGE	TEMPLATE TO USE
.NET	ASP.NET
.NET Core	ASP.NET Core
C++	.NET Desktop
Go	Go
Java	Gradle

LANGUAGE	TEMPLATE TO USE
JavaScript	NodeJS with Gulp (Grunt is also an option)
Xcode	Xcode

Q & A

Where can I read articles about DevOps and CI/CD?

What is Continuous Integration?

What is Continuous Delivery?

What is DevOps?

What kinds of version control can I use

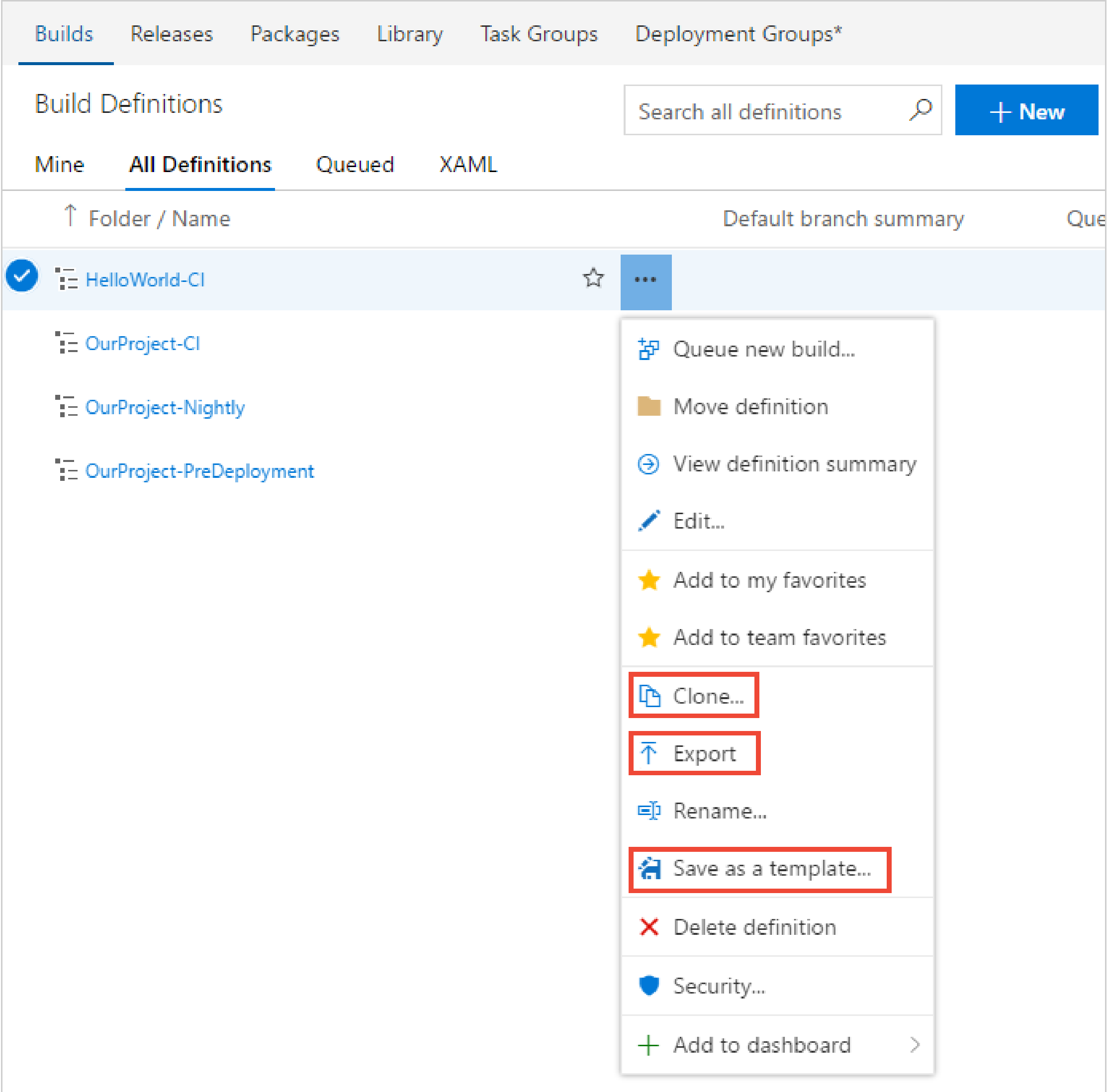
We've used a Git repository in VSTS to keep things focused on CI/CD for this tutorial.

When you're ready to get going with CI/CD for your app, you can use the version control system of your choice:

- Clients
  - Visual Studio Code for Windows, macOS, and Linux
  - Visual Studio with Git for Windows or Visual Studio for Mac
  - Visual Studio with TFVC
  - Eclipse
  - Xcode
  - IntelliJ
  - Command line
- Services
  - VSTS
  - Git service providers such as GitHub and Bitbucket
  - Subversion

How do I replicate a pipeline?

If your pipeline has a pattern that you want to replicate in other definitions, clone it, export it, or save it as a template.



After you clone a pipeline, you can make changes and then save it.

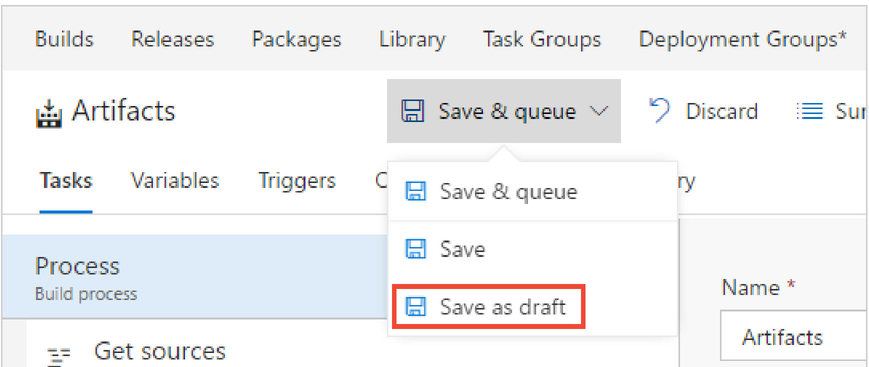
After you export a pipeline, you can import it from the **All pipelines** tab.

After you create a template, your team members can use it to follow the pattern in new pipelines.

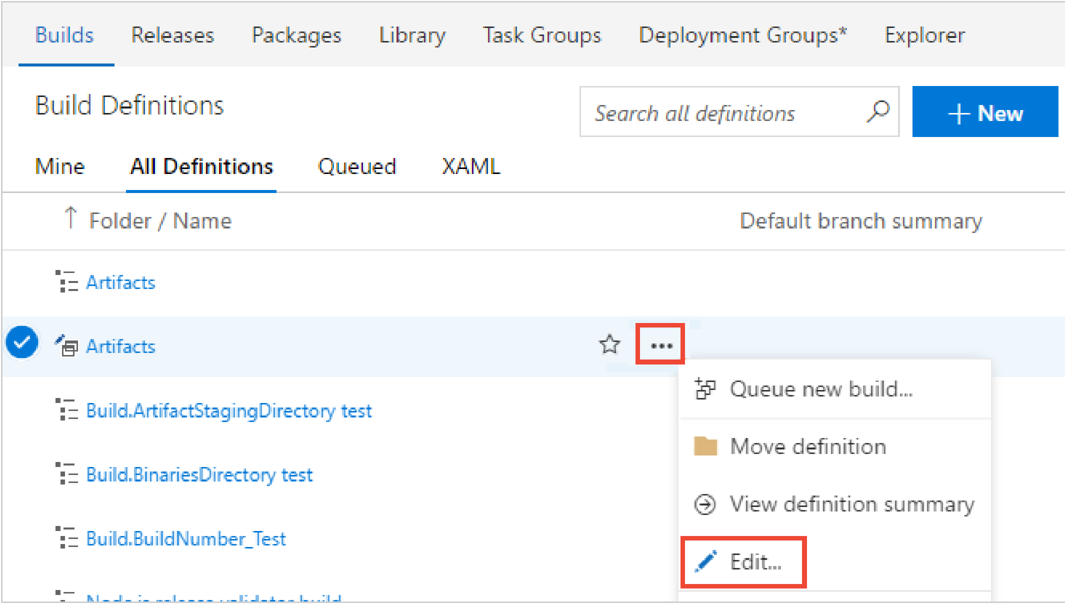
> **TIP**
> If you're using the **New Build Editor**, then your custom templates are shown at the bottom of the list.

How do I work with drafts?

If you're editing a build pipeline and you want to test some changes that are not yet ready for production, you can save it as a draft.



You can edit and test your draft as needed.



When you're ready you can publish the draft to merge the changes into your build pipeline.

Or, if you decide to discard the draft, you can delete it from the **All Pipeline** tab shown above.

What else can I do when I queue a build?

You can queue builds automatically or manually.

When you manually queue a build, you can, for a single run of the build:

- Specify the queue into which the build goes.
- Add and modify some variables.
- Add demands.
- In a Git repository
  - Build a branch or a tag.
  - Build a commit.
- In a TFVC repository
  - Specify the source version as a label or changeset.
  - Run a private build of a shelveset. (You can use this option on either a Microsoft-hosted agent or a self-hosted agent.)

Where can I learn more about build pipeline settings?

To learn more about build pipeline settings, see:

- Getting sources
- Tasks
- Variables
- Triggers

- Options
- Retention
- History

How do I programmatically create a build pipeline?

REST API Reference: Create a build pipeline

> **NOTE**
>
> You can also manage builds and build pipelines from the command line or scripts using the VSTS CLI.

Plan and track work

8/10/2018 • 3 minutes to read

VSTS

You add work items to plan and manage your project. You use different types of work items to track different types of work—such as user stories or product backlog items, tasks, bugs, or issues. You can describe the work to be done, assign work, track status, and coordinate efforts within your team.

Learn how to add work items from the web portal and view work items you've created.

Prerequisites

You can start adding work items once you connect to a project. If you don't have an organization or project yet, create one in VSTS.

Add a work item

NOTE

The **New navigation** feature, which provides a vertical navigation experience, is in preview for VSTS. When you enable new navigation, you automatically enable several new agile tool features that are described in the New Work Hubs blog post.

On-premises TFS users can choose **Previous navigation** for guidance.

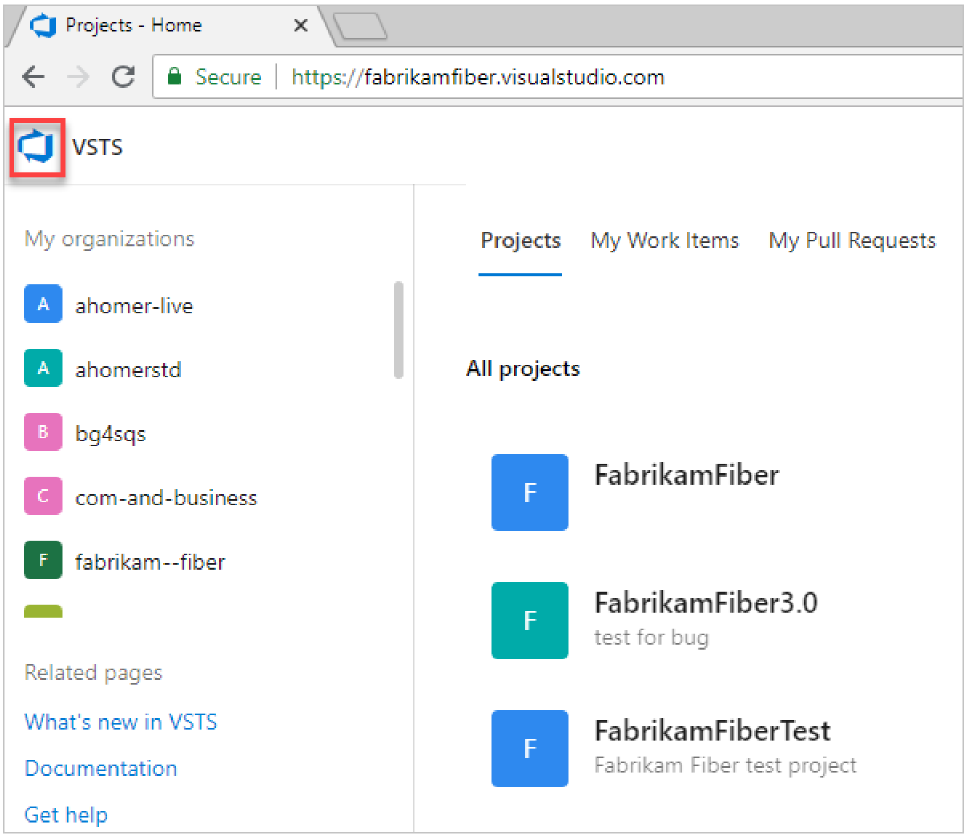
- New navigation
- Previous navigation

1. From your web browser, open the team project for your VSTS organization. If you don't have a team project, create one now. If you haven't been added as a team member, get invited now.

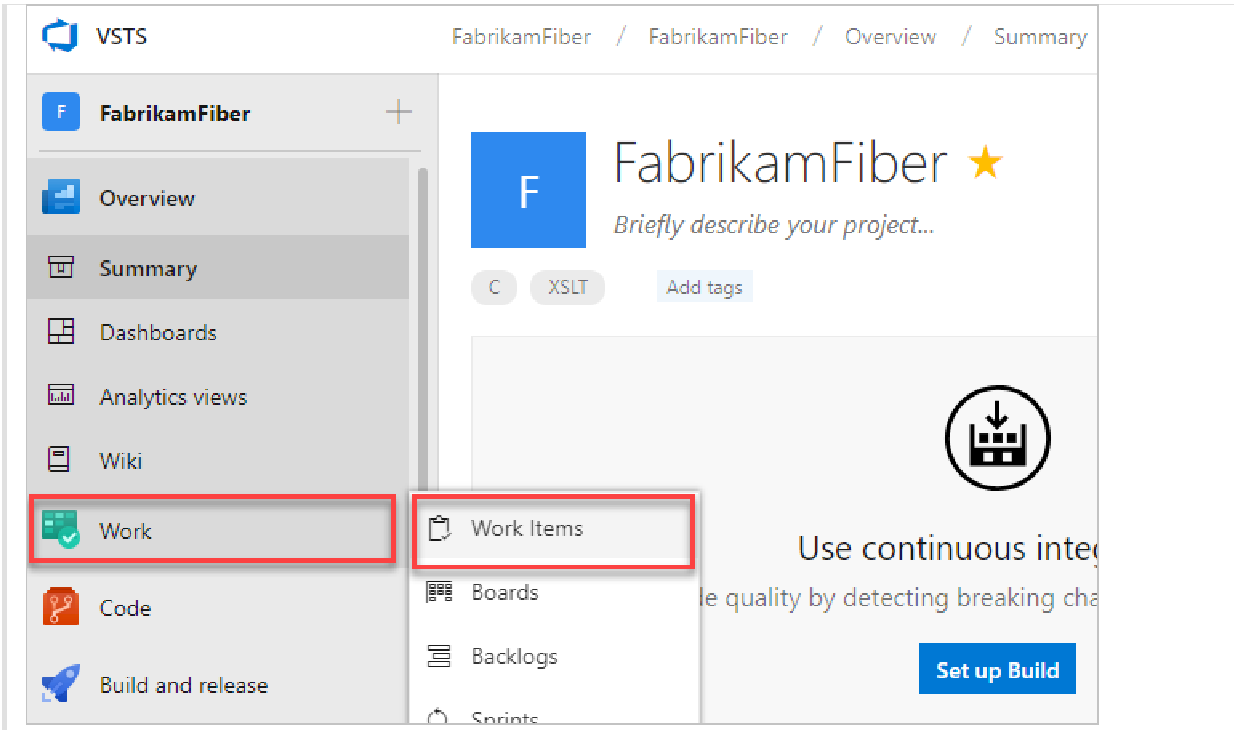
   The web browser URL follows this pattern: `https://{organization name}.visualstudio.com/{project name}`

   You can use this URL to quickly open the project, substituting the {organization name} and {project name} with your specific organization and project name (remove braces).

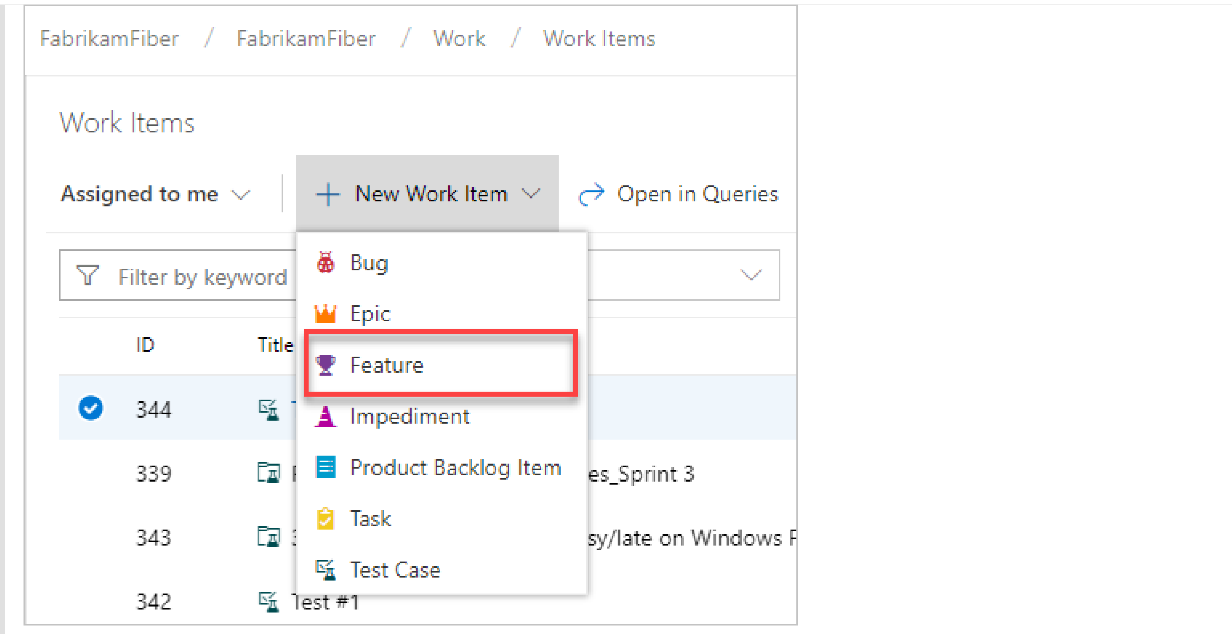
   If you don't see the team or team project you want, choose the VSTS icon to browse all projects and teams.



2. Select **Work** > **Work Items**.



3. Choose **New Work Item** and select the work item type you want. Here, we choose to create a **Feature**.



4. Enter a title and then **Save** the work item. Before you can change the State from its initial default, you must save it.

   That's it!

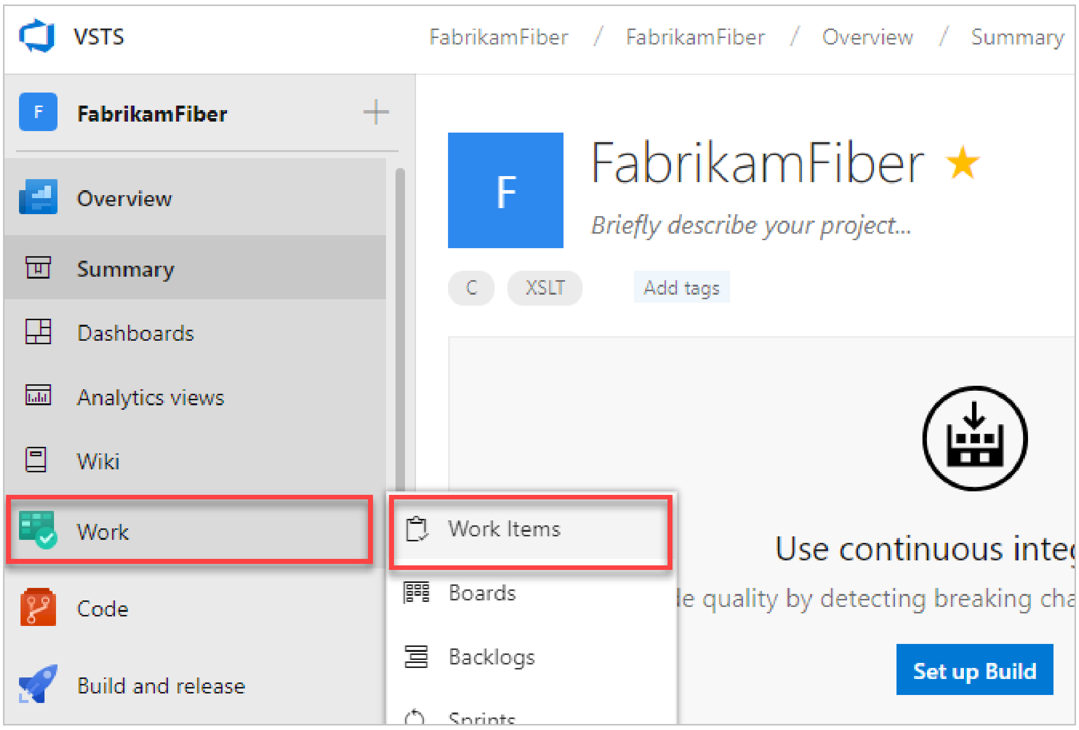
Create as many work items as you need of the type you need to track the work you want to manage.

> **NOTE**
>
> Depending on the process chosen when the team project was created—Scrum, Agile, or CMMI—the types of work items you can create will differ. For example, backlog items may be called product backlog items (Scrum), user stories (Agile), or requirements (CMMI). All three are similar: they describe the customer value to deliver and the work to be performed.
>
> For an overview of all three processes, see Choose a process.

View the work items you've just created

- New navigation
- Previous navigation

1. Go to your project and select **Work** > **Work Items**.



2. To view any work item listed, select the title.

For more information on using the Work Items page, see View and add work items.

Try this next

Add & run manual tests

To quickly add backlog items, such as user stories or requirements, see these topics:

Create your backlog Kanban quickstart

Or, learn more about planning and tracking work.

Add, run, and update inline tests

8/10/2018 • 4 minutes to read

VSTS

A quick and easy way to start manual testing is to add the test to the user story or bug you want to test. From the Kanban board, you can quickly define inline tests, or a set of manual tests, for a backlog item. Not only can you add tests, you can run them and update their status. If you're new to working with the Kanban board, see Kanban quickstart.

Tests you create from the Kanban board are automatically linked to the user story or backlog item.

Open your Kanban board

> **NOTE**
>
> The **New navigation** feature, which provides a vertical navigation experience, is in preview for VSTS. When you enable new navigation, you automatically enable several new agile tool features that are described in the New Work Hubs blog post.
>
> On-premises TFS users can choose **Previous navigation** for guidance.

- New navigation
- Previous navigation

1. From your web browser, open the project for your VSTS organization and select the **Work** hub. If you don't have a team project, create one now. If you haven't been added as a team member, get invited now.

   The URL follows this pattern: `https://{organization name}.visualstudio.com/{project name}/_backlogs`

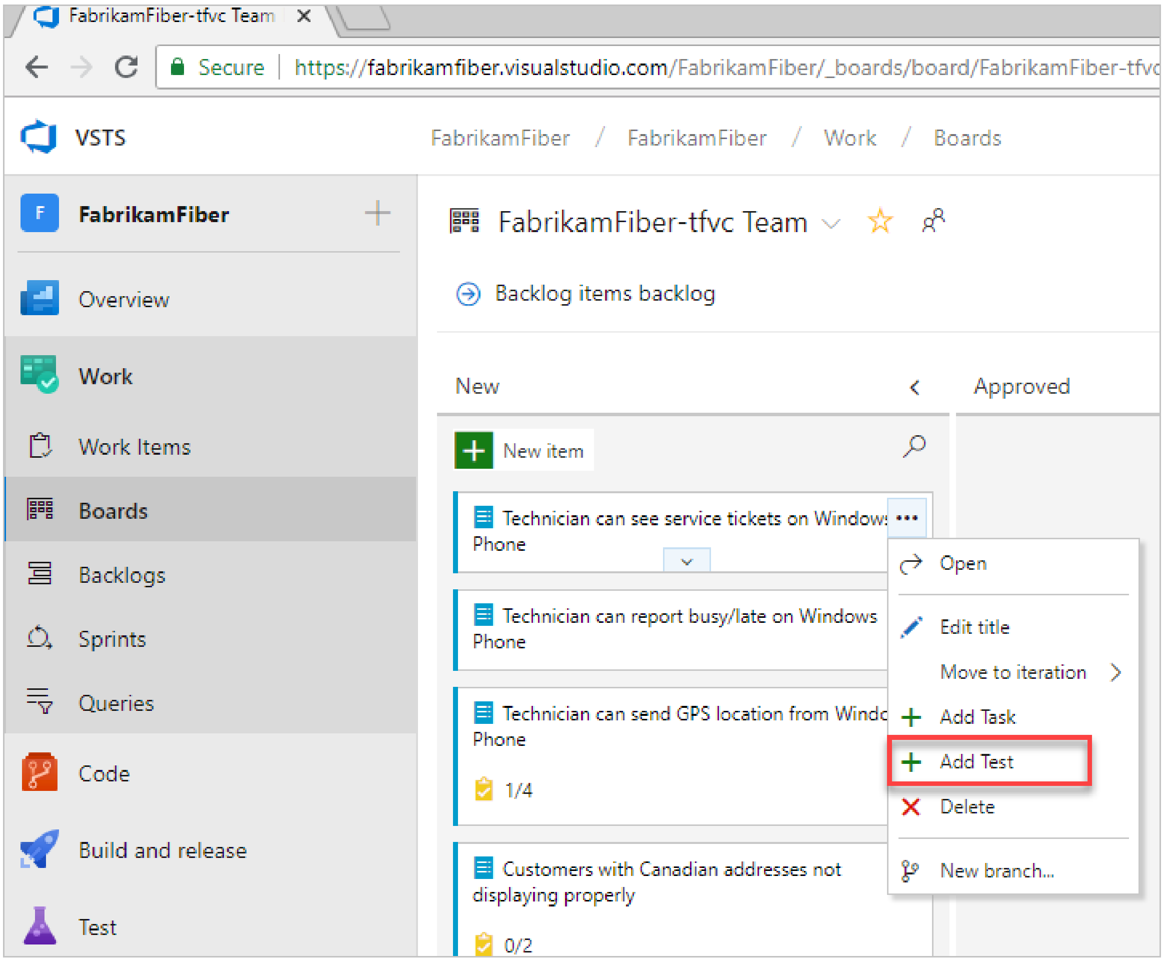
   If you don't see the team or project you want, choose the VSTS icon to browse all projects and teams.

2. Select **Boards** to open the Kanban board.

Add tests

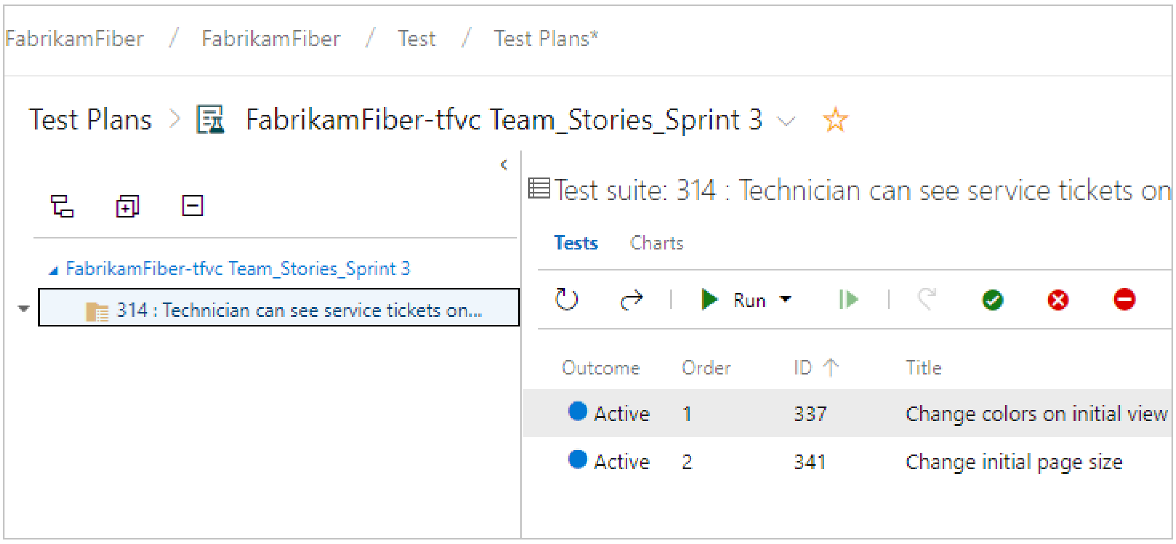
- New navigation
- Previous navigation

1. To start adding tests, open the menu for a work item.

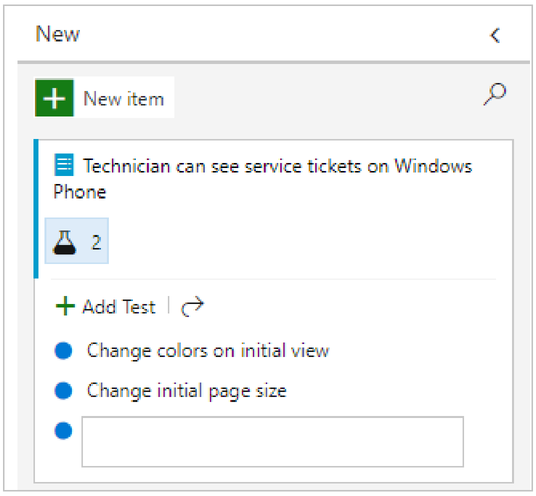


Adding inline tests is the same as adding test cases to a test suite. A default test plan and test suite are automatically created under which the manual test cases are grouped.

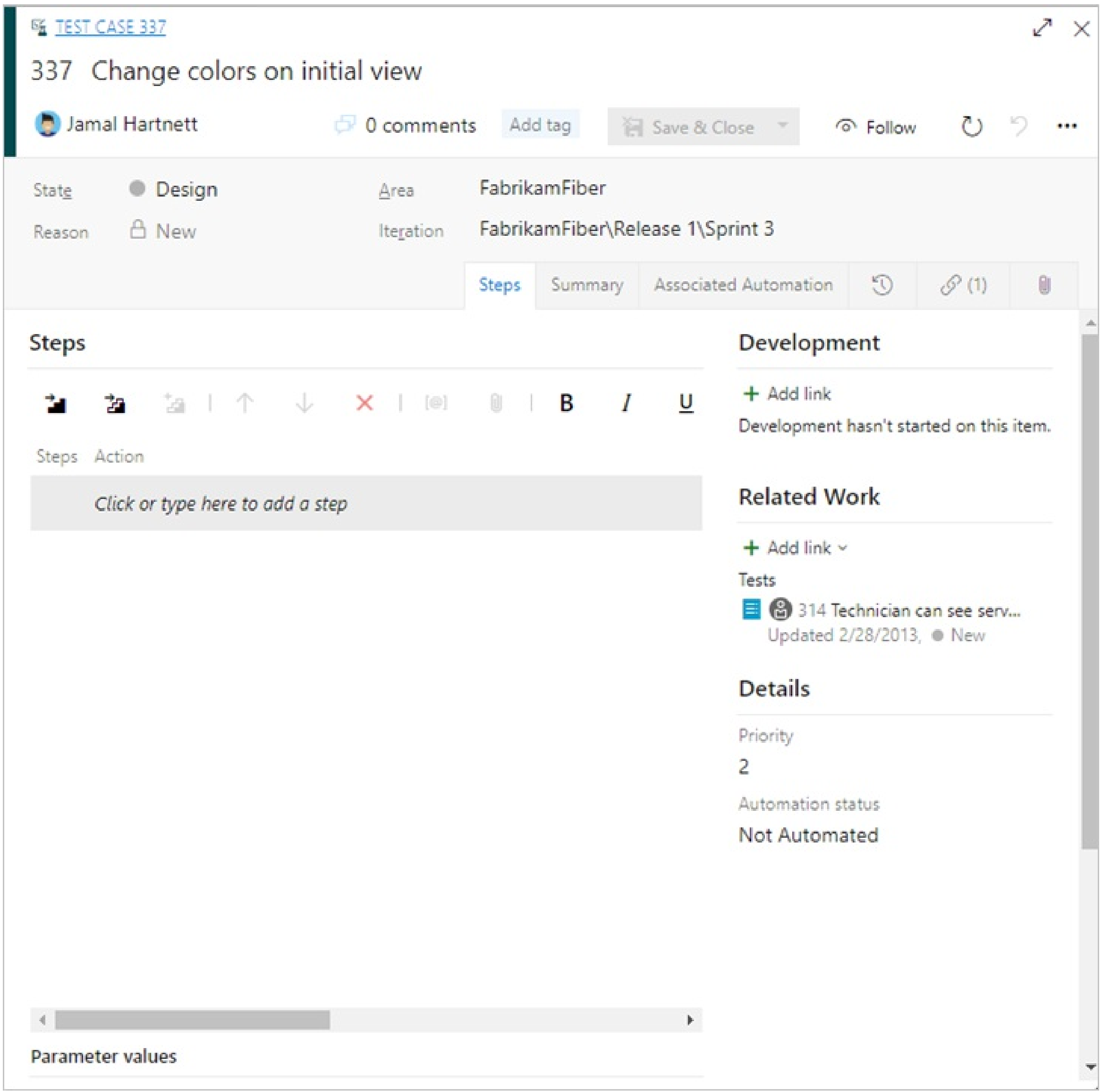
For example, a test suite is created for the following user story and inline tests are added to that suite. User story 314 is highlighted, which has two manual tests defined with IDs of 337 and 341.



2. If you have a number of tests to add, simply keep entering each title and choose **Enter**.



To add details to the test case, open it. You can select the title, double-select the inline item, or open the context menu and choose **Open**.



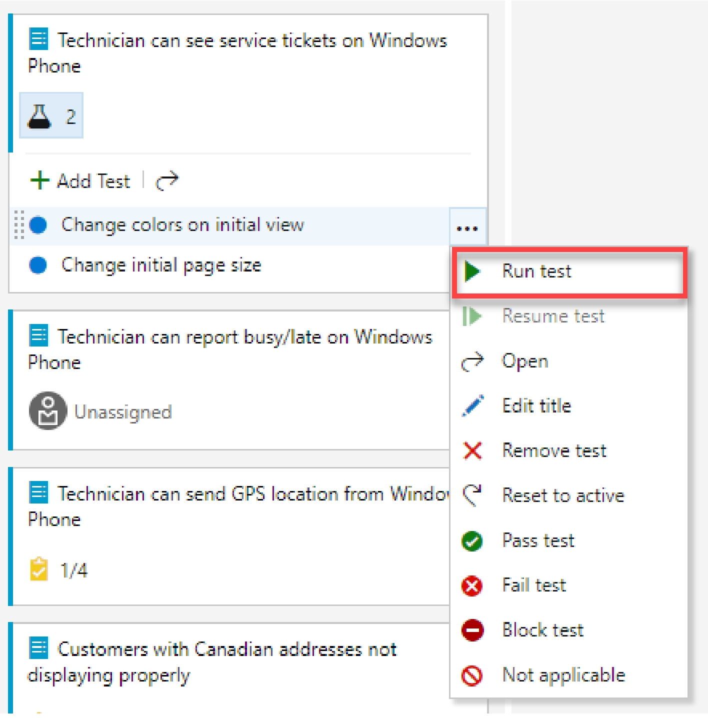
See Create manual tests to learn more about defining tests.

> **IMPORTANT**
>
> Prior to running the test, you must add details.

Run a test

- New navigation
- Previous navigation

Run the test by selecting **Run test** from the ⋯ actions menu for the inline test.

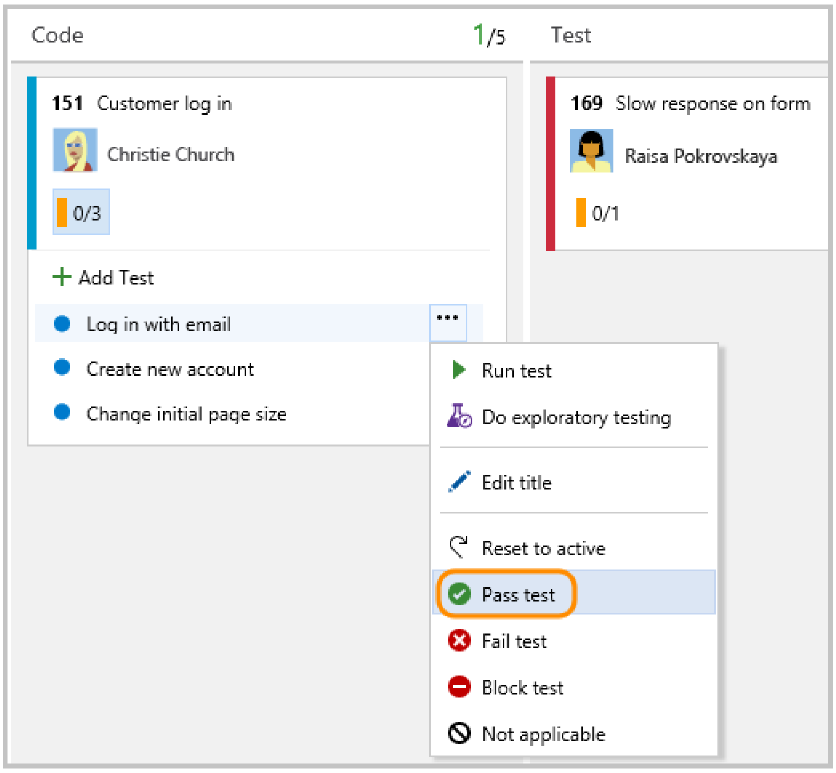


Microsoft Test Runner starts in a new browser instance. For details on running a test, see Run manual tests.

Update the status of a test

- New navigation
- Previous navigation

You can update the status of the test from the ⋯ actions menu.

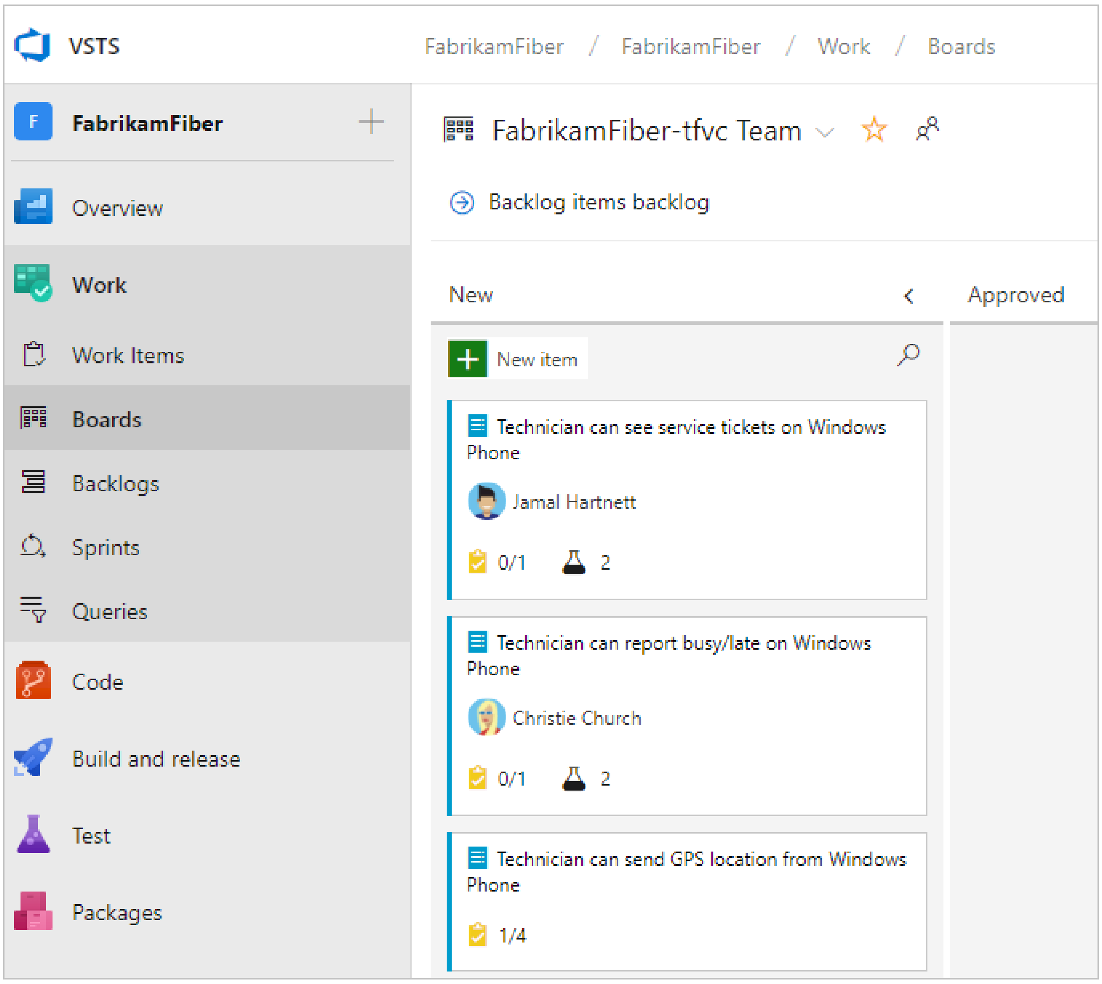


Updating the status of tests enable you to track test results.

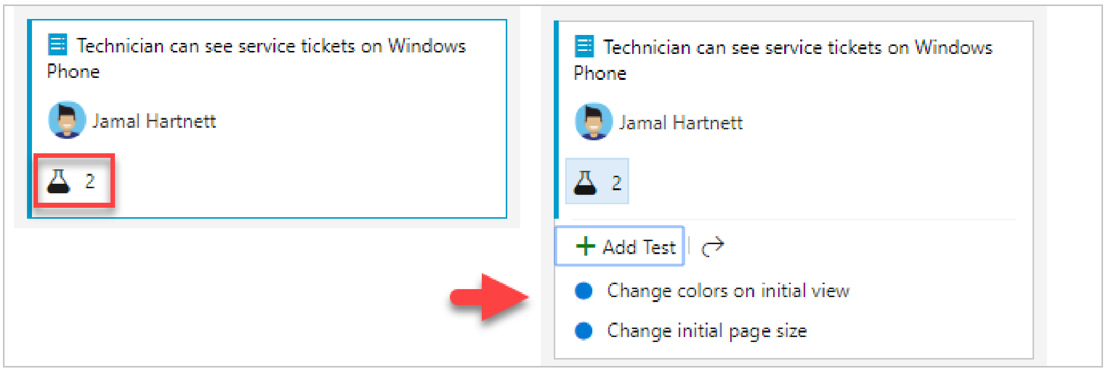
Expand or collapse inline tests

- New navigation
- Previous navigation

Upon first opening the Kanban board, you'll see an un-expanded view of checklists and tests.



Simply select the inline test summary to expand a collapsed set of tests. Select the same summary to collapse an expanded list.



Next steps

Use inline tests for lightweight traceability and to manage manual tests for user stories or other backlog items that they support.

Learn more about test case management Kanban quickstart

To initiate web-based exploratory testing for a user story, you need to install the Exploratory testing , see Exploratory test your web app directly in your browser.

Install free extensions for VSTS

6/29/2018 • 2 minutes to read

VSTS

To add new features and capabilities to your Visual Studio Team Services (VSTS) account, install extensions from the Visual Studio Marketplace. You can install free, preview, or paid extensions, this quickstart goes covers installing a free extension.

