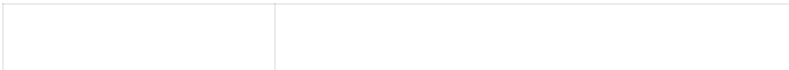




Scrum/DevOps partie 1

12.03.2024



Formateur: David Wannier

Durée: 03h30

Introduction

Planning

1. Pourquoi et qu'est-ce que l'industrialisation logicielle ?

- L'industrialisation logicielle vise à appliquer des principes de production industrielle au développement logiciel pour améliorer l'efficacité, la qualité et la rapidité de livraison des produits logiciels.

625-1 Software industrialisation

TFS 2012/2013

Definition and acceptance tests by customer team :

- User Stories to define client vision
- Graphic User Interface with mockup
- Acceptance Test definition

eXtreme Programming concepts :

- Continuous Integration
- Test Driven Development
- UnitTest and IntegrationTest running automatically
- Code coverage control
- Pair Reviewing
- Acceptance Test during iteration
- Agile Testing (iterative, incremental)

HES-SO Valais-Wallis
Page 4



Organization in short iteration (2-4 weeks) :

- The whole team pointing to the same objective
- Permanent knowledge transfer : no 'guru'
- Working software after each iteration



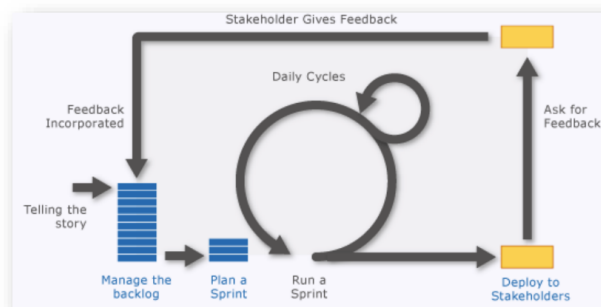
2. Cycle de vie logiciel ALM :

- ALM (Application Lifecycle Management) couvre toutes les phases du cycle de vie d'un logiciel, depuis la conception initiale jusqu'à la maintenance et la gestion des versions. Il inclut la planification de projet, le développement, les tests, le déploiement et la gestion continue.

625-1 Chapter 02 Applying ALM with Visual Studio

- Role of different ALM Processes

- Plan a release
- Define project-tracking process
- Scope a project



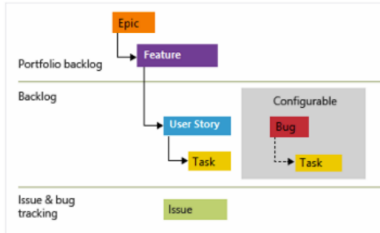
Source : <http://www.microsoftvirtualacademy.com/Content/ViewContent.aspx?et=2785&m=2778&ct=16623#bid=91gygzRMSw>
HES-SO Valais-Wallis



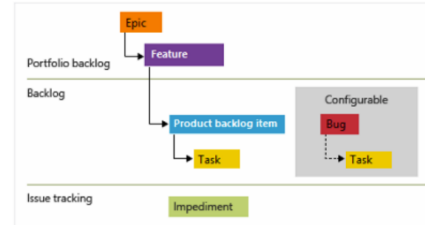
625-1 Chapter 02 Applying ALM with Visual Studio


- Choose your template

AGILE TEMPLATE



SCRUM TEMPLATE



Source : <http://www.microsoftvirtualacademy.com/Content/ViewContent.aspx?et=2785&m=2780&ct=16627#bid=91gyzRSMsw>
 HES-SO Valais-Wallis
 Page 4



3. User Stories :

- Les user stories sont des descriptions simples et concises des fonctionnalités du logiciel, écrites du point de vue de l'utilisateur final. Elles sont évaluées en story points, une unité de mesure qui reflète la complexité et l'effort nécessaire pour les implémenter.

