

الجزائرية الديمقراطية الشعبية  
République Algérienne Démocratique et Populaire  
وزارة التعليم العالي والبحث العلمي  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



N° Réf :.....

## Centre Universitaire de Mila

Institut des sciences et de la technologie

Département de Mathématiques et Informatique

**Mémoire préparé En vue de l'obtention du diplôme de licence  
en: - Filière informatique général**

## Thème

**Conception et réalisation d'un site web dynamique  
d'une Auto-école**

**Préparé par** : Chebbat yehya  
Graiche charaf eddine  
Tria zahia

**Encadré par:** abderrezak samira

**Année universitaire :2013/2014**

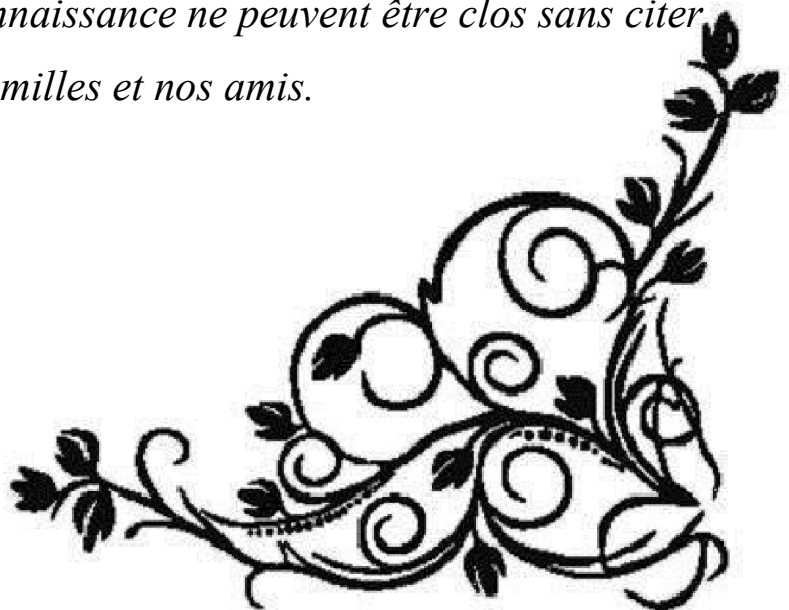


# *Remerciement*

*Grâce à dieu miséricordieux tout puissant  
Qui nous a éclairé le chemin de réussite, et de nous avoir  
Guidé Durant toute notre étude*

*Au terme de ce travail, Nous tenons également à exprimer nos  
sincères remerciements à tous ceux qui ont contribués de loin au de  
prés à l'élaboration de ce projet de fin d'étude surtout Mem  
Abderrezak Samira pour leurs aides qu'ils nous ont fournie pour la  
réalisation de ce travail, qu'il trouve ici l'expression de nos profond  
respect et de nos sincères reconnaissance.*

*Nous remercions et reconnaissance ne peuvent être clos sans citer  
nos familles et nos amis.*





*Dédicace*

*A Nos très chers parents*

*A Nos frères*

*A Nos sœurs*

*A tous ceux qui sont chers :*

*Nos grandes familles, Nos amis(es),*

*A toutes la promotion de l'informatique  
de centre universitaire de Mila*

*2013-2014*

*YAHIA & CHARAF & ZAHIA*



# ***Sommaire :***

## *Introduction générale*

### **Chapitre 01 : technologie web**

Introduction : .....	1
1. Internet : .....	1
1.1. Définition d'internet: .....	1
1.2. Histoire d'internet : .....	2
2. World Wide Web : .....	2
2.1. Définition : .....	2
2.2. Le cycle de vie d'un site Web : .....	3
2.3. Les technologies du Web : .....	3
2.3.1. L'architecture Client/Serveur : .....	3
3. Les pages web : .....	4
3.1) Les pages web statiques .....	4
3.2) Les pages Web dynamiques .....	5
3.2..1 : Les avantages du site dynamique .....	6
3.3 classification des sites Web : .....	6
Conclusion .....	7

### **Chapitre02 : *Présentation le langage de modélisation UML***

1. Introduction .....	8
2. La modélisation : .....	9
2.1 C'est quoi un modèle? .....	9
2.2 Pourquoi modéliser ? .....	9
2.3 Définition d'UML .....	10
2.4. Caractéristiques d'UML .....	11
3. Présentation des vues et diagrammes d'UML .....	11

