

EVALIPSIA

**LOGICIEL D'ÉVALUATION DE
COMPÉTENCES EN
INFORMATIQUE**

DOSSIER DE CONCEPTION

SOIDRIDDINE MOUDERE

CNAM IHM

07/11/03

Table des matières

CHAPITRE 1 : ÉTUDE PRÉALABLE.....	5
1 Présentation.....	5
1.1 IPSIA en quelques mots.....	5
1.2 Le projet	5
2 Etude de l'existant.....	5
2.1 Méthodes d'évaluation les plus courantes.....	5
2.1.1 L'entretien d'évaluation.....	6
2.1.2 Le questionnaire à choix multiple.....	6
2.1.3 La situation/problème.....	6
2.2 L'évaluation dans les centres orientations.....	6
2.3 L'évaluation selon IPSIA.....	7
2.3.1 Sa méthode.....	7
2.3.2 Sa conception de l'évaluation.....	7
2.4 La procédure actuelle.....	7
2.4.1 Evaluation orale.....	7
2.4.2 QCM sur culture informatique générale.....	7
2.4.3 Synthèse des informations sous forme projet de formation.....	8
2.5 Le logiciel de test par QCM.....	8
2.5.1 Son fonctionnement.....	8
2.5.2 Ses thèmes.....	8
2.5.3 La base de données sous forme de fichier texte.....	8
3 Les améliorations souhaitables.....	8
3.1 Amélioration fonctionnelles.....	9
3.1.1 Evolution de la base des QCMs.....	9
3.1.2 Réalisation de test interactif.....	9
3.1.3 Procédure de génération automatique de profil de compétence.....	9
3.1.4 Rejeu de l'évaluation par relecture des résultats détaillés.....	9
3.2 Améliorations IHM.....	9
3.2.1 Sélection des tests de l'évaluation.....	9
3.2.2 Réalisation d'évaluation.....	9
3.2.3 Interprétation des résultats d'évaluation.....	9
3.2.4 Mise à jour de la base de données.....	9
4 Les contraintes.....	10
4.1 Les fonctionnelles.....	10
4.1.1 Le type de public visé.....	10
a) Les utilisateurs.....	10
b) Les décideurs.....	10
4.2 Les non fonctionnelles.....	11
4.2.1 Mise à jour du logiciel existant.....	11
4.2.2 Développement en .Net.....	11
5 La solution proposée.....	11
5.1 Justification.....	11
5.1.1 Améliorations IHM.....	11

5.2 Son fonctionnement.....	11
5.2.1 Liste des fonctions systèmes :.....	11
5.2.2 Ses résultats : les profils de compétence graphique.....	12
5.2.3 Les différents types de tests.....	12
5.2.4 Les flux de données.....	12
5.2.5 Les fonctions systèmes supplémentaires (optionnelles).....	12
a) L'assistant d'évaluation	12
b) Procédure d'évaluation pas à pas automatisée par thème.....	12
5.3 Plan.....	12

CHAPITRE 2 : ANALYSE.....14

1 Plan.....	14
2 Le dictionnaire des données.....	14
2.1 Définitions.....	14
2.1.1 Les catégories.....	14
2.1.2 Les propriétés.....	14
2.2 Liste des thèmes par domaine de formation.....	15
2.3 Liste des objectifs fonctionnels par thème.....	16
3 Analyse statique.....	16
3.1 Les acteurs du système.....	17
3.2 Acteurs et système statique.....	17
3.3 Tableau des cas d'utilisation.....	17
3.4 Description des cas d'utilisation principaux.....	18
3.4.1 L'identification.....	18
3.4.2 Sélection d'une évaluation.....	18
3.4.3 Paramétrage d'une évaluation.....	18
3.4.4 Réaliser un test (scénario principale).....	19
3.4.5 Réaliser un test avec vérification de ses réponses.....	19
3.4.6 Générer profil de compétences.....	19
3.4.7 Afficher les résultats détaillés.....	20
3.5 Synthèse.....	20
4 Analyse dynamique.....	21
4.1 Le système globale.....	21
4.2 Le système détaillé.....	22
4.2.1 L'identification.....	22
a) Le candidat existe dans la base des candidats.....	22
b) Le candidat n'existe pas dans la base des candidats.....	22
4.2.2 La réalisation d'un test.....	23
a) Demande et réalisation de test (modèle générique).....	23
b) Réalisation d'un QCM.....	24
c) Variante de réalisation avec vérification de la réponse.....	25
4.2.3 La génération d'un profil de compétence.....	26
4.2.4 L'affichage des résultats.....	27
a) Résultats détaillés.....	27
b) Profil de compétence.....	28
4.3 Synthèse.....	28

5 Définition de l'architecture.....	28
5.1 L'architecture logicielle de l'application.....	28
5.1.1 Le noyau fonctionnel.....	28
a) Son rôle et sa composition.....	28
b) Son interface.....	29
5.1.2 Les formulaires graphiques.....	29
a) L'identification.....	29
b) La sélection d'évaluation.....	29
c) Réalisation d'un test du type QCM.....	30
d) Le résultat de d'évaluation.....	30
e) Le profil de compétence.....	30
5.2 Le déploiement de la solution.....	30

CHAPITRE 3 : CONCEPTION DÉTAILLÉE ET IMPLÉMENTATION...

32

1 Module : profil de compétence.....	32
1.1 Principe de fonctionnement.....	32
1.2 Son implémentation.....	32
1.2.1 L'interface Profil.....	32
1.2.2 La classe Notation.....	32
1.2.3 Le contrôle DiagrammeEtoile.....	32
a) Mise en évidence de détails par survol de région(Axe, légende).....	33
b) Accès aux résultats détaillés.....	33
1.3 Implémentation particulières.....	33
a) Les exercices à correction automatique.....	33
1.4 Impression du profil de compétences.....	33
2 Module : base de données.....	33
2.1 Rôle.....	33
2.2 Son fonctionnement.....	33
2.3 Ses opérations.....	34
2.3.1 GetLigne.....	34
2.3.2 PutLigne.....	34
2.4 Conception détaillée.....	34
2.4.1 Association table fichier ID_BD.....	34
2.4.2 Recherche simple d'enregistrement (Sélection et Projection).....	35
a) Le <exp1> OU <exp2>.....	35
b) Le <exp1> ET <exp2>.....	35
2.4.3 Recherche complexe d'enregistrement.....	35
a) Opérations unaires avec rangement des nuplets.....	35
b) Sélection par valeur avec borne.....	35
c) Opérations binaires avec rangement des nuplets.....	35
2.4.4 Transformation des requêtes.....	36
a) Chaîne de caractère uniforme.....	36
b) Nombre en entier.....	36
2.4.5 Connexion.....	36
a) Base de données SQL.....	36
b) Base de fichier .bd.....	36
2.5 Diagramme de classe d'implémentation.....	36

2.6 Description des tables/fichiers.....	36
2.6.1 Données principales.....	36
2.6.2 Données particulières.....	37
3 Module : gestionnaire de fichier de configuration.....	38
3.1 Définition de la syntaxe du fichier.....	38
3.1.1 Caractère de commentaire.....	38
3.1.2 Définition d'une section.....	38
3.1.3 Déclaration de valeur de variable.....	38
3.2 Les fonctionnalités.....	38
3.2.1 L'initialisation.....	38
3.2.2 La lecture d'une variable.....	38
3.2.3 La création ou le positionnement d'une variable.....	38
4 Module : charte graphique du logiciel.....	38
4.1 Les boîtes de dialogues.....	38
4.1.1 Les erreurs.....	39
4.1.2 Les avertissements.....	39
4.2 Font et couleurs.....	39
4.3 Navigabilité.....	39
4.3.1 Rôle et intérêt.....	39

CHAPITRE 1 : ÉTUDE PRÉALABLE

1 Présentation

1.1 IPSIA en quelques mots

Entreprise créée en 2001, avec pour objectifs de dispenser des formations Informatiques pour les PME-PMI. Le cœur de ses formations se situe autour de la bureautique. Mais très vite IPSIA a aperçu que des formations efficaces sont des formations bien adaptées aux profils des stagiaires et à l'environnement de travail de celui-ci. C'est pour cela qu'IPSIA met l'accent sur l'évaluation. L'évaluation que l'on peut aussi appeler la méthode IPSIA c'est trois ans d'expérience dans le domaine qui ont permis le développement de méthode et d'outil d'évaluation comme le logiciel Evalipsia, ou encore la mise en place de projets de fin de formation, des cas d'étude concrets qui placent le stagiaire en situation-problème lui permettant ainsi de constater sa progression.

IPSIA c'est des cours collectifs et/ou individuel pour tout le monde qui souhaite ou qui a besoin d'exploiter l'ordinateur à des fins personnels ou professionnels. Les formations bureautiques sont les dispensés, mais si votre besoin est d'évaluer le gain que l'outil informatique pourrait vous apporter.

1.2 Le projet

Evalipsia est un logiciel, un outil qui automatise la méthode d'évaluation des stagiaires. L'évaluation consiste à placer le stagiaire face à plusieurs questionnaires à choix multiples et à comptabiliser les réponses exactes par domaine afin d'en mesurer le niveau de maîtrise. Cette évaluation est effectuée pré et post formation, dans le premier cas elle évalue le niveau du stagiaire par domaine et met en évidence les difficultés et/ou les lacunes de celui-ci afin d'adapter le contenu de la formation. Dans le second cas, elle vérifie que l'enseignement dispensé a bien atteint l'objectif de formation.

L'originalité de l'approche d'évaluation du client vient de la représentation du niveau. Celle-ci a la particularité de traduire en un seul graphique bidimensionnel la visualisation du niveau absolu d'une personne en Informatique. Ce qui permet à la personne de représenter son niveau estimé et ensuite à IPSIA de fournir, suite à l'évaluation, le niveau réel et le niveau à atteindre pour la personne.

De plus les questions permettent d'évaluer l'autonomie de la personne face à l'ordinateur. De contrôler si celui-ci maîtrise le vocabulaire nécessaire à la compréhension des messages d'erreur et surtout s'il son niveau actuelle lui permet de communiquer efficacement avec un personnel compétent afin d'effectuer le diagnostic de la panne et si le cas le permet de réaliser les actions correctives.

Evaluation automatique des compétences en bureautique. Word , Excel.

Evalipsia est conçu sur le retour d'expérience de formation. L'ensemble des questionnaires qui le constitue sont élaborés selon le même principes décrit ci-après. Si les questionnaires sont présentés selon des thèmes (Internet, bureautique, multimédia, réseaux, maintenance, etc..) ceux-ci sont évalués selon un besoin fonctionnel pré-configuré. Par exemple, si l'on choisit le publipostage, les questions posées auront pour objectifs la vérification de la maîtrise des outils et méthodes nécessaires et suffisantes à la réalisation de cette fonction.

2 Etude de l'existant

2.1 Méthodes d'évaluation les plus courantes

2.1.1 *L'entretien d'évaluation*

Il est la forme la plus répandue de technique d'évaluation. Il permet à l'évaluateur de vérifier les compétences recherchées par un entretien au cours duquel l'évaluateur pose des questions qu'il élabore en fonction des réponses du candidat. Dans cet exercice l'évaluateur a la possibilité d'apprécier et de s'adapter au candidat. L'adaptation porte aussi bien sur le niveau de la compétence à mettre en évidence que sur la forme du questionnement.

L'inconvénient majeur de cette méthode est que pour une évaluation globale le processus est fastidieux pour l'évaluateur.

2.1.2 *Le questionnaire à choix multiples*

C'est la solution technique la plus souvent élaborée pour faciliter la correction. Elle permet de simplifier, une seule interprétation de la réponse. Le processus de correction peut être automatisé ou délégué ainsi que les critères de correction.

2.1.3 *La situation/problème*

Il s'agit d'un exercice élaboré lorsque la compétence à mettre en évidence est particulière à un contexte et trop complexe pour en terme de succession d'action pour être formulée sous la forme d'un QCM. Cette forme d'exercice a l'avantage, lorsqu'il est bien conçu de ne mettre en évidence que la compétence recherchée.

L'inconvénient est qu'il demande une très bonne connaissance métier pour sa création mais aussi pour sa correction.

2.2 L'évaluation dans les centres d'orientation

Les centres d'orientation ont recours à l'évaluation pour établir un bilan de compétences. Or ce bilan ne se limite pas à l'appréciation des outils maîtrisés par le candidat, la mesure des acquis techniques ne représente qu'une étape parmi les sept décrites ci-après :

Préambule : Etude de la demande et faisabilité du bilan avant de prendre un engagement réciproque et définir un calendrier.

Déroulement du bilan :

1 Définition précise des objectifs, étude du parcours personnel et professionnel et identification des compétences.

2 Analyse des acquis : connaissances générales et spécifiques, aptitude à acquérir de nouvelles connaissances dans des domaines privilégiés.

