

Getting Started

Follow steps below, to get started

Pre-requisites

1. Install Docker

```
winget install -e Docker.DockerDesktop
```

2. Ensure .NET 9 SDK is installed

```
dotnet --version
```

3. Ensure Aspire workload is up to date

```
dotnet workload list  
dotnet workload update
```

4. Ensure Dapr is installed and initialized

```
winget install -e Dapr.CLI
```

```
dapr init
```

```
dapr --version
```

Restore tools

Restore dotnet tools

```
dotnet tool restore
```

Database Migration

Restore tools

```
dotnet tool restore
```

Migration

Run following command, from PowerShell 7+, if Database schema is modified

```
dotnet ef migrations add "<migration-message>" -s  
.\src\CICDTemplate.MigrationService\CICDTemplate.MigrationService.csproj -p  
.\src\CICDTemplate.Infrastructure\CICDTemplate.Infrastructure.csproj
```

Note: Replace `<migration-message>` with a meaningful migration name

Mutation Testing

Using Stryker

Restore tools

```
dotnet tool restore
```

Run mutation tests

```
dotnet stryker --solution .\CICDTemplate.sln -r "html" -r "progress" -r "markdown"
```

Mutation tests are time consuming, hence disabled by default from `build.ps1` file.

Highly recommend running them, before raising a PR

Code Coverage

Restore tools

```
dotnet tool restore
```

Execute tests

```
dotnet test -c Release .\CICDTemplate.slnx --collect:"XPlat Code Coverage" --  
settings .\coverage.runsettings
```

Generate Coverage Report

```
dotnet reportgenerator "-reports:./**/TestResults/*/coverage.cobertura.xml" "-  
targetdir:./.coverage" -reporttypes:"Html_Dark;SonarQube"
```

Check outdated nuget packages

```
dotnet list package --outdated
```

DAPR Components

Block	Environment	Component
State Store	Redis	Redis
Configuration	Redis	Redis
Pub / Sub	In-Memory	Redis
Secret Store	Local File (json)	Kubernetes Secrets

Pub / Sub

| Redis already setup

```
apiVersion: dapr.io/v1alpha1
kind: Component
metadata:
  name: pubsub
spec:
  type: pubsub.redis
  version: v1
  metadata:
    - name: redisHost
      value: localhost:6500
```

Smoke Test - on Kubernetes

- [x] Verify CRON
- [x] Verify Pub / Sub
- [x] Aspire Dashboard
- [x] Read secret
- [x] Read Config
- [x] Verify Caching

To verify secrets, create a kubernetes secret

```
kubectrl create secret generic hello --from-literal=hello=world
```

To verify config, create a key value pair, on redis, key type string