3.1 Diagrammes structurels ou diagrammes statiques.....	11
3.2 Diagrammes comportementaux ou diagrammes dynamiques.....	12
3.3. Diagramme de cas d'utilisation.....	14
3.4. Diagramme de classe.....	17
3.5. Diagramme d'activité.....	20
3.6. Le diagramme de séquence/séquence système .....	21
3.7. Le diagramme de classe participante .....	24
3. Processus Unifié (UP) .....	25
3.1 Définition .....	25
3.2 Principe : .....	25
4. Conclusion : .....	27

## **Chapitre 03: Analyse des besoins et la conception**

<b>1.Introduction .....</b>	<b>29</b>
Partie 1 : Identification des besoins.....	30
1-Elaboration du cahier des charges : .....	30
1.1. Présentation du système:.....	30
1.2. La définition des grands choix techniques :.....	30
1.3. Les exigences fonctionnelles : .....	30
1.4. Les besoins techniques.....	30
1.5. La description du contexte du système .....	31
1. Modèle de domaine .....	54
2. Diagramme de classe participante : .....	55
2.1 :s'authentifier : .....	55
2.1.1 : S'authentifier gérant : .....	55
2.1.2 : S'authentifier candidat/membre : .....	55
2.2-inscrire : .....	56
2.3-passer test : .....	57
2.4-ajouter cour : .....	57
2.5-ajouter annonce .....	58

2.6-supprimer annonce :	58
2.7-criée un compte candidat :	59
2.8-supprimer un compte candidat :	60
2.9-modifier candidat :	61
2.10-ajouter les résultats :	61
3-Diagramme d'activité navigation :	62
3.1-authentifier :	62
3.2-inscrire :	63
3.3-passer test :	64
3.4-ajouter cour :	65
3.5-ajouter annonce :	66
3.6-supprimer annonce :	67
3.7-criée un compte candidat :	68
3.8-supprimer un compte candidat :	69
3.9-modifier un compte candidat :	70
3.10-ajouter résultat :	71
1. Introduction	72
2. Diagrammes d'interactions	72
2.1-s'authentifier:	73
2.2-inscrire :	74
2.3-passer test :	75
2.4-ajouter cour :	76
2.5-ajouter annonce :	77
2.6-supprimer annonce :	77
2.7-criée un compte candidat :	78
2.8-supprimer un compte candidat :	79
2.9-modifier un compte candidat :	80
2.10-ajouter résultat :	81
3. Diagramme de classes métiers	82
Conclusion:	82

## Chapitre 04 : implémentation

Introduction .....	84
1. Outils de travail : .....	84
Les Langages de programmation .....	84
1.1/ Le langage PHP (Personnel Home Page) .....	84
1.2. MYSQL : .....	85
2. Les outils de développement : .....	86
2.1. Dreamweaver : .....	86
2.2. Wamp Server: .....	87
2.3. Adobe Photoshop .....	87
2.4. PHPMYAdmin : .....	88
3. Le passage du diagramme de classe au modèle relationnel <b>Erreur ! Signet non défini.</b>	
2. Implémentation de la base de données .....	90
2. Organigramme générale du site: .....	92
3. Présentation de quelques pages de l'application .....	93
3.1 La page d'accueil : .....	93
3.2 Page contactez-nous .....	94
Page d'accueil gérant .....	95
Page ajouter annonces .....	96
Ajouter les résultats .....	97
Mise à jour des comptes .....	98
Page d'accueil visiteur .....	99
Page inscription .....	100
Page d'accueil membre : .....	101
Page télécharger livre : .....	102
Page passer test .....	102
Passé test1 .....	104
Page Passé test2 : .....	106
Page d'accueil candidat .....	107
Page résultats .....	108

Conclusion : ..... 108

*Conclusion générale*

*References bibliographies*

# Liste des figures

## Chapitre 01 : technologie web

Figure 1.1 : L'internet .....	1
Figure I.2.: Les pages web statiques.....	4
Figure I.3.: site web dynamique.....	5
Chapitre 02 :Présentation le langage de modélisation UML	