3 Mise en évidence des intérêts professionnels et des motivations au travail.

4 Analyse de l'image de soi et des principaux traits de personnalité.

5 Approfondissement sur l'orientation en fonction des capacités propres, élaboration d'objectifs, ouverture constructive sur l'avenir. Définition d'un champ d'évolution/reconversion professionnelle.

6 Remise en fin de bilan d'un rapport détaillé faisant la synthèse des différentes approches (entretiens, batterie de test, analyse graphologique...), dont chaque phase a été restituée oralement au fil des séances.

Le taux d'erreur de l'évaluation

Le pourcentage de la mesure des acquis dans le bilan de compétence

La différence de rendement entre les techniques d'évaluation

La différence de coût entre les différentes techniques

Tous ces arguments justifient l'emploi de QCM simple en papier

2.3 L'évaluation selon IPSIA

2.3.1 Sa méthode

IPSIA collabore avec vous dans l'expression de votre besoin de formation afin d'adapter aux mieux ses solutions de formation à votre profil.

IPSIA évalue le niveau d'autonomie des stagiaires face à l'ordinateur et adapte le contenu de la formation en fonction.

Réalisation d'un profil fonctionnel à atteindre, d'un profil fonctionnel à atteindre et d'un profil général de départ

2.3.2 Sa conception de l'évaluation

L'expression du besoin de formation

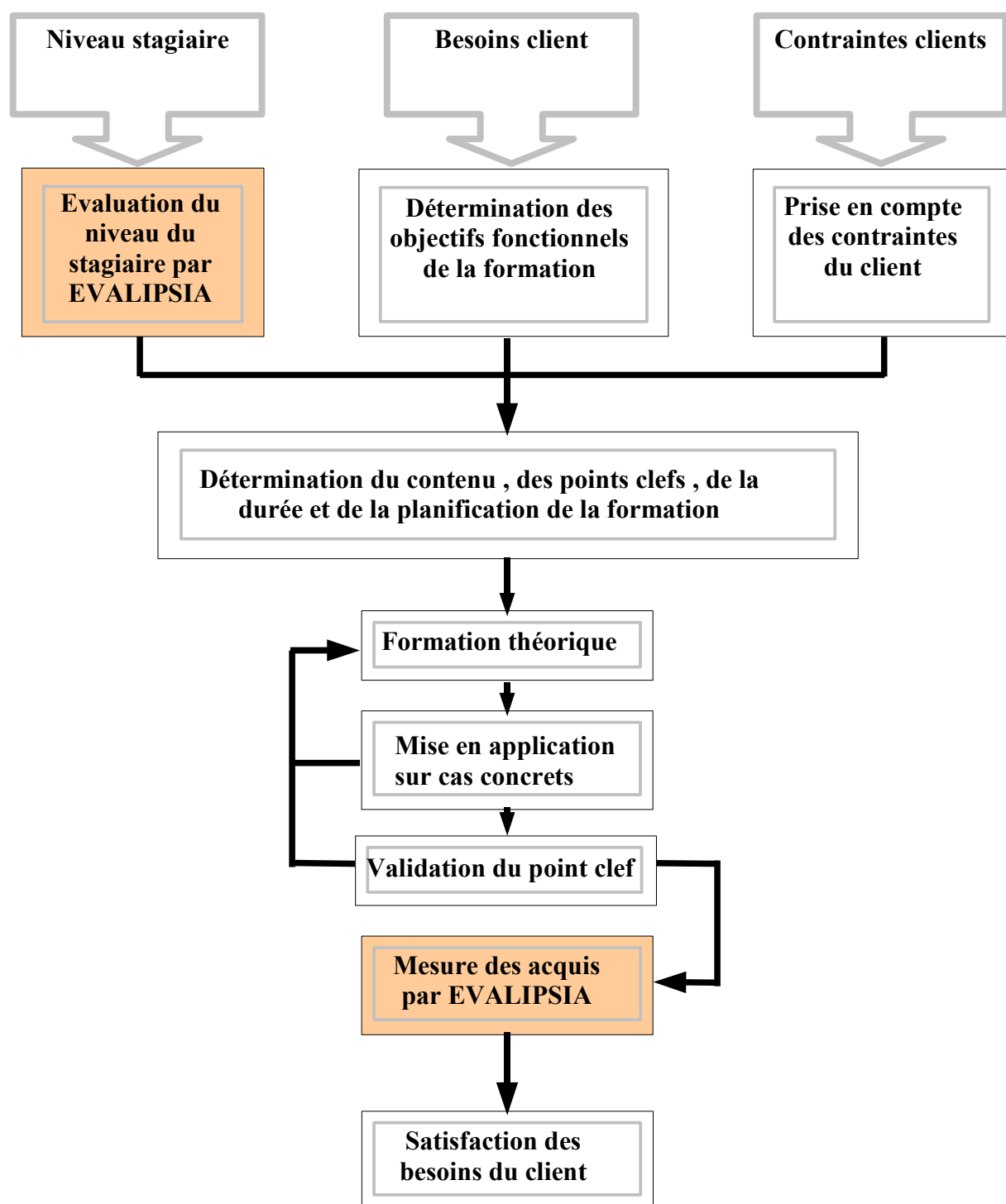
Évaluation des besoins fonctionnelles : il s'agit de recueillir les besoins exprimés par le client afin de déterminer l'objectif fonctionnel de la formation. Concrètement, ce processus se déroule naturellement lors d'une rencontre ou d'une conversation téléphonique. À l'issue de cette entretient le conseiller en formation d'IPSIA traduit le besoin fonctionnel du client en un ensemble d'outils à maîtriser pour être capable d'améliorer le traitement du besoin fonctionnel.

Évaluation du niveau fonctionnel de départ : cette évaluation consiste à vérifier et à établir le niveau réel du stagiaire par rapport aux outils à maîtriser spécifiés par l'évaluation du besoin fonctionnel.

Évaluation des contraintes non fonctionnelles : cette étape permet d'évaluer la maîtrise des contraintes d'environnement par le stagiaire. Il s'agit de vérifier aux préalables ses capacités à traiter les événements intempestifs liés à l'utilisation courante d'un ordinateur qui diminuent le rendement. Cette vérification qualitative du niveau général du stagiaire se fait par un logiciel qui place le stagiaire face à l'ordinateur et à un questionnaire à choix multiples. Le résultat détermine le niveau de maîtrise de l'environnement de l'ordinateur par le stagiaire, ce qui permet d'ajuster efficacement le contenu de la formation

Vérification du bénéfice de la formation : à l'issue de l'apprentissage de outils et des méthodes, les acquis sont évalués de deux manières. Premièrement, par ré-évaluation à travers Evalipsia pour contrôler la progression. Deuxièmement, par la réalisation d'un projet de fin de formation qui place le stagiaire dans des conditions réelles d'utilisation des outils et des méthodes de travail acquises.

Sa conception de l'évaluation s'intègre dans un processus de formation qu'IPSIA tente de mettre en place dans sa démarche qualité. Le schéma suivant illustre ce processus :



2.4 Sa procédure actuelle

Le détail de la méthode actuelle est issue du recueil d'information obtenue par interview de la responsable d'IPSIA et par observation d'entretien entre un conseiller IPSIA et un candidat à formation. Les paragraphes suivant indiquent les principaux axes observés.

2.4.1 *Évaluation orale*

L'essentiel de l'évaluation est une évaluation orale mené par le conseiller IPSIA qui recueille dans l'ordre :

- L'objectif du candidat
- Les informations personnelles : nom, prénom, niveau scolaire
- Les informations professionnelles : secteur d'activité, postes occupés, fonction

exercée

- Les compétences estimées en générale et relativement à l'objectif visé.
- Le matériel Informatiques personnelle

2.4.2 *QCM sur culture informatique générale*

Dans le cas où le conseiller IPSIA juge nécessaire ou bien dans le cas où le candidat fait part de son souhait d'être évalué plus précisément, alors celui-ci réalise une évaluation par QCM sur l'informatique générale. Le résultat de ce QCM permet de valider ou de préciser la concordance entre le niveau estimé et le niveau réel du candidat.

2.4.3 *Synthèse des informations sous forme projet de formation*

A l'issue de l'évaluation orale et/ou de l'évaluation plus précise par QCM, le conseiller IPSIA rédige un projet de formation en mettant en concordance les objectifs du candidat, son niveau et ses contraintes professionnelles. Ce projet de formation est ensuite donné au client sous la forme d'un devis de formation.

2.5 Le logiciel de test par QCM

Les QCMs sur la culture générale ont été réalisés sous la forme d'un logiciel qui présente des QCMs organisés par thèmes.

2.5.1 *Son fonctionnement*

A chaque question le candidat a le droit de vérifier si sa réponse est exacte ou non. Où sinon de passer à la question suivante. A l'issue du test le résultat est fourni sous la forme d'un cumul des bonnes réponses par rapport au nombre de questions total du QCM réalisé.

2.5.2 *Ses thèmes*

- frappe :
- clavier :
- gravure :
- Internet :
- le bureau :
- les supports amovibles :
- les logiciels :
- les supports amovibles :
- les documents :

2.5.3 *La base de données sous forme de fichier texte*

La base des données d'un QCM est réalisée avec un fichier texte contenant les informations suivantes :

Sur la première ligne 1^{er} colonne :

- le nom du test

A partir des lignes suivantes et sur chaque ligne :

- Numéro de la question (nombre entier à partir de 1)
- Case à cocher (si le QCM contient de 0 à plusieurs réponses)
- Option (si la question possède une et une seule réponse)
- Titre de la question
- Les propositions [1..8] : les textes des propositions de réponse à la question
- La solution (suite des numéros de propositions)

3 Les améliorations souhaitables

Comme tenu de l'existant, l'idée d'améliorer l'outil d'évaluation a été exprimé pour les raisons suivantes. Tout d'abord, l'outil doit être modifier afin de permettre son intégration dans le processus de formation d'IPSA. Pour cela, il doit permettre la mise en évidence de la personnalisation du processus de formation. Fournir un résultat directement et facilement exploitable dans la rédaction du projet de formation.

3.1 Amélioration fonctionnelles

Pour atteindre les objectifs cités précédemment les paragraphes suivant détaillent les principaux axes d'amélioration souhaités

3.1.1 Evolution de la base des QCMs

Celle-ci doit être enrichie afin de couvrir plus efficacement l'autonomie sur le poste de travail dans les domaines du multimédia, de la bureautique et de l'Internet.

3.1.2 Réalisation de test interactif

Des exercices interactifs dans le domaine du secrétariat devront être introduit afin de contrôler la maîtrise opérationnelle de l'outil en plus de la connaissance théoriques de celui-ci.

3.1.3 Procédure de génération automatique de profil de compétence

Obtenir la génération automatique du profil de compétence à partir des résultats des tests réalisés. Le calcul du profil étant possible grâce à la classification de chaque test selon des critères de niveau et d'appartenance à un thème et un domaine.

3.1.4 Evaluation adaptative

Une sélection automatique des tests à réaliser en fonction des résultats aux tests en cours de réalisation. Il s'agit pour le logiciel de s'adapter au candidat afin de réaliser une évaluation centré sur son niveau afin de recueillir un maximum de données utiles pour l'élaboration de son profil de compétence.

3.2 Améliorations IHM

3.2.1 Sélection des tests de l'évaluation

L'objectif est de simplifier la sélection d'évaluation en proposant une accès direct à des évaluations pré-définie selon des critères de classification à définir. Le tout étant d'éviter le paramétrage de l'évaluation par la sélection manuelle de critères

3.2.2 Réalisation d'évaluation

Volonté de réaliser une interface à manipulation directe : Public non averti. Aucune connaissance de l'utilisation de l'ordinateur à priori.

3.2.3 Interprétation des résultats d'évaluation

Réduction de la distance d'interprétation des résultats détaillées des tests et du profil de compétence par une représentation graphique simple. La réduction de la distance d'évaluation doit être suffisante pour permettre une accessibilité par le candidat.

4 Les contraintes

Les contraintes de réalisation sont fixées principalement par la volonté de conserver et d'améliorer l'existant.

4.1 Les fonctionnelles

4.1.1 Le type de public visé

Deux types de public sont visés à travers ce logiciel. Ils se distinguent par leur relation avec le logiciel. Les deux paragraphes suivant distinguent et précise la nature de leur relation.

a) Les utilisateurs

Des formations individualisées sur l'outil informatique (bureautique, Internet ou multimédia) avec un besoin d'autonomie sur le poste de travail dispensé aux catégories sociales suivantes : individu exerçant une profession libérale, ayant un emploi C.E.S (ANPE) ou un particulier.

De manière générale il s'agit de personnel ayant un niveau minimal de niveau IV ou V

Voir le tableau suivant pour le descriptif des différents niveau.