User Stories definition 625-1 Chapter 02 Applying ALM with Visual Studio

- Product Backlog items : User Stories

As a/an	I want to...	so that...
moderator	create a new game by entering a name and an optional description	I can start inviting estimators
moderator	invite estimators by giving them a url where they can access the game	we can start the game
estimator	join a game by entering my name on the page I received the url for	I can participate
moderator	start a round by entering an item in a single multi-line text field	we can estimate it
estimator	see the item we're estimating	I know what I'm giving an estimate for
estimator	see all items we will try to estimate this session	I have a feel for the sizes of the various items
moderator	see all items we try to estimate this session	I can answer questions about the current story such as "does this include"
moderator	select an item to be estimated or re-estimated	the team sees that item and can estimate it

- A good story is INVEST
 - 1)Independent
 - 2)Negotiable
 - 3)Valuable to customers
 - 4)Estimable
 - 5)Small
 - 6)Testable

Source : XP-Dave 2011 conf David Wannier

4. Tâches :

- Les tâches sont des actions spécifiques nécessaires pour réaliser une user story. Elles sont évaluées en heures. Les tâches ne sont définies et planifiées que si la user story correspondante est validée et incluse dans le sprint.

5. Version Control System (VCS) :

- Le contrôle de version permet de suivre et de gérer les modifications du code source. Il inclut des pratiques comme le versionnage, la gestion des branches et les fusions de code. Le choix de l'hébergement (par exemple, GitHub) et la stratégie de branchement (par exemple, GitFlow) sont cruciaux pour la gestion efficace du code.

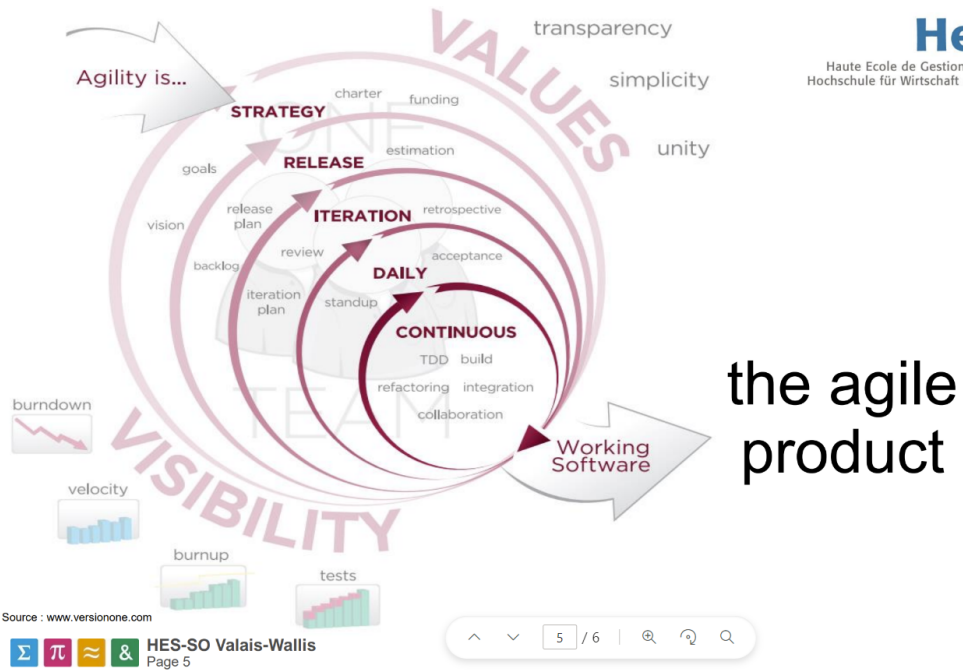
6. Systèmes de suivi des défauts (DTS) :

- Un DTS (Defect Tracking System) est utilisé pour enregistrer, suivre et gérer les bugs et les problèmes signalés. La gestion des bugs peut se faire avant ou après le déploiement CI/CD. Un bon système de feedback, comme un bouton de signalement dans l'application, facilite la remontée des bugs par les utilisateurs.

