# Hands-on lab

## Lab: Setup Azure DevOps

September 2020

In this lab, you will set-up a (free) account for Azure DevOps, which gives you the opportunity to work on unlimited private projects with up to 5 team members and practice some DevOps processes.

You will create a git repo and set-up your git commandline client to access this repo.

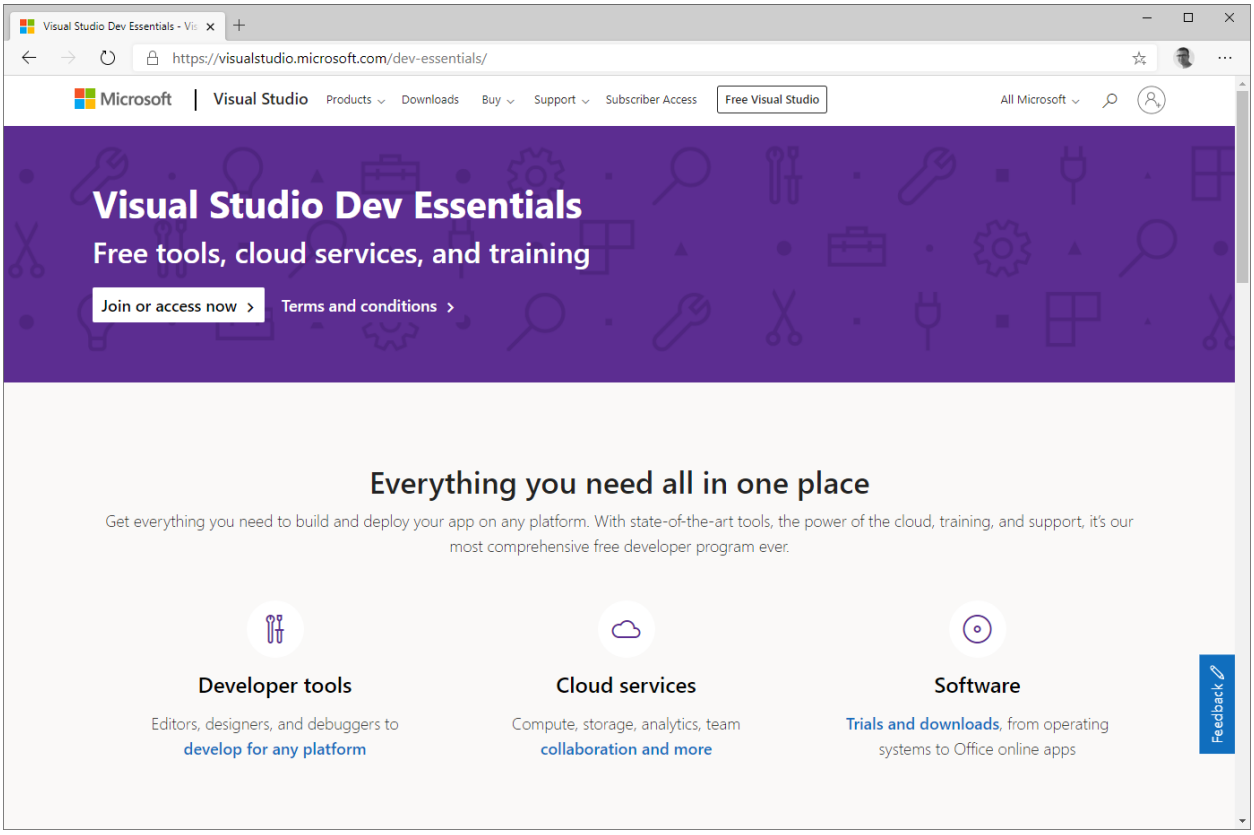
## Step 1: Use your PXL account, this is a Microsoft account!

Because your PXL mail address is known by Microsoft (it is an Office 365 student account), you can use this account for logging into Azure DevOps.

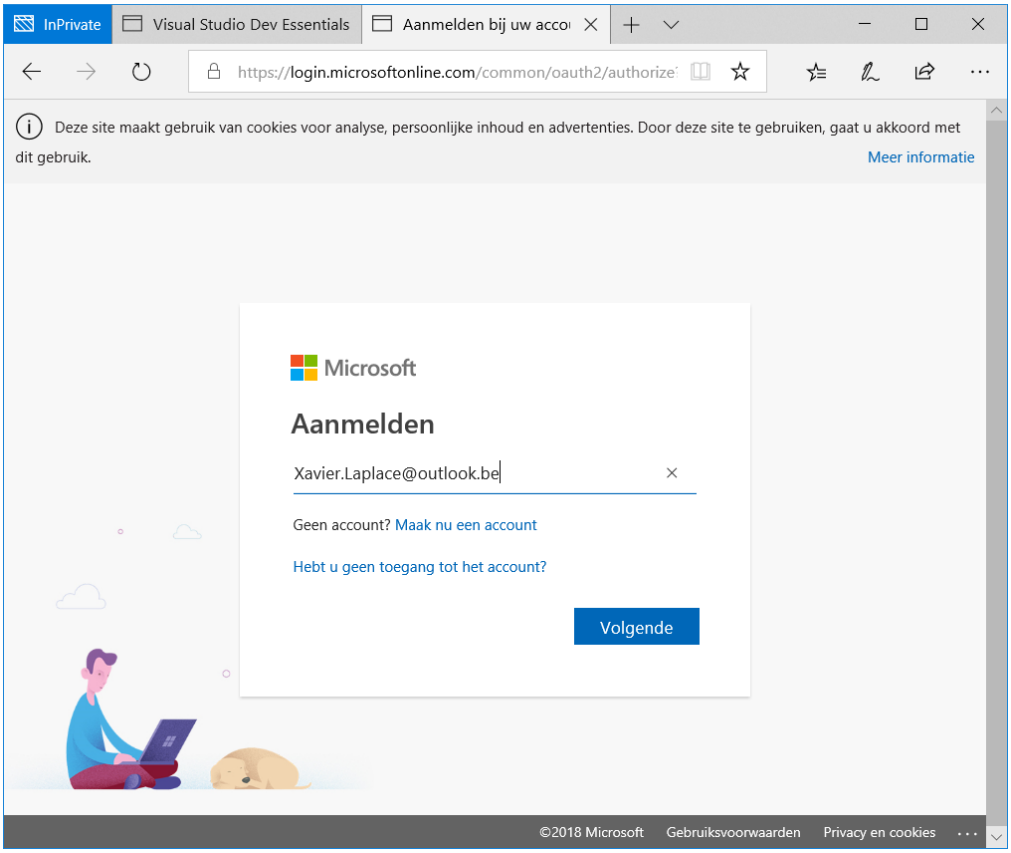
> **Remark**
> While you *could* use your own account (or Github account) for registering with Azure DevOps, using an organization account makes it easier for finding and adding known users to an existing project.

