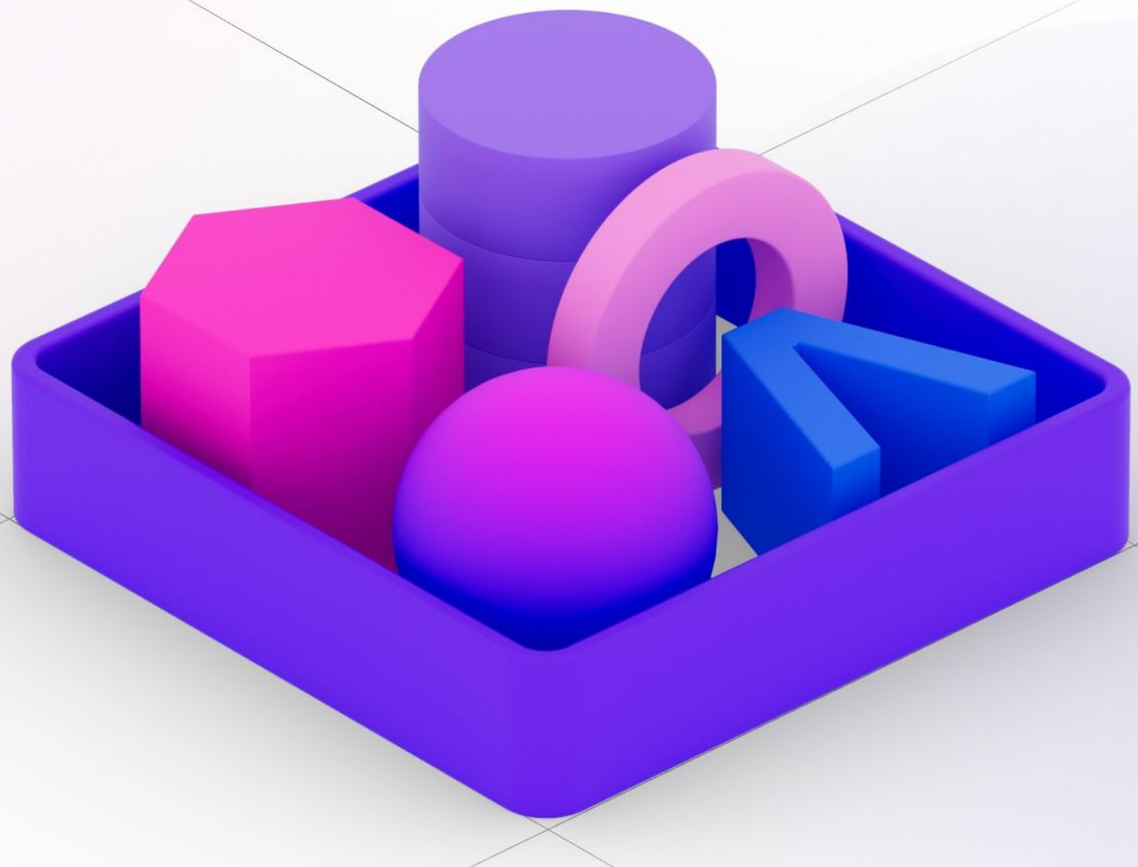
.NET Conf



Build .NET MAUI Apps with DevOps

David Ortinau, *Target of jokes by Sweeky*

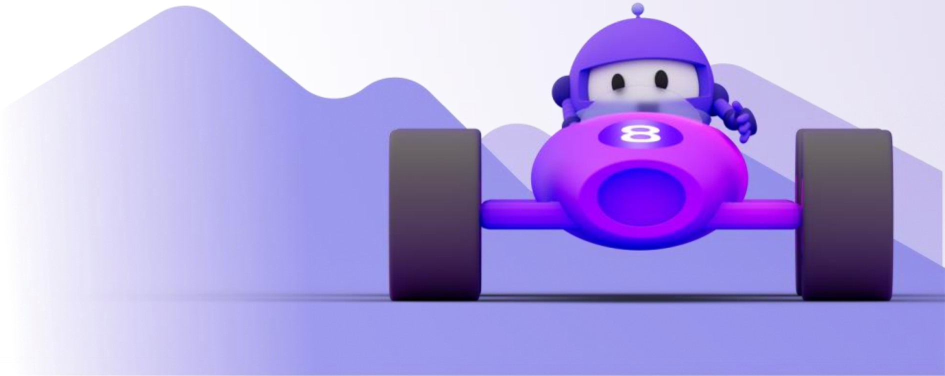
