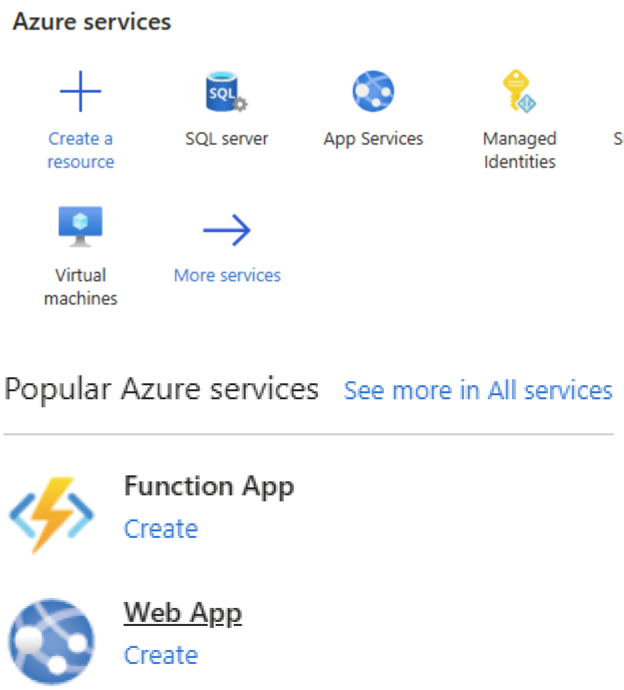
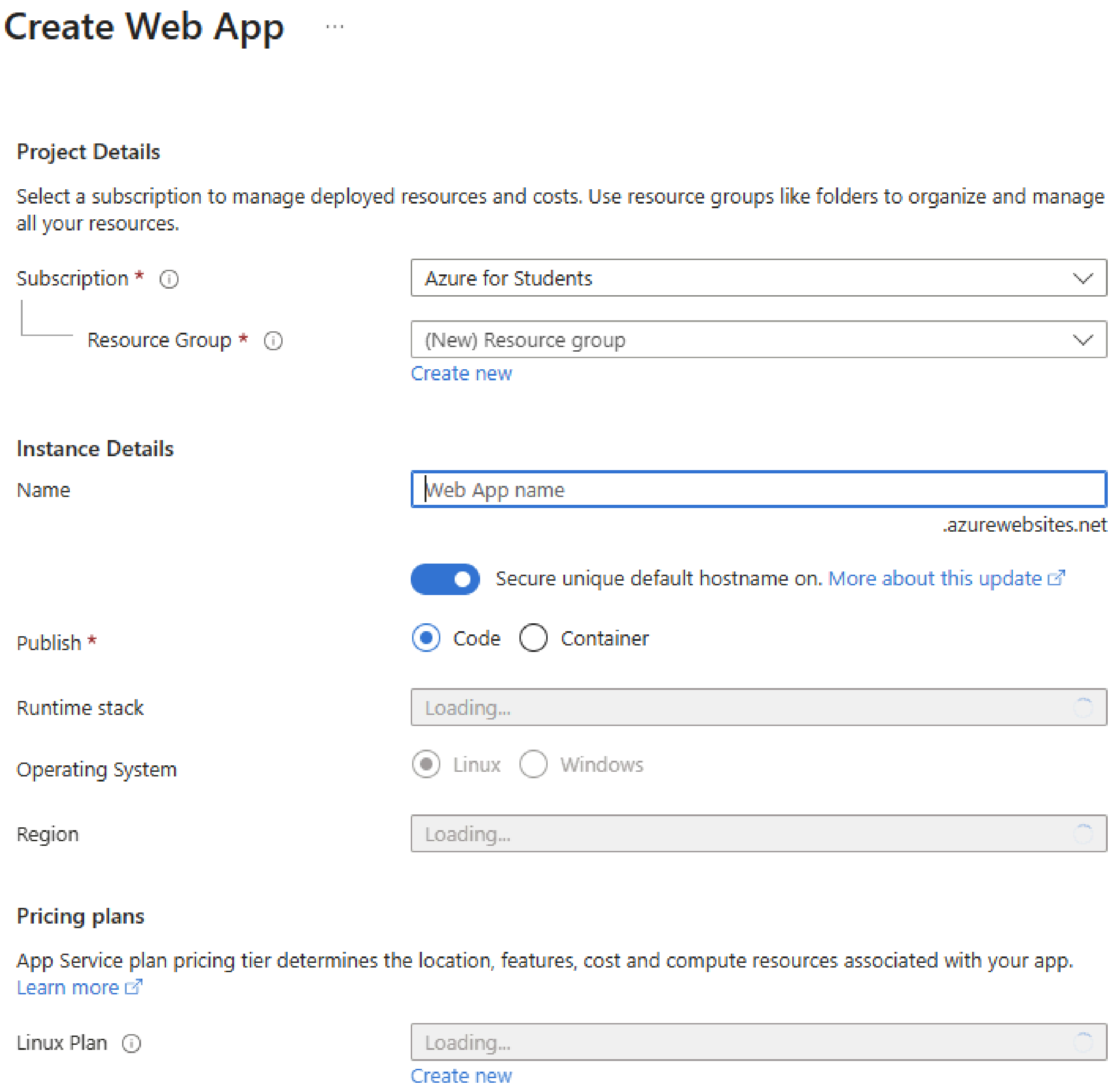
Azure Setup

