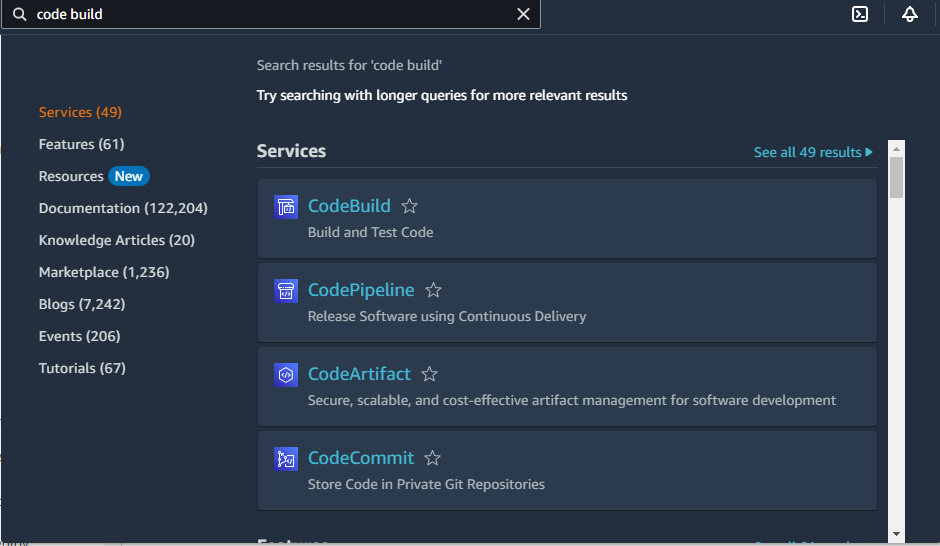
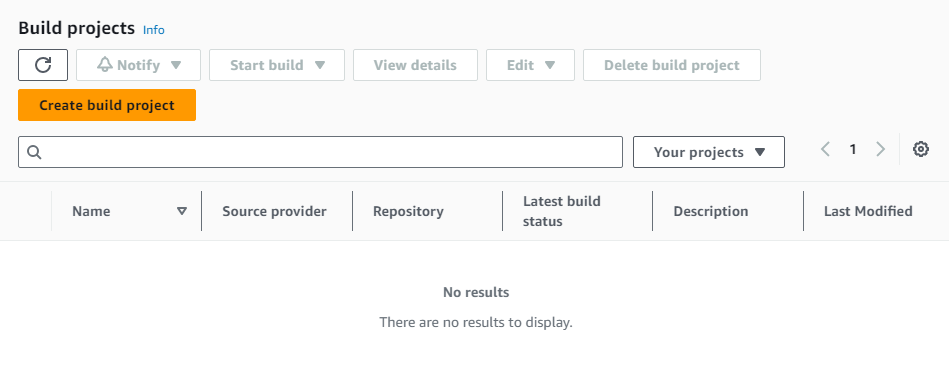
**AWS CODE-BUILD**

**STEP 1:** Search for **CodeBuild** on your AWS console search bar



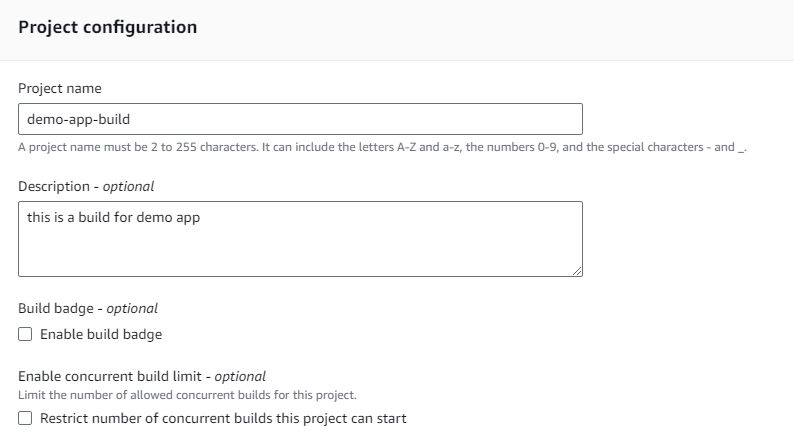
**STEP 2**: Click on **Create Build Project**

Click here



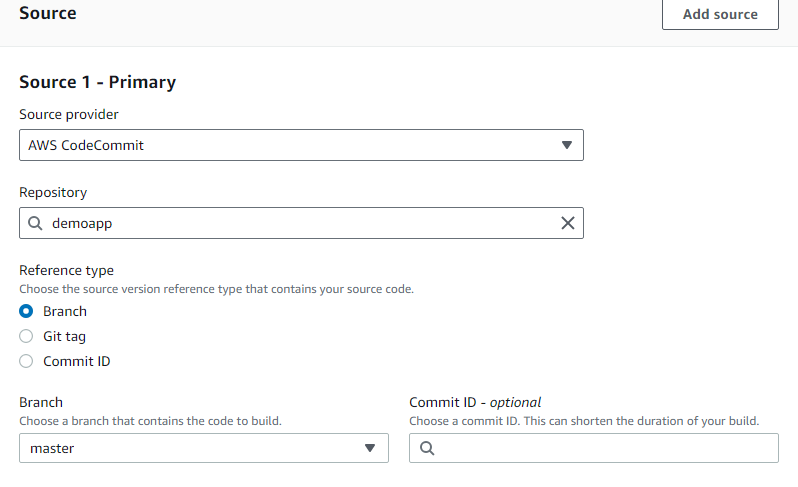
**STEP 3**: Give a **name** for your build project and provide suitable **description**