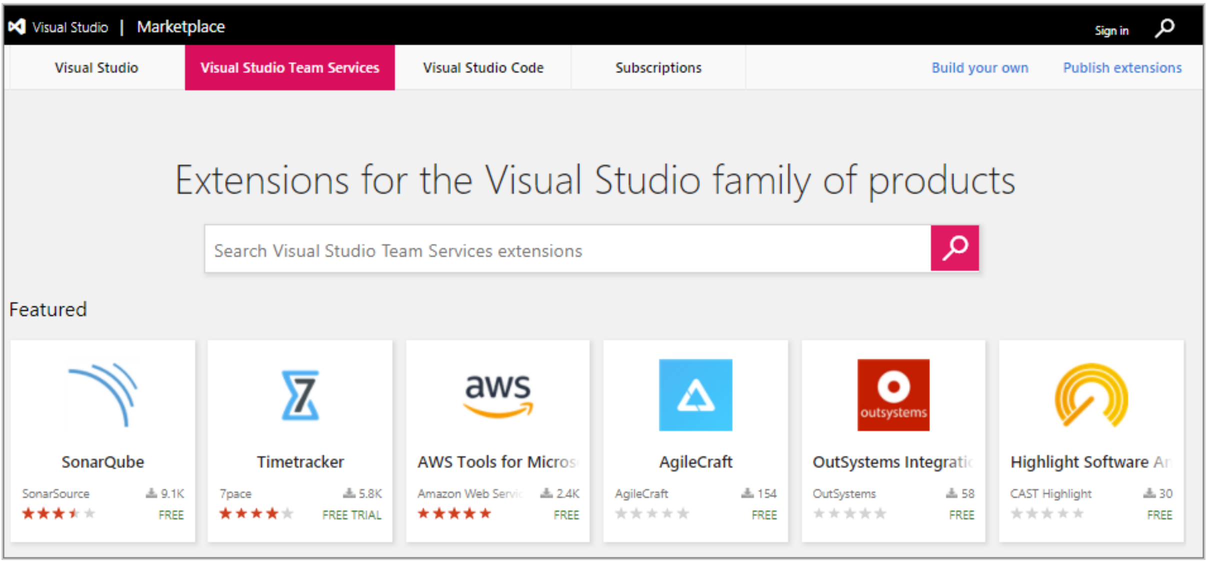
> **TIP**
>
> To learn about building your own VSTS extensions, see developing and publishing extensions.

What do I need to install an extension?

- Only VSTS project collection administrators or account owners can install extensions. If you don't have permissions, you can request extensions instead.
- Private extensions must be shared with your VSTS account to be installed. Check out the publishing documentation for information on how to share private extensions.

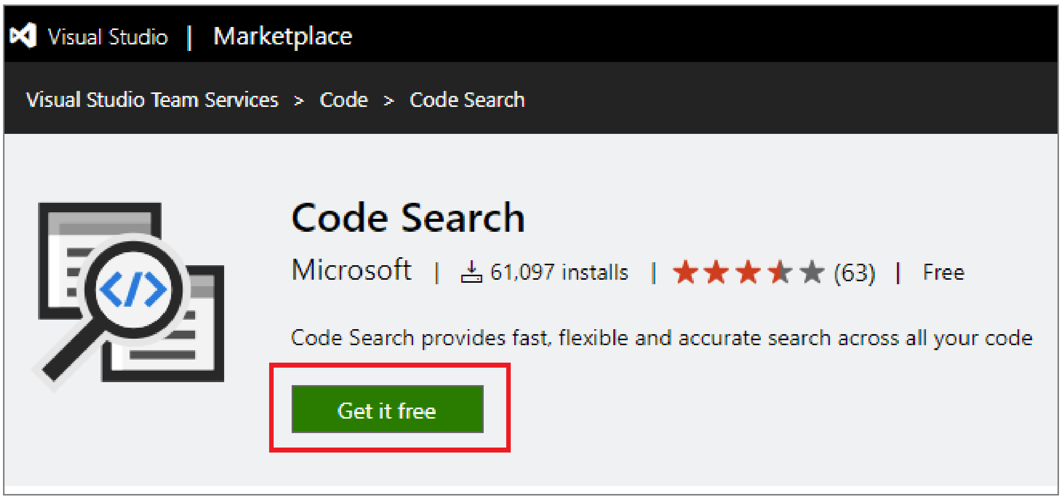
Install the extension

1. Sign in to the Visual Studio Marketplace > VSTS.

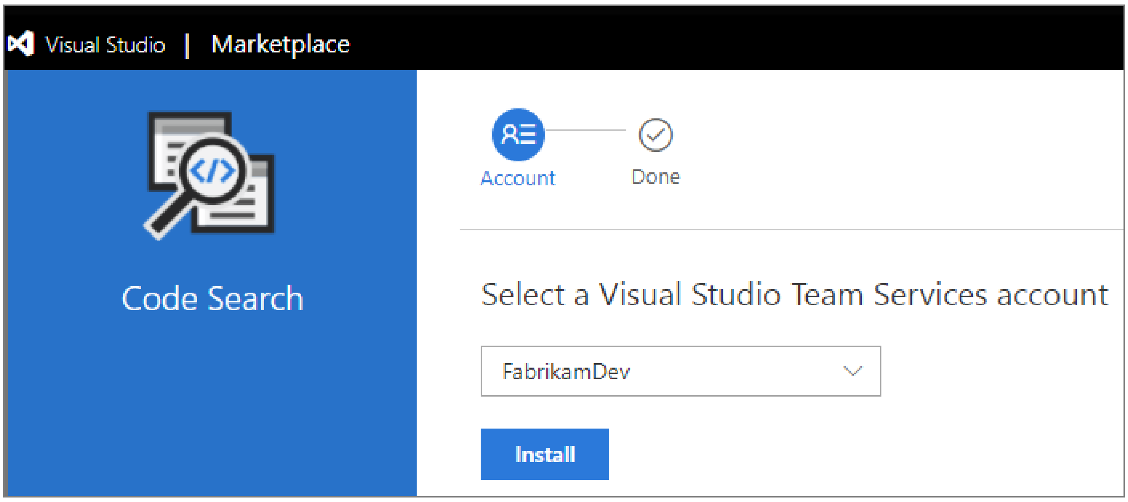


2. Find and select the extension that you want to install. For this Quickstart, you can choose **Code Search**.

3. Click *Get it free*.

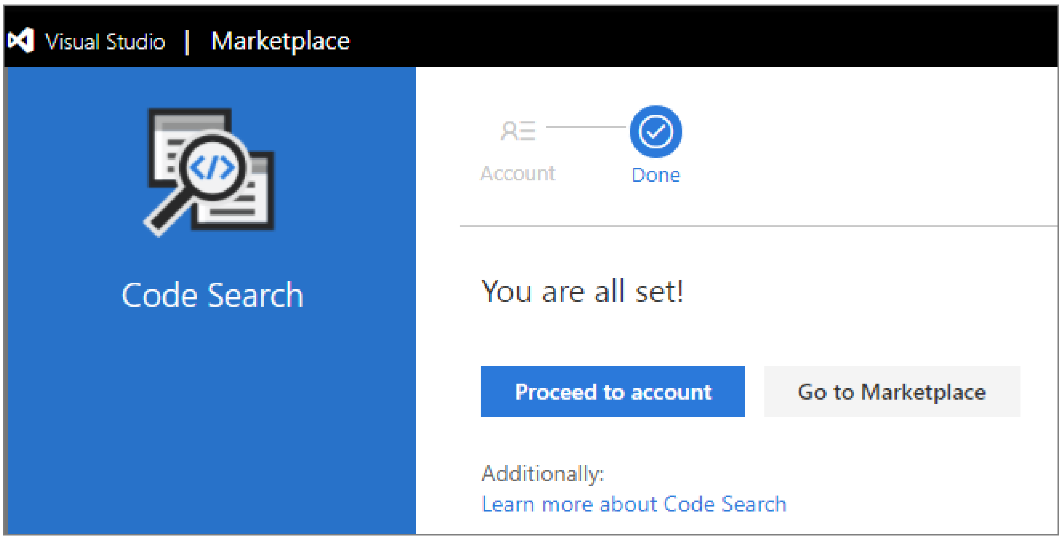


4. Now you're on the acquisition page, select your VSTS account and click *Install* to install this extension.



- Why don't I see any VSTS accounts?
- Why can't I install this extension?

5. Your extension is now installed! You can now go to your VSTS account to use your extension. Also, tell your team about this extension, so they can start using its capabilities too.



Next Steps

You can find answers to common problems on the troubleshooting page.

View permissions for yourself or others

8/9/2018 • 3 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

You can view your permissions or those defined for another which have been set in Visual Studio Team Services (VSTS) or Team Foundation Server (TFS). That way, if you don't have the permission to access a feature or function, you can request it from the right resource.

Permissions are set at the collection, project, and object level as described in About permissions and groups. So to view the permissions you have, you need to open the permissions at the object, project, or collection level.

Prerequisites

- You must have a team project to connect to. If you don't have a team project yet, create one.
- You must be a member of the Project Valid Users Group or Project Collection Valid Users Group to view permissions.

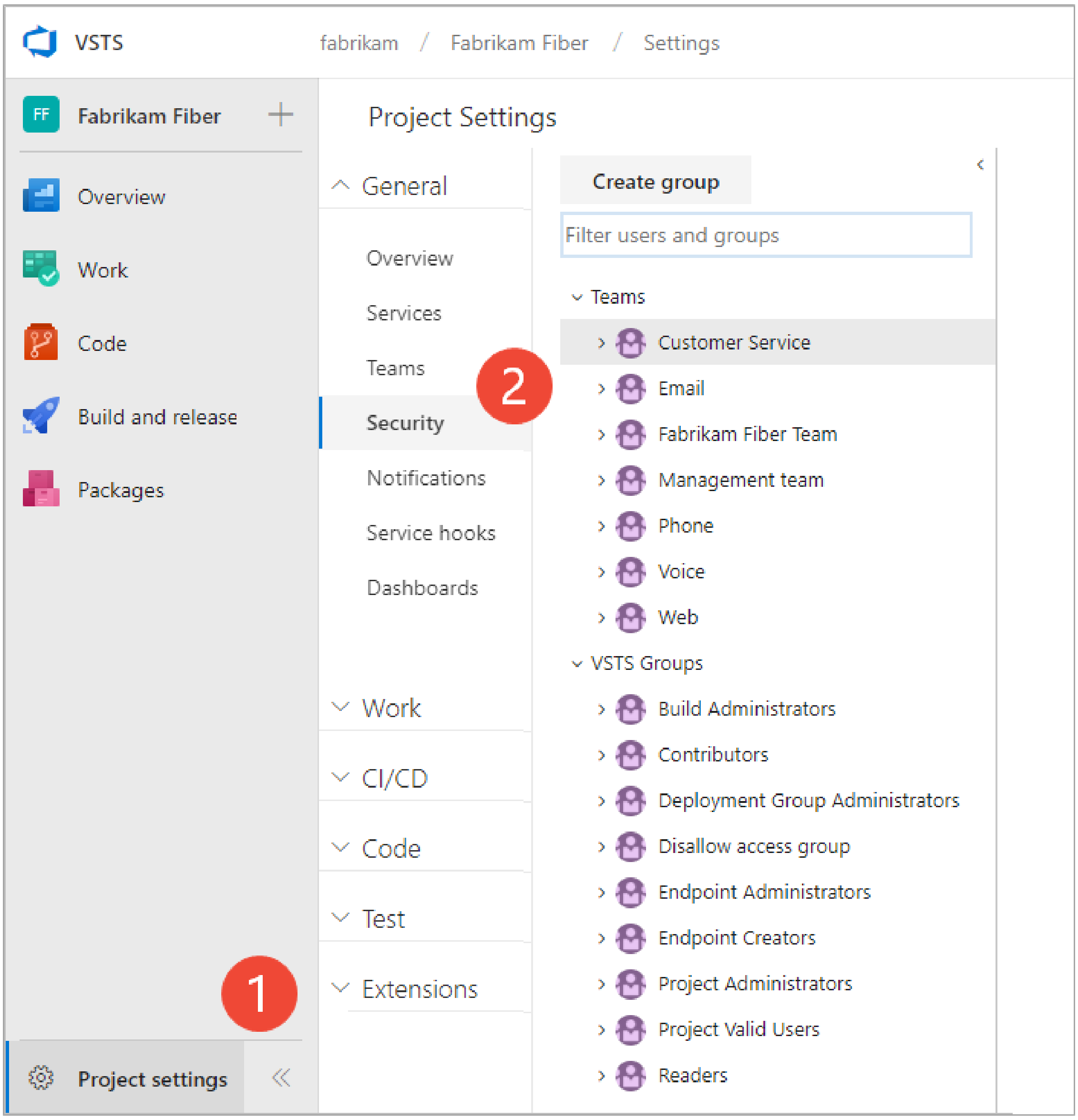
> **NOTE**
> This article shows how to view permissions assigned to a user as the project-level. However, the steps are similar whether you work from the Security dialog of an object or at the organization or collection-level.

View project-level permissions

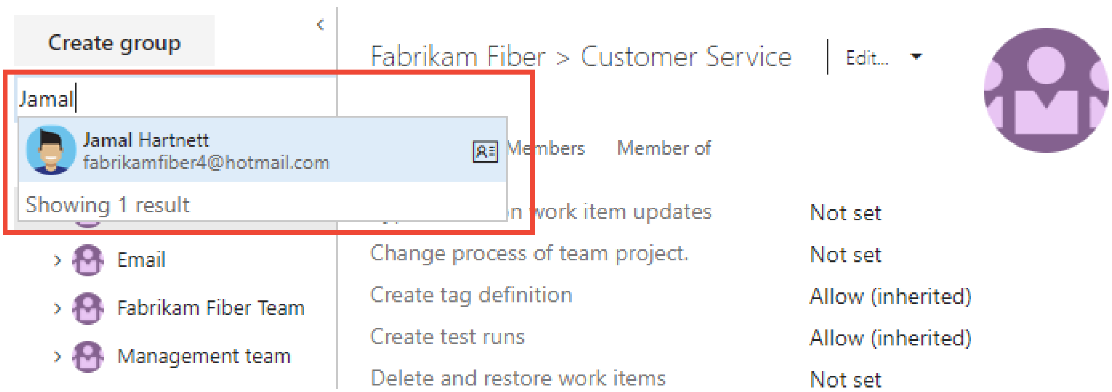
> **NOTE**
> Choose **Previous navigation** when you see a top-level blue bar. Choose **New navigation** if you see a vertical sidebar or if you enabled the **New Navigation** preview feature. The vertical sidebar, along with other navigational features, is enabled when the **New Navigation** preview feature has been enabled for the signed-in user or the VSTS organization. To learn how to use the web portal effectively, see Navigation Basics.
>
> For on-premises TFS users, choose **Previous Navigation** for guidance.

- New navigation
- Previous navigation

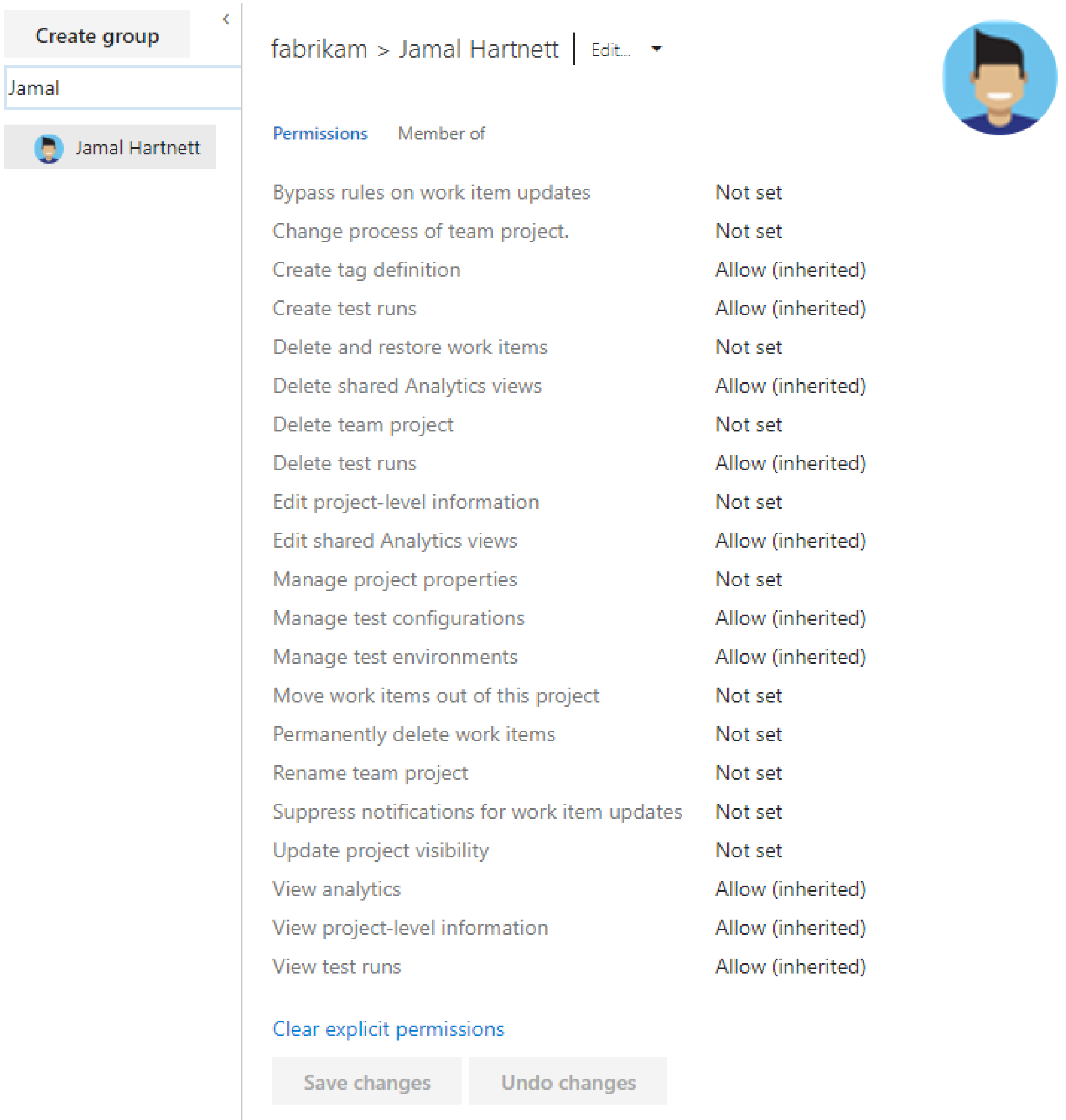
1. Choose **Project Settings** and then **Security**.



2. Begin typing the name into the *Filter users and groups* box. The system will automatically show the names that begin with the characters you type.

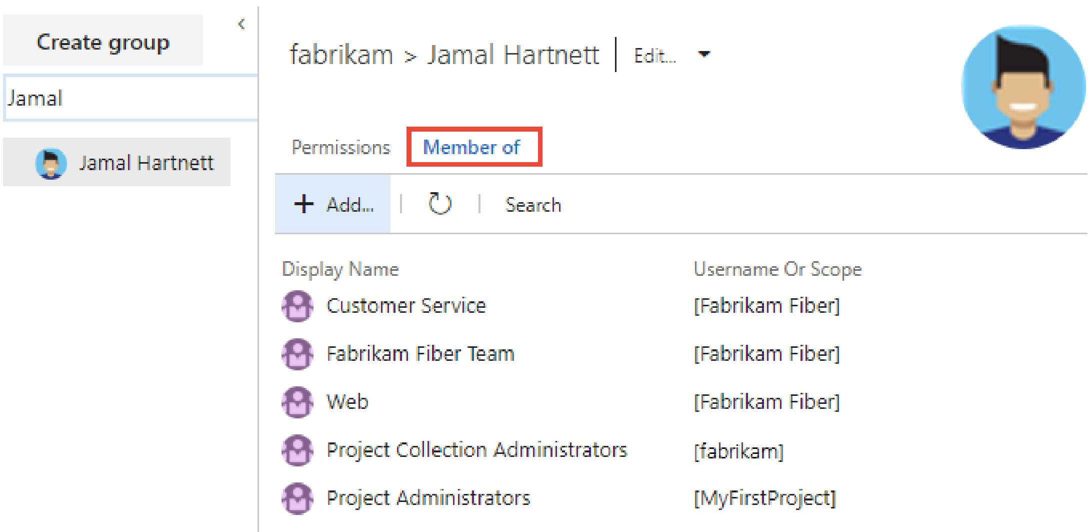


3. Choose the name you want. The project-level permissions you have set are based on the groups you belong to or those specifically set for your account.



4. Choose **Member of** to see which security groups the user belongs to.

   Here we see that *Jamal Hartnett* belongs to several teams as well as the Project Collection Administrators group.



New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

View collection-level permissions

1. Open the admin context for the organization or collection. Choose the gear settings icon, and then **Security**.

2. Follow steps 2 through 4 in the procedure outlined previously for view project-level permissions.

View object-level permissions

You can define the security or permissions for a number of objects. You access them from the context menu of the object.

From the web portal, open the Security dialog for the object whose permissions you want to set. For specific instructions, see the following articles:

WIKI & DASHBOARD PERMISSIONS	DEVOPS PERMISSIONS	AGILE/WORK TRACKING PERMISSIONS
- README & Wiki - Dashboards	- Git branch - Git repository - TFVC - Builds - Release pipeline security - Approvals and approvers	- Area and iteration paths - Work item query and folder - Plan permissions

Next steps

Look up the organization owner or a project administrator

Set personal or team favorites

8/10/2018 • 7 minutes to read

VSTS | TFS 2018 | TFS 2017.1

By favoriting an artifact or view, you can quickly access it. An artifact is a specific container, definition, or configurtion of a feature or tool—such as a project, repository, build pipeline, or query. As your code base, work tracking efforts, developer operations, and organization grows, you'll want to be able to quickly navigate to those artifacts of interest to you and your team. Setting favorites allows you to do just that.

You can set favorites for yourself or your team. Team favorites are a quick way for members of your team to quickly access shared resources of interest. You favorite an item for yourself by clicking the ☆ star icon. The favorited item will then show up easily from one or more directory lists. You set favorites for a team through the context menu for the artifact.

You can set favorites for the following artifacts from the indicated page.

SERVICE/AREA	ARTIFACTS	PAGE
Organization & projects	Projects and teams	**Projects** (personal favorites only)
Dashboard	Team dashboards	**Dashboards**
Code	Repositories	**Code>Branches** (personal favorites only)
Work, Agile tools for a team	Backlogs, boards, sprint backlogs, sprint taskboards	**Work>Queries** (requires enabling the **New Navigation** or **New Navigation**)
Work	Queries	**Work>Queries**
Work	Delivery plans	**Work>Plans** (personal favorites only, requires installation of the Delivery Plans extension)
Build and Release	Pipeline definitions	**Build & Release>Build** or **Release**
Test	Test plans	**Test>Test Plans** (personal favorites only)

SERVICE/AREA	ARTIFACTS	PAGE
Organization & projects	Projects and teams	**Projects** (personal favorites only)
Code	Repositories	**Code>Branches** (personal favorites only)
Work	Queries	**Work>Queries**

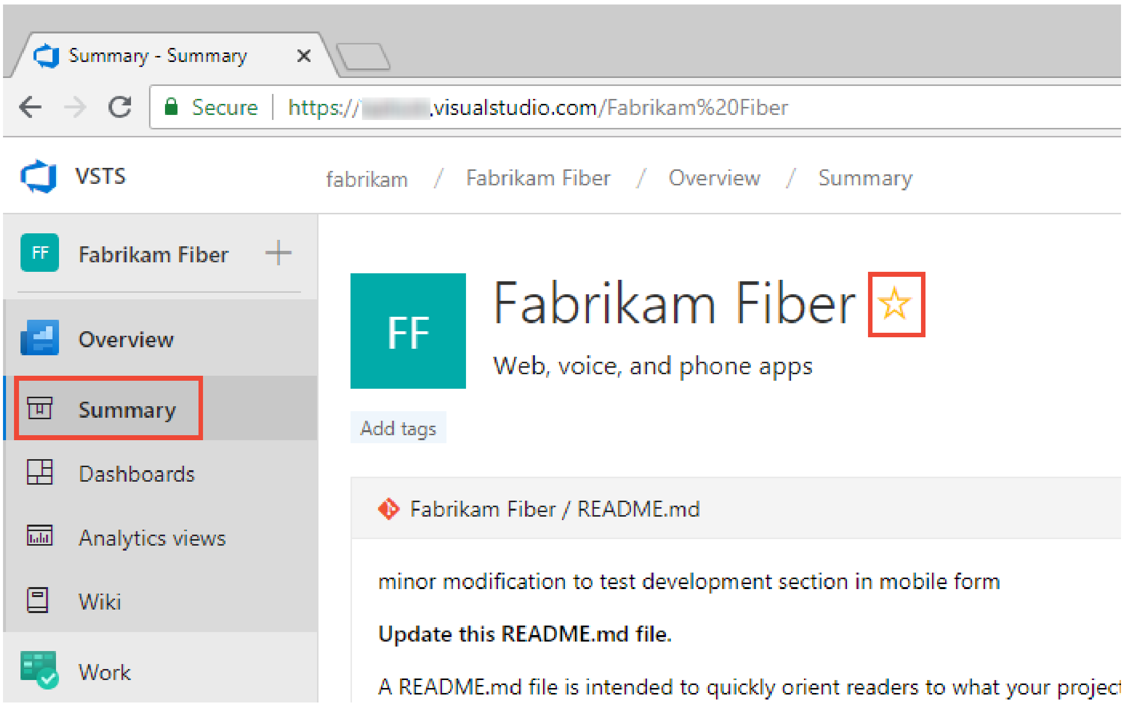
SERVICE/AREA	ARTIFACTS	PAGE
Work	Delivery plans	**Work>Plans** (personal favorites only, requires installation of the Delivery Plans extension)
Build and Release	Pipeline definitions	**Build & Release>Build** or **Release**
Test	Test plans	**Test>Test Plans** (personal favorites only)

Favorite a project or team

> **NOTE**
>
> Choose **Previous navigation** when you see a top-level blue bar. Choose **New navigation** if you see a vertical sidebar or if you enabled the **New Navigation** preview feature. The vertical sidebar, along with other navigational features, is enabled when the **New Navigation** preview feature has been enabled for the signed-in user or the VSTS organization. To learn how to use the web portal effectively, see Navigation Basics.
>
> For on-premises TFS users, choose **Previous Navigation** for guidance.

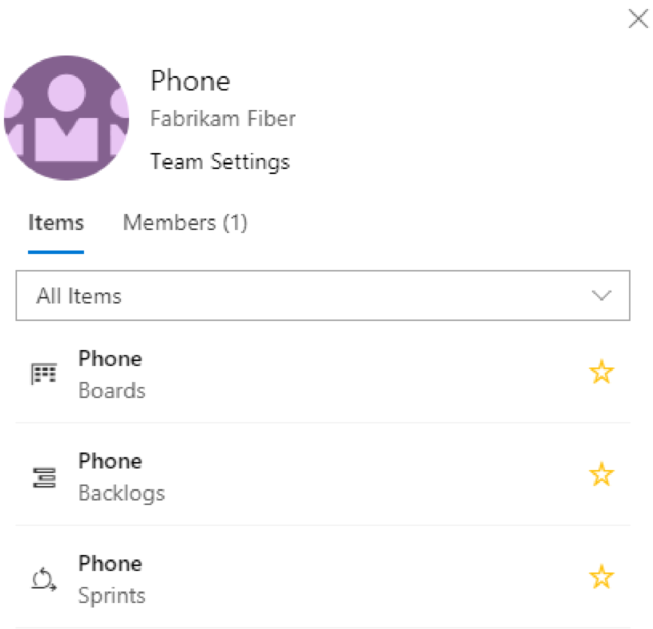
- New navigation
- Previous navigation

1. To favorite a project, open the project **Summary** page and choose the star icon.



2. To favorite a team artifact, open **Work>Backlogs** or **Work>Boards**. Select the team you want to favorite from the team selector and choose the star icon.

3. To favorite other team artifacts, choose the team icon, and then choose the star icon next to one of the listed artifacts.

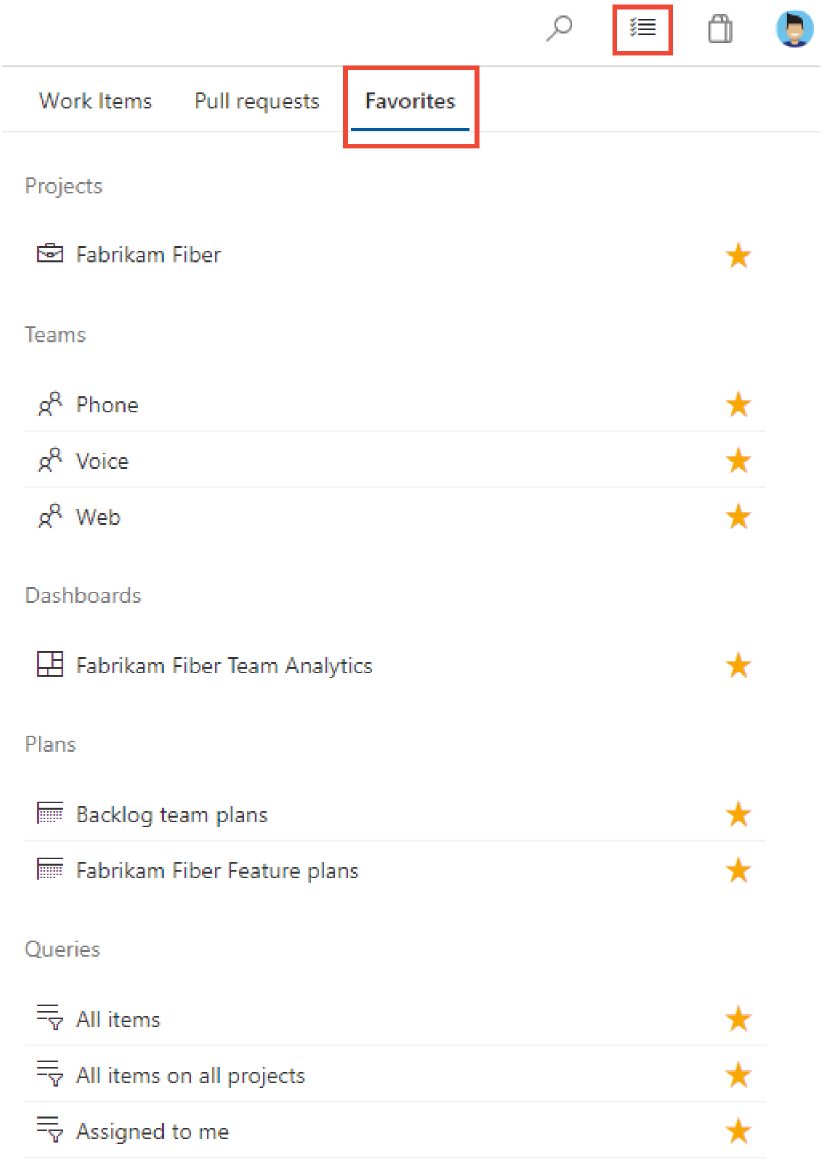


New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

View personal favorites

- New navigation
- Previous navigation

Access your favorites by choosing the inbox icon, and then choosing **Favorites**.



New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

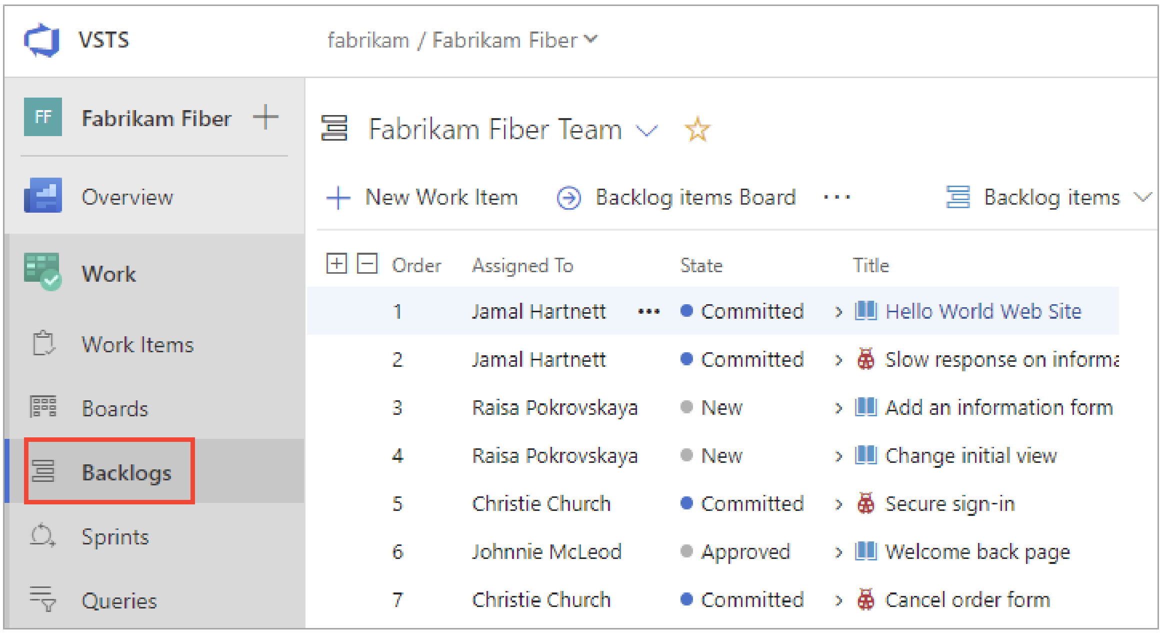
Favorite an Agile tool

- New navigation
- Previous navigation

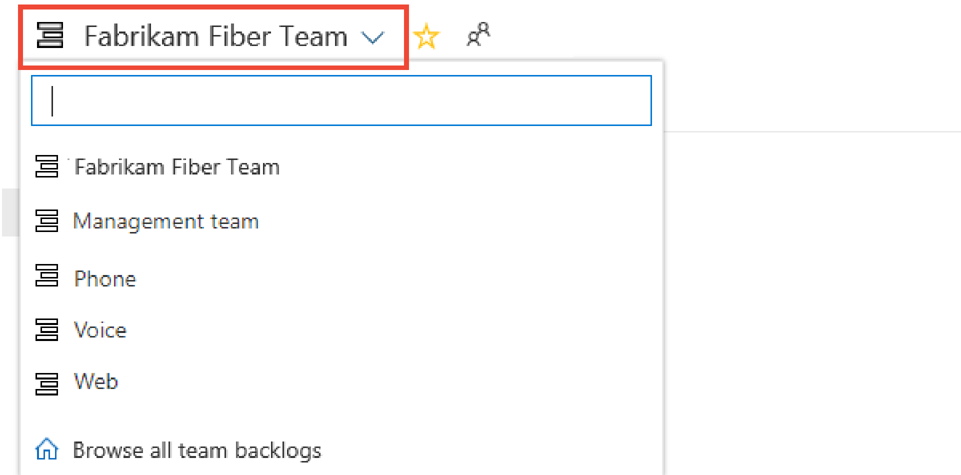
You can favorite several Agile tools for a team from a **Work** page.

1. From **Work**, choose the page of interest, such as **Boards**, **Backlogs**, or **Sprints**.

   For example, here we choose (1) **Work** and then (2) **Backlogs**.



To choose a specific team backlog, open the selector and select a different team or choose the **Browse all team backlogs** option. Or, you can enter a keyword in the search box to filter the list of team backlogs for the project.



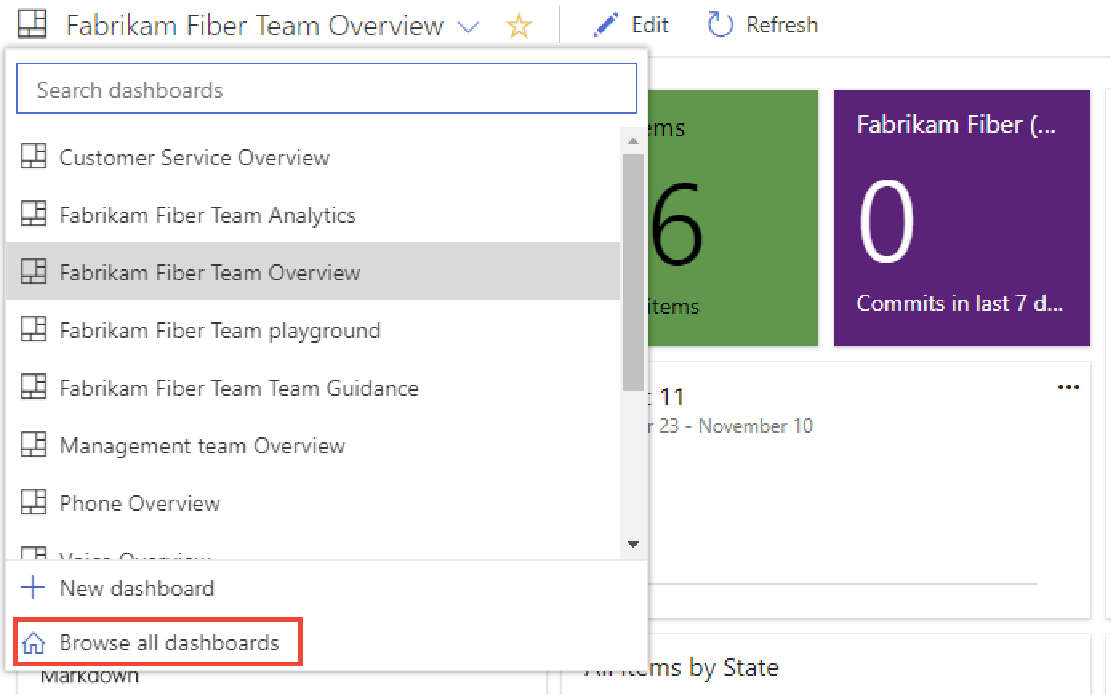
2. Choose the ☆ star icon to favorite a team backlog. Favorited artifacts (★ favorited icon) appear appear on your **Favorites** page and towards the top of the team backlog selector menu.

New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

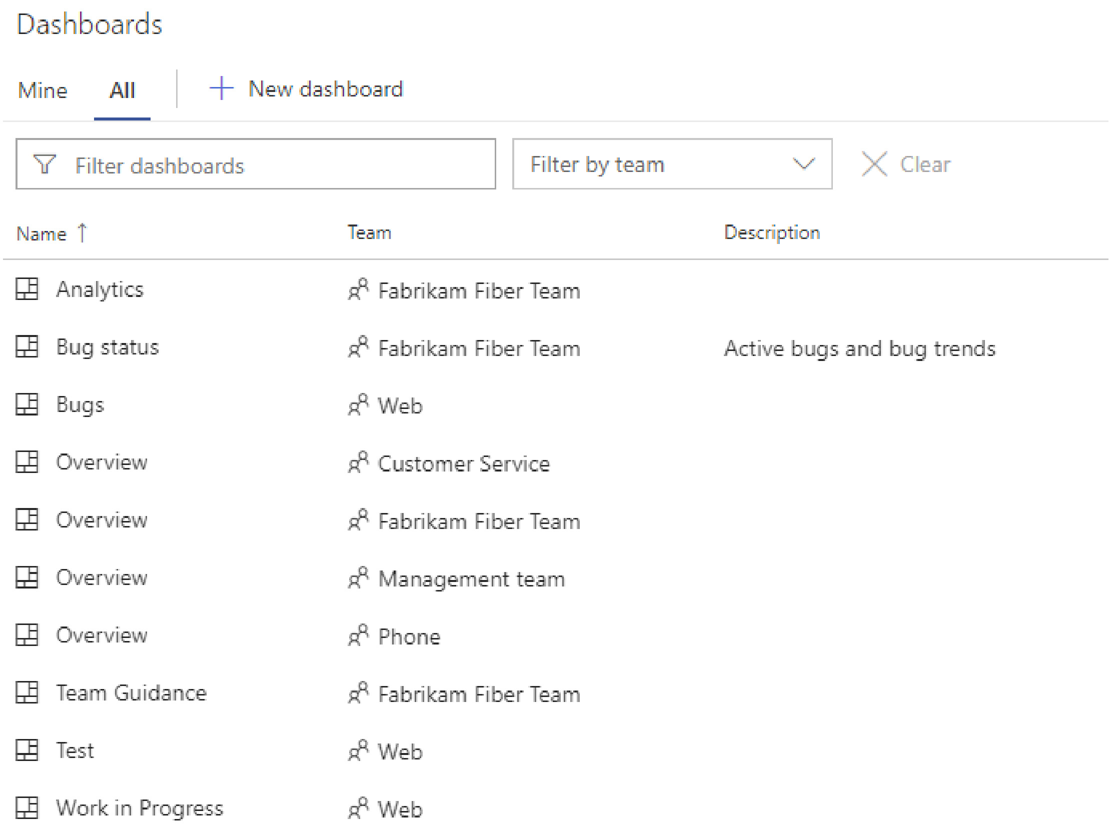
Favorite a dashboard

- New navigation
- Previous navigation

1. From **Overview>Dashboards**, open the selector and choose the **Browse all dashboards** option.



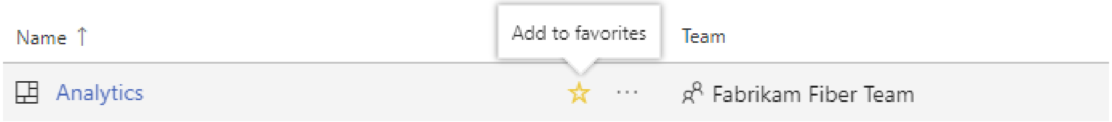
2. The **Mine** page shows your favorited dashboards, and all dashboards of teams that you belong to. The **All** page (shown below) lists all dashboards defined for the project in alphabetical order. You can filter the list by team or by keyword.



TIP

You can change the sort order of the list by choosing the column label.

3. To favorite a dashboard, hover over the dashboard and choose the ☆ .



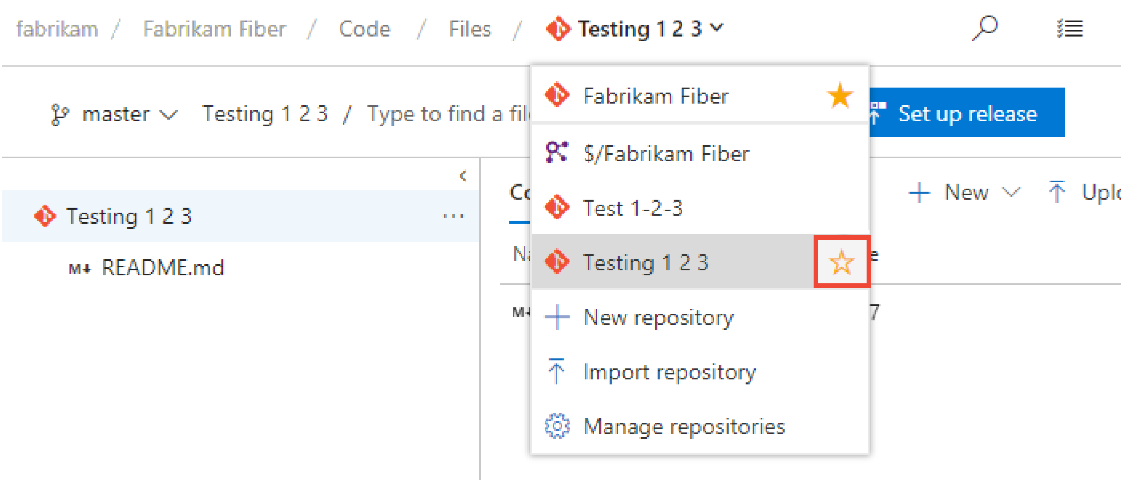
Favoriting a dashboard will cause it to appear on your **Favorites** page and towards the top in the **Dashboards** selection menu.

New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Favorite a repository

- New navigation
- Previous navigation

From any **Code** page, open the repository selector and choose the ☆ star icon for the repository you want to favorite.