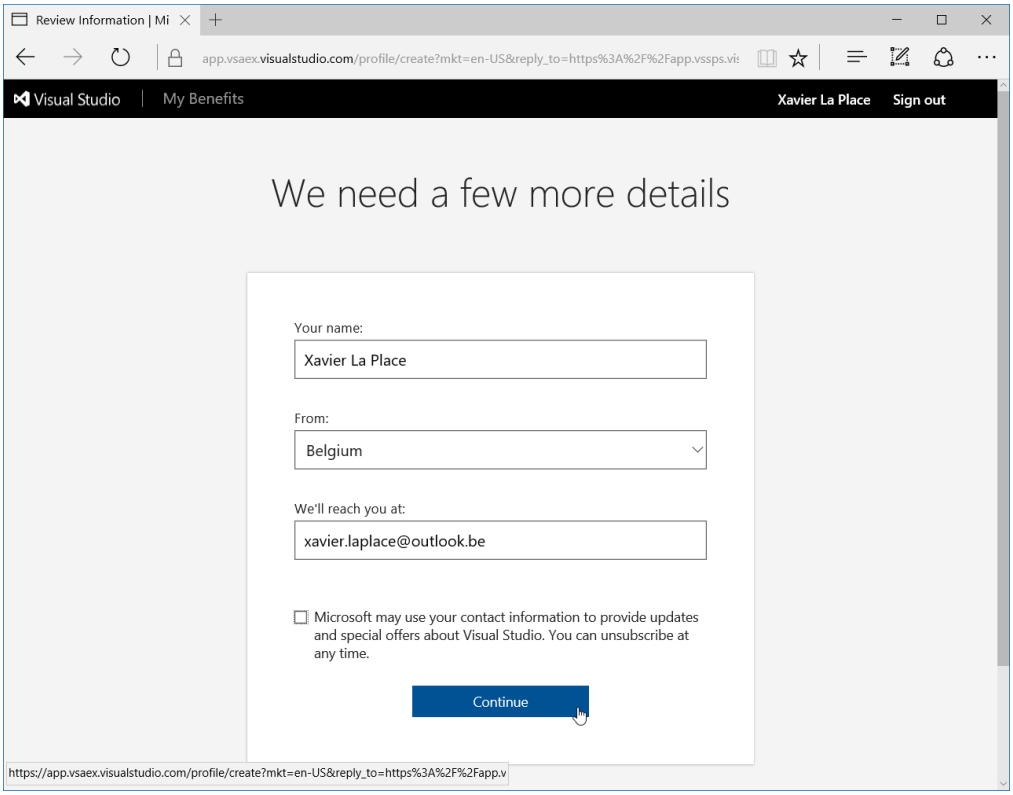
## Step 2: Log in to Visual Studio Dev Essentials

Visit [https://www.visualstudio.com/dev-essentials/](https://www.visualstudio.com/dev-essentials/) and hit the "Join or access now"-button.



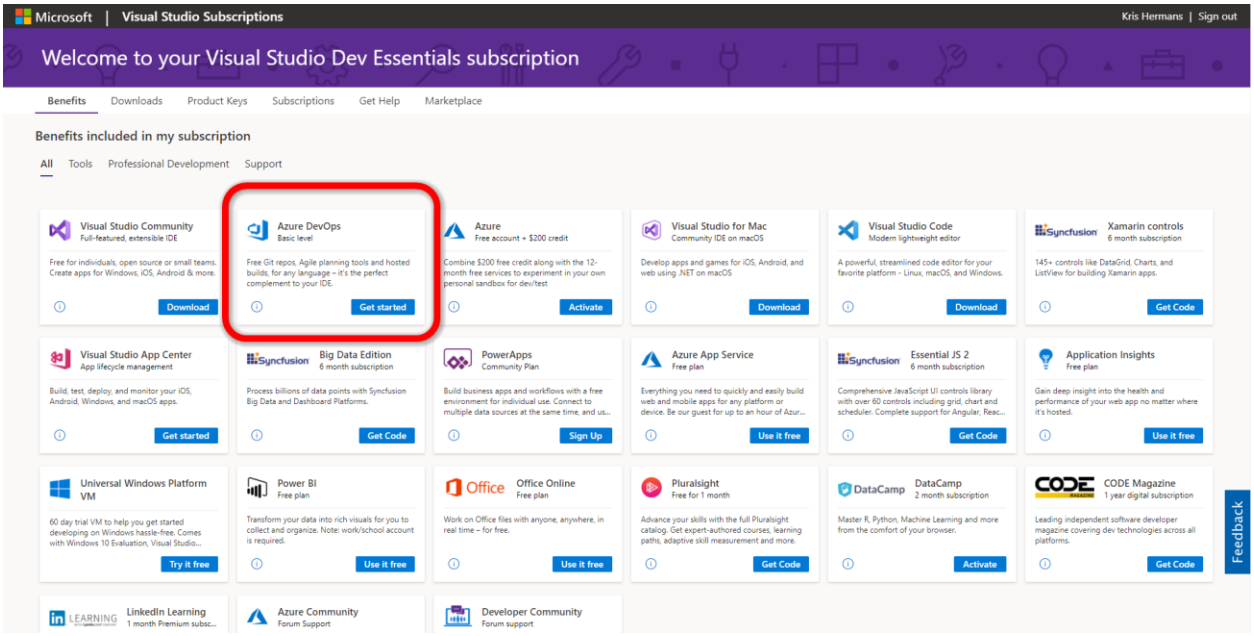

Next, log in with your (school) Microsoft account:


InPrivate
Visual Studio Dev Essentials
Aanmelden bij uw acco
https://login.microsoftonline.com/common/oauth2/authorize
Deze site maakt gebruik van cookies voor analyse, persoonlijke inhoud en advertenties. Door deze site te gebruiken, gaat u akkoord met dit gebruik.
Meer informatie
Microsoft
Aanmelden
Xavier.Laplace@outlook.be
Geen account? Maak nu een account
Hebt u geen toegang tot het account?
Volgende
©2018 Microsoft
Gebruiksvoorwaarden
Privacy en cookies

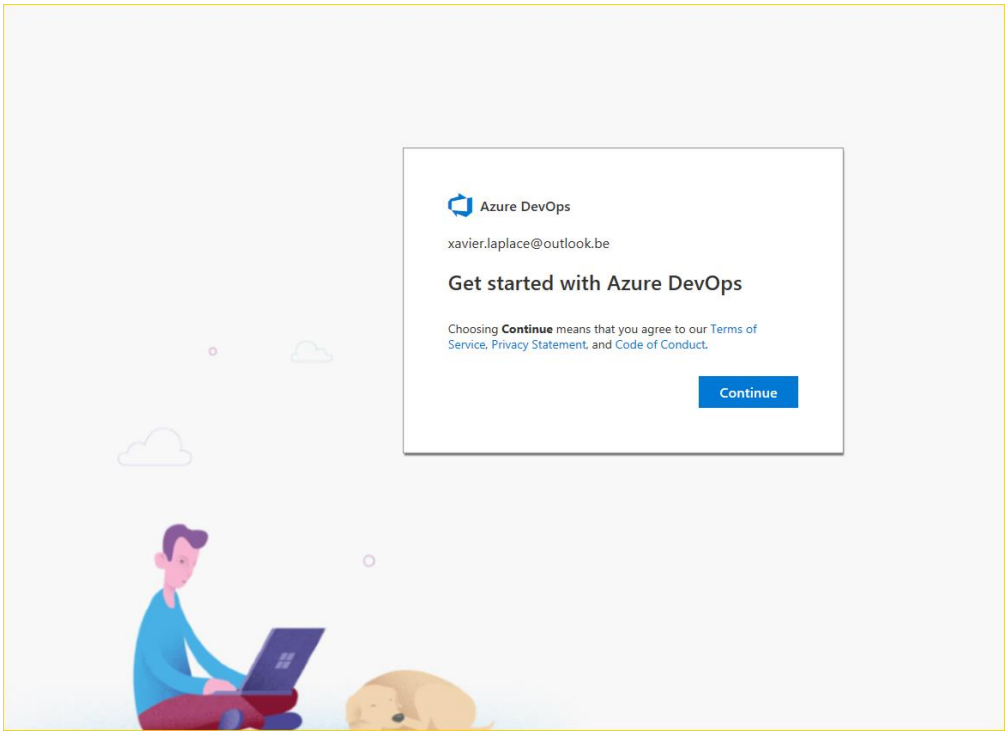

Review Information | Mi
app.vsaex.visualstudio.com/profile/create?mkt=en-US&reply_to=https%3A%2F%2Fapp.vssps.vis
Visual Studio
My Benefits
Xavier La Place
Sign out
We need a few more details
Your name:
Xavier La Place
From:
Belgium
We'll reach you at:
xavier.laplace@outlook.be
Microsoft may use your contact information to provide updates and special offers about Visual Studio. You can unsubscribe at any time.
Continue
https://app.vsaex.visualstudio.com/profile/create?mkt=en-US&reply_to=https%3A%2F%2Fapp.v

Now you arrive on a landing page with all the offerings from Visual Studio Dev Essentials. Some interesting features are:

- $200 yearly free credits on Azure (you could use this to host a web app)
- A free plan for Visual Studio App Center (DevOps for mobile apps)
- Azure DevOps. This is the one we need here.

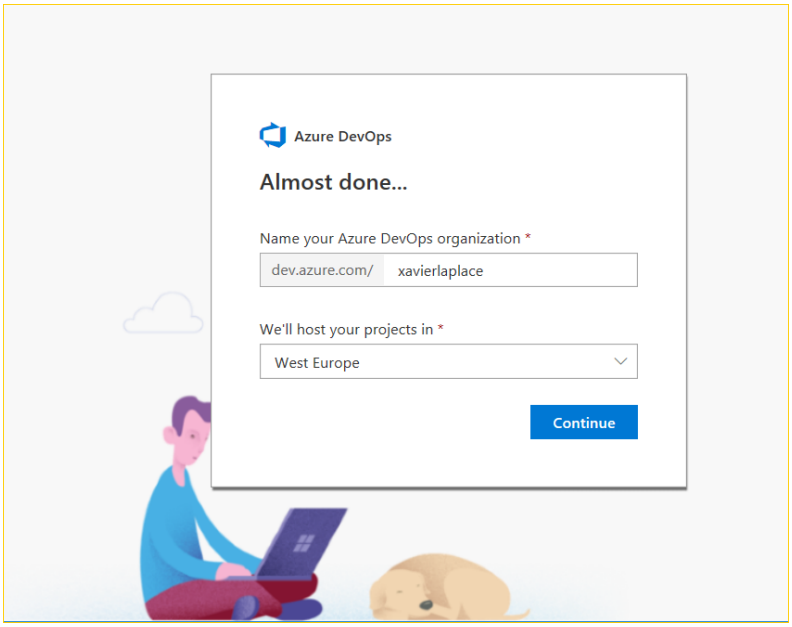



Click on the “Get started” link for Azure DevOps, possibly you will have to accept the “Terms and Conditions”.