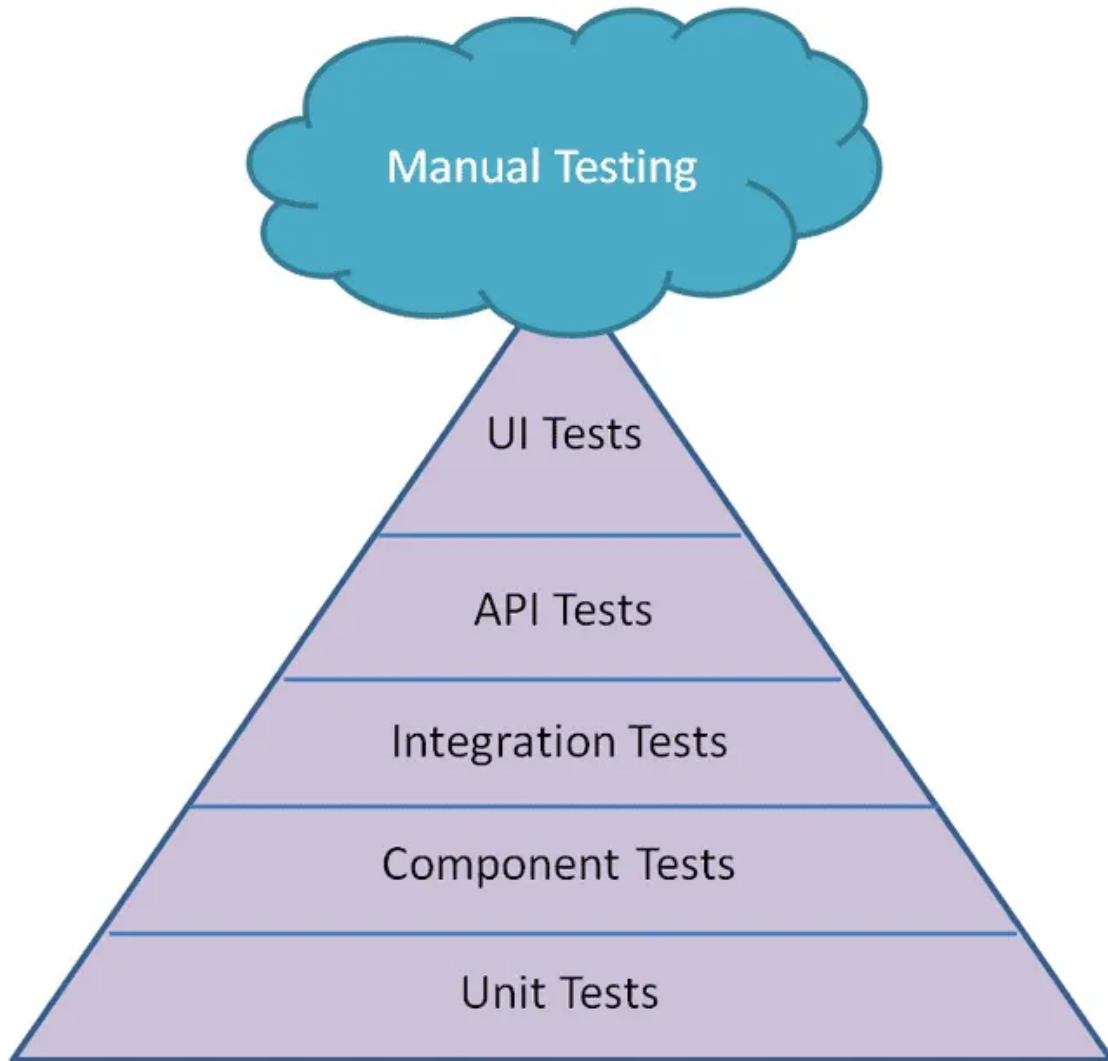
7. Intégration Continue (CI) :

- La CI est une pratique de développement où les développeurs intègrent fréquemment leur code dans un dépôt central, suivi par des builds automatiques. Cela permet de détecter rapidement les erreurs et de réduire les conflits d'intégration.