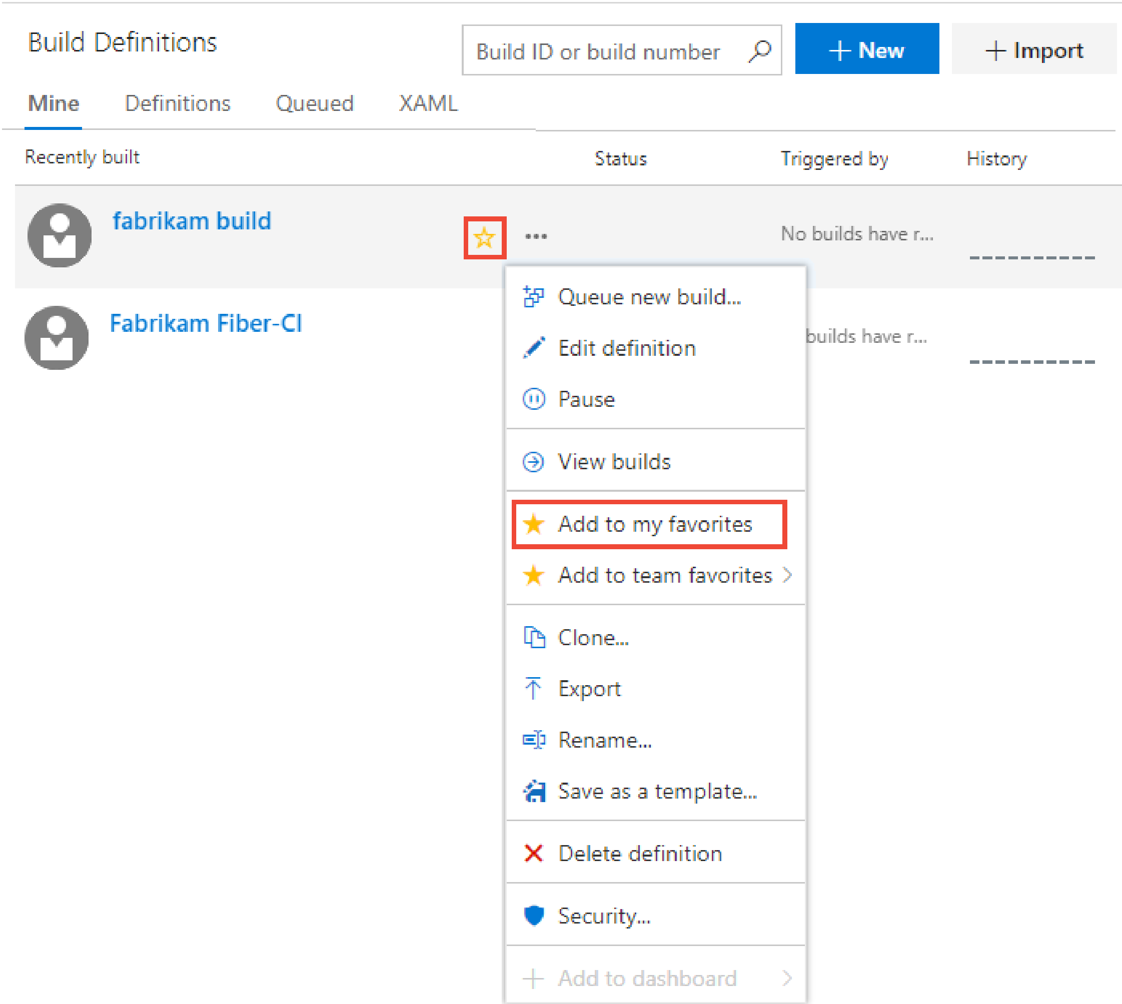


New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Favorite a build pipeline

- New navigation
- Previous navigation

Open **Build and Release>Builds** and choose either **Mine** or **Definitions** page. Choose the ☆ star icon next to the build definition you want to favorite. Or, open the context menu of the build definition, and then select **Add to my favorites** or **Add to team favorites**.



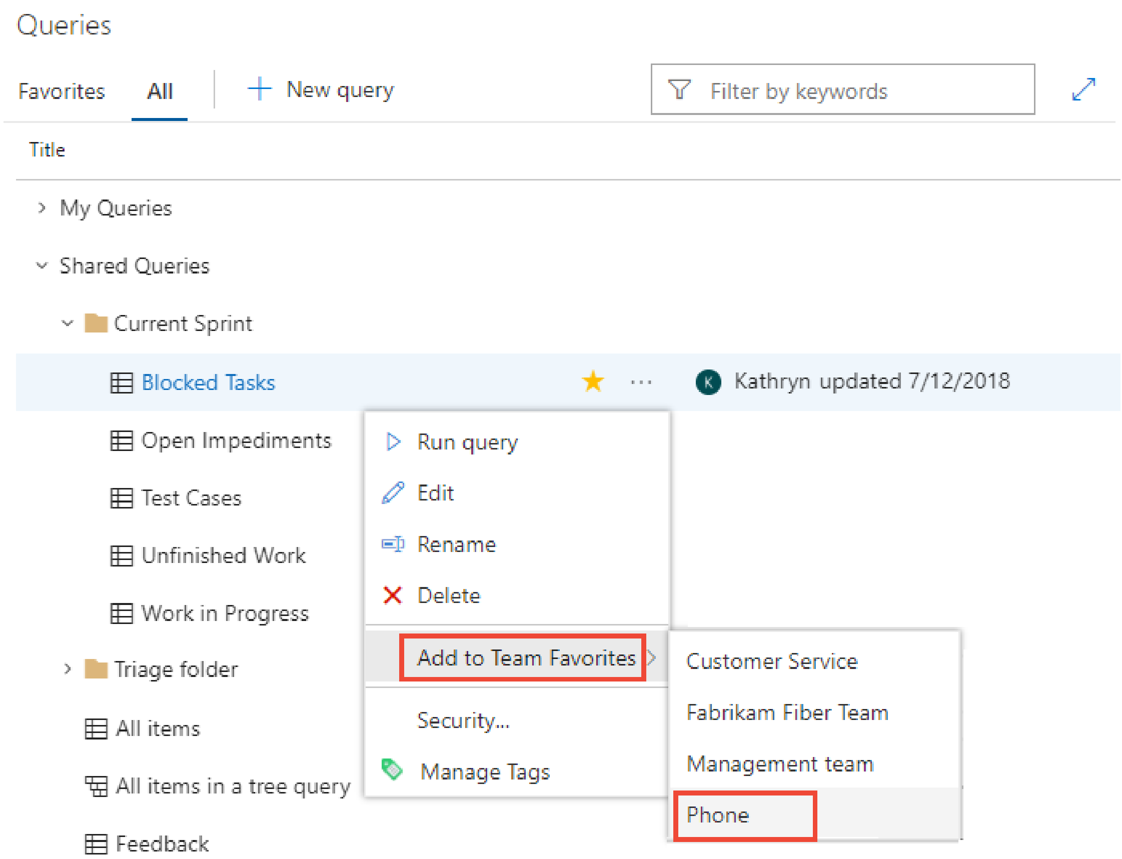
New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Favorite a shared query

- New navigation
- Previous navigation

Open **Work>Queries** and choose the **All** page. Expand a folder as needed. Choose the ☆ star icon next to the query you want to favorite.

Or, open the context menu of the query, and then select **Add to Team Favorites**, and then select from the list of teams.



You can also set a query as a personal favorite by opening the query and choosing the ☆ star icon.

New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Favorite a delivery plan

To mark a delivery plan as a favorite, open the **Work>Plans** page and choose the ☆ star icon next to the Delivery Plan.

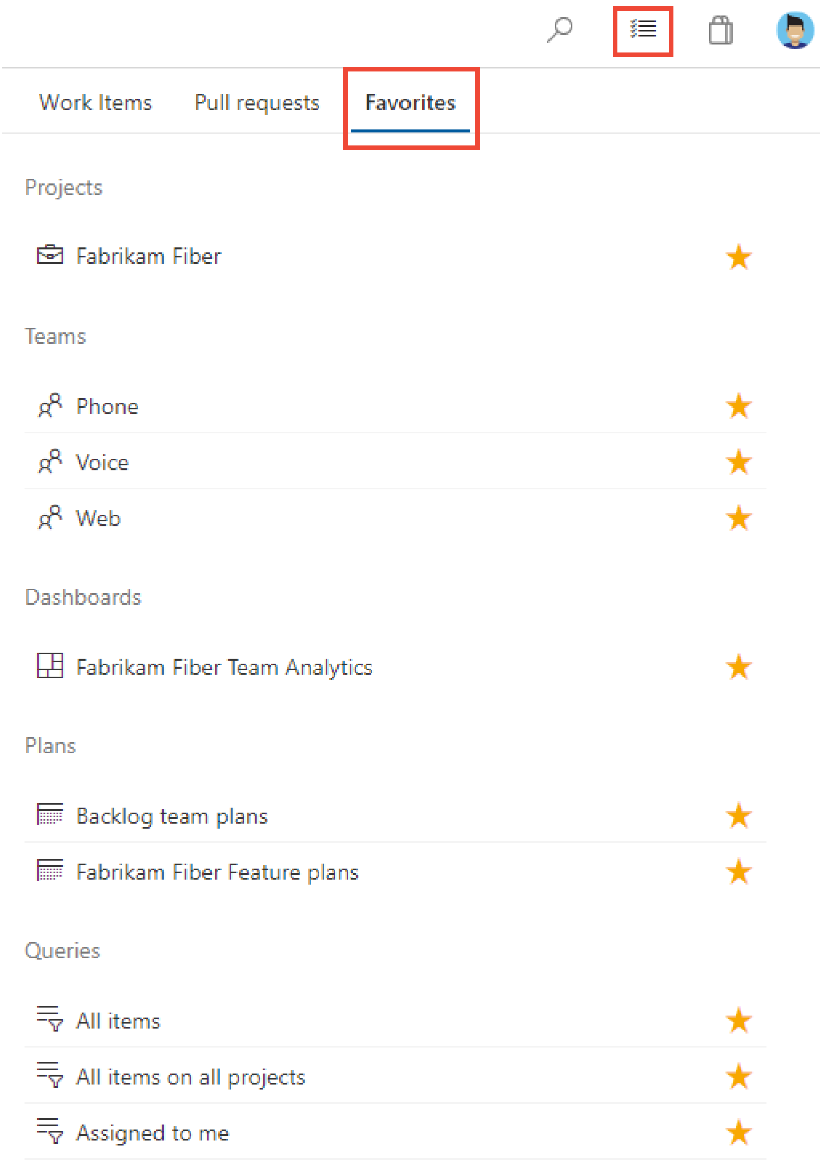
Favorite a test plan

To mark a test plan as a favorite, open the **Test>Test Plans** page and choose the ☆ star icon next to a test plan from the menu that shows All test plans.

Unfavorite an artifact

- New navigation
- Previous navigation

You can unfavorite an artifact from your **Favorites** page. Choose the inbox icon, and then choose **Favorites**. Choose the ★ favorited icon of a currently favorited artifact.



Similarly, you can unfavorite an artifact from the same page where you favorited it.

New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Related articles

- Manage personal notifications
- Set your preferences

Follow a work item or pull request

8/10/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017

> **NOTE**
>
> The Follow a work item feature is available from TFS 2017 and later versions. The Follow a pull request feature is available from TFS 2017.1 and later versions. To update your on-premises TFS, visit the Visual Studio downloads page for Team Foundation Server.

To get notified of changes made to a work item or a pull request, you can elect to follow them.

This topic shows you how to:

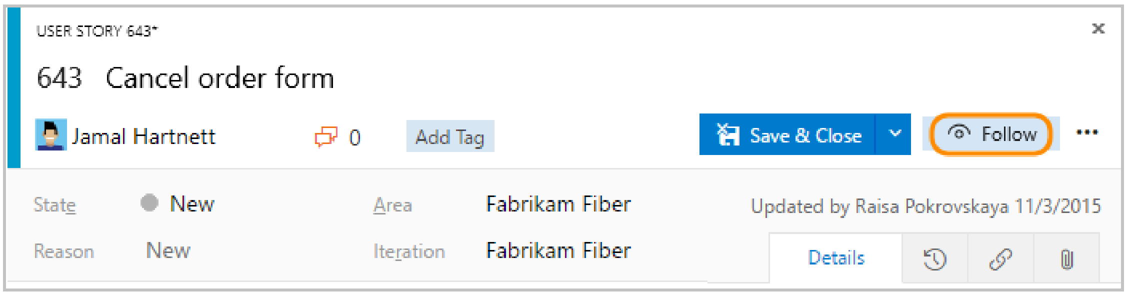
- Follow a work item
- Follow a pull request
- Manage work items that you're following

You must configure an SMTP sever in order for team members to receive notifications.

> **NOTE**
>
> The Follow a pull request feature is available from TFS 2017.1 and later versions. To update your on-premises TFS, visit the Visual Studio downloads page for Team Foundation Server.

Follow a work item

When you want to track the progress of a single work item, click the Follow icon. This signals the system to notify you when changes are made to the work item.



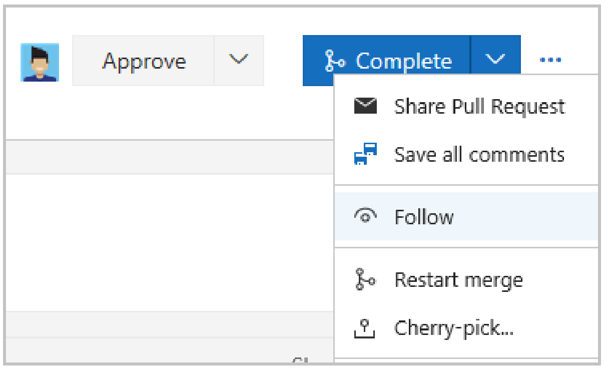
You'll only receive notifications when other members of your team modifies the work item, such as adding to the discussion, changing a field value, or adding an attachment.

Notifications are sent to your preferred email address, which you can change from your user profile

To stop following changes, click the Following icon.

Follow a pull request

To track the progress of a single pull request, click the Follow option from the context menu. This signals the system to notify you when changes are made to the PR.



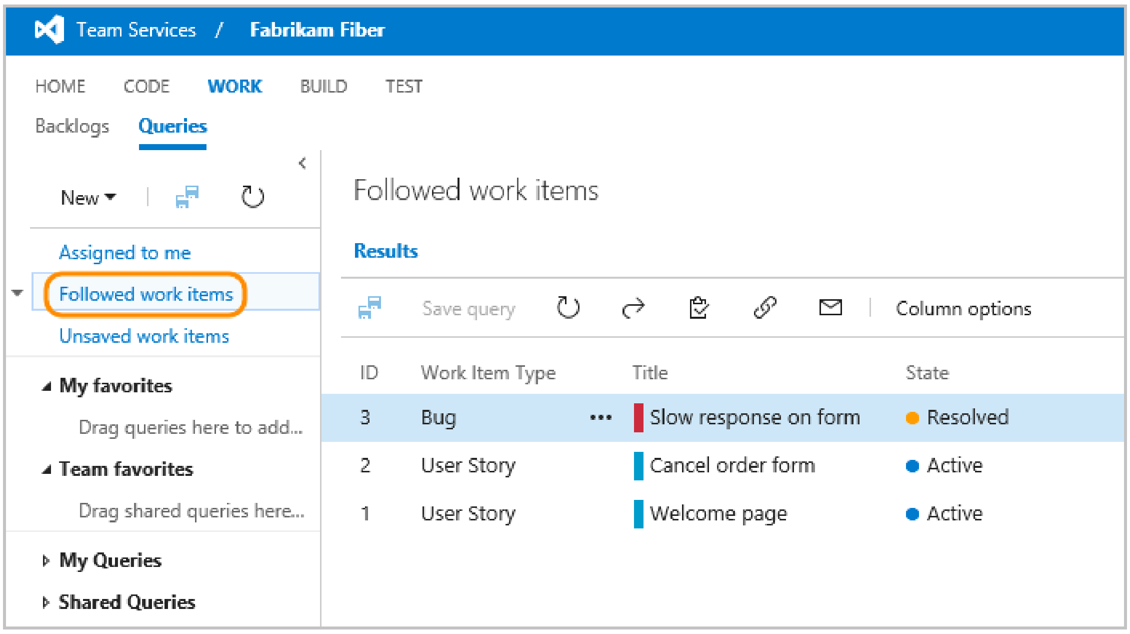
You'll only receive notifications when other members of your team modifies the PR, such as adding to the discussion or adding an attachment.

Notifications are sent to your preferred email address, which you can change from your user profile.

To stop following changes, open the PR context menu and click the Following icon.

Manage work items that you're following

You can review and manage all the work items you've selected to follow from the Followed work items page. You access this page from the **Work>Queries** page.



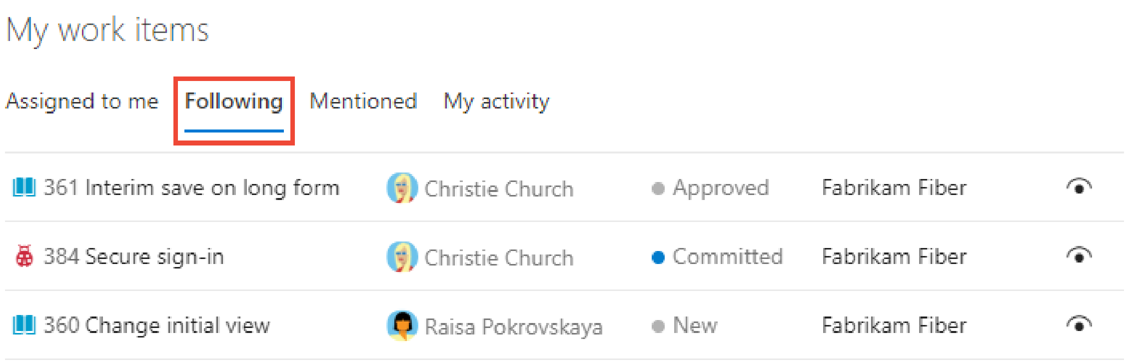
From this view, you can view all items you're following across all projects within the account. Also, you can perform similar actions supported with a query results view, such as:

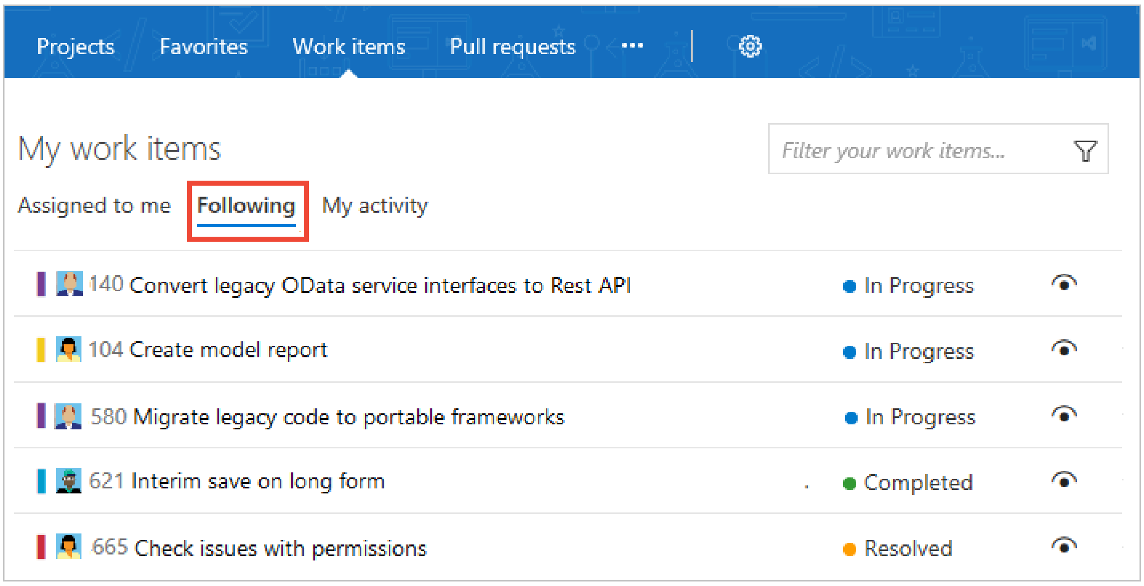
- Refresh the view
- Add or remove visible columns
- Sort the order of specific columns
- Filter results by text or tags
- Set work item pane
- Enter full screen mode.

However, this view isn't based on a query so you can't modify the parameters of the view.

You can also view and manage work that you're following from your home pages. To learn more, see Work across projects.

::: moniker range="vsts"





Â

Related articles

- Manage personal notifications
- Set team notifications
- View and update work items via the mobile work item form

Q: Can I add someone else to follow a work item or PR?

A: You can't add another team member to follow a work item or pull request at this time. You can subscribe them to get notified based on select criteria, such as when a work item is create or modified, or a pull request is created. For details, see Manage team notifications.

::: moniker-end

Get started as a Stakeholder

8/10/2018 • 8 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

With Stakeholder access, you can add and modify work items, approve releases, and view dashboards. You can check project status and provide direction, feedback, feature ideas, and business alignment to a team.

Use this topic to learn:

- How to sign-in to an account
- How to add a work item
- How to view the product backlog and add new work to it
- How to view work in progress on the Kanban board
- Find work assigned to you, or query for other work items
- Understand the set of features you have access to

First time signing in

1. Click the link provided in the email invitation you should have received. Or, open a browser window and enter the URL for the web portal.

   `http://AccountName.visualstudio.com/DefaultCollection/ProjectName`

   `http://ServerName:8080/tfs/DefaultCollection/ProjectName`

   For example, to connect to the server named *FabrikamPrime* and project named *Contoso*, type `http://FabrikamPrime:8080/tfs/DefaultCollection/Contoso`.

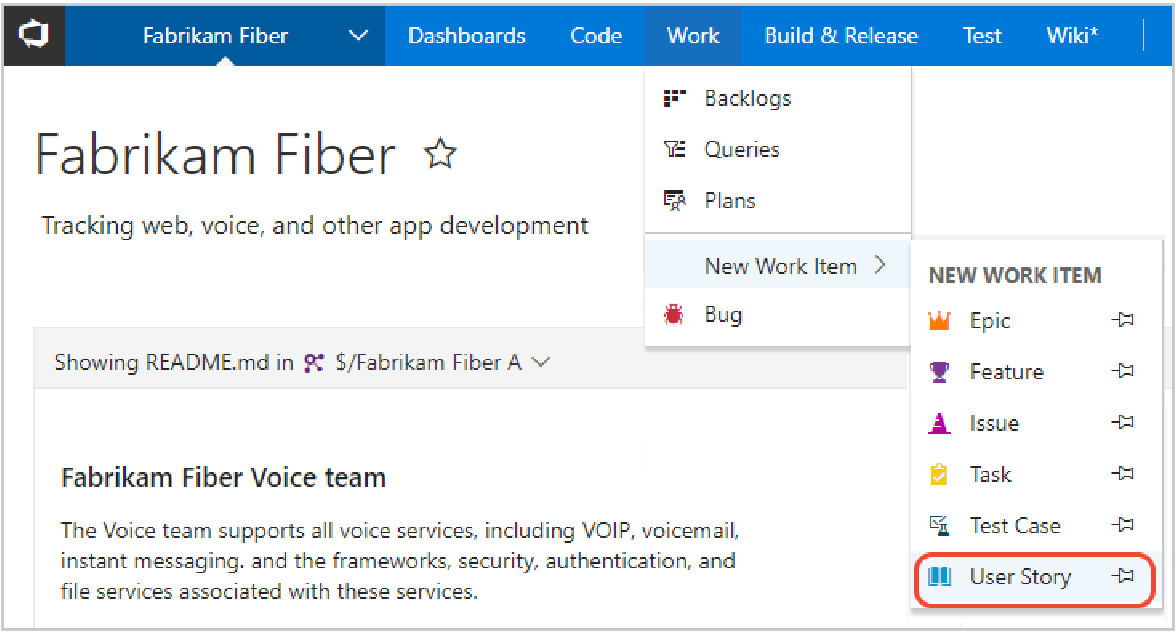
2. Enter your credentials. If you aren't able to sign in, ask the account owner or project administrator to add you as a member of the team project with Stakeholder access.

Add a work item

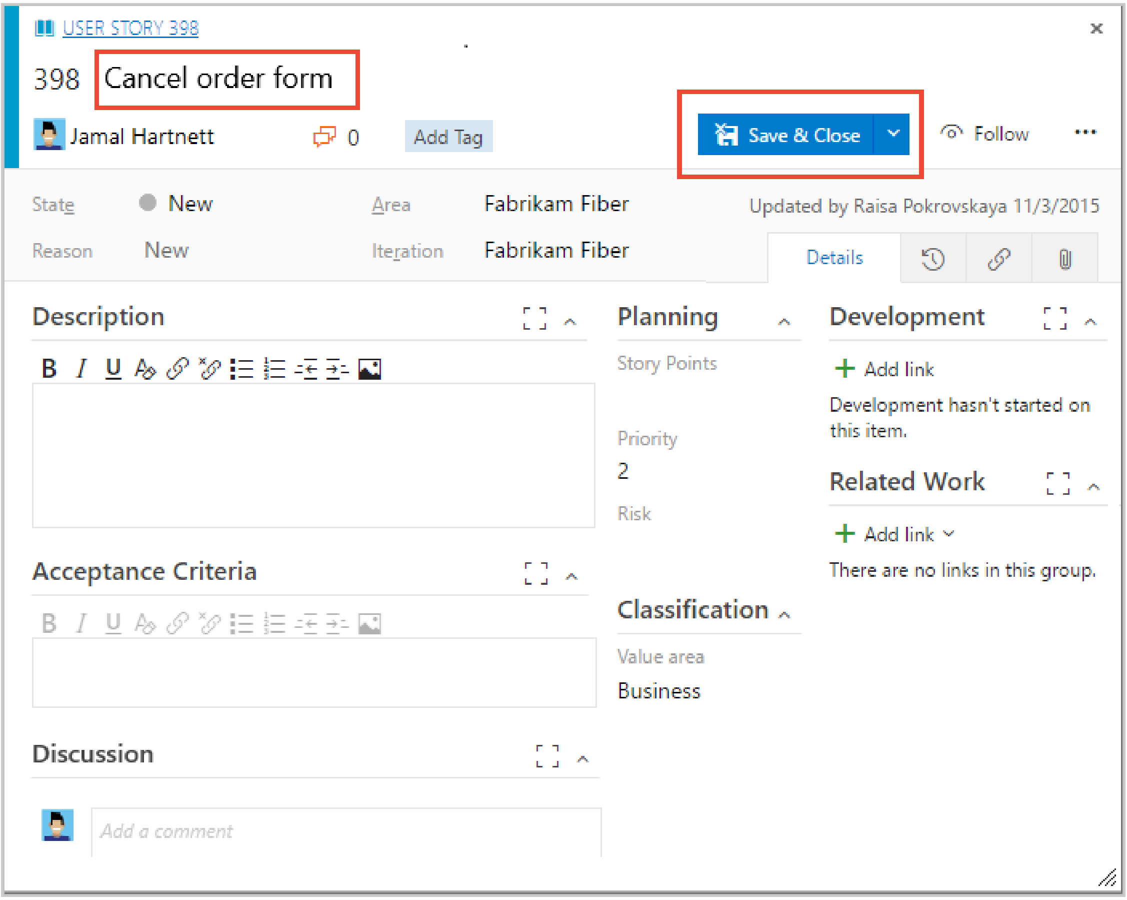
You might see different work item types in your view based on the process selected for your team project: Scrum, Agile, or CMMI.

> **TIP**
>
> Use the Work Items page which supports personalized views of work items assigned to you or ones you've recently viewed or updated. Also, you can add work items from this page.

From the Work hub, choose a work item, for example User Story, from the New Work Item list of options. Click the pin icon to have it show up within the Work hub drop down menu.



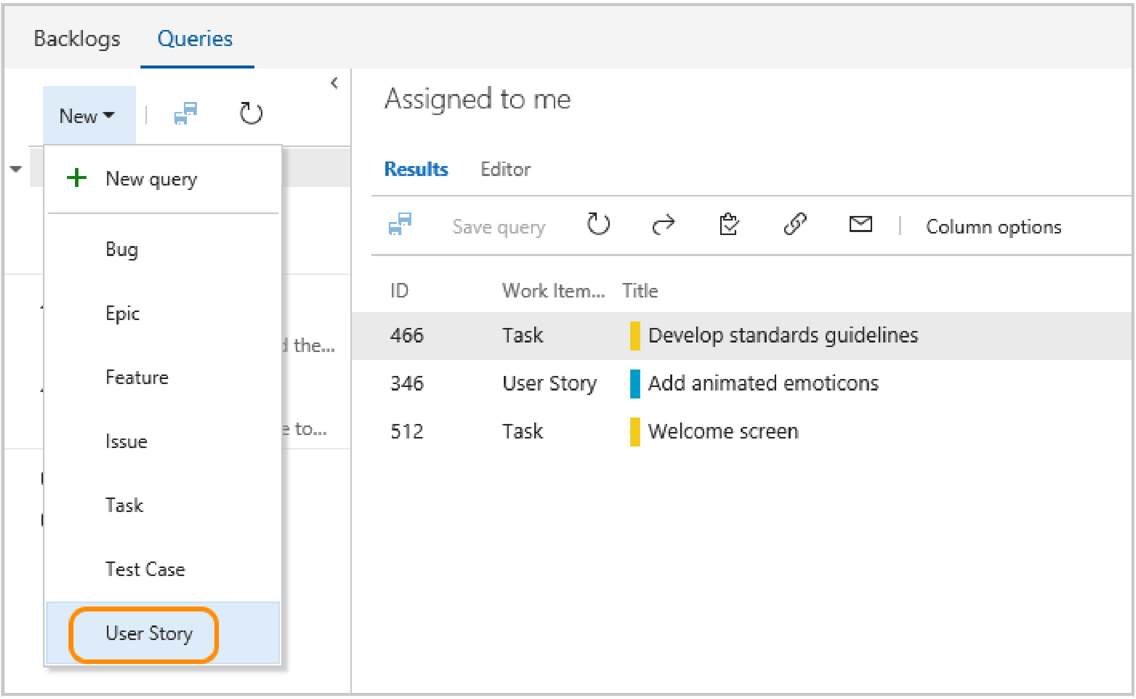
Enter a title and then save the work item. Before you can change the State from its initial default, you must save it. For descriptions of each field, see Work item field index.



NOTE

A caution icon on a tab indicates values that violate validation rules. You must correct information on that tab in order to save the work item.

From the Queries page, choose a work item from the New drop down menu.



Enter a title and then save the work item. Before you can change the State from its initial default, you must save it.

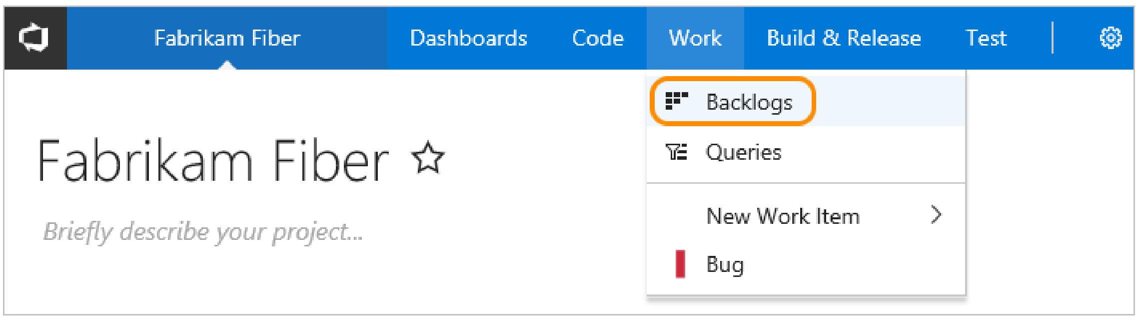
You can add existing tags to any work item to support filter backlogs and queries.

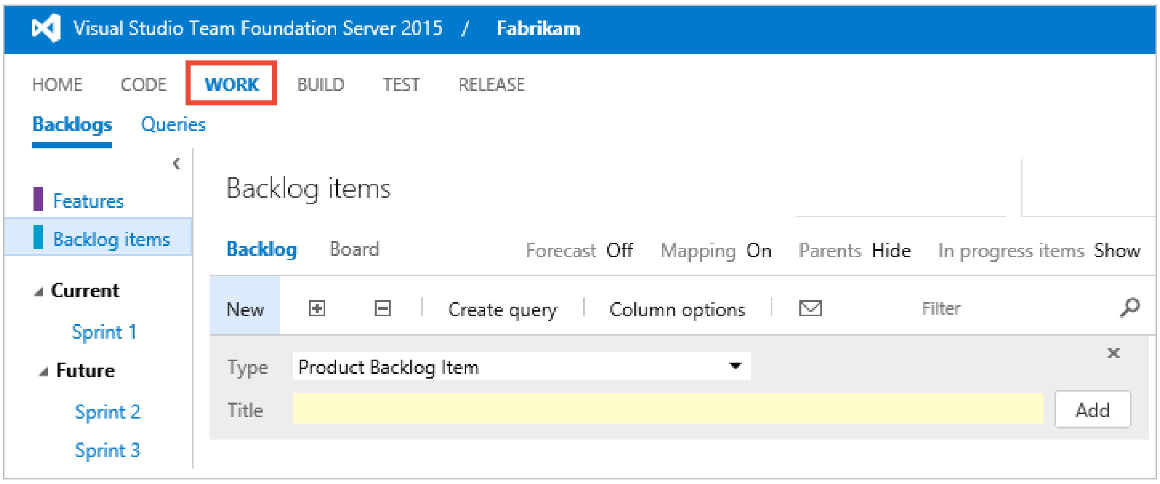
Work items you add are automatically scoped to your team's area and iteration paths. To change the team context, see Switch team project or team focus.

> **NOTE**
>
> Depending on the process chosen when the team project was created—Scrum, Agile, or CMMI—the types of work items you can create will differ. For example, backlog items may be called product backlog items (Scrum), user stories (Agile), or requirements (CMMI). All three are similar: they describe the customer value to deliver and the work to be performed.
>
> For an overview of all three processes, see Choose a process.

Check the backlog or add new work

From your web browser, open your product backlog from the **Work** hub, **Backlogs** page. Work appears in the backlog in priority order.



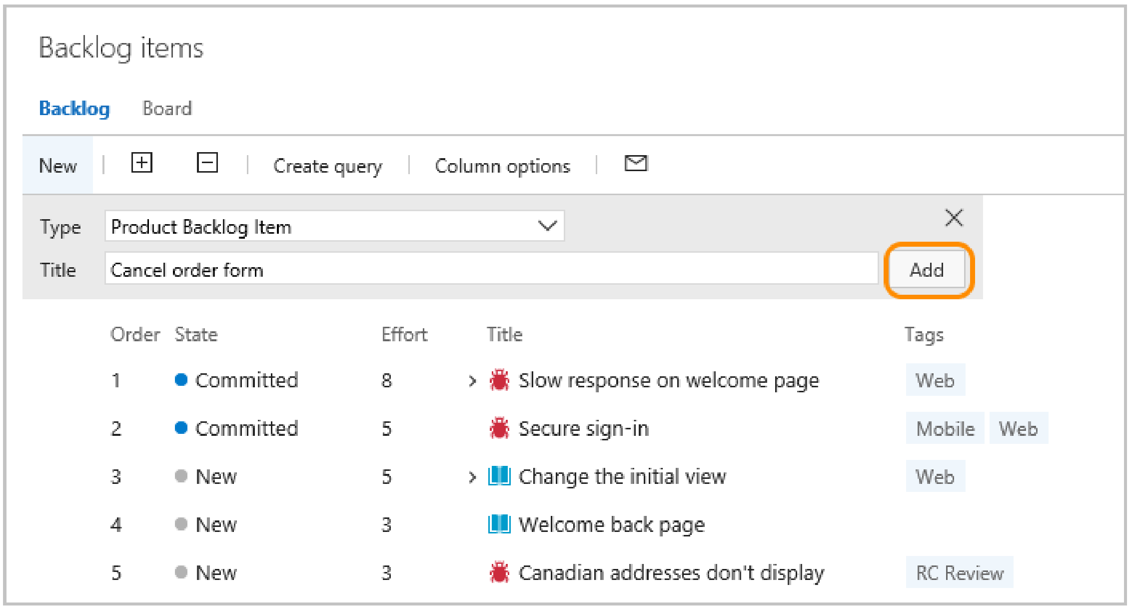


TIP

If you don't see the team or team project you want, click the project icon to browse all team projects and teams.

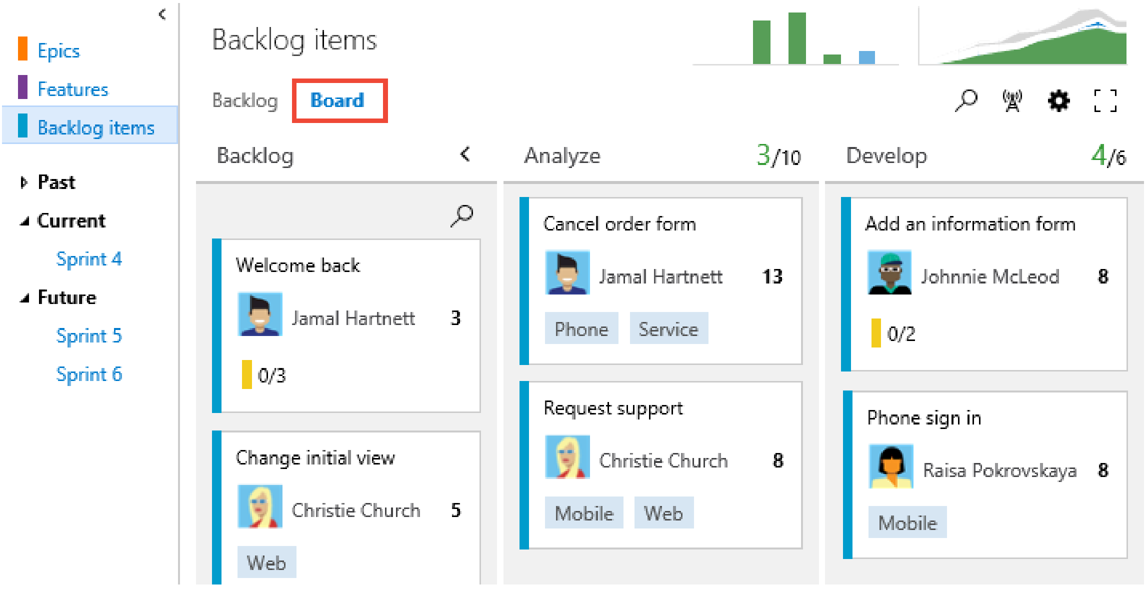
To view or edit a work item, select it and choose Enter.

To add a new item, select the type and then name it. Your items are added to the bottom of the list.



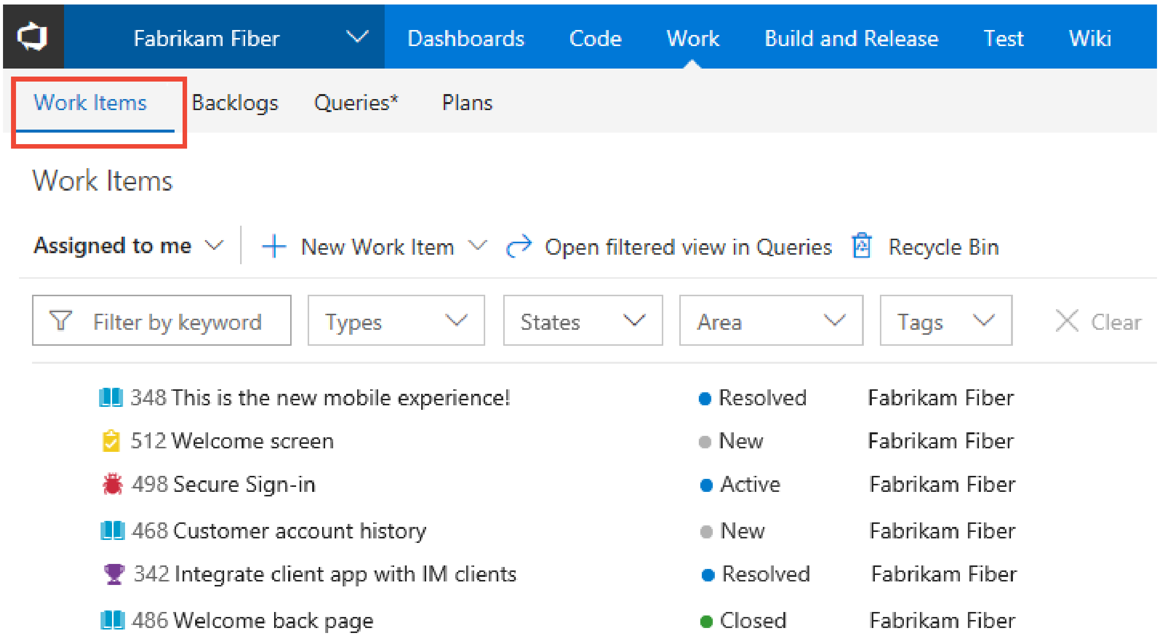
Check work in progress

To view the team's work status, click **Board** to open the Kanban board. Click the title of an item to open or edit it.

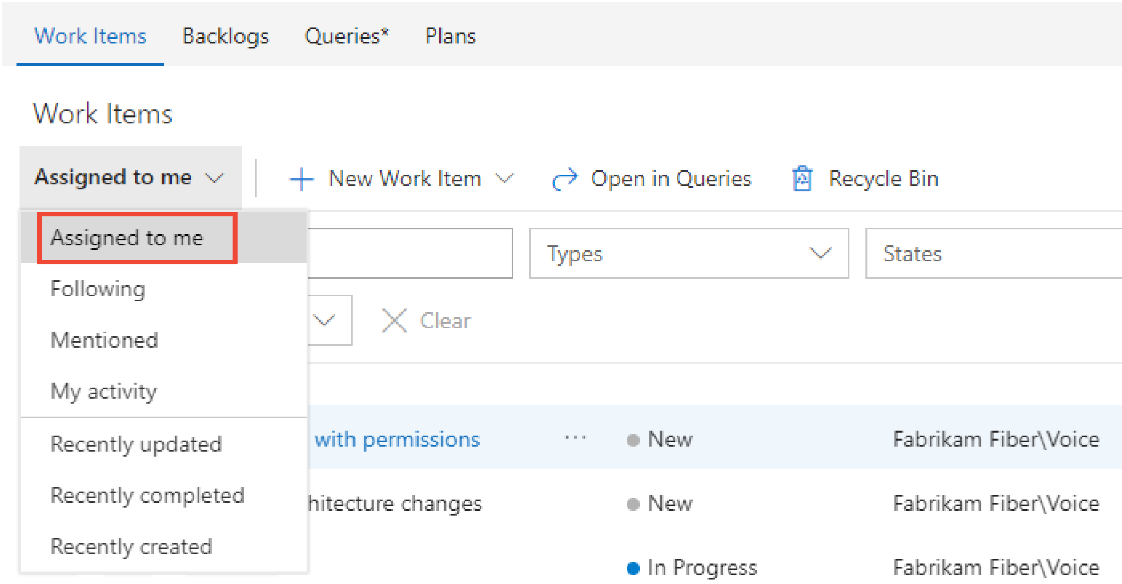


Find work assigned to you, or query for other work items

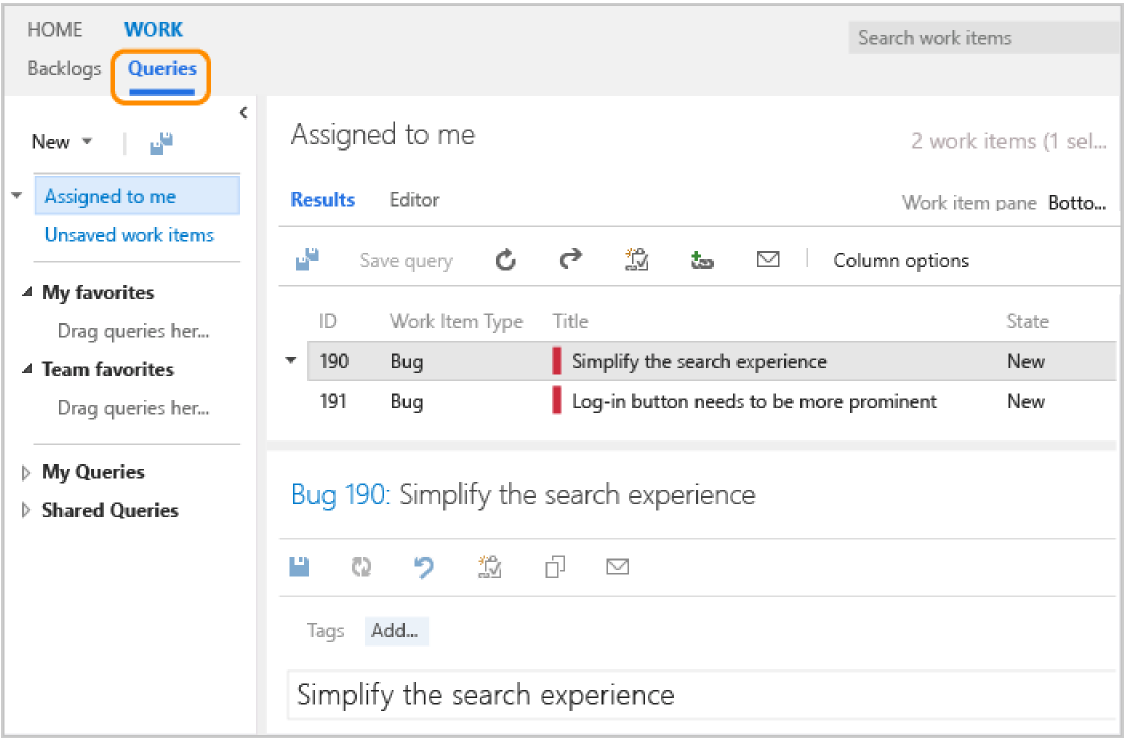
Click the **Work** hub, and then the **Work Items** page.