NIVEAU	DEFINITION	INDICATIONS
V	Personnel occupant des emplois exigeant normalement un niveau de formation équivalent à celui du brevet d'études professionnelles (BEP) ou du certificat d'aptitude professionnelle (CAP), et par assimilation, du certificat de formation professionnelle des adultes (CFPA) du premier degré.	Ce niveau correspond à une qualification complète pour l'exercice d'une activité bien déterminée avec la capacité d'utiliser les instruments et les techniques qui s'y rapportent. Cette activité concerne principalement un travail d'exécution qui peut être autonome dans la limite des techniques qui y sont afférentes.
IV	Personnel occupant des emplois de maîtrise ou d'ouvrier hautement qualifié et pouvant attester d'un niveau de formation équivalent à celui du brevet professionnel (BP), du brevet de technicien (BT), du baccalauréat professionnel ou du baccalauréat technologique.	Une qualification de niveau IV implique davantage de connaissances théoriques que le niveau précédent. Cette activité concerne principalement un travail technique qui peut être exécuté de façon autonome et/ou comporter des responsabilités d'encadrement (maîtrise) et de coordination.
III	Personnel occupant des emplois qui exigent normalement des formations du niveau du diplôme des Instituts Universitaires de Technologie (DUT) ou du brevet de technicien supérieur (BTS) ou de fin de premier cycle de l'enseignement supérieur.	La qualification de niveau III correspond à des connaissances et des capacités de niveau supérieur sans toutefois comporter la maîtrise des fondements scientifiques des domaines concernés. Les capacités et connaissances requises permettent d'assurer de façon autonome ou indépendante des responsabilités de conception et/ou d'encadrement et/ou de gestion.
II	Personnel occupant des emplois exigeant normalement une formation d'un niveau comparable à celui de la licence ou de la maîtrise.	A ce niveau, l'exercice d'une activité professionnelle salariée ou indépendante implique la maîtrise des fondements scientifiques de la profession, conduisant généralement à l'autonomie dans l'exercice de cette activité.
I	Personnel occupant des emplois exigeant normalement une formation de niveau supérieur à celui de la maîtrise.	En plus d'une connaissance affirmée des fondements scientifiques d'une activité professionnelle, une qualification de niveau I nécessite la maîtrise de processus de conception ou de recherche.

b) Les décideurs

Les décideurs ne sont pas des personnes pour lesquels l'utilisation du logiciels est destiné. Il s'agit de personnes amenées à juger de l'utilité, la nécessité, du gain de ce logiciel dans le processus de formation proposé par la société IPSIA. Ceux sont généralement des personnels de niveau II et I sensibles aux justifications, à la cohérence et aux résultats de la démarche mise en application par le logiciel.

4.2 Les non fonctionnelles

4.2.1 *Mise à jour du logiciel existant*

En effet, une version actuelle d'Evalipsia permet de réaliser des QCMs et d'obtenir le nombre de réponse exactes obtenues. La base de QCM devra donc être conservée.

4.2.2 *Développement en .Net*

Le logiciel a été réalisé sous Visual Studio .Net en Visual Basic. Le nouvelle version devra donc être développer pour le Framework .Net. Les parties modulaires pourront être développées dans d'autres langages du .Net (C# ou C++) mais les formulaires devront être écrite en Visual Basic .Net.

5 La solution proposée

La solution proposée consiste à réaliser les améliorations proposées en respectant les contraintes de développement proposées mais selon des processus de conception modulaires et itératifs.

5.1 Justification

Afin de tenir compte de la contrainte d'accessibilité et de la volonté d'améliorer l'interprétation des résultats, les améliorations proposées visent à :

- **réduire les distances d'évaluations** des résultats d'évaluation en proposant une représentation graphique sous forme de diagramme étoile des compétences en informatique.
- réaliser une **interface centrée sur l'utilisateur** qui est une candidat de niveau IV ou V en ne proposant que des options simples et valides lors de la réalisation de l'évaluation.
- **respecter une cohérence** et préparer l'apprentissage du candidat par la définition d'une charte graphique pour le logiciel (dimension et position des boutons, couleurs représentatifs des niveaux, formalismes des messages d'erreurs, navigation par bouton)

5.2 Son fonctionnement

Le fonctionnement du logiciel du point de vue du candidat réalisant une évaluation consiste à :

- une identification
- une sélection d'évaluation (Choix réaliser par le conseiller)
- réalisation de tests
- visualisation du résultat et du profil de compétence

Ce fonctionnement nécessite l'intégration de fonctionnalités dans le logiciel détaillé par les paragraphes suivants

5.2.1 *La liste des fonctionnalités*

Authentification : permet de réaliser une personnalisation du logiciel, une accès aux fonctions avancées du logiciel en fonction de l'identité de la personne connecté. Elle permet de stocker des renseignement sur le profil du candidat.

Sélection d'évaluation : des listes d'évaluation seront proposées. Elles seront regroupées selon le

domaine ou le thème de compétence visé. Pour les évaluation mettant en oeuvre des tests visant à qualifier une compétence particulière avec la possibilité de réaliser une correction automatique, ceux ci seront regroupés dans une rubrique test à correction automatique. Enfin une dernière rubrique comprendra les évaluations personnalisées dont l'ensemble des tests ont été choisi manuellement.

Tableau récapitulatif des rubriques de la sélection d'évaluation :

Rubriques d'évaluation	Type d'évaluation
Domaine	Evaluation du type QCM se reportant à un domaine particulier avec sélection automatique des questions
Thème	Evaluation du type QCM se reportant à un thème d'un domaine particulier avec sélection automatique des questions
Personnalisée	Evaluation contenant des tests de tout type sélectionner manuellement au préalable
Automatique	Evaluation composé de tests particuliers à correction automatique

Afin de favoriser l'accessibilité, des interacteurs devront visualiser le niveau de difficulté, la durée approximative et le nombre de tests qui composent l'évaluation.

La réalisation d'évaluation : elle est particulière à chaque type de tests (voir le paragraphe suivant). La réversibilité dans la validation de réponse à un test est volontairement limitée à la question courante. En effet l'ordre des questions n'est pas fortuit, si le candidat à la possibilité de naviguer au sein des questions déjà valider, il n'aura pas la possibilité de corriger et de modifier sa réponse pour ne pas dénaturer la lecture et la nature du critère évalué. Tous les tests réalisés par un candidat sont enregistrés en notant la date de réalisation afin d'effectuer un suivi pédagogique de son évolution.

La génération du profil de compétence : cette fonction sera possible grâce à la mise en place d'une base de donnée qui contiendra pour chaque test des associations précisant le niveau, le domaine, les thèmes et les objectifs fonctionnels connexes. A partir des résultats d'une évaluation une liste exhaustive des associations est établies. Pour chaque association une mesure du niveau est déterminé en fonction du score obtenue du nombre de question se rapportant à celle-ci. Une fois l'ensemble des associations déterminé et interprété, un profil de compétence est fournie selon chacune des vues. Le profil de compétence listera les notes dans chaque niveau pour les différentes vue que sont : la vue par domaine, par thème ou par objectif fonctionnel. Les différents niveaux seront débutant, moyen et avancé.

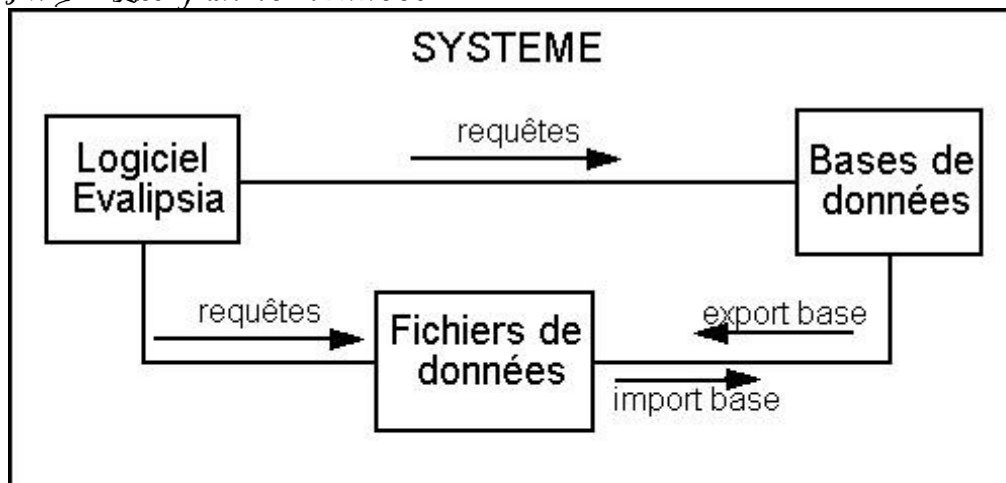
5.2.2 *Les différents types de tests*

Différents types de tests composent et composeront le logiciels. Il fournisse tous un résultat exploitable par calcul et visualisable au travers d'un diagramme étoile. Les différent type de tests sont les suivants :

- Question à choix multiples QCM : il représente la majorité des tests présents dans la base de données et sont destinées à couvrir la totalité des domaines et à vérifier les connaissances théoriques des candidats.
- Exercices interactif à correction automatique. Exemple test de frappe pour mesurer la justesse et la rapidité de recopie d'un texte. Ce type de test est destiné à rapprocher le candidat des condition réel de la mise en oeuvre de la compétence afin de vérifier plus sûrement son niveau de maîtrise.
- Exercices avec une correction manuelles : ils nécessitent un suivi et une correction par un conseillé. Ils peuvent être réalisés en dehors du logiciel Evalipsia mais le

résultat sera réintroduit et stocké dans le logiciel par le correcteur.

5.2.3 Les flux de données



5.3 Plan

- **L'étude de conception des améliorations proposées** : étude globale de l'évolution du logiciel. Réalisation d'un dossier de conception décrivant les contraintes et les interactions entre la base de données de tests, la réalisation du test et la générations des profils de compétences.
- **Etude détaillée et réalisation du module d'affichage des profils de compétences** : réalisation de l'étude détaillé et intégration du module d'affichage et de génération des profils de compétence.
- **Création de nouveau tests** : amélioration de la base de test QCM et de tests interactifs Word et Excel.
- **Amélioration de la base de données de tests** : étude, intégration et migration de la base de données du format texte vers une base de données PostgreSQL.

CHAPITRE 2 : ANALYSE

1 Plan

L'analyse du projet consiste à modéliser la solution proposée pour la conception de l'évolution du logiciel Evalipisa. La première étape consiste en une analyse des données qui constituent la base des tests. Cette étude spécifie les propriétés de classement des tests par rapport à la connaissance métier. La classification permet de spécifier les dépendances ainsi que leurs appartenances. Leur recensement est effectué de manière exhaustive.

Dans un second temps, une modélisation par les cas d'utilisations met en évidence les différentes fonctions du logiciel ainsi que les différents acteurs qui les mettent en oeuvre. Puis, à partir de la liste exhaustive des cas d'utilisations et du comportement dynamique de l'application la définition de l'architecture et de son déploiement sont présentés.

2 Le dictionnaire des données

Avant de réaliser, une énumération exhaustive du contenu des domaines, des thèmes et des objectifs fonctionnels, nous commençons par définir ses termes dans le paragraphe suivant.

2.1 Définitions

2.1.1 Les catégories

Domaine : ensemble vaste définissant l'utilisation de l'ordinateur selon le type de données et les documents manipulés.

Thème : compris dans un domaine, le thème précise le type d'utilisation de l'ordinateur selon un type de données manipulés.

Objectif fonctionnel : Orthogonale au thème et de fait au domaine, il vise la fonction à qualifier qualitativement et quantitativement en terme de compétence. Autrement dit, il justifie de la capacité de l'utilisateur à réaliser une tâche dans un domaine et selon un thème précis.

2.1.2 Les propriétés

Les propriétés s'appliquent aux objectifs fonctionnels afin de définir leur relation les uns par rapport aux autres et aussi leurs relations par rapport à leur domaine et à leur thème d'appartenance. Deux types de propriété sont définis ci-après :

Le niveau : il s'agit d'une propriété qui caractérise l'objectif fonctionnel afin de le positionner le degré de difficulté et à fortiori le degré de connaissance du candidat qualitativement de manière absolue.

Les différents niveau existant sont :

- **niveau 1** : débutant ; occasionnel
- **niveau 2** : confirmé ; moyen
- **niveau 3** : avancé ; bon
- **niveau inconnu** : non classifié

Le pré-requis : il caractérise un objectif fonctionnel afin de définir l'ordre et l'ensemble des objectifs fonctionnels à avoir atteint avant de s'attaquer à celui-ci. Il permet d'élaborer le planning de formation.

