

Introduction à la méthode Behavior-Driven Development (BDD)

Durée: 1 jour

Public cible: Chargé de projet

À qui s'adresse cette formation ?

Analystes fonctionnels, développeurs, architectes fonctionnels et architectes logiciels.

Objectifs de la formation

- Décrire le BDD, en quoi cela consiste et pourquoi l'utiliser
- Définir les requis fonctionnels avec des spécifications et exemples
- Organiser le tout avec les user stories et du développement itératif
- Améliorer la communication afin de maximiser la valeur affaires des produits livrés par l'équipe de développement

Description sommaire

Imaginez ne pas seulement pouvoir automatiser vos tests unitaires, mais aussi des tests fonctionnels. Pouvoir comprendre un besoin seulement par la lecture de quelques exemples. Combler le fossé de communication entre le domaine d'affaires et le développement. Bref, passer des spécifications à des exemples, puis à concevoir le code tout en ayant toujours en tête la vision et les objectifs d'affaires. L'approche BDD ou Behavior-Driven Development a été inventée, puis perfectionnée, pour pouvoir répondre à tout cela. Cette introduction enseigne les fondements de cette approche et tout ce qui est nécessaire pour la mettre en œuvre de manière efficace à travers une équipe de développement agile.

Contenu du plan de cours

Définition du BDD

- Quel est le problème ?
- Pourquoi et quand s'en servir ?
- La place parmi les autres types de tests

- Le cycle du BDD/TDD

Vision et impacts

- Pourquoi fait-on cela ?
- L'approche « Impact Mapping »
- Planifier nos itérations et livraisons par le « story mapping »

Le formalisme « Gherkin » ou « Given-When-Then »

- Description
- Comment s'en servir, même en français !

Atelier de spécification

- Explorer, découvrir et définir les comportements
- Comment l'organise-t-on ?
- L'importance des « 3 amigos »

Problèmes courants

- Trucs et astuces
- Patrons de la spécification par l'exemple
- Scripts de tests vs. Spécifications
- Partir avec des mauvaises « user stories »

Outillage

- Bref survol des possibilités que les outils peuvent apporter
- Rapports et automatisation

Approche et méthode pédagogique

Présentation magistrale avec plusieurs activités et réflexions à faire seul ou en équipe. Un cas fictif sera utilisé tout au long de la journée et évoluera en fonction des notions apprises. Prenez note qu'il

n'y aura pas d'exercice technique, soit de la programmation, lors de cette formation. C'est pourquoi elle vise autant le côté analyse que développement.

Prérequis

Avoir une connaissance de base de l'agilité.

Recommandations

Se familiariser avec les concepts de base

- Méthodes agiles (Scrum, user stories)
- Cycle de développement logiciel
- Identifier un contexte projet réel (exigences, communication métier–TI)
- Réfléchir à vos enjeux actuels : incompréhension des besoins défauts de communication entre équipes qualité des spécifications