

Devoir Maison - Semaine 3

Quelques outils

⚠ L'objectif de cet exercice est de vous entraîner à parcourir rapidement des documentations. Tentez d'y répondre au mieux et au plus vite. Vous ne devriez y passer plus de 2 heures.

On utilise GitLab pour faire le versioning du code, le tracking des issues, la roadmap, l'intégration continue et le déploiement continu. C'est beaucoup de choses différentes, qui sont ordinairement réalisées par des services séparés. Pour les outils suivants couramment utilisés dans l'industrie logicielle, déterminer leur usage et à quoi ils correspondent dans GitLab : jenkins, asana, circleCI, Jira, trello, travis, waffle.io, Azure DevOps.

Quel est sont les différences / les similarités entre maven, PyPI, sonarqube, gradle, nexus, docker hub, artifactory, yarn, npm ?

A quels besoin de documentation répondent :

- le « user manual »
- la référence technique
- un tutoriel « getting started » ?

Pour tous les services précédemment cités et lorsque cela s'applique, identifier le lien officiel vers ces trois types de documentations.

Quels services vous ont paru particulièrement bien / mal documentés ?

Afin de guider vos choix de technologies, de quels aspects tenez-vous plus compte et dans quel ordre ?

En bonus, visiter les sites suivants : <https://devhints.io/>, <https://dev.to/>, <https://learnxinyminutes.com/>, <https://excalidraw.com/>, <https://regex101.com/>, <https://jsoneditoronline.org/>, <https://codesandbox.io/>, <https://speed.cloudflare.com/>, <https://miro.com/>, <https://metroretro.io/>
(Sélection d'outils inspirée de @jesuisundev.)

Typescript - Classes, interfaces et types

Quelles sont les différences entre un type et une interface ? Dans quelles situations une solution est-elle préférable qu'une autre ?

Faire le Beginner/Learner Challenge (et optionnellement, le Intermediate/Advanced Challenge) :

<https://devblogs.microsoft.com/typescript/type-treat-2021-day-1/>