

Chapitre 1 : Introduction

Partie 1 : Introduction :

I. Système d'Information (SI)

Un SI est un ensemble organisé de ressources (matériel, logiciel, personnel, données, procédures) qui permet d'acquérir, de traiter, de stocker et de diffuser des informations dans et entre des organisations.

II. Cycle de vie d'un logiciel

- Phase:
 1. **Identification des besoins** : Recueil des exigences à travers des interviews et analyses métier.
 2. **Analyse** : Détermination de ce que le système doit être et comment il peut être utilisé, incluant la structure et les relations des éléments.
 3. **Conception** : créer de solutions techniques pour les descriptions définit lors de l'analyse uncluant les stuctures et les algorithmes.
 4. **Implémentation** : Programmation et réalisation technique.
 5. **Test** : Identification des défauts de conception ou de programmation.
 6. **Maintenance** :
 - **Corrective** : raitement des bugs.
 - **Évolutive** : Intégration de nouveaux besoins ou de changements technologiques.

Partie 1 : QCM:

1. Qu'est-ce qui compose un système d'information ?

- a) Seulement des ordinateurs et des logiciels.
- b) Des ordinateurs, des logiciels, des données, et des procédures.
- c) Uniquement des personnes et des données.
- d) Des données et des procédures uniquement.

2. Quelle étape du cycle de vie du logiciel sert à comprendre ce que le logiciel doit faire ?

- a) Conception
- b) Analyse
- c) Test
- d) Maintenance

3. Peux-tu donner un exemple de quelque chose qui fait partie d'un système d'information ?

- a) Un crayon et un papier
- b) Un logiciel de gestion de stock
- c) Une table et des chaises
- d) Une machine à café

4. Quelle est la différence entre la maintenance corrective et la maintenance évolutive ?

- a) La maintenance corrective change le design du logiciel, tandis que l'évolutive corrige des bugs.
- b) La maintenance corrective ajoute de nouvelles fonctionnalités, tandis que l'évolutive corrige des bugs.
- c) La maintenance corrective corrige des bugs, tandis que l'évolutive ajoute de nouvelles fonctionnalités ou s'adapte à des changements technologiques.
- d) Il n'y a pas de différence; les deux termes désignent la même chose.

Partie 2 : La Modélisation :

I. Définition

- **C'est quoi ?** : La modélisation est l'acte de créer des modèles qui sont des représentations simplifiées de la réalité. Elle aide à comprendre, spécifier, et documenter des systèmes complexes en excluant les détails inutiles pour se concentrer sur l'essentiel.

- **Utilité :**
 - Spécifier et valider le système.
 - Guider la construction du système.
 - Organiser les données et le comportement du système.
 - Documenter le système.
 - Maîtriser la complexité et assurer la cohérence.
 - Réduire les coûts et les délais.

II. Vues d'un Système d'Information

- **Axe fonctionnel** (Vue utilisateur) : Décrit le fonctionnement et les besoins du système.
- **Axe statique** (Vue structurelle) : Décrit la structure et les éléments statiques du système.
- **Axe dynamique** (Vue comportementale) : Décrit les changements d'état et les comportements dynamiques du système.

Exemple

- Système à modéliser : voiture miniature de collection
 - Vue de la carrosserie et des versions de châssis.
 - Dessin technique des châssis.

III. Langages de Modélisation

- **Langages formels** : Utilisent des mathématiques pour des preuves formelles (ex : Z, B, VDM).
- **Langages semi-formels** : Plus graphiques et faciles à utiliser (ex : MERISE, UML).

Approches :

- **Cartésiennes** : Décomposent les fonctions jusqu'à obtenir des fonctions simples.
- **Systémiques** : Combinaison de structure et comportement, modélisation des données et traitements.
- **Objet** : Vision du système comme un ensemble d'objets interagissant, avec une grande cohérence entre objets et comportements.

Partie 2 : QCM

1. Quel le but principal de la modélisation ?

- a) Augmenter la complexité des systèmes.
- b) Réduire les coûts et délais de développement.
- c) Ignorer les besoins des utilisateurs.
- d) Augmenter les détails inutiles.

2. Qu'est-ce qu'une vue fonctionnelle dans la modélisation ?

- a) Description des changements d'état du système.
- b) Description des éléments et de la structure du système.
- c) Description du fonctionnement du système et des besoins.
- d) Description des méthodes de programmation.

3. Quel langage de modélisation est considéré comme semi-formel ?

- a) Z
- b) UML
- c) VDM
- d) B

4. Quelle approche de modélisation considère le système comme un ensemble d'objets ?

- a) Cartésienne
- b) Systémique
- c) Objet
- d) Aucune des précédentes

Chapitre 2 : Analyse Fonctionnelle

But de l'Analyse :

- **Vue Utilisateur :** Explique comment le système doit fonctionner selon ce que les utilisateurs veulent.
- **Objectif :** Créer une application qui fait ce que les utilisateurs attendent tout en respectant les règles ou limitations.

Diagramme de Cas d'Utilisation (DCU)

C'est un diagramme qui montre comment les utilisateurs interagissent avec le système. Il montre les principales actions que le système peut faire et ses limites.

- **Pourquoi c'est important ? :** Il aide à comprendre ce que les utilisateurs attendent du système sans entrer dans les détails techniques.

