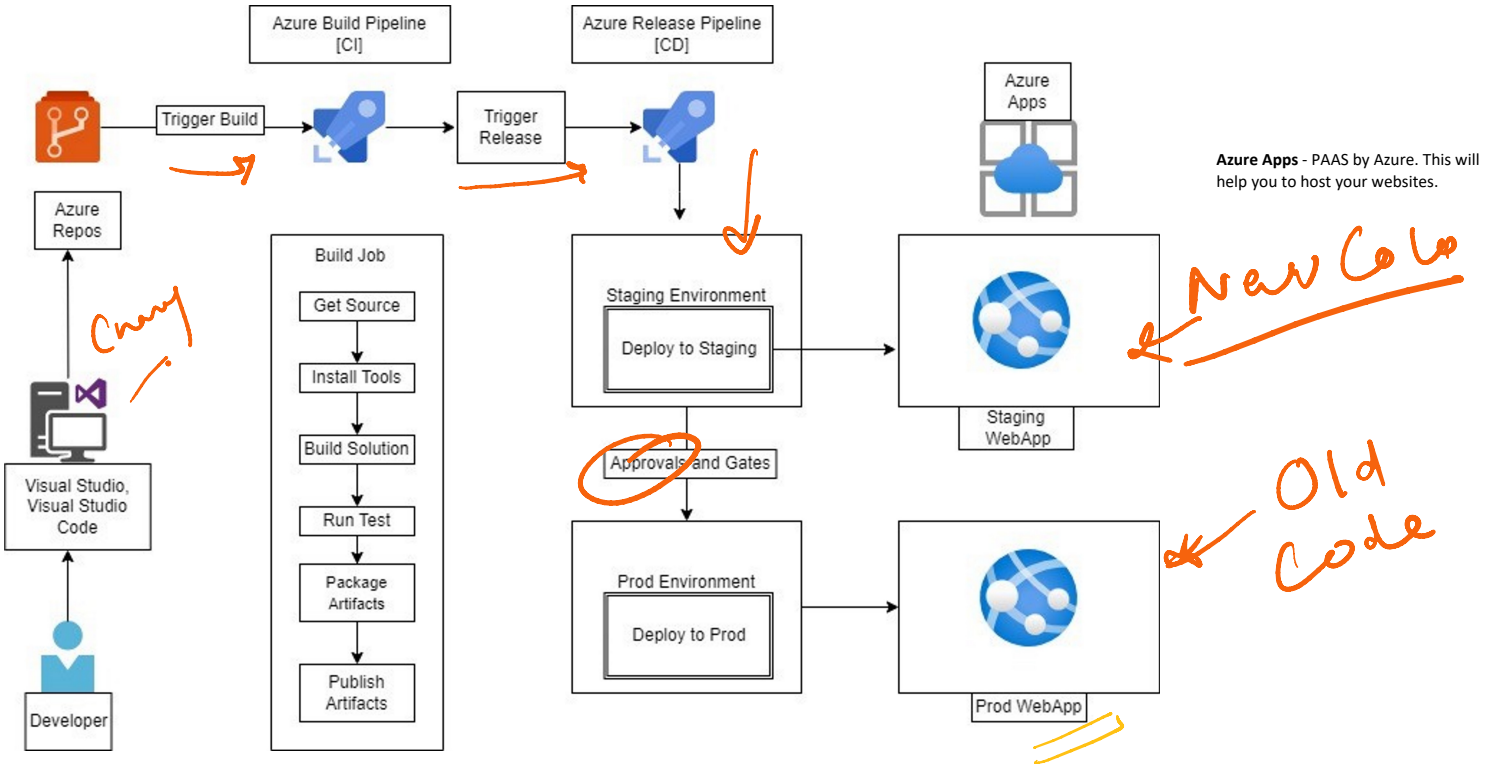
Day 5 Continuous Deployment + Containers

DEMO THAT IS BEING IMPLEMENTED



Azure Apps - PAAS by Azure. This will help you to host your websites.

