Agile development of data science projects

This document describes how a data scientist can execute a data science project in a systematic, version controlled, and collaborative way within a project team by using the Team Data Science Process (TDSP). The TDSP is a framework developed by Microsoft that provides a structured sequence of activities to execute cloud-based, predictive analytics solutions efficiently. For an outline of the personnel roles, and their associated tasks that are handled by a data science team standardizing on this process, see Team Data Science Process: roles and tasks.

This topic includes instructions on how to:

1. So *sprint planning* for work items involved in a project.
2. *Add work items* to sprints.
3. Create and use an *agile-derived work item template* that specifically aligns with TDSP lifecycle stages.

The following instructions outline the steps needed to set up a TDSP team environment using Azure Boards and Azure Repos in Azure DevOps. If your group uses a different code hosting platform, the team lead tasks generally don't change, but the way to complete the tasks is different. For example, linking a work item with a Git branch might not be the same with GitHub as it is with Azure Repos.

The following figure illustrates a typical sprint planning, coding and source-control workflow involved in implementing a data science project:
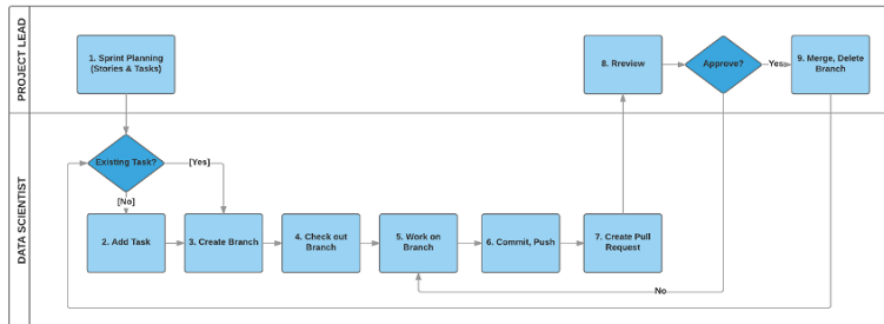

Figure 1: 1

1. Work item types
2. Plan sprints
3. Add a Feature to the backlog
4. Add a User Story to the Feature
5. Add a Task to a User Story
6. Use an agile TDSP work template

1

7. Create Agile Data Science Process work items

1. Work item types

In the TDSP sprint planning framework, there are four frequently used types of **work items**: **Feature**, **User Story**, **Task**, and **Bug**. Each team project maintains a single backlog for all work items. There is no backlog at the git repository level under a team project. Here are their definitions:

- **Feature**: A feature corresponds to a project engagement. Different engagements with a client are considered different Features. Similarly, it is best to consider different phases of a project with a client as different Features. If you choose a schema such as - to name your Features, then you can easily recognize the context of the project/engagement from the names themselves.
- **User Story**: User Stories are work items needed to complete a Feature (project) end-to-end. Examples of User Stories include:
  - Get Data
  - Explore Data
  - Generate Features
  - Build Models
  - Operationalize Models
  - Retrain Models
- **Task**: Tasks are assignable work items that need to be done to complete a specific User Story. For example, Tasks in the User Story *Get data* could be:
  - Get SQL Server credentials
  - Upload data to Azure Synapse Analytics
- **Bug**: Bugs are issues in existing code or documents that must be fixed to complete a Task. If Bugs are caused by missing work items, they can escalate to be User Stories or Tasks respectively.

Data scientists may feel more comfortable using an agile template that replaces Features, User Stories, and Tasks with TDSP lifecycle stages and substages. To create an agile-derived template that specifically aligns with the TDSP lifecycle stages, see Use an agile TDSP work template.

> [AZURE.NOTE] TDSP borrows the concepts of Features, User Stories, Tasks, and Bugs from software code management (SCM). The TDSP concepts might differ slightly from their conventional SCM definitions.

2. Plan sprints

Many data scientists are engaged with multiple projects, which can take months to complete and proceed at different paces. Sprint planning is useful for project prioritization, and resource planning and allocation. In Azure Boards, you can

easily create, manage, and track work items for your projects, and conduct sprint planning to ensure projects are moving forward as expected.

