

Série de TD # 1

Exercice 1 : Questions de cours.

1. Citer les quatre principaux facteurs de qualité d'un logiciel. Proposer deux autres facteurs et dire dans quelles circonstances, ces deux derniers deviennent – ils plus importants que les autres ?
2. Expliquer pourquoi le modèle de cascade ne reflète pas exactement les activités de développement du logiciel.
3. Établir un tableau montrant pour chacune des étapes du modèle en V du cycle de vie des logiciels, les documents à délivrer en sortie.
4. Citer les points forts et les points faibles de chacun des modèles en V et à incréments.
5. Comparer le prototypage jetable par rapport au prototypage non jetable.
6. Montrer, pour chaque modèle de cycle de vie, les domaines d'application les plus appropriés.
7. Quelles sont les qualités requises dans un cahier de charges ?
8. Comment peut-on réduire l'écart entre les besoins réels d'un client et les besoins exprimés dans le cahier de charges ?

Exercice 2 : Choix d'une Solution Informatique

Une université voudrait s'équiper d'un système intégré de gestion des étudiants et qui prendrait en compte tous les détails concernant les étudiants y compris les informations personnelles, les cours suivis et les notes obtenues aux examens. Les trois approches possibles sont :

1. Acheter un système de gestion de bases de données et développer son propre système basé sur cet outil.
2. Acheter un système comparable à celui d'une autre université et le modifier pour ses propres besoins.
3. Se joindre à un groupe d'autres universités, établir un cahier des charges commun, contacter une société de logiciels qui développera un seul système pour tous.

Question : Identifier deux risques possibles pour chacune de ces stratégies et proposez des techniques de résolution de risque qui permettraient de décider quelle approche adopter ?

Exercice 3 : Difficulté de Spécifier un Besoin Fonctionnel

Soit la spécification de la règle de notation à un examen suivante :

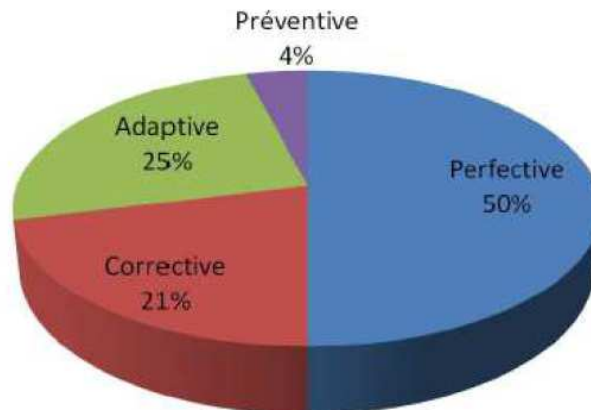
« L'examen est un ensemble de 20 questions à réponses multiples. Chaque bonne réponse à une question rapporte 1 point. Chaque mauvaise réponse fait perdre 1/3 de point. Chaque question sans réponse donne 0 point. »

Question : Pensez-vous que cette spécification est claire ? Afin de le vérifier, calculez chacun la note des 3 étudiants suivants : (Recensez les résultats possibles et Proposez une spécification plus précise).

	Réponse correcte	Incorrecte	Sans	Double réponse
Ahmed	10	5	5	
Nora	4	16		
Wissam	10	3	4	3 (1 juste, 1 faux)

Exercice 4 : Importance de la Maintenance du Logiciel

Ce schéma représente les variantes de maintenance logicielle existant sur le marché. En effet, après la mise en production ou exploitation d'un produit logiciel commence une phase très importante pour la longévité de ce dernier. Cette phase se nomme maintenance.



Questions :

1. D'après le schéma, quels sont les types de maintenance considérés et qu'elle est sa signification ?
2. Selon vous que représente le pourcentage associé à chaque type de maintenance ?
3. Sur un même produit logiciel, pensez-vous qu'il soit possible d'opérer plus d'un type de maintenance ?

Exercice 5 : Analyse d'un Système « La Pompe à insuline »

Une pompe à insuline est un appareil médical que l'on place sur une partie du corps d'un malade diabétique (bras, cuisse, jambe, ...) afin de contrôler la glycémie et injecter la quantité appropriée d'insuline dans le corps du patient de manière automatique. Les figures 1 et 2 illustrent respectivement un modèle de pompe ainsi que le principe général de son fonctionnement.



