

# Introduction

La branche livrable-1 contient la première version fonctionnelle d'EasySave, conformément au cahier des charges de ProSoft.

Cette version initie le développement d'une application console en .NET Core pour la gestion de travaux de sauvegarde.

## Prérequis

### 1. Installation process

- Clonez la branche livrable-1 et ouvrez le projet avec Visual Studio 2022.
- Veillez à avoir .NET Core 6.X installé.

### 2. Software dependencies

- .NET Core 6.X
- Visual Studio 2022

### 3. Latest releases

Veuillez consulter l'onglet des releases pour accéder à la dernière version du livrable 1 (Pipelines -> Releases) :

[https://dev.azure.com/faikmehmeti/G4-FISA-24/\\_release?\\_a=releases&view=mine&definitionId=1](https://dev.azure.com/faikmehmeti/G4-FISA-24/_release?_a=releases&view=mine&definitionId=1)

## Build and Test

- Un pull request vers la branche develop-livrable-1 entraîne automatiquement un processus de build et de tests (tests unitaires uniquement).
- Un pull request vers la branche livrable-1 entraîne automatiquement un processus d'intégration continue, allant du build, aux tests, jusqu'à la publication d'une release.

Ces processus sont consultables via les pipelines : [https://dev.azure.com/faikmehmeti/G4-FISA-24/\\_build](https://dev.azure.com/faikmehmeti/G4-FISA-24/_build)

NOTES : Un seul agent pourra gérer l'exécution d'une unique pipeline, le parallélisme n'est pas possible.

## Contribute

Pour contribuer à cette branche, voici le processus à suivre et les nommages à respecter :

## Depuis Boards (Menu latéral)

## 1. Dans backlogs :

- Ajout d'un User Story (Fonctionnalité) -> Button (New Work Item) -> Title: description ...
- Ajout d'une tâche fixbug -> Button (New Work Item) -> Title: description...

## 2. Dans boards (Tableau Kanban) :

- Mettre à jour l'état de la tâche -> Boards (Tableau KanBan : To Do, Doing, Done)
- Active (Fonctionnalité en cours de dev), Closed (Fonctionnalité fini).
- Ne pas mettre la card sur l'état résolved pour garder une trace, sinon elle disparaît.

## 3. Work items :

- Aller : sur la tâche que vous voulez effectuer
- Pour associer : une branche à la tâche : create a branch
- Nom de la branche : feature/NumTache-description-courte ou fixbug/NumTache-description-courte
- Based on : Toujours tirer la branche depuis develop-livable-1

## 4. Push/Pull Request/Merge Conflicts on Develop :

- Lors du push, notif : Create a pull request
- Into : develop-livable-1
- Mettre une description détaillée
- Attente du merge conflicts + Pipeline Build & Test
- Complete PR -> Complete merge

## 5. Une fois la PR mergée sur Azure Devops, mettre à jour la branche livable-1 si tout fonctionne

### • En local :

- git checkout develop-livable-1
- git pull origin develop-livable-1 (Pour récupérer les PR mergées)
- Tester sur l'application si tout marche

# Depuis Repos (Menu latéral)

### • Sur azure devops :

- Créer une nouvelle PR (onglet Pull requests) avec cette fois into livable-1
- Attente du merge conflicts + Pipeline Build & Test & Push Release
- !!! Attention lors du merge validé décocher supprimer cette branche !!!

# Namespace EasySave.ViewModels

## Classes

### [BaseViewModel](#)

Classe abstraite BaseViewModel

### [JobViewModel](#)

Classe JobViewModel

### [LangueViewModel](#)

### [MainViewModel](#)

# Release note 1 Test