

Vous répondez dans un fichier MonNom_4.txt en suivant le format de réponse attendu (le 4 correspond au numéro du sujet). Vous déposez le sujet sur la plateforme des devoirs dans BUT2 R3.03 SUJET 4.

Q1 Frontière EF/ENF dépendante du contexte

Considérez la fonctionnalité « envoyer une notification push au mobile de l'utilisateur lorsque son mot de passe est modifié ».

Tâches :

- (a) Montrez, en construisant deux contextes applicatifs très différents, comment cette même capacité peut être considérée comme EF dans un cas et comme ENF dans l'autre.
- (b) Pour chaque contexte, rédigez la (ou les) exigence(s) correspondante(s) avec un niveau de détail suffisant pour être testable.
- (c) Discutez l'impact de ce choix de classification sur l'organisation du backlog (User Stories).

Format de réponse attendu :

- Q1-a:
 - Contexte 1: ...
 - Contexte 2: ...
- Q1-b:
 - Exigence-contexte-1: ...
 - Exigence-contexte-2: ...
- Q1-c: liste à puces : chaque puce = un impact (backlog, etc.).

Q2 Refonte d'un backlog non INVEST

On vous fournit les trois User Stories suivantes pour une application de gestion de tâches :

- US1 : « En tant qu'utilisateur, je veux un tableau de bord complet afin d'avoir une vue globale. »
- US2 : « En tant qu'utilisateur, je veux que l'interface soit moderne afin d'avoir envie de l'utiliser. »
- US3 : « En tant qu'administrateur, je veux gérer tout le système afin de contrôler la plateforme. »

Tâches :

- (a) Analysez chacune de ces stories à la lumière de l'acronyme INVEST (au moins un argument par lettre non respectée).
- (b) Transformez ce backlog en un ensemble d'au moins six User Stories conformes à INVEST, en veillant à ce qu'elles soient indépendantes et testables.
- (c) Pour deux de vos nouvelles stories, proposez un jeu minimal mais suffisant de critères d'acceptation.

Format de réponse attendu :

- Q2-a: 3 lignes, par ex. - US1: probleme INVEST =
- Q2-b: liste à puces : US-A, US-B, etc.
- Q2-c: pour 2 stories, bloc :
 - Story X:
 - - Critere 1: ...
 - - Critere 2: ...

Q3 Boutique en ligne : gestion automatique des ruptures de stock

Considérez une boutique en ligne qui souhaite introduire le processus métier « Gestion automatique des ruptures de stock » : lorsqu'un produit est en rupture, le système doit empêcher toute nouvelle commande, proposer de s'inscrire à une alerte, et déclencher une commande fournisseur.

Tâches :

- (a) Décrivez ce processus métier sous forme de récit structuré (au moins 8 étapes), en précisant clairement le rôle de chaque acteur (Client, Employé logistique, Système).

- (b) À partir de ce processus, proposez les modifications à apporter au Diagramme de Classes du Domaine du e-commerce standard (nouvelles classes ou attributs nécessaires).
- (c) Définissez un scénario de test d'acceptation complet qui vérifie de bout en bout que le besoin métier initial (éviter les commandes sur produits indisponibles tout en capturant la demande) est satisfait.

Format de réponse attendu :

- Q3-a: liste numérotée (1..8).
- Q3-b: liste à puces (nouveaux attributs/classes).
- Q3-c: bloc texte Given ... When ... Then ... sur 3–5 lignes.

Q4 Complémentarité User Stories / Cas d'utilisation

Expliquez de façon argumentée la complémentarité entre User Stories et Cas d'Utilisation dans une démarche réelle de projet :

- (a) En prenant un exemple concret (au choix : e-commerce, InnovateBox, bibliothèque, etc.), montrez comment *une* User Story peut donner naissance à *plusieurs* Cas d'Utilisation.
- (b) Discutez les risques d'un projet qui n'utiliserait que des User Stories *sans* Cas d'Utilisation détaillés pour des fonctionnalités complexes.
- (c) Inversement, discutez les risques d'un projet qui n'utiliserait que des Cas d'Utilisation formels sans backlog de User Stories priorisées.

Format de réponse attendu :

- Q4-a: liste à puces : 1 story + liste de CU qui en découlent.
- Q4-b: paragraphe court.
- Q4-c: paragraphe court.