

E4D : ETUDE DE CAS**Durée : 5 heures****Coefficient : 5****CAS SECOLOG**

*Ce sujet comporte 11 pages dont 2 pages d'annexes.
Le candidat est invité à vérifier qu'il est en possession d'un sujet complet.*

Matériels et documents autorisés

- Lexique SQL sans commentaire ni exemple d'utilisation des instructions.
- Règle à dessiner les symboles informatiques.
- Calculatrice : Toutes les calculatrices de poche, y compris les calculatrices programmables, alphanumériques ou à écran graphique sont autorisées pour cette épreuve à condition que leur fonctionnement soit autonome et qu'il ne soit pas fait usage d'imprimante (C. n° 99-186 du 16-11-1999).

Liste des annexes

- Annexe 1* Schéma conceptuel du domaine « Gestion des propositions commerciales »
Annexe 2 Schéma relationnel du domaine « Gestion des formations »
Annexe 3 Documents comptables (exercice 2000)

Barème

Dossier 1	Gestion des développements	4,5 points
Dossier 2	Gestion des propositions commerciales	2,5 points
Dossier 3	Gestion des formations	4 points
Dossier 4	Gestion des sessions de formation	6 points
Dossier 5	Équipement des salles de formation	3 points
	Total	20 points

Le contexte de l'étude

SECOLOG, Société Européenne de Conception de Logiciel, est une SSII¹ spécialisée dans le développement d'applications de gestion. Solidement implantée au niveau régional, elle regroupe une quinzaine de personnes occupant des bureaux situés dans un immeuble récent. L'activité de l'entreprise est centrée autour de trois pôles d'activité :

- Le développement d'applications.
- Le support technique apporté aux clients.
- La formation des utilisateurs.

L'entreprise emploie six développeurs, un commercial, deux techniciens, une secrétaire polyvalente et deux formateurs. L'encadrement est assuré par deux chefs de projet et le directeur, fondateur de la société.

Dossier 1 : Gestion des développements

L'existant

L'entreprise utilise déjà l'outil informatique pour gérer une partie de son activité. C'est le cas pour la paie, la comptabilité générale et la gestion de clientèle, domaines pour lesquels SECOLOG utilise ses propres logiciels.

L'entreprise souhaite étendre l'informatisation de son système d'information au domaine qui touche à ses activités de développement.

Les besoins de l'entreprise

- **Organisation d'un développement**

Les demandes des clients de SECOLOG portent soit sur des logiciels standards, soit sur des développements spécifiques (c'est-à-dire des applications adaptées à des besoins particuliers du client). La présente étude ne couvre que la gestion de ces développements spécifiques.

Lorsqu'un client sollicite une étude pour la réalisation d'un développement spécifique, on crée un dossier appelé « affaire » : chaque affaire est identifiée par un numéro et caractérisée par une date, les coordonnées du client (code, nom, adresse, téléphone et adresse électronique), un descriptif sommaire et le code d'état de l'affaire. En effet, au cours de son traitement, l'affaire passera par différents états : "en étude", "en cours de réalisation", "en attente de règlement", "terminée", "abandonnée", ...

Si le client accepte la proposition commerciale de SECOLOG, un découpage de l'application à développer est effectué. Chaque partie constitue un dossier technique dont le suivi est confié à un chef de projet. Chacun de ces dossiers techniques est caractérisé par un numéro et un nom. Afin de faciliter le suivi du dossier technique, on souhaite conserver la trace des dates prévues pour le début et la fin de la réalisation, ainsi que les dates effectives correspondantes.

Chez SECOLOG, les employés sont décrits par un matricule qui les identifie, un nom, un prénom et une date d'embauche. Pour le présent domaine d'étude, seuls les deux chefs de projets et les développeurs d'applications sont concernés. Un chef de projet est responsable d'un certain nombre de dossiers techniques qu'il découpe en différentes tâches. Chaque tâche est numérotée séquentiellement par dossier technique, possède un nom mnémotechnique ainsi qu'une spécification technique et une date limite de réalisation. Chaque tâche est confiée pour exécution à un développeur.