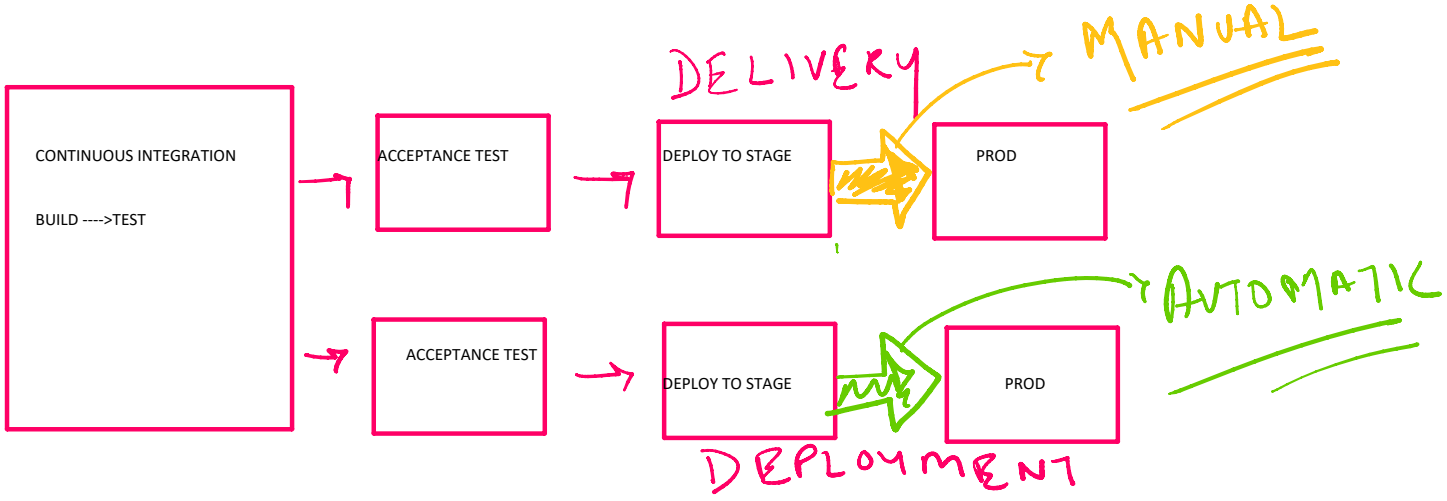
CD - Continuous Deployment/Delivery

Continuous Delivery

- It automatically deploys all code changes to a stage/test/prod environment after the build stage .
- This means that on top of automated testing you have an automated release process.
- There is human intervention.

Continuous Deployment

- There is NO human intervention



DEMO PART 1- Create a App in Local Env

- Use Visual Studio 2022.
- Create a Basic .NET Application.
- BUILD and then DEPLOY the app locally.

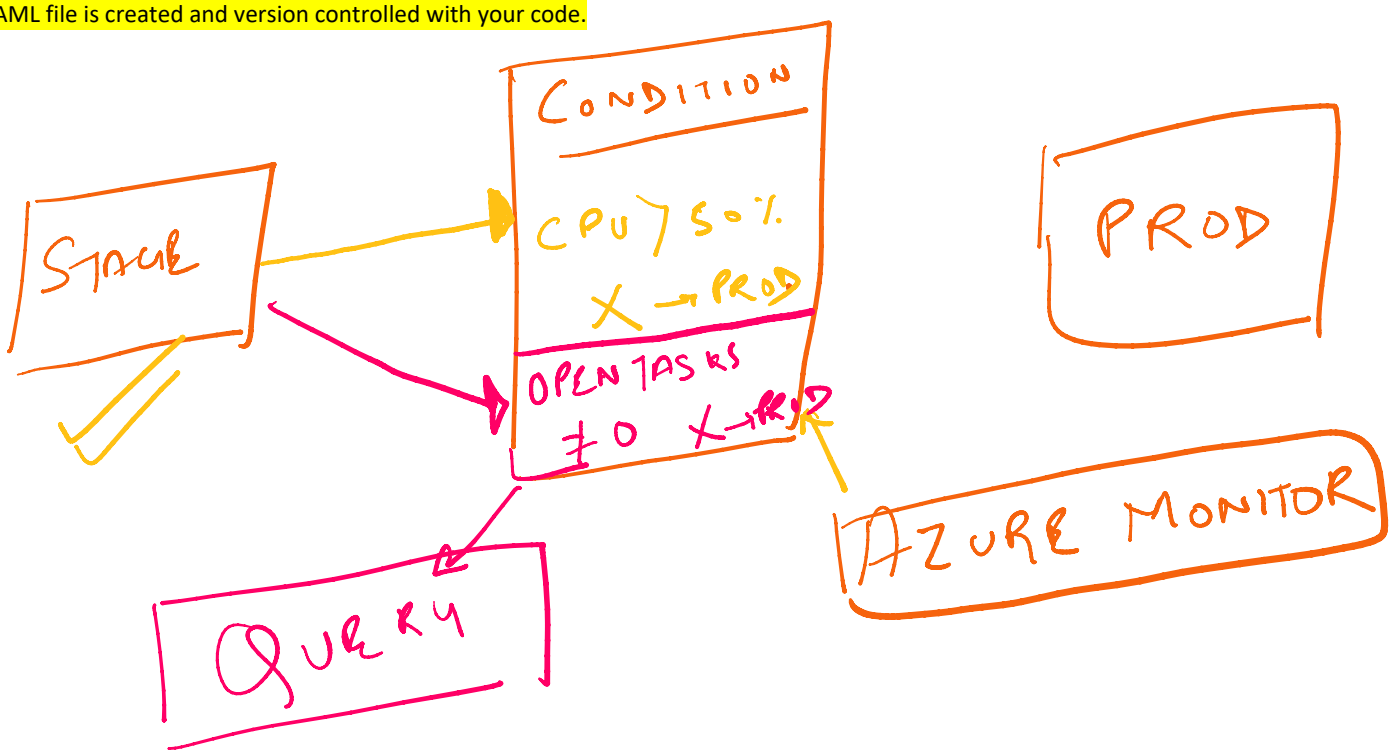
- VSCODE -> Used the existing solutions offered by VS Code to build a solution on .NET -> Gave the name for my app and selected the right framework [.Net 8.0] -> I had a solution file

