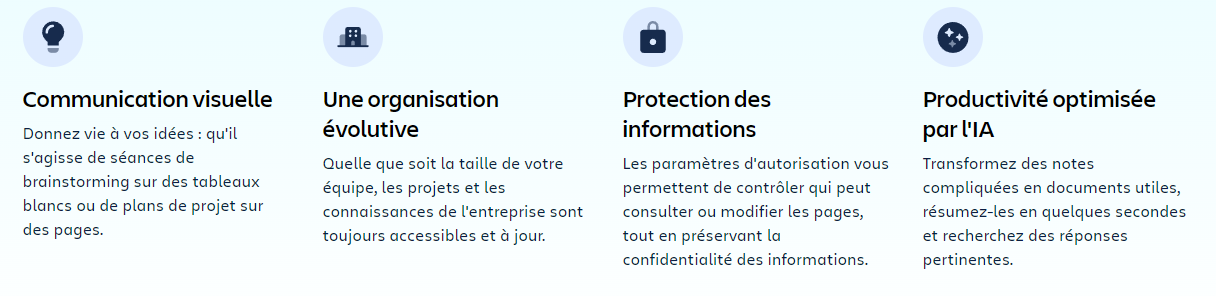
**Petit overview sur .Net :**

.NET est un ensemble d'outils et de technologies de développement créé par Microsoft. C'est un choix populaire pour créer une grande variété d'applications, de sites web à des logiciels d'entreprise robustes. Avec des langages comme C# et des frameworks comme ASP.NET, il offre des solutions complètes pour le développement et le déploiement, souvent intégrées étroitement avec les services cloud de Microsoft. En bref, c'est une boîte à outils polyvalente pour les développeurs qui veulent créer des applications puissantes et évolutives.

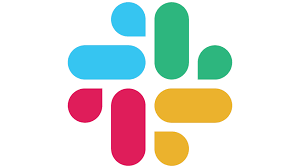
**Plan :**

Confluence : C’est un espace de travail en équipe où la connaissance et la collaboration se rejoignent. Les pages dynamiques permettent à votre équipe de disposer d'un endroit pour créer, capturer et collaborer sur le projet ou l'idée de votre choix. Grâce aux espaces, votre équipe peut structurer, organiser et partager les tâches, afin que chacun de ses membres dispose d'une visibilité sur les connaissances institutionnelles ainsi que d'un accès aux informations nécessaires pour optimiser le travail.



Redmine :

Redmine est une application web de gestion de projet et de ticketing multiplateforme, principalement utilisé avec la méthode agile Scrum. Elle s’adresse essentiellement aux développeurs et aux petites et moyennes entreprises qui recherchent un outil de gestion de projet gratuit et flexible.



Slack :

C’est une plateforme de communication d'équipe qui permet la collaboration en temps réel à travers des canaux de discussion, des messages directs et des intégrations avec d'autres outils. Il facilite la communication entre les membres de l'équipe, la coordination des projets et le partage d'informations.

Petite comparaison entre les deux plateformes

|  |  |  |
| --- | --- | --- |
| Critère | Redmine | Slack |
| Type | Application de gestion de projet | Plateforme de communication d'équipe |
| Coût | Gratuit (open-source) | Plan gratuit limité, options payantes pour des fonctionnalités avancées |
| Communication | Limitée aux commentaires sur les problèmes | Communication en temps réel, messagerie instantanée |
| Intégrations | Moins d'intégrations disponibles | Nombreuses intégrations avec d'autres outils et services |

**Code :**

Github et Github : vu dans la section de springboot



Bitbucket : C’est une autre plateforme de développement de code source basée sur Git, mais elle est principalement utilisée par les équipes qui utilisent les outils et les services d'Atlassian. Elle offre des fonctionnalités similaires à GitHub et GitLab, telles que le suivi des problèmes, les demandes de tirage, etc.