Le travail des développeurs consiste alors à développer des modules. Tous les modules développés sont catalogués et les informations les concernant sont stockées pour être consultées ultérieurement. On souhaite connaître, pour chacun des modules, son numéro, son nom, le langage dans lequel il a été développé, son type (EXE, DLL, ActiveX, Java Bean, ...) et sa description technique.

¹ Société de service en ingénierie informatique

ISE4D

- **Suivi des développements**

Chaque module est sous la responsabilité d'un développeur. Un module donne lieu à plusieurs versions, caractérisées par un numéro de version relatif au module concerné, une date de fin de développement et une description de la version.

Une tâche est ainsi constituée de plusieurs versions de module et une même version de module peut être associée à plusieurs tâches.

Pour suivre et contrôler l'activité des développeurs, l'application mémorise le contenu des comptes rendus effectués quotidiennement par les développeurs. Pour des raisons de délais, plusieurs développeurs peuvent intervenir sur une version de module. Il est donc nécessaire de connaître le temps passé par version de module, par développeur et par jour de travail, ainsi que les remarques concernant le travail réalisé.

- **Validation des travaux**

Quand une tâche est achevée, elle donne lieu à une validation interne dont on retient la date effective.

Quand toutes les tâches relatives à un dossier technique sont validées, le chef de projet organise avec le client une réunion de présentation technique. Lors de cette rencontre, la partie développée est soumise à la critique du client qui acceptera ou refusera le dossier technique. On a donc parfois besoin d'effectuer plusieurs présentations avant qu'un dossier technique ne soit accepté définitivement. Le système d'information doit conserver une trace de ces présentations (date de présentation, avis du client, description globale des remarques formulées).

TRAVAIL A FAIRE :

À partir des règles de gestion énoncées ci-dessus, donner une représentation conceptuelle des données du domaine « Gestion des développements » en utilisant le formalisme du modèle entité-association.
--

Dossier 2 : Gestion des propositions commerciales**Annexe à utiliser : annexe 1**

M. DUBY, le commercial de la SECOLOG, démarché une clientèle potentielle que des études de marché confiées à des sociétés spécialisées ont permis de caractériser. Pour M. DUBY, cette clientèle est composée de « prospects ».

Lorsqu'il prend contact avec un prospect, M. DUBY lui présente la gamme des produits de la société et évoque les possibilités de développement d'applications spécifiques.

M. DUBY mémorise la visite, même si le prospect n'a pas manifesté d'intérêt pour les produits de la société.

Si le prospect est intéressé, M. DUBY lui propose de réaliser une étude : une nouvelle proposition commerciale est créée dont l'état est "à l'étude".

La proposition est étudiée dans le détail : analyse du cahier des charges, des règles de gestion, des documents, ... En fonction du volume des traitements à réaliser, le commercial détermine de manière approximative le temps de réalisation et les coûts prévisibles. La proposition commerciale est alors soumise au responsable de SECOLOG pour accord : l'état de la proposition prend la valeur "à viser".

Quand l'accord du responsable est obtenu, après d'éventuelles retouches, la proposition commerciale est transmise au prospect. L'état de la proposition est alors "envoyée".

Si le prospect manifeste son accord en apposant sa signature sur le document, la proposition commerciale tient lieu de contrat. SECOLOG en accuse réception, le prospect devient naturellement un client portefeuille de l'entreprise (les visites et les propositions le concernant sont supprimées), les informations caractéristiques du contrat sont mémorisées. Après découpage de l'application à développer, la phase de réalisation peut commencer.

Régulièrement, les propositions envoyées aux prospects sont vérifiées. Si le prospect n'a pas répondu au bout de deux semaines, une relance est envoyée (état "relancée"). Au-delà d'un nouveau délai de deux semaines, la proposition sera considérée comme "sans suite".

TRAVAIL A FAIRE :

Donner une représentation conceptuelle des traitements correspondant au domaine « Gestion des propositions commerciales » en utilisant un formalisme de type « événement-résultat ».

Pour chaque opération, le candidat précisera :

- Les événements déclencheurs, le cas échéant, les règles d'émission et les résultats obtenus.
- Les « objets » concernés (entités ou associations du schéma présenté en **annexe 1**).
- Les types d'actions effectuées sur les objets : consultation, ajout, modification, suppression et les éventuels changements d'état que ces objets subissent.