Sweeky Satpathy, *Makes fun of David*

Build .NET MAUI Apps with DevOps

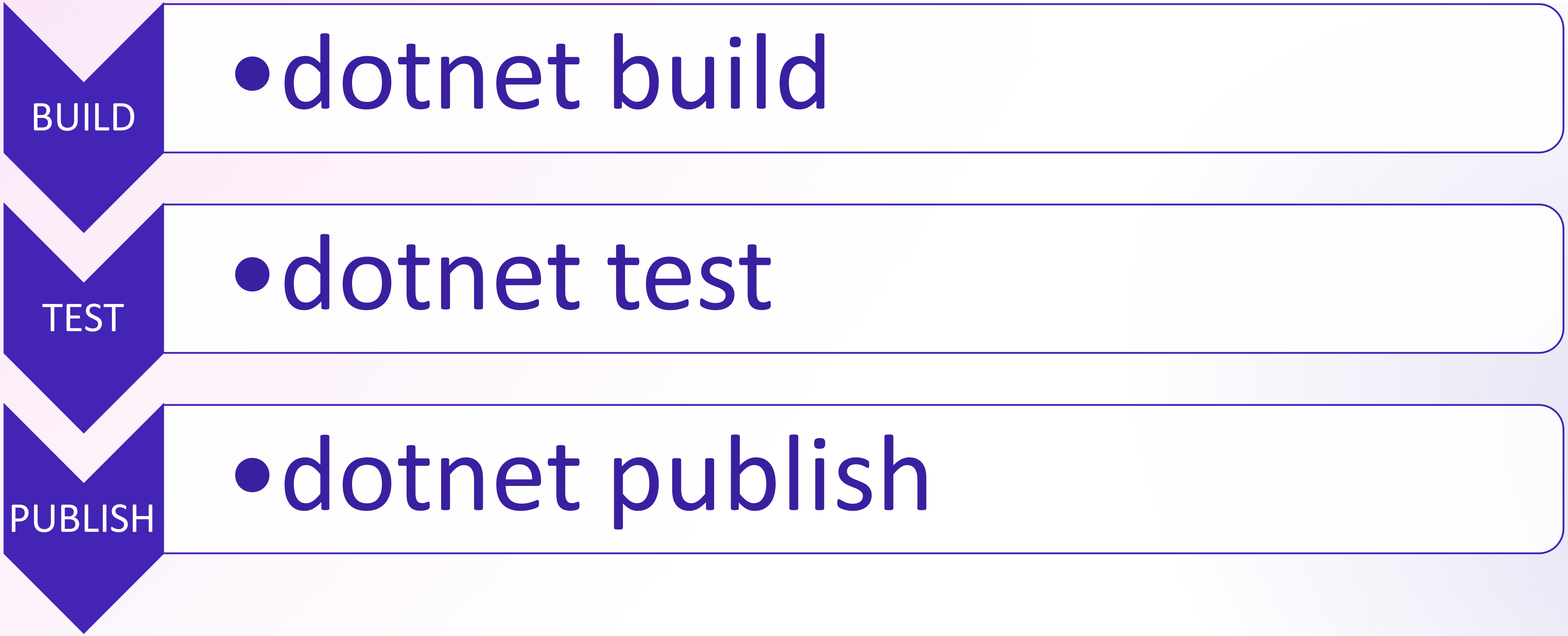
David Ortinau, *Principal Product Manager*