2.2 Liste des thèmes par domaine de formation

<i>Titre du domaine</i>	<i>Description du domaine</i>	<i>Titre du thème</i>
Bureautique	Activité de secrétariat	<ul style="list-style-type: none"> • Traitement de texte • Tableur, feuille de calcul • Présentation
Multimédia	Manipulation de l'image, du son, de la vidéo	<ul style="list-style-type: none"> • Editeur d'images • Editeur de photos • Editeur de sons • Montage vidéo • Scanner, imprimer document
Internet	Navigation, communication, recherche d'information sur le réseau Internet	<ul style="list-style-type: none"> • Envoi, réception de courrier • Recherche d'information • Navigation • Télécharger • Choisir sa connexion • Paramétrer sa connexion • Gestion de site Internet
Autonomie sur le poste de travail	Capacité à réagir aux erreurs types dont la provenance est liée à l'utilisation nominale de l'ordinateur et du logiciel	<ul style="list-style-type: none"> • Gestion de fichier et des dossiers • Utilisation du Bureau • Mise en oeuvre de l'ordinateur • Sauvegarde et restauration des documents • Sauvegarde et restauration de l'environnement de travail • Ajout de périphériques • Installation de logiciels • Spécification des besoins logiciels • Spécification des besoins matériels
Réseaux	Services, protocoles inter ordinateur sur réseau local(LAN) ou étendu(WLAN)	<ul style="list-style-type: none"> • Partager ses dossiers et ses fichiers et ses imprimantes • Partager sa connexion Internet • Poste client sur un réseau domestique • Poste client sur réseau domestique sans fils

2.3 Liste des objectifs fonctionnels par thème

<i>Autonomie sur le poste de travail</i>	
<ul style="list-style-type: none"> • Utiliser le clavier • Utiliser la souris • Mise en route de l'ordinateur • Arrêter l'ordinateur • Connaître l'architecture de l'ordinateur • Distinguer les différents logiciels de l'ordinateur • Organiser son travail à partir du bureau • Créer et utiliser les icônes du bureau • Déplacer, copier ou supprimer des fichiers ou des dossiers • Lancer une application • Fermer, réduire agrandir une fenêtre • L'extension des fichier ou comprendre l'association fichier/application • Distinguer les unités informatiques et leurs notations • Appréhender les ordre de grandeurs des unités informatique • Connaître les différents formats de documents • Compresser un document • L'édition de documents « images » • L'édition de documents « sons » • L'édition de documents « vidéos » • Archiver ou dé-archiver des documents • Sauvegarder et restaurer le secteur d'amorçage et la table des partitions • Utiliser les outils de partitionnement de disque dur • Connaître les méthodes de préventions et de résolution de pannes 	<ul style="list-style-type: none"> • Connaître les différents types de support de masse • Distinguer les différents types de support de masse amovible • Savoir choisir le support de masse • Maîtriser les paramètres d'un scanner • Choisir un scanner • Maîtriser les paramètres d'impression • Choisir une imprimante • Méthode d'organisation des fichiers • Sélectionner le support de sauvegarde du travail • Connaître les outils de sauvegarde inclus dans le système de l'ordinateur • Utiliser les outils des gravures de CD/DVD • Comprendre le démarrage d'un ordinateur • Utiliser, configurer le BIOS • Créer et utiliser une disquette de démarrage • Créer des comptes pour séparer les droits des utilisateurs • Connaître quelques politiques de sauvegarde manuelle • Faire une restauration partielle du système par point de restauration • Mise à jour des logiciels systèmes • Faire une Sauvegarde/restauration complète du système • Installer un système d'exploitation • Installer plusieurs systèmes d'exploitation sur un ordinateur • Partager des fichiers, des dossiers et des imprimantes sur un réseau domestique • Partager connexion Internet • Configurer le protocole TCP/IP • Configuration d'un poste client pour accéder à un réseau local Ethernet (LAN) • Configuration d'un poste client pour accéder à un réseau local sans fil (LAN Wifi)

3 Analyse statique

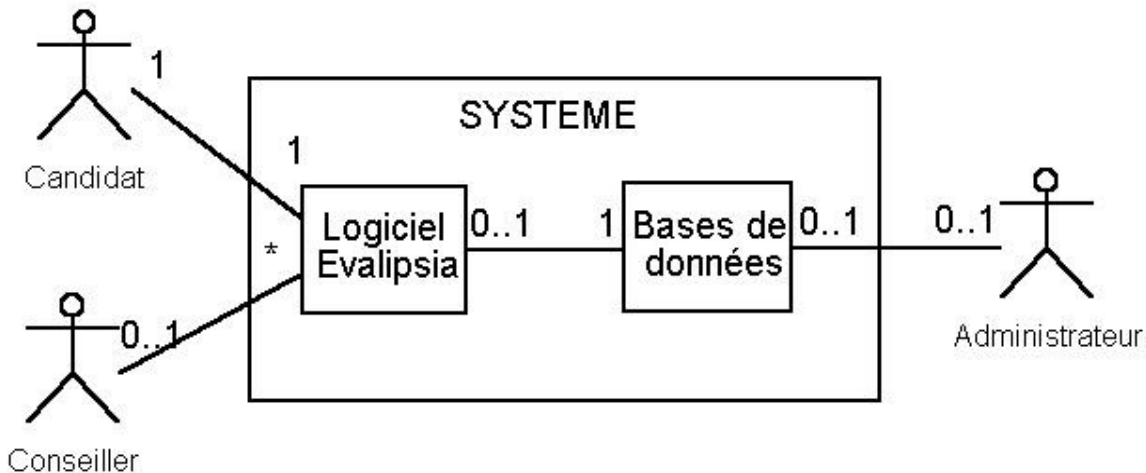
L'analyse statique du système consiste par réaliser une approche de la solution par une identification par les cas d'utilisations. La modélisation utilisé pour décrire ces différents cas d'utilisation est un dérivé de la méthode de modélisation orienté objet UML dans l'utilisation des cas d'utilisation pour identifier les différents tâches ainsi qu'un dérivé du modèle de MAD pour la description des tâches. La modélisation de la solution s'attachera dans une premier temps à décrire les différentes tâches au travers de l'analyse des différents cas d'utilisation pour ensuite décrire le comportement dynamique de l'ensemble de l'application pour finir par une définition des objets (intéacteur

graphique/formulaire et noyau fonctionnel) qui la constitue.

3.1 Les acteurs du système

- l'administrateur : il gère les base de données du logiciel
- le candidat : il utilise le logiciel pour subir une évaluation
- le conseiller : il paramètre le logiciel pour sélectionner les tests de l'évaluation et réaliser la correction des tests semi-automatique et manuel

3.2 Acteurs et système statique



Cette vue du système permet de montrer le nombre de type acteurs pouvant interagir avec le système à un instant donné. On constate donc que l'utilisation du logiciel nécessite au moins un acteur du type candidat, que accessoirement un conseiller peut intervenir sur le logiciel de même qu'un administrateur, qui lui interagira avec le sous-ensemble base de données.

3.3 Tableau des cas d'utilisation

Le tableau suivant liste de manière exhaustive les cas d'utilisation identifiés. Pour chaque cas l'acteur est précisé ainsi que l'action ou la réaction émis ou reçu par l'acteur.

<i>Cas d'utilisation</i>	<i>Acteur</i>	<i>Messages émis/reçus par les acteurs</i>
Gestion de la base de données des tests	administrateur	Émis : Ajout, suppression, modification
Gestion de la base de données des objectifs fonctionnels	administrateur	Émis : Ajout, suppression, modification
Gestion de la base de données des thèmes	administrateur	Émis : Ajout, suppression, modification
Gestion de la base de données des profils de compétences type	administrateur	Émis : Ajout, suppression, modification
Saisie son profil de compétence estimé	candidat	Emis : Notes estime par thème
Saisie des informations	candidat	Emis : information personnelle, modification informations personnelle
identification	Candidat, conseiller	Emis : information d'identification

<i>Cas d'utilisation</i>	<i>Acteur</i>	<i>Messages émis/reçus par les acteurs</i>
Réaliser un test	candidat	Emis : réponse au test
Générer profils de compétences	Candidat, conseiller	Emis : paramètres de génération et sélection des tests
Afficher les résultats	Candidat, conseiller	Emis : demande résultat Reçus : Profils de compétence ou résultats brut
Correction de exercice manuel ou semi-automatique	conseiller	Emis : note au test
Paramétrage d'une évaluation	conseiller	Emis : liste des objectifs du test

3.4 Description des cas d'utilisation principaux

Une description textuelle des cas d'utilisation est réalisés en précisant pour chacun d'eux les pré et post conditions, le type d'acteur mis en jeux, la description du cas nominal et des cas alternatifs.

3.4.1 *L'identification*

L'acteur est soit un conseiller soit un candidat.

Pré-condition : néant

Cas nominal :

- l'acteur saisie un nom et un prénom
- si le candidat existe l'identification réussi.
- si le candidat n'existe pas le système crée une nouvelle entrée dans la base de données des candidats et renvoie le l'acteur vers un formulaire de saisie d'informations sur le candidat.
- le système se paramètre pour réaliser toutes autres opérations sous l'identité du candidat identifié.

Post-condition :

Un candidat est identifié

3.4.2 *Sélection d'une évaluation*

l'acteur est soit un conseillé, soit un candidat, soit l'administrateur.

Pré-condition :

Le candidat doit être identifié par le système.

Cas nominal :

- L'acteur choisi une catégorie parmi les 4 catégories d'évaluations proposées : domaine, thème, personnalisé ou de test automatique.
- Puis il sélectionne une évaluation sachant que plus la couleur est sombre plus la difficulté du tests est élevé.

Post-condition :

Liste de test à réaliser par le candidat

3.4.3 Paramétrage d'une évaluation

l'acteur est un conseiller soit un candidat

Pré-condition :

Les pré-conditions pour réaliser un paramétrage sont que le candidat soit identifié par le système et qu'au moins 1 test existe dans la base de donnée de test

Cas nominal :

- l'acteur positionne les critères de filtrage des tests réalisable selon leur domaine, leur thème leur niveau.
- Il sélectionne les tests qu'il souhaite réalisé
- Il sélectionne la durée approximative de l'évaluation souhaitée.

Post-condition :

Liste de test à réaliser par le candidat

3.4.4 Réaliser un test (scénario principale)

L'acteur est un candidat.

Pré-condition :

- test sélectionné parmi la base de test
- un candidat est identifié par le système
- Au moins 1 test dans la base de donnée de test

Cas nominal :

- Le candidat réalise le test courant
- Le candidat valide sa réponse et passe automatiquement au test suivant.
- A la fin du test, le système enregistre le résultat de chaque test dans la base de données.
- Le système affiche le score atteint par le candidat à l'évaluation.

Post-conditions :

Test enregistré dans la base des tests réalisés.

3.4.5 Réaliser un test avec vérification de ses réponses

L'acteur est un candidat.

Pré-condition :

- test sélectionné parmi la base de test
- un candidat est identifié par le système
- Au moins 1 test dans la base de donnée de test

Cas optionnel :

- Le candidat réalise le test courant
- Le candidat demande la vérification de sa réponse.
- Le système valide la réponse, indique au candidat le résultat puis passe automatiquement au test suivant
- A la fin du test, le système enregistre le résultat de chaque test dans la base de données.
- Le système affiche le score atteint par le candidat à l'évaluation.

Post-conditions :

Test enregistré dans la base des tests réalisés.

3.4.6 Générer profil de compétences

L'acteur est un candidat ou un conseiller.

Pré-condition :

- l'identité du candidat existe dans la base de données des candidats
- au moins un test réalisé par le candidat

Cas nominal :

- Dans la liste des tests réalisés, sélectionner pour un candidat donné les tests selon les critères de domaine, thème et objectifs fonctionnels, date..
- Le système génère le profil de compétence et l'affiche.

Post-conditions :

Profils de compétences calculé

Cas optionnel : Génération du profils à l'issue d'une évaluation

Ajout dans les pré-conditions :

liste de test réalisée par une évaluation

Modification dans dans le déroulement :

Le système génère le profil de compétence à partir des résultats des test de l'évaluation réalisé

3.4.7 Afficher les résultats détaillés

Les acteurs candidats ou conseiller peuvent demander l'affichage des résultats d'une évaluation.

Les pré-conditions :

- l'identité du candidat existe dans la base de données des candidats
- au moins un test réalisé par le candidats

Cas nominal :

- Dans la liste des tests réalisés par un candidat sélectionner des tests en fonction de la date (date ou interval de temp)
- demander l'affichage des résultats de test.
- Si simple affichage des résultats alors afficher résultats de chaque test (titre du test + score)

Variante du cas d'utilisation :

- Dans le cas où la pré condition 1 n'est pas respecté le système signal à l'acteur l'erreur et demande une nouvelle identification.
- Dans le cas où la pré condition 2 n'est pas respecté le système signal à l'acteur l'erreur et renvoi l'acteur est (si c'est un candidat) à l'interface de sélection de test.

Post-conditions : néant

3.5 Synthèse

L'analyse des cas d'utilisation permet de distinguer des rôles distincts. Chacun de ces rôles est encapsulé dans des classes objets décrits dans les classes candidates ci-dessous.

Candidat : fournis les informations sur le candidat et conserve l'historique des évaluations réalisées par celui-ci.

Test : élément d'évaluation d'une partie d'un objectif fonctionnel. Il contient un type de test, le test,

le résultat au test sous la forme d'une note.

