

Configuración/mantenimiento del pipeline y su vínculo con el tablero

Utilizando GitHub Actions se automatizan procesos y verificaciones del proceso de desarrollo vinculadas al tablero.

Se utilizó una workflow dividida en diferentes partes:

- Build:
 1. Ejecución de Build de toda la solución. No hay una columna en el tablero destinada a Build ya que este proceso es automático.
 2. Subida del resultado del build - Release de la Web Api con el .exe
- Test:
 3. Ejecución de los test
 4. Subida de los resultados de los test

La workflow se ejecuta cada vez que se realiza un push o merge a la rama main.

```
1  name: build-test-dotnet
2
3  on:
4    push:
5      branches: [ main ]
6    pull_request:
7      branches: [ main ]
8
```

se utilizó Windows para la workflow tomando en cuenta los requerimientos de ambiente de producción de la documentación. La misma workflow se usó para el Build y el Test.

```
9  jobs:
10    build:
11
12      runs-on: windows-latest
13      name: Build and Test
14      steps:
```

Se especificó la versión de .net utilizada en este proyecto

```

- uses: actions/checkout@v3
- name: Setup .NET
  uses: actions/setup-dotnet@v2
  with:
    dotnet-version: 3.1.x

```

Se descargan las dependencias. Es necesario para poder realizar el build y los tests que van a continuación.

```

- name: Restore dependencies
  run: dotnet restore ../Material Obligatorio/Obligatorio/Ministerio de Turismo/Ministerio de Turismo.sln

```

Se realiza el Build de toda la solución utilizando la configuración de Release.

```

- name: Build
  run: dotnet build --configuration Release --no-restore ../Material Obligatorio/Obligatorio/Ministerio de Turismo/Ministerio de Turismo.sln


```

Se subió el resultado del Build como Artefacto de la Workflow. Se puede descargar para ejecutar el back-end en Windows.

```

- name: Upload a Build Artifacts
  uses: actions/upload-artifact@v2.2.2
  with:
    name: BuildRelease
    path: ../Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.WebApi/bin/Release/netcoreapp3.1/*

```

Artifacts	
Produced during runtime	
Name	Size
 BuildRelease	11.6 MB

Se corrieron todos los tests y se realizó un reporte. Los tests de SpecFlow no se corrieron en esta workflow ya que generaban un problema con el Build desde GitHub (pero no con el Build y ejecución de tests localmente). Se dejó como una Issue para consultar y resolver en la siguiente entrega.

```

- name: Test
  run: dotnet test ./"Material Obligatorio/Obligatorio/Ministerio de Turismo/Ministerio de Turismo.sln" -l:trx || true
  shell: bash
- if: always()
  name: Report
  uses: dorny/test-reporter@v1
  with:
    name: DotNET Tests Results
    path: '**/*.trx'
    reporter: dotnet-trx
    fail-on-error: 'false'

```

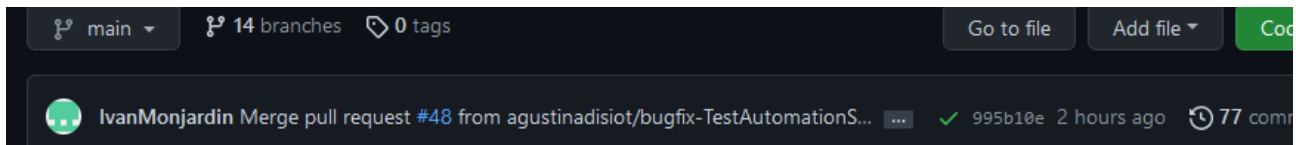
Por último se realiza un reporte de los test el cual se puede visualizar en el repositorio

```

- name: Upload test results
  uses: actions/upload-artifact@v2.2.2
  with:
    name: GitHubActionsTestsResults
    path: '**/*.trx'

```

Se puede observar el resultado de la pipeline en el mismo repositorio:



Esto nos permite detener el desarrollo si sucede que la pipeline no es exitosa. Siguiendo lo recomendado en el DevOps Handbook cuando ocurrió el Build automático en main nos juntamos todos a encontrar la solución y no continuamos hasta resolver el problema.

Jobs

succeeded 28 minutes ago in 0s

Build and Test

DotNET Tests Results

DotNET Tests Results

tests 326 passed

Report	Passed	Failed	Skipped	Time
Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.BusinessLogic.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_38.trx	37			2s
Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.DataAccess.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_25.trx	61			3s
Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.Domain.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_19.trx	116			2s
Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.JSONImporter.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_32.trx	8			2s
Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.Models.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_43.trx	34			2s
Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.ServiceRegistration.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_38.trx	24			2s
Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.WebApi.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_47.trx	38			2s

Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.XMLImporter.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_31.trx
8 ✓
2s

✓ Material Obligatorio/Obligatorio/Ministerio de Turismo/MinTur.BusinessLogic.Test/TestResults/runneradmin_fv-az90-669_2022-05-22_23_38_38.trx

37 tests were completed in 2s with 37 passed, 0 failed and 0 skipped.