Éléments d'un DCU

- **Acteurs :** Ce sont les utilisateurs ou les choses qui utilisent le système. Ils peuvent être des personnes ou des objets comme des machines.
 - **Principal :** Celui qui commence une action et voit le résultat.
 - **Secondaire :** Celui qui aide à finir une action sans la commencer.
 - **Héritage :** Un acteur général peut faire tout ce qu'un acteur plus spécifique peut faire.
- **Cas d'Utilisation :** Ce sont les actions principales que le système peut réaliser de début à fin.
 - **Inclusion :** Quand une action inclut toujours une autre action.
 - **Extension :** Quand une action peut parfois inclure une autre action.
 - **Généralisation :** Quand une action générale peut être détaillée en actions plus spécifiques.

Chapitre 3 : Analyse Dynamique

Objectifs

- **Vue Comportementale** : Examine comment le système réagit et change en réponse aux actions des utilisateurs.

Diagramme de Séquences Système

- **But** : Montre les interactions entre les utilisateurs (acteurs) et le système, en suivant l'ordre dans lequel ces interactions se produisent.
- **Éléments Clés** :
 - **Objets et Acteurs** : Représentés par des rectangles, avec des lignes de vie qui descendent verticalement.
 - **Messages** : Échanges d'informations, représentés par des flèches horizontales.
 - **Fragments Combinés** : Utilisés pour montrer des interactions plus complexes, comme des conditions ou des boucles, avec des opérateurs spéciaux (ex : loop, alt, opt).

Diagramme d'Activités

- **But** : Décrit les étapes d'un processus ou d'une opération, montrant le flux de contrôle de l'activité à l'autre.
- **Éléments Clés** :
 - **Activités** : Représentées par des rectangles arrondis.
 - **Transitions** : Flèches qui montrent le passage d'une activité à l'autre.
 - **Nœuds de Contrôle** : Points qui coordonnent le flux, incluant des nœuds de décision, de fusion, et de bifurcation.
 - **Swimlanes** : Divisions du diagramme qui montrent quelles parties de l'organisation sont responsables de certaines actions.

Chapitre 4 : Analyse Statique

Objectifs

- **Vue Statique** : Examine la structure interne du système sans considérer le facteur temporel.

Diagramme de Classes

- **Utilité** : Montre comment le système est structuré à travers des classes qui définissent des objets avec des attributs (caractéristiques) et des opérations (actions).
- **Phases** :
 - **Analyse** : Description préliminaire des entités avec quelques attributs et méthodes.
 - **Conception** : Description complète avec tous les détails des attributs et méthodes.

Éléments d'un Diagramme de Classes

- **Classe** : Un modèle pour les objets, peut représenter des choses concrètes (comme des avions) ou abstraites (comme des commandes).
- **Attributs** : Caractéristiques d'une classe, avec visibilité indiquée (public, privé, protégé, package).
- **Opérations** : Fonctions ou méthodes que les objets de la classe peuvent exécuter.

Associations

- **Définition** : Relations entre deux ou plusieurs classes.
- **Propriétés** : Incluent le nom, la visibilité, et la multiplicité des relations.
- **Types** :
 - **Agrégation** : Relation tout/partie où des parties peuvent exister séparément du tout.
 - **Composition** : Forme plus forte d'agrégation où les parties ne peuvent pas exister séparément du tout.

Héritage

- **Concept** : Permet à des classes dérivées (subclasses) de recevoir les attributs et opérations de leurs classes de base (superclasses), simplifiant la réutilisation de code et la spécialisation des objets.

Questions pour QCM

1. Quel est le but principal du diagramme de séquences ?

- a) Représenter la structure statique du système.
- b) Visualiser le comportement interactif du système.
- c) Décrire les attributs des classes.
- d) Représenter les bases de données.

2. Dans un diagramme de classes, que symbolise une flèche avec un triangle plein à son extrémité ?

- a) Une association
- b) Une agrégation
- c) Un héritage
- d) Une composition

3. Qu'est-ce qu'une agrégation dans un diagramme de classes ?

- a) Une relation où les objets enfants peuvent exister indépendamment de l'objet parent.
- b) Une relation qui indique l'utilisation d'une classe par une autre.
- c) Une relation où la destruction de l'objet parent entraîne la destruction des objets enfants.
- d) Une représentation des méthodes d'une classe.

4. Quelle est la différence entre agrégation et composition ?

- a) L'agrégation permet une indépendance entre les objets, tandis que la composition indique une dépendance forte.
- b) L'agrégation et la composition sont des termes interchangeables en UML.
- c) La composition est utilisée pour les interfaces, tandis que l'agrégation est utilisée pour les classes abstraites.
- d) Aucune différence; les deux représentent des relations d'héritage.

5. Qu'est-ce qu'un diagramme d'activités en UML ?

- a) Il montre les interactions entre les classes.
- b) Il décrit le flux de contrôle d'une activité à l'autre.
- c) Il représente uniquement les relations d'héritage.
- d) Il détaille les opérations et les attributs d'une classe.

Exercice Pratique

Cas pratique : Vous travaillez sur un système de gestion de bibliothèque. Vous avez les éléments suivants à modéliser :

- Livre
- Bibliothèque
- Copie du Livre (chaque livre peut avoir plusieurs copies)

Relations à déterminer :

1. Entre Livre et Copie du Livre.
2. Entre Bibliothèque et Livre.

Question : Pour chacune des relations ci-dessus, indiquez s'il s'agit d'une association (agrégation ou composition) ou d'un héritage, et justifiez votre réponse.