Figure 2.1 image développeur UML.....	9
Figure 2.2 : historique UML[1].....	9
Figure 2.3: Schéma de la hiérarchie des diagrammes d'UML 2.0 .....	13
Figure2.4: Représentation d'un acteur .....	14
Figure2.5 : représentations de cas d'utilisation. ....	14
Figure2.6 : représentations d'une note .....	15
Figure2.7 : représentation de relation d'association. ....	15
Figure2.8 : représentations de relation d'inclusion. ....	15
Figure2.9 : représentations de relation d'extension.....	15
Figure 2.10: relation de généralisation.....	16
Figure2.11: représentations d'une classe.....	17
Figure2.12 : Relation d'association. ....	17
Figure 2.13: représentation relation d'agrégation.....	18
Figure 2.14: représentation relation de composition.....	18
Figure2.15 : représentation relation de généralisation.....	19
Figure2.16 : fragment Alt.....	22
Figure2.17 : fragment opt.....	22
Figure2.18 : schéma qui présente fragment loop .....	23
Figure 2.19 : Principe et phase du UP.....	25

## Chapitre03 : Analyse des besoins et la conception

Figure : Diagramme de contexte dynamique .....	36
Figure 3.1 : Diagramme de cas d'utilisation.....	37
Figure 3.2 : Diagramme de séquence du cas d'utilisation « S'authentifier».....	47
Figure 3.3 :Diagramme de séquence du cas d'utilisation «consulter les information»...48	

Figure 3.4 :Diagramme de séquence du cas d'utilisation «consulter les annonces».....	48
Figure 3.4 : Diagramme de séquence du cas d'utilisation « S'inscrire».....	49
Figure 3.5 :Diagramme séquence système «passer test». ....	50
Figure 3.6 : Diagramme de séquence système «télécharger cour».....	51
Figure 3.7 :Diagramme de séquence système «consulter les résultats d'examen».....	51
Figure 3.9 : Diagramme de séquence système «modifier cour».....	52
Figure 3.10 :Diagramme de séquence système «modifier les information». ....	53
Figure 3.11 :Diagramme de séquence système «ajouter des annonces».....	53
Figure 3.12 :Diagramme de séquence système «supprimer des annonces».....	54
Figure 3.13 :Diagramme de séquence système «criée un compte candidat».....	54
Figure 3.14 : Diagramme de séquence système «supprimer un compte candidat».....	55
Figure 3.15 :Diagramme de séquence système «modifier un compte candidat».....	55
Figure 3.16 :Diagramme de séquence système «Ajouter résultat».....	56
Figure 3.17 :Modèle de domaine.....	58
Figure 3.18 :Diagramme de classes participantes « S'authentifier gérant».....	59
Figure 3.19 : Diagramme de classes participantes «S'authentifie candidat/membre»....	59
Figure 3.20 : Diagramme de classes participantes du cas d'utilisation «S'inscrire».....	60
Figure 3.21 : Diagramme de classes participantes du cas d'utilisation «passer test».....	61
Figure 3.22 :Diagramme de classes participantes du cas d'utilisation «ajouter cour». ...	62
Figure 3.23 :Diagramme de classes participantes «ajouter annonce». ....	62
Figure 3.24 : Diagramme de classes participantes «supprimer annonce».....	63
Figure 3.25 : Diagramme de classes participantes «criée un compte candidat».....	63
Figure 3.26 :Diagramme de classes participantes «supprimer uncompte candidat». ....	64
Figure 3.27 :Diagramme de classes participantes «modifier un compte candidat». ....	64
Figure 3.28 : Diagramme de classes participantes «ajouter résultat». ....	65
Figure 3.29 :Diagramme d'activité du cas d'utilisation « s'authentifier». ....	66
Figure 3.30 :Diagramme d'activité du cas d'utilisation « s'inscrire».....	67
Figure 3.32 :Diagramme d'activité du cas d'utilisation « ajouter cour».....	69
Figure 3.33Diagramme d'activité du cas d'utilisation « ajouter annonce».....	70
Figure 3.34 :Diagramme d'activité du cas d'utilisation « supprimer annonce».....	71
Figure 3.35 :Diagramme d'activité du cas d'utilisation « crée un compte».....	72
Figure 3.36 :Diagramme d'activité du cas d'utilisation « supprimer un compte». ....	73
Figure 3.37 :Diagramme d'activité du cas d'utilisation « modifier un compte».....	74
Figure 3.38 :Diagramme d'activité du cas d'utilisation « ajouter résultat».....	75