Test suite	Passed	Failed	Skipped	Time
MinTur.BusinessLogic.Test.Importing.ImporterManagerTest	2 ✓			157ms
MinTur.BusinessLogic.Test.Pricing.ResortPricingCalculatorTest	4 ✓			6ms
MinTur.BusinessLogic.Test.ResourceManagers.AdministratorManagerTest	5 ✓			61ms
MinTur.BusinessLogic.Test.ResourceManagers.CategoryManagerTest	2 ✓			4ms
MinTur.BusinessLogic.Test.ResourceManagers.ChargingPointManagerTest	2 ✓			10ms
MinTur.BusinessLogic.Test.ResourceManagers.RegionManagerTest	2 ✓			4ms
MinTur.BusinessLogic.Test.ResourceManagers.ReservationManagerTest	6 ✓			152ms
MinTur.BusinessLogic.Test.ResourceManagers.ResortManagerTest	5 ✓			21ms
MinTur.BusinessLogic.Test.ResourceManagers.ReviewManagerTest	1 ✓			11ms
MinTur.BusinessLogic.Test.ResourceManagers.TouristPointManagerTest	3 ✓			10ms
MinTur.BusinessLogic.Test.Security.AuthenticationManagerTest	5 ✓			234ms

✓ MinTur.BusinessLogic.Test.Importing.ImporterManagerTest

✓ GetImportersReturnsAsExpected
✓ ImportResourcesDoesAsExpected

✓ MinTur.BusinessLogic.Test.Pricing.ResortPricingCalculatorTest

✓ TotalPriceForAccommodationCalculatedCorrectly1
✓ TotalPriceForAccommodationCalculatedCorrectly2
✓ TotalPriceForAccommodationCalculatedCorrectly3

En la tercera entrega se agregaron más funcionalidades con GitHub Actions como se explicó en el informe:

Se hicieron las siguientes modificaciones usando GitHub Actions:

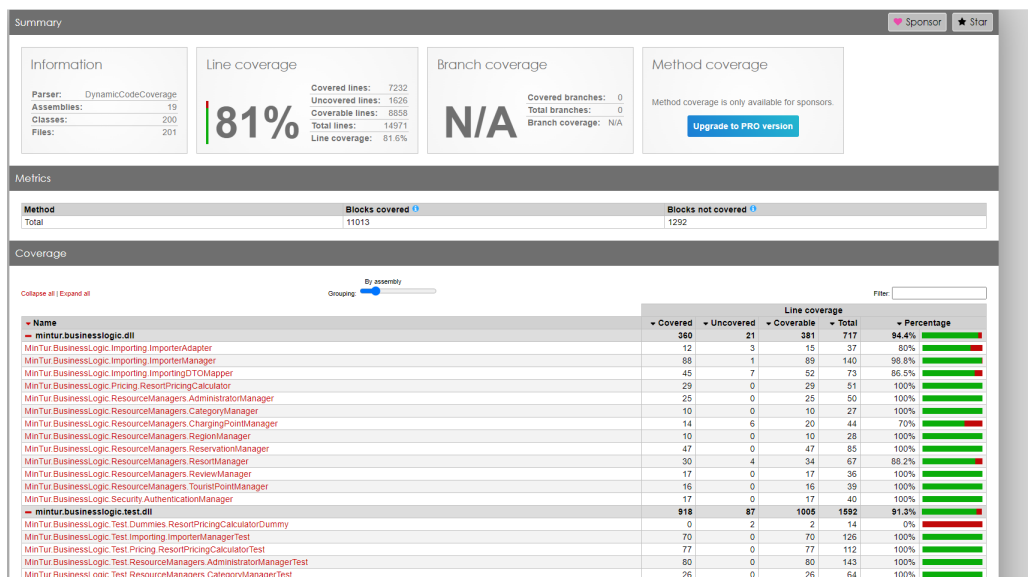
- Se automatizó el proceso de Build del front end y se sube el resultado como Artefacto de la workflow

```

29
30   - name: Build
31     run: |
32       npm install
33       npm run build
34   working-directory: './Material Obligatorio/Obligatorio/MinTurFrontend'
35
36   - name: Upload build results
37     uses: actions/upload-artifact@v2.2.2
38     with:
39       name: FrontDist
40       path: './Material Obligatorio/Obligatorio/MinTurFrontend/dist/MinTurFrontend'

```

- Se arregló problema con Build del SpecFlow (se requirió agregar “dependencias” que no eran necesarias localmente pero si en el GitHub Actions, ej. using System.Threading.Tasks)
- Code Coverage - Se sube como artefacto del workflow, es un html que se ve de la siguiente manera:



```

54
55   # Code coverage
56
57   - name: Install dotnet-coverage
58     run: dotnet tool install --global dotnet-coverage
59
60   - name: Merge Projects Tests
61     run: dotnet-coverage merge -r . -f xml
62     working-directory: Material Obligatorio/Obligatorio/Ministerio de Turismo\TestCoverageResults
63
64   - name: Install reportgenerator
65     run: dotnet tool install --global dotnet-reportgenerator-globaltool --version 5.1.9
66
67   - name: Generate report
68     run: reportgenerator "-reports:output.coverage.xml" "-targetdir:out"
69     working-directory: Material Obligatorio/Obligatorio/Ministerio de Turismo\TestCoverageResults
70
71   - name: Upload report
72     uses: actions/upload-artifact@v2.2.2
73     with:
74       name: CodeCoverageResult
75       path: './Material Obligatorio/Obligatorio/Ministerio de Turismo\TestCoverageResults/out/**'

```

