# Maintenance Évaluation 3

## Mon analyse technique

La fonction à ajouter dans mon programme de League of Legends serait d’implanter un bouton pour pouvoir recevoir des donations de la part des personnes qui consultent le site web. Pour ce faire, il faudra créer un nouvel onglet donation contenant :

* La première page contiendrait un court texte pour décrire le but du site web et pourquoi les gens devrait faire une donation. Au bas du petit texte informatif se trouverait un bouton « Faire une donation » qui nous amènerait à la prochaine page.
* La deuxième page serait un petit formulaire pour savoir sous quel nom la personne souhaite faire sa donation et combien et désire donner. Une fois tout cela remplit, l’utilisateur pourra cliquer sur le bouton donation de Paypal qui se trouvera au bas du formulaire.
* Un compte Paypal vous sera fournis pour recevoir les donations. Ensuite il ne reste qu’à intégrer un peu de code pour faire fonctionner le formulaire avec le bouton de notre compte Paypal.
* Je veux que si la transaction est effectuée avec succès, les détails soient sauvegardés dans une nouvelle table de la base de données. La table se nommera « donation ». Les données que je souhaite avoir dans cette table sont : l’id de la donation, le montant de la donation, le nom du donateur, la date et l’heure de la donation.
* Lorsque la transaction est terminée, l’utilisateur est retourné à l’écran d’accueil du site.
* Un fichier CSS sera fournis pour que les pages soient en harmonie avec les reste du site.

## Situation de refactorisation

Bonjour,

Il ne faut pas que tu écrives des classes beaucoup trop grosses, parce que même si toi tu te comprends dans ces classes gigantesques, les autres programmeurs qui vont passer sur ton code dans le futur ne seront certainement pas capable de se retrouver parmi toutes ces lignes. En faisant plusieurs petites classes avec des noms pertinents, il sera possible à n’importe qui voulant lire ton code de se retrouver rapidement.

Ensuite, pour l’encapsulation des données, un utilisateur n’a pas besoin de connaitre la structure des données dans l’objet pour pouvoir la modifier. Donc, pour ne pas qu’il puisse modifier directement les attributs de l’objet, on implante des méthodes comme get et set selon nos propres conditions. Cela nous garantis que l’utilisateur entrera des données conforme à ce que l’on souhaite avoir en bout de ligne dans notre application.

## Text TDD

Selon les articles que j’ai lus et selon mes expériences jusqu’à maintenant avec les TDD, je suis définitivement en désaccord avec cette pratique. Premièrement, faire des TDD prend énormément de temps. Ce sont des ressources investit dans quelque chose qui n’est pas nécessaire et qu’on peut très bien se passer. Ensuite, lorsque quelqu’un programme des tests, il est possible qu’il y est des erreurs dans les TDD et que les tests passent même si le code aussi contient des erreurs. Pire encore, si le code n’a pas d’erreurs et que les tests contiennent des erreurs, il se peut que les tests ne passent pas et nous allons perdre du temps à essayer de débuguer notre code qui, en réalité, est correct.

<http://www.agilerecord.com/test-driven-development-good-bad/>

<https://dzone.com/articles/whats-wrong-test-driven>