Figure 3.39 : Diagramme d'interactions du cas d'utilisation « S'authentifier».....	77
Figure 3.40 : Diagramme d'interactions du cas d'utilisation « inscrire» .....	78
Figure 3.41 :Diagramme d'interactions du cas d'utilisation « passer test».....	79
Figure 3.42 : Diagramme d'interactions du cas d'utilisation « ajouter cour» .....	80
Figure 3.43 :Diagramme d'interactions du cas d'utilisation « ajouter annonce».....	80
Figure 3.46 : Diagramme d'interactions « supprimer un compte candidat».....	82
Figure 3.47 :Diagramme d'interactions « modifier un compte candidat» .....	83
Figure 3.48 :Diagramme d'interactions du cas d'utilisation « ajouter résultat».....	84
Figure 3-49 : Le diagramme de classes métiers.....	85

## *Chapitre 04 : implémentation*

Figure 4.1 : Dreamweaver version 8.0 .....	86
Figure 4.2 :Adobe Photoshop version:8.0 .....	87
Figure 4.3 : PHPMYAdmin version 5.3.5.0 .....	88
Figure 4.4 : structure de la base de données.....	90
Figure 4.5 : <i>organigramme du site</i> .....	92
Figure 4.6 : <i>page d'accueil</i> .....	93
Figure 4.7 : Page contactez-nous .....	94
Figure 4.8 : Page d'accueil gérant .....	95
Figure 4.9 : Page ajouter annonces .....	96
Figure 4.10 : Page ajouter résultats.....	97
Figure 4.11 : Page Mise à jour des comptes.....	98
Figure 4.12 : Page d'accueil visiteur .....	99
Figure 4.13 : Page d'inscription.....	100
Figure 4.14 : Page d'accueil membre .....	101
Figure 4.15 : Page télécharger livre.....	102
Figure 4.16 : Page passer test.....	103
Figure 4.17 : Page passer test1.....	104
Figure 4.18 : Page passer test2.....	106
Figure 4.19 : Page d'accueil candidat .....	107
Figure 4.20 : Page résultats. ....	108



# *Introduction générale*

## **Introduction générale :**

Aujourd'hui, l'utilisation de l'Internet dans les différents domaines (économique, banque, commerce...) est devenu très nécessaire. Le but de l'Internet n'est plus comme à l'origine, de transmettre quelques lignes de messages, mais d'échanger des documents électroniques, des données informatisées, des informations dans tous les domaines .

Dans le domaine commercial , l'Internet se présente comme un outil d'information de tout premier plan, ainsi que des nouveaux aspects et des nouvelles idées ont apparu dans ce domaine.

Notre travail consiste à développer un site web dynamique pour une auto-école, permettant la communication entre le gérant, les moniteurs et les candidats de l'auto-école.

Le site développé permettra d'obtenir des informations générales sur l'auto-école, de voir les annonces concernant les examens, les cours, les nouveautés des lois de conduite et toutes autres utilités. Il permet aussi de télécharger des cours et des livres concernant le code de la route et la conduite. Le site donnera au candidat la possibilité de passer des tests préliminaires en ligne (test de code et test de conduite).

Après l'introduction générale expliquant la problématique à résoudre, notre mémoire est ensuite organisé en quatre chapitres :

**Le chapitre I** : Intitulé « technologie Web », résume les concepts fondamentaux sur lesquels est basée cette technologie.

**Le chapitre II** : présente le langage de modélisation UML ainsi que la démarche simplifiée UP qui servira pour l'analyse et la conception de notre site web.