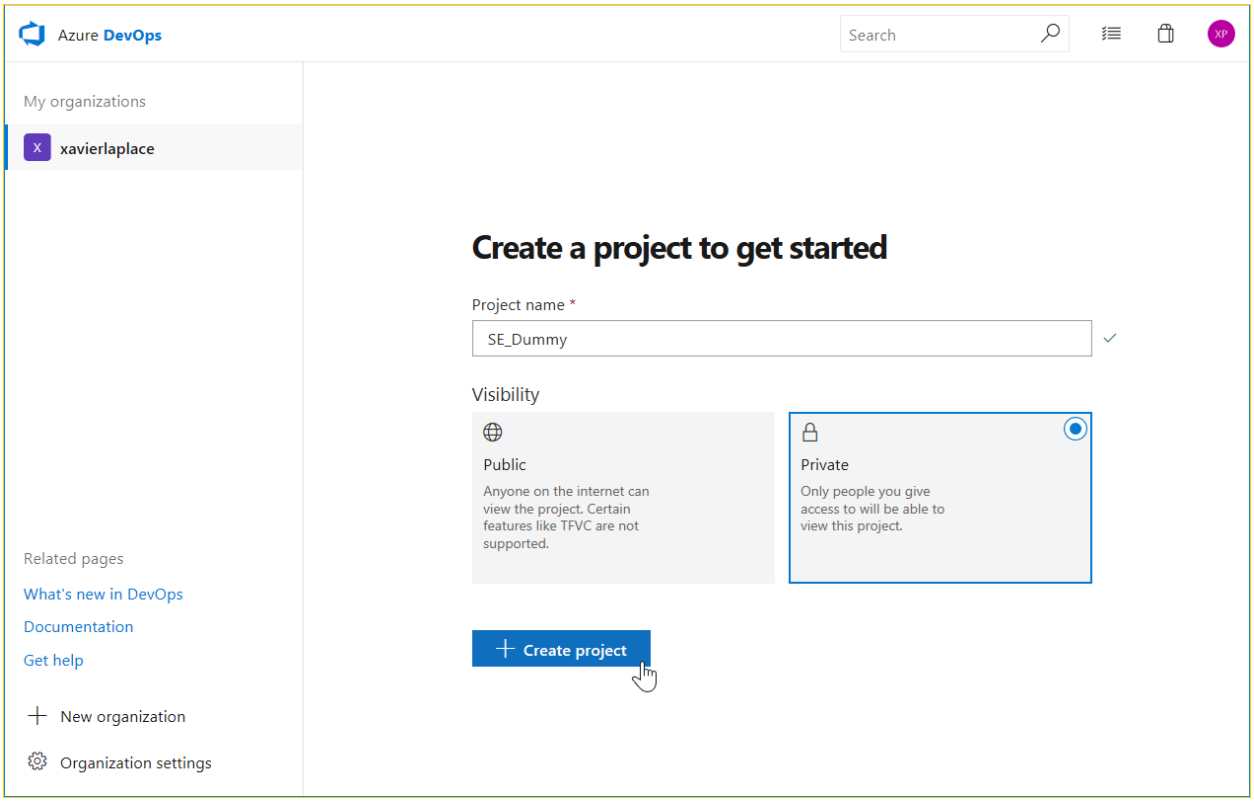



In Azure DevOps, you organize your projects into **organizations**. An organization can be shared with multiple users (up to 5 for a free account). You can create multiple organizations, e.g. to group related projects.

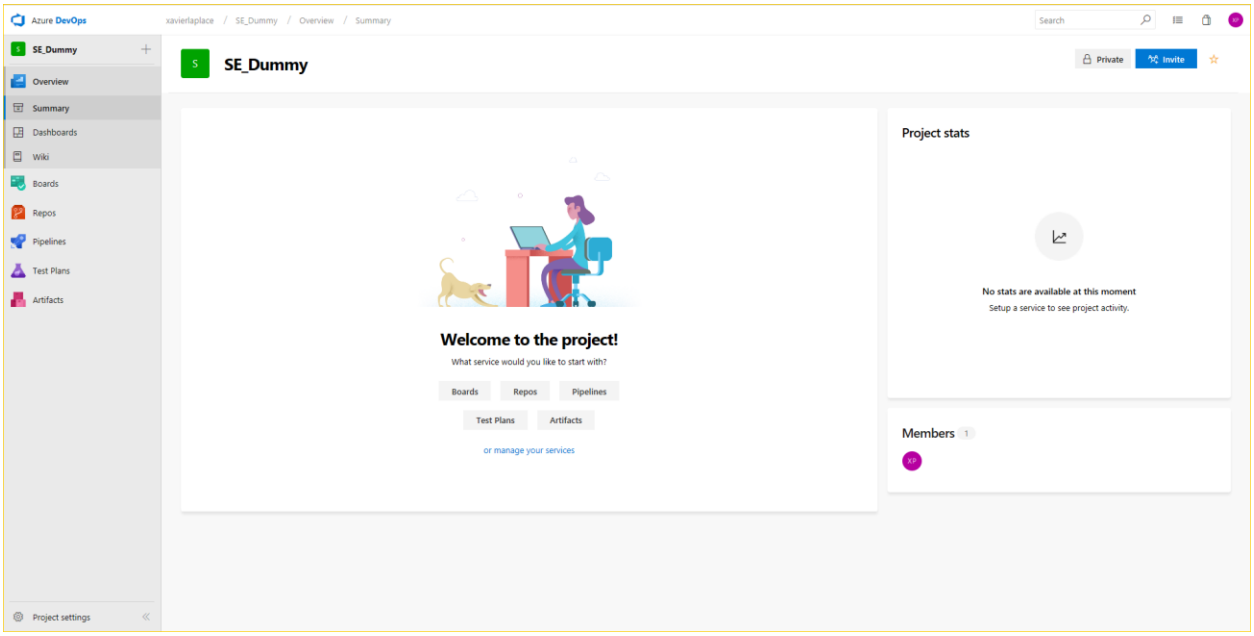
For now, just create 1 organization named after your username. E.g.: dev.azure.com/**xavierlaplace**




Next, create your first project. Name it “SE_Dummy” and choose for private Visibility.




Now you arrive on the landing page of your project and you can get to work. Notice the invite button on the upper right corner to add other team members to your project.



## Step 3: Get yourself familiarized with the docs

Documentation for Azure DevOps is nicely organized. We will give an overview and some directions to explore.

https://docs.microsoft.com/en-us/azure/devops

This is the landing page for all the docs.

https://docs.microsoft.com/en-us/azure/devops/organizations/accounts/create-organization

This gives more info on creating organizations. We created the organization “XavierLaPlace” (or: YourUserName). For this course, only one organization is sufficient, but you could create more than one if needed.

https://docs.microsoft.com/en-us/azure/devops/organizations/security/add-users-team-project

This tutorials helps you to create projects within an organization and add users to a team. Do this for your own project. One team member creates a project in his/her organization and adds the other team members.

https://docs.microsoft.com/en-us/azure/devops/pipelines/create-first-pipeline

Gives you more info on building pipelines.