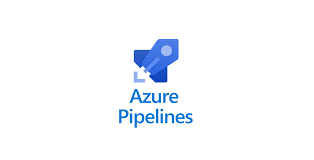
|  |  |  |  |
| --- | --- | --- | --- |
| Critère | GitHub | GitLab | Bitbucket |
| Type | Hébergement de code source | Plateforme de développement | Plateforme de développement |
| CI/CD intégré | Non | Oui | Non |
| Licence | Propriétaire | Open-source | Mixte |
| Modèle de développement | Open-source, utilisé par la communauté et les entreprises | Open-source, utilisé par la communauté et les entreprises | Propriété d'Atlassian, utilisé par les entreprises qui utilisent les produits Atlassian |

**Build :**

MsBuild :



MSBuild (Microsoft Build Engine) est l'outil de build natif pour les projets .NET. Il est largement utilisé pour compiler, générer des artefacts, et automatiser le processus de build dans les projets .NET. Il prend en charge la compilation de projets .NET dans divers formats, y compris les applications web, les applications de bureau et les bibliothèques de classes.

Azure Pipelines :

C’est un service d'intégration continue et de déploiement continu (CI/CD) proposé par Microsoft Azure. Il offre une solution complète pour automatiser le processus de build, de test et de déploiement des applications. Il prend en charge les projets .NET, offrant une intégration transparente avec les services Azure.

**Test :**

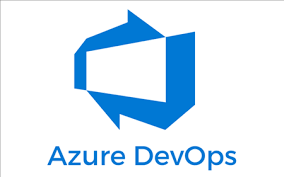
NUnit :

C’est un framework de test unitaire open-source pour .NET. Il offre un ensemble complet d'outils pour écrire et exécuter des tests unitaires dans des projets .NET. prend en charge le cadre de test basé sur les attributs, permettant aux développeurs d'écrire des tests clairs et expressifs à l'aide d'annotations simples. Il offre des fonctionnalités avancées telles que les paramètres de test, les théories (pour les tests paramétrés), les assertions flexibles, et les annotations pour la configuration des tests.

xUnit.net :

C’est un framework de test open-source pour .NET, basé sur le modèle de test xUnit. Il est conçu pour être simple, extensible et facile à utiliser. Il suit le modèle de test xUnit, où chaque test est une méthode publique qui ne prend aucun argument et utilise des assertions pour vérifier le comportement attendu . Il est souvent préféré pour sa simplicité et sa clarté, ce qui en fait un choix populaire parmi les développeurs .NET.

**Release :**

Azure DevOps :

C’est une suite d'outils proposée par Microsoft pour la gestion du cycle de vie des applications, y compris la gestion de projet, le contrôle de version, l'intégration continue, et le déploiement continu.

Il permet de créer des pipelines de déploiement personnalisés pour automatiser la publication d'applications .NET sur diverses plateformes, y compris Azure, des serveurs locaux, des conteneurs, etc.

**Deploy :**

Octopus Deploy :

C’est une plateforme de déploiement continu qui facilite le déploiement d'applications .NET sur divers environnements, y compris des serveurs locaux, des machines virtuelles, des conteneurs, et des services cloud. Il prend en charge les déploiements automatisés ainsi que les déploiements manuels, offrant ainsi une flexibilité pour les différentes étapes du processus de déploiement.

Il offre des fonctionnalités avancées telles que la gestion des configurations, le suivi des versions, les notifications, et l'intégration avec des outils de gestion de version comme Git et TFS.

**Operate :**

ServiceNow :



C’est une plateforme cloud qui offre une suite complète de services ITSM, ainsi que des fonctionnalités de gestion des services métier (BSM), de gestion des opérations de sécurité (SecOps) et de développement d'applications.

ServiceNow offre une solution complète pour la gestion des services informatiques, en automatisant et en intégrant des processus tels que la gestion des incidents, la gestion des changements, la gestion des versions, etc.

Opsgenie :

C’est une plateforme de gestion des alertes et des incidents, conçue pour aider les équipes informatiques à répondre rapidement aux incidents, à minimiser les temps d'arrêt et à améliorer la disponibilité des services.

Opsgenie permet de collecter, de trier et de hiérarchiser les alertes en fonction de règles de routage et de planification personnalisées, en garantissant que les bonnes personnes sont informées au bon moment.

Il offre des fonctionnalités avancées telles que les escalades d'alertes, les rotations d'astreinte, les intégrations avec des outils de surveillance, etc.

**Monitor :**

La phase finale du cycle DevOps, consiste à **surveiller l'environnement de production**. Cette phase est gérée par un système de supervision capable de collecter et visualiser graphiquement les métriques de données, tout en fournissant des analyses sur le comportement, les performances et les erreurs des applications et de l’infrastructure.