Sweeky Satpathy, *Principal Software Engineer*

A .NET Dev in a DevOps World



dotnet CLI 🤝 DevOps Pipeline



BUILD
- dotnet build

TEST
- dotnet test

PUBLISH
- dotnet publish

dotnet CLI 🤝 DevOps Pipeline

BUILD

- dotnet build

TEST

- dotnet test

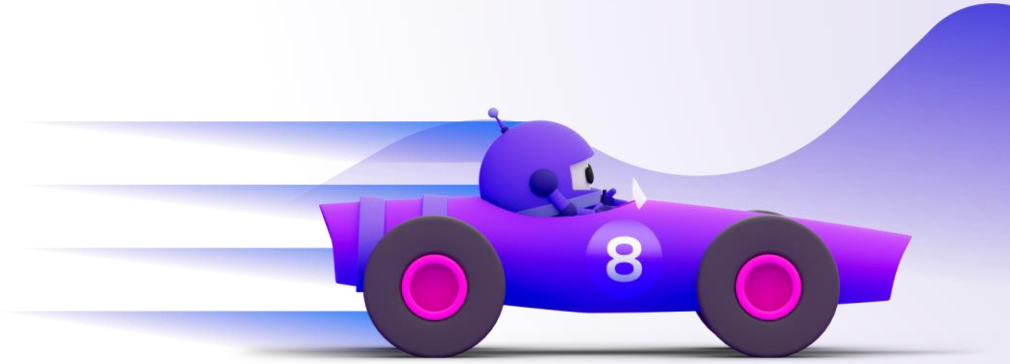
PUBLISH

- dotnet publish

MOBILE

- use published artifacts in App Store/Play Store

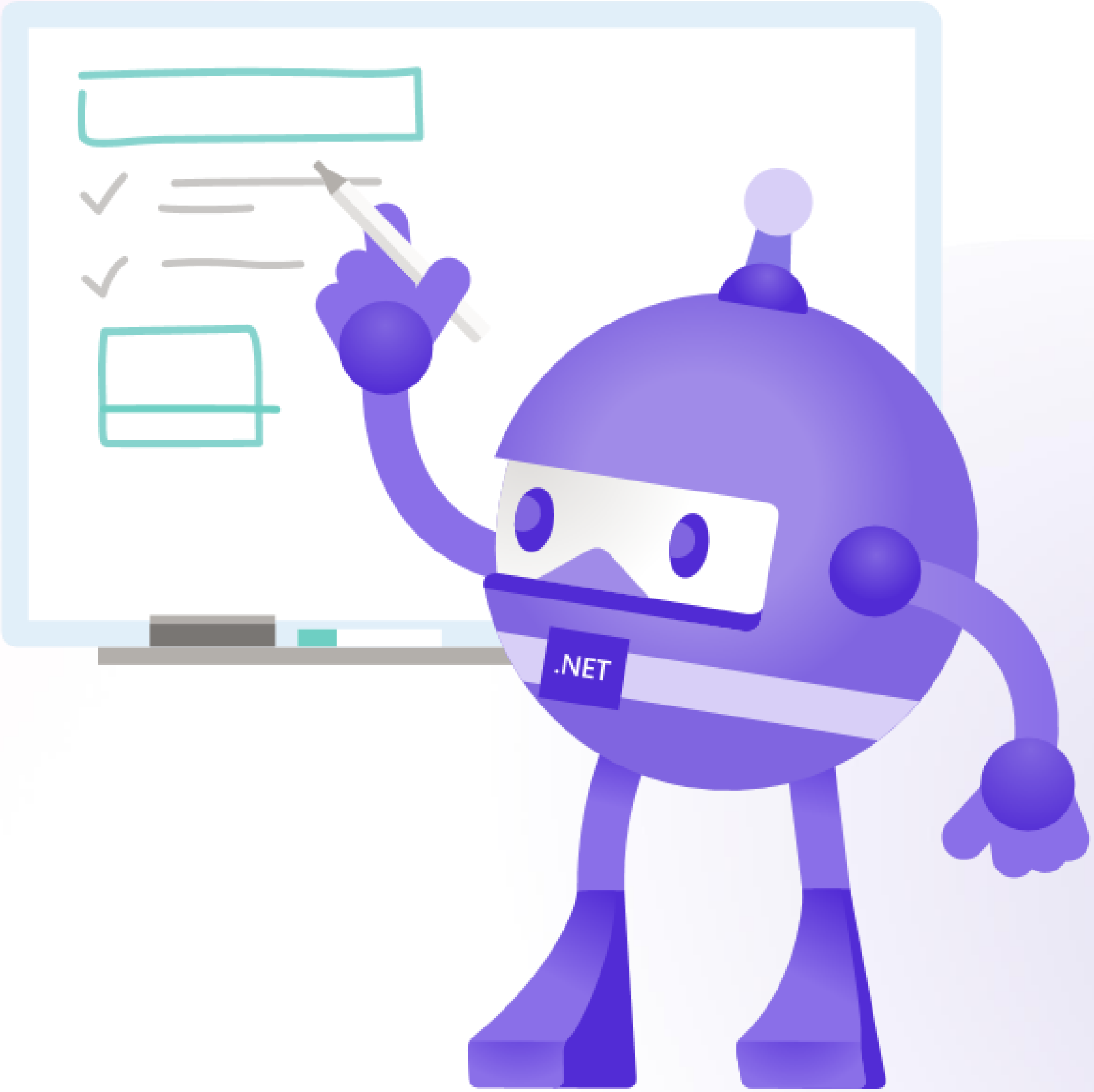
Let’s write some YAML, or not…?