1. Minna Azurei home lehele, vajadusel sisse logida.
   https://portal.azure.com/#home

2. Azure Services alt valida suurt “+” nuppu nimega “Create a Resource” ja sealt valida Web App



3. Subscription -> Azure for Students (Sinul pruugib teistsugune subscription olla, siis lihtsalt vali see teine)
   Name -> SarkPartChatter
   Publish -> Code
   Runtime Stack -> .NET9 (Vali mis versiooniga sinu projekt on tehtud.)
   OS -> Windows
   Region -> Norway East (või mis iganes on sinule lähim)



Create Web App ···

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ	Azure for Students
Resource Group * ⓘ	(New) Resource group Create new

Instance Details

Name	Web App name .azurewebsites.net
	Secure unique default hostname on. More about this update
Publish *	● Code ○ Container
Runtime stack	Loading...
Operating System	● Linux ○ Windows
Region	Loading...

Pricing plans

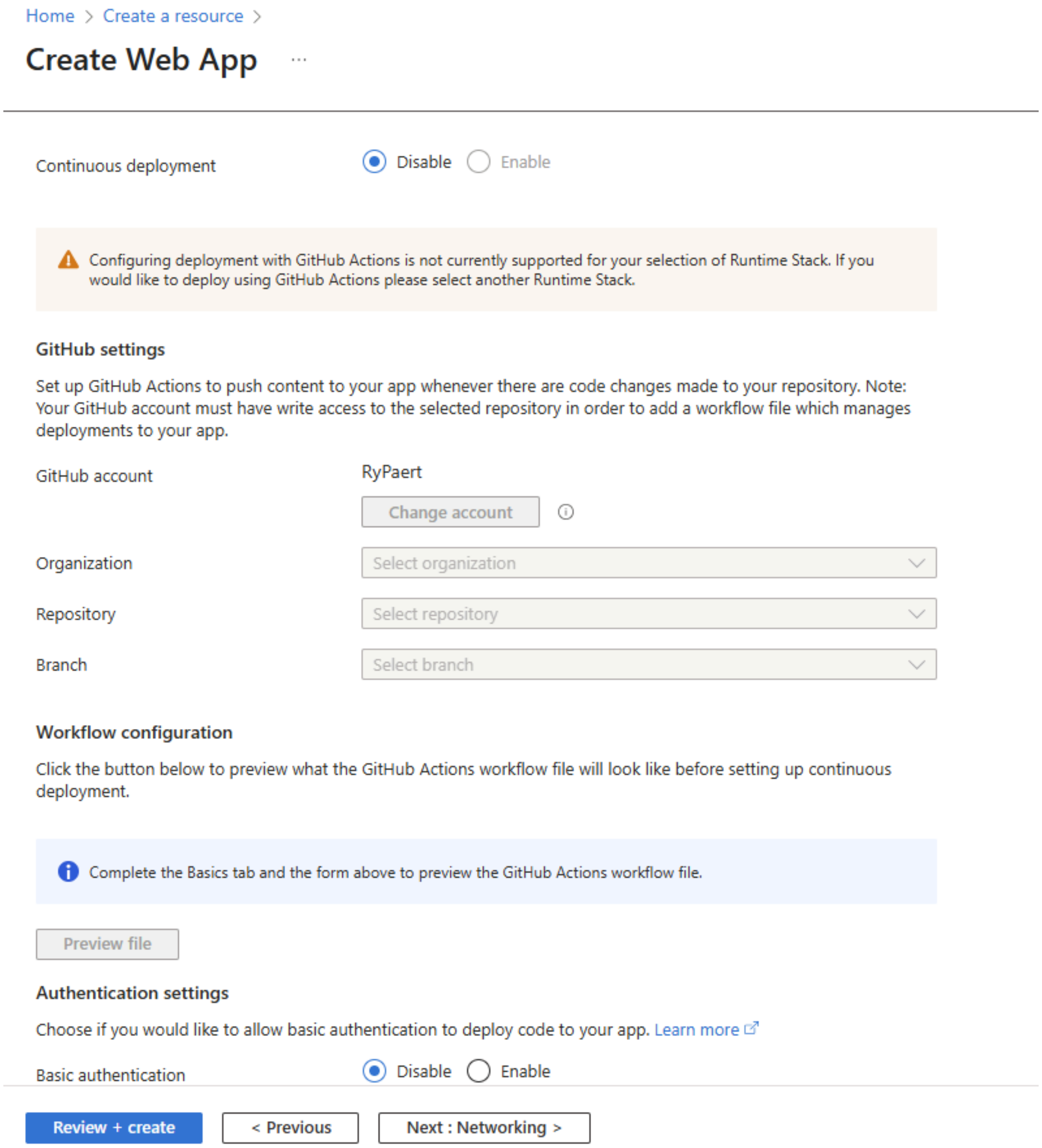
App Service plan pricing tier determines the location, features, cost and compute resources associated with your app. Learn more

Linux Plan ⓘ	Loading... Create new

4. Mine Deployment tabile.

4.1.

Enablei Continuous deployment



Home > Create a resource >

Create Web App ···

Continuous deployment ● Disable ○ Enable

⚠ Configuring deployment with GitHub Actions is not currently supported for your selection of Runtime Stack. If you would like to deploy using GitHub Actions please select another Runtime Stack.

GitHub settings

Set up GitHub Actions to push content to your app whenever there are code changes made to your repository. Note: Your GitHub account must have write access to the selected repository in order to add a workflow file which manages deployments to your app.

GitHub account	RyPaert Change account ⓘ
Organization	Select organization
Repository	Select repository
Branch	Select branch

Workflow configuration

Click the button below to preview what the GitHub Actions workflow file will look like before setting up continuous deployment.

ⓘ Complete the Basics tab and the form above to preview the GitHub Actions workflow file.

Authentication settings

Choose if you would like to allow basic authentication to deploy code to your app. Learn more

Basic authentication ● Disable ○ Enable

Github settingute alt, linki oma konto ja, siis vali missugust Repositoryt ja missugust Branchi sa avalikustada tahad. Meie j uhul olid sätted sellised.