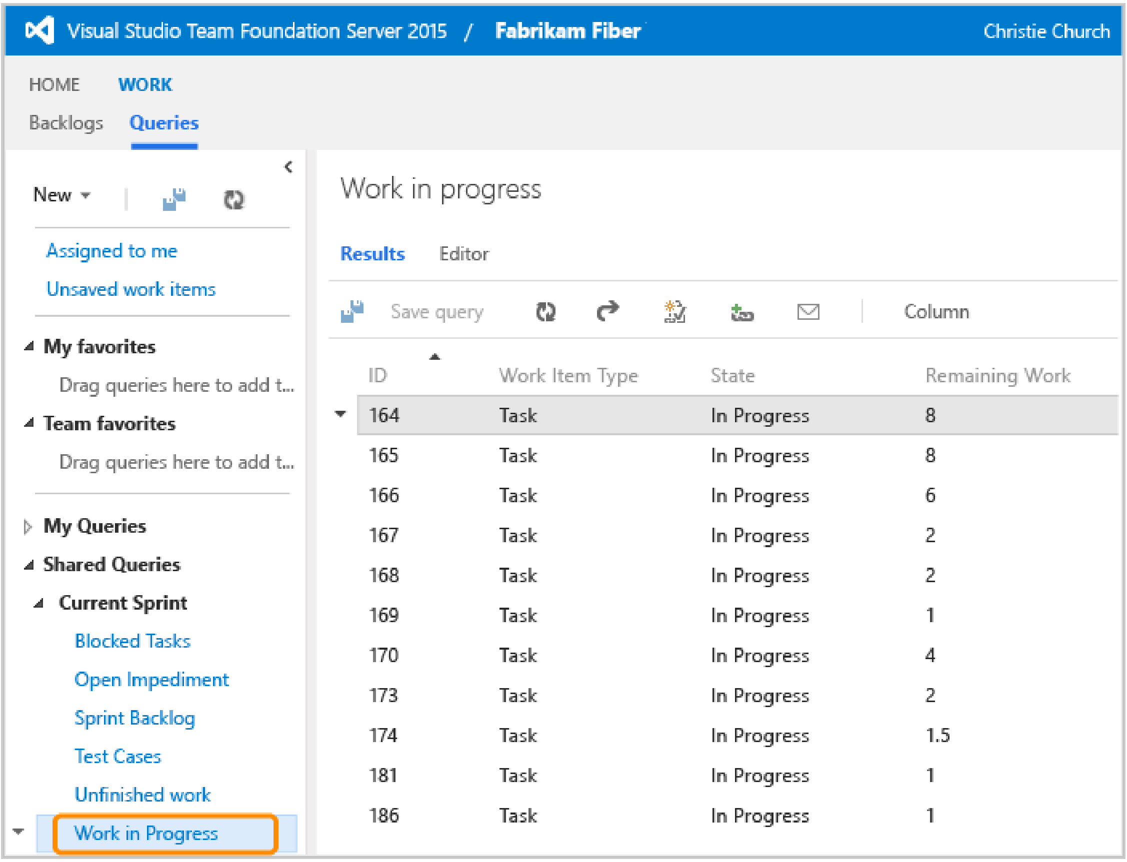
Choose **Assigned to me** from the drop-down menu, you can focus on relevant items inside a team project using one of the seven pivots as described next. Additionally, you can filter and sort each pivot view. For details, see View and add work items using the Work Items page.



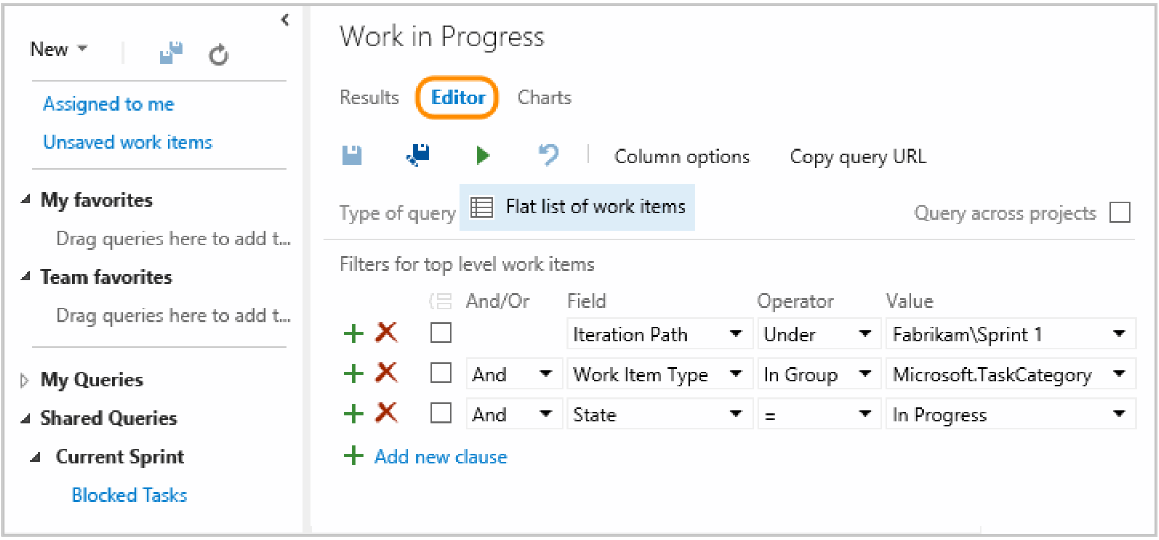
Open the Queries page to see the list of work items assigned to you.



Or, open any of the queries defined in the Shared Queries folder.



And, you can create new queries or edit existing queries and save them under My Queries folder.



Stakeholder feature access

The following features are available to Stakeholders from the web portal.

WORK	ACCOUNT, DASHBOARDS, WIKI, AND NOTIFICATIONS
- View, create, and modify work items [1] - View, add, and modify items on backlogs [2] - View, and modify items on sprint backlogs [2] - View, and modify items on the task board [2, 3] - View, and modify items (Kanban) [2, 3] - Add tasks to the checklist (Kanban) [5, 3] - Follow changes made to work items - View the cumulative flow diagram - View, create, and save queries [4] - Submit, view, and change feedback responses - Change work item type	- Navigate to teams, team projects, and more - View project welcome pages [6] - View team dashboards - Manage personal notifications - Invite users and assign licenses [7] - View wiki pages [8] **Build & Release** - View releases [5] - Approve a release

Notes:

1. Stakeholders can assign existing tags to work items, but not create new tags.
2. Stakeholders cannot change the backlog priority order (all items are added at the end of the backlog), assign items to an iteration using drag and drop, use the mapping pane or forecasting.
3. Stakeholders cannot move cards on the board to update status, set the values of fields shown on cards, or set or view team capacity.
4. Stakeholders can save queries under My Queries but cannot save under Shared Queries.
5. Stakeholders can only view and approve releases.
6. Stakeholders cannot view markdown README files defined for repositories.
7. In order to add users and assign licenses, stakeholders must be added to the Project Collection Administrators group. To learn more, see Manage users and access.
8. Stakeholders have read-only permissions to wiki pages. These permissions can't be changed.

The following features are available to Stakeholders from the web portal.

WORK	ACCOUNT, DASHBOARDS, WIKI, AND NOTIFICATIONS
- View, create, and modify work items [1] - View, add, and modify items on backlogs [2] - View, and modify items on sprint backlogs [2] - View, and modify items on the task board [2, 3] - View, and modify items (Kanban) [2, 3] - Add tasks to the checklist (Kanban) [5, 3] - Follow changes made to work items - View the cumulative flow diagram - View, create, and save queries [4] - Submit, view, and change feedback responses - Change work item type	- Navigate to teams, team projects, and more - View project welcome pages [6] - View team dashboards - Manage personal notifications - View wiki pages [7] **Build & Release** - View releases [5] - Approve a release

Notes:

1. Stakeholders can assign existing tags to work items, but not create new tags.
2. Stakeholders cannot change the backlog priority order (all items are added at the end of the backlog), assign items to an iteration using drag and drop, use the mapping pane or forecasting.
3. Stakeholders cannot move cards on the board to update status, set the values of fields shown on cards, or set or view team capacity.
4. Stakeholders can save queries under My Queries but cannot save under Shared Queries.
5. Stakeholders can only view and approve releases.
6. Stakeholders cannot view markdown README files defined for repositories.
7. Stakeholders have read-only permissions to wiki pages. These permissions can't be changed.

The following features are available to Stakeholders from the web portal of the listed TFS or later version. Those not annotated are available from all versions. To determine your platform or TFS version, see Platform and version support.

WORK	ACCOUNT, DASHBOARDS, AND NOTIFICATIONS
- View, create, and modify work items [1] - View, add, and modify items on backlogs [2] - View, and modify items on sprint backlogs [2] - View, and modify items on the task board [2, 3] - View, and modify items (Kanban) [2, 3] - Add tasks to the checklist (Kanban) [2, 3] (TFS 2015.1) - Follow changes made to work items (TFS 2017) - View the cumulative flow diagram - View, create, and save queries [4] - Submit, view, and change feedback responses	- Navigate to teams, team projects, and more (TFS 2017) - View project welcome pages [6] (TFS 2017) - View team dashboards (TFS 2015) - Manage personal notifications (TFS 2017) - Set personal alerts for changes to work items (TFS 2013, 2015) - **Build & Release** - View releases [5] (TFS 2015.2) - Approve a release [5] (TFS 2015.2)

Notes:

1. Stakeholders can assign existing tags to work items, but not create new tags.
2. Stakeholders cannot change the backlog priority order (all items are added at the end of the backlog), assign items to an iteration using drag and drop, use the mapping pane or forecasting.
3. Stakeholders cannot move cards on the board to update status, set the values of fields shown on cards, or set or view team capacity.
4. Stakeholders can save queries under My Queries but cannot save under Shared Queries.
5. Stakeholders can only view and approve releases.
6. Stakeholders cannot view markdown README files defined for repositories.

If you choose a feature that's not available to you as a stakeholder, you'll receive an error message indicating that you have insufficient permissions when you try to complete the task.

Features stakeholders can't access

If you need access to the following features—which support the daily work of product owners, team leads, developers, testers, and project administrators—you need to be have Basic access:

- Change the priority of an item within a backlog
- Delete work items or move work items to another team project
- Create shared queries, view charts, and modify the home page
- View Delivery Plans (a Marketplace extension)
- Access the full set of features of the Code, Build, Test, and Release hubs
- Participate in team rooms, which capture interactive, detailed conversations about the project.

> **NOTE**
>
> Stakeholders can view administrative pages that support managing permissions, area and iteration paths, and more; however, for the most part, they can't modify any objects on these pages. The one exception is their ability to manage users and access.

Related articles

For a comparison chart of Stakeholder vs Basic access, see this feature matrix. See also these quickstart guides:

- Add work items
- Create your backlog

- Kanban quickstart

If you want to provide a group of users access to provide feedback, then you don't need to give them Stakeholder access. Instead, simply give reviewers permissions to provide feedback.

Key concepts for working with VSTS and TFS

8/9/2018 • 6 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

The set of platforms, services, and tools you have access to through Visual Studio Team Services (VSTS) can be overwhelming. Before you start using our products, you'll want to become familiar with how they fit together. You'll gain that understanding here as well as pointers to additional topics and tutorials to gain confidence in using our products to develop your software.

Collaborative, integrated software development

VSTS, our hosted cloud offering, and Team Foundation Server (TFS), our on-premises platform, provide small teams as well as enterprises the services and tools to support developing and continuously deploying software. Even sole developers can use our platforms to manage their software and deploy their apps.

The three main areas that support software development include:

- Source control to manage versioning of software files
- Tracking tools to support planning and tracking work, code defects, issues and more
- DevOps tools to support building, testing, and continuous release of software apps.

Source control

Source or version control systems allow developers to collaborate on code and track changes made to the code base. Source control is an essential tool for multi-developer projects.

Our systems support two types of source control: Git (distributed) or Team Foundation Version Control (TFVC), a centralized, client-server system. Both systems enable you to check-in files and organize files within folders, branches, and repositories.

With Git, each developer has a copy on their dev machine of the source repository including all branch and history information. Each developer works directly with his or her own local repository, and changes are shared between repositories as a separate step.

Developers can commit each set of changes and perform version control operations such as history and compare without a network connection. Branches are lightweight. When developers need to switch contexts, they create a private local branch. Developers can quickly switch from one branch to another to pivot among different variations of the codebase. Later, they can merge, publish, or dispose of the branch.

> **NOTE**
> Git in VSTS and TFS is standard Git. You can use Visual Studio with third-party Git services, and you can also use third-party Git clients with TFS.

With TFVC, developers have only one version of each file on their dev machines. Historical data is maintained only on the server. Branches are path-based and created on the server.

Work tracking and Agile tools

Software development projects require ways to easily share information and track the status of work, tasks, issues, or code defects. In the past, you might have planned and track work using one or more tools such as Microsoft Excel, Microsoft Project, a bug tracking system, or a combination of tools. Now, many teams have adopted Agile methods and practices to support planning and development.

Our systems provide several types of work items which you use to track features, requirements, user stories, tasks, bugs, and issues. Each work item is associated with a work item type and a set of fields that team members update as information becomes available and progress is made.

For planning purposes, you have access to several types of backlogs and boards to support the main Agile methods—Scrum, Kanban, or Scrumban.

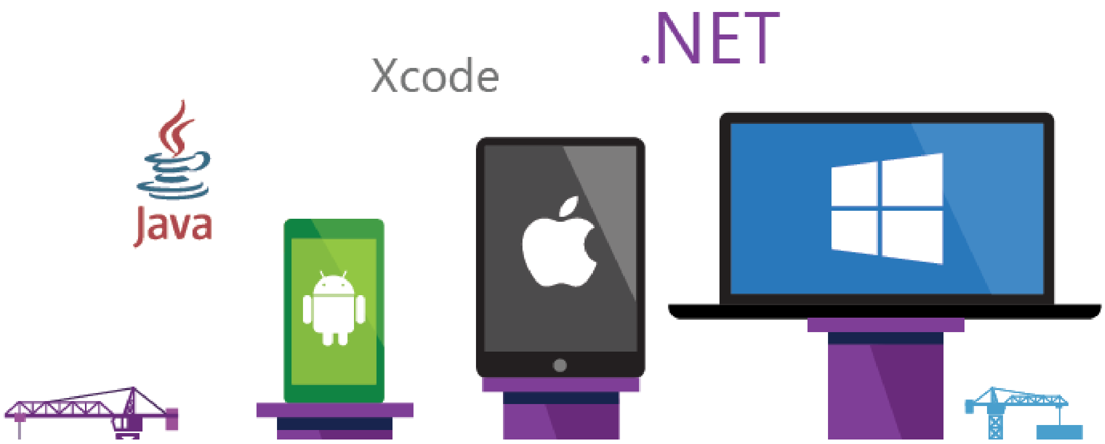
- Product backlog, used to create and prioritize stories or requirements
- Kanban, used to visualize and manage the flow of work as it moves from inception to in progress to done
- Sprint backlogs, used to plan work to complete during a sprint cycle, a regular 2 to 4 week cadence teams use when implementing Scrum
- Task board, used during daily Scrum meetings to review work completed, remaining, or blocked

Project managers and developers share information by tracking work items on the backlogs and boards. Useful charts and dashboards round out the picture helping teams monitor progress and trends.

DevOps and continuous integration

Rapid and reliable release of software comes from automating as many processes as possible. Our systems support build, test, and release automation.

- You can define builds to automatically run whenever a team member checks in code changes
- Your build pipelines can include instructions to run tests after the build runs
- Release pipelines support managing deployment of your software builds to staging or production environments



Scaling

Both VSTS and TFS are enterprise-ready, supporting teams of any size, from tens to thousands. VSTS provides a scalable, reliable, and globally available hosted service. It is backed by a 99.9% SLA, monitored by our 24x7 operations team, and available in local data centers around the world.

To learn more, see About projects and scaling your organization. Also, for stories and short videos on how Microsoft transitioned from waterfall to Agile, see Scaling Agile Across the Enterprise.

Customization and configuration of resources

You can configure and customize most services and applications to support your business needs or the way your teams work. For a comprehensive view of what resources can be configured, see About team, project, and organizational-level settings.

- **Dashboards**: Each team can configure their set of dashboards to share information and monitor their progress.
- **Source control**: For each Git repository, you can apply branch policies and define branch permissions. For TFVC repositories, you can set check-in policies.
- **Work tracking**: You can add fields, change the workflow, add custom rules, and add custom pages to the work item form of most work item types. You can also add custom work item types. For details, see Customize an inheritance process.
- **Build and Release**: You can fully customize your build and release pipelines, define build steps, release environments, and deployment schedule. For details, see Build and Release.
- **Test**: You can define and configure test plans, test suites, and test cases as well as configure test environments; additionally you can add test steps within your build pipelines. For details, see Exploratory & Manual Testing, Load testing, and et up continuous testing for your builds.

- **Dashboards**: Each team can configure their set of dashboards to share information and monitor their progress.
- **Source control**: For each Git repository, you can apply branch policies and define branch permissions. For TFVC repositories, you can set check-in policies.
- **Work tracking**: You can add fields, change the workflow, add custom rules, and add custom pages to the work item form of most work item types. You can also add custom work item types. For details, see Customize the On-premises XML process model.
- **Build and Release**: You can fully customize your build and release pipelines, define build steps, release environments, and deployment schedule. For details, see Build and Release.
- **Test**: You can define and configure test plans, test suites, and test cases as well as configure test environments; additionally you can add test steps within your build pipelines. For details, see Exploratory & Manual Testing, Load testing, and et up continuous testing for your builds.

Extensibility

In addition to all the pre-built functionality available to you, you can add to it in the following ways:

- Visual Studio Marketplace : Provides extensions that you can install either on your organization, server, or Visual Studio client
- Service hooks: Enable you to perform tasks on other services when events happen within your team project hosted on VSTS or TFS
- REST APIs: Provide the ability to create custom extensions that plug into VSTS or TFS
- Visual Studio SDK: Helps you extend Visual Studio features or integrate new features into Visual Studio. You can distribute your extensions to other users, as well as to the Visual Studio Marketplace.

Resources

- Pricing

Source control

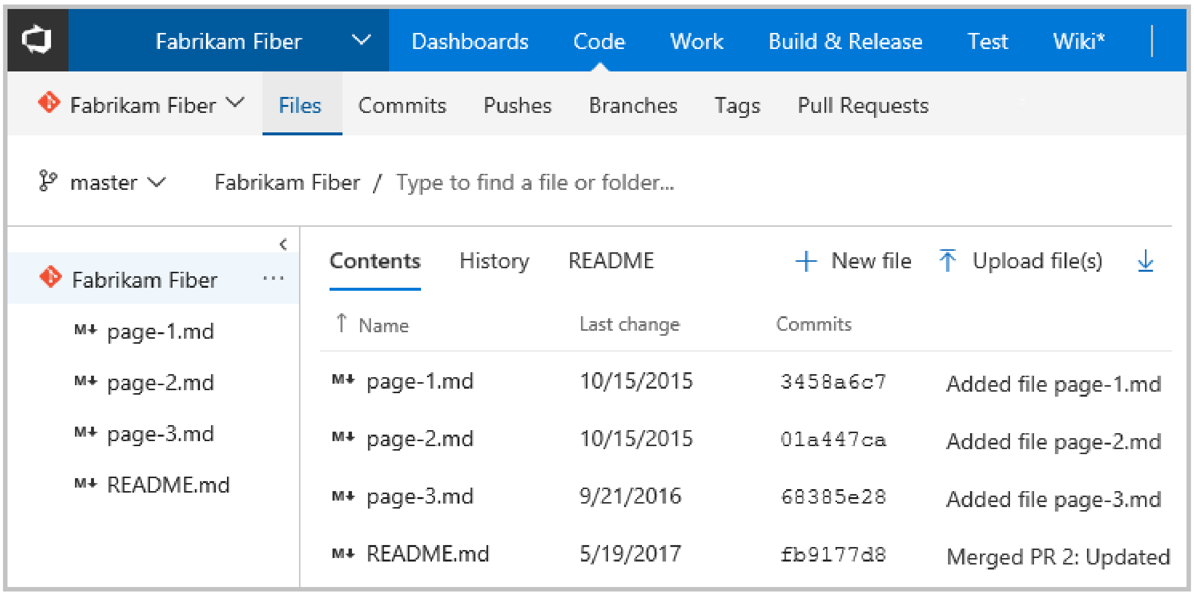
8/10/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

Source control, also referred to as version control, systems allow developers to collaborate on code and track changes made to the code base. Source control is an essential tool for multi-developer projects.

Our systems support two types of source control: Git (distributed) or Team Foundation Version Control (TFVC), a centralized, client-server system. Both systems enable you to check-in files and organize files within folders, branches, and repositories.

You manage your repos, branches, and other code development operations from the **Code** hub.



With Git, each developer has a copy on their dev machine of the source repository including all branch and history information. Each developer works directly with his or her own local repository, and changes are shared between repositories as a separate step.

Developers can commit each set of changes and perform version control operations such as history and compare without a network connection. Branches are lightweight. When developers need to switch contexts, they create a private local branch. Developers can quickly switch from one branch to another to pivot among different variations of the codebase. Later, they can merge, publish, or dispose of the branch.

> **NOTE**
> Git in Visual Studio, VSTS and TFS is standard Git. You can use Visual Studio with third-party Git services, and you can also use third-party Git clients with TFS.

With TFVC, developers have only one version of each file on their dev machines. Historical data is maintained only on the server. Branches are path-based and created on the server.

Try this next

Start sharing your code or getting your code under source control.

Code with Git

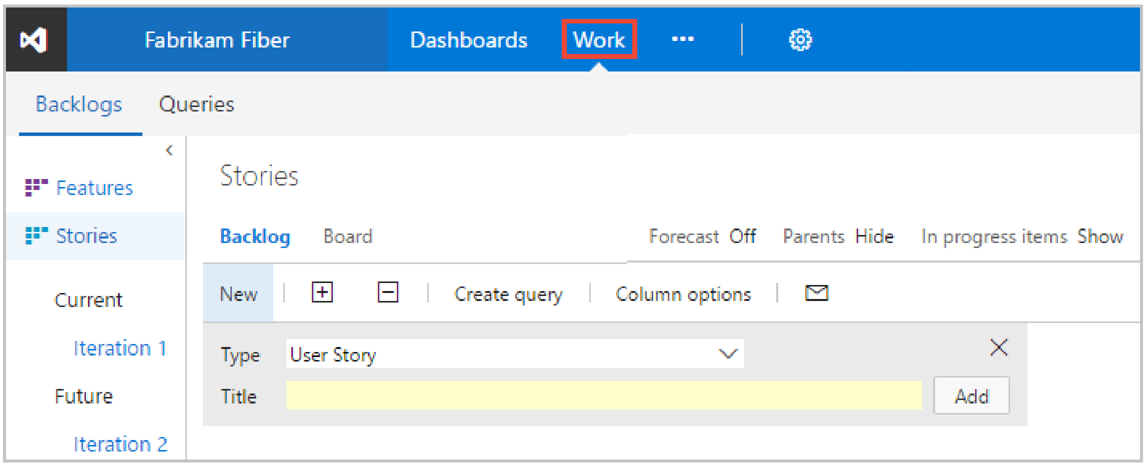
About Agile tools and Agile project management

8/9/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

You plan and track your project using the suite of Agile tools you access from the web portal. Agile tools support the core Agile methods—Scrum and Kanban—used by software development teams today. Scrum tools support defining and managing work within sprints, setting capacity, and tracking tasks. Kanban tools allow you to manage a continuous flow of work via an interactive sign board.

You access all Agile tools from **Work**. If you're new to Agile, see What is Agile? for an overview.



Most Agile tools are scoped to a team. This supports team autonomy as well as scaling the system.

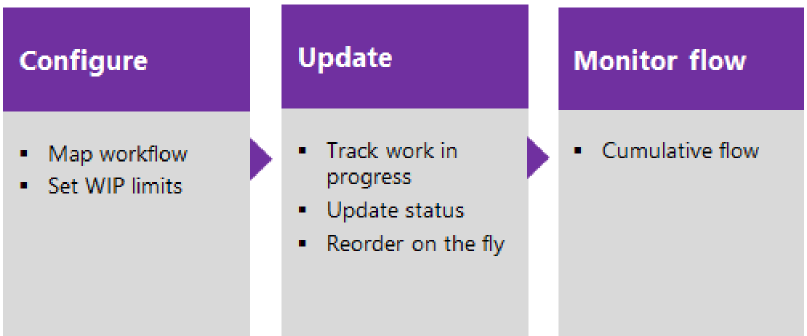
BACKLOGS	TASK BOARDS	KANBAN BOARDS	OTHER
- Product backlog - Portfolio (Epics, Features) backlogs - Sprint backlogs	- Task boards - Sprint burndown - Capacity planning	- Kanban board - Task checklists - Epics & Features boards	- Delivery plans - Forecast - Velocity

NOTE

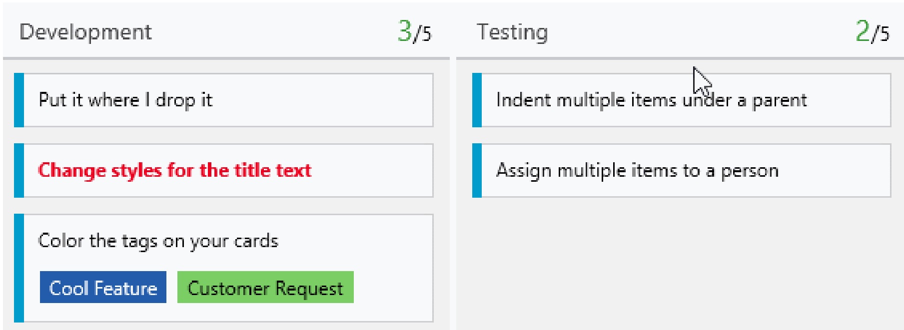
To understand the differences between backlogs, boards and Delivery plans, see Backlogs, boards, and plans.

Kanban method and tools

Kanban uses a visual interactive board to plan and show progress using cards. Your Kanban board is fully customizable to support the workflow used by your team.

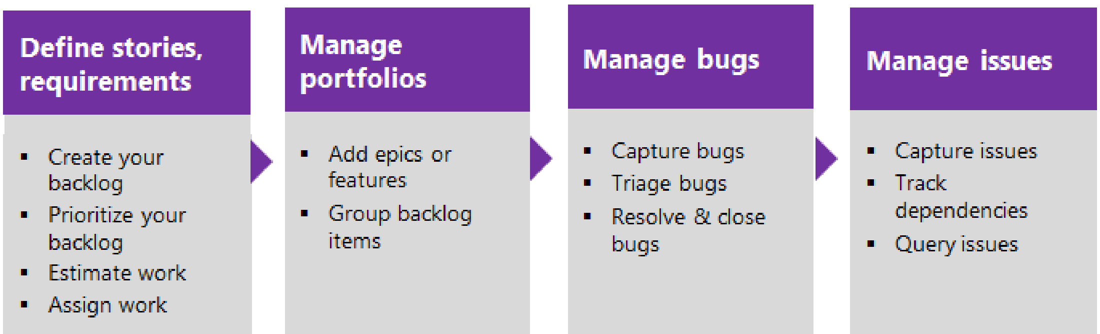


You update the status of work by dragging card to another column on the Kanban board. You can even change the order of items as you move a card to a new column.



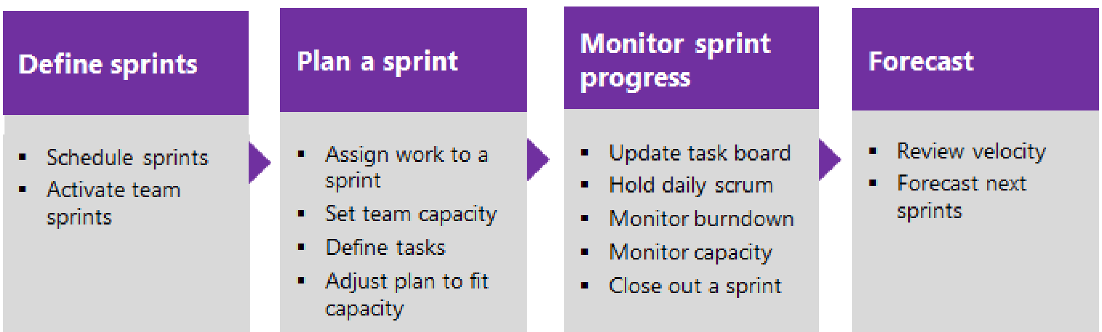
Define work items and create your backlog

You build your project plan by creating a backlog of work items that represent the features, requirements, user stories, or other work to perform. Portfolio backlogs provide support for organizing work in a hierarchical fashion and tracking major product initiatives or scenarios that rely on many stories or requirements. Different types of work items help you track different types of work, such as user stories, tasks, bugs, issues, and more.



Scrum method and tools

The Scrum method uses sprints to plan work to perform by a team within a specific time period and cadence. To get started, several sprints are predefined for your team. If you're new to Scrum, get an overview from What is Scrum?.



You can quickly assign work items to a sprint by dragging and dropping them from the product backlog to the sprint.

Work across projects

Use your Project page to view and quickly navigate to teams, projects, branches, work items, pull requests and other objects that are relevant to you. For details, see Work across projects.

Try this next

Take these tools for a test run by signing up for a free VSTS organization. From there, you're ready to create your backlog.

Once you've started tracking work, you'll want to monitor and track progress, identify trends, and share your plans. See Charts, dashboards, and widgets.

Related articles

You access tools provided by VSTS and TFS by connecting from a client to the server, either in the cloud or on-premises. Some web portal tools require additional Visual Studio Subscriptions or Advanced/VS Enterprise access. To learn more, see Permissions and access for work tracking.

For an overview of all your customization options, see Customize your work tracking experience.

Tools and clients that connect to VSTS and TFS

8/9/2018 • 7 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

Our platform of software development tools began over 20 years ago with the releases of Visual Basic and Visual Studio as an integrated development environment (IDE). Visual Studio supports a number of plug-ins which extend its functionality. In particular, the Team Explorer plug-in allows the Visual Studio client to connect to VSTS and TFS to support source control, work tracking, build, and test operations.

The set of tools available to you that interface with VSTS and TFS include:

- Desktop client developer tools
- Office integration tools
- Web based tools
- Command-line tools
- Marketplace extensions
- REST APIs

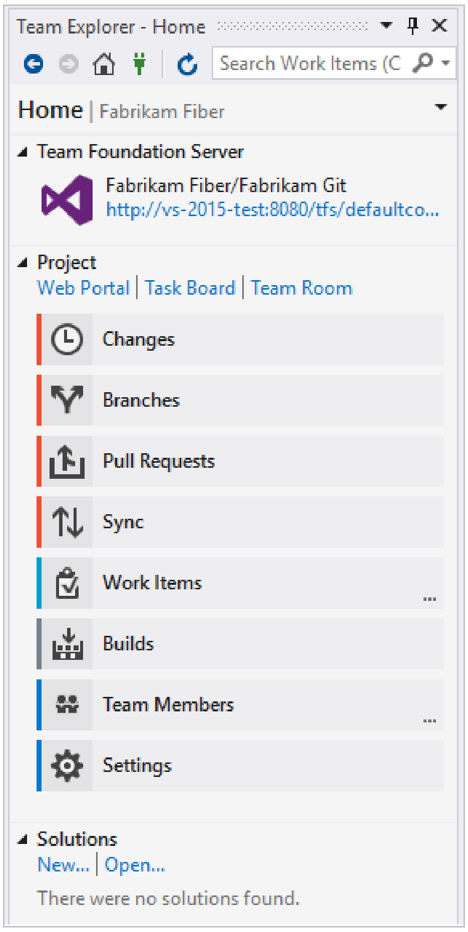
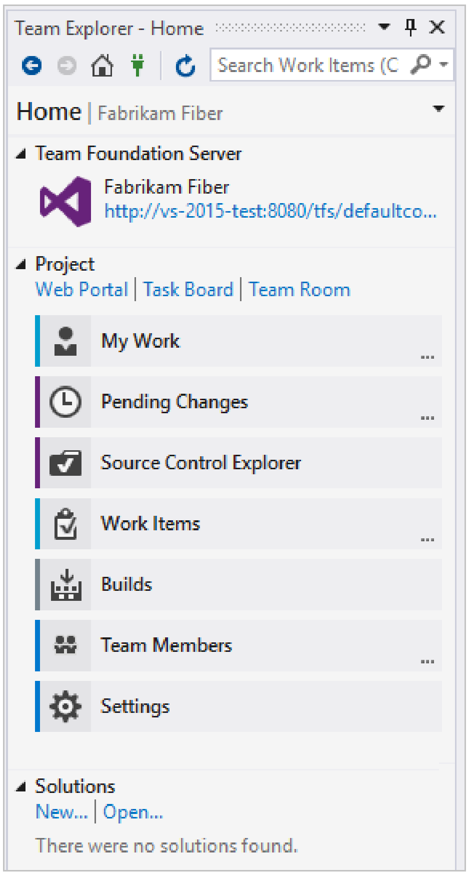
Desktop client developer tools

Developers have access to a variety of tools through these versions of Visual Studio and plug-ins. to download any version of Visual Studio, visit the Visual Studio Downloads page. To understand what features you get with the Visual Studio versions, see Compare Visual Studio Offerings.

- **Visual Studio Community**: A fully-featured and extensible IDE for creating modern applications for Android, iOS, and Windows, as well as web applications and cloud services. (Replaces Visual Studio Express)
- **Visual Studio Professional**: Development tools and services to support individual developers or small teams
- **Visual Studio Enterprise**: Integrated, end-to-end development tools and solutions for teams of any size and those with a need to scale Supports designing, building and managing complex enterprise applications
- **Visual Studio Test Professional**: Provides access to Microsoft Test in addition to development tools to support quality and collaboration throughout the development process
- **Visual Studio Team Explorer**: A free solution for non-developers to interact with Team Foundation Server and Visual Studio Team Services.
- **Eclipse/Team Explorer Everywhere**: Free plug-in to support teams running Eclipse on Linux, macOS, or Windows that connect to VSTS or TFS
- **Android Studio with the VSTS Plugin for Android Studio**: Free plug-in to support Android developers and connect to Git repositories on VSTS or TFS
- **IntelliJ with the VSTS Plugin for IntelliJ**: Free plug-in to support developers who use IntelliJ IDEA or Android Studio to connect to Git repositories on VSTS or TFS
- **Visual Studio Code**: Free, open-source code editor with a free extension to support connecting to Git repositories on VSTS or TFS.

Team Explorer plug-in

Team Explorer, a plug-in to all Visual Studio versions, provides connects Visual Studio to team projects defined in VSTS or TFS. You can manage source code, work items, and builds. To learn more, see Work in Team Explorer.

<table>
<tr><th>HOME PAGE WITH GIT</th><th>HOME PAGE WITH TFVC</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Office integration tools

When you install any edition of Visual Studio or Team Foundation Server Standalone Office Integration 2015 (free), the Team Foundation plug-in installed to integrate work item tracking with select Office clients. The Team Foundation plug-in installs to your existing Office client. The plug-in supports Office 2007, Office 2010, or Office 2013 versions.

- **Excel**: Use Excel to add and bulk modify work items.
- **Project**: Using Project you can plan projects, schedule tasks, assign resources, and track changes. You have access to features that TFS doesn't support, such as a project calendar, Gantt charts, and resource views.
- **Project Professional**: With Project Professional and the Team Foundation Server Extensions for Project Server you can manage projects that synchronize data that exists in both TFS and Project Server. Project managers and software development teams can use the tools that they prefer, work at the level of precision that supports their needs, and easily share information.
- **PowerPoint Storyboarding**: Lets you illustrate user stories and requirements using PowerPoint. The Team Foundation plug-in installs to your existing PowerPoint client.

> **IMPORTANT**
> Support for integrating TFS with Project Server is deprecated for TFS 2017. However, synchronization support is provided by a third party. See Synchronize TFS with Project Server for details.

Task-specific clients

The following clients support specific tasks, such as managing testing efforts, providing feedback, or modifying work items.

- **Microsoft Test Manager**: Allows you to manage your test efforts, create and run manual tests, and create and track bugs that are found during test efforts. Test Manager installs with Visual Studio Test Professional and Visual Studio Enterprise.
- **Test & Feedback extension (previously called the Exploratory Testing extension)**: Provides a lightweight, plug-in to a web browser. Stakeholders can respond to feedback requests for user stories and features generated in VSTS or TFS. This extension is free to stakeholders.
- **Microsoft Feedback Client**: Your stakeholders can use **Microsoft Feedback Client** to record feedback for your application as video, audio, or type-written comments. This client installs with all versions of Visual Studio, or can be installed from the free download. All feedback is stored in the work item data store and requires stakeholders to have required permissions.

> **IMPORTANT**
>
> Test Manager is deprecated for TFS 2017.

Browser-based web tools

Web portal

The collaboration tools supported through the web portal are summarized under Essential services. On VSTS, new features are deployed usually every three weeks, and for TFS, usually quarterly. For release notes, see VSTS Features Timeline.

You can use these browsers to access the web portal (VSTS and TFS).

VERSION	EDGE	INTERNET EXPLORER	SAFARI (MAC)	FIREFOX	CHROME
VSTS	most recent	11 and later	9.1 and later	most recent	most recent
TFS 15	most recent	11 and later	9.1 and later	most recent	most recent
TFS 2015	most recent	9 and later	5 and later	most recent	most recent
TFS 2013		9 and later	5 and later	most recent	most recent

Edge, Firefox, and Chrome automatically update themselves, so VSTS and TFS support the most recent version.

To learn more, see Navigation basics.

Browser-based extensions

The following extensions are available from the Visual Studio Marketplace and are built and maintained by the VSTS product team.

- Test Manager: Run tests using your browser with simple pass/fail of steps, add comments/attachments, take screenshots and file bugs. You can accomplish all of this with automatic end-to-end traceability.
- Package Management: Build packages of reusable code components and share them. The Package Management extension enables continuous delivery workflows by supporting multiple packaging protocols such as NuGet and npm . It makes packages available to your team, your builds, and your releases.
- Code search: Increases cross-team collaboration and code sharing by enabling developers to quickly locate relevant information within the code base of all team projects hosted within an organization or collection. With it, you can discover implementation examples, browsing definitions, and find error text.
- Work item search: Quickly find relevant work items by searching across all work item fields over all projects in an organization. Perform full text searches across all fields to efficiently locate relevant work items. Use in-line

search filters, on any work item field, to quickly narrow down to a list of work items.

Find additional extensions from the Marketplace

Application monitoring tools

To monitor your applications you can use Application Insights for web apps or HockeyApp for mobile apps.

Monitor web applications with Application Insights

Application Insights is an extensible Application Performance Management (APM) service for web developers. Use it to monitor your live web application. It will automatically detect performance anomalies. It includes powerful analytics tools to help you diagnose issues and to understand what users actually do with your app. It's designed to help you continuously improve performance and usability. It works for apps on a wide variety of platforms including .NET, Node.js and J2EE, hosted on-premises or in the cloud.

With Application Insights you can:

- Gain actionable insights through application performance management and instant analytics
- Detect and diagnose exceptions and application performance issues
- Monitor Azure websites, including those hosted in containers, plus websites on-premises and with other cloud providers
- Seamlessly integrate with your DevOps pipeline using VSTS, GitHub, and our webhooks
- Get started from within Visual Studio, or monitor existing apps without re-deploying

To learn more, see Microsoft Azure - Application Insights.

Monitor mobile applications with HockeyApp

With HockeyApp you can develop, distribute, and beta-test your mobile apps. HockeyApp supports:

- Android, Cordova, iOS, macOS, Unity, Windows, and Xamarin apps
- Live, reliable crash reports
- Collect in-app feedback from real users
- Open-source SDKs to let you know what code is running in your apps
- Integration with your existing build system and work item management solution

To learn more, see Microsoft Azure - HockeyApp.

Command-line tools

You can perform many code development and administrative tasks using command line tools.

- Git commands
- TFVC commands
- TFSConfig
- TFSDeleteProject
- TFSSecurity
- TFSServiceControl
- witadmin (work item tracking)

Marketplace extensions

While Visual Studio, VSTS, and TFS provide a wealth of features and functionality, they also provide a means to extend and share that functionality.

Extensions are simple add-ons that you can use to customize and extend your DevOps and work tracking experiences. Written with standard technologies—HTML, JavaScript, CSS—you can develop your own extensions using your preferred dev tools.

You build extensions using our RESTful API Library. You publish them to the Visual Studio Marketplace, where you can privately maintain them or share with millions of developers that use Visual Studio, VSTS, and TFS.

To learn more, visit the Marketplace and Overview of extensions.

REST APIs

The VSTS and TFS APIs are based on REST, OAuth, Json and service hooks—all standard web technologies broadly supported in the industry.

REST APIs are provided to support building extensions to VSTS and TFS. To learn more, see REST API overview.

Related articles

- Key concepts
- Essential services
- Software development roles
- Pricing

Software development roles supported by VSTS and TFS

8/10/2018 • 4 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

If you are a sole developer, or work on a small team, chances are that you participate in many activites—performing tasks associated with issue tracking, feature planning, coding, testing, build, and deployment.

However, if you work within a large organization, you're probably more focused on a specific set of tasks that are traditionally aligned with one or two specific roles, such as software development, project management, and DevOps.

This topic provides a roadmap to support you in quickly coming up to speed on the features and tasks available to you based on the role you perform.

Contributor roles

Team members are contributors who have access to the code base, work item tracking, Agile tools, build pipelines, test tools, and more. If you need to lock down specific areas to a select set of contributors, you can do that through the permission management.

Software developers

Developers use Visual Studio or other tools to develop their applications. They then check in their changes to a Git or TFVC repository hosted in VSTS or TFS. From the web portal or supported IDE, they can view repositories, check history, and more.

- To get started using Git, see one of these resources:
  - Share your code with Git and Visual Studio.
  - Share your code in Git using Eclipses.
  - Share your code in Git using Xcode.
  - Share your code in Git using IntelliJ.
  - Get Started with Git and VSTS.
- To get started using TFVC, see one of the following resources:
  - Develop and share your code in TFVC using Visual Studio
  - Share your code in TFVC using Eclipse
  - Share your code in TFVC using Xcode

Project managers

The role of project manager typically encompasses planning the feature set to deliver, setting priorities, and tracking the status of work, code defects, and customer issues. The suite of web-based Agile tools provide PMs with the views and features they need to perform these tasks. All work is captured within a work item. Each work item represents a specific type such as a user story, task, or bug.

- Use the product backlog to quickly define and prioritize user stories, features, and other work items
- Use the sprint backlog and task board to implement Scrum practices
- Use the Kanban board to work with Kanban methods
- Use queries to list and update work items, create status and trend charts, and post charts to dashboards
- Use dashboards to share information, status, and trends with your team or organization

To get started, see Get started with Agile tools to plan and track work.

If you are used to using Excel or Project to plan and track your work, you can still use these tools and integrate with VSTS and TFS. See Bulk modify using Excel and Create your backlog and tasks using Project.

