

Des Lego, des bonnes pratiques et du fun !



PARISTESTCONF

Cécile Especel

facilitatrice / scrum master / coach /
pipette d'huile



Idée originale de Mike Bowler et de Bryan Beecham

<https://gargoylesoftware.com/resources/>

#LEGO© exercises

Faisons des jeux !

Aujourd'hui vous allez être des super-héroïnes et super-héros : une équipe de dev ! On va, avec des Lego, essayer de comprendre en quoi consiste le code et par la même comprendre le fabuleux métier de dev avec ces super-pouvoirs et parfois avec ces super méchants.

- Clean code : 15mn
- La simplicité : 15mn
- Refacto : 30mn
- Intégration continue : 30mn

Clean code

Clean Code - préparation

Chaque équipe a 2 tas de Lego

Chaque équipe trie 1 de ses tas : par forme, par couleur, etc.

L'autre tas reste intact

Clean Code - dev

Go pour dev 2 structures qui contiennent une pièce de chaque type

Rouge 2x3 \neq Rouge 2x4

Rouge 2x3 \neq Bleu 2x3

Temps	Equipe 1	Equipe 2
Pile 1		
Pile 2		

Clean code - Débrief

- Qu'avez vous ressenti ? Comment ça s'est passé ?
- Y'a-t-il un tas de Lego avec lequel c'était plus simple de bosser ? Qu'est-ce que rendait plus simple son utilisation ?
- En comparant le temps que vous avez mis à construire chaque pile : y'a-t-il une pile pour laquelle ça a été plus rapide ? Laquelle et pourquoi ?
- Et si vous comptez le temps que ça vous a pris de trier le tas, est-ce que ça change le résultat ?
- Regardez le travail de l'autre équipe : est-ce que la manière de trier le tas a de l'importance ?
- Sans savoir ce que vous allez devoir construire après, y'a-t-il un tas avec lequel vous avez plus envie de bosser ?
- Est-ce que vous pensez que si vous aviez eu plus de temps pour trier le 1er tas, ça aurait changé beaucoup de chose ?

Simplicité



Simplicité - clean

Nettoyez votre environnement de construction, plus aucun
Lego

Simplicité - clean

Pourquoi cet environnement est-il parfait ?

Simplicité - clean

- Pas de bug
- Satisfait tous les besoins (c'est-à-dire aucun)
- Il tourne vite et indéfiniment
- Il ne coûte rien
- Il a été livré en un rien de temps
- C'est très facile de construire dessus
- Il est très facile à comprendre et à maintenir
- etc.

Simplicité - itération 1

Construisez 1 personne et une maison

Simplicité - Itération 1 - debrief

- Est-ce que quelqu'un n'a construit que la personne ou que la maison ?
- Est-ce que quelqu'un a utilisé toutes les briques qui étaient dispo ? Est-ce que vous l'auriez fait si vous aviez eu plus de temps ?
- Si on vous avait dit que chaque brique coûtait 1000€, est-ce que vous auriez changé votre archi ? Qu'est-ce qui aurait été différent ?

Simplicité - itération 2

Construisez 1 personne et une maison

Simplicité - Itération 1 - debrief

- Est-ce que quelqu'un n'a construit que la personne ou que la maison ?
- Est-ce que quelqu'un a construit quelque chose de plus grand et/ou de plus compliqué qu'à la 1ère itération ?
- Est-ce que quelqu'un a construit quelque chose de plus petit et/ou de plus simple que la 1ère fois ?
- Est-ce que vous pourriez faire encore plus simple ?
- Est-ce que quelqu'un n'a utilisé qu'une seule brique de 1x1 ?
- Est-ce que la maison aussi aurait pu être construite avec une seule brique de 1x1 ?

