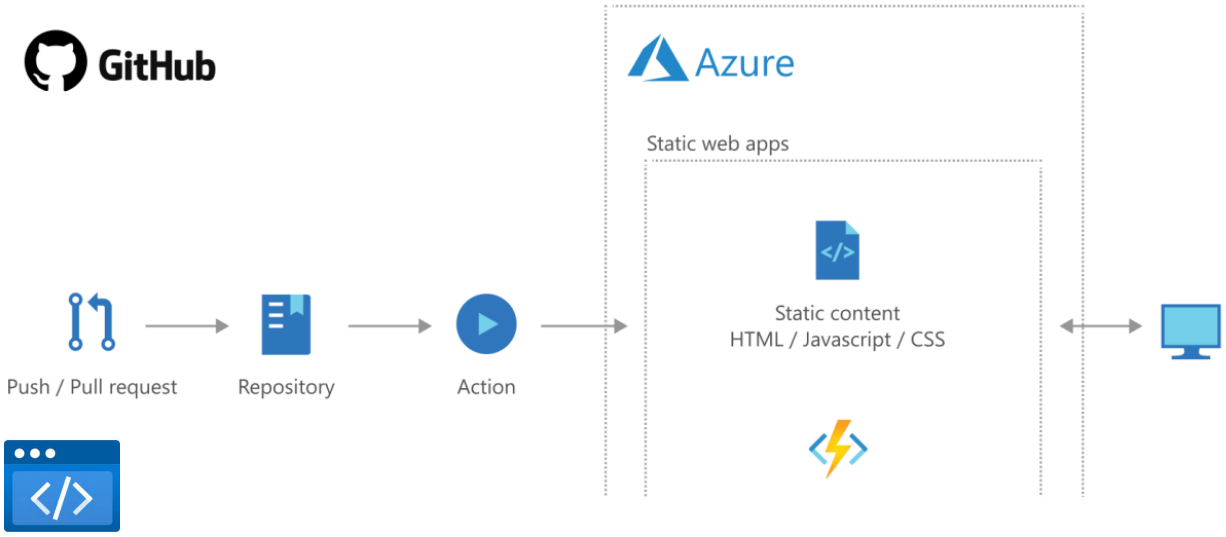



# Stories 17 : Azure Static Web App

Microsoft Azure for hosting and deploying **static** web applications. Static web apps are typically built using HTML, CSS, and JavaScript and do not rely on server-side processing. Azure Static Web Apps simplifies the deployment and hosting of these apps.

Here are some key features and steps associated with Azure Static Web Apps:

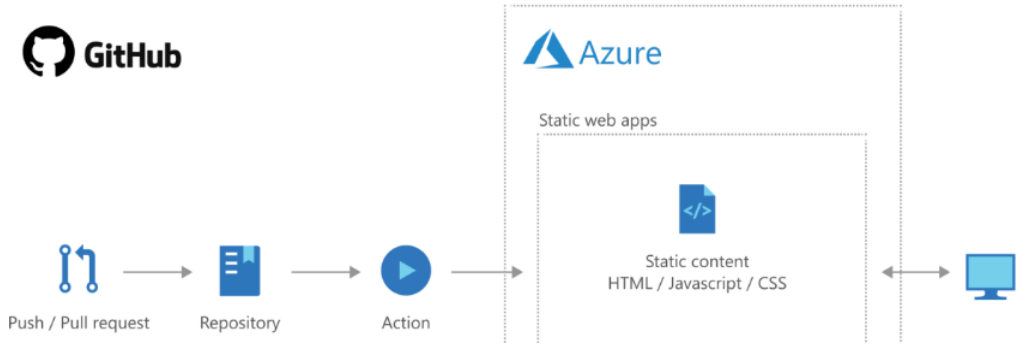
1. **GitHub Integration:** Azure Static Web Apps integrates seamlessly with GitHub repositories. You can connect your repository to Azure Static Web Apps and set up automatic deployments based on changes to your GitHub repository.
2. **Build and Deploy Workflow:** When you push changes to your connected GitHub repository, Azure Static Web Apps can automatically build and deploy your web application. It supports popular static site generators and frameworks.