Dossier 3 : Gestion des formations

Annexe à utiliser : annexe 2

Concernant la gestion des formations proposées par SECOLOG à ses clients, une étude a déjà été réalisée et a conduit à la production du schéma relationnel décrit dans l'**annexe 2**.

TRAVAIL A FAIRE :

Rédiger en langage SQL les requêtes correspondant aux besoins suivants :

- a) Créer la table SESSION en précisant les contraintes de clé primaire et de clé étrangère.
- b) Afficher les dates de session, les libellés de formation et le nom des produits étudiés au cours des sessions de formation suivies par le stagiaire numéro 911 pendant la période comprise entre le 1^{er} mars 2000 et le 15 mai 2000.
- c) Afficher le numéro des formateurs et le nombre de sessions de formation qu'ils ont assurées. Seuls les formateurs ayant assuré au moins 10 sessions de formation feront partie du résultat.
- d) Supprimer la deuxième session d'une formation dont le code est égal à 'F405'.
- e) Autoriser l'utilisateur DUPARC à insérer, mettre à jour et supprimer des lignes dans la table SALARIÉ.

Dossier 4 : Gestion des sessions de formation**Annexe à utiliser : annexe 2**

Pour gérer les accès à la base de données, l'entreprise utilise des curseurs générés à l'aide d'une classe *JeuEnregistrements* dont l'interface est décrite ci-dessous :

Classe JeuEnregistrements

Public

{méthodes}

procédure Initialiser(s : chaîne)

procédure Avancer

procédure Reculer

fonction Début : booléen

fonction Fin : booléen

procédure Fermer

fonction ExtraitValeur(s : chaîne) : variant

*{constructeur}**{avance dans le curseur d'une ligne}**{recule dans le curseur d'une ligne}**{indique si le début du curseur est atteint}**{indique si la fin du curseur est atteinte}**{ferme le curseur et libère les ressources}**{renvoie la valeur du champ de nom s}*

Fin classe

Remarques :

- Le type « variant » fait référence à un type générique pouvant contenir aussi bien des chaînes que des nombres. On utilise ce type car on ne sait pas à l'avance le type du champ stocké dans la base.
- Pour effectuer une concaténation de valeurs de type « chaîne » ou de type « variant », on peut utiliser l'opérateur « + ».

Utilisation d'un objet :

Pour instancier un objet de cette classe, on utilise le constructeur Initialiser(s : chaîne) qui réalise la connexion à la base et le chargement en mémoire du curseur défini dans l'instruction SQL passée en paramètre.

Par exemple :

TabPersonnel : JeuEnregistrements {déclaration de l'objet}

TabPersonnel.Initialiser ("SELECT * FROM Personnel ;") {instanciation de l'objet}