DEMO PART 2 - Push the Code to Azure Repos

- Next we are going to integrate Azure repos as my remote repository and push our code to Azure Repos.

DEMO PART 3 - Start The CI in Azure Pipeline

- Selected the right repo + right project.
- Azure Pipeline created a YAML for us.
- YAML file is created and version controlled with your code.



DEPLOYMENT GROUP

- Collection of servers that for a logical unit.
- They have similar traits. [Same apps + dependencies installed]

- Create a VM on you on prem/any public cloud/if you want you can use Azure too for testing.
- Install your app.
- Install your dependencies.
- Update the patches [Security + windows]
- Run the POWERSHELL SCRIPT given to you by Azure Devops.
- Your server will be registered and visible in the Azure portal.

TASK Groups

- Same tasks like build, test and push you need to do in a different pipeline.
- Task Groups.
- Redudant build.

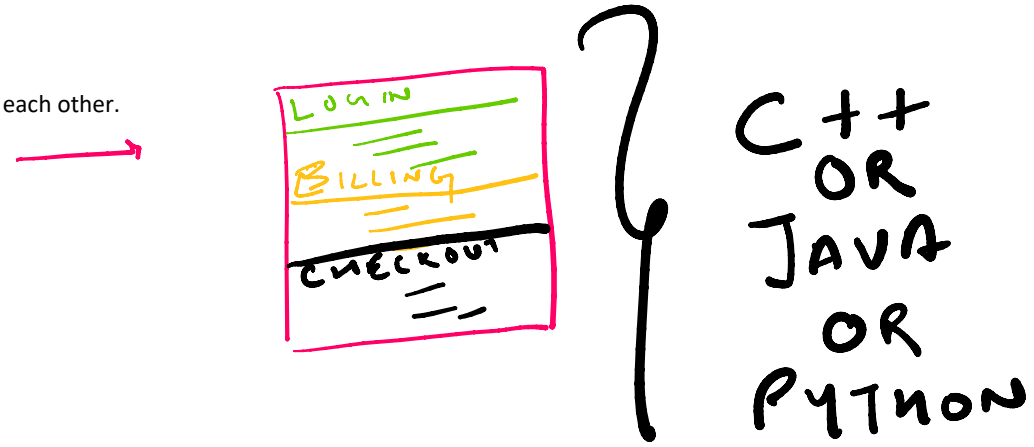
Library

Create set of variables that you can share across multiple pipelines.