For more information about sprint planning, see Scrum sprints.

For more information about sprint planning in Azure Boards, see Assign backlog items to a sprint.

3. Add a Feature to the backlog

After your project and project code repository are created, you can add a Feature to the backlog to represent the work for your project. For details on how to create a project click here. For details on how to create a code repository click here

1. From your project page, select **Boards > Backlogs** in the left navigation.

2. On the **Backlog** tab, if the work item type in the top bar is **Stories**, drop down and select **Features**. Then select **New Work Item**.
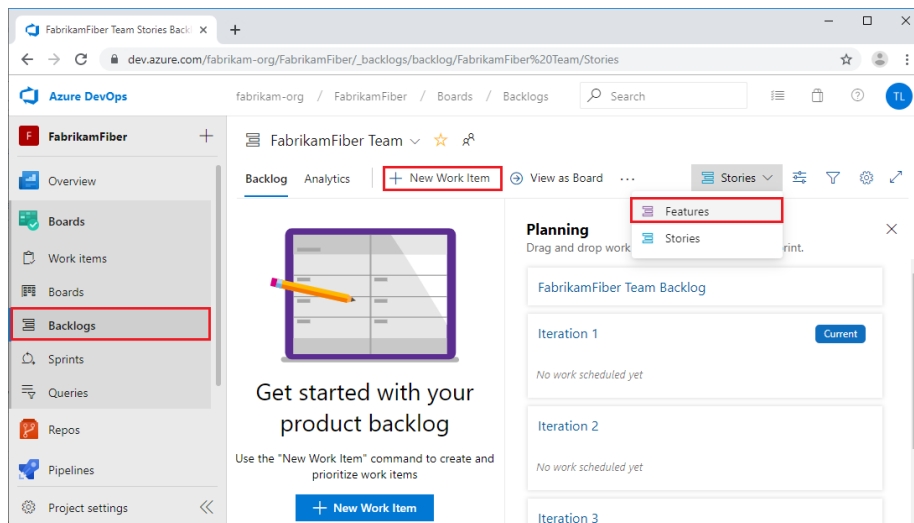
3. Enter a title for the Feature, usually your project name, and then select **Add to top**.
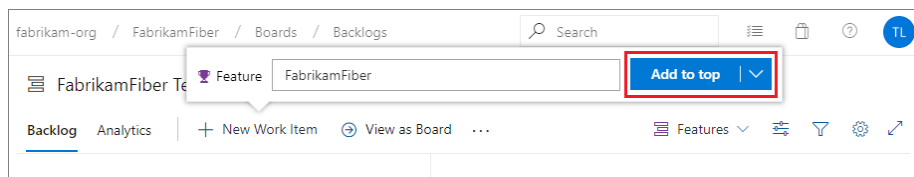
4. From the **Backlog** list, select and open the new Feature. Fill in the description, assign a team member, and set planning parameters.

You can also link the Feature to the project's Azure Repos code repository by selecting **Add link** under the **Development** section.