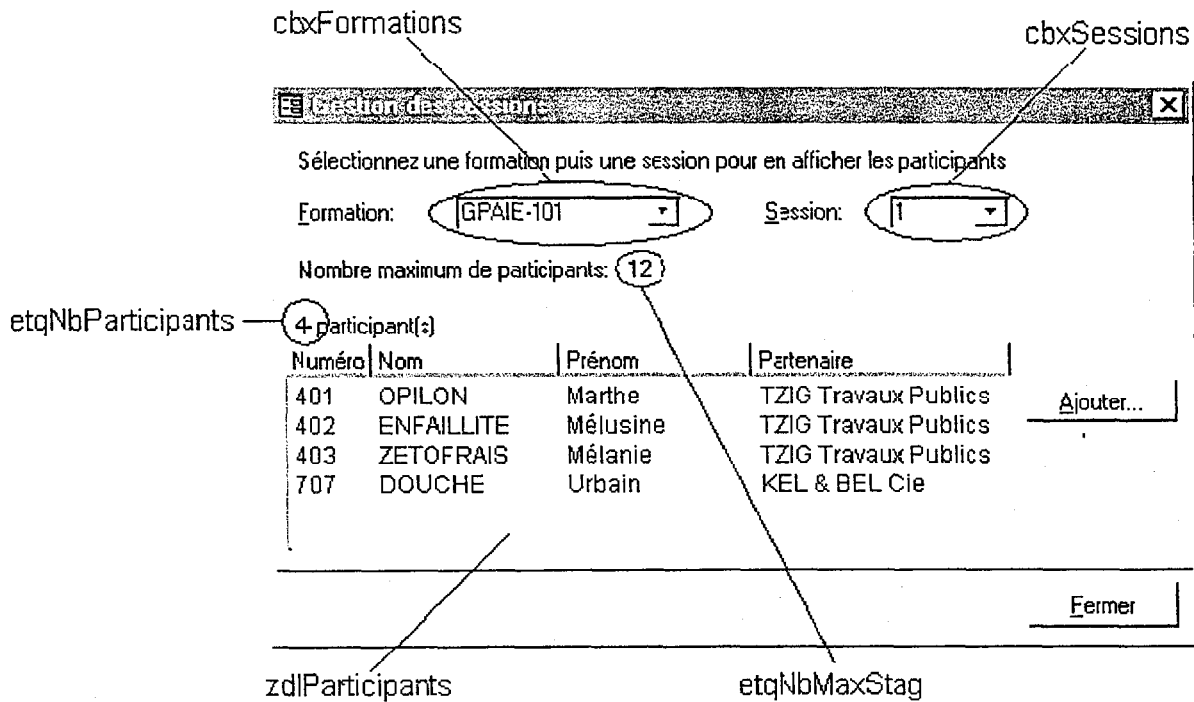
instancie l'objet TabPersonnel contenant tous les enregistrements de la table PERSONNEL.

Vnom_personne : Chaîne de caractères

Vnom_personne ← TabPersonnel.ExtraitValeur ("NomSalarié")

Récupération de la valeur de la colonne NomSalarié pour l'enregistrement courant

Lors du développement de l'application destinée à gérer les sessions de formation, la maquette suivante a été construite :



Les principaux objets graphiques utilisés sont :

- deux zones de listes déroulantes nommées respectivement *cbxFormations* et *cbxSessions*,
- une étiquette *etqNbParticipants* qui indique le nombre de participants à la session sélectionnée,
- une étiquette *etqNbMaxStag* qui renvoie le nombre maximum de stagiaires pour la formation sélectionnée,
- une zone de liste *zdlParticipants* contenant les participants à la session.

Comportement de la maquette :

- Le choix d'une formation met à jour les valeurs de la zone de liste déroulante *cbxSession* (toute formation a au moins une session), actualise le contenu de la zone de liste *zdlParticipants* et l'étiquette *etqNbMaxStag*.
- Le choix d'une session actualise la zone de liste *zdlParticipants* : événement ***cbxSessions_SurChangement()***.
- Les deux événements précités affectent le nombre de participants indiqués dans l'étiquette *etqNbParticipants*.

On donne les principales propriétés et méthodes de ces objets :

Pour les zones de listes (déroulantes ou non) :

- *Méthodes* :

procédure AjouteLigne (c1 : chaîne [; c2 : chaîne...])

Ajoute la chaîne passée en paramètre dans la liste. Si la zone de liste possède plusieurs colonnes, on peut passer une liste de chaînes séparées par des points virgules.

Par exemple, pour obtenir le résultat affiché dans la maquette présentée ci-dessus, on invoque la méthode de la manière suivante :

```
zdlParticipants.AjouteLigne("401";"Opilon";"Marthe";"TZIG Travaux publics")
```

Remarque : la donnée "TZIG Travaux publics" provient du champ *NomClient* de la table *CLIENT* dont dépend le stagiaire.

procédure **ViderListe**

Supprime en une opération toutes les lignes de la zone de liste.

- *Propriété* :

Valeur : variant

Renvoie la valeur sélectionnée dans la liste. Par exemple, la récupération de la valeur de la zone de liste déroulante *cbxFormations* :

```
var1 ← cbxFormations.Valeur
```

place la chaîne "GPAIE-101" dans la variable *var1*.

Remarque : cette valeur correspond à l'attribut *CodeForm*, clé primaire de la relation *FORMATION*.

