# R2.03 : Aide à la SAE : Exception métier, Mob-Programming & Exemples

## **Exercice 1: Exception métier**



en **mode déconnecté** :

Laissez vos ordinateurs éteints pour le moment et prenez une feuille de papier et regroupezvous par groupe de SAE 😊

1. Réfléchir à (au moins) une exception métier que vous pourrez utiliser dans le code de votre SAE. Ecrire ci-dessous l'implémentation de cette(ces) exception(s)

### 2. Mettre en place une exception métier dans la SAE en mode « mobprogramming »

Connaissez-vous le mob-programming?

"Mob Programming is a software development approach where the whole team works on the same thing, at the same time, in the same space, and on the same computer. This extends the concept of pair programming from two people working together to the entire team continuously collaborating on a single computer to deliver a single work item at a time."

(Extrait: https://www.agilealliance.org/glossary/mob-programming)

En mob-programming, il n'y donc qu'une seule personne qui écrit le code : le driver son rôle est en quelque sorte le scribe, c'est la paire de mains qui fait le relais entre les idées qui sont partagées par l'équipe et l'ordinateur.



Vous pouvez maintenant passer en mode connecté 🗐

... mais en vous mettant en mode mob-programming :

- Un seul ordinateur sera utilisé par groupe de SAE et la personne qui sera au clavier de cet ordinateur sera le **driver**.
- Sur cet ordinateur, vous devrez disposer du code du dernier commit de votre SAE (si ce n'est pas le cas faire un pull depuis votre remote)

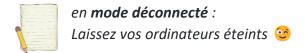
Regroupez-vous autour du driver pour mettre en place une exception métier dans votre SAE en respectant pas à pas les étapes suivantes :

⇒ Etape 1 : Implémentation de l'exception métier dans un sous-paquetage util(itaire)s du paquetage métier

# ⇒ Etape 2 : Déclencher l'exception « au bon endroit » et la propager - 2.1 Déclencher l'exception : Quelle instruction devez-vous utiliser pour déclencher une exception ? - 2.2 Propager exception en signanlant qu'une exception passe dans une méthode ② ...... A vous de trouver le bon endroit pour déclencher l'exception métier que vous venez d'écrire ....

⇒ Etape 3 : Traiter l'exception métier « au bon endroit »  Quelle instruction permet de traiter une exception ?
Quelle instruction permet de récupérer le message embarqué par une exception ?
⇒ Etape 4 : Une fois votre code fonctionnel, vous pouvez procéder à un commit avec un message explicite qui indique l'exception métier que vous venez d'ajouter.

... En attendant que tous les groupes aient mis en place une exception métier dans leur SAE, vous pouvez réfléchir et mettre en place d'autre(s) exception(s) métier dans votre SAE ...



### **Exercice 2: Recherche d'exemples**

Dans un développement logiciel, les exemples sont primordiaux pour développer un logiciel de qualité et ils interviennent aussi bien en amont qu'en aval de la phase d'implémentation.

En amont, la **recherche d'exemples** devrait, en principe faire partie **des ateliers** à mener dans une **phase d'analyse** pour faciliter une compréhension commune et non ambiguë des besoins entre les différentes parties prenantes du projet. Ils sont une aide précieuse lors de la **spécification** des besoins.

Ils permettent notamment d'illustrer, de mieux comprendre et clarifier les règles métiers et de mettre en évidence des « cas limites » d'utilisation.

⇒ En aval, une fois l'implémentation réalisées, les exemples pourront être implémentées sous forme de tests pour vérifier et valider le bon comportement du logiciel ⓒ

Rechercher des exemples est donc efficace si les exemples sont proposés, écrits et discutés de manière collaborative, vous allez donc faire cet exercice par groupe de SAE 😊

Quels exemples pourriez-vous proposer pour spécifier et vérifier la validité des règles métiers de votre SAE ?