After you edit the Feature, select **Save & Close**.
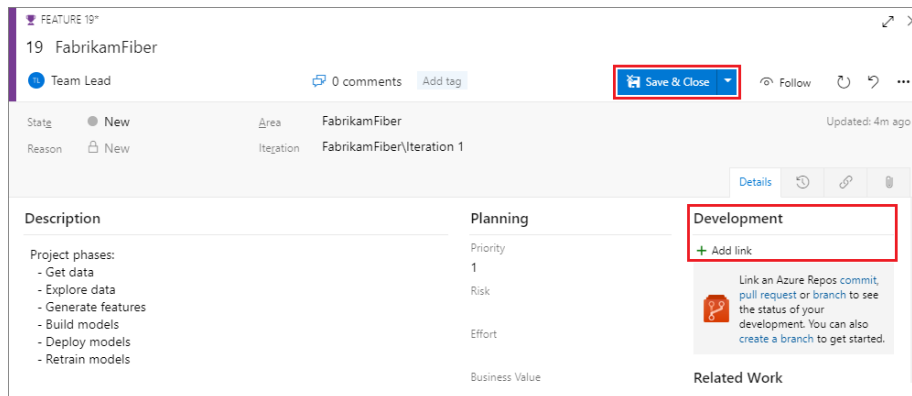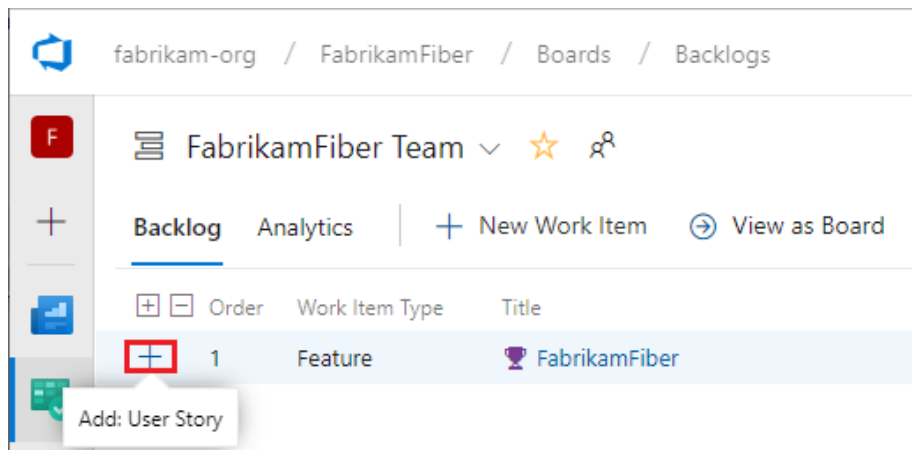

4. Add a User Story to the Feature

Under the Feature, you can add User Stories to describe major steps needed to complete the project.

To add a new User Story to a Feature:

1. On the Backlog tab, select the + to the left of the Feature.

2. Give the User Story a title, and edit details such as assignment, status, description, comments, planning, and priority.

You can also link the User Story to a branch of the project's Azure Repos code repository by selecting **Add link** under the **Development** section. Select the repository and branch you want to link the work item to, and then select **OK**.
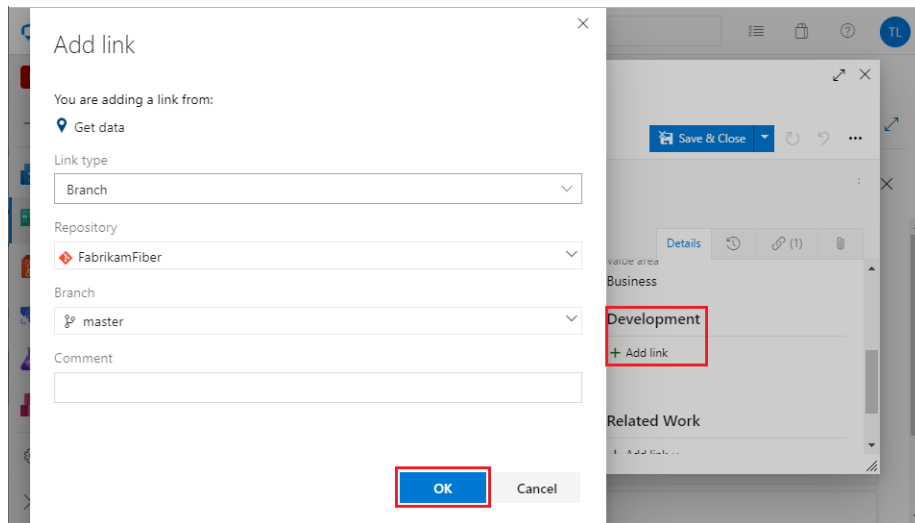
3. When you're finished editing the User Story, select **Save & Close**.