Profil : sous entendu profil de compétence. Il possède une liste d'objectifs fonctionnels organisés hiérarchiquement évaluer à partir de tests réalisés ainsi que la liste de thèmes relatifs aux objectifs fonctionnels. Il est généré à partir du résultat d'une liste de tests réalisés ou à partir d'une estimation de niveau pour des objectifs fonctionnels préalablement sélectionnés.

Objectif fonctionnel : énonce une capacité à réaliser une tâche. Il se quantifie selon une note pouvant être binaire ou décimale.

Thème : il décrit le domaine d'utilisation de l'ordinateur et englobe un liste d'objectif fonctionnel.

SGBD : gère le stockage et l'accès aux données de Candidat, de Tests, de Test réalisés et de Profil

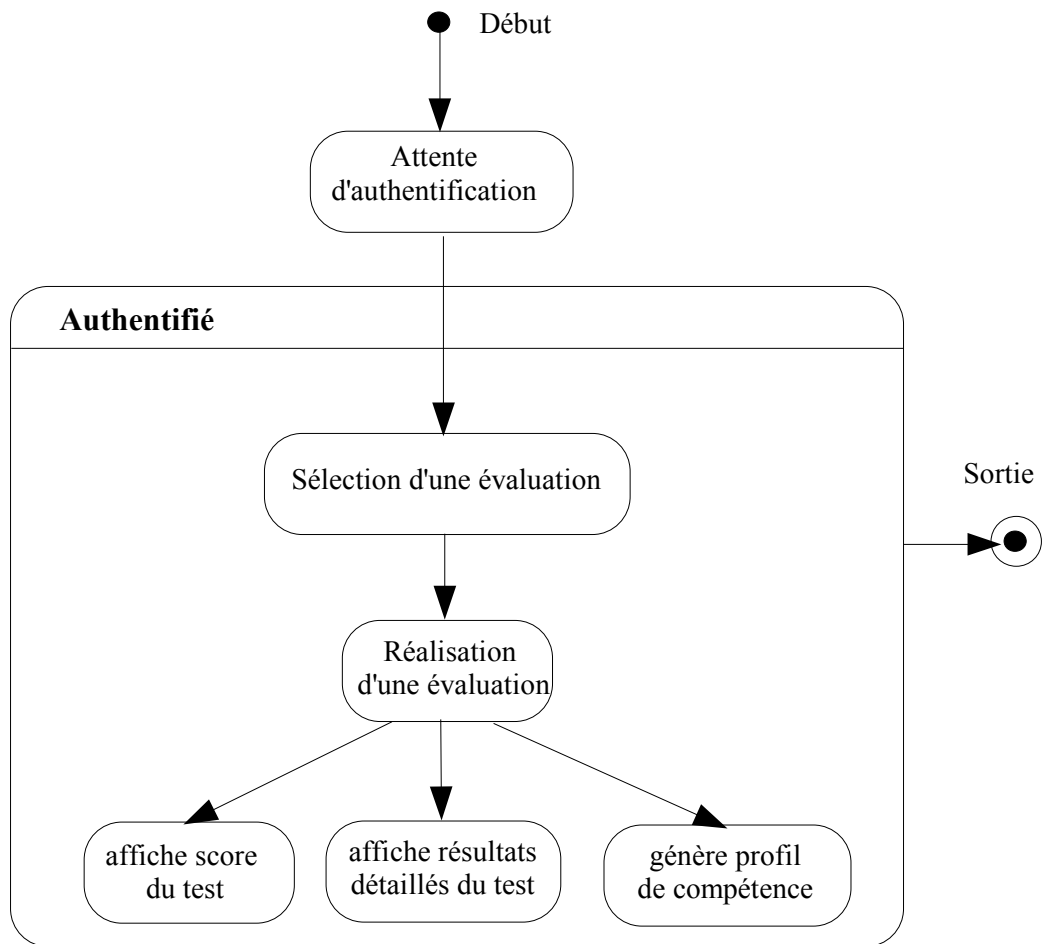
IHM : ensemble d'objets graphiques réalisant les interacteurs par lesquels les acteurs interagissent avec l' ensemble des classes précédentes.

4 Analyse dynamique

Le but de l'analyse dynamique est de mettre en évidence les interactions entre les différents cas d'utilisation afin de déterminer de manières exhaustives leur nombre ainsi que leur particularité. Une analyse globale du système est effectué avant de s'intéresser aux interactions qu'impliquent les cas d'utilisations identifiés dans les paragraphes précédents.

4.1 Le système globale

L'analyse dynamique globale du système montre l'enchaînement des cas d'utilisation en indiquant les différents états du système. Ces état sont illustrer dans le diagramme d'état suivant :



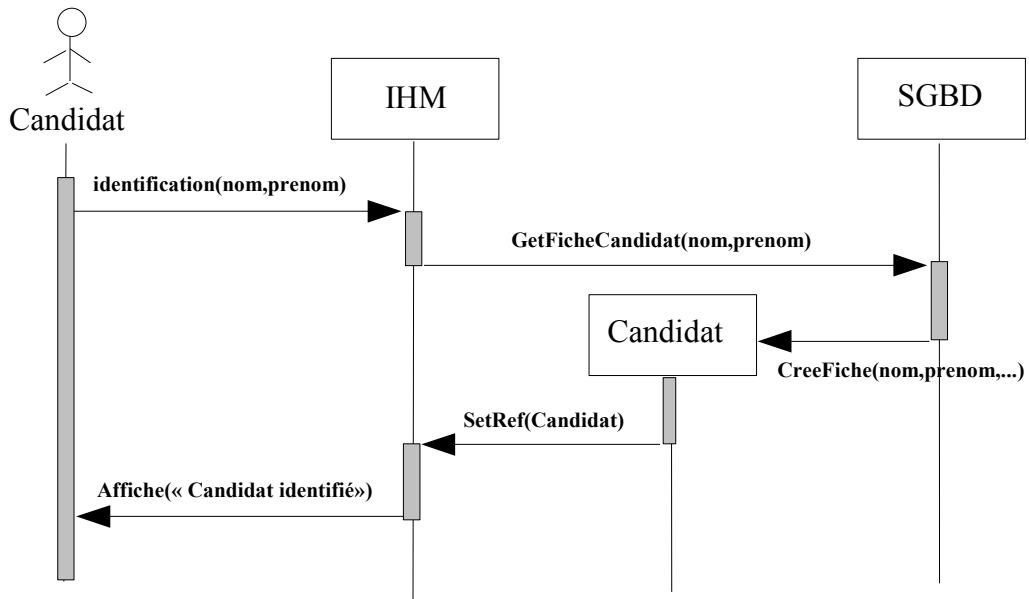
On constate qu'il existe un super état dans lequel l'application exécute la majorité de ces cas d'utilisation. Il s'agit de l'état authentifié. Un autre élément à remarquer est qu'à tout moment dans ce super état l'acteur a la possibilité de sortir de cette état et de terminer l'application.

4.2 Le système détaillé

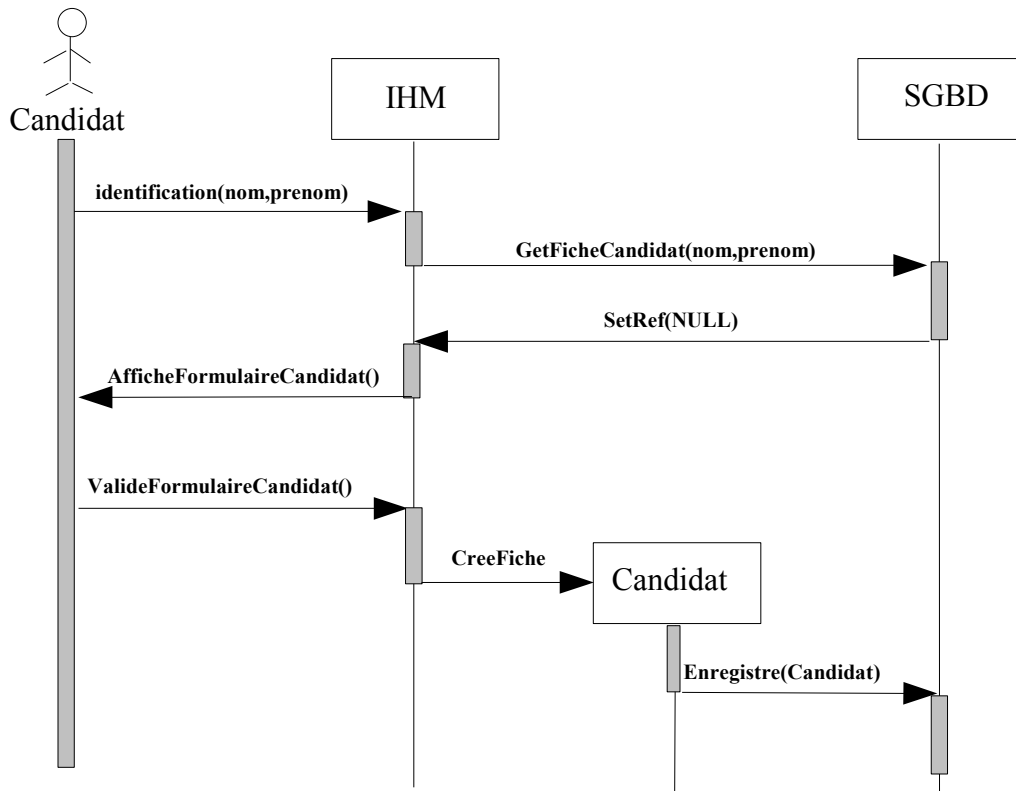
L'analyse dynamique du système indique l'enchaînement des cas d'utilisation et la manière dont les classes décrites dans le paragraphe précédent sont mises en oeuvre. Une description détaillée par des diagrammes de séquence des cas d'utilisation principaux est réalisée dans les paragraphes suivants :