Agility



Pyramide des Tests



1. **Unit Testing** : Tests unitaires permettant un retour rapide (environ 1 seconde). Ils vérifient le bon fonctionnement des plus petites unités de code.
2. **Integration Testing** : Tests d'intégration offrant un retour moyen (quelques minutes). Ils vérifient les interactions entre différentes unités ou modules.
3. **E2E Testing** : Tests de bout en bout (E2E) fournissant un retour lent (plusieurs heures). Ils simulent des scénarios réels pour vérifier que l'application fonctionne correctement du début à la fin.
4. **UX Testing** : Tests d'expérience utilisateur (UX) réalisés manuellement. Très lents, ils évaluent la facilité d'utilisation et l'interface utilisateur.

Les tests de sécurité doivent être effectués à chaque niveau de la pyramide pour garantir la robustesse du logiciel.

Outils :

- UI : Xray, Selenium, Cypress, etc.

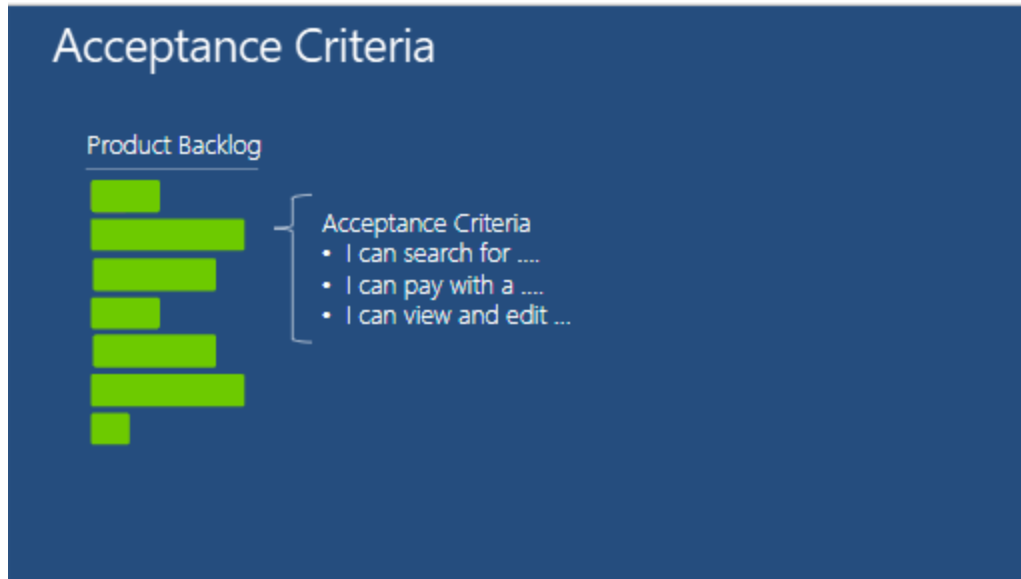
Objectif : Dynamiser le développement pour réduire le temps d'ajout de nouvelles fonctionnalités et des tests, améliorant ainsi la qualité globale du produit.

SCRUM

1. **Équipe unie vers un objectif commun :** Chaque membre de l'équipe travaille en collaboration pour atteindre les objectifs fixés pour le sprint.
2. **Transfert de connaissance permanent :** Éviter la dépendance à un seul expert (pas de gourou). Partager les connaissances régulièrement pour maintenir un niveau de compétence élevé au sein de l'équipe.
3. **Logiciel fonctionnel après chaque itération :** Chaque sprint se termine avec un logiciel opérationnel, prêt à être déployé ou présenté au client.

Pour mettre ceci en place, il faut :

1. **Définition et tests d'acceptation par l'équipe cliente :** Les critères d'acceptation sont définis par l'équipe cliente et servent de base pour valider les user stories.
 - Concepts de l'eXtreme Programming (XP) : Pratiques comme le développement par test (TDD), les intégrations fréquentes, la rétroaction rapide et le développement itératif.



