

# SIT213 BE2

# IMT Atlantique

---



**IMT Atlantique**

Bretagne-Pays de la Loire  
École Mines-Télécom

**14 SEPTEMBRE**

---

Créé par :

LE DUC Elouan

MAQUIN Philippe

LE GRUIEC Clément

LE JEUNE Matthieu

FRAIGNAC Guillaume

---

## BE 2 : Test & Validation

Dans le cadre du projet SIT213, nous élaborons un simulateur. L'objet de la présente séance est de réfléchir à la problématique du test et de la validation de ce système, et de choisir une mise en œuvre.

### ➤ **Quelles parties peut-on vraiment tester avec un programme de test ?**

Le programme de test pourra uniquement tester la cohérence des résultats en les comparant avec un modèle théorique. C'est donc le code utile du programme qui sera réellement testé. Les valeurs analysées seront celles accessibles par des observateurs. Nous n'aurons donc pas toutes les valeurs prises par le programme à chaque instant.

### ➤ **Comment améliorer la testabilité ?**

Pour améliorer la testabilité, il est nécessaire de bien décrire dans la documentation les méthodes qui sont utilisées. Ainsi, le programme de test pourra prendre en compte un plus grand nombre de cas d'usage possible.

### ➤ **Comment faire du test de non-régression à chaque nouvelle itération du projet ?**

Il est primordial de vérifier à chaque itération que le nouveau code passe également les anciens tests. Sauf exception, des tests en échec seront considérées comme une régression.

### ➤ **Quelle confiance accorder aux résultats de simulation, et comment accroître cette confiance ?**

Le respect du cahier des charge par le simulateur est un critère de confiance nécessaire mais n'est pas suffisant. On peut accorder notre confiance dans le simulateur dans la mesure où ce dernier respecte le modèle théorique qui lui est associé. Néanmoins, cela nécessite de tester le simulateur avec en entrée les plages de valeurs qui sont définies dans le cahier des charges. Cela peut potentiellement exiger l'écriture de nouveaux tests. Ainsi, pour augmenter notre confiance dans notre programme de simulation, nous devons ajouter de nombreux tests au fil de son développement. Ces tests devront inclure le maximum de cas d'usage possible et couvrir le maximum de code possible.

---

➤ **Comment comparer des simulations lorsqu'elles reposent sur des comportements aléatoires (message, bruit, ...) ?**

Pour comparer de manière équitable il faut trouver une « seed » qui génère un message aléatoire que l'on jugera intéressant.

Il doit posséder tous les changements d'états (ex : 00, 01, 11, 10). Ensuite, il faut réutiliser cette « seed » pour chaque simulation. Ainsi on peut étudier les différences de comportements de la simulation entre les différentes étapes et avec différentes formes.

➤ **Comment organiser l'équipe spécifiquement pour les tests (p. ex. développeurs et testeurs différents) ?**

Les fonctionnalités développées par un membre de l'équipe seront d'abord testées par ce dernier. Dans un second temps, le code devra être testé rigoureusement par un ou plusieurs autres membres de l'équipe. Ainsi, cela imposera une redondance dans la vérification des méthodes développées. Dans l'idéal, cette procédure de tests devra être réalisée à chaque merge d'une sous branche dans la branche master du projet git.

## Conclusion

Au cours de ce BE nous avons pu évoquer un point important de notre projet, les tests. Nous avons pu échanger de manière constructive sur la manière de mener nos validations logicielles et sur le degré de confiance que l'on peut accorder à nos résultats. Nous en avons conclu qu'il faut nécessairement prendre en compte le plus de cas d'usage possibles et qu'il n'est pas possible d'accorder une confiance aveugle à nos tests.