Environment Setup

- Installed Software on GitHub Actions VM : Available Environments
- Installed Software on Azure DevOps VM : Available Environments
- Self Hosting
  - Download .NET 8
  - Install .NET MAUI Workloads

  dotnet workload install maui



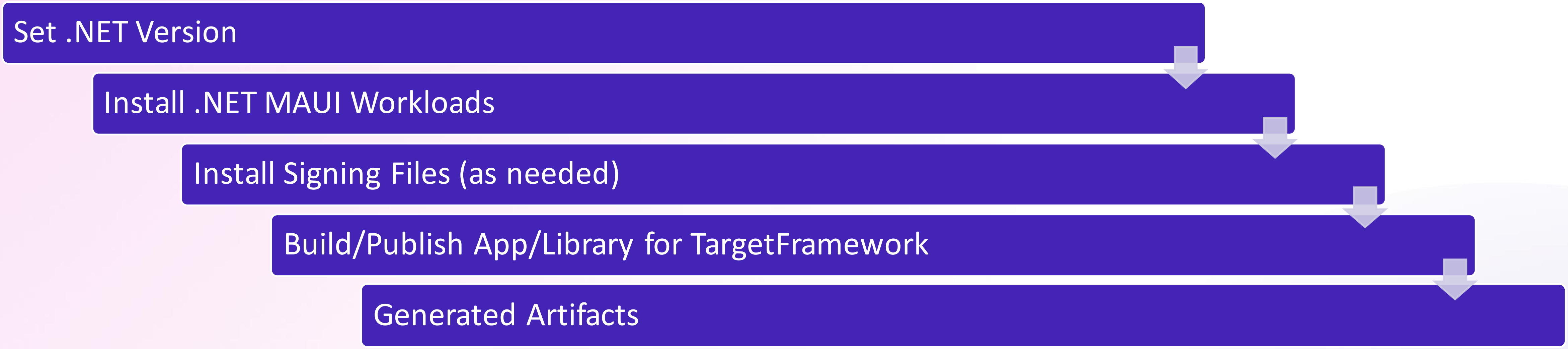
Secret Files

- Guide Apple Signing
- Guide Android Signing
- Guide Windows Signing

- GitHub Actions requires Encrypted secrets.
- Apple Actions Tasks for Code Sign Cert and Provisioning