DevOps: builders, testers, and release managers

One of the main advantages to working with VSTS or TFS is the suite of tools and integrated functionality that supports build, testing, and deploying software applications. Here are the main DevOps associated tasks supported by VSTS and TFS:

- Define builds
- Unit test your code
- Run tests with your builds
- Performance test your apps
- Perform exploratory tests
- Define, manage, track, and approve releases
- Deploy applications to Azure, a virtual machine, Docker containers, and more

To get started, see the overviews provided here: Build & Release and Test.

Stakeholders

With stakeholder access, anyone within your organization can check project status and provide feedback. Stakeholders can track project priorities and provide direction, feature ideas, and business alignment to a team. They can contribute to plans by adding and modifying work items. They can't, however, contribute to the code base or exercise test tools.

Stakeholder access essentially provides free access to a limited set of feature to project sponsors and supporters. To learn more, see Work as a Stakeholder.

Administrator roles

A distinct advantage to working in VSTS is the reduced overhead of server maintenance. That said, there are still several administrative tasks required to support a collaborative, integrated software development environment

The main tasks are grouped here by membership within a security group or role:

Team administrators

Responsible for configuring team settings which include:

- Backlog and board settings
- Team area(s) and iterations (sprints)
- Team members
- Team dashboards
- Team work item templates
- Team alerts

To get started, see Manage team assets.

Project administrators

Responsible for project level settings, including:

- Area paths and Iteration paths
- Project permissions and repository security
- Customize work tracking objects (TFS only)
- Build agents, pools, and service connections

- Test and release retention policies

Project collection administrators

Responsible for organization or collection-level settings. These include:

- Manage billing
- Add and manage team projects
- Manage collection-level settings and permissions
- Customize work tracking processes
- Install and manage extensions (install custom or Marketplace extensions)

To get started, see Organization Management.

TFS administrators

Responsible for installing, upgrading, and maintaining an on-premises TFS deployment. Tasks include:

- Install TFS
- Update servers running TFS
- Manage database backups
- Server administrative settings and permissions
- Build retention policies
- Add and manage team project collections

To get started, see Server Administration (TFS).

Related notes

- Key concepts
- Essential services

Understand differences between VSTS and TFS

8/9/2018 • 10 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

VSTS and Team Foundation Server (TFS) both provide an integrated, collaborative environment that supports Git, continuous integration, and Agile tools for planning and tracking work.

VSTS is the cloud offering that provides a scalable, reliable, and globally available hosted service. It is backed by a 99.9% SLA, monitored by our 24-7 operations team, and available in local data centers around the world.

Team Foundation Server is the on-premises offering built on a SQL Server back end. Enterprises typically choose on-premises TFS when they need their data to stay within your network, or they want access to SharePoint sites and SQL Server reporting services that integrate with TFS data and tools.

While both offerings provide the same essential services, compared with TFS, VSTS provides organizations the following added benefits:

- Simplified server management
- Immediate access to the latest and greatest features
- Improved connectivity with remote sites
- A transition from capital expenditures (servers and the like) to operational expenditures (subscriptions).

Use this topic to determine which offering—cloud or on-premises—meets your organizational needs by considering these important areas:

- Fundamental differences between TFS and VSTS
- Differences in specific feature areas between TFS and VSTS

For each area, we'll discuss both the current state of the world and the expected impacts from short and medium-term plans. Check back here for updates, because this information may change frequently.

If you're on TFS and considering moving to VSTS, read Migrate data from TFS to VSTS to understand your options.

Fundamental differences between TFS and VSTS

When you plan a move, there are a few fundamental differences between TFS and VSTS that are important for you to understand.

Scope and scale data

TFS has three options for scoping and scaling data—deployments, team project collections, and team projects. In the simplest case, deployments are just servers. Deployments can also be more complicated, however, including everything from a two-server deployment where SQL is split out on a separate machine to high availability farms comprising lots of servers. Team project collections serve as containers for security and administration in addition to serving as physical database boundaries. They are also used to group related team projects. Finally, team projects are used to encapsulate the assets of individual software projects, including source code, work items, and so on. Learn more about these concepts at Manage team project collections.

VSTS is slightly different. It currently only has two options for scoping and scaling data—organizations and team projects. Organizations in VSTS get their own URLs (for example, `https://contoso.visualstudio.com`) and always contain exactly one team project collection. Organizations can contain multiple team projects, like TFS team project collections.

We are planning a third option for scoping and scaling data in VSTS—a new entity called an Enterprise. Rather than adding support for multiple team project collections within an organization, multiple organizations could be grouped within an organization. Additionally, we will merge organizations and their single team project collections into a single entity. The organization will be similar to the TFS deployment, and the organization will be similar to the TFS collection.

To be ready to use the organization entity, we recommend that you create organizations in VSTS whenever you would have created collections in TFS. In the short term, having your work split across multiple organizations can cause some problems, but we plan to address these when the organization entity is introduced. In particular:

- You purchase VSTS users per organization, meaning that paid users only have access to the VSTS organization in which the payment is made. If you have users who need access to multiple organizations, Visual Studio subscriptions can be an attractive option, since subscribers can be added to any number of VSTS organizations at no charge. We arealso considering other ways we might make access to multiple organizations grouped into an organization available.
- You currently have to administer organizations one at a time, which can be cumbersome when you have many organizations. We're working to support enterprise-wide policies.

Authentication

With TFS, you typically connect to an intranet server (for example, `https://tfs.corp.contoso.com:8080/tfs`). You authenticate with Windows Authentication and your Active Directory (AD) domain credentials. Usually this process is transparent, and you'll never see any kind of sign-in experience.

With VSTS, you connect over the public internet (for example, `https://contoso.visualstudio.com`). You'll either authenticate with Microsoft Organization credentials or with Azure Active Directory (Azure AD) credentials, depending on your VSTS organization setup. You can also set up Azure AD to require features like multi-factor-authentication, IP address restrictions, and so on.

We recommend that organizations configure their VSTS organizations to use Azure AD rather than Microsoft Accounts. This provides a better experience in many scenarios and more options for enhanced security.

Users and groups

In TFS, you provide users access to deployments by adding Active Directory (AD) groups to various TFS groups (for example the Project Contributors group for an individual team project). The AD group memberships are kept in sync. As users are added and removed in AD they also gain and lose access to TFS.

In VSTS, you can use a similar mechanism to provide access to groups of users by adding Azure AD groups to TFS groups. If you use Microsoft Accounts instead of Azure AD, you will have to add users one at a time.

Manage user access

In TFS and VSTS, you can give free access to work item features to an unlimited number of Stakeholders. Also, unlimited Visual Studio subscribers can have access to all Basic features at no additional charge. You only need to pay for other users who need access.

In TFS, all use is on the honor system. To set access levels for users based on their licenses, use specify their access levels administration page. For example, assign unlicensed users Stakeholder access only. Users with a TFS Client Access License (CAL) can have Basic access. Visual Studio subscribers can have either Basic or Advanced access, based on their subscriptions. Note that TFS does not attempt to verify these licenses or enforce compliance.

In VSTS, you must assign an access level to each user in your organization's Users hub. VSTS validates Visual Studio subscribers as they sign in. You can assign Basic access for free to five users without Visual Studio subscriptions. To give Basic access to more users, you'll need to set up billing for your organization and pay for more users. Otherwise, all other users get Stakeholder access.

If you use Azure AD groups to provide access to groups of users, VSTS will assign appropriate access levels to them automatically when they sign in for the first time. For VSTS organizations configured to use Microsoft Accounts for sign-in, you will have to assign access levels to each user explicitly.

Security and data protection

Many organizations want to know more about data protection when they consider moving to the cloud. Microsoft is committed to ensuring that VSTS projects stay safe and secure. We have technical features and business processes in place to deliver on that commitment. You can also take steps to secure your data. Learn more in our Data Protection Overview white paper.

Key feature differences between VSTS and TFS

Even though VSTS is a hosted version of TFS, there are some differences between the features available in the two products. Some TFS features are not supported in VSTS at all—for example, VSTS does not support integration with SharePoint or Project Server.

Process customization

You customize the work tracking experience in two different ways depending on the supported process model:

> - For VSTS, you use the **Inheritance** process model which supports WYSIWYG customization
> - For TFS, you use the **On-premises XML** process model which supports customization through import/export of XML definition files for work tracking objects

While the **On-premises XML** process model option is quite powerful, it also can cause a number of problems. Chief among these is that processes for existing team projects do not update automatically when TFS is upgraded.

For example, TFS 2013 introduced several new features which depended on new work item types and other process template changes. When you upgrade from TFS 2012 to TFS 2013, each team project collection gets new versions of each of the "in the box" process templates which include these changes. However, these changes are not automatically incorporated in existing team projects. Instead, after you finish upgrading you have to include them in each team project by using the Configure Features wizard or a more manual process.

To avoid these issues in VSTS, custom process templates and **witadmin.exe** have always been disabled (only export functions are enabled). This has enabled us to automatically update all team projects with each VSTS upgrade. Meanwhile, the product team has been working hard to make customizing processes possible in ways that we can support easily and continuously. These first of these changes was recently introduced, and more changes are on the way.

With these new VSTS process customization capabilities, you can make customizations directly within the VSTS Web UI. If you want to customize your processes programmatically, you can also make customizations through REST endpoints. When you customize team projects in this way, those projects will continue to update automatically when we release new versions of their base processes with VSTS updgrades.

To learn more, see Customize your work tracking experience.

Over time we will support more and more types of process customizations with this new approach. If you need process customization features which are not yet available and cannot wait for them, a second option for process customization in VSTS is available, referred to as **Hosted XML** process model, and in private preview and by request only.

With this option, you import customized process templates. This option is quite similar to using custom process templates in TFS, except that:

- Restrictions exist in the customizations that can be imported into VSTS.
- Process templates are associated with all team projects created from them, and changes made to the process are reflected in each team project.

Team projects in organizations which participate in this process customization private preview will not update automatically with VSTS upgrades.

Reporting

Both TFS and VSTS have a variety of tools to give you insight into the progress as well as the quality of your software projects. These include:

- Dashboards and lightweight charts, available in both TFS and VSTS. These are very easy to set up and use.

In addition, with VSTS you have access to the following services:

- Analytics service and Analytics widgets. The Analytics service is optimized for fast read-access and server-based aggregations.
- PowerBI integration which supports getting Analytics data into Power Bi reports, providing a nice combination of simplicity and power.
- OData support which allows you to directly query the Analytics Service from a supported browser and then use the returned JSON data as you desire. Enterprise organizations can generate queries that span multiple team projects or their entire VSTS organization.

We plan to make these services available in TFS in a future release.

The following reports and dashboards are only available in TFS:

- Excel reports
- SQL Server Reporting Services (SSRS) reports
- SharePoint dashboards

And, available today only in VSTS:

- A PowerBI connector which provides a nice combination of simplicity and power. We plan to make it available in TFS in a future release.
- Analytics which is the reporting platform for Visual Studio Team Services (VSTS), which answer quantitative questions about the past or present state of your projects. This service provides a concise data model over the VSTS suite. Built for reporting, the Analytics Service is optimized for fast read-access and server-based aggregations. We plan to make it available in TFS in a future release.

Related articles

- Key concepts
- Essential services
- Client-server tools
- Software development roles
- Pricing - VSTS
- Pricing - TFS

Enable preview features

8/10/2018 • 2 minutes to read

VSTS

As new features are introduced, you can turn them on or off. That way, you can try them out, provide feedback, and work with those features that meet your requirements.

Some features provide a new user interface and functionality, which can be managed per user or team member. Others support a default experience for the account and are managed by an account administrator.

> **NOTE**
>
> You can turn on or off select features for VSTS. Preview features become available first on VSTS and then become standard features with an update to TFS. At some point, the preview feature moves out of preview status and becomes a regular feature of the web portal.

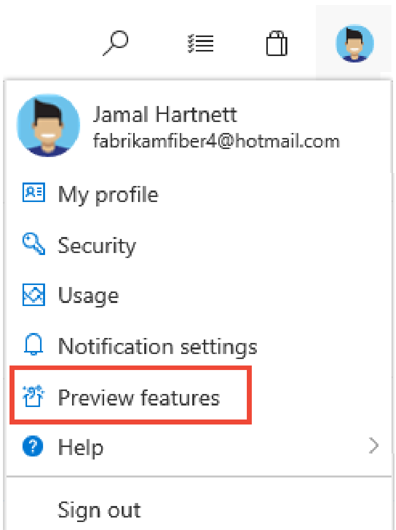
PREVIEW FEATURES PER USER	PREVIEW FEATURES PER VSTS ORGANIZATION
- New Navigation - New release progress views - New Releases Hub	- Build with multiple queues - Build YAML Definitions - Git Forks - New build results page - New Navigation - New release progress views - New Releases Hub

Enable features for your use

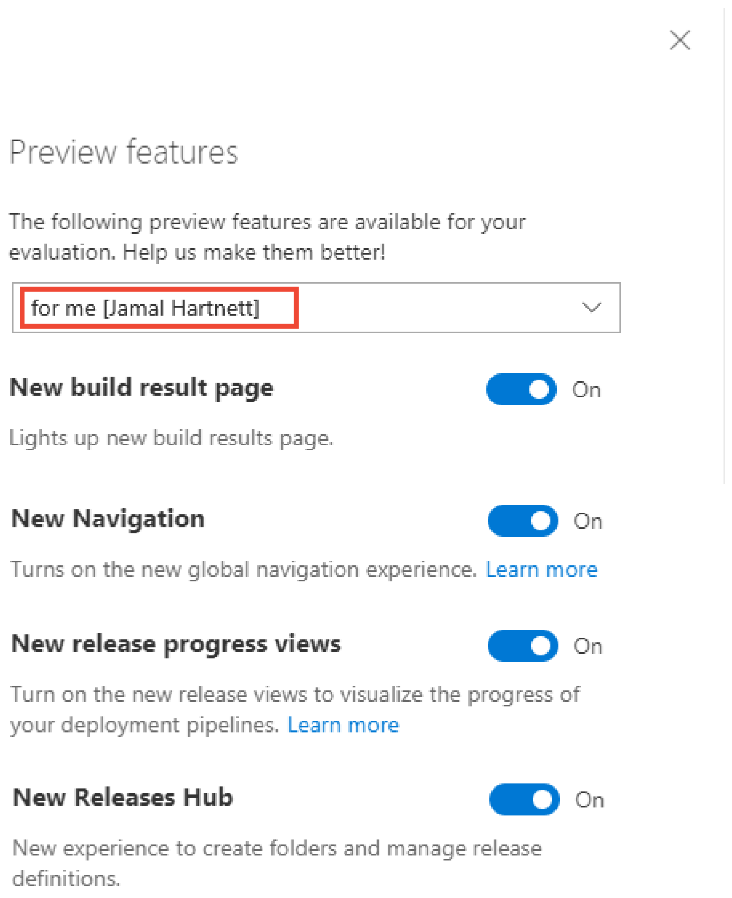
From time to time, a new feature is introduced in Preview mode, which allows you to turn it on or off.

- New navigation
- Previous navigation

1. To access the Preview features options, open your profile menu, and select **Preview features**.



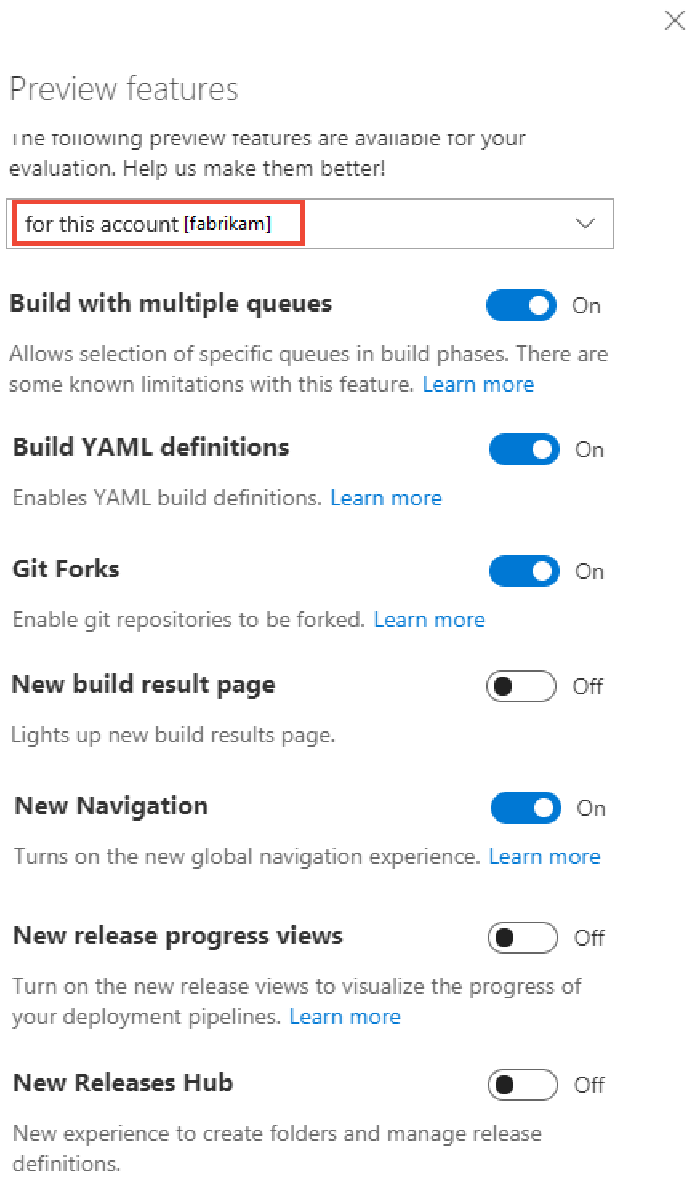
2. To enable or disable a feature, choose the slider.



Enable features at the organization (account) level (for all users)

When you enable a feature at the organization level, you essentially turn it on for all users of your account. Each user can then disable the feature if they so choose.

> **TIP**
>
> If you don't see the **for this account** menu option, then you aren't an account administrator. To get added as one, see Add administrators, set permissions at the team project or collection level.



Features now enabled for all VSTS projects

- Approval gates in releases
- Combine email recipients
- New Dashboards Experience
- New experience in Code, Work Item, & Wiki search
- New Queries Experience
- New Release Definition Editor
- New Test Plan Experience
- New Work Items Hub
- NuGet.org upstream sources
- Out of the box notifications.
- Pull Request Status Policy
- Symbol server
- Streamlined User Management
- Task tool installers
- Team expansion for notifications
- Updated package experience
- Wiki

Use @mentions

8/10/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015.2

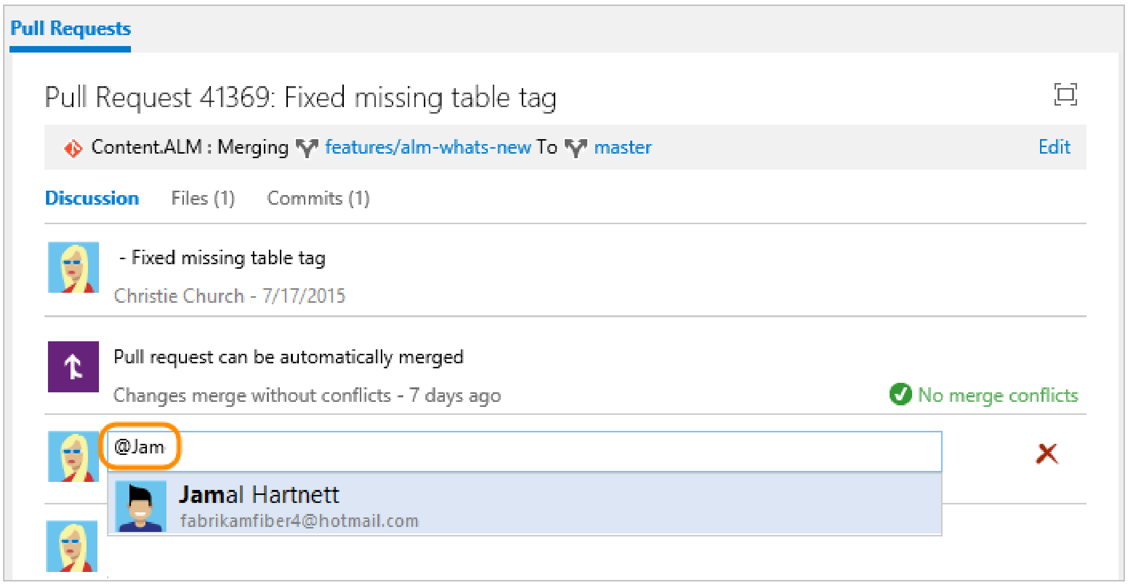
The **@mention** control allows you to quickly pull someone into a work item or pull request.

> **NOTE**
>
> The **@mention** control is available from TFS 2015 Update 2 and later versions.

In order for team members to receive notifications, you must configure an SMTP sever.

When leaving a code comment in a pull request, you can type **@** to trigger the **@mention** identity picker. From the identity selector, you'll see a list of those people that you have you've recently mentioned. You can choose one of those names or type in the name of the person you are looking for to perform a directory search.

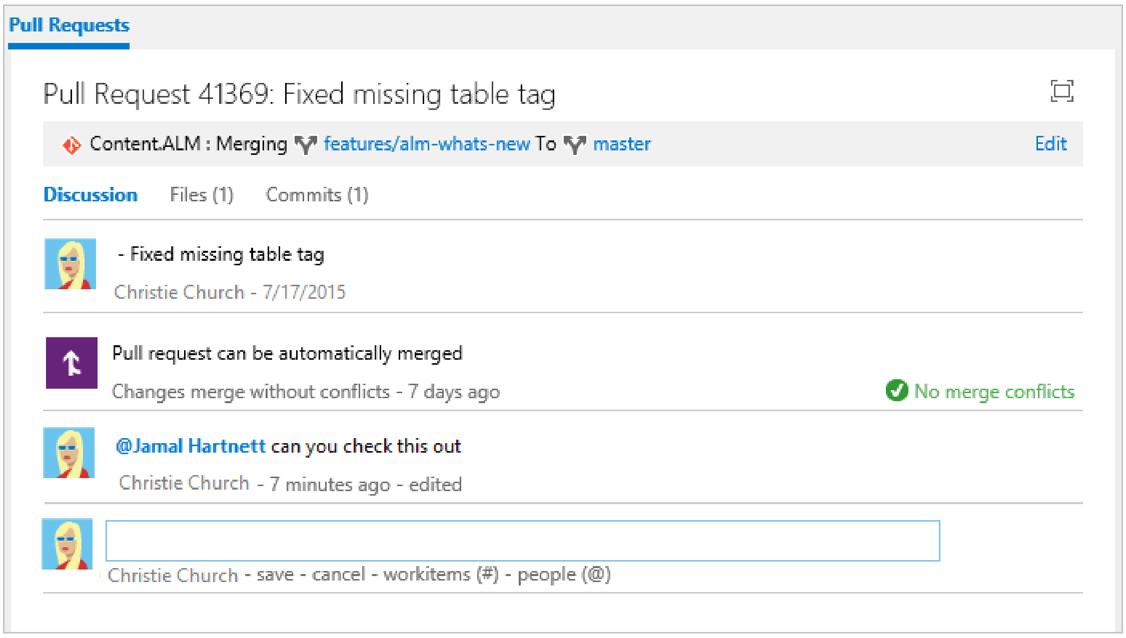
To filter the list, enter the user name or alias until you've found a match.



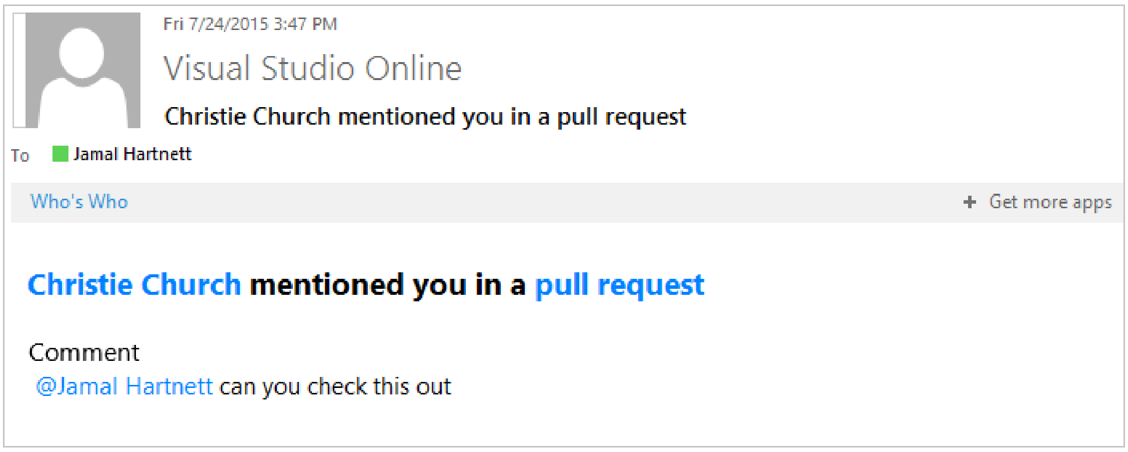
You can also use group mentions. Simply type the name of a team or a security group, choose the search icon and then select from the options listed.

To **@mention** a user you've never selected previously, just continue typing to perform your search against the full directory.

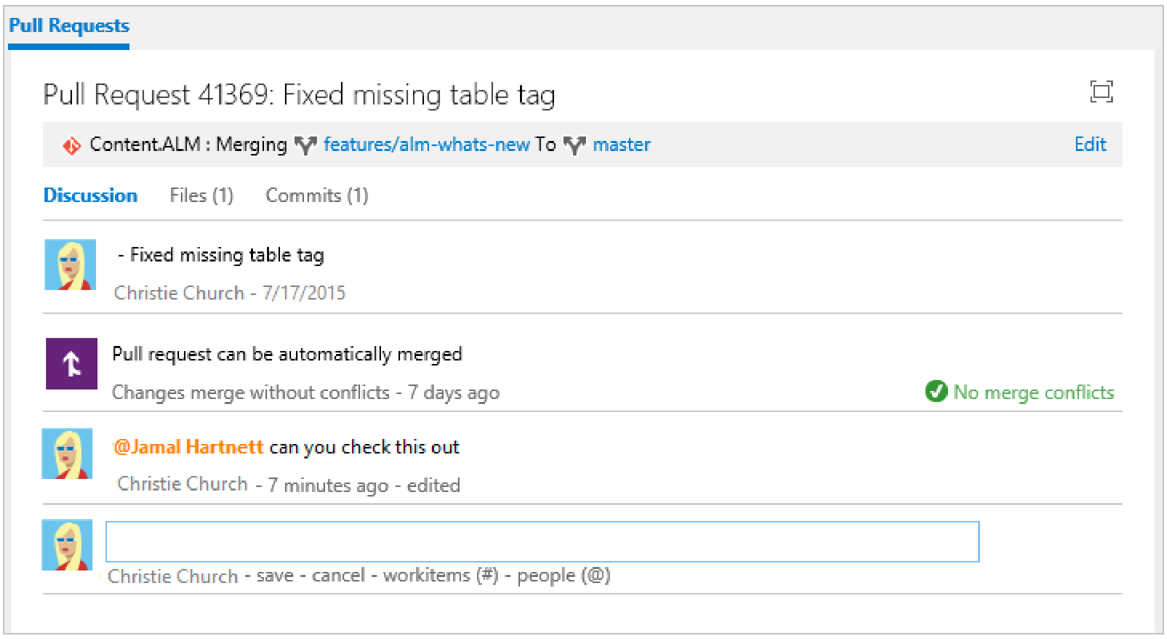
Names of those that you mention appear in blue text. Choose the**@mention link name** to open the user's contact card, which can provide you additional context for why they were pulled into the conversation.



Upon completion of your selection and text entry, your **@mention** user will receive an email alerting them about the mention.



When viewing their own mentioned names in conversations, users will notice that their own name is are highlighted in orange text.



You can use the **@mention** control in pull request discussions, commit comments, changeset comments, and shelveset comments. You can also use the **@mention** control in any text field on the work item form.

Related articles

- Work item form controls
- Pull requests

Use #ID to link to work items

8/10/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015.1

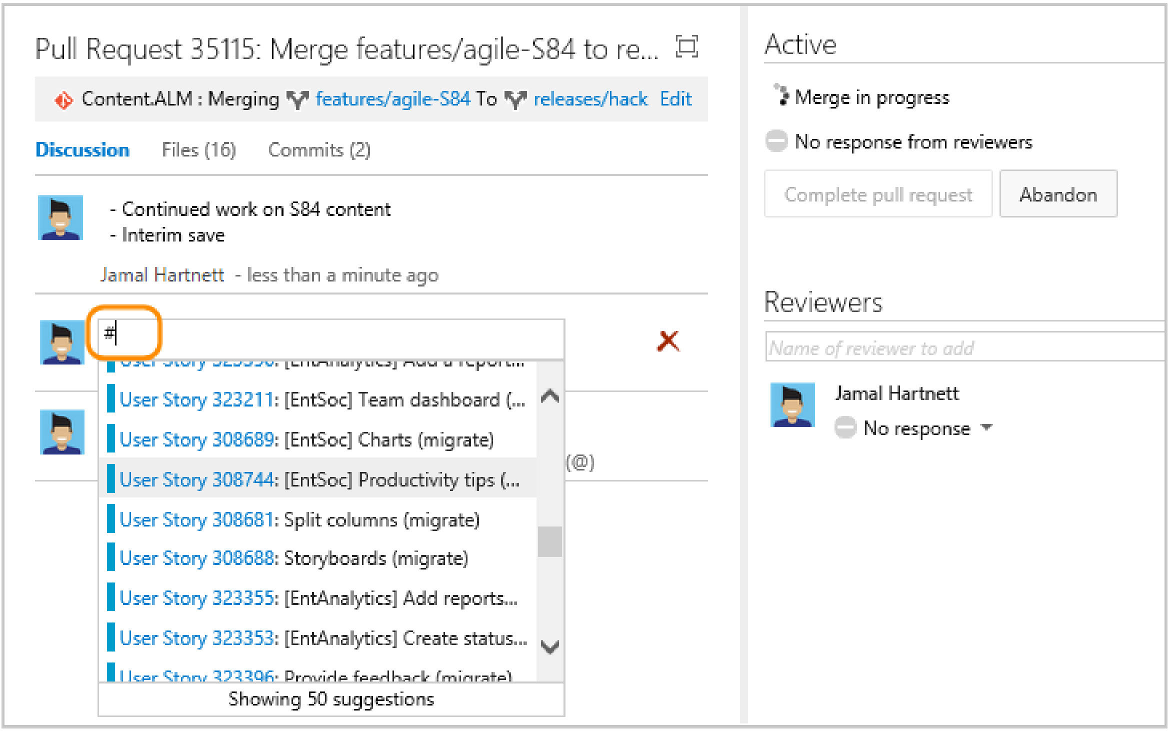
Use the **#ID** control to quickly link objects to work items.

> **NOTE**
> The **#ID** special control feature is available from TFS 2015 Update 1 and later versions.

Link a pull request to a work item

When leaving a code comment in a pull request, you can type **#** to trigger the **#ID** work item picker. The picker displays a list of 50 work items that you have recently modified or that are assigned to you.

You can narrow the list of suggested work items by entering keywords that match the work item type, ID, or title, or you can enter the exact work item ID.



To further filter the list, continue entering keywords until you've found a match. You can enter up to five keywords.

Link to work items in pull requests, comments, and commits

You can also use the **#ID** control in pull request discussions, commit comments, changeset comments, and shelveset comments.

> **NOTE**
> Requires TFS 2015 Update 2 or a later version.

Link to work items from a Wiki page

You can use the **#ID** control to link to a work item from within a Wiki page.

To learn more about the built-in wiki, see Add & edit wiki pages.

Related articles

- Link work items
- Save work with commits
- Pull requests
- Check in your work to the team code base

Work across projects

8/9/2018 • 6 minutes to read

VSTS | TFS 2018 | TFS 2017

When you work on several projects, you can use select pages to support your cross-project work. These pages provide quick access and filter functions to support your work within a single project or work you're doing across several projects.

For example, you can quickly access and navigate to the following artifacts defined across projects.

- New navigation
- Previous navigation

- **Projects**: Team projects and teams within the projects that you work in
- **Work items**: Work items assigned to you
- **Pull requests**: Pull requests you've initiated or that are relevant to you across all team projects you work in
- **Favorites**: Artifacts—such as projects, teams, repositories, shared queries, and more —that you've favorited

New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

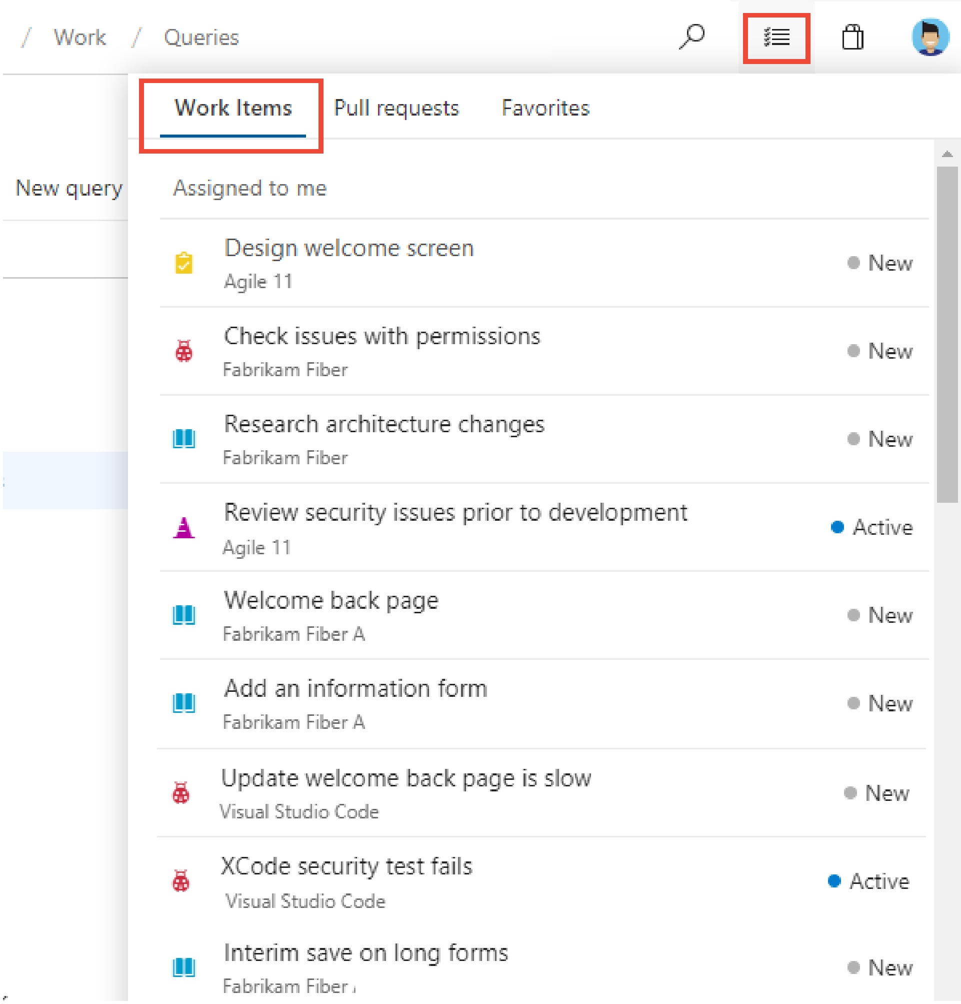
To switch to another project, see Switch project, repository, or team.

View and open work items

> **NOTE**
>
> Choose **Previous navigation** when you see a top-level blue bar. Choose **New navigation** if you see a vertical sidebar or if you enabled the **New Navigation** preview feature. The vertical sidebar, along with other navigational features, is enabled when the **New Navigation** preview feature has been enabled for the signed-in user or the VSTS organization. To learn how to use the web portal effectively, see Navigation Basics.
>
> For on-premises TFS users, choose **Previous Navigation** for guidance.

- New navigation
- Previous navigation

To view work items assigned to you across projects, choose the inbox icon, and then choosing **Work Items**.



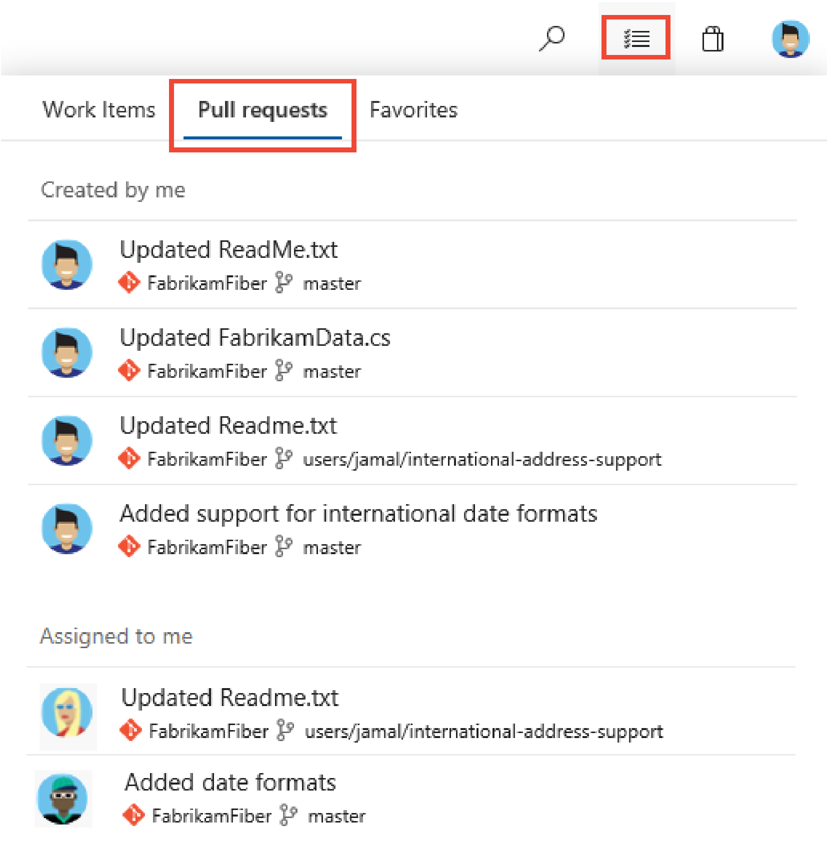
To view work items assigned to you for a specific project, open **Work>Work Items>Assigned to Me**.

New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

View and open pull requests

- New navigation
- Previous navigation

To view pull requests you created or that are assigned to you across projects, choose the inbox icon, and then choosing **Pull requests**.



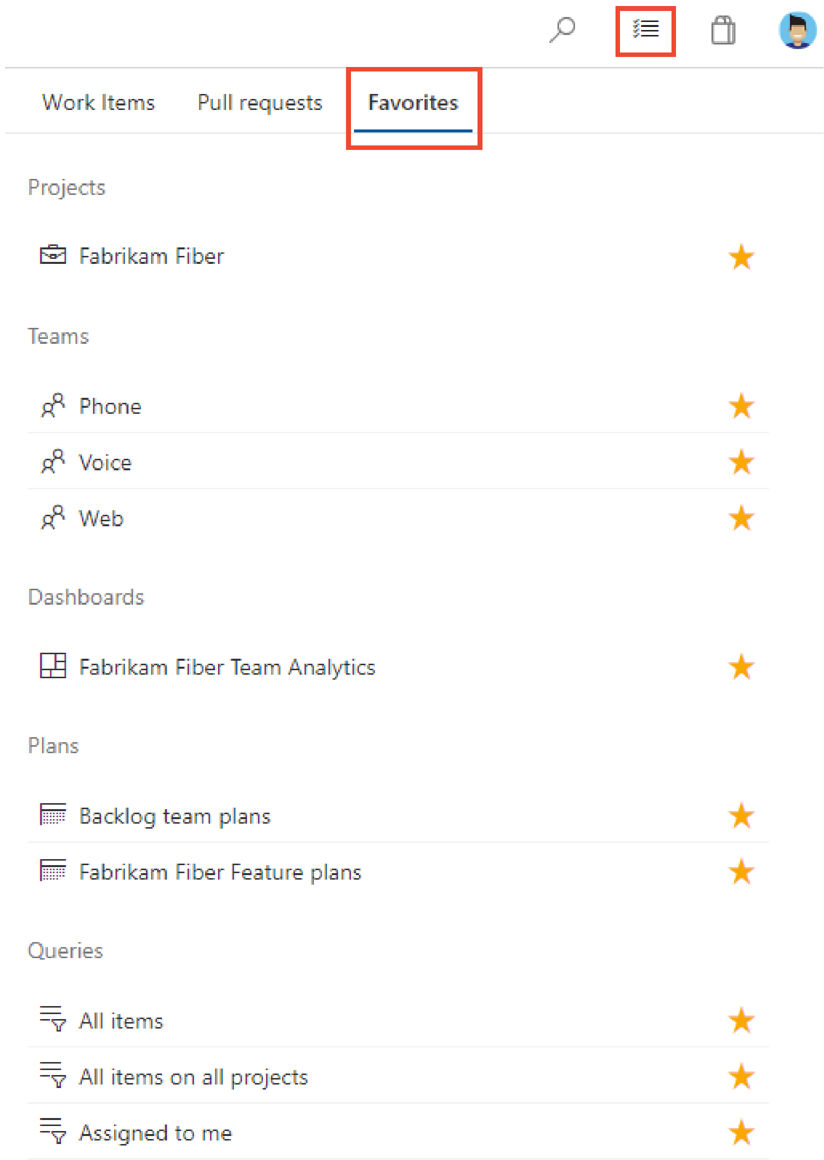
New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

View and open favorites

You can view favorites you've set across projects. To learn more about working with favorites, see Set personal or team favorites.

- New navigation
- Previous navigation

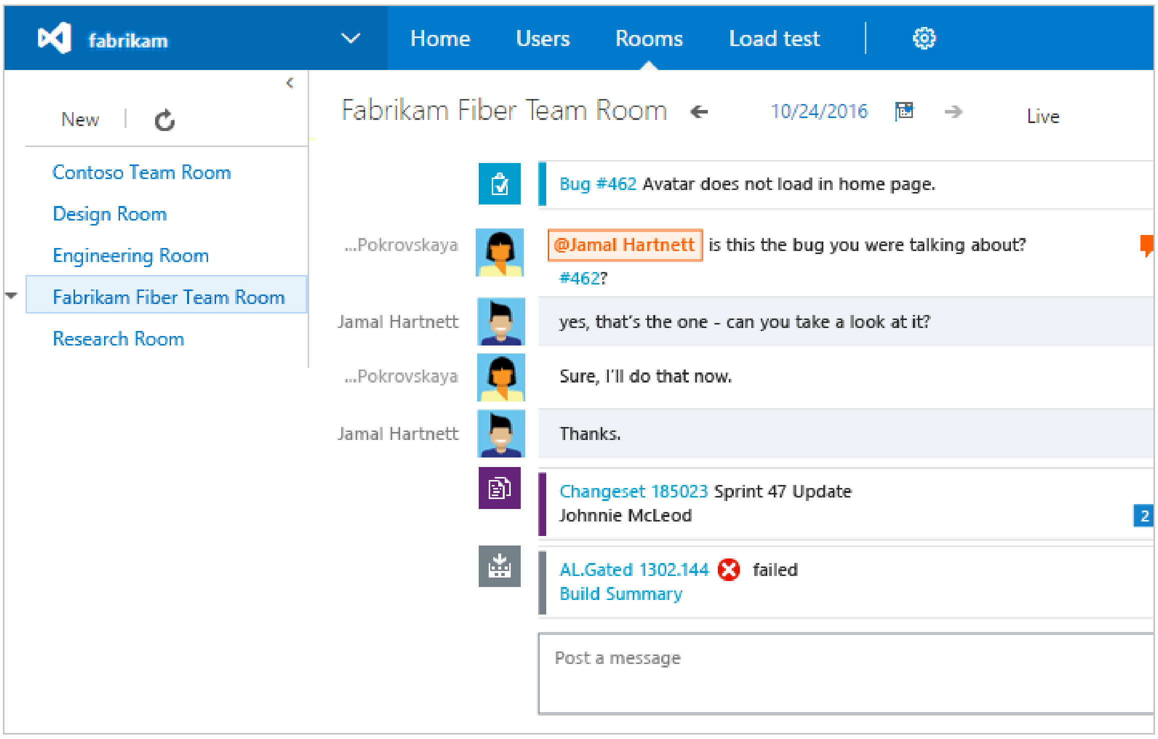
Choose the inbox icon, and then choosing **Favorites**.