5. Add a Task to a User Story

Tasks are specific detailed steps that are needed to complete each User Story. After all Tasks of a User Story are completed, the User Story should be completed too.

To add a Task to a User Story, select the + next to the User Story item, and select **Task**. Fill in the title and other information in the Task.
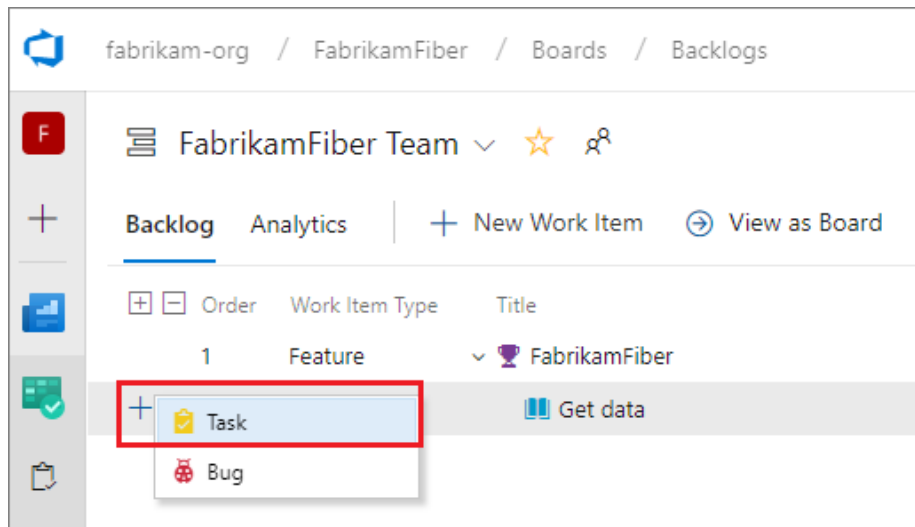

After you create Features, User Stories, and Tasks, you can view them in the **Backlogs** or **Boards** views to track their status.
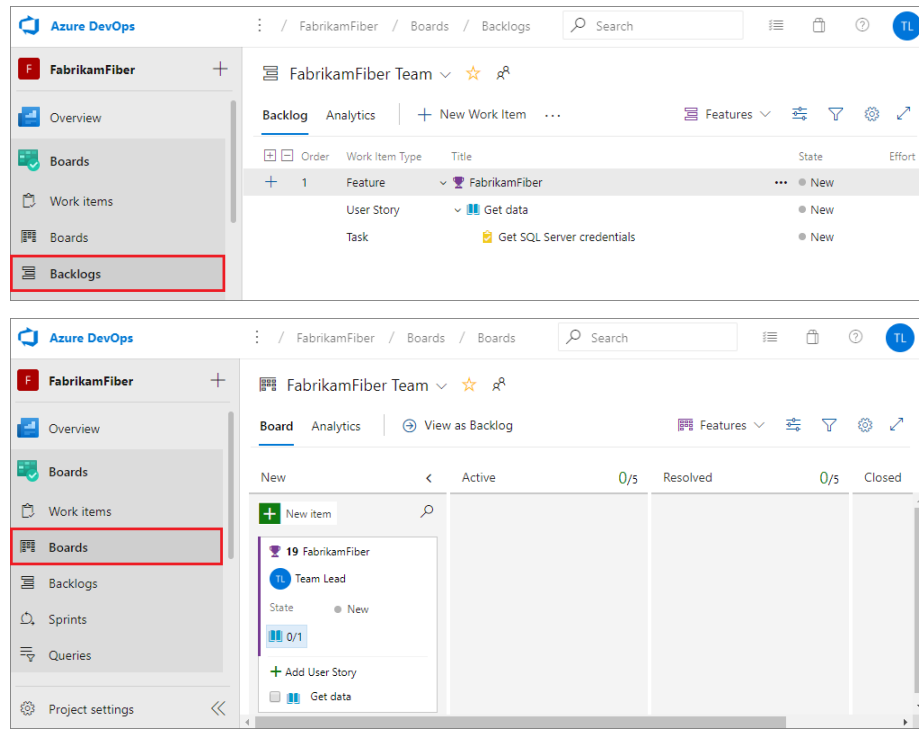
6. Use an agile TDSP work template