> **Remark**
>
> If you follow along, this don’t go exactly as is told in this tutorial. More specifically: you have to edit the azure-pipelines.yml by hand in your repo, push it, end then a build will start.

https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema

Editing yaml-files is the preferred way for building pipelines. Familiarize yourself with the syntax and its possibilities.

https://docs.microsoft.com/en-us/azure/devops/pipelines/tasks/index

This is an extensive Task catalog for things you can automate within your build pipelines.

## Step 4: create a git repo

On the left side of your SE_Dummy project, click on “Repos”. Now you see the default SE_Dummy git repo that gets created when you created the Project.

> **Remark**
>
> It’s important to realize that you can have *multiple* repo’s per Project. For instance, if your project involves a front-end and a back-end solution, you can now separate this into two repo’s.

From this page you have multiple options to get started:

- Clone the repo to your computer
- Push existing code
- Initialize with a README or .gitignore

Initialize your repo with a README and try to clone is to your computer using your own preferred git tools. Next push some changes to the repo and verify that changes are shown on the site.

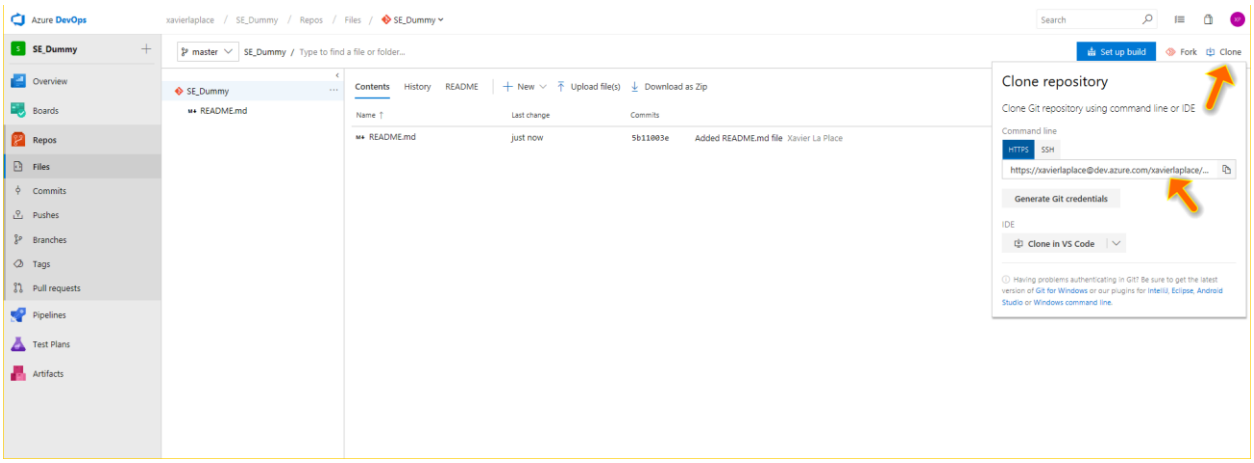
> **Remark**
>
> The repository that you created is a private repository. Azure DevOps gives away unlimited private repos for everyone!

## Step 4: Clone your repo using the command line client

Now inside the git prompt, navigate to a folder where you can clone the SE_Dummy repo. E.g. on your desktop (for testing purposes). Now create a folder “vsts” where you can checkout the repo:

```
cd Desktop
mkdir devtest
cd devtest
```

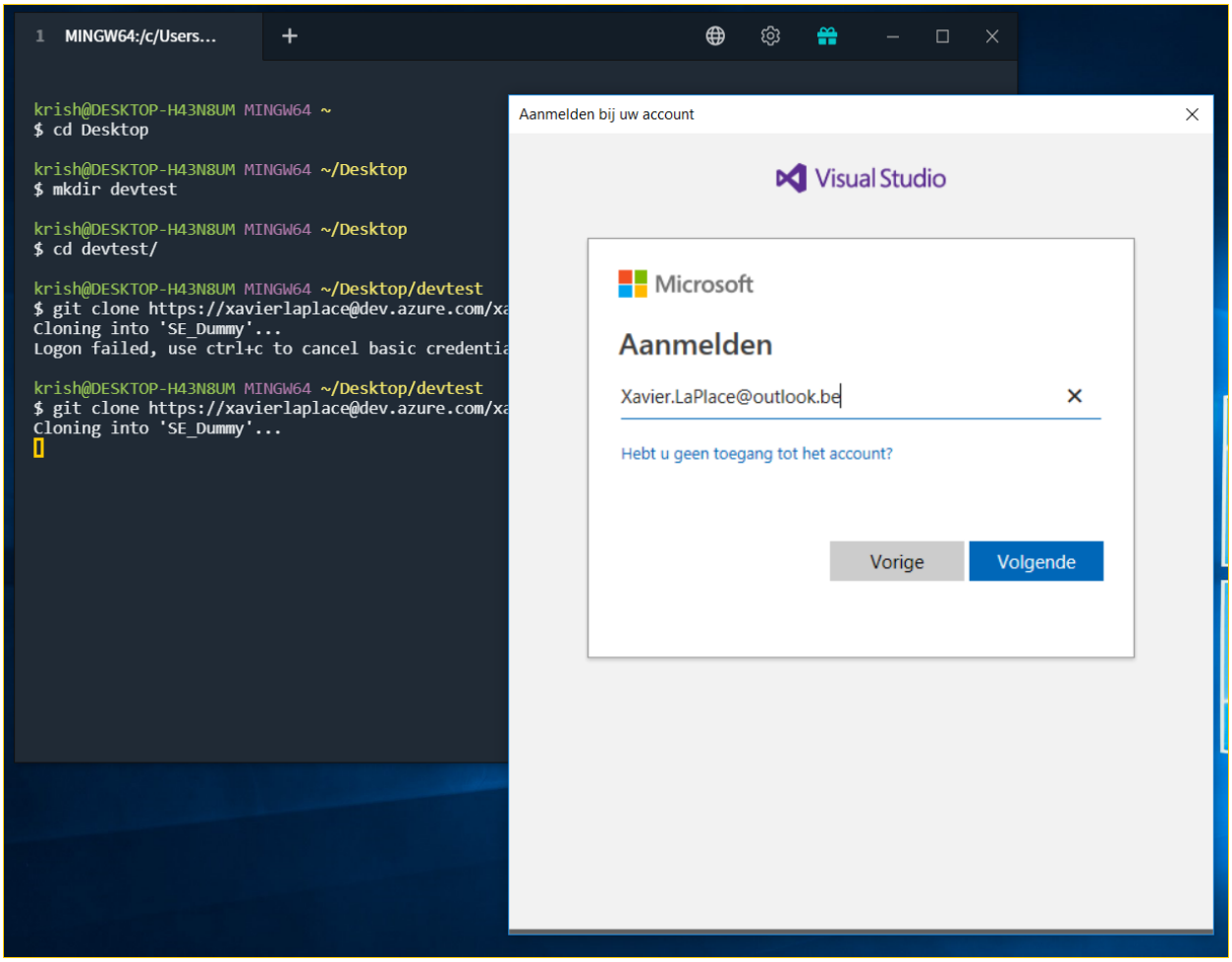
Now find out the clone URL, this you find on the Project page.