New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Open a team room

To open a team room, choose **Projects>Rooms**. You'll see all the team rooms defined for collection of projects. Choose the name of a team room which you have access. You only have access to those team rooms of which you are a team member.



To learn more about team rooms, see Collaborate in a team room.

> **NOTE**
>
> Team Rooms have been deprecated for VSTS and TFS 2018 and later versions as described in this blog post, Deprecation of the Team Rooms in VSTS and TFS. Several good solutions are available that integrate well with TFS that support notifications and chat, such as Microsoft Teams and Slack. As a result, the Team Room feature will be deprecated from both Visual Studio Team Services (VSTS) and Team Foundation Server (TFS).

Related articles

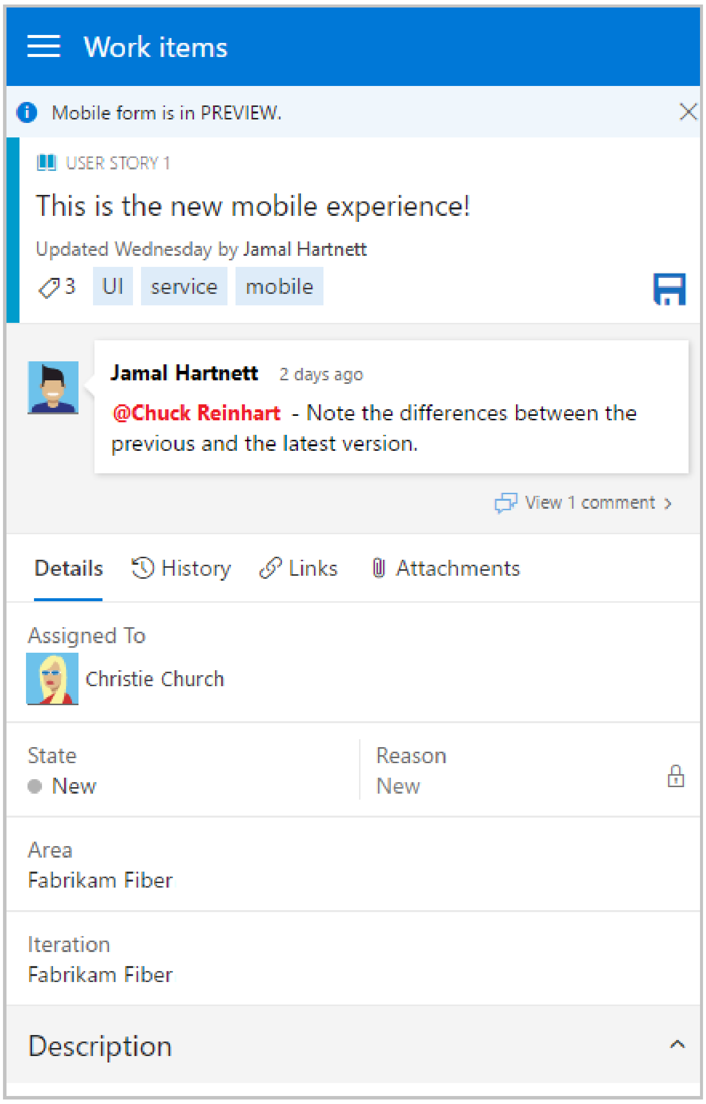
- Enable preview features
- Connect to team projects
- View and update work items via the mobile browser

View and update work items via the mobile browser

8/10/2018 • 3 minutes to read

VSTS | TFS 2018

With the mobile browser and work item form, you gain on-the-go features to stay on top of the latest updates made to work tracking. When you click any work item link on your mobile device, it will open a mobile-friendly version of the work item. From there, you can update the work item or access all work items assigned to you or that you're following.



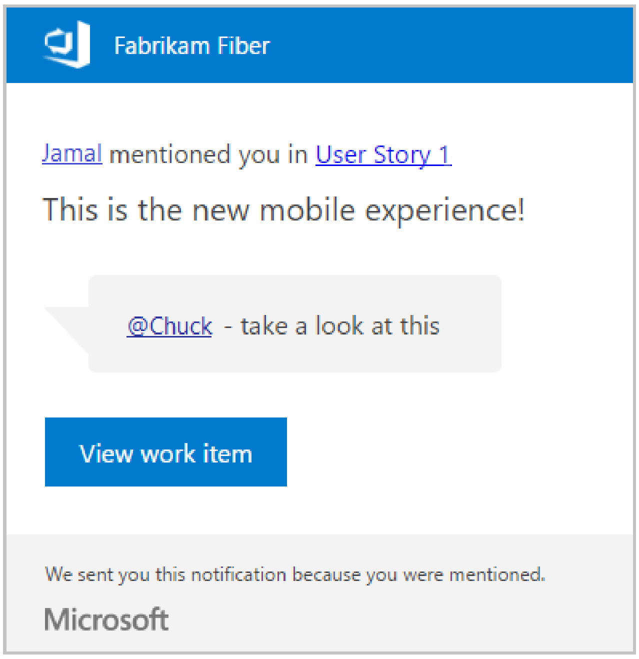
> **NOTE**
>
> The mobile browser supports work tracking in VSTS OR TFS 2018 and later versions. To sign up for VSTS, go to VSTS. The mobile browser is not an app, but a mobile view into select features. There is nothing to download. You access the mobile browser by clicking a link from a work item you receive in your mobile email application.

> **NOTE**
> The mobile browser is available for TFS 2018 and later versions. To download TFS 2018, see the TFS 2018 Release Notes. The mobile browser is not an app, but a mobile view into select features. There is nothing to download. You access the mobile browser by clicking a link from a work item you receive in your mobile email application.

Open the mobile work item form

The mobile work item form will open when you click **View work item** from an email you receive from your mobile device. You'll receive this type of email under these circumstances:

- Changes were made to a work item you're following
- You were **@mentioned** in a discussion
- A notification is sent based on the work item alerts you've set using Manage personal notifications.



Update a work item

Within the mobile form, you can do almost everything you can do from the web portal form. Here are the actions you can take in the order they appear in the mobile form:

- Add and remove tags
- View and add to the discussion, click on the comment to add to the discussion
- View and update any field within the form (Assign to, State, Area, Iteration, Description, and more)
- View and open a link within the Development section
- View History
- View and open a link from the Links tab
- Open and add an attachment from the Attachments tab

Actions not available to you:

- You can't initiate a development operation
- You can't add a link

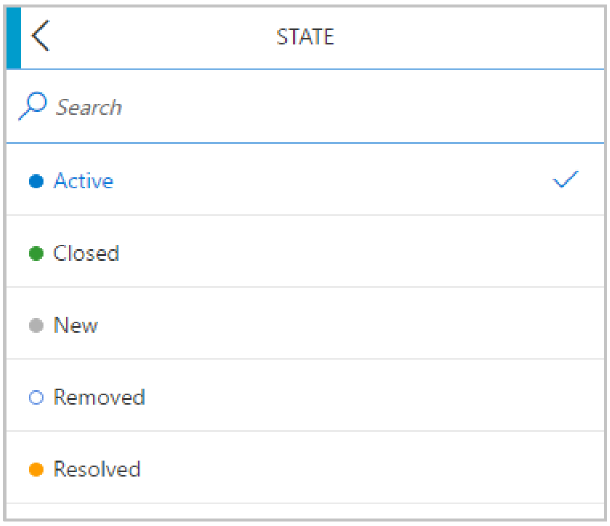
Interact with mobile form controls

Mobile form controls operate as follows:

- Click any field to edit it and the form changes to a full-screen experience. For example, some of the most common actions such as changing the state of an item, moving to a different area path, adding an attachment, and creating/removing tags are all supported.
- When done, click the return option.
- Remember to click the save icon to save your changes!

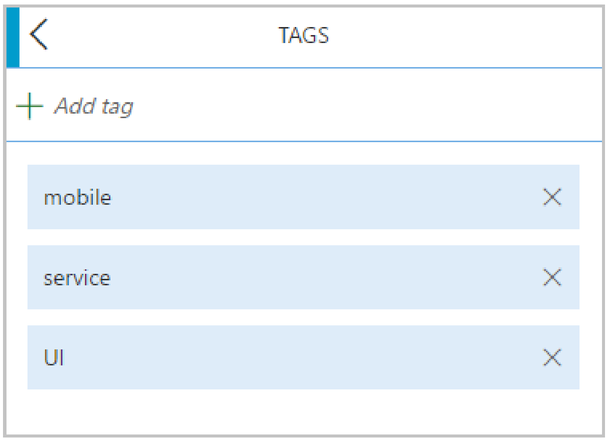
Update status (change State)

To update the state, click the state you want.



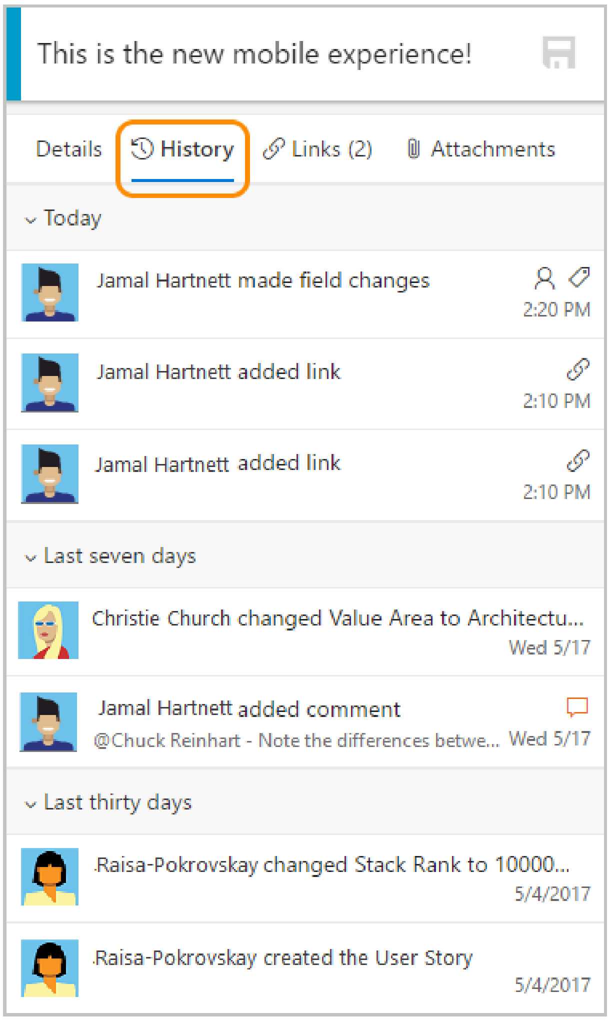
Add or remove tags

To add a tag, type the text you want.



View history

Click the History tab to view history.



View and open work items in your activity lists

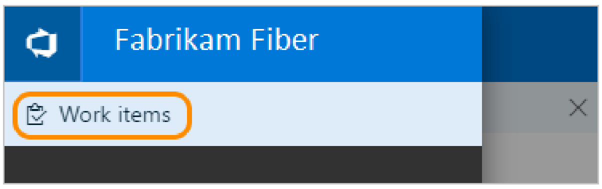
From within the mobile work item form, you can access your work items. The mobile browser allows you to view and open work items which fall into these categories:

- **Assigned to me**: lists all work items assigned to you
- **Following**: lists all work items that you have elected to follow
- **My activity**: lists all work items that you have recently viewed or updated.

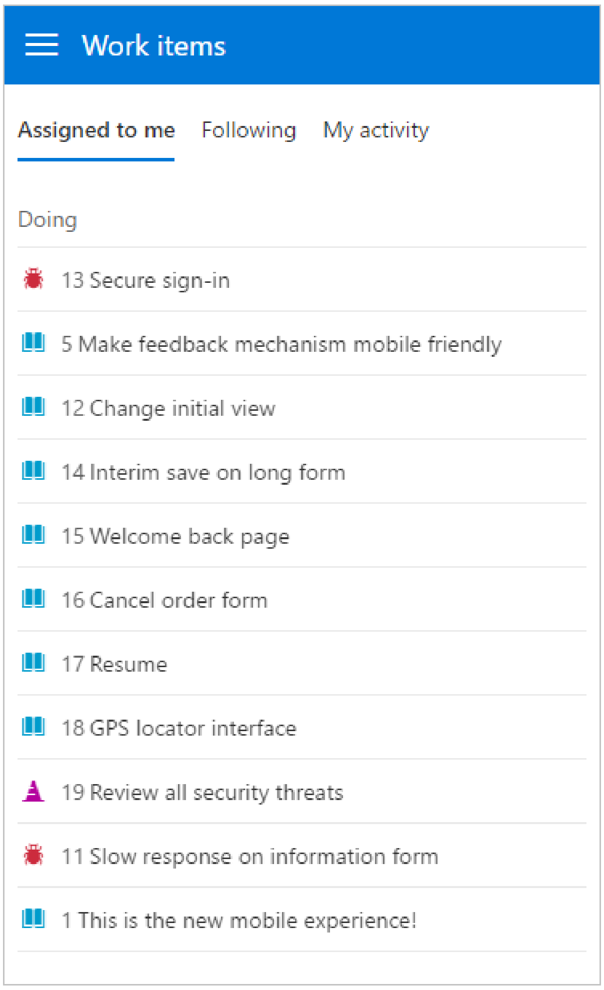
The lists available from each page span all team projects that you work in.

To access a list, first click the list control from the work item form you've opened.

Then, click **Work items**.



The browser opens to the **Assigned to me** page. From there, you can choose **Following** or **My activity** to access the other pages. To learn more about the **Work Items** view, see View and add work items.



Related articles

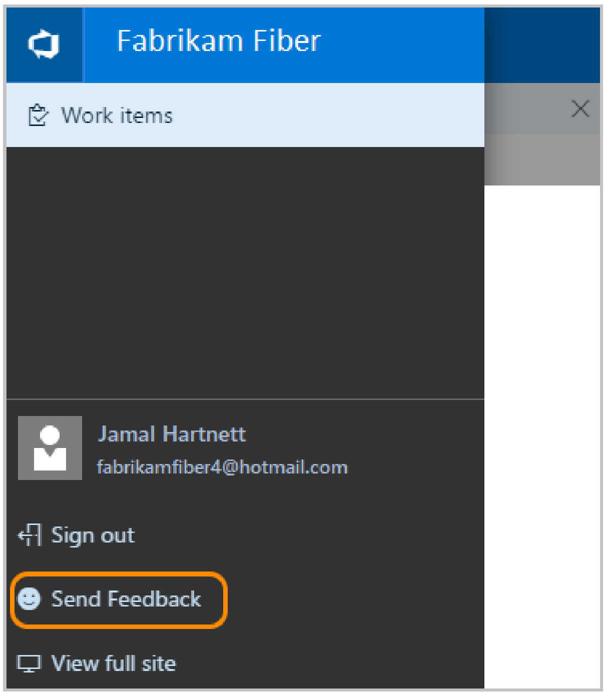
Additional experiences are in the works to improve and expand on the mobile experience. For more information, see the blog post: The mobile work item form (preview).

- Set personal notifications
- Set team notifications
- Follow a work item

Provide feedback for the mobile experience

Help us improve the mobile experience.

To provide feedback, click the list control from the work item form and then click **Send Feedback**. To complete the feedback, select either the smile or frown and optionally enter a comment.


Fabrikam Fiber
Work items
Jamal Hartnett
fabrikamfiber4@hotmail.com
Sign out
Send Feedback
View full site

Manage your notifications

8/10/2018 • 3 minutes to read

VSTS | TFS 2018 | TFS 2017.1 | Previous versions

NOTE

Feature availability: This topic applies to VSTS, TFS 2017 Update 1, and later versions. If you work from an on-premises TFS 2017 or ealier versions, see Set alerts, get notified when changes occur. For on-premises TFS, you must configure an SMTP sever in order for team members to see the Notifications option from their organization menu and to receive notifications.

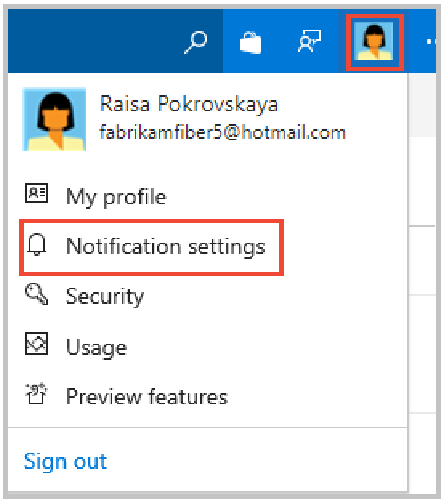
As changes occur to your code base, builds, work items, and other operations, you can receive email notifications. For example, you can set an alert to be notified whenever a bug that you opened is resolved or a work item is assigned to you.

Use this topic to learn how to:

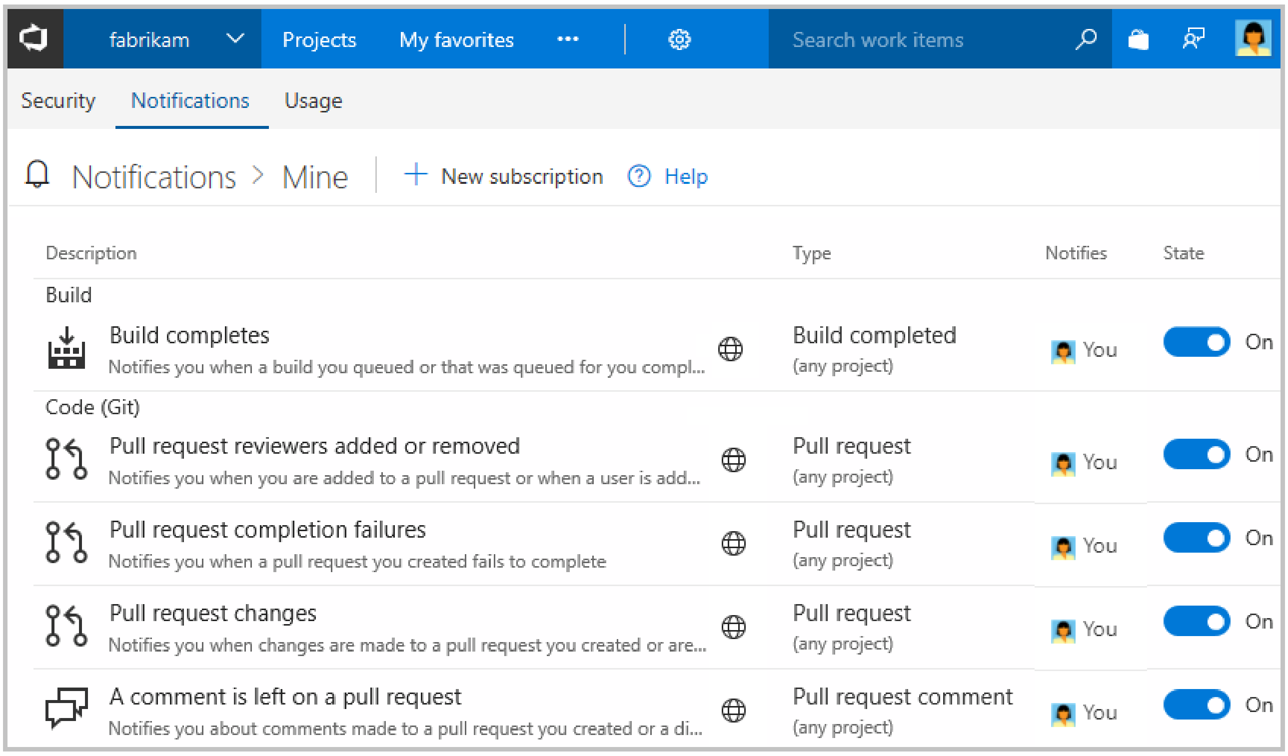
- View your notifications
- Add a custom subscription
- Unsubscribe or opt out of a team or project subscription

View your personal notifications

From the web portal, click the icon with your initials or picture icon, and select **Notification settings** from the drop-down menu.



This view shows all subscriptions that you have created or that have been created by an administrator. Subscriptions let you control what you are notified about. Those notifications you're subscribed to are indicated with the State as **On**.

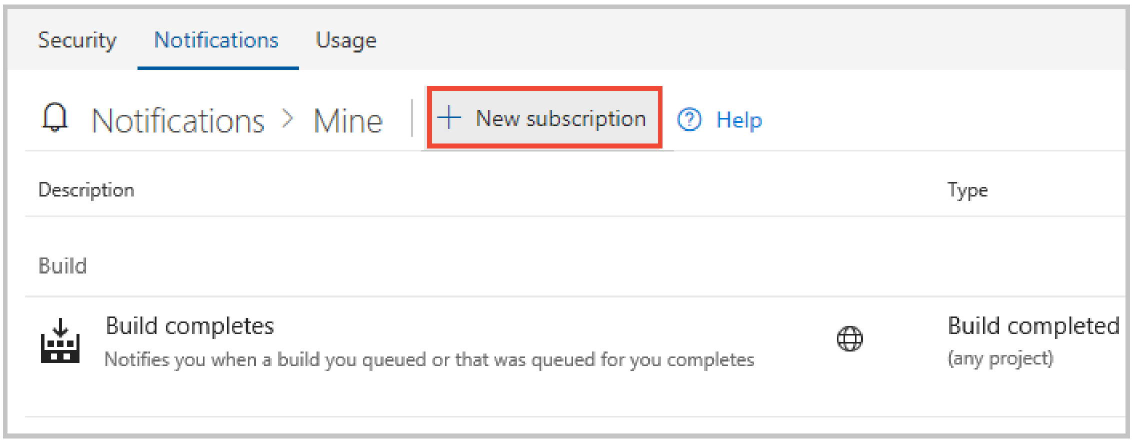


A subscription can be just for you, or if you are a team admin, can be shared by everyone in the team.

Add a custom subscription

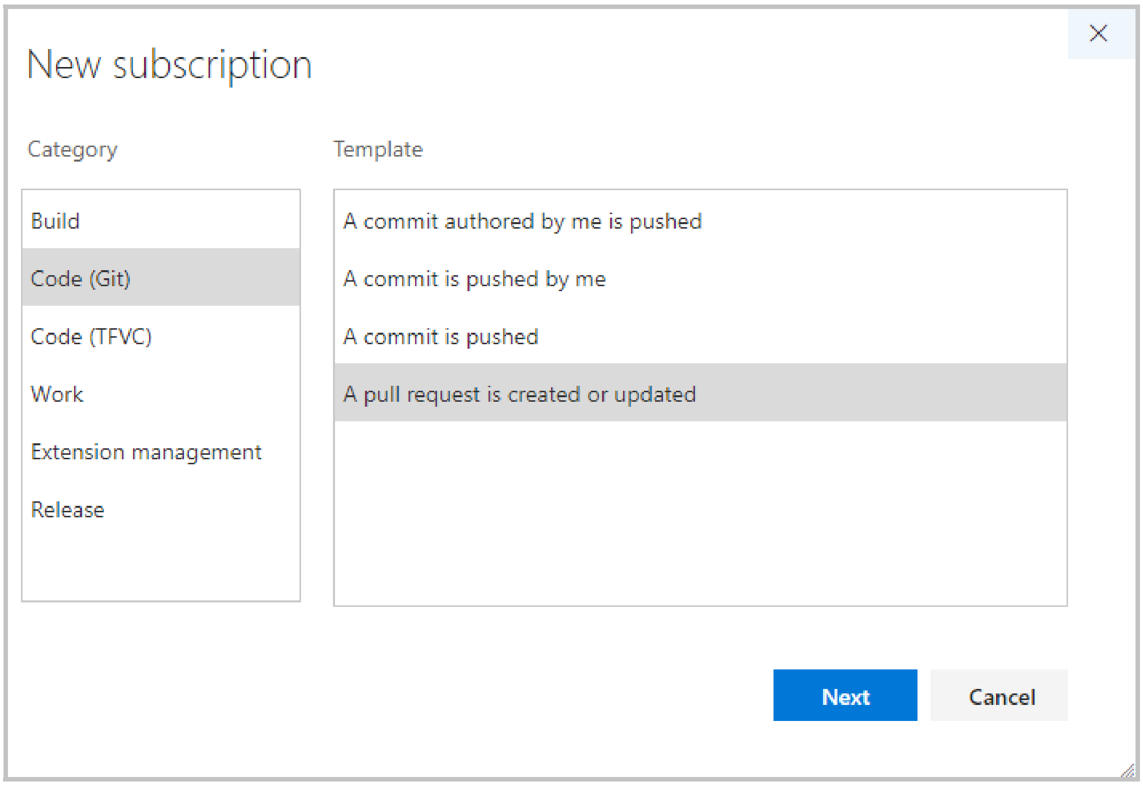
With custom personal subscriptions, you can define precise criteria for the events you want to receive notifications for. In contrast to a default subscription which will only notify the users or groups directly associated with an event, a custom subscription can notify you about any event.

1. From your Notifications page, click **New subscription**.

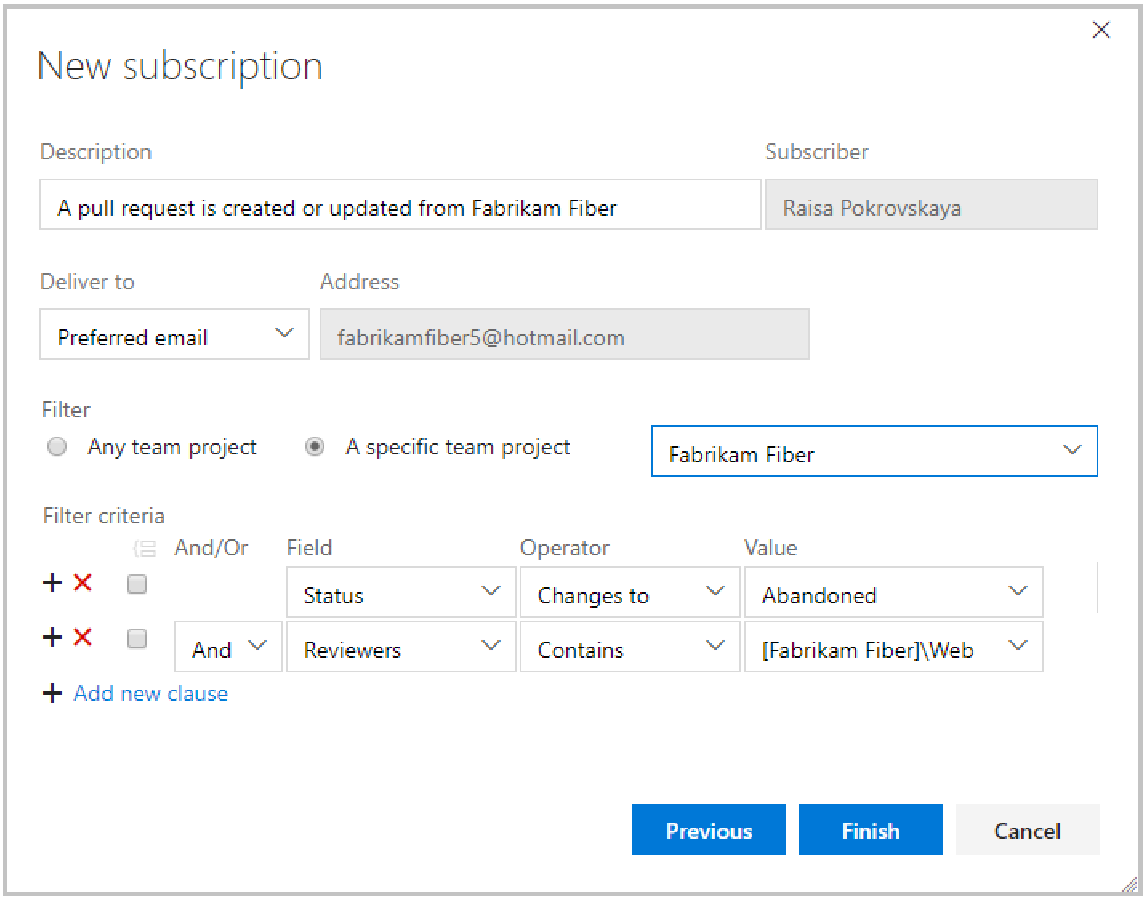


2. Choose the category and template you want to use. For a list of supported templates, see Default and supported notifications.

   Here we choose to get notified when a pull request is created within a specific project, Fabrikam Fiber.



3. Modify the description to help you identify the subscription later. Also choose an email address for notifications to be delivered to. By default, your preferred email address is used. optionally, include one or more fields to further specify the event criteria.



> **NOTE**
>
> The fields available for filtering event criteria differ depending on the category and template you select.

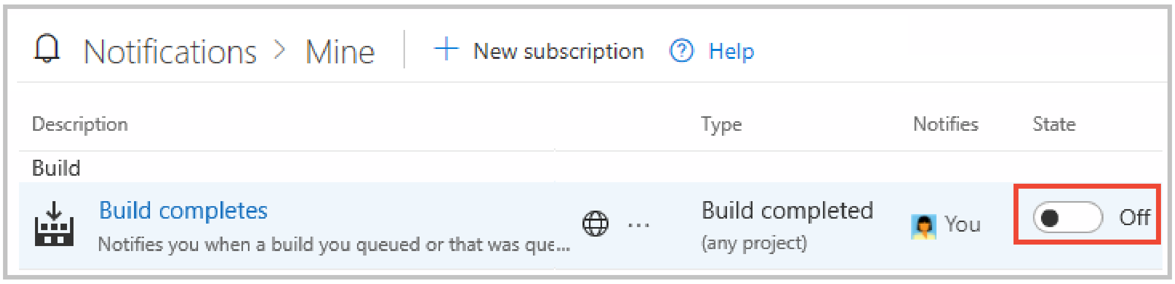
4. Click **Finish** when done. The subscription now appears in the list under the category you selected.

Unsubscribe or opt out of a team or OOB subscription

You can choose to not receive notifications for certain team subscriptions by opt'ing out of the subscription.

To unsubscribe from any notification, even one that you've defined, slide the State **On/Off** indicator to the Off position.

For example, here we turn off the Build completes subscription.



> **NOTE**
> Whether you are an administrator or not, toggling a shared team subscription from your notification settings only impacts you and not other team members.

Related notes

- Default and supported notifications
- Follow a specific work item
- Manage notifications for a team
- Change your preferred email address

Limitations

- The user interface no longer support creating plain text email subscriptions.

Set your preferences

8/10/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015

From your profile page, you can change your picture or other account preferences. Specifically, you can set the following:

PROFILE	SECURITY	OTHER
- Your picture - Your display name - Your preferred email - Locale settings	- Personal access tokens - Alternate authentication credentials - OAuth authorizations - SSH public keys	- Manage personal notifications - Usage - Enable preview features

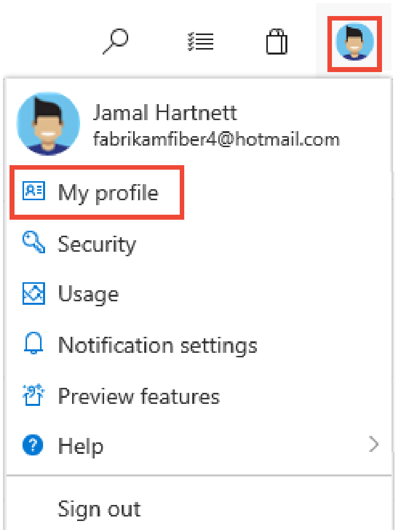
Locale settings include language, date and time pattern, time zone, and user interface theme.

Change profile settings

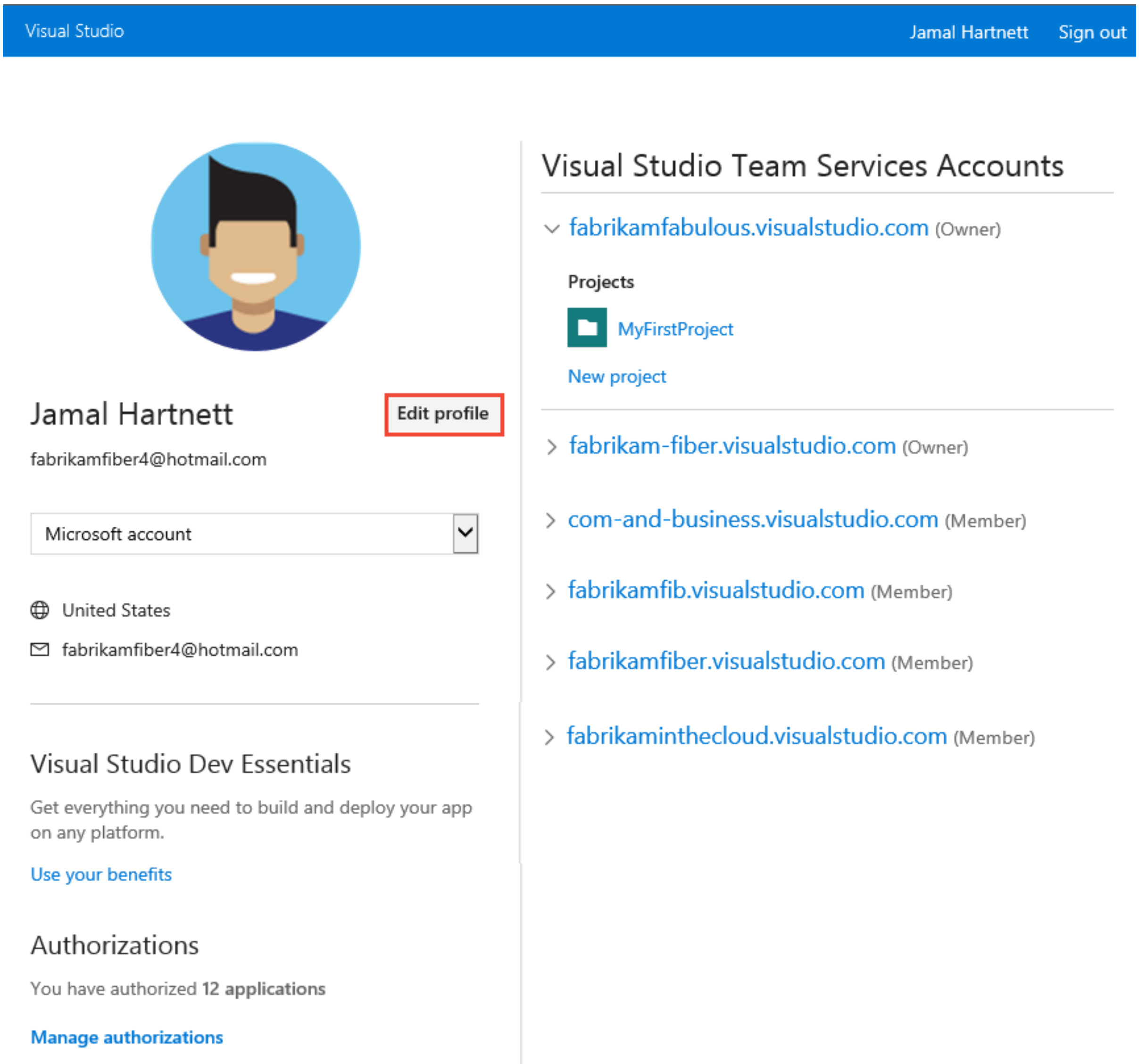
> **NOTE**
>
> Choose **Previous navigation** when you see a top-level blue bar. Choose **New navigation** if you see a vertical sidebar or if you enabled the **New Navigation** preview feature. The vertical sidebar, along with other navigational features, is enabled when the **New Navigation** preview feature has been enabled for the signed-in user or the VSTS organization. To learn how to use the web portal effectively, see Navigation Basics.
>
> For on-premises TFS users, choose **Previous Navigation** for guidance.

- New navigation
- Previous navigation

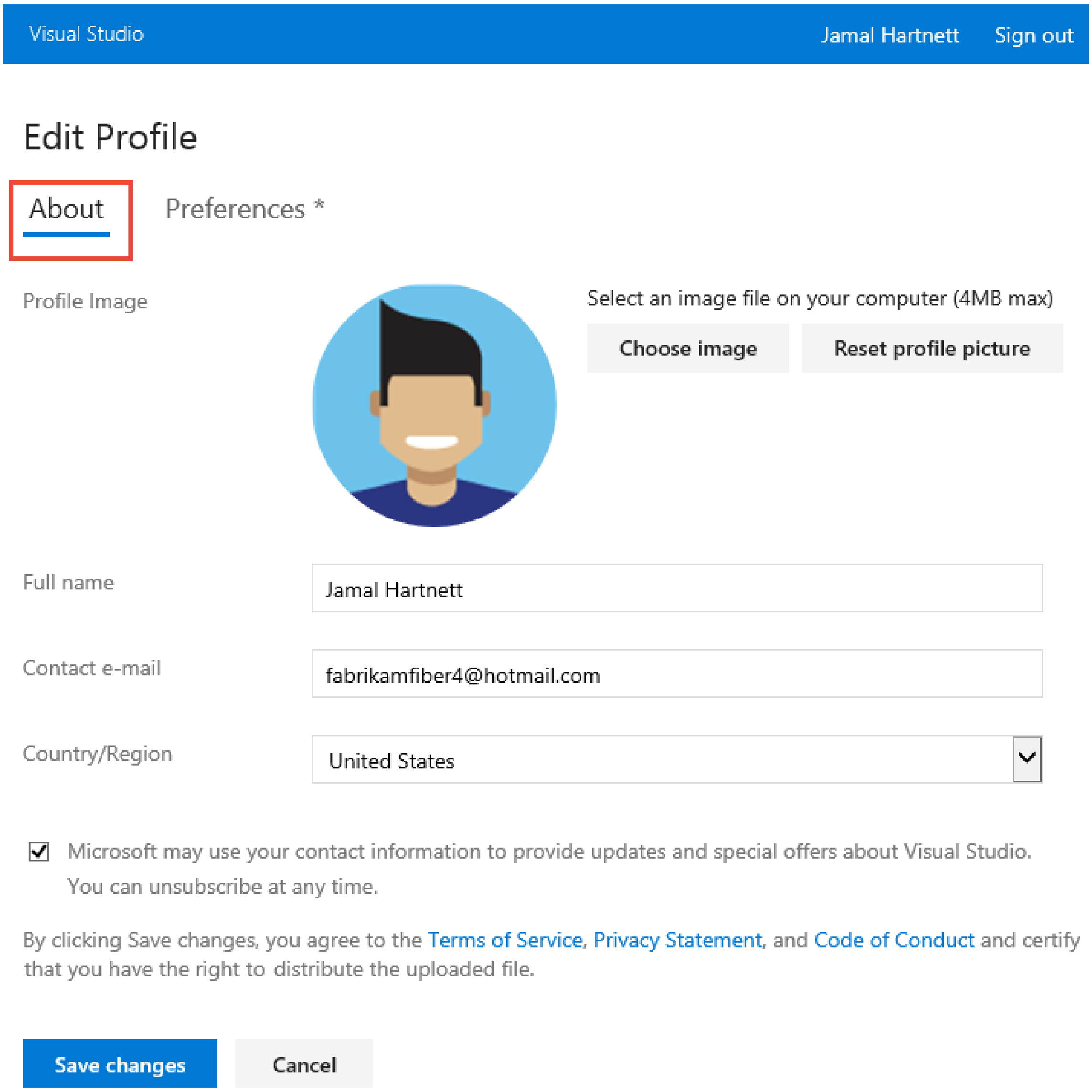
1. To change your user preferences, open your profile menu.



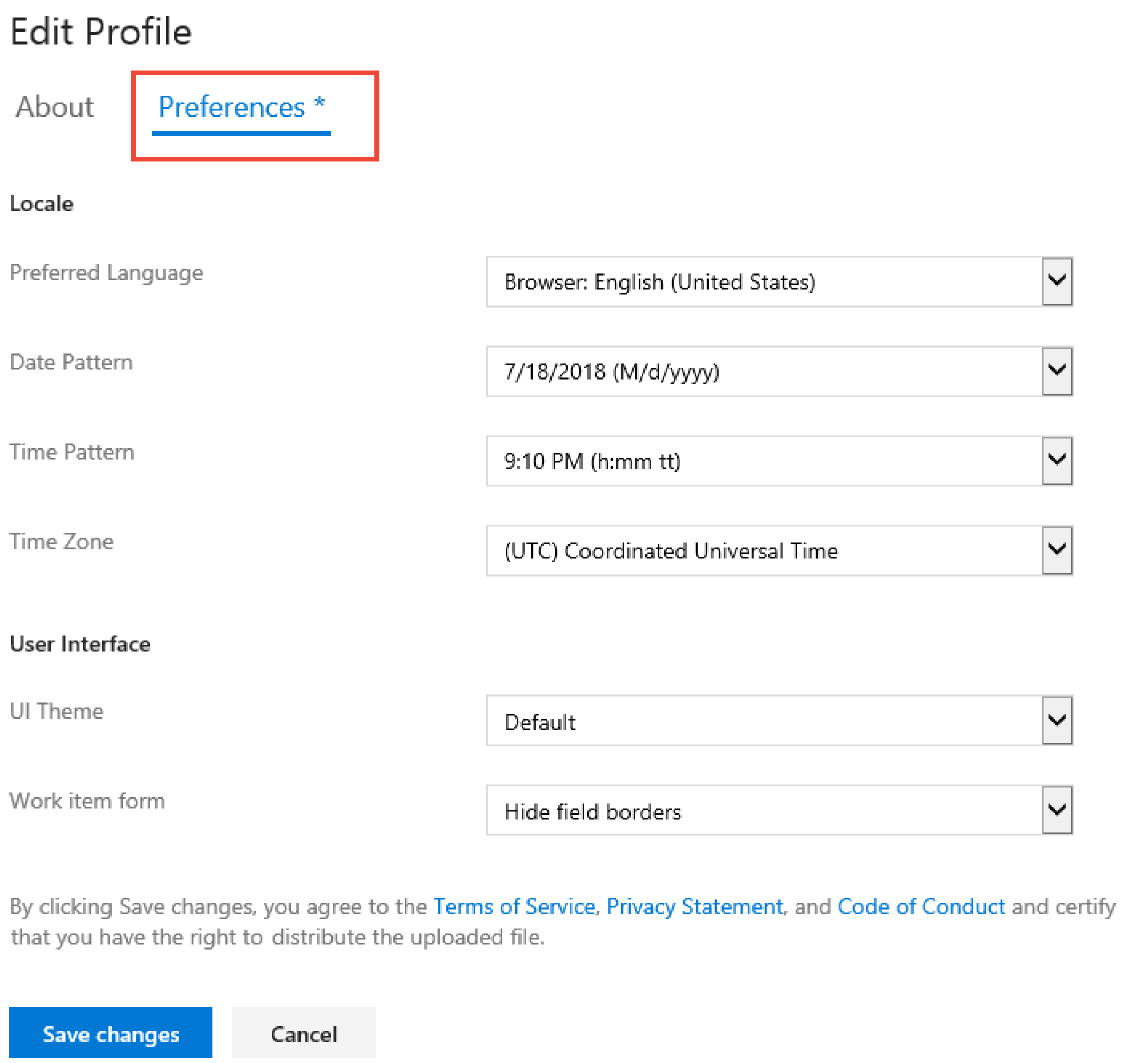
2. Choose **Edit profile**.



3. From the **About** page, you can change your profile picture, change your display name, contact information, and country.



4. From the **Preferences** page, you can change your preferred language, date and time pattern, time zone, UI theme, and whether or not borders appear on work item forms for fields.



New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Related articles

- Set favorites

2 minutes to read

Troubleshoot connecting to a team project in VSTS or TFS

8/9/2018 • 5 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

Troubleshoot VSTS connectivity

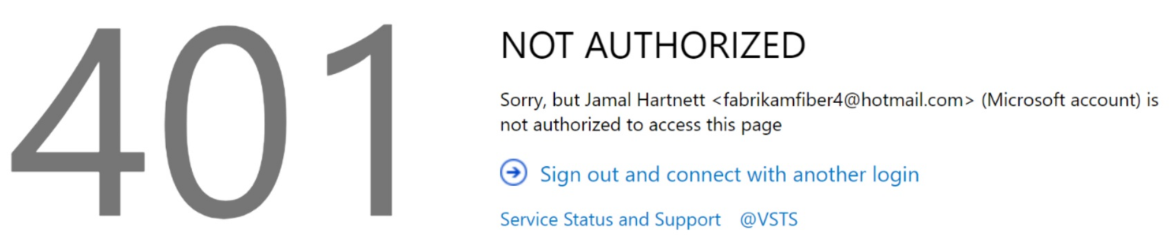
As a first step in resolving connectivity issues with VSTS, do the following:

1. Sign out of your browser. You can do this by choosing the Visual Studio Sign out link.
2. Delete your browser cookies. For most browsers, you can do this by entering CTRL+SHIFT+DEL.
3. Open Internet Explorer (IE) and delete cookies. Visual Studio IDE uses IE cookies.
4. Close all browsers and the Visual Studio IDE.
5. Use a private browser session to retry the connection to VSTS. If the issue is with VS IDE, remove the connection to VSTS after doing above and then re-add it.