Bibliographie

- *User Stories Applied* par Mike Cohn : Un guide complet sur la rédaction et l'utilisation des user stories.
- *Continuous Integration* par Paul Duvall, Steve Matyas et Andrew Glover : Explique les meilleures pratiques pour implémenter et maintenir une infrastructure d'intégration continue.

Outils pour SCRUM

- **Tuleap** : Suite open-source utilisée pour des projets complexes comme le développement de l'avion Concorde.
- **JIRA** : Outil populaire de gestion de projet et de suivi des tâches.
- **Azure DevOps** : Plateforme intégrée pour le développement, les tests et la livraison continue.

Work Packages

- **Équipe** : Chaque équipe a son propre backlog de tâches.
- **Release** : Combinaison de tous les work packages pour une version donnée.
- **Thème** : Sujet principal du développement.
 - **User Story** : Fonctionnalité utilisateur spécifique.

- **Tâches** : Actions nécessaires pour réaliser les user stories.

Story Points : Les story points sont des estimations relatives de l'effort nécessaire pour compléter une user story. Ils sont affinés à chaque sprint pour refléter la capacité de l'équipe.

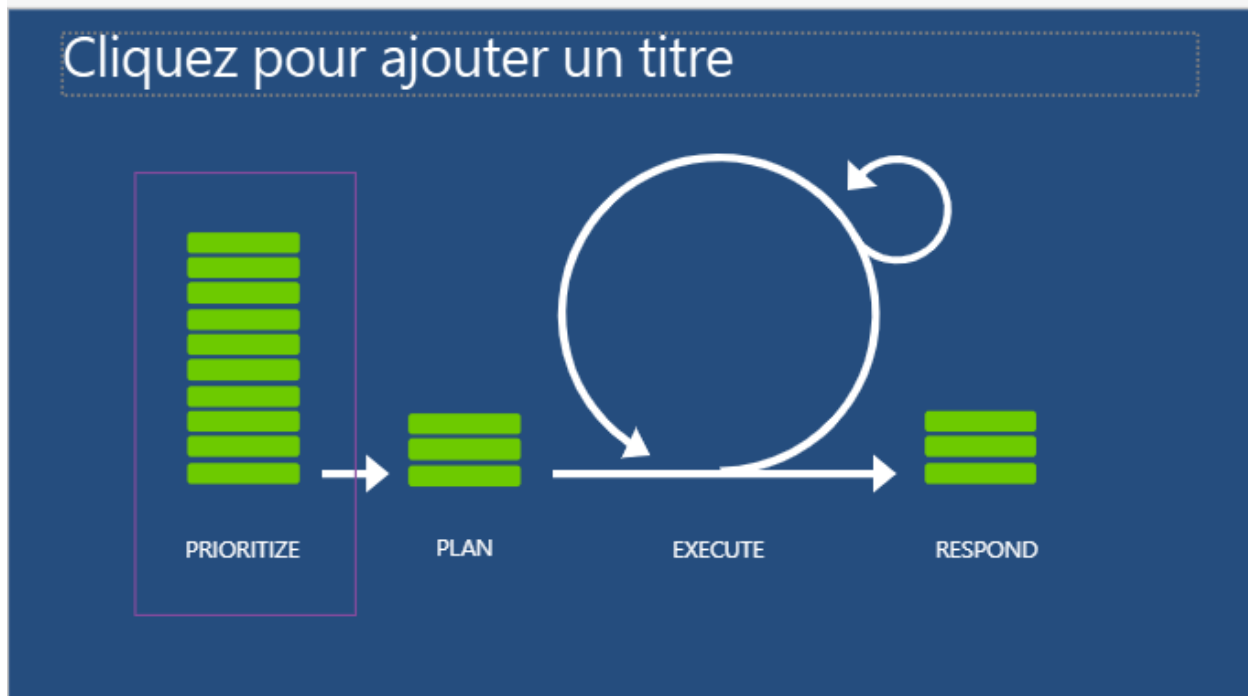
Tous les outils ne gèrent pas la fin des sprints de la même manière. Par exemple, Azure DevOps permet de déplacer les user stories inachevées vers le sprint suivant, tout en conservant les tâches terminées dans le sprint actuel.

