Pushing changes from your local machine to Team Foundation Server (TFS) involves several steps. TFS can be used with both Git and TFVC (Team Foundation Version Control). Below, I'll explain how to push changes using both systems.

Using TFS with Git:

1. Ensure Your Local Repository is Connected to TFS:

- **Clone the Repository:**
  - If you haven't already, clone the repository from TFS to your local machine.
  - Use the command:
    ```bash
    git clone https://tfs-server-url/path-to-your-repo.git
    ```
  - Alternatively, you can use Visual Studio's "Clone Repository" feature.

- **Navigate to the Repository:**
  - Change to the directory of the repository:
    ```bash
    cd path-to-your-repo
    ```

2. Make Changes Locally:

- Modify the files you need to change.
- Use `git status` to check which files have been modified.

2. Make Changes Locally:

- Modify the files you need to change.
- Use `git status` to check which files have been modified.

3. Stage the Changes:

- Stage the files that you want to commit:

```bash
git add .
```

- This stages all changes. If you want to stage specific files, list them individually:

```bash
git add file1 file2
```

4. Commit the Changes:

- Commit your changes locally with a meaningful commit message:

```bash
git commit -m "Your commit message here"
```

5. Push the Changes to TFS:

- Push the committed changes to the remote repository on TFS:

```bash
git push origin branch-name
```

↓

- Replace `branch-name` with the name of the branch you're working on, e.g., `master` or

```
git push origin branch-name
```

- Replace `branch-name` with the name of the branch you're working on, e.g., `master` or `develop`.

Using TFS with TFVC:

If you're using TFVC instead of Git, the process is slightly different. TFVC is a centralized version control system, so the workflow involves checking out files, editing them, and then checking them back in.

1. Connect to TFS:

- Open Visual Studio.
- Go to `Team > Manage Connections > Connect to a Project`.
- Select your TFS project and connect to it.

2. Get the Latest Version:

- Make sure your workspace is up to date with the latest changes:
  - Right-click on your solution or project in Solution Explorer.
  - Select `Get Latest Version (Recursive)`.

3. Check Out the Files:

- In TFVC, files are automatically checked out when you edit them.
- You can also manually check out files by right-clicking them and selecting `Check Out for Edit`.

4. Make Changes Locally:

- Edit your files as needed.

- You can also manually check out files by right-clicking them and selecting `Check Out for Edit`.

4. Make Changes Locally:

- Edit your files as needed.

5. Check In Changes:

- Once you've made your changes, check them in:
    - Right-click on your solution or project in Solution Explorer.
    - Select `Check In`.
- You can provide a comment describing the changes.

6. Resolve Conflicts (if any):

- If someone else has made changes to the same files, you might need to resolve conflicts before checking in.

7. Shelve Changes (Optional):

- If you're not ready to check in but want to save your work, you can shelve your changes:
    - Go to `Team Explorer`.
    - Click `Shelve Pending Changes`.
    - Provide a name for the shelveset and save it.

Additional Tips:

- **Branching and Merging:** If you're working on a feature branch, ensure you're merging your changes into the correct branch before pushing or checking in.
- **Build and Test:** Always build and test your changes locally before pushing them to avoid breaking the build for others.