1. From your Azure DevOps organization main page, select **Organization settings** from the left navigation.

2. In the **Organization Settings** left navigation, under **Boards**, select **Process**.

3. In the **All processes** pane, select the ... next to **Agile**, and then select **Create inherited process**.
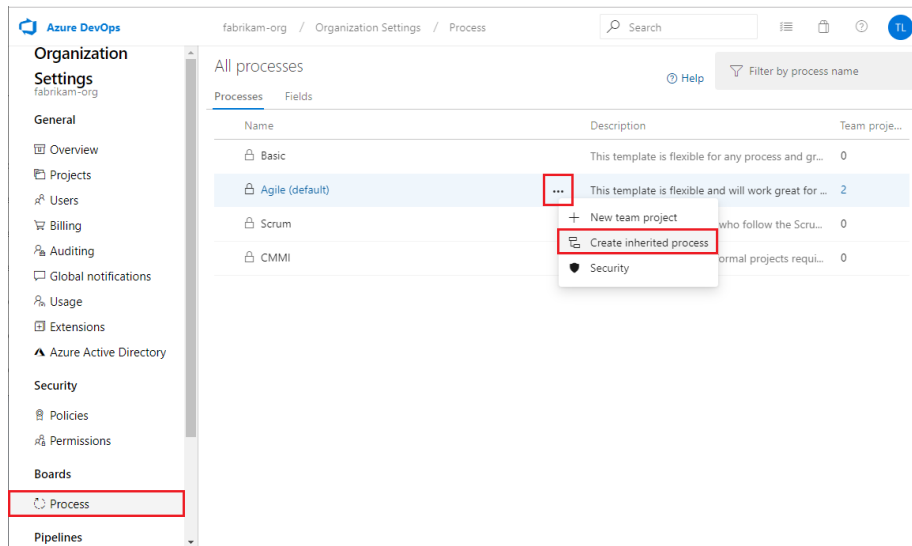
4. In the **Create inherited process from Agile** dialog, enter the name *AgileDataScienceProcess*, and select **Create process**.

Create inherited process from Agile

Create a new inherited process to enable customizations.

🔒 Agile [system process]

⌐ AgileDataScienceProcess ✱

Description

Learn more ↗

Create process   Cancel

5. In **All processes**, select the new **AgileDataScienceProcess**.

6. On the **Work item types** tab, disable **Epic, Feature, User Story,** and **Task** by selecting the ... next to each item and then selecting **Disable**.
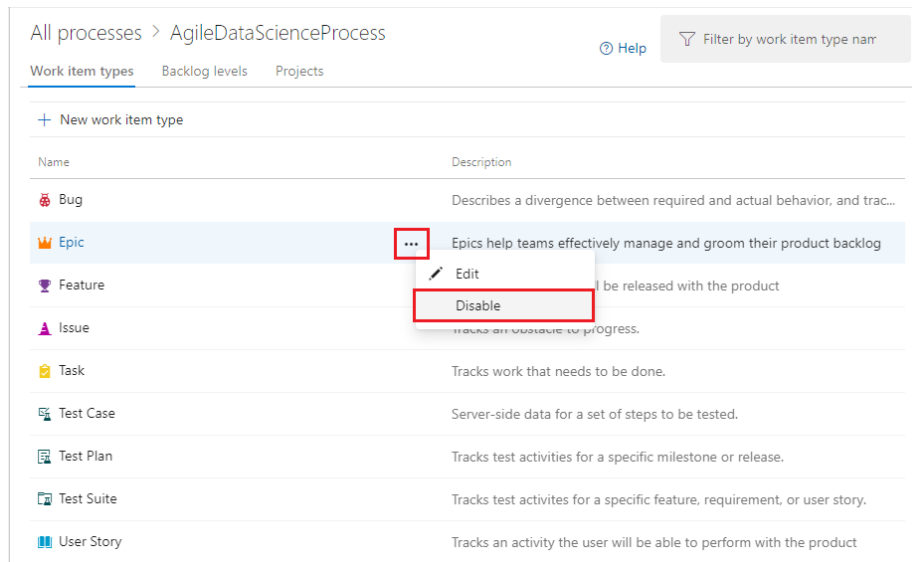
7. In **All processes**, select the **Backlog levels** tab. Under **Portfolios backlogs**, select the ... next to **Epic (disabled)**, and then select **Edit/Rename**.