Prometheus :

C’est un système de surveillance open-source conçu pour surveiller les applications et les services distribués. Il collecte des métriques à intervalles réguliers à partir des cibles configurées, telles que des applications, des services, des serveurs, etc et offre des requêtes flexibles pour analyser et agréger les métriques, ainsi que des alertes pour détecter les conditions anormales.

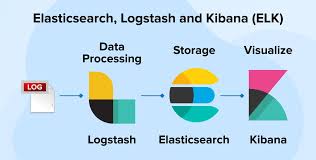
ELK (Elasticsearch, Logstash, Kibana) :

C’est une suite de logiciels open-source pour la collecte, la centralisation, la recherche et la visualisation des données de journalisation.

Elasticsearch : Moteur de recherche et d'analyse distribué utilisé pour stocker et interroger les données de journalisation.

Logstash : Outil de collecte, d'analyse et de transformation des données de journalisation à partir de différentes sources.

Kibana : Interface de visualisation et d'analyse des données de journalisation, permettant de créer des tableaux de bord interactifs et des visualisations



**Test :**

JUnit :



C’est un framework de test unitaire pour la plateforme Java.

Il permet aux développeurs de créer des tests pour vérifier le comportement des différentes parties de leur code Java, notamment les classes et les méthodes.

JUnit fournit un ensemble de classes et d'annotations qui facilitent l'écriture, l'exécution et la gestion des tests unitaires.

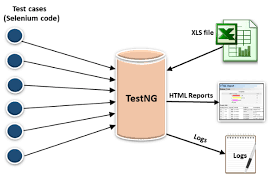
Les tests écrits avec JUnit peuvent être regroupés en suites de tests et exécutés automatiquement à chaque fois que des modifications sont apportées au code.

TestNG :

TestNG est un autre framework de test pour la plateforme Java, conçu pour offrir des fonctionnalités plus avancées que JUnit.

Il prend en charge différents types de tests, y compris les tests unitaires, les tests d'intégration, les tests fonctionnels et les tests de régression.

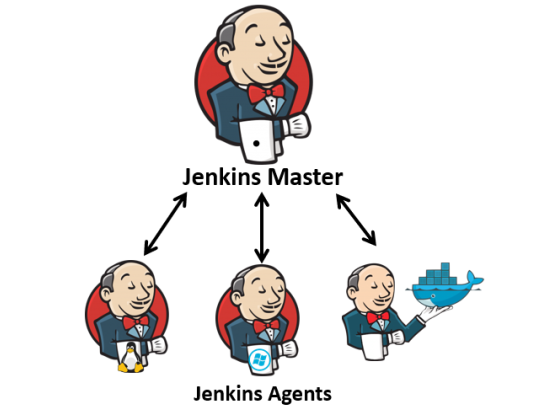
TestNG permet de définir des tests à l'aide d'annotations, de grouper les tests dans des suites de tests et de spécifier des dépendances entre les tests.

Il offre également des fonctionnalités telles que la parallélisation des tests, la génération de rapports détaillés et la configuration flexible des paramètres de test.

**Release :**

Jenkins est un outil **d'automatisation de serveur open source** qui permet aux développeurs d'intégrer rapidement des changements dans leur projet et de trouver des problèmes tôt en automatisant les processus de construction et de test.

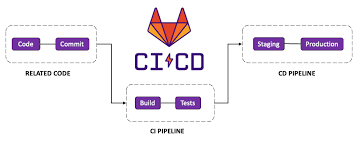
Jenkins suit un **modèle maître-esclave** où le serveur maître (**Jenkins Master**) délègue les tâches aux noeuds esclaves, ce qui permet de distribuer les charges de travail. Ces agents peuvent alors être déployés sur **n'importe quel type de plateforme** : [machine virtuelle](https://blent.ai/blog/a/virtualisation-tout-comprendre), image [Docker](https://blent.ai/blog/a/docker-a-quoi-ca-sert), etc.



Gitlab CI/CD :

GitLab CI/CD est une fonctionnalité intégrée de GitLab, une plateforme de gestion de dépôts Git.

Il offre un système d'intégration continue et de déploiement continu intégré directement dans le processus de développement sur GitLab.

GitLab CI/CD utilise des fichiers de configuration (".gitlab-ci.yml") pour définir les étapes de construction, de test et de déploiement des applications.

**Deploy :**

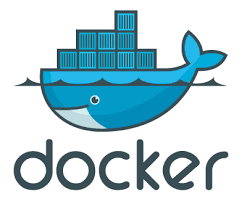
Docker :

Docker est une plateforme open-source qui permet de développer, d'emballer et de distribuer des applications dans des conteneurs légers et portables.

