



Filière :	Développement des Systèmes d'Information (DSI)
Épreuve :	Conception des Applications Informatiques (CAI)

Durée :	4 heures
Coefficient :	50

CONSIGNES

- ✓ Le sujet comporte 3 dossiers ;
- ✓ Chaque dossier doit être traité dans une feuille séparée.

Barème de notation

Dossier 1 : Étude de projet de la société	12 points
Dossier 2 : L'approvisionnement des magasins & stock	14 points
Dossier 3 : Gestion des ventes	14 points
Total	40 points

- ✓ Il sera pris en considération la qualité de la rédaction lors de la correction.
- ✓ Aucun document n'est autorisé.

ÉTUDE DE CAS : « ENFANT MALIN » pour accessoires d'enfants et jouets

La société « **Enfant Malin** », spécialisée dans la vente et la distribution des accessoires et jouets pour enfants, souhaite renouveler sa stratégie commerciale en développant un réseau de magasins et une centrale d'achat. Ces magasins portent le même sigle et réalisent la vente en détails des produits. La centrale d'achat est considérée comme seul fournisseur aux différents magasins du réseau.

DOSSIER I : ÉTUDE DU PROJET DE LA SOCIÉTÉ

(12 pts)

On souhaite mettre en place un logiciel permettant la gestion du siège administratif, de la centrale d'achat et des magasins. Le projet de réalisation de cette application est composé des tâches suivantes :

Lot	Tâche	Signification	Durée (jours)	Prédécesseurs
Lot 1	A	Conception du module siège administratif	20	-
	B	Élaboration des jeux de test	30	-
	C	Programmation du module siège administratif	30	A
	D	Test du module siège administratif	10	B, C
Lot 2	E	Conception du module centrale d'achat	40	-
	F	Élaboration des jeux de test	40	-
	G	Programmation du module centrale d'achat	25	E
	H	Test du module « centrale d'achat »	20	F, G
Lot 3	I	Conception du module magasin	15	-
	J	Élaboration des jeux de test	22	-
	K	Programmation du module magasin	15	I
	L	Test du module magasin	08	J, K
Lot 4	M	Intégration	15	D, H, L
Lot 5	N	Validation	07	M

1. QUESTIONS PRÉLIMINAIRES

- 1 - 1. Définir la maîtrise d'ouvrage dans un projet informatique ? (1 pt)
- 1 - 2. Quelle est la différence entre le test unitaire et le test d'intégration ? (1 pt)
- 1 - 3. En quelle(s) étape(s), se trouvent les tests (*unitaire et d'intégration*) dans le cycle de vie en cascade ? (1 pt)

2. PLANIFICATION DU PROJET

- 2 - 1. Établir le réseau PERT en se basant sur la liste des tâches citées précédemment. (1 pt)
- 2 - 2. Déduire la durée minimale du projet. (0,5 pt)
- 2 - 3. Représenter pour les tâches (*D et K*), sur le tableau suivant, les dates au plutôt et au plus tard ainsi que les marges totale et libre. (2 pts)

Tâche	Au plutôt		Au plus tard		Marge totale	Marge Libre
	Début	Fin	Début	Fin		
D						
K						

- 2 - 4. Donner la liste des tâches composant le **chemin critique**. (0,5 pt)
- 2 - 5. Élaborer le diagramme de **GANTT** de ce projet sachant qu'ils existent deux personnes pouvant effectuer toutes les tâches possibles et travaillant en plein temps. (1,5 pt)
- 2 - 6. Quelle est la durée effective du projet ? (0,5 pt)

3. ESTIMATION DES CHARGES

Le tableau suivant représente l'estimation des charges correspondant au code source à produire pour chaque lot.

LOT	NOMBRE DE LIGNE EN (ISL)
Lot 1	1000
Lot 2	1250
Lot 3	500

- 3 - 1. Calculer la charge (*en Mois Homme*) de la programmation de chacun des 3 lots (*Lot1, Lot2 et Lot3*) en utilisant la méthode **COCOMO** : (1 pt)

TYPE DE PROJET	CHARGE (mois homme)	DURÉE (mois)
Simple (<50 000 lignes)	$C = 3,2 (Kisl)^{1,05}$	$D = 2,5 (C)^{0,38}$
Moyen (<300 000 lignes)	$C = 3 (Kisl)^{1,12}$	$D = 2,5 (C)^{0,35}$
Complexe ($\geq 300 000$ lignes)	$C = 2,8 (Kisl)^{1,2}$	$D = 2,5 (C)^{0,32}$

Kisl : nombre de lignes de code divisé par 1000 (kilo ligne de codes).