Statuts

- **Closed** : Accepté par le client.
- **Resolved** : Terminé par le développeur.
- **Active** : En cours.

Product Owner (P.O.)

- **Priorise les tâches** : Le P.O. est responsable de la priorisation des user stories et des tâches dans le backlog, en fonction des besoins et des objectifs du projet.



Gestion des Bugs

- **1 bug = 1 user story** : Chaque bug est traité comme une user story, évalué en début de sprint et corrigé dans le sprint suivant.
- **1 bug = 1 tâche** : Réserver des story points pour les bugs, basés sur des mesures historiques.

Critères d'Acceptation

Les critères d'acceptation définissent ce qui est nécessaire pour considérer une user story comme terminée et fonctionnelle. Ils sont testés dans les tests GUI de la pyramide des tests.

Revue de Sprint

- **Revue de Sprint** : Focus sur les user stories complétées, pas les tâches individuelles.
- **Planification de Sprint** : Planification des user stories, suivie de l'évaluation des tâches par l'équipe de développement.

Haute Ecole de Gestion & Tourisme
Hochschule für Wirtschaft & Tourismus

625-1 Chapter 02 Applying ALM with Visual Studio

- Define sprint capacity (number of hours per day)

Team Member	Capacity Per Day	Activity	Days Off
Anna Herрман	3		1 day
Britan Keller	4		0 days
Cameron Skinner	5		0 days
Julia Byrnes	2		0 days

Source : <http://www.microsoftvirtualacademy.com/Content/ViewContent.aspx?et=2785&m=2780&ct=16627#bid=91gyzRSMSw>

HES-SO Valais-Wallis



Definition of Done

Avant de passer une user story en "closed", elle doit passer avec succès les tests unitaires et répondre à tous les critères d'acceptation définis.

10 Things to Take Away

1. Define done
2. Fail fast
3. Understand your team's velocity
4. Finish what you start
5. Bugs are real work
6. Write unit tests – always
7. Autonomy, mastery, purpose
8. Do the right thing... at the right time
9. Plan until you need to learn
10. Quality is not a variable

Estimation des Coûts et des Délais

Pour estimer les coûts et les délais d'un projet avec des sprints :

1. Calculer le coût d'un story point :

- Réaliser un premier sprint.
- Évaluer le nombre de story points réalisables dans le temps imparti.
- Calculer le coût de l'équipe de développement par story point.

2. Technique de journée idéale :

- Estimer qu'une demi-journée idéale = 1 story point (sans distractions).
- Multiplier par le nombre de jours nécessaires en fonction des story points estimés.

3. Cône d'incertitude :

- Calculer le temps et le coût total, puis ajuster en multipliant par un facteur (par exemple, 2 ou 3).

- Affiner les estimations à chaque sprint en fonction des données accumulées.

Conclusion

Le succès d'un projet SCRUM dépend de la planification minutieuse, de l'évaluation précise des story points, et de la gestion efficace des sprints et des bugs. Utiliser des outils adaptés et suivre les bonnes pratiques permet de maximiser l'efficacité et de garantir la satisfaction du client. SCRUM, avec ses cycles itératifs et ses revues fréquentes, offre une approche structurée et adaptable pour le développement logiciel, permettant de répondre rapidement aux changements et aux besoins des clients.