Pour les étiquettes :

- *Propriété* :

Légende : variant

La légende représente le texte affiché dans l'objet.

TRAVAIL A FAIRE :

En s'aidant de la classe *JeuEnregistrements* et du schéma relationnel du domaine *Formations* donné en **annexe 2** :

4.1 Écrire l'algorithme de la fonction

Fonction *CompteParticipants* (Données *f* : chaîne; *s* : entier) : entier.

Cette fonction accède à la base de données pour compter le nombre de participants à une session de formation donnée (code formation = *f*, n° de session = *s*). La fonction retourne le nombre de participants donc la valeur zéro lorsqu'il n'y a aucun participant.

Le comptage peut être réalisé :

- soit à l'aide d'une requête SQL et d'une fonction SQL de comptage,
- soit à l'aide d'une requête SQL et de la classe *JeuEnregistrements*.

4.2 Écrire l'algorithme de la procédure événementielle associée au choix d'une session. La procédure sera identifiée par l'en-tête suivant :

Procédure *cbxSessions_SurChangement()*.

4.3 La fonction ***CompteParticipants*** étant utilisée dans plusieurs applications, il est envisagé de la stocker dans la base de données.

- Préciser le type de client-serveur auquel correspond cette nouvelle architecture applicative.
- Expliquer la démarche à suivre pour créer et utiliser un traitement stocké (procédure ou fonction) dans une base de données.

Dossier 5 : Équipement des salles de formation**Annexe à utiliser : annexe 3**

L'entreprise SECOLOG est actuellement équipée d'un réseau local comportant un serveur (nommé Serv01), cinq postes de travail dédiés au développement, deux postes de travail pour les commerciaux et deux postes de travail pour la gestion et l'administration.

L'entreprise veut proposer prochainement à ses clients un nouveau type de formation en libre-service ainsi qu'un système d'autoformation et de veille technologique à ses employés.

Dans cet objectif, elle a décidé d'installer dans ses locaux deux ateliers informatiques en réseau. Les salles qui accueilleront ces ateliers seront distantes l'une de l'autre d'une dizaine de mètres et d'environ 25 mètres du point d'accès au réseau téléphonique.

Le premier atelier, de nom AF-1, comprendra douze postes de travail et permettra aux stagiaires de se familiariser avec la manipulation d'outils bureautiques standards (le choix de la suite bureautique n'est pas encore fait) et d'outils Internet (navigateur, client de messagerie, client FTP, client de forums de discussion, IRC, ...).

Le second atelier, de nom AF-2, comprendra huit postes de travail et permettra aux utilisateurs de se former aux produits développés par l'entreprise, ceci par l'utilisation directe des produits mais aussi à l'aide d'un système de questions à choix multiples (QCM), accessible via un intranet. Toutefois, l'atelier AF-2 ne disposera pas d'accès à l'Internet.

L'entreprise vient d'acquérir un serveur auquel les postes des deux salles AF-1 et AF-2 seront connectés. Ce serveur (nommé Serv02), relié à un routeur, assurera les connexions à l'Internet.

Le comptable vous charge de préparer la mise à jour des documents comptables de synthèse pour l'année 2001 (exercice comptable du 1^{er} janvier au 31 décembre). Concernant le matériel informatique, la seule acquisition de l'exercice 2001 est le serveur Serv02 mis en service le 1^{er} avril 2001. Il a été facturé, installation comprise, 30 000 FRF hors taxes.

Le tableau des immobilisations et le tableau des amortissements du matériel informatique de l'entreprise pour l'exercice 2000 sont fournis en **annexe 3**.

TRAVAIL A FAIRE :

- 5.1** Sans entrer dans le détail de la configuration technique des postes et du serveur, rédiger une courte note sur le choix d'une architecture qui permettra de connecter les postes des ateliers AF-1 et AF-2 au serveur (topologie, câblage, connecteurs réseau, électronique active, systèmes d'exploitation, ...).
- 5.2** Présenter le plan d'amortissement linéaire du serveur Serv02.
- 5.3** À partir des documents comptables de l'exercice 2000 (**annexe 3**), présenter le tableau des immobilisations et le tableau des amortissements pour l'exercice 2001.
- 5.4** Présenter l'extrait de bilan au 31/12/2001 pour la ligne « matériel informatique ».