4.2.1 L'identification

a) Le candidat existe dans la base des candidats

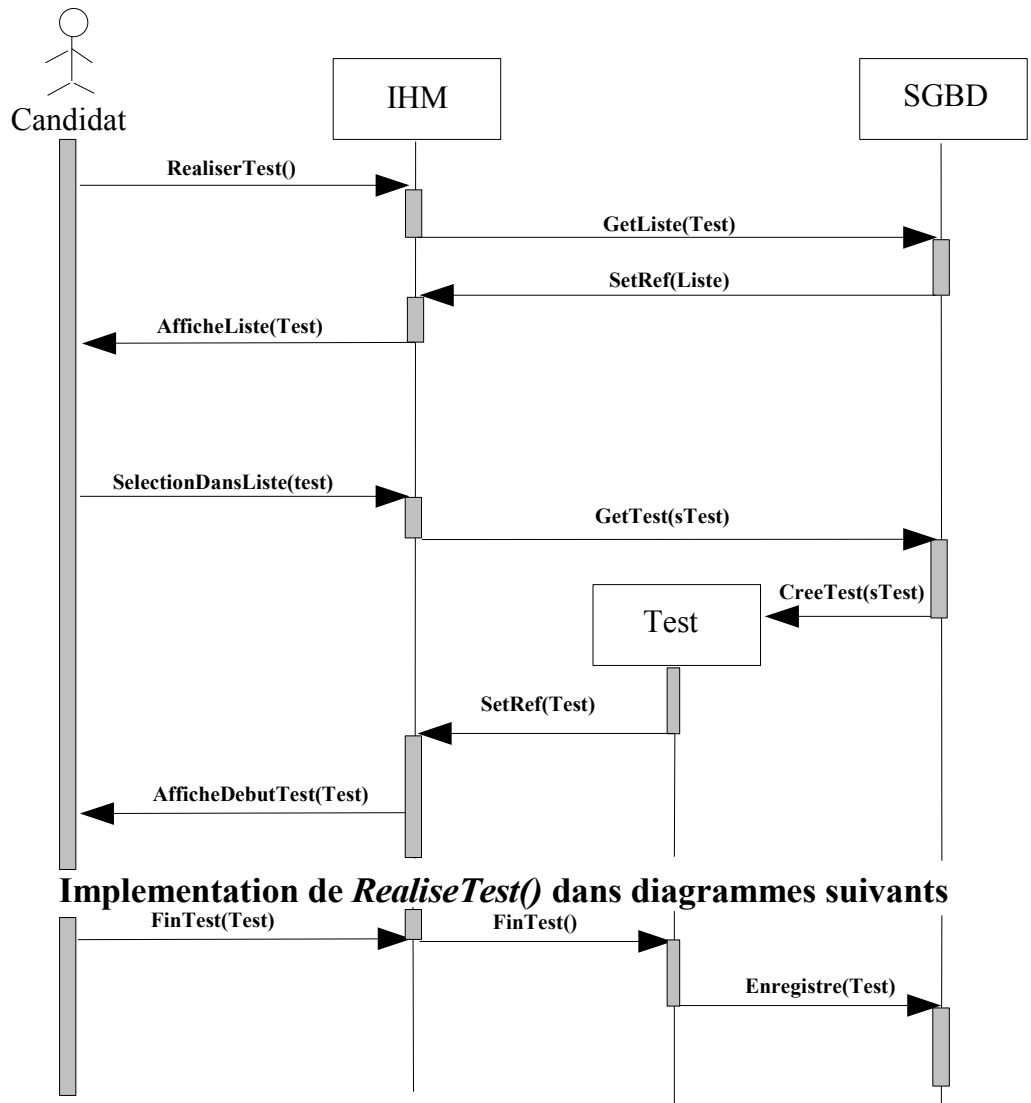


b) Le candidat n'existe pas dans la base des candidats

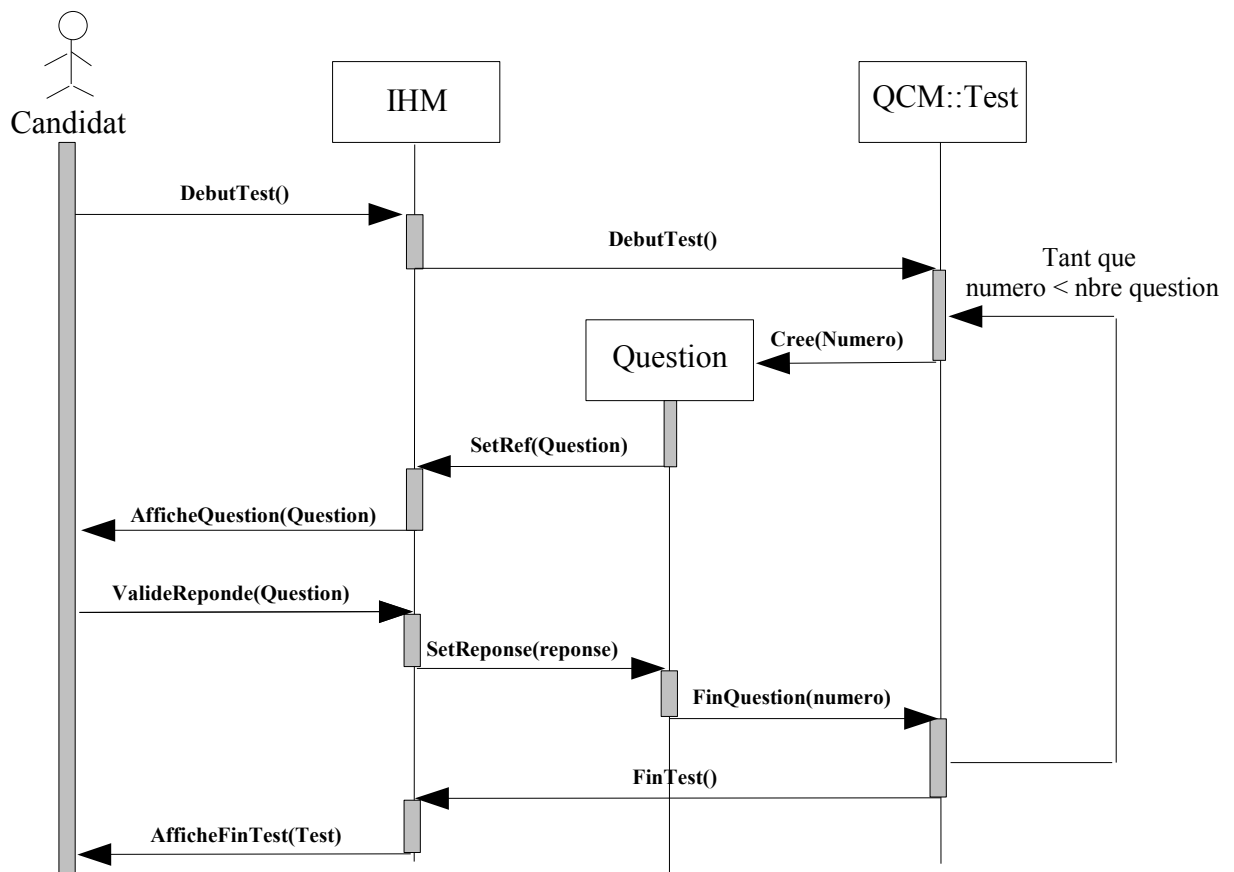


4.2.2 *La réalisation d'un test*

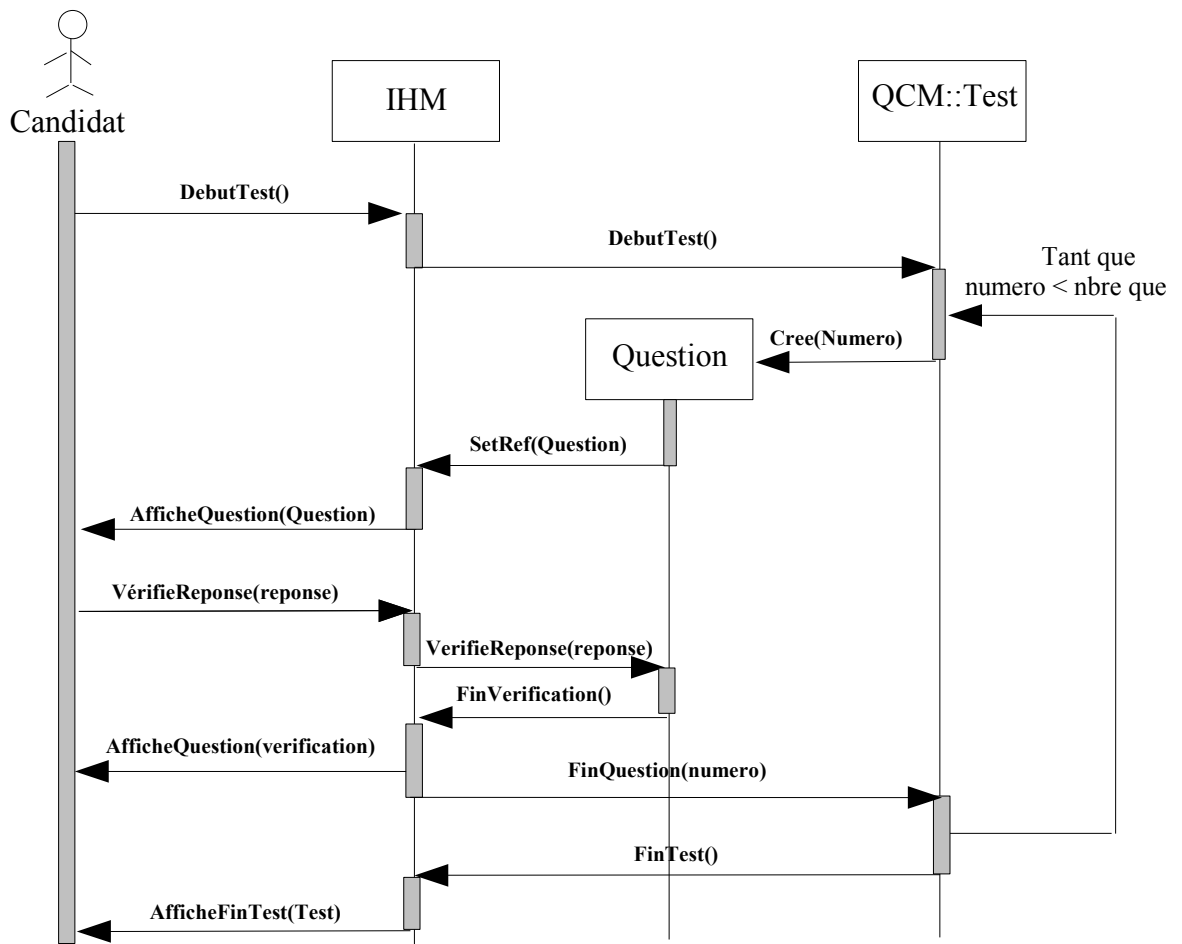
a) *Demande et réalisation de test (modèle générique)*