Troubleshoot signing in to VSTS

There are two types of identities that can sign in to VSTS - Microsoft accounts (MSA) and Azure Active Directory (Azure AD) accounts. Depending on your account, you may experience one of the following errors.

401 - Not Authorized



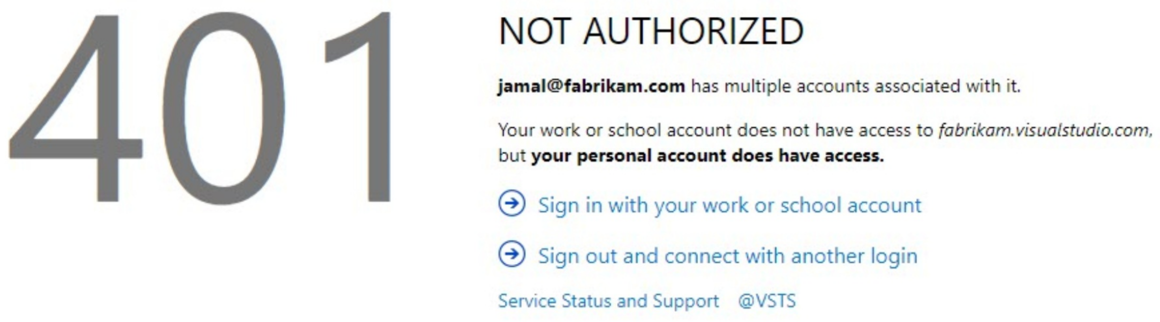
The most common error page encountered is the "401 Not Authorized" error. This occurs when your identity does not have permission to enter a VSTS organization. Common reasons for this include:

- Your identity is not a member of the target VSTS organization
- Your identity has an invalid or missing license assignment

If you believe you are a member of the organization but are blocked by this error page, feel free to contact customer support.

Scenario 1: Your work or school Azure AD account does not have access, but your personal MSA account does.

401 - Work or school, or Personal account



This is a highly specific 401 error case, where there exists both a personal Microsoft account (MSA) and a work or school account (Azure AD) with the same sign in address. You have signed in with your work or school account, but your personal account is the identity that has access to the VSTS organization.

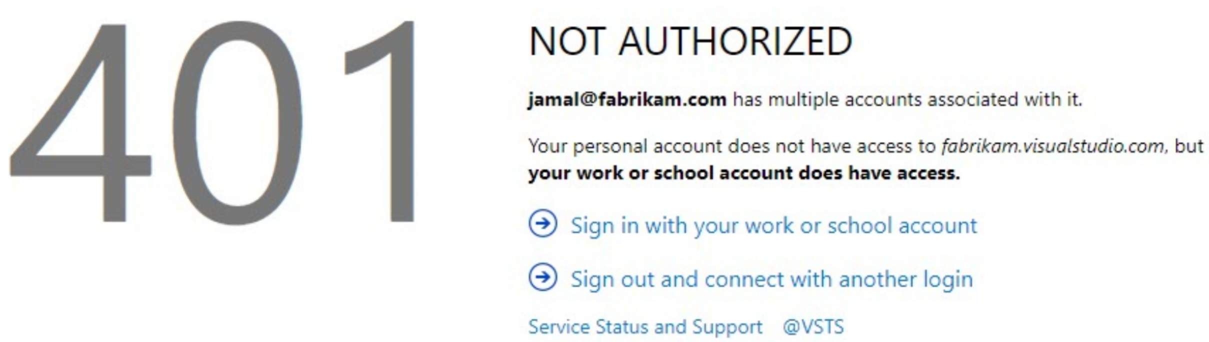
Mitigation

In some cases, you may be unaware that you have two identities with the same sign in address. The work or school Azure AD account may have been made by an administrator upon onboarding to Office365 or Azure AD. You must choose the "sign in with your personal MSA account" link to sign you out of your current work or school Azure AD account and begin a sign in prompt as the personal MSA. After authentication, you should have access to the VSTS organization.

[!Tip] To never see this prompt again, you can rename your Microsoft account. This will make it so that only one identity (your work or school account or Azure AD account) for your sign in address.

Scenario 2: Your personal MSA does not have access, but your Azure AD account does. This is the opposite version of the above 401 error page. In this case the personal account (Microsoft account identity) did not have access to the VSTS organization and the work or school account (Azure AD identity) does. The same guidance above applies, only in reverse.

> 401 - Work or school, or Personal account



Still unable to access your organization? In cases where you enter your credentials correctly, but instead of redirecting to your VSTS organization you are redirected back to the original sign in page, we recommend clearing all cookies and re-attempting to sign in. If that does not fix the issue, please contact customer support.

Troubleshoot TFS connectivity

Here's a list of the most frequently encountered connection problems and what to do about them. Start at the top and follow it in the order indicated.

1. Verify that you have required permissions.

   If the errors that you receive indicate read-only or blocked actions, you might not have permission to act on the data.

2. Verify that your computer is connected to the network and can access network resources.

3. Verify that TFS hasn't been taken offline. Talk with your TFS administrator.

4. Verify whether your team project has been moved to another team project collection in TFS. If it has been moved, you must create a connection to the new server name.

For additional troubleshooting tips, see TF31002: Unable to connect to this Team Foundation Server.

Switch between different organizations in VSTS

When you use two or more VSTS organizations that are linked to Azure Active Directories, such as organizations that are created from the Microsoft Azure Portal, the sign-out function may not work as expected. For example, you can't switch between different organizations to connect to multiple organizations that are linked to directory tenants.

When this problem occurs, a blank screen flashes several times followed by one of the error messages below after you connect to or add a new connection in "Connect to Team Foundation Server" dialog box:

> TF31003: Either you have not entered the necessary credentials, or your user account does not have permission to connect to the Team Foundation Server
>
> TF31002: Unable to connect to this Team foundation Server

To resolve this problem, apply Visual Studio 2013.2 or install a later version. Go to the Visual Studio download website.

For a workaround in which you delete your browser cookies, see the following support article, You can't switch between different organizations in Visual Studio Online.

Connect to a TFS with Secure Sockets Layer (SSL) configured

If you connect to a TFS instance that has SSL configured, then you'll need to install a certificate and clear the client cache. For details, see Set up HTTPS with Secure Sockets Layer (SSL) for TFS, Configuring Client Computers.

Clear the cache on client computers

When the on-premises TFS configuration changes, such as when moving or splitting a project collection, you might have to clear the cache.

1. Log on to your client computer for Team Foundation by using the credentials of the user whose cache you want to clear.

2. Close any open instances of Visual Studio.

3. Open a browser, and browse to one of the following folders, depending on the operating system that is running on the client computer:

   - **Windows 10** *Drive*:\Users<i>UserName\AppData\Local\Microsoft\Team Foundation\6.0\Cache

   - **Windows 8** *Drive*:\Users<i>UserName\AppData\Local\Microsoft\Team Foundation\4.0\Cache

   - **Windows Vista or Windows 7** *Drive*:\Users<i>UserName\AppData\Local\Microsoft\Team Foundation\2.0\Cache

4. Delete the contents of the Cache directory, including all subfolders.

TF31002: Unable to connect to this Team Foundation Server {0}. Team Foundation Server URL: {1}

8/9/2018 • 5 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

You might receive this error when you try to connect to VSTS or an on-premises Team Foundation Server from Visual Studio.

You receive this error when you try to connect to VSTS

PROBLEM	RESOLUTION
You don't have an active account or license.	Check with your administrator that you're a member of the account and have an active, valid license. See Assign licenses to users for details.
Your VSTS organization is connected to the Azure Active Directory.	When your VSTS organization is connected to a directory that is associated with an Office 365 or Microsoft Azure subscription, only members in the directory can access the account. Check with your directory administrator to have them create an organizational account for you or add your account to the directory as external member.
You can't switch between different organizational accounts.	If you work with several VSTS organizations that connect to different directories, such as accounts that are created from the Microsoft Azure Preview Portal, the sign-out function might not work as expected. For example, you can't switch between different organizational accounts to connect to multiple accounts that are linked to directory tenants. When this problem occurs, you see a flashing blank sign in dialog box several times. Then, you receive either TF31002 or TF31003 error after you connect to or add a new connection in "Connect to Team Foundation Server" dialog box. To resolve this problem, apply the most recent Visual Studio update . To learn more, see KB Article ID 2958966, You can't switch between different organizational accounts in Visual Studio Online.
You want to sign in to VSTS from Visual Studio using different credentials.	See Connect to projects, Sign in with different credentials.

You receive this error when you try to connect to an on-premises TFS from your client computer

If you determine that you're receiving this error from one computer but not others, or others aren't receiving this error, then check the problem resolutions outlined below.

PROBLEM	RESOLUTION
Your password has expired.	Verify that you typed your user account and password correctly, and that your password hasn't expired.
You've entered an incorrect server URL.	Verify that you have typed the server URL correctly including the server name, port number, and protocol (http/https). See Connect to projects to learn more.
The TFS configuration has changed.	If the configuration for the on-premises TFS has changed, you must create a new connection. You might also need to clear the client cache.
You work remotely and need to connect to a TFS Proxy server to check in files to Team Foundation version control.	You need to configure Visual Studio to connect to TFS Proxy.
You're connecting to a later version of TFS than your Visual Studio client version.	Your version of Visual Studio or Team Explorer might be incompatible with Team Foundation Server. You might need to install one or more GDR packs. See Requirements and compatibility for details.
Your firewall is blocking TFS services.	See Allow a program to communicate through Windows Firewall.
Visual Studio stops responding when you run a query in Visual Studio.	Your computer might be configured to bypass the proxy server. You should verify the configuration of the BypassProxyOnLocal setting on your computer. For more information, see BypassProxyOnLocal Configuration.

Several users receive this error when they try to connect to an on-premises TFS

If the problem occurs on more than one computer, you should contact your TFS administrator to confirm whether the server is operational and available on the network.

As an administrator, you should check the event logs for the application-tier server to try to pinpoint the problem. Also, you can use the following table to determine whether the server is misconfigured. In the table, problems that are more likely to occur appear first. Therefore, you should try the resolutions in the order in which they appear so that you increase the chance that you can solve the problem quickly.

PROBLEM	RESOLUTION
The *TFSService* account password has expired or is incorrect.	Many services for Team Foundation Server will stop running when the service account for Team Foundation has expired. For more information, see Change the service account or password for Team Foundation Server.
The application-tier server for Team Foundation is unavailable.	You should verify whether each required service is running. If a required service is not running, you must restart it. If necessary, set it to start automatically. For more information, see Stop and start services, application pools, and websites.
The network is unavailable.	You should verify whether your network is operational.
A website identity for Team Foundation is configured incorrectly.	You should verify or correct the server binding assignments that are made to websites for Team Foundation.

PROBLEM	RESOLUTION
Access to a website for Team Foundation has been restricted.	You should verify or correct restrictions that are made to those websites that are based on IP addresses and domain names.
The firewall or ports are configured incorrectly.	You should verify or correct port binding assignments for websites and port assignments for the firewall. First, you should open the administration console for Team Foundation, display the Application Tier page, and review the URL assignments. If necessary, you can click **Change URL** to modify the URL of a website. Next, you should verify the port assignments for Internet Information Services (IIS) and the ports that are allowed through the firewall. For more information, see Review Server Status and Settings and Verify or Correct Port Assignments.
Trust relationships between domains are not configured correctly.	If a group of users cannot access Team Foundation Server, you might have trust issues between domains.
When users connect to different versions of TFS from Visual Studio, for example, they connect to TFS 2012 and then TFS 2008, they can get the TF31002 error.	This can occur because the GUIDs for the TFS 2012 collection are the same as that for TFS 2008. This confuses the local client cache because it tries to maintain the same GUID based local cache for both the 2008 server and the new Project Collection in 2012. To fix this, you need to run the **TFSConfig ChangeServerID** command. See TFSConfig ChangeServerID command.

If the previous resolutions do not solve the problem, go to the MSDN Forums - Visual Studio Team System — Team Foundation Server - Administration.

Troubleshoot network connections and whitelist addresses

6/13/2018 • 2 minutes to read

VSTS

If you are having network connection issues to VSTS, using NuGet, or connecting from Visual Studio 2015 and later versions, it may be because your security appliances are blocking connections now that Visual Studio uses TLS 1.2.

To fix this issue, update the security appliances in order to support TLS 1.2 for the following connections:

List of URLs for sign-in and licensing connections:

- `https://management.core.windows.net`
- `https://login.microsoftonline.com`
- `https://login.live.com`
- `https://go.microsoft.com`
- `https://graph.windows.net`
- `https://app.vsspsext.visualstudio.com`

A more general list of URLs for signing in to VSTS and Azure:

- *.windows.net
- *.microsoftonline.com
- *.visualstudio.com
- *.microsoft.com
- *.live.com
- cdn.vsassets.io -- hosts our CDN content
- *.gallerycdn.vsassets.io -- hosts VSTS extensions
- static2.sharepointonline.com -- hosts some resources that we use in "office fabric" UI kit (fonts, etc).

NuGet connections:

- *.azurewebsites.net
- *.nuget.org

> **NOTE**
> Privately owned NuGet server URLs may not be included in the list above. You can check the NuGet servers you are using by opening up `%APPData%\Nuget\NuGet.Config`.

Get VSTS and TFS product support and provide feedback

8/10/2018 • 3 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

We're always working to improve VSTS, and we want you to be part of the process! Share your feedback and ideas with us, or join our communities.

Do you need to:

- **Get advice?** Visit StackOverflow for VSTS or TFS.
- **Report a bug?** Submit it through our Developer Community for VSTS or TFS.
- **Suggest a feature or a fix?** Submit your idea or issue through UserVoice for VSTS.
- **Find out what's new in VSTS?** Check out the current VSTS Release Notes. These notes are updated every three weeks.

VSTS and TFS Product Support

The primary support hubs for VSTS and TFS are:

- VSTS Support
- TFS Support

For technical support:

- Basic support
- Premium support

For billing support:

- VSTS (from the Azure Support Portal)

Documentation feedback

All docs on docs.microsoft.com have a ratings tool in the lower right-hand corner of the page. It asks "Is this content helpful?" Answer **Yes** or **No** depending on your experience.

You can add more detailed feedback by clicking the "Tell us more" link after selecting **Yes** or **No**. Check an appropriate box and add what we can do to improve the content for you! Although we cannot reply back, we collect and review these regularly, and use your sentiments in our content planning.

Tips for effective feedback

If you just want to vent about the product or the docs, that's okay. It helps us a lot to know when you're happy or unhappy with an experience. For the most impact, though, provide details so we can better understand what we're doing right or wrong.

- Provide a little context. What problem were you trying to solve? At what point did it go wrong?
- What's your role? We don't need personal or professional details. Are you a dev? A manager? A business

owner? When we understand our audience, we can come up with better solutions for you and other customers doing similar work.

- What version of the product were you using? What other products were you using with it?

The best feedback we get is clear and precise. For example:

- Product feedback: "I'm a project manager for a small start-up. I am using VSTS. I am trying to create work item templates through the UI, but my changes don't seem to persist. It's not clear what I'm doing wrong."
- Doc feedback: "I'm a dev in a big organization that works on Java apps. I tried to use Maven with your build system in TFS 2017 Update 1 (15.112.26307.0), but I couldn't get the configuration shown in the docs to work."

The more details, the better!

What platform/version am I using?

You can tell what platform you use from the URL you use to connect to VSTS or TFS.

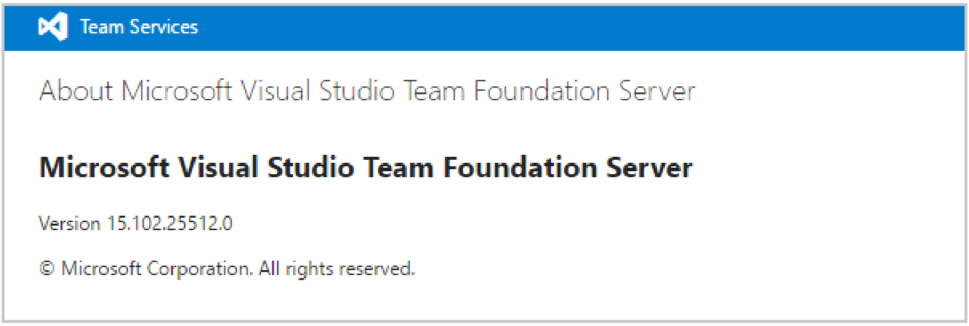
VSTS

A VSTS URL consists of an organization name and visual studio.com, for example: `https://{yourorganization}.visualstudio.com`.

To learn the version number, enter the following address in a web browser:

```
https://OrganizationName.visualstudio.com/_home/About
```

A page similar to the one shown will open showing the version number.



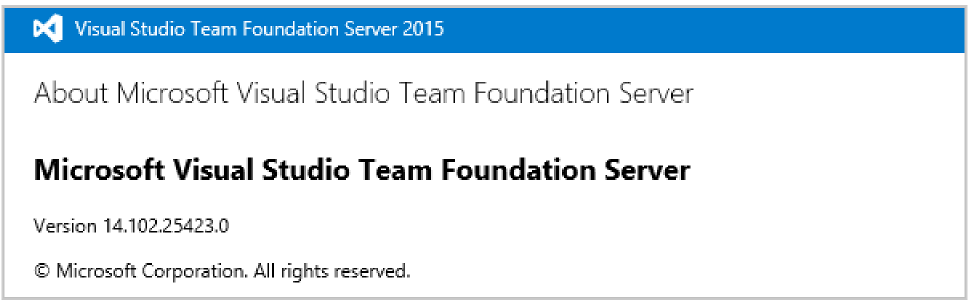
TFS

A TFS URL consists of a server name and collection, for example: `https://ServerName:8080/tfs/CollectionName`.

To learn the version number, enter the following address in a web browser:

```
https://ServerName:8080/tfs/_home/About
```

A page similar to the one shown will open showing the version number.



TFS RELEASE	UPDATE	VERSION NUMBER
TFS 2018	RTW	16.122.27102.1
	RC2	16.122.26918.3
	RC1	16.121.26818.0
TFS 2017	Update 3	15.117.27024.0
	Update 3 RC	15.117.26912.0
	Update 2	15.117.26714.0
	Update 1	15.112.26307.0
	RTW	15.105.25910.0
	RC1	15.103.25603.0
TFS 2015	Update 3	14.102.25423.0
	Update 2.1	14.95.25229.0
	Update 2	14.95.25122.0
	Update 2 RC 2	14.95.25029.0
	Update 2 RC 1	14.95.25005.0
	Update 1	14.0.24712.0
	Update 1 RC 2	14.0.24626.0
	Update 1 RC 1	14.0.24606.0
	RTM	14.0.23128.0
	RC2	14.0.23102.0
	RC	14.0.22824.0
	CTP	14.0.22604.0
TFS 2013	Update 5	12.0.40629.0
	Update 4	12.0.31101.0
	Update 4 RC	12.0.31010.0
	Update 3	12.0.30723.0

TFS RELEASE	UPDATE	VERSION NUMBER
	Update 3 RC	12.0.30626.0
	Update 2	12.0.30324.0
	RTM	12.0.21005.1
	RC	12.0.20827.3
TFS 2012	Update 4	11.0.61030.0
	Update 3	11.0.60610.1
	Update 2	11.0.60315.1
	CU 1	11.0.60123.100
	Update 1	11.0.51106.1
	RTM	11.0.50727.1
TFS 2010	CU 2	10.0.40219.371
	SP1	10.0.40219.1
	RTM	10.0.30319.1
TFS 2008	SP1	9.0.30729.1
	RTM	9.0.21022.8
TFS 2005	SP1	8.0.50727.762
	RTM	8.0.50727.147

Related articles

- Report a problem with Visual Studio

Default permissions and access for VSTS and TFS

8/10/2018 • 5 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

To connect and use the functions and features that VSTS and TFS provides, users must be added to a group with the appropriate permissions. The most common built-in groups include Readers, Contributors, and Project Administrators. These groups are assigned the default permissions as listed below.

In addition to permissions, access to specific features are controlled by the access level assigned to a user. Contributors and administrators should be added to Basic (paid) access. Stakeholder access is available to support free access to a limited set of features by an unlimited set of stakeholders.

For a complete reference of all built-in groups and permissions, see Permissions and groups. For information about assigning access levels and supporting stakeholder access, see Manage users and access for VSTS, and Change access levels for TFS.

Code: Git and TFVC

You can connect to your code from the Code hub or the web portal, and using Xcode, Eclipse, IntelliJ, Android Studio, Visual Studio, Visual Studio Code. For an overview of code features and functions, see Git and Use Team Foundation Version Control (TFVC). Stakeholders have no access to the Code hub or its features.

From the team project admin content for Version Control, you can set permissions on a repository. From the **Code>Branches** page, you can set permissions for a specific branch and set branch policies.

Git

TASK	READERS	CONTRIBUTORS	BUILD ADMINS	ACCOUNT OWNER/ PROJECT ADMINS
Clone, fetch, pull, and explore the contents of a repository	✓	✓	✓	✓
Unlimited private Git repositories		✓	✓	✓
Create branches and tags, manage notes		✓	✓	✓
Create, delete, and rename repositories				✓
Manage permissions, manage branches and branch policies				✓
Powerful semantic code search		✓	✓	✓

TFVC

TASK	READERS	CONTRIBUTORS	BUILD ADMINS	ACCOUNT OWNER/ PROJECT ADMINS
Contribute to a centralized version control, including Code Review (Check in, label, lock, merge, pend a change)	Read only	✓	✓	✓

Check in, revise, undo, unlock other users' changes				✓
Manage branches, manage permissions				✓
Powerful semantic code search		✓	✓	✓

Build and release

You can define and manage your builds and releases from the web portal, **Build and Release** hub. For an overview of build and release management features and functions, see Continuous integration on any platform.

From the **Build and Release** hub, you can set permissions for all or individual build pipelines, release pipelines, task groups, or variable groups. See Set build and release permissions.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	BUILD ADMINS	ACCOUNT OWNER/ PROJECT ADMINS	RELEASE ADMINS
View build and release pipelines		✓	✓	✓	✓	✓
Define builds with continuous integration			✓	✓	✓	
Define releases, manage deployments, manage releases with Release Management			✓		✓	✓
Approve releases	✓		✓		✓	✓
Package Management (5 users free)			✓		✓	✓
Queue builds, edit build quality			✓	✓	✓	
Manage build queues and build qualities				✓	✓	
Manage build retention policies, delete and destroy builds			✓	✓	✓	

Administer build permissions				✓	✓	
Manage release permissions					✓	✓
Create and edit task groups			✓	✓	✓	✓
Manage task group permissions				✓	✓	✓
Can view library items such as variable groups		✓	✓	✓	✓	✓
Use and manage library items such as variable groups				✓	✓	✓

Package Management feeds

Feeds have three levels of access: Owners, Contributors, and Readers. Owners can add any type of identity—individuals, teams, and groups—to any access level. To set permissions, see Secure feeds using permissions.

PERMISSION	READER	CONTRIBUTOR	OWNER
List and restore/install packages	✓	✓	✓
Push packages		✓	✓
Unlist/deprecate packages		✓	✓
Delete/unpublish package			✓
Edit feed permissions			✓
Rename and delete feed			✓

Test

You can define and manage manual tests from the web portal, **Test** hub. For an overview of manual test features and functions, see Testing overview.

You set test permissions at the team project level from the admin context Security page.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	ACCOUNT OWNER/ PROJECT ADMINS
Exploratory testing, view test runs		✓	✓	✓
Exploratory testing, create and delete test runs			✓	✓
Provide feedback using the Test & Feedback extension	✓	✓	✓	✓
Request feedback using the Test & Feedback extension			✓	✓
Manage test configurations and test environments			✓	✓
Manage test plans and test suites			✓	✓
Test Manager (purchased separately)			✓	✓

Agile tools and work tracking

You can connect to work items from the **Work** hub of the web portal and using Eclipse, Visual Studio, Excel, Project, and other clients. For an overview of work tracking features and functions, see About Agile tools. Stakeholders have limited access to select work tracking functions as described in Work as a stakeholder.

In addition to the permissions set at the project level via the built-in groups, you can set permissions for the following objects: area and iteration paths, queries and query folders, and delivery plans.

The team administrator role supports configuration of team settings. To be added as a team administrator, see Configure team settings and add team administrators.

NOTE

There are no UI permissions associated with managing tags. Instead, you can manage them using the TFSSecurity command line tool.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ACCOUNT OWNER/ PROJECT ADMINS
View work items, including bugs, requirements, and tasks	✓	✓	✓	✓	✓
Create and edit work items, follow a work item	✓		✓	✓	✓
Change work item type	✓		✓	✓	✓
Move or delete work items			✓	✓	✓
Search and query work items, save work item queries	✓	Can't save queries	✓	✓	✓

View backlogs, boards, and plans	✓	✓	✓	✓	✓
Provide feedback	✓	✓	✓	✓	✓
Request feedback			✓	✓	✓
Agile tools (Kanban boards, backlogs, sprint planning, portfolio management)	limited interactions	view only	✓	✓	✓
Configure Agile tools, set team defaults				✓	✓
Create new work item tags	Can assign existing tags		✓	✓	✓
View, add, and configure Delivery Plans		view only	✓	✓	✓
Customize project information (area paths, iteration paths, and work tracking processes)					✓
Powerful semantic work tracking search			✓	✓	✓

Charts, dashboards, and other web portal features

You can define and manage dashboards from the web portal, **Dashboard** hub. For an overview of dashboard and chart features, see Dashboards.

You set dashboard permissions at the team level from the team dashboard page.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ACCOUNT OWNER/ PROJECT ADMINS
View charts and dashboards	✓	✓	✓	✓	✓
Create work item and test tracking charts			✓	✓	✓
View the project page	✓	✓	✓	✓	✓
Edit the project page					✓
Navigate using the Project pages	✓	✓	✓	✓	✓
Add and configure dashboards			With permissions set	✓	✓

Analytics

From the **Analytics** hub, you can create and manage Analytics views. An Analytics view provides a simplified way to specify the filter criteria for a Power BI report based on the Analytics Service data store. The Analytics Service is the reporting platform for VSTS. To learn more, see What is the Analytics Service?.

You set permissions for the service at the project level, and for shared Analytics views at the object level.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ACCOUNT OWNER/ PROJECT ADMINS
View Analytics service		✓	✓	✓	✓
View, edit, and delete a shared Analytics view		✓	✓	✓	✓

Notifications, alerts, and team collaboration tools

To manage notifications, see Manage personal notifications and Manage team notifications.

> **NOTE**
>
> There are no UI permissions associated with managing notifications. Instead, you can manage them using the TFSSecurity command line tool.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ACCOUNT OWNER/ PROJECT ADMINS
Set personal notifications or alerts (see Note 1)	✓		✓	✓	✓
Set team notifications or alerts				✓	✓
Set project-level notifications or alerts					✓
Participate in Team (chat) rooms			✓	✓	✓
READMEs	See Note 2	✓	✓	✓	✓
View Wikis	✓	✓	✓	✓	✓
Provision or create a Wiki					✓
View the project page	✓	✓	✓	✓	✓
Edit the project page					✓
Navigate using the Project pages	✓	✓	✓	✓	✓
Request feedback		✓	✓	✓	✓
Provide feedback	✓	✓	✓	✓	✓
Powerful semantic code search			✓	✓	✓
Powerful semantic work tracking search			✓	✓	✓

Notes

1. Team (chat) rooms have been deprecated for VSTS and TFS 2018 and later versions.
2. Can view team project READMEs, but not READMEs defined for a repository.

Related notes

- Add users to a project or team
- Permissions and groups reference
- About access levels
- Navigation basics

Keyboard shortcuts for VSTS, TFS, and Team Explorer

8/9/2018 • 10 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

You can use the keyboard shortcuts listed in this topic when you work within Visual Studio Team Services (VSTS), the web portal for Team Foundation Server (TFS), or Team Explorer. In addition to these shortcuts, you can assign your own shortcuts in Visual Studio from the **Tools/Options/Environment/Keyboard** page.

For specific guidance on navigating within the web portal, see Navigation basics.

Web portal

You can use these keyboard shortcuts when working in the web portal for VSTS or TFS.

Navigate

Ctrl+Alt,a	Move focus to admin link
Ctrl+Alt,h	Move focus to ? help link
Ctrl+Alt,s	Move focus to search box
Ctrl+Alt,	Move focus to next section
Ctrl+Alt,	Move focus to previous section

Navigate within lists

Tab	Move focus
→	Move focus left/right
	Move focus up/down
Ctrl+Home	Move focus to top of list
Ctrl+End	Move focus to bottom of list
Ctrl+ ↑ ↓	Move item up/down within list
Shift ↑ ↓	Highlight consecutive items
Menu	Open context menu
Esc	Dismiss context menu
→ ←	Move focus left/right
↑ ↓	Move focus up/down
Enter	Choose selected menu item

Web portal, global shortcuts

Type **?** to access the Global and page-specific shortcuts.

> **NOTE**
>
> **Feature availability:** The following shortcuts are available from the web portal for VSTS and TFS 2015.2 and later versions. Type **?** to access the Global and page-specific shortcuts.

You can use the following keyboard shortcuts from the web portal.

	? Show shortcuts **p** Projects and teams **g,h** Go to the Dashboards hub **g,w** Go to the Work hub **g,c** Go to the Code hub **g,b** Go to the Build-Release hub **g,t** Go to the Test hub **g,s** Go to Settings (Admin context) **f,n** Focus next section **f,p** Focus previous section **s** Move focus to search

Page-specific shortcuts only work when in a specific page. For example, type **g c** to open the **Code** page, and then type **c p** to create a pull request. These navigation shortcuts work as long as the focus is not on an input control.

Code

You can use the following keyboard shortcuts when working in the **Code** hub. To view the valid shortcuts, enter **Shift+?** from the **Code** hub or when editing a wiki page.

> **NOTE**
>
> **Feature availability:** The following shortcuts are available from the web portal for VSTS and TFS 2015.2 and later versions.

	Code **r** Select repository **e** Open explorer **h** Open history **b** Open branches (Git) **q** Open pull requests (Git) **c,p** Create pull request (Git) **c** Open changesets (TFVC) **v** Open shelvesets (TFVC) **File Explorer** **1** Open contents **2** Open history **t** Move focus to directory path **w** Select branch (Git) **y** Switch to commit (Git) **c,b** Create branch (Git)

Work>Work Items

You can use the following keyboard shortcuts when working in the **Work** hub, **Work Items** page which is currently in preview. To enable it, see Enable preview features.

> **NOTE**
>
> **Feature availability:** The following shortcuts are available from the web portal for VSTS.

	Work **l** Open backlog **b** Open board **i** Open current iteration **t** Open task board **q** Open queries **z** Toggle full screen **Work items** **Ctrl+Shift+f** Filter results **Ctrl+c** Copy to clipboard **Delete** Delete

Work item form shortcuts

You can use the following keyboard shortcuts when interacting with a work item form. To view the valid shortcuts, enter **Shift+?** from within the form.

> **NOTE**
>
> **Feature availability:** The following shortcuts are available from VSTS.

	Work Item Form **Alt+i** Assign work item to me **Ctrl+Shift+d** Go to discussion **Ctrl+s** Save changes **Shift+Alt+c** Copy work item title **Ctrl+Shift+,** Move to left tab (page) **Ctrl+Shift+.** Move to right tab (page)

Also, you can use the following keyboard shortcuts when working within the work item form in the web portal, both VSTS and TFS.

FORMAT, COPY, PASTE RICH TEXT	SAVE AND CLOSE	
Ctrl+b Bold text **Ctrl+i** Italicize text **Ctrl+u** Underscore text **Ctrl+Spacebar** Clear formatting **Ctrl+c** Copy text **Ctrl+v** Paste copied text	**Ctrl+s** Save changes **Ctrl+Enter** Save and close **Esc** Close work item	

Work>Backlogs

You can use the following keyboard shortcuts when working in the **Work** hub, **Backlogs** pages. These shortcuts work when you are on a product backlog, portfolio backlog, or sprint backlog page.

	Backlogs **Ctrl+Home** Move item to top **m,b** Move item to backlog **m,i** Move item to current iteration **m,n** Move item to next iteration **n** Open new item panel **Ins** Add child **Ctrl+Shift+f** Filter results **r** Show/Hide Parents

Work>Boards

You can use the following keyboard shortcuts from any Kanban board, that is, when working in the **Work** hub, **Board** page.

> **NOTE**
> **Feature availability:** The following shortcuts are available from the web portal for VSTS and TFS 2015.2 and later versions.

	Kanban Board **n** Add new item **c** Add new child item **Home** Select first item **Enter** Open item **Ctrl+Shift+f** Filter results **Ctrl+** Move item up **Ctrl+** Move item down **Ctrl+** Move item left **Ctrl+** Move item right **Ctrl+Home** Move item to top of column **Ctrl+End** Move item to bottom of column **Ctrl+Shift+** Move item to swimlane above **Ctrl+Shift+** Move item to swimlane below **F2** Rename item **e** Show/hide empty fields **o** Expand all swimlanes **u** Collapse all swimlanes **Shift+Pageup** Select first/next swimlane above **Shift+Pagedown** Select last/next swimlane below

Work>Queries

You can use the following keyboard shortcuts when working with queries in the web portal. To view the valid shortcuts, enter **Shift+?** from the **Work>Queries** page.

> **NOTE**
>
> **Feature availability:** The following shortcuts are available from VSTS or TFS 2015.2 or later versions.

	Queries
	c q Add new query **r** or **Alt+r** Refresh query **Alt+q** Return to query **j or Alt+n** Select next item **k or Alt+p** Select previous item **Ctrl+Shift+f** Filter results

Work>Plans

You can use the following keyboard shortcuts when interacting with a delivery plan. To view the valid shortcuts, enter **Shift+?** when viewing a plan from the **Work>Plans** page.

> **NOTE**
>
> **Feature availability:** The following shortcuts are available from the web portal for VSTS and TFS 2017.2 and later versions. Type **?** to access the Global and hub-specific shortcuts.

Work	Delivery plan	
l Open backlog **b** Open board i Open current iteration **t** Open task board **q** Open queries **z** Toggle full screen mode	**Home** Select first item **Enter** Open item **n** New item **Ctrl+** Move item up **Ctrl+** Move item down **Ctrl+** Move item left **Ctrl+** Move item right	**Shift+** Pan timeline left **Shift+** Pan timeline right **u** Collapse all backlogs **o** Expand all backlogs **Shift+pageup** Focus on previous team **Shift+pagedown** Focus on next team **Ctrl+Shift+f** Filter results

Test>Test Plans, Parameters, and Runs

You can use the following keyboard shortcuts when working in the **Test** hub.

> **NOTE**
>
> **Feature availability:** The following shortcuts are available from the web portal for VSTS and TFS 2015.2 or later versions.

	Test **n** Open test plans **m** Open shared parameters **r** Open runs **h** Open machines
	Test plan **1** Open tests **2** Open charts **e** Execute tests **t,b** Mark selected tests as blocked **t,f** Fail selected tests **t,n** Mark selected tests as NA **t,p** Pass selected tests **t,r** Reset tests to active **Ctrl+Shift+f** Filter results **v,g** View grid
	Parameters **1** View parameter set grid **2** Open properties **c,s** Add parameter set **c,t** Add test case **v,t** Toggle test cases pane
	Test runs **1** Test runs **2** Filter

Wiki

> **NOTE**
>
> Keyboard shortcuts to manage Wiki pages are supported on TFS 2018.2 or later versions. To download TFS 2018.2, see Team Foundation Server 2018 Update 2 Release Notes.

You can use the following keyboard shortcuts when managing or editing Wiki pages. To view the valid shortcuts, enter **Shift+?** from the **Wiki** hub or when editing a wiki page.

> **NOTE**
>
> **Feature availability**: The following shortcuts are available from the web portal for VSTS and TFS 2018.2 and later versions.

Wiki (manage)

n Add new page
e Edit page
c Add new sub-page

Ctrl+ Move page up the order
Ctrl+ Move page down the order

Ctrl+Shift+f Filter page

Wiki edit

Ctrl+b Bold text
Ctrl+i Italicize text
Ctrl+k Insert hyperlink
Ctrl+c Copy text
Ctrl+v Paste copied text

Ctrl+s Save changes
Ctrl+Enter Save and Close
Esc Close

Team Explorer navigational shortcuts

Use these shortcuts when working in Team Explorer.

Navigate

Ctrl+0,a Open web portal
Ctrl+0,b Open Build
Ctrl+0,c Open Connect
Ctrl+0,d Open Documents
Ctrl+0,e Open Branches (Git)
Ctrl+0,g Open Changes (Git)
Ctrl+0,h Open Home
Ctrl+0,m Open My Work (TFVC)
Ctrl+0,p Open Pending changes (TFVC)
Ctrl+0,r Open Reports
Ctrl+0,s Open Settings
Ctrl+0,w Open Work items
Ctrl+0,y Open Synchronization (Git)

Ctrl+' Move focus to search box
Alt+0 Move focus to top of page
Alt+1...9 Move focus to visible section [1 thru 9]
Alt+ ↑ ↓ Move focus to next/previous section

Context menu

↓ Open a context menu
Esc Dismiss a context menu
→ ← Move focus left/right
↑ ↓ Move focus up/down
Enter Choose Context menu

Work item commands

Alt+m,g Open work item
Alt+m,i Add a work item
Alt+m,q Add a query
Shift+Alt,c Copy selected work item
Shift+Alt,l Link to new work item
Enter Open selected work item

You can use query results shortcuts whenever you have a list of work items, such as the query results view or a list of linked work items within a work item form.

QUERY EDITOR	ACTION	QUERY RESULTS	ACTION
	Move focus left/right		Scroll left/right
	Move focus up/down	**PgUp/PgDn**	Scroll up/down
Shift+	Highlight consecutive clauses	**Shift+**	Highlight consecutive rows

QUERY EDITOR	ACTION	QUERY RESULTS	ACTION
Shift+	Move focus left one field at a time	**Shift+Alt,n**	Move focus to next item
Shift+	Move focus right one field at a time	**Shift+Alt,p**	Move focus to previous item
End	Move focus to end of current clause	**End**	Move focus to bottom of list
Enter	Move focus down	**Enter**	Open selected work item
Tab	Move focus right, one field at a time	**Home**	Move focus to top of list
Ctrl+c	Copy selected clause	**+/-**	Expand/collapse current row
Ctrl+s	Save changes (editor)	**Ctrl+s**	Save changes (results)
Ctrl+v	Paste copied clause	**F5**	Refresh
Del	Delete contents of current field or clause		

Related articles

- Keyboard shortcuts for Microsoft Test Manager
- Customize Visual Studio keyboard shortcuts
- Default keyboard shortcuts for Visual Studio
- Accessibility Features of Visual Studio
- Navigation basics

Install Team Explorer

Team Explorer is a plug-in to Visual Studio. By installing the free Visual Studio Community, other Visual Studio version, or Visual Studio Team Explorer 2017 you gain access to Team Explorer.

Learn more about working in Team Explorer.

Syntax guidance for Markdown files, widgets, wikis, and pull request comments

8/9/2018 • 13 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015

Having the right guidance at the right time is critical to success. To support your team or contributors to your project, use markdown to add rich formatting, tables, and images to your project pages, readme files, dashboards, and pull request comments.

You can provide guidance to your team in these places using markdown:

- Team project wiki (provisioned wiki)
- Publish code as wiki
- Add Markdown to a dashboard
- Project vision page or Welcome pages
- Repository Readme files
- Pull request comments

- Add and edit wiki pages
- Add Markdown to a dashboard
- Project vision page or Welcome pages
- Repository Readme files
- Pull request comments

- Add Markdown to a dashboard
- Project vision page or Welcome pages
- Repository Readme files
- Pull request comments (Supported in TFS 2017 and later versions)

In this topic you'll find some basic Markdown syntax guidance. You can use both common Markdown conventions and GitHub-flavored extensions.

Headers

Structure your comments using headers. Headers segment longer comments, making them easier to read.

Start a line with a hash character `#` to set a heading. Organize your remarks with subheadings by starting a line with additional hash characters, for example `####`. Up to six levels of headings are supported.

Example:

```
# This is a H1 header
## This is a H2 header
### This is a H3 header
#### This is a H4 header
##### This is a H5 header
```

Result:

Paragraphs and line breaks

Make your text easier to read by breaking it up with paragraphs or line breaks.

In pull request comments, press Enter to insert a line break and begin text on a new line.

While in a Markdown file, wiki, or widget, enter two spaces prior to the line break to begin a new paragraph, or enter two line breaks consecutively to begin a new paragraph.

In pull request comments, press Enter to insert a line break and begin text on a new line.

In a Markdown file or widget, enter two spaces prior to the line break to begin a new paragraph, or enter two line breaks consecutively to begin a new paragraph.

In pull request comments, press Enter to insert a line break and begin text on a new line.

In a Markdown file or widget, enter two spaces prior to the line break to begin a new paragraph, or enter two line breaks consecutively to begin a new paragraph.

Example - pull request comment:

```
Add lines between your text with the Enter key.
This spaces your text better and makes it easier to read.
```

Result:
Add lines between your text with the Enter key.
This spaces your text better and makes it easier to read.

Example - markdown file or widget:

```
Add two spaces prior to the end of the line.(space, space)
This adds space in between paragraphs.
```

Result:
Add two spaces prior to the end of the line.

This adds space in between paragraphs.

Quotes

Quote previous comments or text to set context for your comment or text.

Quote single lines of text be putting a `>` before the text. Use multiple `>` characters to nest quoted text. Quote blocks of lines of text by using the same level of `>` across multiple lines.

Example:

```
> Single line quote
>> Nested
>> multiple line
>> quote
```

Result:

> Single line quote
>> Nested quote
>> multiple line
>> quote

Horizontal rules

Add a horizontal rule by adding a new line that's just a series of dashes `---`. There must be a blank line above the line containing the `---`.

Example:

```
above

----
below
```

Result:

above

---

below

Lists

Organize related items with lists. You can add ordered lists with numbers, or unordered lists with just bullets.

Ordered lists start with a number followed by a period for each list item. Unordered lists start with a `-`. Begin each list item on a new line. In a Markdown file or widget, enter two spaces prior to the line break to begin a new paragraph, or enter two line breaks consecutively to begin a new paragraph.

Ordered or numbered lists

Example:

```
 0. First item.
 0. Second item.
 0. Third item.
```

Result:

1. First item.
2. Second item.
3. Third item.

Bullet lists

Example:

```
- Item 1
- Item 2
- Item 3
```

Result:

- Item 1
- Item 2
- Item 3

Nested lists

Example:

```
1. First item.
   - Item 1
   - Item 2
   - Item 3
1. Second item.
   - Nested item 1
   - Nested item 2
   - Nested item 3
```

Result:

1. First item.
   - Item 1
   - Item 2
   - Item 3
2. Second item.
   - Nested item 1
   - Nested item 2
   - Nested item 3

Links

In pull request comments and wikis, HTTP and HTTPS URLs are automatically formatted as links. Also, within pull requests, you can link to work items by typing the # key and a work item ID, and then choosing the work item from the list.

You can escape auto suggestion of work items by prefixing # with a backslash (`\`). E.g. This can be useful if you want to use # for color hex codes.

In markdown files and widgets, you can set text hyperlinks for your URL using the standard markdown link syntax:

```
[Link Text](Link URL)
```

When linking to another Markdown page in the same Git or TFVC repository, the link target can be a relative path or an absolute path in the repository.