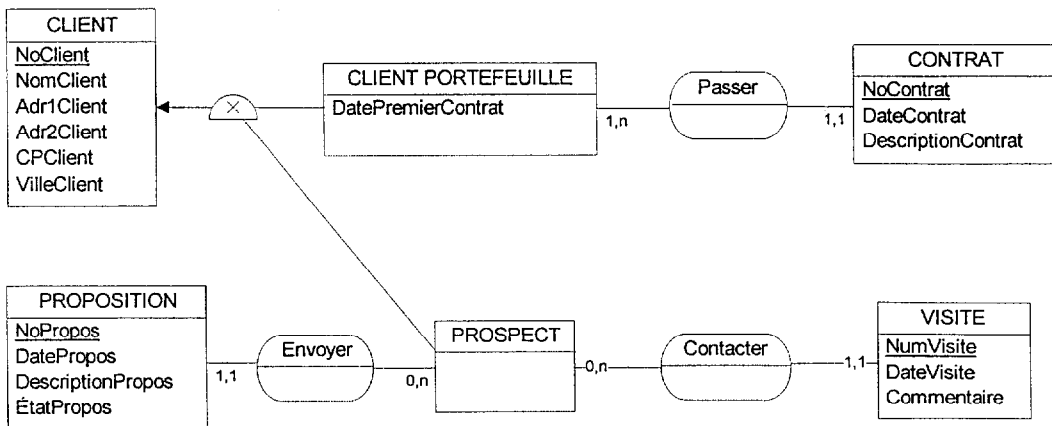


Schéma relationnel du domaine « Gestion des formations »

Annexe 2

SALARIÉ (Matricule, NomSalarié, PrénomSalarié, Adr1Salarié, Adr2Salarié, CPSalarié, VilleSalarié, DateEmbauche)

PRODUIT (CodeProduit, NomProduit)

SPÉCIALISER (Matricule#, CodeProduit#)

FORMATION (CodeForm, LibelléForm, NbMaxStagiaires, CodeProduit#)

SESSION (CodeForm#, NoSession, DateSession, Matricule#)

STAGIAIRE (NoStagiaire, NomStagiaire, PrénomStagiaire, NoClient#)

PARTICIPER (CodeForm#, NoSession#, NoStagiaire#)

CLIENT (NoClient, NomClient, Adr1Client, Adr2Client, CPClient, VilleClient)

Remarques :

- Les clés primaires sont soulignées et les clés étrangères sont suivies du caractère #.
- *NbMaxStagiaires* désigne le nombre maximum de stagiaires pour une formation.
- La clé étrangère *Matricule#* de la relation *SESSION* référence le formateur de la session.
- La relation *CLIENT* provient de l'entité *CLIENT* du domaine « Gestion des propositions commerciales » et représente donc les organisations en relation avec l'entreprise ; les stagiaires en formation sont issus de ces entreprises.

Tableau des immobilisations de l'exercice 2000 : Matériel informatique

Éléments	Valeur brute à l'ouverture de l'exercice	Augmentations	Diminutions	Valeur brute à la clôture de l'exercice
Serveurs	150 000			150 000
Postes de travail	180 000	40 000	21 400	198 600
Imprimantes	25 000	35 000		60 000
Scanners		5 400		5 400
TOTAL	355 000	80 400	21 400	414 000

Remarque :

À la fin de l'exercice 2000, tous les éléments du matériel informatique ont moins de 2 ans.

Tableau des amortissements de l'exercice 2000 : Matériel informatique

Éléments	Amortissements cumulés au début de l'exercice	Augmentations	Diminutions	Amortissements cumulés à la fin de l'exercice
Serveurs	40 000	50 000		90 000
Postes de travail	72 000	68 000	16 800	123 200
Imprimantes	5 000	20 000		25 000
Scanners	0	1 800		1 800
TOTAL	117 000	139 800	16 800	240 000

Remarque :

Tous les éléments du matériel informatique sont amortis en linéaire sur 3 ans avec une première annuité calculée *pro rata temporis*.