8. In the **Edit backlog** level dialog box:

a. Under **Name**, replace **Epic** with *TDSP Projects*.
b. Under **Work item types on this backlog level**, select **New work item type**, enter *TDSP Project*, and select **Add**.
c. Under **Default work item** type, drop down and select **TDSP Project**.
d. Select **Save**.

9. Follow the same steps to rename **Features** to *TDSP Stages*, and add the following new work item types:

- *Business Understanding*
- *Data Acquisition*
- *Modeling*
- *Deployment*

10. Under **Requirement backlog**, rename **Stories** to *TDSP Substages*, add the new work item type *TDSP Substage*, and set the default work item type to **TDSP Substage**.

11. Under **Iteration backlog**, add a new work item type *TDSP Task*, and set it to be the default work item type.

After you complete the steps, the backlog levels should look like this:

7. Create Agile Data Science Process work items

You can use the data science process template to create TDSP projects and track work items that correspond to TDSP lifecycle stages.

1. From your Azure DevOps organization main page, select **New project**.

2. In the **Create new project** dialog, give your project a name, and then select **Advanced**.

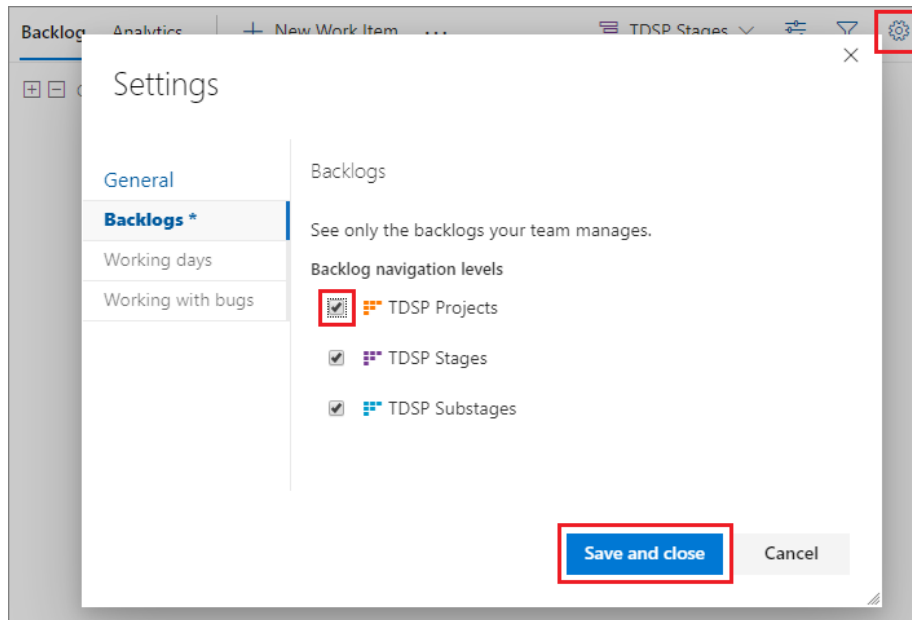3. Under **Work item process**, drop down and select **AgileDataScience-Process**, and then select **Create**.