- 3 - 2. Calculer la charge des autres étapes de chacun des 3 Lots (*Lot1, Lot2 et Lot3*) en utilisant le tableau des ratios suivant : (1 pt)

ÉTAPES	RATIOS
Conception	10% de la charge total du projet
Jeu de tests	30% de la charge de la programmation
Programmation	60% de la charge total du projet
Test	20% de la charge de la programmation

3 - 3. Calculer la charge de l'intégration et de la validation de tout le projet (*Lot4 et Lot5*), sachant qu'elle constitue 25% des charges de la programmation. (0,5 pt)

3 - 4. Calculer la charge totale du projet. (0,5 pt)

DOSSIER II : L'APPROVISIONNEMENT DES MAGASINS & STOCK (14 pts)

❖ L'APPROVISIONNEMENT DES MAGASINS :

La société « **Enfant Malin** » souhaite mettre en place une solution informatique assurant les fonctionnalités suivantes :

- Chaque utilisateur doit s'identifier et s'authentifier pour accéder à l'application avec les droits et les privilèges autorisés.
- Le **responsable des ventes** élabore les commandes et les envoie à la **centrale d'achat**, après validation par le **directeur du magasin**.
- Lors de la création d'une ligne de commande, le **responsable des ventes** peut consulter le **catalogue** émis par la **centrale d'achat**, afin de récupérer les informations d'un produit puis mentionner la quantité à commander.
- L'état du stock de chaque magasin est actualisé après la réception des commandes ou en cas de restructuration des produits vendus par le **responsable des ventes**.
- Le **comptable**, à partir de son compte, peut exporter les données comptables vers un autre logiciel conçu pour la comptabilité.
- Le **directeur du magasin** peut, dans des cas occasionnels, assurer toutes les tâches du responsable des ventes.

1. Élaborer le diagramme de cas d'utilisation correspondant. (4 pts)

2. La spécification du scénario nominal de « *élaborer une commande* » est décrit par :

« Pour créer une commande, le responsable des ventes doit s'identifier et s'authentifier. Puis, il indique le numéro et la date de la commande. Le système lui propose une liste de produits disponibles. Le responsable des ventes effectue son choix et précise la quantité. Cette opération de choix est répétée jusqu'à la fin de la commande. Le système enregistre la commande et envoie une notification au directeur du magasin ».

Décrire ce cas d'utilisation par un diagramme de séquence. (3 pts)

3. Élaborer le diagramme d'états de transition de l'objet **commande**. (2 pts)

❖ GESTION DU STOCK :

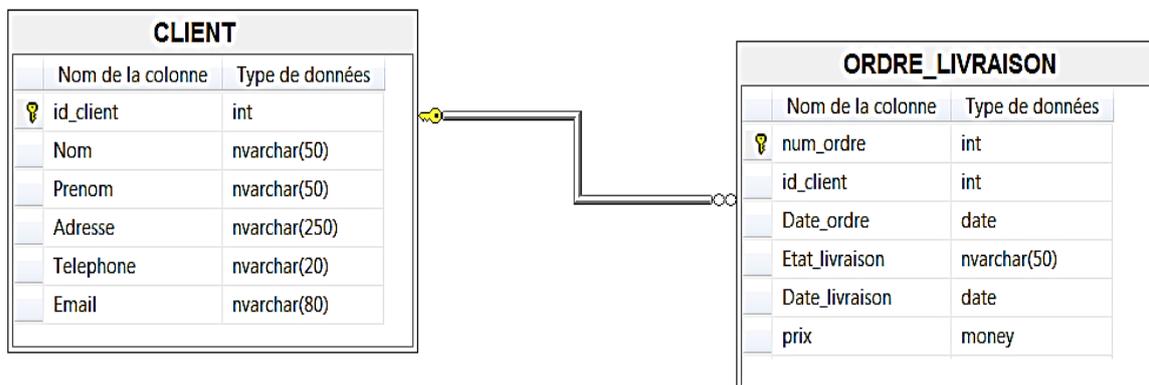
- ✓ Chaque **produit** peut être vendu en **pack** ou en **unité**. Il est caractérisé par un code unique, une description, un prix de vente, une quantité en stock et une marque.
- ✓ Un **produit en unité** peut être *accessoire* ou *jouet* est caractérisé par une catégorie d'âge.
- ✓ un **accessoire**, en plus, est caractérisé par une taille, tandis que le **jouet** possède un type.
- ✓ Un **pack** est composé d'autres **produits en unité** et caractérisé par un prix promotionnel, une date de début et une date de fin.
- ✓ La quantité en stock de chaque produit doit être augmentée ou diminuée selon l'opération effectuée (*vente, création d'un pack, ...*).