Il utilise des conteneurs pour encapsuler les applications et leurs dépendances, ce qui permet de garantir leur portabilité et leur cohérence lors du déploiement sur différentes infrastructures.

Docker fournit un ensemble d'outils pour créer, exécuter et gérer des conteneurs, ainsi qu'un registre centralisé pour le stockage et le partage d'images de conteneurs.

Les commandes phares :



docker build : Construit une image Docker à partir d'un fichier Dockerfile.

docker run : Exécute un conteneur à partir d'une image Docker.

docker stop : Arrête l'exécution d'un ou plusieurs conteneurs.

docker rm : Supprime un ou plusieurs conteneurs.

docker ps : Affiche les conteneurs en cours d'exécution.

docker images : Affiche les images Docker disponibles localement.

docker pull : Télécharge une image Docker depuis un registre.

Kubernetes :

C’est un système open-source pour l'automatisation du déploiement, de la mise à l'échelle et de la gestion des applications conteneurisées.

Il permet de déployer des applications dans des environnements distribués, de gérer les ressources et de garantir leur disponibilité et leur performance.

Kubernetes utilise des concepts tels que les pods, les déploiements, les services et les réplicas pour orchestrer le déploiement et la gestion des conteneurs.

Commandes phares :

kubectl create : Crée un objet Kubernetes à partir d'un fichier de configuration.

kubectl apply : Applique les changements à un objet Kubernetes.

kubectl get : Affiche les ressources Kubernetes.

kubectl describe : Affiche des détails sur une ressource Kubernetes.

kubectl delete : Supprime une ressource Kubernetes.

kubectl scale : Modifie le nombre de répliques d'un déploiement.

kubectl rollout : Gère les déploiements et les mises à jour de manière progressive.

Ansible :

C’est un outil open-source d'automatisation et de gestion de configuration, conçu pour simplifier les tâches de déploiement, de configuration et de gestion des infrastructures.

Il utilise des playbooks (fichiers YAML) pour décrire les tâches à effectuer sur les différents nœuds du réseau, de manière déclarative et sans nécessiter l'installation d'agents sur les machines cibles.

Ansible peut être utilisé pour automatiser des tâches de déploiement d'applications, de configuration de serveurs, de gestion des mises à jour, etc.

Commandes phares :

ansible-playbook : Exécute un playbook Ansible sur les nœuds cibles.

ansible : Exécute une commande ad hoc sur les nœuds cibles.

ansible-galaxy : Interagit avec Galaxy, le site de partage de rôles Ansible.

ansible-vault : Gère les fichiers chiffrés utilisés pour stocker des données sensibles.

ansible-doc : Affiche la documentation pour un module Ansible spécifique.

ansible-pull : Exécute un playbook sur un nœud cible, en tirant le playbook depuis un dépôt Git.

**Operate :**

Zabbix est un logiciel de surveillance réseau et d'infrastructure qui offre une gamme complète de fonctionnalités pour surveiller les performances, les statistiques et les événements des systèmes informatiques.

zabbix\_server : Lance le serveur Zabbix.

zabbix\_agent : Lance l'agent Zabbix sur un hôte à surveiller.

zabbix\_get : Récupère les données de surveillance d'un agent Zabbix.

zabbix\_sender : Envoie des données de surveillance à un serveur Zabbix.

Zabbix web interface : Interface utilisateur web pour configurer, visualiser et gérer les données de surveillance.



**Measure :**

|  |  |  |
| --- | --- | --- |
| **Outil** | **Description** | **Commandes phares** |
| **Prometheus** | - Système de surveillance et d'alerte open-source.  - Collecte, stocke et visualise des métriques de systèmes et d'applications. | - prometheus : Lance le serveur Prometheus.  - promtool : Outil de ligne de commande pour diverses opérations.  - prometheus.yml : Fichier de configuration principal.  - Alertmanager : Gère les alertes. |
| **Grafana** | - Plateforme de visualisation et d'analyse de données open-source.  - Crée des tableaux de bord personnalisés avec des visualisations de données. | - grafana-server : Lance le serveur Grafana  - Datasources : Configuration des sources de données.  - Dashboards : Création et gestion des tableaux de bord.  - Alerting : Configuration des règles d'alerte. |