4. In the newly created project, select **Boards > Backlogs** in the left navigation.

5. To make TDSP Projects visible, select the **Configure team settings** icon. In the **Settings** screen, select the **TDSP Projects** check box, and then select **Save and close.**

6. To create a data science-specific TDSP Project, select **TDSP Projects** in the top bar, and then select **New work item.**

7. In the popup, give the TDSP Project work item a name, and select **Add to top.**
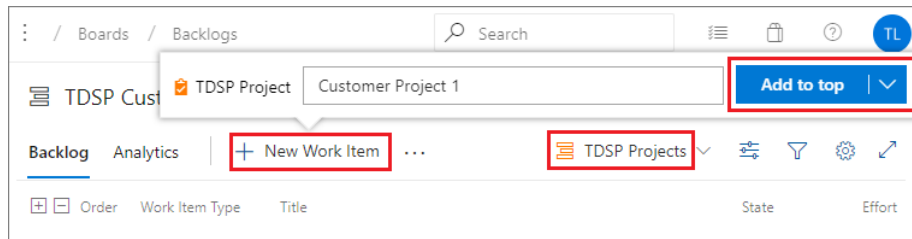

8. To add a work item under the TDSP Project, select the + next to the project, and then select the type of work item to create.
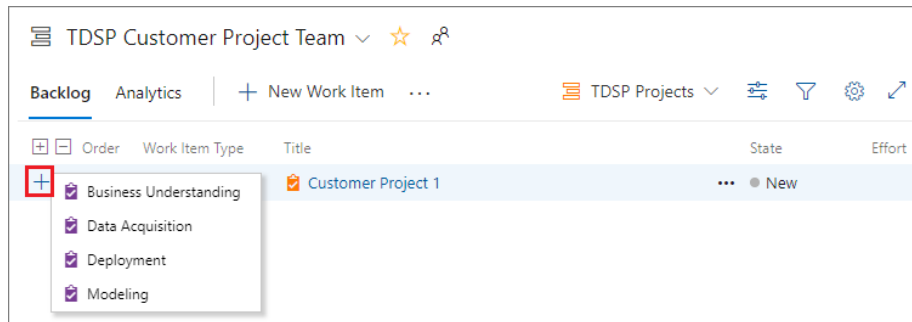
9 Fill in the details in the new work item, and select **Save & Close.**

10. Continue to select the + symbols next to work items to add new TDSP Stages, Substages, and Tasks.

Here is an example of how the data science project work items should appear in Backlogs view:
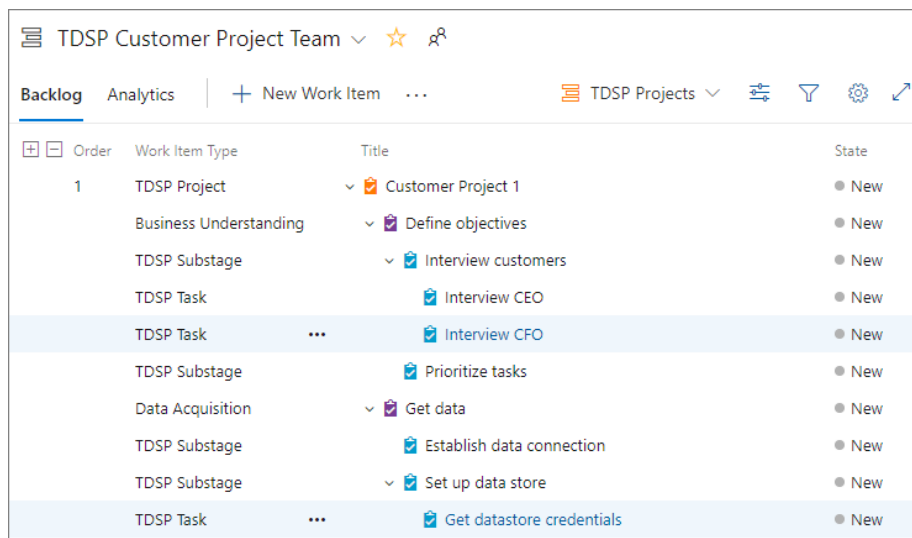
Next steps

Collaborative coding with Git describes how to do collaborative code development for data science projects using Git as the shared code development framework, and how to link these coding activities to the work planned with the agile process.