(The project name I have given here is “demo-app-build” and the description is “this is a build for demo app”)

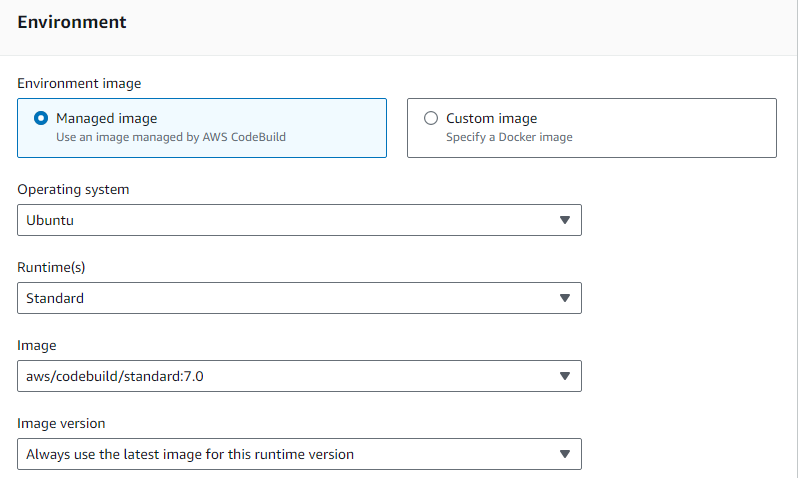


**STEP 4**: Mention the **source** i.e, from where do you want to pull your source code. Various providers are GitHub, BitBucket, AWS CodeCommit, GitHub Enterprise and even S3. Also specify the repository and branch from which you want to pull your code.

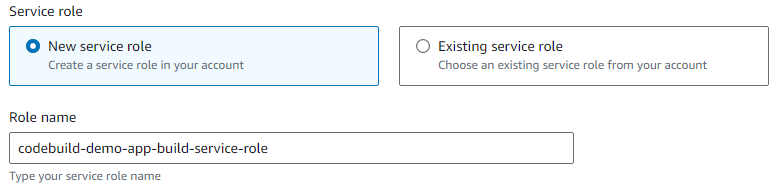
(Here I have maintained my source code in AWS “CodeCommit” with repository name “demoapp” and “master” branch)



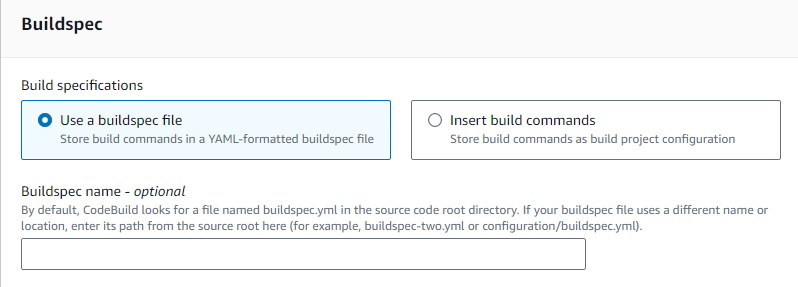
**STEP 5**: In the **environment** section you’ve to specify the operating system, runtime standard, image, image version and you will have the option to use your own custom docker image or image maintained by the AWS CodeBuild



**STEP 6**: Create a **service role** for your build. This is required because your code build may require access to any other aws service ( for example: EC2, CodePipeline, CodeDeploy). Provide a name for the service role you want to create.

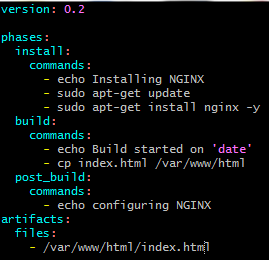


**STEP 7: BuildSpec** – This section is pretty important. BuildSpec is a specification file which contains all the configuration details of various phases that are involved in building of an application package. The default BuildSpec file name is buildspec.yml.

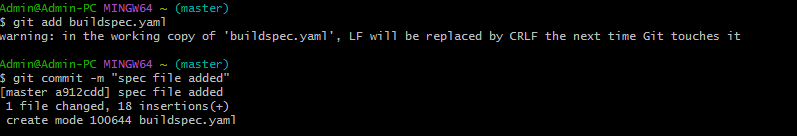


NOTE: No need to mention the buildspec file name if you are using default name.