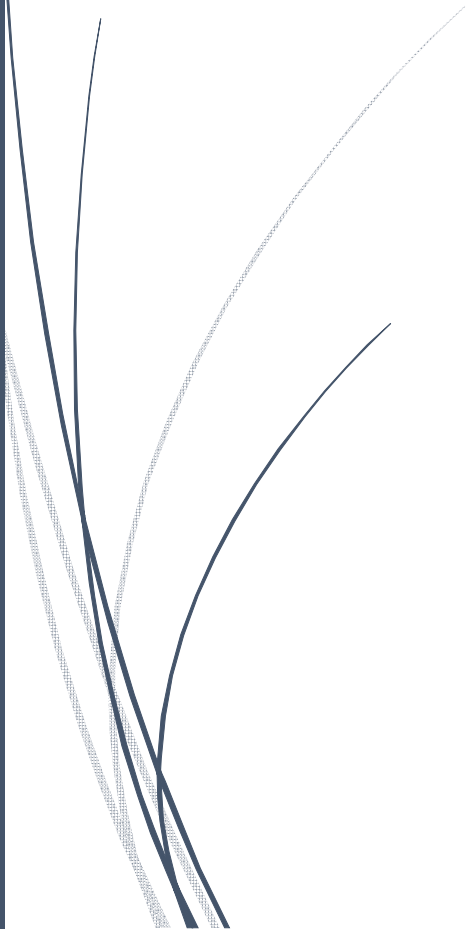
**Le chapitre III** : analyse des besoins et conception de l'application selon la démarche UP.

**Le chapitre IV** : présente les différents langages et outils utilisés dans le développement du site et le passage du diagramme de classe vers le modèle relationnel afin de définir la base de données utilisé dans la réalisation du site. Ainsi, le chapitre expose quelques interfaces de l'application.

Enfin, nous terminons par une conclusion générale qui résume le travail réalisé et envisage les perspectives possibles pour notre système.

A dark blue vertical bar runs down the left side of the page. A blue arrow with a white outline points to the right from the bar, positioned above the chapter title.

# *Chapitre 01: Technologie web*



## **Introduction :**

Dès le début des années 60, plusieurs groupes de scientifiques s'intéressent à la question de la communication entre ordinateurs. A l'époque, ces machines étaient souvent très différentes les unes des autres et la possibilité de communication entre-elles restait une question difficile.

Jusqu'à l'arrivée d'un phénomène qui a marqué le monde de l'informatique .C'est bien celui d'Internet. Ce réseau mondial crée par l'armée américaine, puis utilisé par les chercheurs et autres scientifiques, a connu une croissance phénoménale auprès du grand public avec l'introduction du «World Wide Web».

## 1. Internet :

### 1.1. Définition d'internet:

L'Internet est un système de communication qui permet aux ordinateurs autour du monde de communiquer et de s'échanger de l'information entre eux. Cette communication entre ordinateurs permet plusieurs possibilités et offre une masse d'informations chaque jour plus important dans des domaines comme la médecine, la science et la technologie. [2]



Figure 1.1 : L'internet

### 1.2. Histoire d'internet :

C'est dans les années 60's, en plein milieu de la "guerre froide" entre le bloc de l'est et les pays occidentaux qu'on retrouve les origines de l'Internet. Les militaires américains avaient besoin d'un moyen de coordonner leurs armes nucléaires en cas de guerre atomique. Ils avaient besoin d'un système de communication qui pourrait survivre même si plusieurs bases étaient détruites.

**-1964:** projet ARPANET, pour la création du premier réseau de communication distribué.

Une technologie décentralisée, et pas d'ordinateur principale pour coordonner les communications. Même en cas de panne d'un ordinateur, le fonctionnement du système persiste.

**-1969:** Mise en opération du projet ARPANET qui utilisent un protocole de communication Appelé NCP (Network Control Protocol), assurant une communication sur de longues distances.

**-1973:** les scientifique prennent leur revanche en développant leur propre réseau à travers les universités et en utilisant un protocole de communication amélioré nommé TCP/IP (Transfert Contrôle Protocole /Internet Protocole).les fondation pour l'internet d'aujourd'hui sont placées.

**-1989:** Tim Berner - Lee conçoit la base de ce qui allait devenir le World Wide Web.1995: C'est cette année que le terme "Internet" est entré dans le vocabulaire de la plupart du monde.

C'est surtout à cause d'un "nouveau" service: le World Wide Web (WWW ou W3). Plusieurs personnes confondent encore aujourd'hui le terme WWW avec l'Internet. [2]

## 2. World Wide Web :

### 2.1. Définition :

C'est un ensemble de pages web liées entre elles par des liens hypertextes (un simple clic de souris, nous permet de basculer d'un document web à un autre, tout en appelant des textes, des images, des sons...) format une énorme toile d'araignée mondiale de l'information, qui permet quand on y accroche un fil, de tirer à soi toute la toile (web).

