

Analyse

TD0 – Génie logiciel

[R3.03] Analyse

Exercice n°1 (*Généralités*)

1. Quelle est la différence entre un programme, un ensemble de programme et un logiciel ?
-
2. C'est quoi le génie logiciel ? Est-t-il nécessaire pour développer -
- Le génie logiciel c'est un ensemble de méthodes de travail de bonne pratiques de techniques et d'outils dédiés) la conception, le développement, le test et la maintenance des logiciels.
3. Nous avons vu en cours les activités principales dans un processus de d'enveloppement logiciel. Donnez-les entrées/sorties ainsi qu'une description de chaque activité.
 - a. **Spécification:**
DESC: Comprendre les besoins du client, les objectifs généraux, l'environnement du futur système, ressources dispo, contrainte de performance.
IN: Info fournies par le client.
OUT: Cahier des charges + manuel d'utilisation préliminaire
 - b. **Conception:**
DESC: Elaborer une solution concrète r »laissant la spécification. Description architecturale en composants. Réalisation des fonctionnalités par les composants. Réalisations des exigences non-fonctionnelles.

IN: Cahier des charges fonctionnel.

OUT: Dossier de conception.

c. **Programmation:**

DESC: Implantation de la solution conçue. Choix de l'environnement de développement du/des langages de promotion; de normes de développement...

IN: Dossier de conception.

OUT: Code documenté + manuel d'utilisateur

d. **Vérification & validation:**

DESC: Vérification: assurer que le logiciel satisfait sa spécification. Validation: assurer que les besoins du client sont satisfaits.

IN: Système.

OUT: Rapports de test, certificats, preuves ...

e. **Déploiement:**

DESC: Installation, formation, assistance.

IN: Code exécutable installable, manuels d'installation et d'utilisation.

OUT: Utilisation du logiciel, feedback utilisateurs validation.

f. **Maintenance:**

DESC: Installation, formation, assistance.

IN: nouveau besoins, bugs constatés, remarques générales, version actuelle.

OUT: nouvelle version, manuel d'utilisation à jour.

g. **Documentation:**

DESC: Une activité qui accompagne chaque étape.

4. Donnez la liste des qualités requises dans un cahier de charges.

a. Quelques éléments en complément du support de cour:

- Bon niveau de **généralité**.

- **Formulation** adéquate des besoins. Problème bien décrit.

- Etre **précis**, non ambigu malgré l'usage d'un langage

informel (*ou semi-formel*).

- Etre **complet** (*pas commission involontaire*).

- Etre **cohérent** (*Pas d'inférence de fonctionnalités*).
- Etre **vérifiable**: critères de validation définis. Evaluer la faisabilité des besoins. Faire éventuellement une maquette, une simulation.
- Etre **modifiable**: facilité à exprimer un changement ou ajout de besoin.

Exercice n°2 (*Critère de qualité*)

1. Donnez la liste des principaux facteurs de qualité d'un logiciel

- Validité
- Utilisabilité.
- Performance
- Fiabilité
- Sécurité
- Maintenabilité
- Portabilités
- Intéropérabilité.

2. Quel est le seuil minimal (*entre min, moyen et max*) à avoir pour chaque facteur des systèmes suivants :

| | Port. | Util. | Val. | Perf. | Séc. | Maint. |
|-----------------|-------|-------|------|-------|------|--------|
| Machine à laver | MAX | MAX | MIN | MIN | MIN | MIN |
| JV | MOY | MAX | MOY | MAX | MIN | MAX |
| Mail | MAX | MOY | MOY | MIN | MAX | MOY |
| BankApp | MAX | MAX | MOY | MIN | MAX | MAX |
| SimGen | MIN | MIN | MAX | MAX | MIN | MIN |

Exercice n°3 (*Cycle de vie*)

1. Reprenez le modèle en cascade vu en cours. Donnez les caractéristiques et les inconvénients d'un tel modèle.

Quelques éléments en complément du support de cours:

- Cycle de vie linéaire sans aucune évaluation entre le début du projet et la validation.
- Le projet est découpé en phases successives dans le temps.
- A chaque phase correspond une activité principale bien précise produisant un certain nombre de livrables.
- Chaque phase ne peut remettre en cause que la phase précédente.

Inconvénients:

- Les vrais projets suivent rarement un développement séquentiel.
- Etablir tous les besoins au début d'un projet est difficile.
- Aucune validation intermédiaire (*Aucune préparation des phases de vérification*)
- Sensibilité à l'arrivée de nouvelles exigences (*Refaire toutes les étapes*).

2. Reprenez maintenant le modèle en V et donnez les avantages et les inconvénients d'un tel modèle.

Quelques éléments en complément du support de cours:

- La préparation des dernières phases (validation-vérification) par les premières (construction du logiciel), permet d'éviter d'énoncer une propriété qu'il est impossible de vérifier objectivement après la réalisation.
- Idéal quand les besoins sont bien connus, quand l'analyse et la conception sont claires.

Inconvénient:

- Construit-on le bon logiciel ? Le logiciel est utilisé très (trop) tard.
- Il faut attendre longtemps pour savoir si on a construit le bon logiciel.
- Difficile d'impliquer les utilisateurs lorsqu'un utilisable n'est disponible qu'à la dernière phase.

3. Donnez une comparaison détaillée entre un prototypage jetable et un prototypage évolutif.

Quelques éléments en complément du support de cours:

Jetable:

- Cette approche consiste à produire rapidement une maquette (*prototype jetable*) qui constitue une ébauche du futur système.
- Avec cette approche, on est capable de définir plus explicitement et de manière plus cohérente les besoins des usagers, de détecter les fonctions manquantes et identifier et améliorer les fonctions complexes, comme elle permet de démontrer la faisabilité et l'utilité de l'application.
- Le prototype est ensuite abandonné et le système réel est construit.

Évolutif:

- Un prototype répond à des objectifs différents: on cherche à construire un système éventuellement très incomplet, mais dans son dimensionnement réel de façon à pouvoir faire des essais en vraie grandeur. On détermine d'abord un ensemble minimum de fonctionnalités-mêmes incomplètes de façon à réaliser un premier incrément du logiciel dont on se sert pour analyser le comportement du logiciel. L'utilisateur fournit en retour des informations à la conception qui motive le prototype. Ce processus d'évolution de prototypes continue jusqu'à ce que l'utilisateur soit satisfait du système livré.