Let’s Write some YAML

One Flow to Rule them All



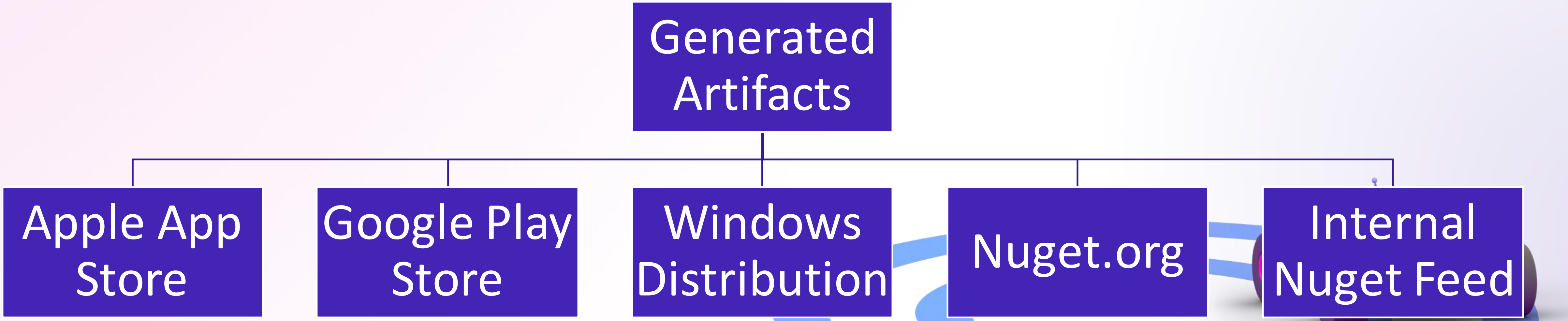
Set .NET Version

Install .NET MAUI Workloads

Install Signing Files (as needed)

Build/Publish App/Library for TargetFramework

Generated Artifacts



Generated Artifacts

Apple App Store

Google Play Store

Windows Distribution

Nuget.org

Internal Nuget Feed

Demo

.NET MAUI Class Library Sample

Demo

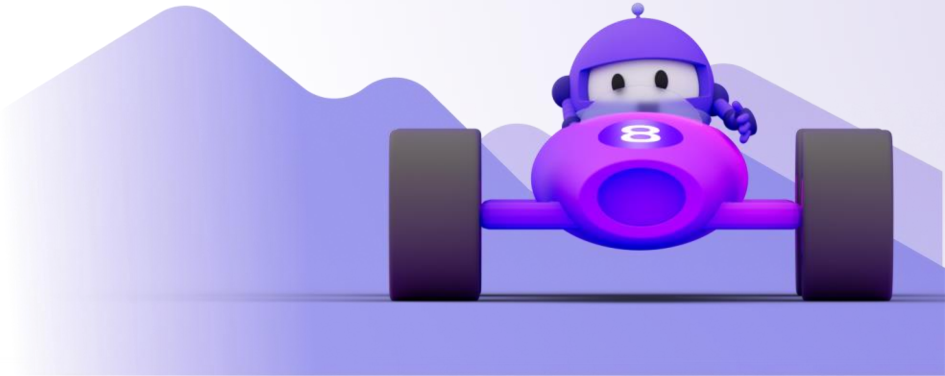
.NET MAUI App Sample (on Windows Agent)

Demo

.NET MAUI Blazor Hybrid App Sample (on Mac Agent)

Let the world enjoy my apps!

Distribute it!



iOS Distribute to Test Flight

GitHub Actions

-
  ```
  - task : App Store upload
     run : |
       xcrun altool --upload-app --type ios –file "/Path/To/Your/Package/your_package_file.ipa" --username "<user_name>" --password "<app_specific_password>”
  ```
- Or FASTLANE! ☺

Azure DevOps

-
  ```
  - task: AppStoreRelease@1
    inputs:
      releaseTrack: 'TestFlight'
      appIdentifier: 'com.maui.TestApp1'
      appType: 'iOS'
      appSpecificId: '<ValueFromAppStoreConnect>'
  ```