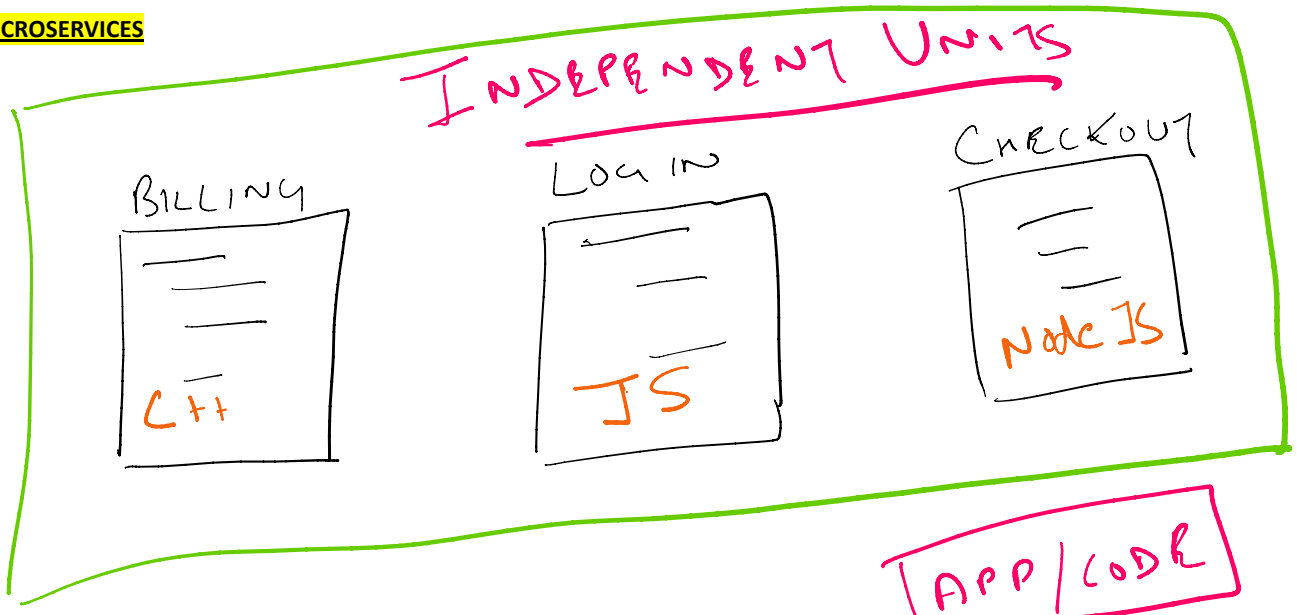
BUILD CONTAINER WITH AZURE DEVOPS

- Container
  - Move your code quickly from one env to another.
  - Address scalability issues.

Monolithic Services
- These codes are tightly coupled with each other.
- Single deployable unit.

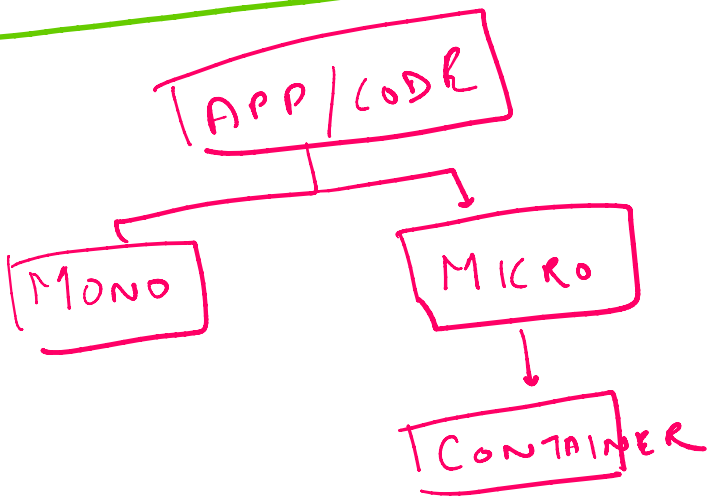


MICROSERVICES



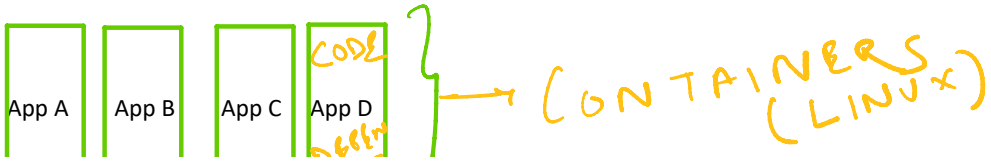
Benefits
- Independent Components
- Easier understanding
- Better Scalability
- Flexible in choosing the underlying tech

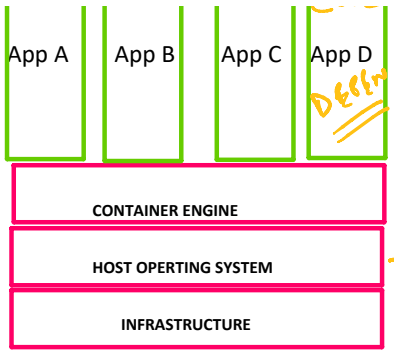
Drawback
- Added layer of complexity +Enablement for engineers



CONTAINERS

- Container is a unit of software that packages up our code and all its dependencies so that the application runs quickly and realibly.
- Independent about the environment.





CONTAINERS ARE SIMPLY PACKAGE OF SOFTWARE AND IT DEPENDENCIES