Supported links for Welcome pages:

- Relative path: `[text to display](./target.md)`
- Absolute path in Git: `[text to display](/folder/target.md)`
- Absolute path in TFVC: `[text to display]($/project/folder/target.md)`

- URL: `[text to display](http://address.com)`

Supported links for Markdown widget:

- URL: `[text to display](http://address.com)`

Supported links for Wiki:

- Absolute path of Wiki pages: `[text to display](/parent-page/child-page)`
- URL: `[text to display](http://address.com)`

> **NOTE**
>
> Links to documents on file shares using `file://` are not supported on VSTS or TFS 2017.1 and later versions. This restriction has been implemented for security purposes.
>
> For information on how to specify relative links from a Welcome page or Markdown widget, see Source control relative links.

Example:

```
[C# language reference](https://msdn.microsoft.com/library/618ayhy6.aspx)
```

Result:

C# language reference

Link to work items from a Wiki page

Simply enter the pound sign (`#`) and enter a work item ID.

> **NOTE**
>
> This feature is available with TFS 2018.2 and later versions.

Source control relative links

Links to source control files are interpreted differently depending on whether you specify them in a Welcome page or a Markdown widget. The system interprets relative links as follows:

- **Welcome page:** relative to the root of the source control repository in which the welcome page exists
- **Markdown widget:** relative to the team project collection URL base.

For example:

WELCOME PAGE	MARKDOWN WIDGET EQUIVALENT
/BuildTemplates/AzureContinuousDeploy.11.xaml	/DefaultCollection/Fabrikam Fiber/_versionControl#path=$/Tfvc Welcome/BuildTemplates/AzureContinuousDeploy.11.xaml
./page-2.md	/DefaultCollection/Fabrikam Fiber/_versionControl#path=$/Tfvc Welcome/page-2.md

Anchor links

Within Markdown files, anchor IDs are assigned to all headings when rendered as HTML. The ID is the heading text, with the spaces replaced by dashes (-) and all lower case. In general, the following conventions:

- Punctuation marks and leading white spaces within a file name are ignored

- Upper case letters are converted to lower
- Spaces between letters are converted to dashes (-).

Example:

```
###Link to a heading in the page
```

Result:

The syntax for an anchor link to a section...

```
[Link to a heading in the page](#link-to-a-heading-in-the-page)
```

The ID is all lower case, and the link is case sensitive, so be sure to use lower case, even though the heading itself uses upper case.

You can also reference headings within another Markdown file:

```
[text to display](./target.md#heading-id)
```

In wiki, you can also reference heading in another page:

```
[text to display](/page-name#section-name)
```

Images

Add images and animated GIFs to your pull request comments, markdown files, or wiki pages to highlight issues or just to liven the discussion.

Use the following syntax to add an image:

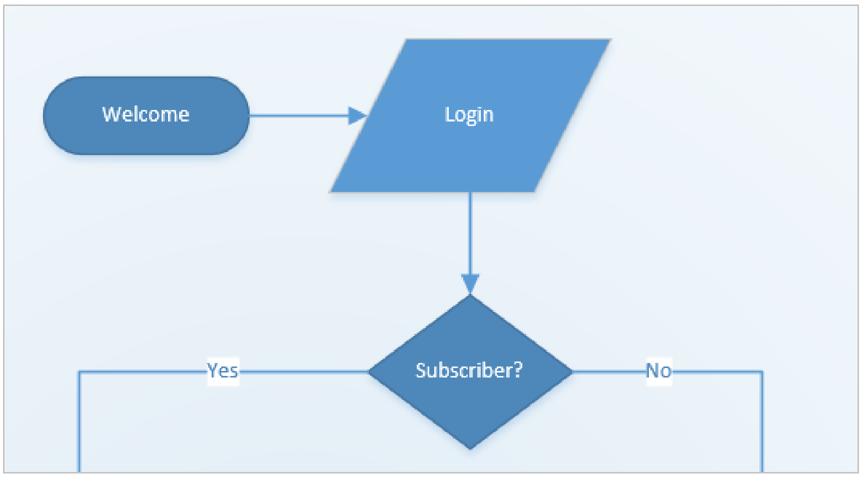
```
![Text](URL)
```

The text in the brackets describes the image being linked and the URL points to the image location.

Example:

```
![Let's use this flow for the login experience](http://dev.fabrikam.net/images/uxflow.png)
```

Result:



The path to the image file can be a relative path or the absolute path in Git or TVFC, just like the path to another Markdown file in a link.

- Relative path:
  `![Image alt text](./image.png)`
- Absolute path in Git:
  `![Image alt text](/_img/markdown-guidance/image.png)`
- Absolute path in TFVC:
  `![Image alt text]($/project/folder/_img/markdown-guidance/image.png)`
- Resize image:
  `![Image alt text]($/project/folder/_img/markdown-guidance/image.png =WIDTHxHEIGHT)`

NOTE

Feature availability: The syntax to support image resizing is only supported in pull requests and the Wiki.

Tables

Organize structured data with tables. Tables are especially useful for describing function parameters, object methods, and other data that has a clear name to description mapping. You can format tables in pull requests, wiki, and markdown files such as READMe files and markdown widgets.

- Place each table row on its own line
- Separate table cells using the pipe character `|`
- The first two lines of a table set the column headers and the alignment of elements in the table
- Use colons (`:`) when dividing the header and body of tables to specify column alignment (left, center, right)
- To start a new line, use the HTML break tag (`<br/>`) (Works within a Wiki but not elsewhere)
- Make sure to end each row with a CR or LF.

Example:

```
| Heading 1 | Heading 2 | Heading 3 |
|-----------|:-----------:|-----------:|
| Cell A1 | Cell A2 | Cell A3 |
| Cell B1 | Cell B2 | Cell B3<br/>second line of text |
```

Result:

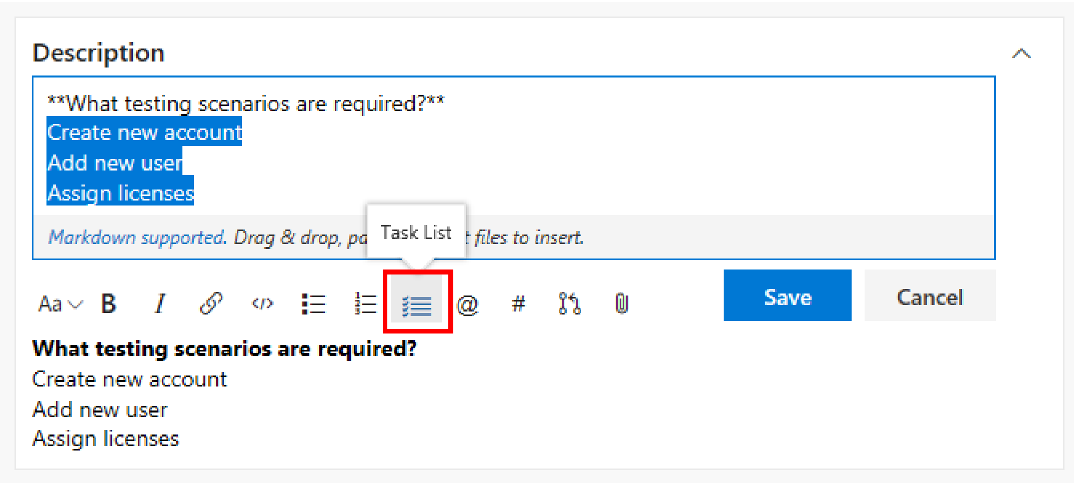
HEADING 1	HEADING 2	HEADING 3
Cell A1	Cell A2	Cell A3
Cell B1	Cell B2	Cell B3 second line of text

Checklist or task list in pull requests or markdown

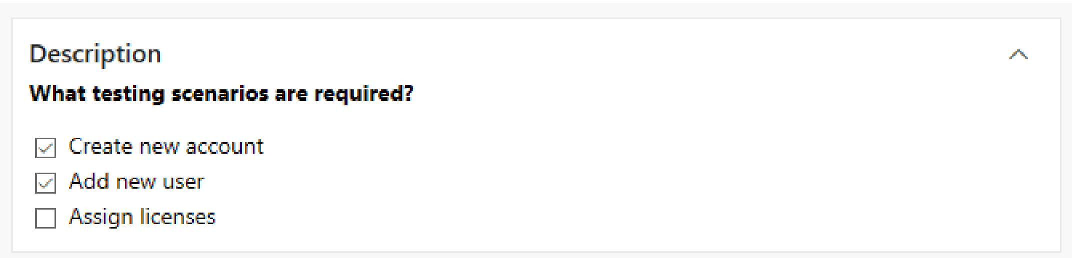
Lightweight task lists are a great way to track progress on a list of todos as either a pull request creator or reviewer in the description or a single, consolidated comment. Click the Markdown toolbar to get started or apply the format to selected text.

You can Use `[ ]` or `[x]` to support checklists. You need to precede the checklist with either `-<space>` or `1.<space>` (any numeral).

Example - Apply the task list markdown to a higlighted list



Once you've added a task list, you can simply check the boxes to mark items as completed. These are expressed and stored within the comment as [ ] and [x] in Markdown.



Example - Format a list as a task list

```
- [ ] A
- [ ] B
- [ ] C
- [x] A
- [x] B
- [x] C
```

Result:

- [ ] A
- [ ] B
- [ ] C
- [x] A
- [x] B
- [x] C

> **NOTE**
>
> A checklist within a table cell isn't supported.

Emphasis (bold, italics, strikethrough)

You can emphasize text by applying bold, italics, or strikethrough to characters:

- To apply italics: surround the text with an asterisk `*` or underscore `_`
- To apply bold: surround the text with double asterisks `**`.
- To apply strikethrough: surround the text with double tilde characters `~~`.

Combine these elements to apply multiple emphasis to text.

> **NOTE**
>
> There is no markdown syntax that supports underlining text. Within a wiki page, you can use the HTML `<u>` tag to generate underlined text. For example, `<u>underlined text</u>` will yield underlined text`.

Example:

```
Use _emphasis_ in comments to express **strong** opinions and point out ~~corrections~~
**_Bold, italizied text_**
**~~Bold, strike-through text~~**
```

Result:
Use *emphasis* in comments to express **strong** opinions and point out ~~corrections~~
Bold, italizied text
~~Bold, strike-through text~~

Code highlighting

Highlight suggested code segments using code highlight blocks. To indicate a span of code, wrap it with three backtick quotes (```` ``` ````) on a new line at both the start and end of the block. To indicate code inline, wrap it with one backtick quote (`` ` ``).

Example:

~~~~
```
$ sudo npm install vsoagent-installer -g
```
~~~~

Result:

```
$ sudo npm install vsoagent-installer -g
```

Example:

```
To install the Microsoft VSTS Cross Platform Build & Release Agent, run the following: `$ sudo npm
install vsoagent-installer -g`.
```

Result: To install the Microsoft VSTS Cross Platform Build & Release Agent run the following: `$ sudo npm install vsoagent-installer`.

Within a markdown file, text with four spaces at the beginning of the line automatically converts to a code block.

Set a language identifier for the code block to enable syntax highlighting for any of the supported languages.

````
``` language
code
```
````

Additional examples:

````
``` js
const count = records.length;
```
````

```
const count = records.length;
```

````
``` csharp
Console.WriteLine("Hello, World!");
```
````

```
Console.WriteLine("Hello, World!");
```

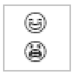
Emoji

In pull request comments and wiki pages, you can use emojis to add character and react to comments in the request. Type in what you're feeling surrounded by `:` characters to get a matching emoji in your text. The full set of emojis are supported.

Example:

```
:smile:
:angry:
```

Result:



To escape emojis, enclose them using the ` character.

Example:

```
`:smile:` `:)` `:angry:`
```

Result:

`:smile:` `:)` `:angry:`

Ignore or escape markdown syntax to enter specific or literal characters

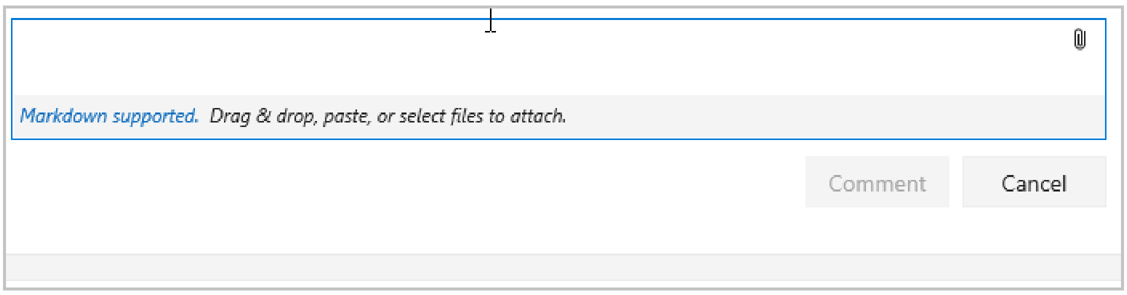
SYNTAX	EXAMPLE/NOTES

To insert one of the following characters, prefix with a backslash:
`\ backslash`
`` ` `` `backtick`
`_ underscore`
`{} curly braces`
`[] square brackets`
`() parentheses`
`# hash mark`
`+ plus sign`
`- minus sign (hyphen)`
`. dot`
`! exclamation mark`
	Some examples on inserting special characters
	Enter `\\` to get \\
	Enter `\_` to get _
	Enter `\#` to get #
	Enter `\(` to get (
	Enter `\.` to get .
	Enter `\!` to get !

Attachments

In pull request comments and wiki pages, you can attach files to illustrate your point or to give more detailed reasoning behind your suggestions. To attach a file, drag and drop it into the comment field or wiki page edit experience. You can also select the paper-clip icon in the upper-right of the comment box or the format pane in wiki page.

> **NOTE**
>
> Attachments in pull requests is available with TFS 2017.1 and later versions.



If you have an image in your clipboard, you can paste it from the clipboard into the comment box or wiki page and it will render directly into your comment or wiki page.

Attachments support the following file formats:

TYPE	FILE FORMATS
Code	CS (.cs), Extensible Markup Language (.xml), JavaScript Object Notation (.json), Layer (.lyr), Windows PowerShell script (.ps1), Roshal Archive (.rar), Remote Desktop Connection (.rdp), Structured Query Language (.sql) - **Note: Code attachments are not permitted in PR comments**
Compressed files	ZIP (.zip) and GZIP (.gz)
Documents	Markdown (.md), Microsoft Office Message (.msg), Microsoft Project (.mpp), Word (.doc and .docx), Excel (.xls, .xlsx and .csv), and Powerpoint (.ppt and .pptx), text files (.txt), and PDFs (.pdf)
Images	PNG (.png), GIF (.gif), JPEG (both .jpeg and .jpg)
Visio	VSD (.vsd and .vsdx)
Video	MOV (.mov), MP4 (.mp4)

NOTE

Not all file formats are supported within pull requests, such as Microsoft Office Message (.msg) files.

Attaching non-image files creates a link to the file in your comment. Update the description text between the brackets to change the text displayed in the link. Attached image files render directly into your comment or wiki pages.

Once you save or update a comment or wiki page with an attachment, you can see the attached image(s) and can select links to download attached files.

HTML Tags

In wiki pages, you can also create rich content using HTML tags.

NOTE

Pasting rich content as HTML is supported in TFS 2018.2 and later versions.

Example - Embedded video

```
<video src="path of the video file" width=400 controls>
</video>
```

**For example: **

```
<video src="https://sec.ch9.ms/ch9/7247/7c8ddc1a-348b-4ba9-ab61-51fded6e7247/vstswiki_high.mp4" width=400
controls>
</video>
```

Or,

```
<video src="_media/vstswiki_mid.mp4" width=400 controls>
</video>
```

Result:

Example - Rich text format

```
<p>This text needs to <del>strikethrough</del> <ins>since it is redundant</ins>!</p>
<p><tt>This text is teletype text.</tt></p>
<font color="blue">Colored text</font>
<center>This text will be center-aligned.</center>
<p>This text contains <sup>superscript</sup> text.</p>
<p>This text contains <sub>subscript</sub> text.</p>
<p>The project status is <span style="color:green;font-weight:bold">GREEN</span> even though the bug count /
developer may be in <span style="color:red;font-weight:bold">red.</span> - Capability of span
<p><small>Disclaimer: Wiki also supports showing small text</small></p>
<p><big>Bigger text</big></p>
```

Result:

This text needs to ~~strikethrough~~ since it is redundant!

`This text is teletype text.`

Colored text

This text will be center-aligned.

This text contains $^{\text{superscript}}$ text.

This text contains $_{\text{subscript}}$ text.

The project status is **GREEN** even though the bug count / developer may be in **red.** - Capability of span

Disclaimer: Wiki also supports showing small text

Bigger text

Mathematical notation and characters

Both inline and block KaTeX notation is supported in wiki pages and pull requests. This includes inserting symbols, Greek letters, mathematical operators, powers and indices, fractions and binomials, and other KaTeX supported elements.

To include mathematical notation, surround the mathematical notation with a `$` sign, for inline, and `$$` for block, as shown in the following examples:

> **NOTE**
>
> This feature is supported within Wiki pages and pull requests for TFS 2018.2 or later versions.

Example: Greek characters

```
$
\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta, \theta, \kappa, \lambda, \mu, \nu, \omicron, \pi, \rho,
\sigma, \tau, \upsilon, \phi, ...
$


$\Gamma,  \Delta,  \Theta, \Lambda, \Xi, \Pi, \Sigma, \Upsilon, \Phi, \Psi, \Omega$
```

Result:

$\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta, \theta, \kappa, \lambda, \mu, \nu, \omicron, \pi, \rho, \sigma, \tau, \upsilon, \phi, \ldots$

$\Gamma, \Delta, \Theta, \Lambda, \Xi, \Pi, \Sigma, \Upsilon, \Phi, \Psi, \Omega$

Example: Algebraic notation

```
Area of a circle is $\pi r^2$

And, the area of a triangle is:

$$
A_{triangle}=\frac{1}{2}({b}\cdot{h})
$$
```

Result:

Area of a circle is πr^2

And, the area of a triangle is:

$$A_{triangle}=\frac{1}{2}({b}\cdot{h})$$

Example: Sums and Integrals

```
$$
\sum_{i=1}^{10} t_i
$$


$$
\int_0^\infty \mathrm{e}^{-x}\,\mathrm{d}x
$$
```

Result:

$$\sum_{i=1}^{10} t_i$$

$$\int_0^\infty \mathrm{e}^{-x}\,\mathrm{d}x$$

Related articles

- Project vision page or Welcome pages
- Readme files

- Pull requests
- Markdown widget
- Dashboards
- Widget catalog
- Wiki

Navigate in Team Explorer

8/10/2018 • 7 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

You use Team Explorer to manage work that is assigned to you, your team, or your team projects, and to coordinate your efforts with other team members to develop a project. Team Explorer is a plug-in that installs with Visual Studio or Eclipse. Working from different platforms, developers and stakeholders can effectively collaborate using Team Explorer connected to team projects hosted on Visual Studio Online or on-premises Team Foundation Server (TFS).

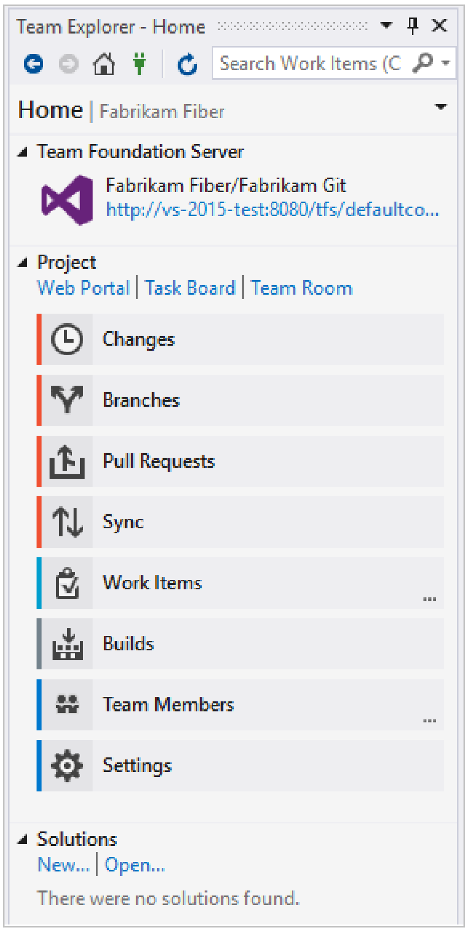
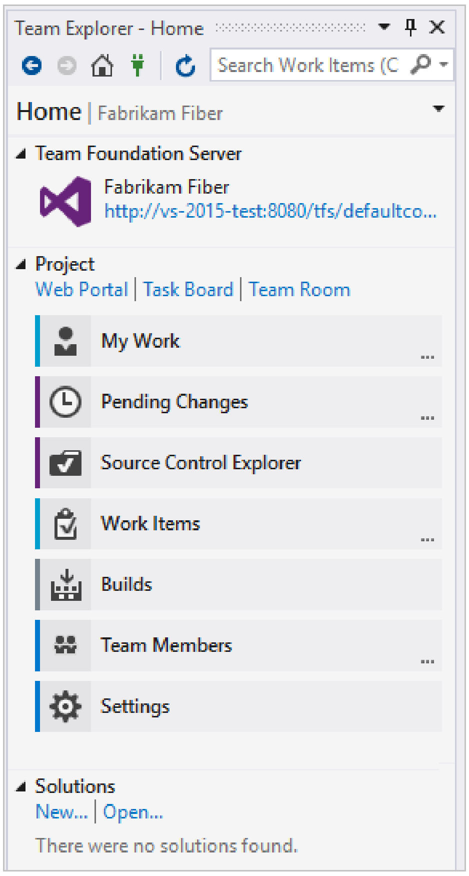
> **TIP**
> You can access the latest version of Visual Studio clients from the Visual Studio Downloads page. Additional options for connecting to VSTS or TFS include:
>
> - Team Explorer Everywhere
> - VSTS Plugin for Android Studio
> - VSTS Plugin for IntelliJ
> - Visual Studio Code For information about compatibility among client and server versions, see Requirements and compatibility.

The operations available to you depend on which source control option-Team Foundation version control (TFVC) or Git-was selected to manage source code when the team project was created.

Team Explorer plug-in for Visual Studio

Team Explorer connects Visual Studio to team projects. You can manage source code, work items, and builds. Or, create a team project.

> **TIP**
> If you open Visual Studio and the Team Explorer pane doesn't appear, click the **View/Team Explorer** menu option.

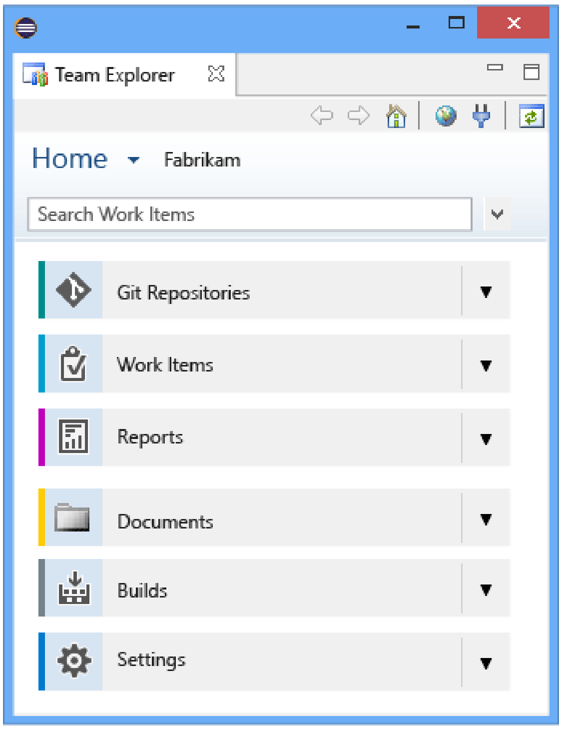
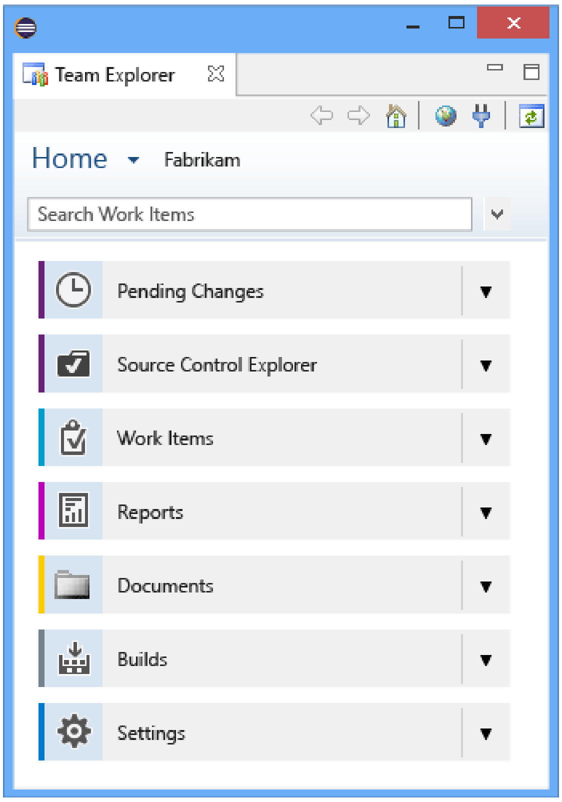
HOME PAGE WITH GIT	HOME PAGE WITH TFVC
	

> **NOTE**
>
> Some pages, such as **Reports** and **Documents**, only appear when an on-premises TFS is configured with the required resources, such as SQL Server Reporting Services and SharePoint.

If you don't need Visual Studio, but do want to connect to VSTS or TFS or get one or more Team Foundation add-ins, you can install the free Visual Studio Community.

Team Explorer plug-in for Eclipse

If you work in Eclipse or on a non-Windows platform, you can install the Team Explorer plug-in for Eclipse. Once installed, you can share your Eclipse projects by adding them to VSTS or TFS using Git or TFVC.

HOME PAGE WITH GIT (ECLIPSE)	HOME PAGE WITH TFVC (ECLIPSE)
	

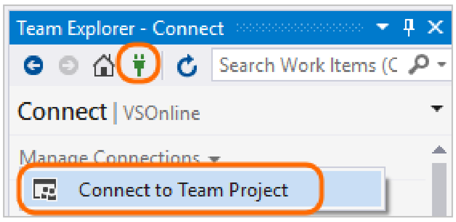
> **NOTE**
>
> Some pages, such as **Reports** and **Documents**, only appear when TFS is configured with the required resources, such as SQL Server Reporting Services and SharePoint.

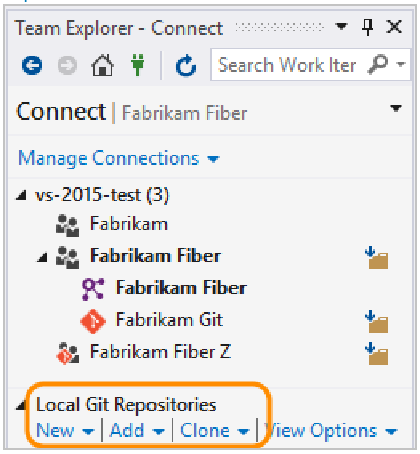
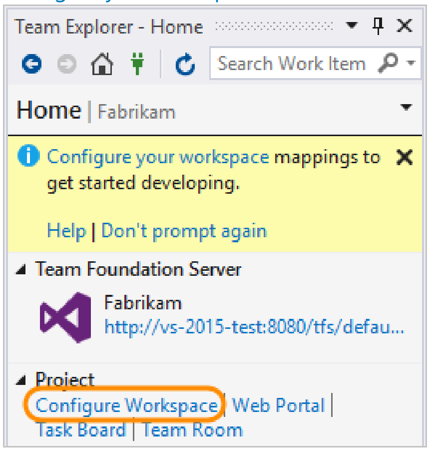
Choose the page to access the task you want

Based on the page you select and the options configured for your team project.

Connect page

From the **Connect** page, you can select the team projects you want to connect to and quickly switch context between team projects.



<table>
<tr><th>GIT: LOCAL GIT REPO</th><th>TFVC: MAP WORKSPACE</th></tr>
<tr><td>If you connect to a Git repo, you also can create, add, or clone repositories.

</td><td>If you connect to a TFVC repo, you'll be prompted to Configure your workspace (TFVC) on first connect.

</td></tr>
</table>

IMPORTANT

From the Visual Studio plug-in, you can Create a team project. The ability to create team projects is not supported from the Eclipse plug-in. You can, however, create team projects from the web portal/collection administration context.

Home, Work, and Build pages

HOME	WORK ITEMS	BUILD
-Configure workspace - Open Web portal - Open Task Board - Open Team Room	- Add work items - Use the query editor to list and manage queries - Organize query folders and set query permissions - Open query in Excel - Open query in Project - Email query results list using Outlook - Create reports from query in Excel (TFS only)	- Create build pipelines - View and manage builds - Manage the build queue

NOTE

If inline images aren't displaying correctly, see Resolve images that don't display in Team Explorer.

Git and TFVC pages

The Git and TFVC repos support different pages and functions. You'll see one or the other pages depending on the team project and repro you connect to. For a comparison of the two repos, see Choosing the right version control for your project.

GIT	TFVC
- **Changes**: Save work with commits - **Branches**: Create work in branches - **Pull Requests**: Review code with pull requests - **Sync**: Update code with fetch and pull)	- **My Work**: Suspend/resume work | Code review - **Pending Changes**: Manage pending changes | Find shelvesets | Resolve conflicts - **Source Control Explorer**: Add/view files and folders

Report and Document pages (TFS only)

The **Report** page opens the Reporting Services report site. This page appears only when your team project has been configured with SQL Server Analysis Services and Reporting Services. Also, the option to **Create Report in Microsoft Excel** appears only when reporting has been configured for the team project.

From the **Document** page, you can open project portal and manage documents and document libraries. This page appears only if your team project has been configured with a SharePoint Products portal.

If your team project is missing one or more pages, you may be able to add functionality to your on premises TFS deployment.

Settings page

From the **Settings** page, you can configure administrative features for either a team project or team project collection. Configuring features in these areas requires you to be a member of a VSTS or TFS administrator group.

Most of the links open to a web portal administration page. Not all settings are available from the Team Explorer plug-in for Eclipse.

TEAM PROJECT SETTINGS	COLLECTION SETTINGS (TFS)
Required membership: Project Administrators - Security (manage project-level permissions) - Group membership: manage group permissions - Source Control: Configure the check-in and check-out policies (TFVC) - Work Item Areas (project-wide) - Work Item Areas (project-wide) - Portal Settings (Enable portal or process guidance) -Project Alerts	Required membership: Project Collection Administrators - Security (manage project-level permissions) - Group membership: manage group permissions - Source Control: Configure the default workspace type for the collection (TFVC) - Process Template Manager: download or upload a process template

To learn more about administration features, see About team, project, and organizational-level settings.

Related articles

You've now got a basic understanding of how to work in the Team Explorer add-in for Visual Studio and Eclipse.

- Troubleshoot connection

Clients that connect to VSTS or TFS

In addition to connecting through Team Explorer, you can connect to a team project from these clients:

- Web portal
- Visual Studio Code
- Visual Studio Community
- Eclipse: Team Explorer Everywhere
- Office Excel
- Office Project
- PowerPoint Storyboarding
- Microsoft Test Manager
- Microsoft Feedback Client

Refresh Team Explorer

If data doesn't appear as expected, the first thing to try is to refresh your client. Refreshing your client updates the local cache with changes that were made in another client or in TFS. To refresh Team Explorer, do one of the following actions:

- To refresh a page that you are currently viewing, choose **Refresh** icon in the menu bar (or choose the F5 key).

- To refresh the team project you currently have selected, choose **Home**, and then choose **Refresh** icon (or choose the F5 key).
- To refresh the set of teams defined for the team project that you currently have selected, choose the Connect icon, and then choose **Refresh** icon (or choose the F5 key).

To avoid potential errors, you should refresh your client application under the following circumstances:

- Process changes are made
- Work item type definitions are added, removed, renamed or updated
- Area or iteration paths are added, removed, renamed or updated
- Users are added to or removed from security groups or permissions are updated
- A team member adds a new shared query or changes the name of a shared query
- A build pipeline is added or deleted
- A team or team project is added or deleted.

Resolve images that don't display in Team Explorer

If an in-line image fails to display in a work item form you view from Visual Studio Team Explorer, but does display in the web portal, your credentials may have expired.

You can resolve it with the following steps.

1. In Visual Studio, click **View>Other Windows>Web Browser** (Or, use the shortcut **Cltr+Alt+R**).
2. In the web browser, locate your account.
3. Login with your account.
4. Refresh your work item in Team Explorer.

Additional tools provided with TFS Power Tools (Visual Studio 2015 & TFS 2015)

By installing TFS Power Tools, you gain access to these additional tools through the Team Explorer plug-in for Visual Studio:

- Process Template Editor
- Additional check-in policies for Team Foundation Version Control
- Team Explorer enhancements including Team Members
- Team Foundation Power Tool Command Line
- Test Attachment Cleaner
- Work Item Templates

Additional requirements may apply.

> **NOTE**
>
> For TFS 2017 and later versions, you can install the TFS Process Template editor from the Visual Studio Marketplace. You can use this version of the Process Editor to modify the old-style work item forms. You can't use it to edit forms associated with the new web forms.

FAQs

8/10/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

Q: How do I get started?

A: To get started in the cloud or on-premises:

- To get started with Visual Studio Team Services (VSTS) begin by creating a user. Step-by-step instructions are provided in Sign up for VSTS.
- To get started with an on-premises TFS, download and install the latest version of TFS. See Install and configure TFS for details.
- If you need to create a team project, create one in VSTSs or set one up in an on-premises TFS.
- If you don't have access to the team project, get invited to the team.
- If it's your first time connecting to a team project, see Connect to a team project.

A: To get started with a client tool: Go to one of these pages to download a version of Visual Studio or client tool plug-in that will support connecting to a team project:

- Visual Studio
- Eclipse/Team Explorer Everywhere
- Android Studio with the VSTS Plugin for Android Studio
- IntelliJ with the VSTS Plugin for IntelliJ
- Visual Studio Code

A: To get started with sharing code, work item tracking, builds, or other tasks: See Software development roles.

Q: What compatibility issues exist between client and server versions?

A: See Requirements and compatibility.

Q: Can stakeholders who don't use Visual Studio participate on our team?

A: Yes. You can provide access to stakeholders who have no CAL for the following activities:

- **Stakeholder access**: This view allows anyone on your team to check project status and provide feedback. Stakeholders can track project priorities and provide direction, feature ideas, and business alignment to a team.

  To grant stakeholders access, add them to the Stakeholder access group.

- **Provide feedback**: To allow your stakeholders to provide feedback, you must grant them specific permissions.

Q: Are there other clients that connect to VSTS or TFS? Are there other tools I can use?

A: Yes. You can connect to a team project from one of these clients:

- Excel (Requires the Team Foundation add-in is installed)
- Project (Requires the Team Foundation add-in is installed)
- Project Professional
- PowerPoint Storyboarding (Requires the Team Foundation add-in is installed)
- Microsoft Test Manager

- Test & Feedback extension (previously called the Exploratory Testing extension)
- Microsoft Feedback Client

> **NOTE**
>
> Native support for integrating TFS with Project Server is deprecated for TFS 2017. However, synchronization support is provided by a third part. See Synchronize TFS with Project Server for details.
> Test Manager is deprecated for TFS 2017.

Also, you can find several open-source clients that have been added to Marketplace extensions. For example, you can install extensions to Visual Studio that support additional features:

- For TFS 2017 and later versions, you can install the TFS Process Template editor from the Visual Studio Marketplace. You can use this version of the Process Editor to modify the old-style work item forms. You can't use it to edit forms associated with the new web forms.
- For TFS 2015 and earlier versions, you can install TFS Power Tools which provides enhancements, tools, and command-line utilities that support increased productivity.

> **NOTE**
>
> Team Foundation Server Power Tools is deprecated for TFS 2017 and later versions.

Related notes

- Key concepts
- Essential services
- Client-server tools
- Software development roles

Have more questions? Search for an answer or pose a question in one of the community forums listed in Provide product and content feedback, Platforms and version support.

Navigation basics

8/9/2018 • 5 minutes to read

VSTS | TFS 2018 | TFS 2017

The web portal for Visual Studio Team Services (VSTS) is organized around a set of services—such as, **Code**, **Work**, **Build and Release**—as well as administrative pages and several task-specific features such as the search box. Each service provides you with one or more pages which support a number of features and functional tasks. Within a page, you may then have a choice of options to select a specific artifact or add an artifact.

The web portal for Team Foundation Server (TFS) is organized around a set of services—such as, **Dashboard**, **Code**, **Work**, **Build and Release**—as well as administrative pages and several task-specific features such as the search box. Each service provides you with one or more pages which support a number of features and functional tasks. Within a page, you may then have a choice of options to select a specific artifact or add an artifact.

> **NOTE**
>
> The web portal is one of several clients that can connect to a team project. Different clients support different features and functions. For a list of all clients that support connection to a team project, see Tools and clients that connect to VSTS and TFS.

Here's what you need to know to get up and running using the web portal.

> **NOTE**
>
> Choose **Previous navigation** when you see a top-level blue bar. Choose **New navigation** if you see a vertical sidebar or if you enabled the **New Navigation** preview feature. The vertical sidebar, along with other navigational features, is enabled when the **New Navigation** preview feature has been enabled for the signed-in user or the VSTS organization. To learn how to use the web portal effectively, see Navigation Basics.
>
> For on-premises TFS users, choose **Previous Navigation** for guidance.

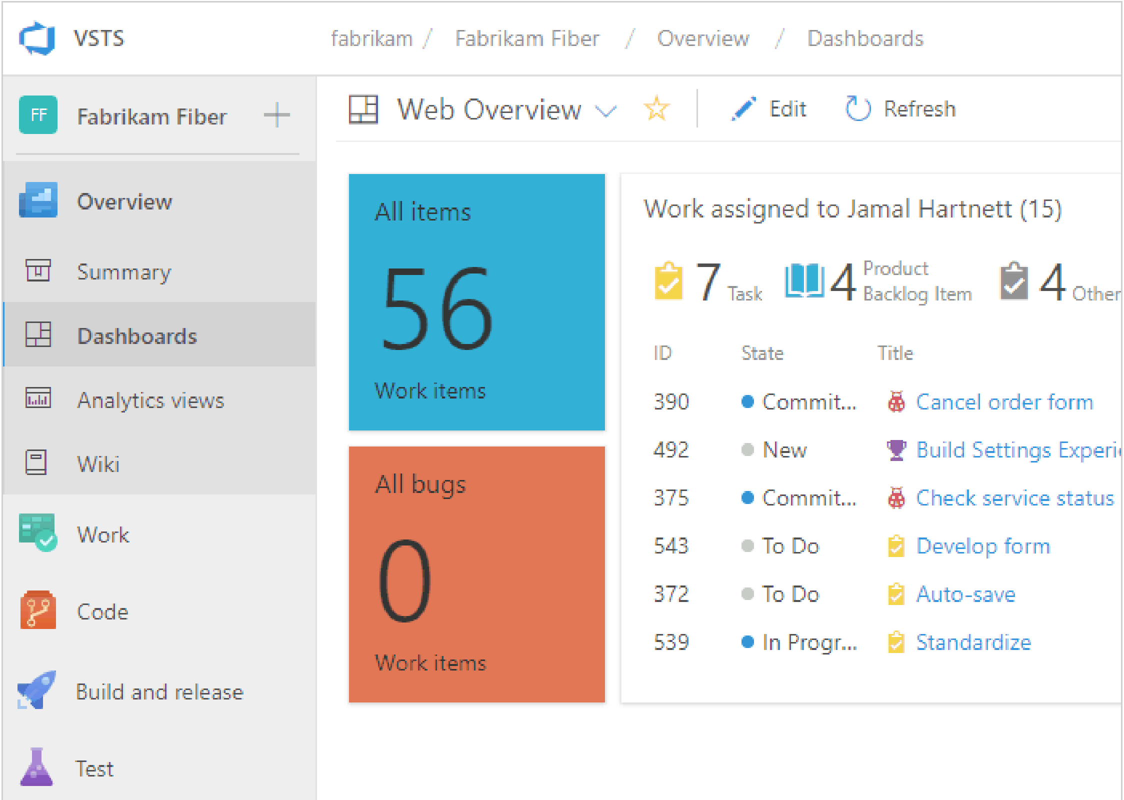
- New navigation
- Previous navigation

- **Switch service or page**: use to switch to a different service or functional area
- **Add an artifact or team**: use to quickly add a work item, Git repo, build or release pipelines, or a new team
- **Switch to a different project**: use to switch to a different project
- **Search box**: use to find code, work items, or wiki content
- **Work across projects**: use to quickly open work assigned to you, your active pull requests, or items you've favorited
- **Work with favorites**: favorite artifacts to support quick navigation
- **Breadcrumbs & selectors**: use to navigate within a service, to open other artifacts or return to a root function
- **Your profile menu**: use to set personal preferences, notifications, and enable preview features
- **Settings**: use to add teams, manage security, and configure other project and organization-level resources.

New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

- New navigation
- Previous navigation

In **New navigation**, you select services—such as **Work**, **Code**, and **Build and Release**—from the sidebar and pages within those services.



New navigation isn't supported on TFS at this time. Choose **Previous navigation** for guidance.

Now that you have an understanding of how the user interface is structured, it's time to get started using it. As you can see, there are a lot of features and functionality.

If all you need is a code repository and bug tracking solution, then start with the Get started with Git and Manage bugs.

To start planning and tracking work, see About Agile tools.

Connect to the web portal, user accounts and licensing

To connect to the web portal, you need your user account added to a project. This is typically done by the organization owner.

Five account users are free as are Visual Studio subscribers and stakeholders. After that, you need to pay for more users. Find out more about licensing from VSTS pricing.

Limited access is available to an unlimited number of stakeholders for free. For details, see Work as a Stakeholder.

To connect to the web portal, you need your user account added to a project. This is typically done by the project administrator.

Limited access is available to an unlimited number of stakeholders for free. For details, see Work as a Stakeholder. Most regular contributors must have a TFS client access license (CAL). All Visual Studio subscriptions include a TFS CAL. Find out more about licensing from TFS pricing.

Refresh the web portal

If data doesn't appear as expected, the first thing to try is to refresh your web browser. Refreshing your client updates the local cache with changes that were made in another client or the server. To refresh the page or object you're currently viewing, refresh the page or choose the **Refresh** icon if available.

To avoid potential errors, you should refresh your client application under the following circumstances:

- Process changes are made
- Work item type definitions are added, removed, renamed or updated
- Area or iteration paths are added, removed, renamed or updated
- Users are added to or removed from security groups or permissions are updated
- A team member adds a new shared query or changes the name of a shared query
- A build definition is added or deleted
- A team or team project is added or deleted.

Differences between the web portal and Visual Studio

Although you can access source code, work items, and builds from both clients, some task-specific tools are only supported in the web browser or an IDE, but not in both.

WEB PORTAL	VISUAL STUDIO

• Product backlog, Portfolio backlogs, Sprint backlogs, Task boards, Capacity planning
• Kanban board
• Dashboards, Widgets, and Charts
• Team rooms
• Request feedback
• Web-based Test Management
• Administration pages to administer accounts, team projects, and teams
	• Task specific interfaces that integrate with Git and TFVC, such as:
	  ◦ **Git:** Changes | Branches | Pull Requests | Sync | Work Items | Builds
	  ◦ **TFVC:** My Work | Pending Changes | Source Control Explorer | Work Items | Builds
	• Greater integration with work items and Office-integration clients. You can open a work item or query result in an office supported client.
	• Additional text formatting options for rich-text fields in work item forms.

Resources

- Project Management
- Project & Organizational Settings

Project Management

8/6/2018 • 2 minutes to read

VSTS | TFS 2018 | TFS 2017 | TFS 2015 | TFS 2013

Add, rename, and delete projects as you scale your organization.

5-Minute Quickstarts

- View permissions
- Look up a project administrator
- Add users to a project or team
- Add administrators or set permissions at the project or collection level

Step-by-Step Tutorials

- Change individual permissions, grant select access to specific functions
- Grant or restrict permissions to select tasks

Concepts

- About areas and iterations
- About teams and Agile tools
- Resources granted to project members

How-to Guides

- Create a project
- Rename a project
- Change service visibility
- Connect to projects

Reference

- Default permissions and access
- Permission lookup guide (Security)

Resources

- New User Guide
- Public Projects
- Security & Identity
- Migrate from TFS to VSTS