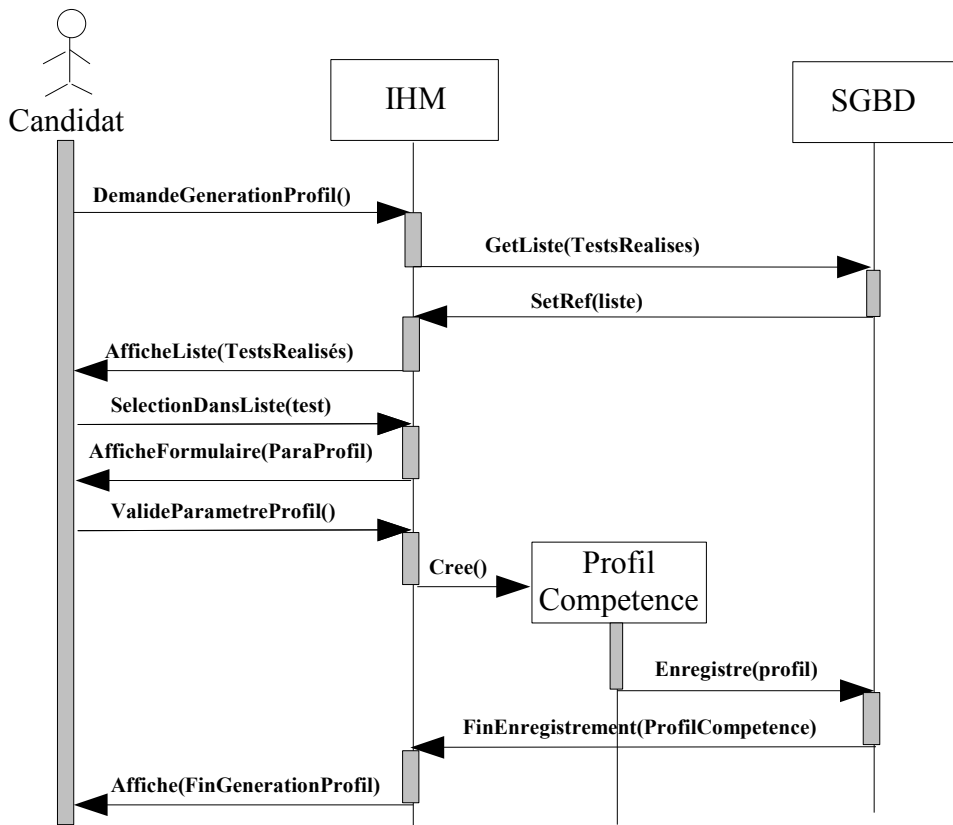
b) Réalisation d'un QCM



c) Variante de réalisation avec vérification de la réponse

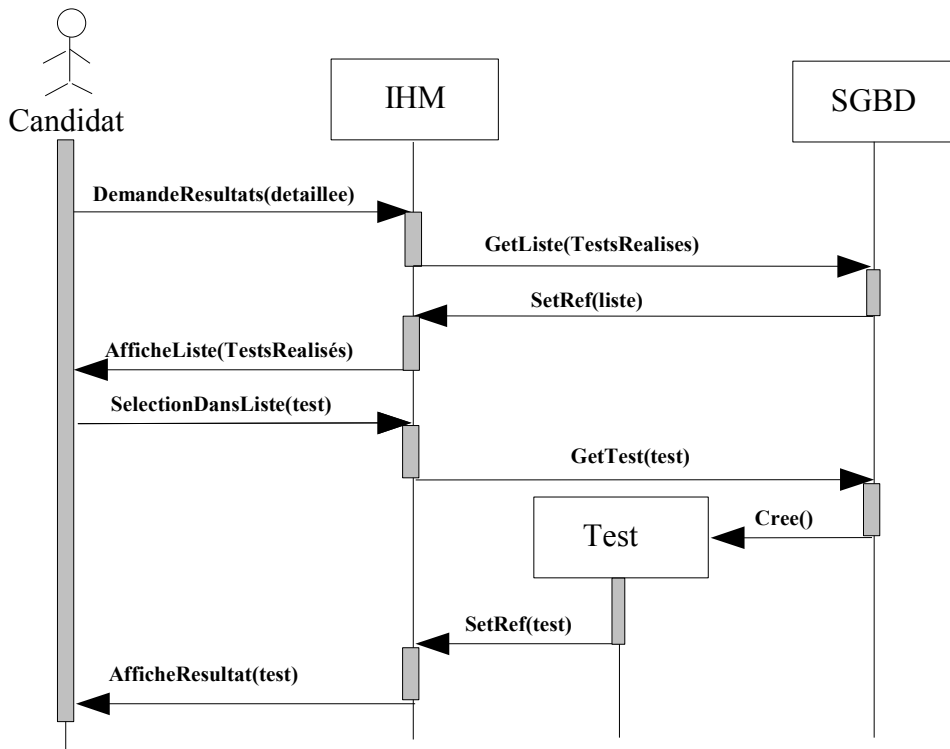


4.2.3 *La génération d'un profil de compétence*

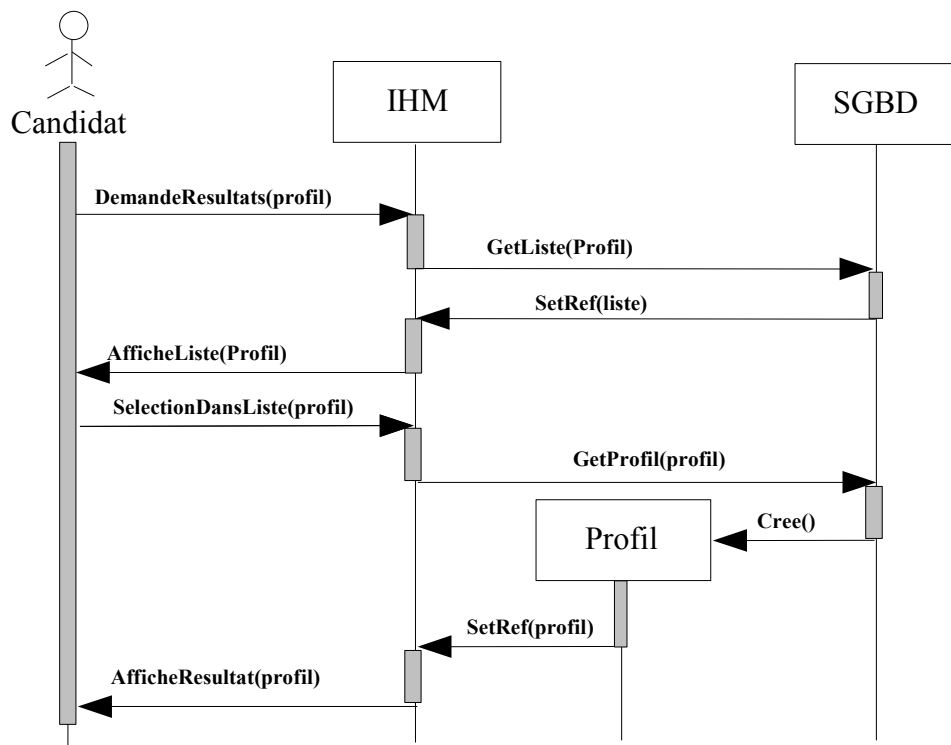


4.2.4 L'affichage des résultats

a) Résultats détaillés



b) Profil de compétence



4.3 Synthèse

5 Définition de l'architecture

La définition de l'architecture consiste à lister et décrire les différents objets de l'application et leur différentes solutions de déploiement.

5.1 L'architecture logicielle de l'application

Le type de l'architecture retenu est une architecture multi agent du type MVC qui permet de réaliser une coupure très forte entre le noyau fonctionnel et les différents agents qui implémentent les interacteurs graphiques (formulaires). La modularité permet de réaliser des développement itératif court et réaliser par des équipe différents mais aussi de privilégier la réutilisation des composants développer.

L'architecture retenue se décompose de la manière suivante :

- des formulaires graphiques(IHM)
- un noyau fonctionnel constitué de plusieurs modules
- une base de données (fichier ou binaire) local ou déportée

5.1.1 Le noyau fonctionnel

a) Son rôle et sa composition

Son rôle est de mémoriser l'état du système est de proposer aux formulaires un accès aux données des tests d'une évaluation. Il garantie la cohérence dans l'enchaînement des appel des méthodes.

Les objets qu'il contient sont les suivants :

- Candidat : gère l'identité et l'identification d'un candidat

- SGBD : système de gestion d'une base de donnée texte ou binaire avec une localisation locale ou déportée
- Config : outils de gestion des paramètres à partir d'une fichier configuration texte

b) Son interface

Son interface est constitué de variables qui reflète son état et de méthodes qui permettent de le faire évaluer.

Variables d'état du noyau fonctionnel :

Identifie : Indique si un candidat a été identifié correctement.

Administrateur : dans le cas ou Identifie est vrai, cette variable indique si le candidat est un administrateur des base de données d'évaluation ou pas

Méthodes du noyau fonctionnel :

LanceEvaluation() : Cette méthode permet en fonction du choix de l'évaluation réaliser de charge la base de test et de lancé le formulaire capable de gérer le type de test correspondant

5.1.2 Les formulaires graphiques

Les formulaires graphiques réalise l'interaction entre l'acteur et le noyau fonctionnel. D'une manière générale, ils sont constitués de deux ensembles. Un ensemble lié à leur rôle respectif et un autre, nommé navigation, qui permet indépendamment du déroulement classique de l'évaluation de naviguer d'un formulaire à l'autre et aussi à l'intérieure d'un même formulaire si celui-ci l'autorise. Cette ensemble de navigation propose systématiquement l'accès à l'aide du logiciel et la sortie du logicielle.

Les paragraphes suivants précises la nature et le rôle des intérateurs présents sur chaque formulaire.

a) L'identification

Le formulaire d'identification a pour rôle d'authentifier le candidat par son nom et son prénom. Il est donc composé de deux champs texte pour accueillir le nom et le prénom.

La casse n'est pas discriminante dans la saisie du nom et du prénom.

Option souhaité : saisie automatique en fonction des identification précédente

b) La sélection d'évaluation

Une fois le candidat identifié celui-ci à la possibilité de sélectionner une évaluation. Celles-ci sont regroupés de diverses manières selon l'objectif de l'évaluation réalisée.

Les différentes catégories de regroupement correspondent en partie aux critères d'association des tests dans la base de données. Ainsi les catégorie retenues sont les suivantes :

Domaine : cette catégorie propose des évaluations regroupant tous les tests associé à un domaine. Pour des exemple de domaine voir le paragraphe XX

Thème : même principe que pour le domaine avec la différence que les tests sont regroupés par leur association de thème. Il s'agit de la catégorie de test la plus utilisée. Elle sera présenté en première dans l'ordre d'apparition des catégories.

Automatique : Dans cette catégorie, des évaluations faisant appel à des formulaires particuliers sont accessibles. Il s'agit pour la plupart de test à correction automatique.

Personnalise : dans cette dernière catégorie tous types de test (automatique, QCM) peuvent être contenu dans l'évaluation. Par contre leur regroupement n'est pas sélection n'est pas automatique mais réaliser par un conseiller. On trouve donc dans cette catégorie des évaluations personnalisées pour un candidat ou pour une catégorie de candidat.

c) Réalisation d'un test du type QCM

Il s'agit d'un formulaire de question à choix multiple. Il présente donc une question et plusieurs propositions de réponses. Une fois les réponses sélectionnées l'utilisateur a la possibilité de valider sa réponse puis et de passer automatiquement à la question suivante, soit de vérifier sa réponse et de passer automatiquement à la question suivante.

Des champs textes sont utilisés pour présenter la question et les propositions de réponses et à chaque proposition de réponse est associé une case à cocher.

Un numéro de question est indiqué afin d'indiquer la progression dans l'évaluation et de faciliter la communication entre candidat évalué et le conseiller évaluateur.

Le bouton précédent de la barre de navigation dans ce formulaire offre la possibilité de re-visualiser les questions précédentes. Le bouton suivant permet de revenir vers la question courante mais ne permet pas d'aller visualiser les questions auxquelles le candidat n'a pas encore répondu. Lorsque la navigation demande la question précédente à la première, la commande est interprétée comme une demande de retour au formulaire de sélection d'évaluation.

d) Le résultat de l'évaluation

Ce formulaire présente le résultat du test réalisé sous forme textuelle. Pour les QCM il s'agit du nombre de bonnes réponses par rapport au total des questions que le test comprend.

Afin d'accéder à l'interprétation des résultats de l'évaluation, le formulaire présente un bouton nommé profil de compétence.

La navigation sur ce formulaire n'autorise que le retour au formulaire de sélection d'évaluation.

e) Le profil de compétence

C'est un formulaire qui présente une interprétation des résultats de test. Il est composé de trois catégories qui sont domaine, objectif fonctionnel et thème qui contiennent chacune les éléments suivants :

un diagramme en étoile dont chaque axe peut représenter un domaine, un thème ou un objectif fonctionnel. À chaque axe est superposée une barre de couleur dont la longueur est proportionnelle à la note obtenue au test réalisé.

une légende qui indique pour chaque couleur d'axe le nom de celui-ci et le score exact obtenu.

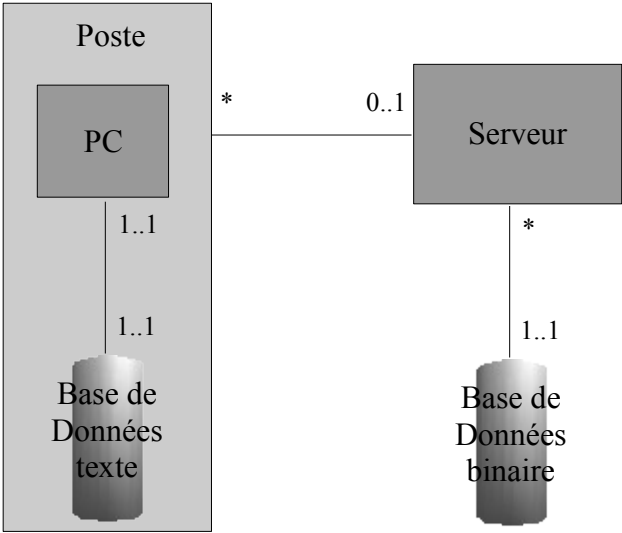
Quatre boutons de sélection qui permettent de visualiser le diagramme étoile correspondant au niveau sélectionné. Les quatre boutons correspondent aux niveaux débutant, moyen, avancé et tous confondu (sorte de moyenne...)

Une quatrième catégorie nommée bilan contient une synthèse textuelle de l'interprétation des résultats des tests. Elle indique des renseignements sur le candidat, sur la nature des tests réalisés et sur la répartition en niveau des tests réalisés.

La navigation sur ce formulaire n'autorise que le retour au formulaire de résultat d'évaluation.

5.2 Le déploiement de la solution

L'utilisation d'une base de données permet deux configurations de déploiement de la solution. Un déploiement (local) sur poste de travail et un déploiement (réseaux) de l'application sur plusieurs postes de travail avec une base de données sur serveur. La sélection du mode de déploiement est réalisée par lecture de paramètre de configuration du logiciel. L'accès aux données réseaux ou locale sera réalisé au travers de module de gestion de base de données du logiciel.



CHAPITRE 3 : CONCEPTION DÉTAILLÉE ET IMPLÉMENTATION

Ce chapitre détail l'implémentation de la solution retenue lors de l'analyse réalisée précédemment. Elle présente chaque classe d'objet identifié par l'analyse sous la forme de module réalisant leur implémentation ainsi que leur interaction.

1 Module : profil de compétence

1.1 Principe de fonctionnement

Le profil de compétence est généré à partir de résultats de tests réalisés. Pour chaque test un score est recueilli selon le domaine, les thèmes et les objectifs fonctionnels associées. Les questions ayant chacun un niveau, le score selon chaque niveau est calculé. Le profil de compétence est donc un ensemble de note résultat d'un test constitué des éléments suivants :

Pour chaque niveau : niveau 1, niveau 2, niveau 3 et niveau inconnu
Pour chaque : domaine, thème, objectif fonctionnel
% de compétence

1.2 Son implémentation

Le calcul de note et du total de question relative à un ensemble est réalisé par la classe **Notation** à agréger au classe **Domaine**, **Theme** et **ObjFonct**. Une interface **Profil** permet de lister les opérations de base d'un profil de compétence soit l'enregistrement et la génération du profil. Un objet graphique permettant de réaliser l'affichage du profil est implémenté sous la forme d'un contrôle utilisateur nommé **DiagrammeEtoile**.

1.2.1 *L'interface Profil*

L'interface Profil sert à lister les méthodes et les propriétés commune à tout Profils de compétence. Elle contient les éléments suivants :

- SetRefQuestionRéalises()
- Enregistre()
- Genere()

1.2.2 *La classe Notation*

La classe Notation est agrégée à chaque ensemble et permet de réaliser un calcul cumulatif du score obtenu pour chaque élément de chaque ensemble.

Pour chaque niveau possède une note et un nombre de tests (questions, exercices).

Par exemple : toute les questions ayant pour **Domaine** : Multimédia verront leur note se cumuler selon le niveau au quelle elles appartiennent. Le Profil de compétence sera donc capable de restituer un pourcentage de compétence selon le **Domaine Multimédia** selon chacun des niveaux.

Une notation mixte est aussi obtenue par la fusion des niveaux selon une pondération relative à leur complexité. Ainsi le niveau 2 et 3 auront une pondération respective de leur niveau de 2 et de 3.

1.2.3 *Le contrôle DiagrammeEtoile*

Il permet de réaliser un affichage graphique des résultats sous la forme de diagramme en étoile.