3. **Serverless :** Azure Static Web Apps allows you to add serverless functions to your static web app. These functions are powered by Azure Functions and enable you to run server-side logic without managing a dedicated server.
4. **Authentication and Authorization:** The service provides built-in authentication and authorization capabilities, allowing you to secure your static web app easily.
5. **Custom Domains:** You can configure custom domains for your static web apps, allowing you to use your own domain name for the deployed application.
6. **Global Content Delivery Network (CDN):** Azure Static Web Apps leverages Azure's global CDN to ensure low-latency access to your static assets across the globe.

**Edge Location :** Edge computing allows devices in remote locations to process data at the "edge" of the network, either by the device or a local server. And when data needs to be processed in the central datacenter, only the most important data is transmitted, thereby minimizing latency.

To get started with Azure Static Web Apps:

1. **Create an Azure Static Web App:**
    - Navigate to the Azure Portal.
    - Create a new Azure Static Web App resource.
    - Connect your GitHub repository during the setup process.




2. **Configure Build and Deploy Settings:**
    - Specify the build commands and output directory for your static web app.
3. **Configure Serverless Functions (Optional):**
    - If needed, add serverless functions to your project and configure them in the Azure portal.
4. **Authentication and Authorization (Optional):**
    - Set up authentication and authorization rules as needed.
5. **Custom Domains:**
    - Configure custom domains if you want to use your own domain name.

Download website  from free css

and push to you git hub account

commands

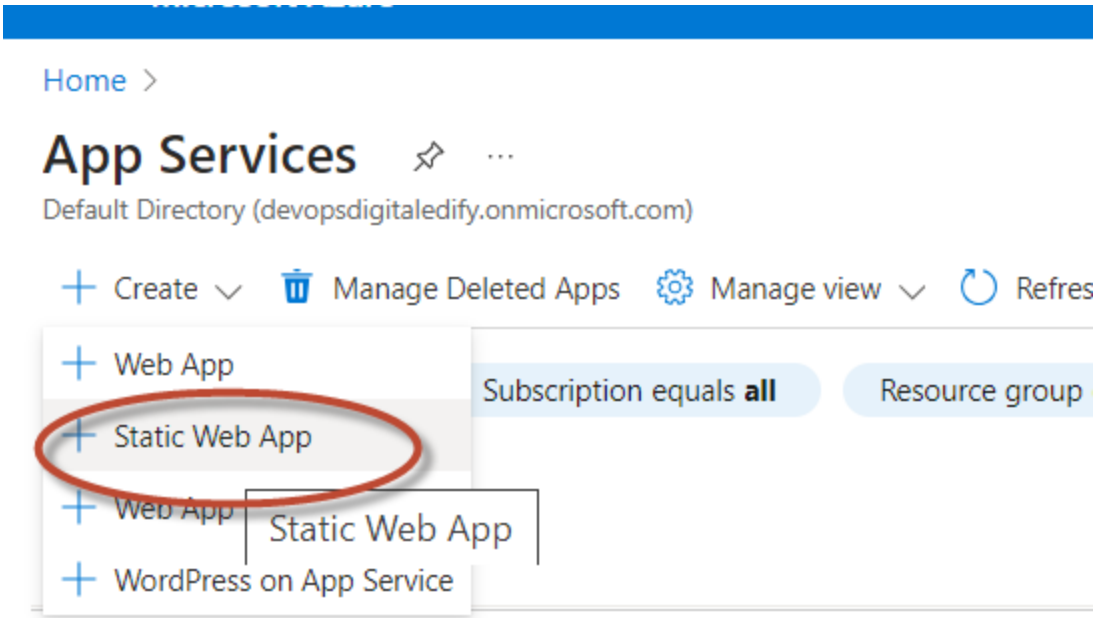
```
git init
git remote -v
git remote add origin https://github.com/mubeen507/finexo.git
git remote -v
git status
git add .
git commit -m " added finexo website"
git status
git branch -M main
git  push origin main
```