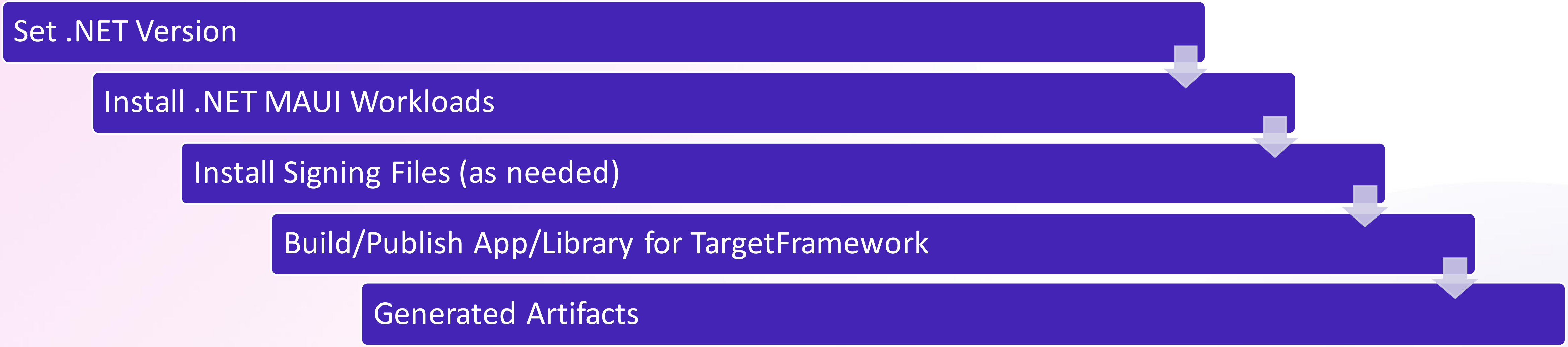
Google Play Distribution

GitHub Actions

```
- name: Build & deploy Android release
  run: bundle exec fastlane android deploy
  env:
    KEYSTORE_FILE: ${{
steps.android_keystore.outputs.filePath }}
    KEYSTORE_PASSWORD: ${{
secrets.KEYSTORE_PASSWORD }}
    KEY_ALIAS: ${{ secrets.KEY_ALIAS}}
    KEY_PASSWORD: ${{ secrets.KEY_PASSWORD
}}
    ANDROID_JSON_KEY_FILE: ${{
steps.service_account_json_file.outputs.fi
lePath }}
```

Azure DevOps

```
- task: GooglePlayRelease@4
  inputs:
    applicationId:
'com.maui.TestApp1’
  action: 'SingleBundle’
  bundleFile: '<pathtoAAB>’
  track: 'internal'
```



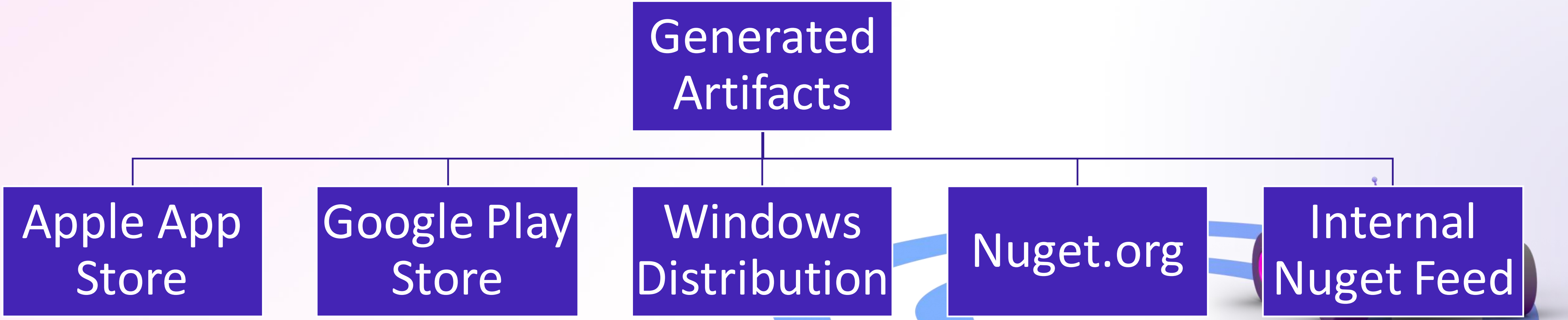
Set .NET Version

Install .NET MAUI Workloads

Install Signing Files (as needed)

Build/Publish App/Library for TargetFramework

Generated Artifacts



Generated Artifacts

- Apple App Store
- Google Play Store
- Windows Distribution
- Nuget.org
- Internal Nuget Feed

Thanks for Listening!

Resources

- Download .NET 8 aka.ms/get-dotnet-8
- Getting Started with DevOps and .NET MAUI
- GitHub Actions Task Test Reporter
- Azure DevOps Task Publish Test Results
- Azure DevOps Task Publish Code Coverage
- Using FastLane with GitHub Actions
- Azure DevOps Publish to Google Play
- Azure DevOps Publish to Apple App Store
- Dotnet nuget Task Documentation

Download .NET 8

https://aka.ms/get-dotnet-8