What is GitHub?

GitHub is an open-source code hosting platform having a version control system. Using GitHub, we can work with the team together on projects from anywhere, anytime. We can host our source code projects in different programming languages and keep track of the changes made by the team using Git.

Below are the steps to add your VS project to Git and upload it to GitHub.

**Step 1**

Open your project in Visual Studio and check the bottom for 'Add to Source Control'.

Graphical user interface, application

Description automatically generated

**Step 2**

Select Git as your version control, as displayed on the below screen.

Graphical user interface, application

Description automatically generated

**Step 3**

It will ask you to save your project solution. Just click on Yes.

Graphical user interface, application

Description automatically generated

**Step 4**

Now, go to Github.com and click on "Start a project", as shown on the below screen. (Before this, you need to create your GitHub account).

Graphical user interface, text, application

Description automatically generated

**Step 5**

Once you start, it will ask you to create a new repository. So here, I am going to give the name 'VersionControlWebApp'. Add some sort of description.

Chose Public or Private on the basis of how you want to manage your repository and click the 'Create Repository' button.

Graphical user interface, text, application, email

Description automatically generated

**Step 6**

In the Code section, there is a 'Quick setup' link as shown below screen.

Graphical user interface, text, application, email

Description automatically generated

**Step 7**

Now, from your project, open Team Explorer - Synchronization, go to 'Push to Remote Repository', and paste the URL from GitHub (shown in step 6), and click on 'Publish'.

![Graphical user interface, text, application, email

Description automatically generated]()

**Step 8**

Once you click, it will push your code branch to GitHub.

![Graphical user interface, text, application, email

Description automatically generated]()

**Step 9**

After publishing, just log in to GitHub with your login credentials.

![Graphical user interface, application

Description automatically generated]()

**Step 10**

Now, from Visual Studio, go to Team Explorer. There, you can see Changes, Branches, Sync, Settings options.

Graphical user interface, application

Description automatically generated

A [Project](https://vercel.com/docs/concepts/projects/overview) groups deployments and custom domains. When you visit the dashboard, you'll see a list of all your Projects. There are two ways to create a new Project on Vercel:

* **Import your existing project**. This can be any web project that outputs static HTML content (such as a website that contains HTML, CSS, and JavaScript), but it will work best when using one of our supported frameworks
* **Use a**[**template**](https://vercel.com/templates)

This walkthrough follows the steps to import a project. If you would prefer to use a template, select your [favorite template](https://vercel.com/templates/next.js/blog-starter-kit), and click **Deploy**.

### [Import an existing project](https://vercel.com/docs/concepts/get-started/deploy#import-an-existing-project)

1. On the ["New Project"](https://vercel.com/new) page, under the "Import Git Repository" section, choose the account that you'd like to link a project from.
2. Find the repository in the list and select **Import**.
3. Vercel will automatically detect the framework and any necessary build settings. However, you can also configure the Project settings at this point including the [build and development settings](https://vercel.com/docs/concepts/deployments/configure-a-build#build-and-development-settings) and [Environment Variables](https://vercel.com/docs/concepts/projects/environment-variables). These can also be set later.
4. Press the **Deploy** button. Vercel will create the Project and deploy it based on the chosen configurations.
5. Enjoy the confetti!

To view your deployment, click on the Project in the dashboard and then click on the **Domain**. This page is now visible to anyone who has the URL.

### [What's next](https://vercel.com/docs/concepts/get-started/deploy#what's-next)

* **Develop** - Your deployment is fully integrated with Git, which means that you can continue developing your project within your normal developer workflow. You can use the development command your framework provides (such as next dev for Next.js) to start running locally.
* **Preview** - When you push to a Git branch, Vercel will create a preview deployment. You can find a link to this deployment within the body of your pull request.
* **Ship** - When you push to your production branch (usually main), Vercel ships your changes to production, making them accessible to everyone.