**Упражнение непрекъсната интеграция**

1. Отворете проекта HouseRenting и се уверете че всички тестове минават без проблем.
2. Качете проекта HouseRenting в новосъздадено хранилище във вашия GitHub. Във файла .gitignore добавете всички файлове от директориите:
   * + HouseRentingSystem.Services/bin
     + HouseRentingSystem.Services/obj
     + HouseRentingSystem.Tests/bin
     + HouseRentingSystem.Tests/obj
     + HouseRentingSystem.Web/bin
     + HouseRentingSystem.Web/obj
     + .vs/
3. Влизате в отдалеченото хранилище и избирате секцията Actions. Избирате workflow

Build and test a .NET or ASP.NET Core project.

1. Генерира се един \*.yml файл, който можете да си прекръстите по избор.
2. Дайте някакво смислено име на workflow-а, което ще се показва в интерфейса на GitHub Actions.

name: Build and Test

on: # Определя събитията, които задействат този workflow

push: # Workflow се стартира, когато има нов commit в посочен клон.

branches: [ "main" ]

pull\_request: # Workflow се стартира, когато се създаде или актуализира pull request

branches: [ "main" ]

jobs: # Определя задачите, които ще се изпълняват.

build: #Задава операционната система, върху която ще се изпълнява задачата

runs-on: ubuntu-latest

steps: # Поредица от инструкции, които се изпълняват в рамките на задачата

# Описание на стъпката

- name: Checkaout the code

uses: actions/checkout@v4

- name: Setup .NET

uses: actions/setup-dotnet@v4

with:

dotnet-version: 8.0.x

- name: Restore dependencies

run: dotnet restore # Тази команда изтегля всички зависимости за build-a на проекта

- name: Build

run: dotnet build --no-restore # Компилиране на кода без повторно инст. на зависимости

- name: Test

run: dotnet test --no-build --verbosity normal # Пускане на тестове

1. Направете commit за да се добави новия yaml-файл в workflows на GitHub.
2. Направете някаква промяна в локалното хранилище на проекта. Когато push-нете промените към GitHub, би трябвало тестовете да се пуснат автоматично.