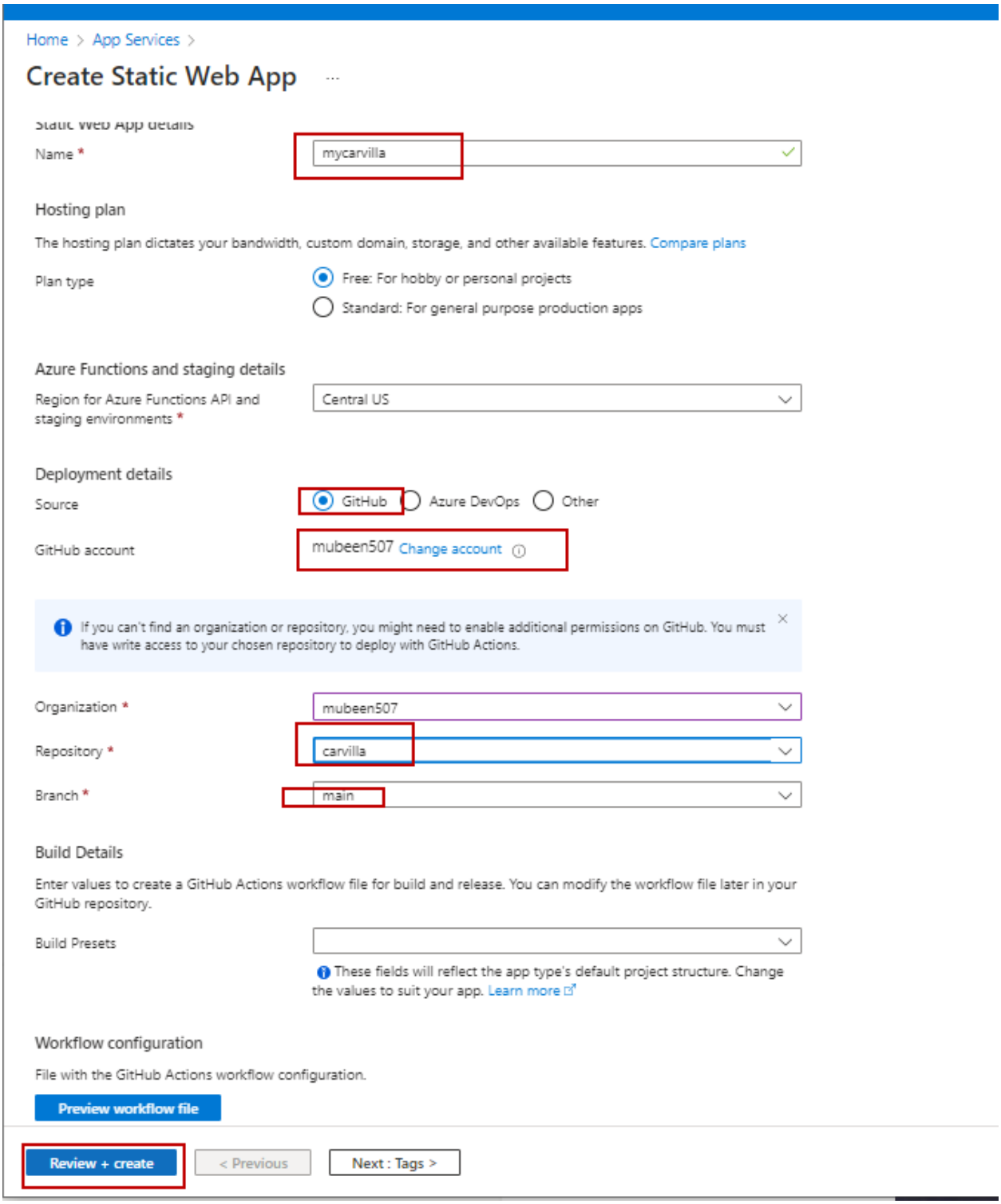
You frok

https://github.com/mubeen507/finexo.git

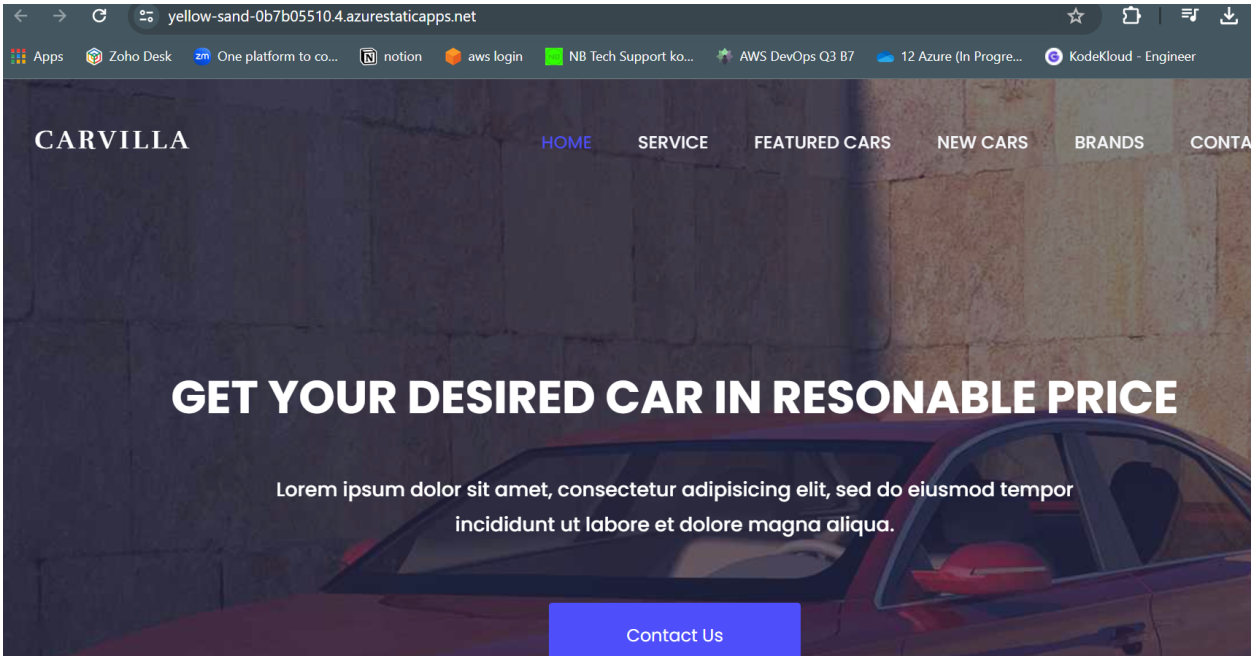
`Lab : Deploy a any static web site using azure app services`

## Step 1 : create  App service  select static web app


Home
App Services
Default Directory (devopsdigitaledify.onmicrosoft.com)
Create
Manage Deleted Apps
Manage view
Refres
Web App
Static Web App
Web App
WordPress on App Service
Subscription equals all
Resource group
Static Web App




once deployment is completed access the url


yellow-sand-0b7b05510.4.azurestaticapps.net
Apps
Zoho Desk
One platform to co...
notion
aws login
NB Tech Support ko...
AWS DevOps Q3 B7
12 Azure (In Progre...
KodeKloud - Engineer
CARVILLA
HOME
SERVICE
FEATURED CARS
NEW CARS
BRANDS
CONTA
GET YOUR DESIRED CAR IN RESONABLE PRICE
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Contact Us