Figure 1 : La Pompe à Insuline



Figure 2 : Principe Général de Fonctionnement

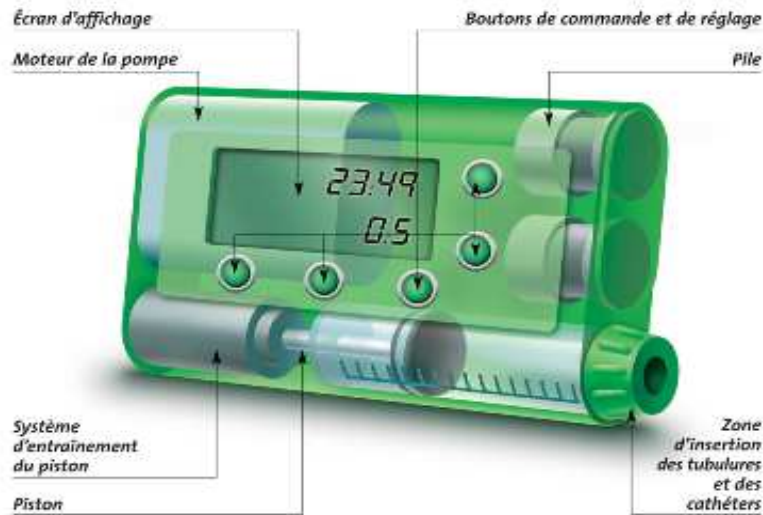


Figure 3 : Détail des Composants de la Pompe

Comme on peut le deviner, la pompe est un système composé d'une partie matérielle « Hardware » et d'une partie logicielle « Software » comme le montre la figure 3.

Questions :

1. D'après le schéma général de fonctionnement (figure1) et les détails (figure 3), quelles sont d'après vous les fonctions à offrir à l'utilisateur final, i-e, le malade ?
2. En termes de risques, donnez les risques majeurs qui pourraient mettre en péril la sécurité ou la vie du diabétique en justifiant.
3. D'après ce qui a été vu en cours, quel modèle de cycle de vie pourrait être le plus adapté pour le développement d'un tel système.
4. A quels domaines d'expertise doit-on faire appel pour la compréhension, la conception puis la programmation de l'algorithme de contrôle (figure 2).

Exercice 6 : Elaboration d'un Cahier des Charges

Imaginez-vous un moment dans la peau d'un informaticien « professionnel » qui a pour mission d'informatiser un cabinet médical. Une première phase importante est l'élaboration d'un cahier des charges qui fera office de contrat entre vous et le médecin « votre client ».

Question :

Expliquer les étapes par lesquelles vous passeriez pour accomplir la mission importante d'élaboration d'un cahier des charges, dont dépendront le succès et la suite du travail de développement.

Exercice 7 : Etude d'un cas d'échec « ARIANE 5 »

Le crash d'ARIANE 5 est le résultat d'une succession d'erreurs techniques et stratégiques. Le calculateur en cause était chargé de fournir des données au cœur du contrôle de vol. Il avait été développé pour Ariane 4, et avait effectué sans problèmes plusieurs vols.

Les erreurs de stratégies sont que le calculateur est repris tel quel sur Ariane 5, sans aucune reprise des spécifications, ni tests complémentaires. Malheureusement les données d'entrées d'ARIANE 5 sont totalement différentes de celles d'ARIANE 4. Ce qui aura pour conséquence de provoquer le bug qui entrainera la destruction du lanceur.

Première erreur technique :

Le code contient tout bêtement une affectation d'une donnée 64 bits vers une donnée 16 bits. Comme les données d'entrées d'ARIANE 5 sont plus importantes que prévu, il arrive qu'une valeur de plus de 16 bits soit affectée à la variable codée sur 16 bits. Une exception est alors logiquement levée.

Deuxième erreur technique :

En langage ADA, un mécanisme permet de récupérer ce type d'exception. Ici aucun mécanisme de protection n'est prévu. L'exception est transmise au cœur du contrôle de vol, qui la traite comme n'importe quelle autre valeur fournie par le calculateur. D'où son comportement aberrant, qui entraîne la perte de trajectoire, puis par conséquent le mécanisme d'autodestruction.

Un deuxième calculateur fonctionnait en redondance à chaud (fonctionnement en parallèle). Mais comme sa conception était en tout point identique au premier, les mêmes causes ont produit les mêmes effets.

Questions :

1. Recenser et classer les défaillances et les fautes conduisant à ces dernières dans ce cas.
2. Quelles leçons peut-on tirer de ce cas ?
3. Le développement de systèmes industriels ou de pointe est souvent complexe et demande beaucoup de temps pour sa réussite. Quels sont les modèles les mieux adaptés pour de tels projets et pourquoi ?