Azure DevOps creation guide

May 2022

Students: Moix Gaetan

Professor: Wannier David

TABLE OF CONTENTS

1. CREATION OF AN ORGANIZATION 1

2. CREATION OF ITEMS ERREUR ! SIGNET NON DEFINI.

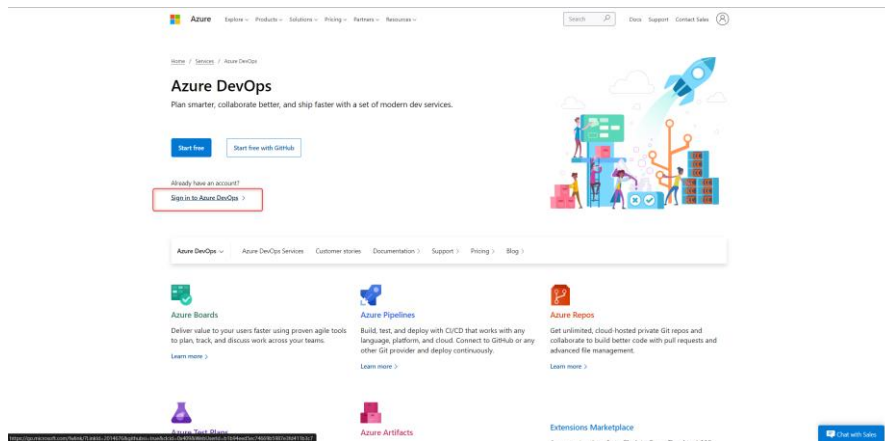
3. IMPORT CODE FROM GITHUB ERREUR ! SIGNET NON DEFINI.

4. CREATE A SIMPLE PIPELINE ERREUR ! SIGNET NON DEFINI.

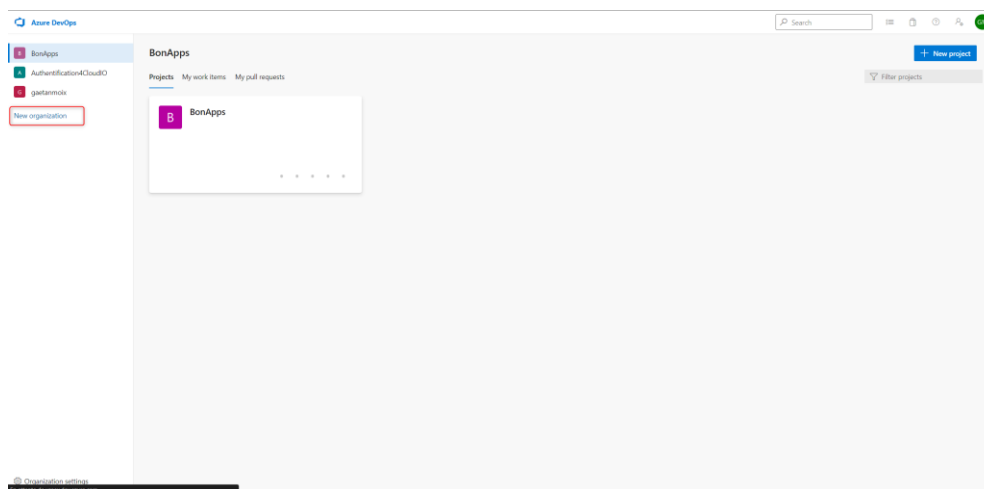
1. Creation of an organization

Go to this link to connect to Azure DevOps platform:

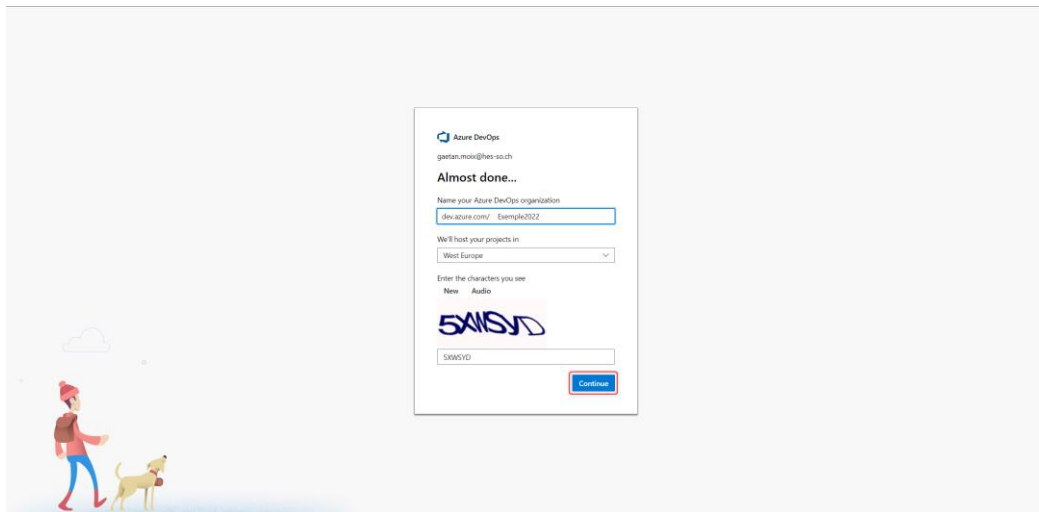
<https://azure.microsoft.com/en-us/services/devops/>