Issue the clone command in the prompt:

```
git clone https://enz
```

If all went well, you will be prompted by the Microsoft authenticator to gain access to your repo. Once authenticated, you don't have to repeat your credentials for every push/pull/etc.



**Remark**

You need your PXL account for gaining access to this repository! Not your Github account!

If you don't get a prompt for authentication, you need to install this with your git client. More info:

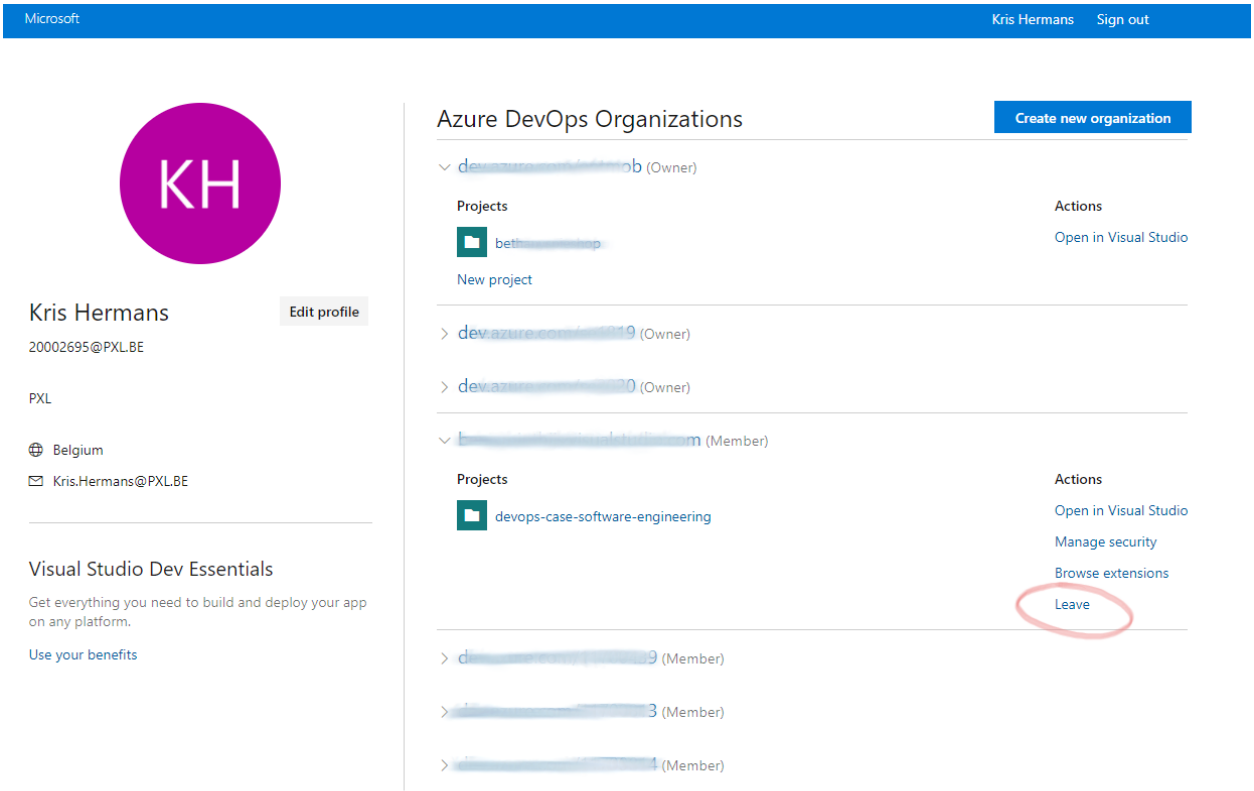
https://git-for-windows.github.io/
https://www.visualstudio.com/en-us/docs/git/set-up-credential-managers

## How to remove yourself from an Azure DevOps organization

If you want to leave an organization in which you are a member (because the project has ended), you van visit the following link:

https://www.tanchunsiong.com/how-to-remove-yourself-from-azure-devops-visual-studio-vsts-online-organization/

Expand the organization link you want to leave and hit "Leave".