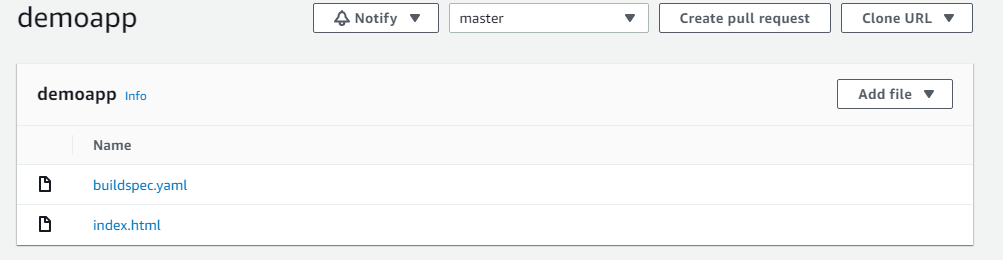
**STEP 8:** Create a BuildSpec file and involve the steps required for building your project. Below is the sample BuildSpec file used for this explaination.



**STEP 9:** Save the BuildSpec file. Your BuildSpec file will be present in the working area. Move it to the staging area and then to the local working repository. Finally push the file to your AWS CodeCommit repository using its URL.

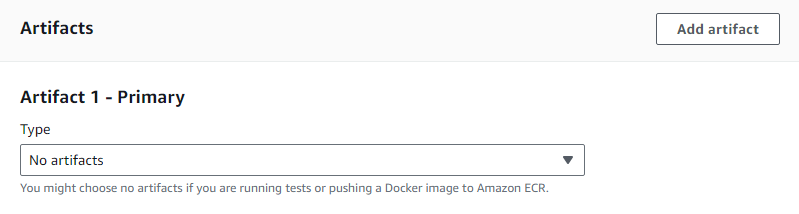




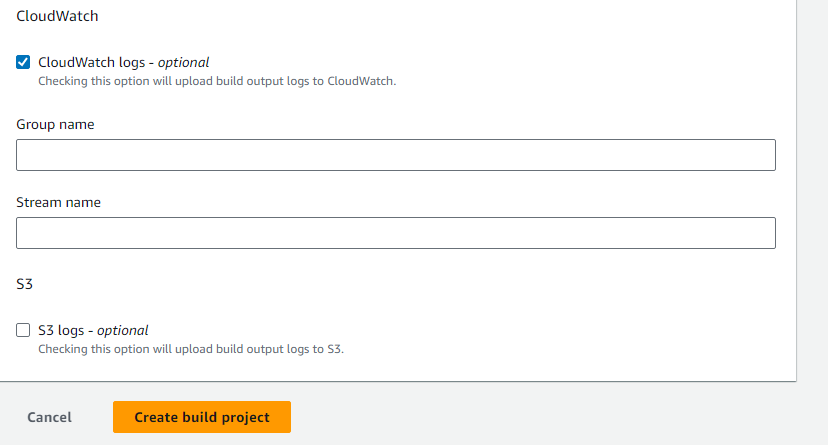


buildspec.yaml file pushed to my “demoapp” codecommit repository

**STEP 10: Artifacts** – This stage allows you to store your build artifacts in the AWS S3 buckets or you can choose not to store any artifacts.



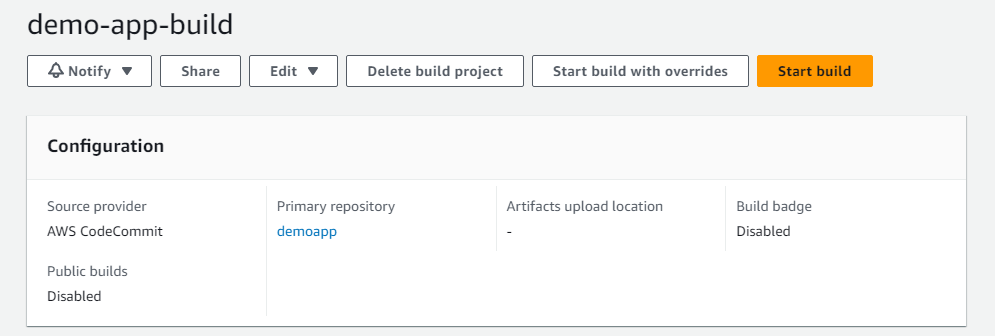
**STEP 11**: You can enable CloudWatch logs for you CodeBuild (Optional). Click on create Build Project.



Click Here

**STEP 12**: Once yov’ve created build project, you’ll be moved to the build projects page. Select the project that you’ve created and click on the start build option available on the top right corner.

Click Here

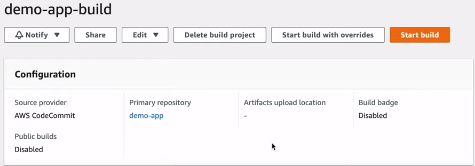


**STEP 13**: Once you start the build, you can the check the Phase details. Phase details indicates whether the steps included in buildspec.yaml is executed successfully or has failed. If all the phases are successfully completed, your application package will ready.

Click Here







NOTE: Code Build stage is successfully completed and can be moved to next stage i.e, Code Depoly