### 2.2. Le cycle de vie d'un site Web :

Le cycle de vie d'un site Web constitué de trois phases principales qui sont :

- **La création** : Correspond à la concrétisation d'une idée qui comporte un grand nombre de phases :
  - La conception, représentant la formalisation de l'idée.
  - La réalisation, correspondant au développement du site Web.
  - L'hébergement, se rapportant à la mise en ligne du site.
  - Le référencement par les moteurs de recherche.
- **L'exploitation** : Correspond à la gestion quotidienne du site, à son évolution et à sa mise à jour. L'exploitation du site englobe notamment les activités suivantes :
  - Assurer un suivi des technologies, du positionnement du site et de celui des concurrents.
  - Promotion et référencement, permettant de développer son audience.
- **La maintenance et la mise à jour** : Représentant l'animation quotidienne du site et le maintien de son bon fonctionnement.

### 2.3. Les technologies du Web :

Le Web repose sur les technologies d'Internet suivantes :

- Le protocole TCP/IP (Transmission Control Protocol/Internet Protocol) pour assurer le transfert de données.
- Le système DNS (domainname system) pour convertir les noms d'hôte en adresse IP.
- Le standard Internet MIME (Multipurpose Internet Mail Extensions) pour indiquer le type des données.

### 2.3.1. L'architecture Client/serveur :

Elle désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs clients du serveur :

- **Le Serveur** : processus qui offre un service au travers du réseau.
- **Le Client** : processus (logiciel) qui demande un service au travers du réseau; c'est à dire programme qui permet à un utilisateur de soumettre des requêtes à un serveur www, et de visualiser le résultat.

### 3. Les pages web :

La page web est l'unité de consultation du web. Ce terme a une signification pratique ; il n'a pas de définition formelle. C'est un document informatique qui peut contenir du texte, des images, des formulaires à remplir et divers autres éléments multimédias et interactifs

#### a) Les pages web statiques :

Les pages web statiques, c'est-à-dire un simple fichier texte contenant du code **HTML**, ils sont très pratiques pour créer un site contenant quelque dizaine de page mais possèdent leurs limites :

- Une maintenance difficile due à l'obligation de modifier chacune des pages.
- L'impossibilité de renvoyer une page personnalisée selon le visiteur.
- L'impossibilité de créer une page dynamiquement selon les entrées d'une base de données. La figure représenté le web statique.

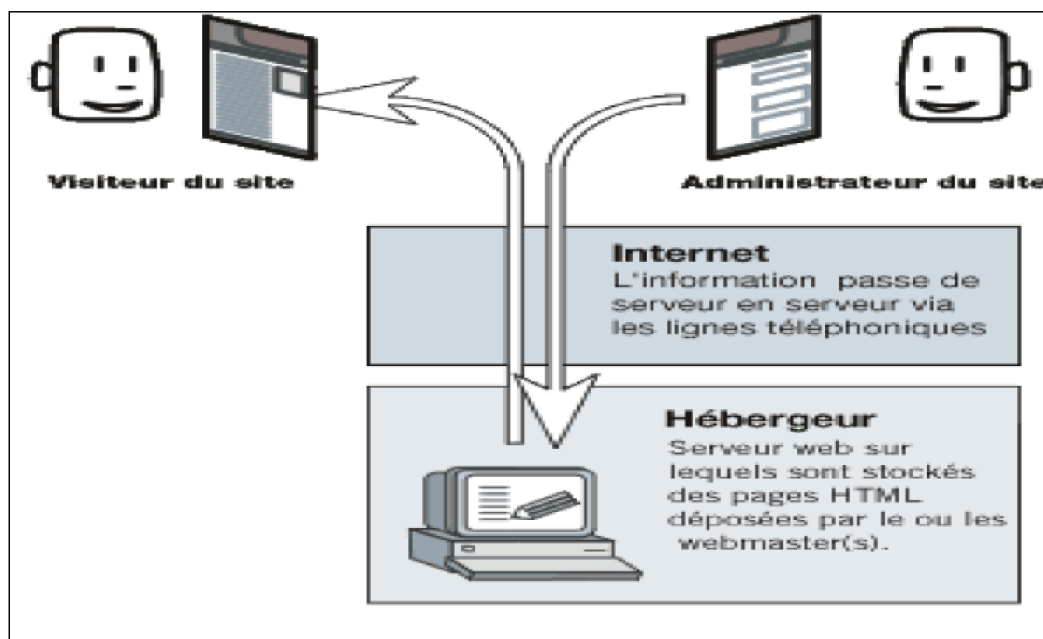


Figure I.2.: Les pages web statiques.