Chaque ensemble représente une branche du diagramme que nous nommons axe. La dimension de l'axe est proportionnel au résultat obtenu.

Le contrôle fournit donc la possibilité d'afficher pour chaque domaine, thème et objectif fonctionnel, et pour chaque niveau, un diagramme en étoile.

Chaque axe est représenté selon une couleur différente et est associé à une légende qui nomme l'axe textuellement et qui indique le pourcentage de réussite obtenue.

L'implémentation des axes est réalisé par une classe **Axe** à l'intérieur du contrôle qui autorise l'ajout d'un note selon son niveau.

a) Mise en évidence de détails par survol de région(Axe, légende)

b) Accès aux résultats détaillés

1.3 Implémentation particulières

a) Les exercices à correction automatique

Les axes représentent des critères de notations.

1.4 Impression du profil de compétences

Cette fonctionnalité permet l'impression d'un document relatant les informations suivantes :

- Informations sur le candidat (table Candidat)
- Information sur l'évaluation réalisée
- Résultats de l'évaluation en diagramme étoile et en textuelle

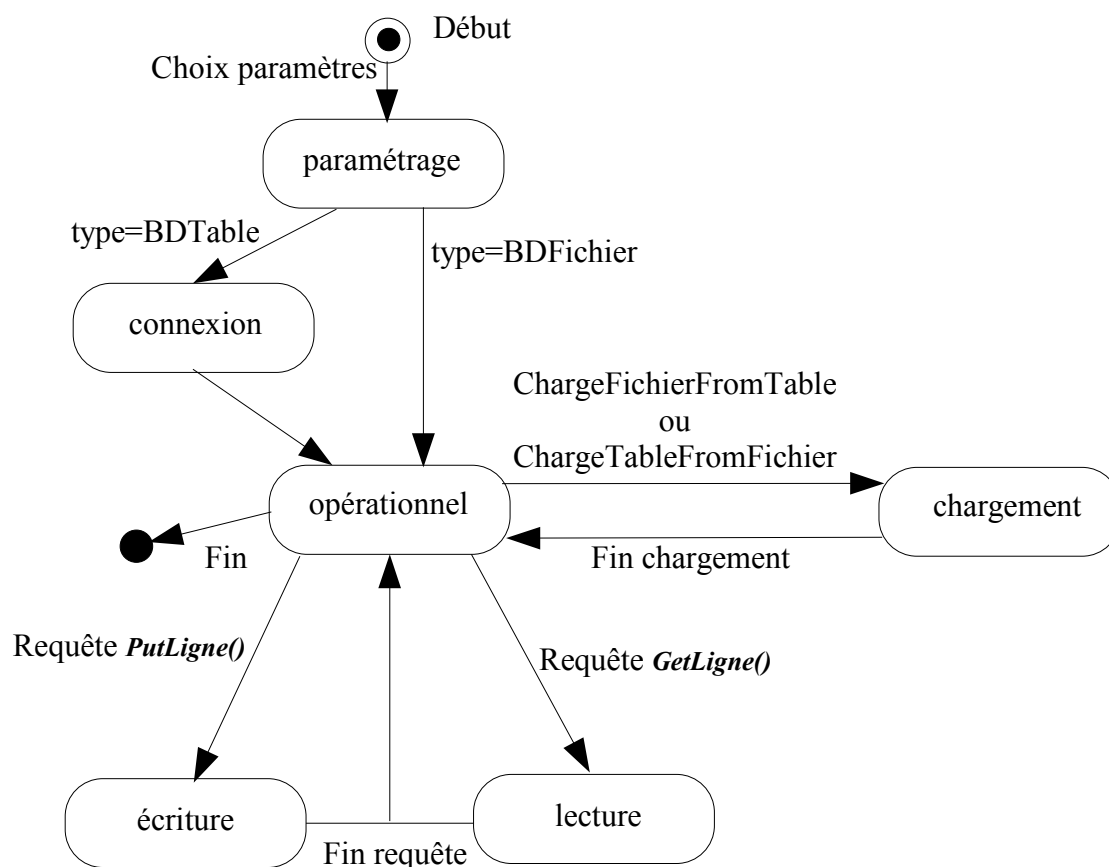
2 Module : base de données

2.1 Rôle

Capable d'interagir avec une base de données de fichier texte ou avec une base de données relationnelle Objet PostgreSQL.

2.2 Son fonctionnement

Le module de base décrit le diagramme d'état suivant :



2.3 Ses opérations

2.3.1 *GetLigne*

Permet de récupérer un tableau de d'enregistrement en fonction de critères de sélection passés en arguments. Les données sont lues dans la table ou dans un fichier en fonction de la configuration du SGBD.

2.3.2 *PutLigne*

Ajoute dans le fichier ou la table un nouvel enregistrement.

2.4 Conception détaillée

2.4.1 *Association table fichier ID_BD*

La référence aux données d'une table ou d'un fichier se fait par un identifiant unique appelé ID_BD. A cette identifiant est associé un nom de fichier et un nom de table. Cette identifiant est communiqué à la classe manipulant ses données.

Exemple pour la classe Candidat :

ID_BD	Nom Fichier	Nom Table
1	Candidat.bd	candidat

2.4.2 *Recherche simple d'enregistrement (Sélection et Projection)*

La recherche est paramétrée par les arguments non vide passés en argument dans la chaîne de caractère. A chaque argument non vide correspond donc la valeur à rechercher pour l'argument sous

entendu.

Les champs des sous chaîne de caractère sont séparés par le caractère de tabulation TAB (vbtab en Visual Basic).

Actuellement, seul deux type de combinaison de question sont implémentés. Le OU pour exprimer plusieurs valeur occurrente pour une attribut donnée et le ET qui indique la combinaison de valeur d'attribut occurrentes.

a) Le <exp1> OU <exp2>

Pour un attribut donné, plusieurs valeurs occurrente peuvent être indiquer. Pour pour les séparer le caractère « | » est utilisé.

Exemple :

Question = <> tab<> tab <val1| val2|val3> tab <>

b) Le <exp1> ET <exp2>

Le ET est implémenté de manière implicite dans la construction d'une question par le fait d'introduire une chaîne de caractère non-nulle pour un attribut.

Une question implémentant le ET entre attribut se réalise de la manière suivante.

Exemple :

Question = <> tab<> tab <le nom a rechercher dans l'attribut nom> tab <>

2.4.3 Recherche complexe d'enregistrement

a) Opérations unaires avec rangement des nuplets

La module doit être capable d'effectuer des opérations unaires et de les enchaîné avec un rangement des nuplets de deux manières par ordre **croissant** ou **aléatoire**.

L'enchaînements d'opération unaires doit réaliser les opération suivantes :

- sélection d'attribut par valeurs
- projection sur un attribut
- rangement des nuplets

Exemple de recherche de Test :

Recherche de Test par Id ou Niveau puis un classement aléatoire ou ordonnée par valeur croissante de l'attribut sujet de la projection (des Ids)

b) Sélection par valeur avec borne

Possibilité de définir une plage de valeur pour réaliser une sélection borné

Exemple recherche de test réalisé :

Recherche de test réalisé pendant une durée

c) Opérations binaires avec rangement des nuplets

Le

Exemple de recherche de Tests réalisés :

Recherche de tests non réalisé par un candidat

2.4.4 *Transformation des requêtes*

a) *Chaîne de caractère uniforme*

Tout en minuscule

Sans espace avant et après

b) *Nombre en entier*

Afin d'améliorer les recherches certains attributs sont stockés sous forme d'entier. Pour distinguer les arguments entiers des argument chaîne de caractère, une méthode de transformation de chaîne de caractères en entier est réalisée.

Une évolution consiste à supporter la transformation de chaîne de caractère en double.

2.4.5 *Connexion*

a) *Base de données SQL*

b) *Base de fichier .bd*

Lors d'une connexion à une base de fichiers deux cas se présente. Premièrement la base de fichiers n'existe pas. Dans ce cas, tous les fichiers lister dans le tableau d'IDBD doivent être créés. Dans le second cas, on vérifie l'existence de tous les fichiers listés.

2.5 Description des tables/fichiers

2.5.1 *Données principales*

table candidat

- Id candidat
- nom
- prénom
- Année naissance
- Adresse
- Code postal
- Ville
- Téléphone
- email

table domaine

- Id domaine
- nom
- description

table thème

- Id thème
- nom
- description

table objectif fonctionnel

- Id objectif fonctionnel
- nom
- description

table Question

- Id Question
- Type de QCM(unique ou multiple)
- niveau
- question
- solution
- nombre de proposition
- liste de proposition[1..9]

table TOQCM

- Id thème
- Id objectif fonctionnel
- Id Question

table DT

- Id domaine
- Id thème

table Test Réalisé

- Id Candidat
- Id Test
- réponse
- score

table d'association évaluation type QCM

- Id Evaluation type
- Id Question

2.5.2 Données particulières

table frappe

- Id
- Niveau
- Texte
- Solution

table clavier

- Id
- niveau
- Texte
- Solution

3 Module : gestionnaire de fichier de configuration

3.1 Définition de la syntaxe du fichier

3.1.1 Caractère de commentaire

Si le caractère « # » est le premier rencontré sur une ligne alors le reste de la ligne est considéré comme du commentaire

3.1.2 Définition d'une section

Une section est caractérisée par une chaîne de caractère contenu entre 2 crochets. La section se termine lorsque le parseur découvre une nouvelle section.

exemple : [main] définit la section « main »

3.1.3 Déclaration de valeur de variable

La déclaration de valeur de variable se compose de la manière suivante :

{Nom de la variable}={Valeur de la variable}

Le nom et la valeur sont compris entre des accolades. Cette syntaxe permet de définir des noms et des valeurs constitués de caractères d'espacement.

3.2 Les fonctionnalités

3.2.1 L'initialisation

Lors de l'initialisation du module deux cas se présente soit un fichier de configuration existe déjà, dans ce cas le logiciel se sert de celui-ci, ou bien aucun fichier existe et dans ce cas le module crée un fichier de configuration.

3.2.2 La lecture d'une variable

Cette fonctionnalité permet de lire une valeur de variable contenu dans le fichier de configuration en connaissant son nom et le nom de sa section. Le module réalise une recherche de la variable dans le fichiers parmi toutes les variables d'une section. Il est donc possible que le fichier de configuration possède des variables du même nom.

3.2.3 La création ou le positionnement d'une variable

Lors de l'association d'une valeur à un nom de variables deux cas se présentent. Le premier cas est celui ou le nom de variable n'existe pas encore dans le fichier de configuration. Dans ce cas, le nom de variable et sa valeur sont ajoutés à la section. Dans le second cas, le nom de variable existe déjà .Dans ce second cas, seul la valeur de la variable est mis à jour.

4 Module : charte graphique du logiciel

4.1 Les boîtes de dialogues

Décomposition en deux type de messages : les avertissements et le erreurs. Ceux deux types de messages se décomposent en une image symbolisant l'erreur, un nom qui fait référence à la tâche ou au module en cause (en gras) et un message qui décrit sommairement l'erreur rencontré et la procédure à suivre le cas échéant (normale).

4.1.1 Les erreurs

Les erreurs par définition indique une incapacité du système à réaliser une tâche. Sa signalisation indique l'incapacité du système à poursuivre la tâche en cour de réalisation et prévient l'utilisateur que celle-ci est annulé. A la fermeture de la boite de dialogue. L'utilisateur est ramené dans l'état stable du système précédent.

4.1.2 Les avertissements

Ils indiquent à l'utilisateur que la tache se réalise dans un scénario exceptionnel ou limite.

Autrement dit, elle est non bloquante.

4.2 Font et couleurs

Pour tout les dessins : font Arial (font). Choix d'une font sans shérif pour améliorer la lisibilité de la légende.

Jeu de dégradé d'orange dans l'expression du niveau de difficulté. Plus la couleur d'une évaluation est foncé plus la difficulté est accru.

4.3 Navigabilité

4.3.1 Règle et intérêt

La charte graphique définit la représentation des fonctions de navigation afin d'homogénéiser la désignation de la fonction. C'est le cas des fonctions « précédent » et « suivant » qui dans le cas de la navigation inter formulaire ou intra formulaire peuvent se représenter de manière identique alors qu'en fonction du formulaire, elle ne réalise pas le même effet.

Une navigation homogène

La navigabilité au sein des interfaces est assurée par un barre de navigation présente sur tous les formulaires. Celle-ci permet de réaliser les fonctions « suivant », « précédent », « quitter » et « aide » à chaque instant.

La représentation choisie pour illustrer les fonctions de navigation correspond en partie à est celle observer dans la navigation Internet.

Les pictogrammes retenues pour la navigation sont les suivants :

<i>Fonction de navigation</i>	<i>Description du pictogramme</i>
Précédent	Flèche vers la gauche
Suivant	Flèche vers la droite
Quitter	Une croix à 45 degrés
Aide	Un point d'interrogation

La localisation de la barre de navigation se situe au bas de chaque formulaire.