- ✓ La création d'un pack se déroule de la façon suivante :
- Nommer le pack.
 - Spécifier la quantité de pack à assembler.
 - Spécifier la période de validation du pack.
 - Indiquer les produits composants le pack, chacun avec sa quantité.
 - Spécifier le prix de vente promotionnel du pack en se basant sur une réduction exprimée en pourcentage.
4. Proposer un diagramme de classes comportant les classes et leurs associations, les attributs et les méthodes les plus importantes pour le système étudié (*gestion du stock*). (5 pts)

DOSSIER III : GESTION DES VENTES

(14 pts)

Le département commercial envoie les commandes payées aux clients sous forme d'ordre de livraison. Ces informations sont stockées dans une base de données, implémentée sous Microsoft SQL Server, dont le modèle relationnel est le suivant.



1. Donner le code Transact SQL permettant la création de la base de données selon les propriétés suivantes : (1,5 pt)

Nom de la base de données	Nom logique de fichier primaire	Nom logique du fichier journal	Chemin d'accès
BD_Ordre	Ordre_livraison	Ordre_livraison_log	E:\Srv_Bd\

2. Donner le code Transact SQL permettant la création des deux tables, en respectant les contraintes suivantes : (3 pts)
- Le champ « Id_Client » de la table « CLIENT » est un identifiant qui s'incrémente automatiquement. (Valeur initiale = 10 ; le pas =10).
 - Les champs « Nom » et « Prenom » de la table « CLIENT » ne peuvent pas être nuls.
 - Le champ « Etat_Livraison » ne peut prendre que l'une des valeurs suivantes : *En cours, Livré, Annulé*.
 - Le champ « Id_Client » de la table « ORDRE_LIVRAISON » est une clé étrangère qui fait référence au champ « Id_Client » de la table « CLIENT ».

3. Donner le code Transact SQL permettant la création d'une procédure qui reçoit en paramètre l'identifiant du client et affiche la liste de ses ordres de livraison. Si l'identifiant n'existe pas, un message d'erreur sera affiché. (2 pts)

```
CREATE PROC Sp_liste_ordres (@id_clt INT)
--(corps de la procédure)
```

4. Donner le code Transact SQL permettant la création d'une fonction scalaire qui retourne le nombre d'ordre de livraison pour un client dont l'identifiant est passé en paramètre. (2 pts)

```
FUNCTION fn_nb_Livraison(@id_clt INT) RETURNS INT
--(corps de la Fonction)
```

5. On souhaite automatiser la contrainte suivante : Dans la table « **ORDRE_LIVRAISON** », la valeur du champ « **Date_Livraison** » doit être ultérieure (Après) à celle du champ « **Date_ordre** ».

- a) Décrire en quelques lignes une solution technique pour automatiser la mise en œuvre de cette contrainte dans le contexte d'un système de gestion de bases de données relationnelles. (0,5 pt)
- b) Donner le code de la solution proposée. (1 pt)

6. Donner le code Transact SQL permettant la création d'une fonction qui retourne, pour une année donnée en argument, une table affichant : le client, le nombre d'ordre et le montant total de ces ordres. (2 pts)

Exemple : pour l'année 2017 on obtient :

ID_Client	Nombre ordres	Montant total
120	15	25 325.50
...

```
CREATE FUNCTION Fn_Resultat(@annee INT) RETURNS TABLE AS
--(corps de la Fonction)
```

7. Donner le code Transact SQL permettant la création du trigger « **Tr_Verifier** » qui permet d'interdire la suppression d'un client qui a au moins un ordre de livraison. (2 pts)