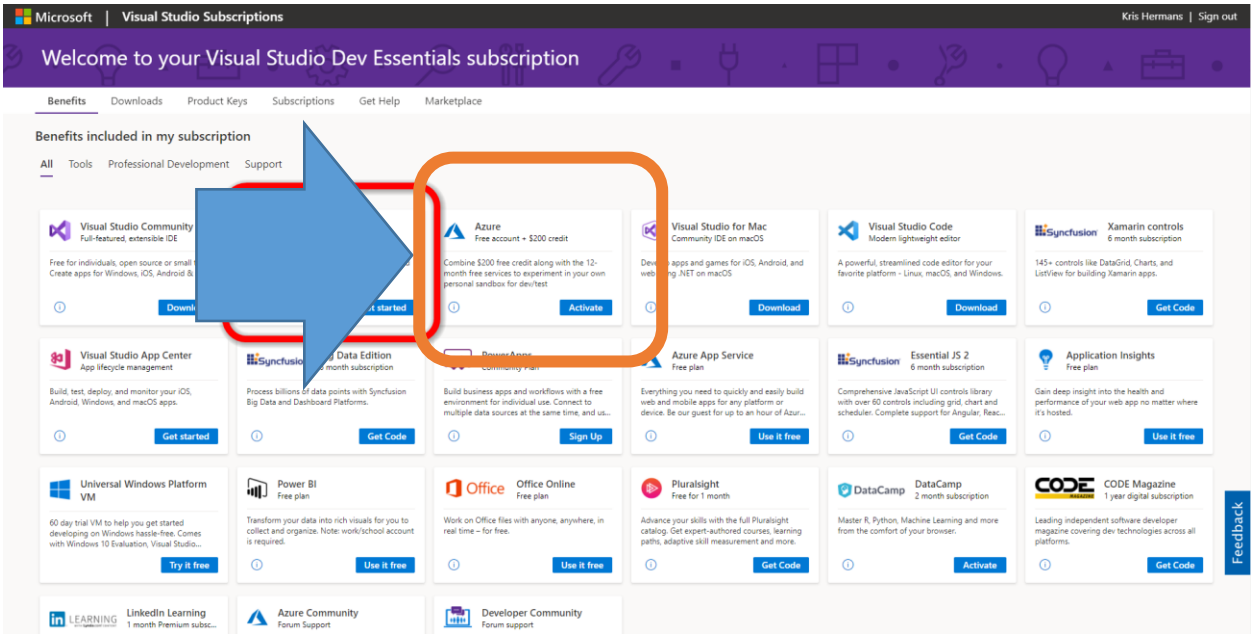


# Hosting options

When you want to deploy your work (by means of a build pipeline), you need a cloud provider to host your app. Azure provides two options.

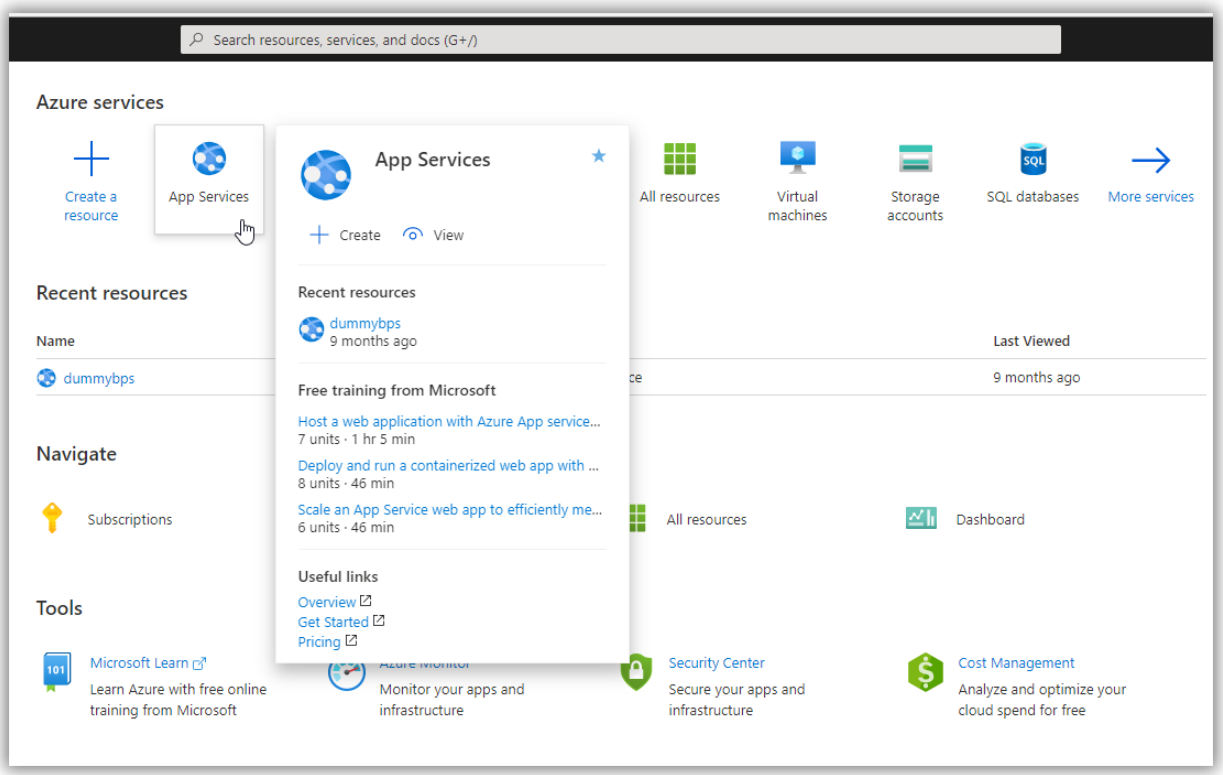
## Azure (via VS Dev Essentials)

When you registered your account for Dev Essentials, there is a possibility to activate your free Azure account for one year.