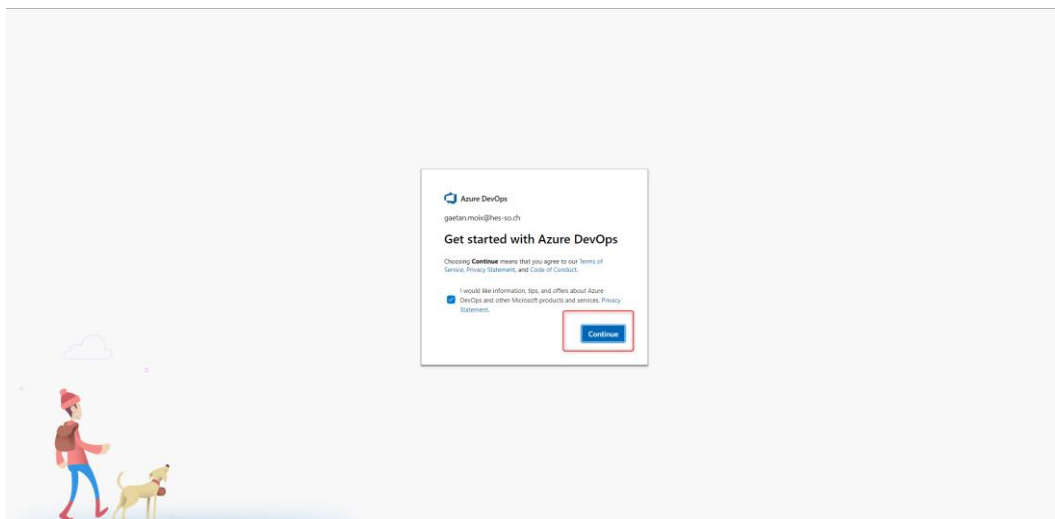
Click on “Sign in to Azure DevOps” then “New organization”



Enter the name of your organization.



Press "Continue".





Create a private project with TFS repository


Create new project



Visibility

 **Public** 


Anyone on the internet can view the project. Certain features like TFVC are not supported.


 **Private**


Only people you give access to will be able to view this project.

By creating this project, you agree to the Azure DevOps [code of conduct](#)

^ Advanced

Version control 

Team Foundation Version Control 

Work item process 

Agile 

Cancel

Create

The screenshot shows the Azure DevOps web interface. At the top, the browser address bar displays <https://dev.azure.com/EASILab>. Below the address bar, there are navigation icons and a list of bookmarks including 'Import bookmarks...', 'Getting Started', 'Progetto formativo cu...', and 'BRIDGE program'. The main header area features the 'Azure DevOps' logo and a navigation menu with the following items: 'EASILab' (highlighted), 'Authentication4CloudIO', 'BonApps', and 'New organization'. The main content area is titled 'EASILab' and contains three tabs: 'Projects', 'My work items', and 'My pull requests'. The 'Projects' tab is active, showing a card for the 'HES00' project with a green square icon containing a white letter 'H'. Below the project card, there are five small grey dots indicating a list of items.