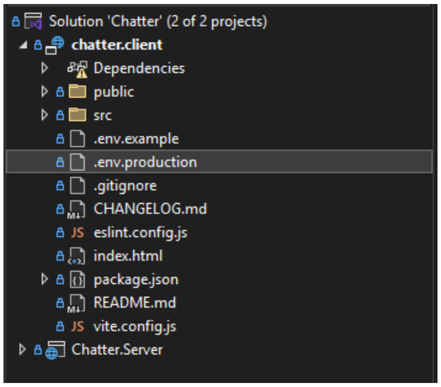
GitHub account: RyPaert

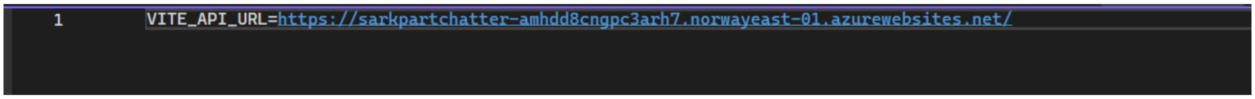
Repository: Chatter’

Branch: Main

5. Vajuta Review + Create
   5.1. Oota kuni Create nupp on vaj utatav ja, siis vaj uta seda.

6. Oma projekti frontendi luua file nimega .env.production, kirj uta sinna sisse API_ URL=sinulink ja pane sinna oma default domaini link mille sa leiad enda loodud ressurssi lehelt.

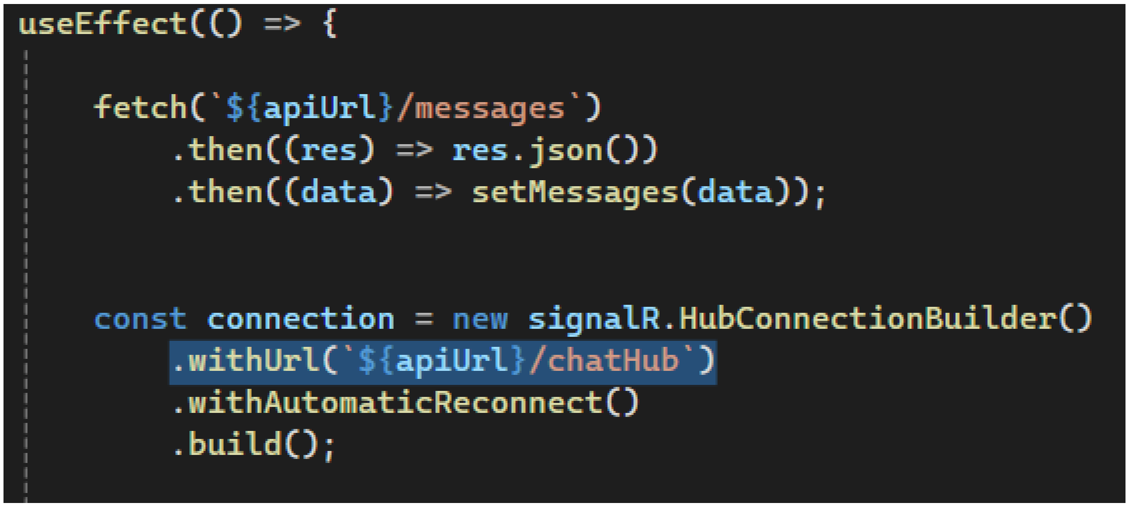




```
VITE_API_URL=https://sarkpartchatter-amhdd8cngpc3arh7.norwayeast-01.azurewebsites.net/
```

7. Meil oli enne koodi sisse hardcodeitud localhost, me muutsime seda ja hoops panime oma 6. Punktis loodud .env-i sinna.

```
const apiUrl = import.meta.env.VITE_API_ URL;
const newConnection = new signalR.HubConnectionBuilder()
    .withUrl("https://localhost:7037/chatHub") ←– vana, selle veid eemaldada
    .withUrl(`${apiUrl}/chatHub`) ←– uus
    .withAutomaticReconnect()
    .build();
```



```
useEffect(() => {

    fetch(`${apiUrl}/messages`)
        .then((res) => res.json())
        .then((data) => setMessages(data));


    const connection = new signalR.HubConnectionBuilder()
        .withUrl(`${apiUrl}/chatHub`)
        .withAutomaticReconnect()
        .build();
```

8. Tee commit ja pushi. Kõik peaks töötama.