## Azure for Students

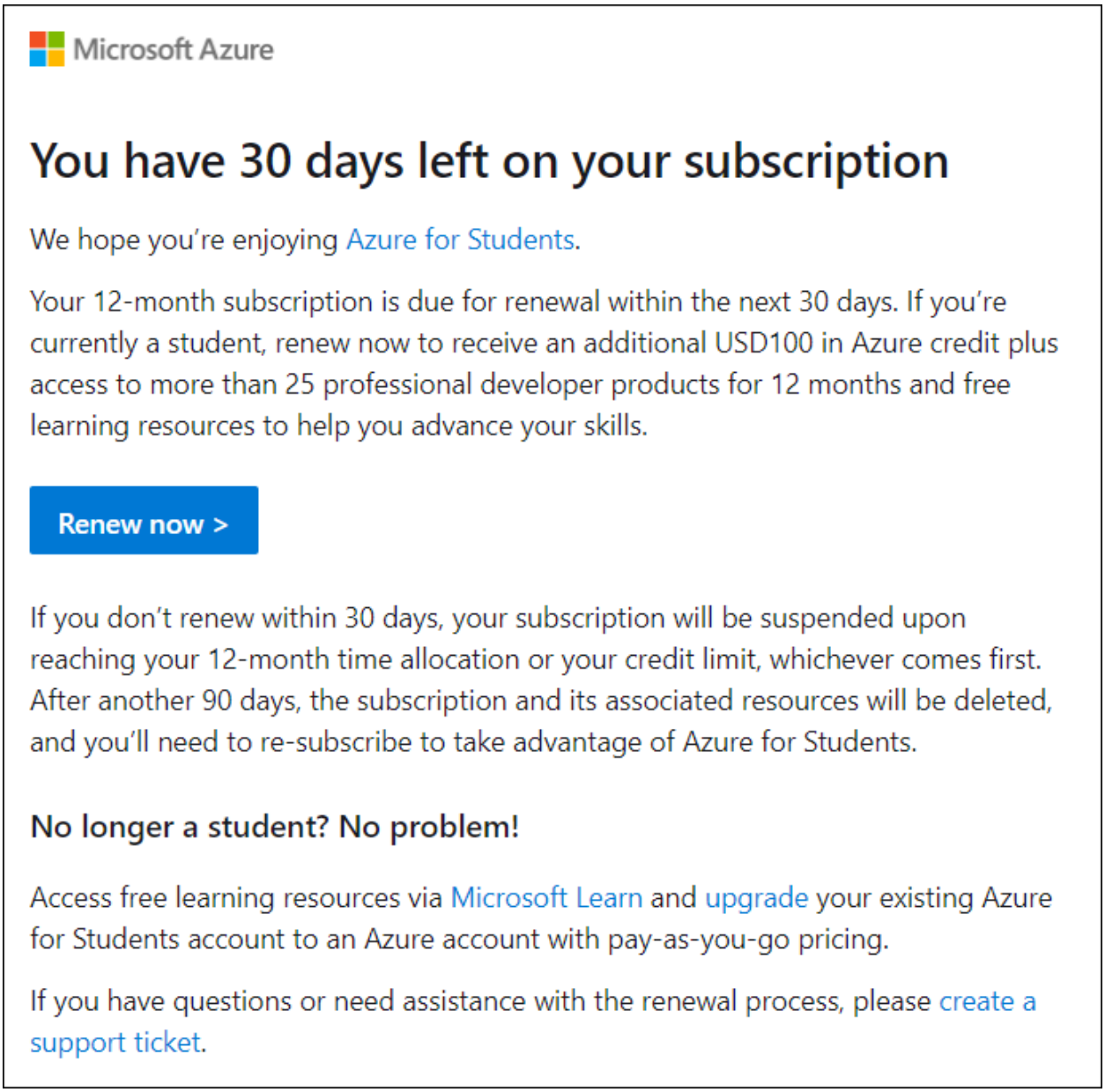
It is also possible to get one year free (limited) Azure credits with your PXL account. Log in to your [portal.azure.com](portal.azure.com) with your PXL account.




Click on “App Services” and then “Create”.

Then you will be prompted to start a free trial. Normally no credit card is required.

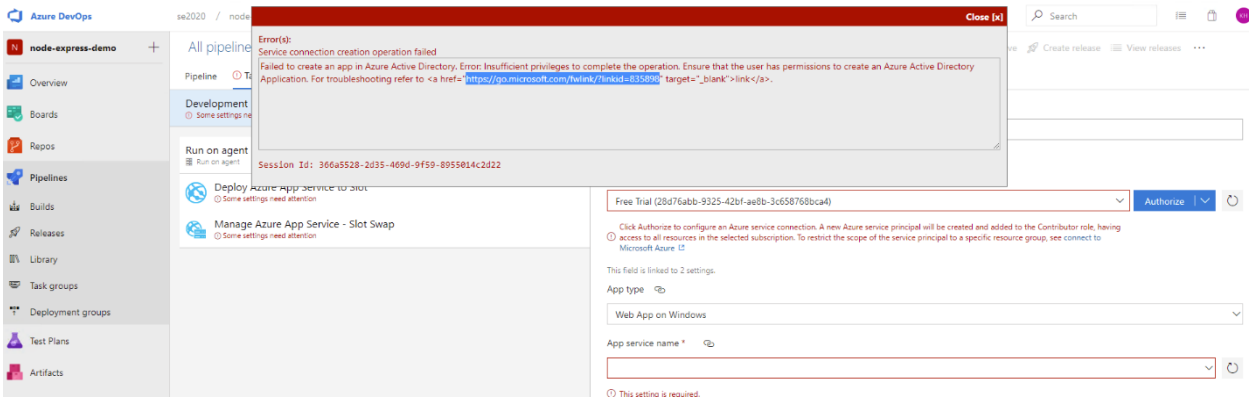
After one year, you get a reminder by mail to renew your account for another year:




The easiest way to get a web application up and running is by means of an App Service. At the moment of writing, this service stays free after the trial period (with limited use). This means you can experiment with deployment of your apps!

> An app service supports apps written in .NET, Java, Node, Python, PHP and Ruby!

When you are building a "Release Pipeline" and follow standard tutorials, you probably will run into this error message ("Failed to create an app in Azure Active Directory: Insufficient privileges"):




This is due to the fact that your release pipeline wants to register your app into the Active Directory from PXL (which is obviously not permitted). At least two workarounds exist:

## Option 1: Deploy using FTP

1. Follow this tutorial to create an app service which allows manual deployment. This guide is for .NET, but other languages are very similar
2. On the Azure Portal for the app service, look for FTP deployment settings. You will find a user/password combination to upload your files
3. This FTP task can be integrated into your workflow in Azure Devops. This article gives you inspiration.

## Option 2: Create an Azure Starter plan (credit card required)

If you are careful, you can create an Azure plan and use app services with only “free resources”. Note that you must register with a credit card and be aware for unexpected costs if you mess up with your settings.

## Option 3: Deploy to another cloud

Sign up for AWS Educate as a student and deploy to AWS.

## Option 4: on premise server

As a last resort, you could deploy (e.g. FTP) to your own private server (e. g. fixed computer at home, raspberry pi, …)