Refacto



Refacto - itération 1 - règles

- Vous allez construire une maison
- Une exigence toutes les 30s
- Refacto interdite = 2 pièces collées ne peuvent être décollées

Refacto - itération 1 - dev

- Commençons par 2 murs
- Puis ajoutons une fenêtre
- La maison doit être complètement fermée, ajouter des murs si besoin
- La maison doit avoir un toit en pente
- La maison doit avoir une cheminée
- La maison doit avoir une porte au rdc
- La maison doit avoir une 2ème fenêtre sur un autre mur
- Finalement la maison doit avoir un toit plat

Refacto - itération 1 - debrief

- Qu'est-ce qui s'est passé ? Comment vous êtes-vous senti ?
- Est-ce que vous étiez aussi rapide à construire la 1ère exigence que la dernière ? Pourquoi ?

Refacto - itération 2 - règles

- Vous allez construire une maison
- Une exigence toutes les 30s
- Refacto autorisée = 2 pièces collées peuvent être décollées

Refacto - itération 2 - dev

- La maison doit être de forme rectangulaire
- La maison doit avoir une fenêtre
- La maison doit être complètement fermée, ajouter des murs si besoin
- La maison doit avoir une porte au rdc
- La maison doit avoir un toit en pente
- La maison doit avoir une cheminée
- La maison doit avoir une 2ème fenêtre sur un autre mur
- Finalement la maison doit avoir un toit plat

Refacto - itération 2 - debrief

- Qu'est-ce qui s'est passé ? Comment vous vous êtes senti ?
- Est-ce que c'était plus facile de construire dans cette itération ?
- Si vous deviez continuer à construire, sur quelle base partiriez-vous ?
- Est-ce que c'était plus rapide de construire dans cette itération ?
- Est-ce que l'acte de refacto vous a ralenti ?

Intégration continue



Intégration continue - règles

- Nous allons transporter des cadeaux dans une contrée très lointaine :)
- 5 équipes :
 - 1 équipe pour les cadeaux,
 - 1 équipe pour les containers,
 - 1 équipe pour le camion,
 - 1 équipe pour la grue,
 - 1 équipe pour le bateau.
- 3 itérations

Intégration continue - itération 1 - dev

- 3 minutes pour construire

Intégration continue - itération 1 - rétro

- Est-ce que les cadeaux vont dans le container ? Est-ce que le container va dans le camion ? etc.
- Qu'est-ce qui s'est bien passé ? Que pouvez-vous améliorer pour la prochaine itération ?

Intégration continue - itération 2 - dev

- 1 minute pour planifier tous ensemble
- 3 minutes pour construire

Intégration continue - itération 2 - rétro

- Est-ce que les cadeaux vont dans le container ? Est-ce que le container va dans le camion ? etc.
- Qu'est-ce qui s'est bien passé ? Que pouvez-vous améliorer pour la prochaine itération ?

Intégration continue - itération 3 - dev

- 1 minute pour planifier tous ensemble
- 3 minutes pour construire

Intégration continue - débrief

- Qu'est-ce qui s'est passé ? Comment vous vous êtes senti ?
- A partir de quand avez-vous réussi à livrer des cadeaux ?
- Qu'est-ce que vous avez amélioré ?
- Avez-vous fait des tests régulièrement De quelle nature ?
- Intégration continue = collaboration + outils pour automatiser

A quoi reconnaît-on un super-méchant ?

- “aller steplé c’est juste une page en plus, tu sais là à côté du bouton qui fait tout”
- “j’ai pas trop le temps de rentrer dans les détails mais je te fais confiance, tu connais le produit”
- “il nous reste 2j avant la deadline, franchement les tests on verra après”
- “5j ? j’suis sûr.e que vous pouvez le faire en 3 ! Challenge accepté ?”
- “toujours en train de ranger toi hein ^^ Aller viens on fait juste un truc vite fait bien fait !”

etc...

Conclusion

Aider les dev à être des super-héroïnes et super-héros en soutenant leur travail

- partager et/ou animer ces ateliers auprès des personnes non dev de votre entreprise (manager / client.e.s / PO etc.)
- partager et/ou animer ces ateliers auprès des étudiants et étudiantes / apprenti.e.s / junior etc.
- si vous avez des fonctions d'organisation / management : prévoyez / budgétisez du temps pour que vos équipe de dev fassent du bon bouleau (pensé / testé / corrigé / adapté) et pas juste du code vite fait mal fait

Feedback



<https://roti.express/r/agn2023-65>