**TDD Reflection**

|  |  |
| --- | --- |
| Author | Lam Nguyen |
| Date | 02-01-2019 |

**My opinion about TDD:**

* **TDD is hard to apply:** Especially for me, a fresher, because I not only apply TDD, but I also have to learn about the technology, how to implement (higher priority, I think). So, before starting a new task, I trend to search on Google first how to code it first, instead of keeping calm and analyze requirement, how many functions we need, and cases to Unit tests.
* **TDD take more time to finish a task:** Instead of implementing the task, we firstly have to take a period to analyze, write test cases. TDD will make the time to implement a feature (task) become longer and the productivity is low (I think).
* **TDD make engineer understand requirement more clearly:** Definitely, more time we read the requirement, the better we understand it. And when we can write right cases to Unit tests, it means we understand the requirement clearly.
* **TDD make engineer to have a habit to write code testable:** When we have test cases already, we trend to write functions which are enable to test with these test cases.

**My questions about TDD:**

* How can we pre- define the type of exception we need in a program (feature)?
* Is there any rules supporting to list the cases used for Unit Test? 🡪 Based on experience, and should define test cases with other people.
* How can we verify test cases we wrote are right (acceptable) or not? 🡪 By reviewing, and experience.