## b) Les pages Web dynamiques :

Par opposition au document statique, les documents dynamiques sont créés du côté serveur à la suite d'une requête client. Ainsi de nombreuses solutions mettant en œuvre un langage de script sur le serveur ont été mises au point :

- ✓ Le langage ASP (Active Server Page) de Microsoft a permis de simplifier l'écriture de tels scripts en manipulant des objets en VB script.
- ✓ Le langage PHP (Personale Home Page) emploie son propre langage et permet de nombreuses fonctionnalités (équivalentes à celles de la technologie ASP).
- ✓ Le langage JSP (Java Server Pages) est les plus récents parmi ces technologies. Elle permet d'utiliser toute la puissance de Java pour créer des pages Web dynamiques.

La figure représenté un site web dynamique

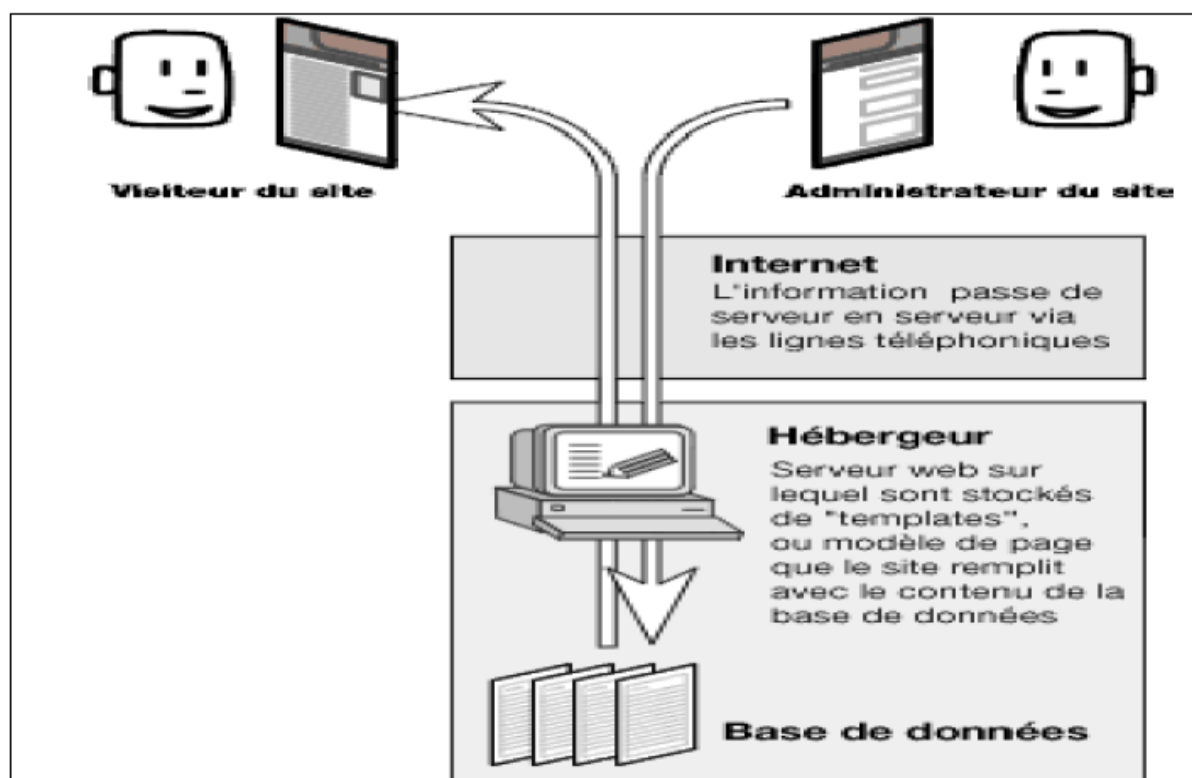


Figure I.3.: site web dynamique

### b.1 : Les avantages du site dynamique :

De nombreux scripts gratuits existent déjà et permettent de réaliser tous les sites qu'on souhaite. Ainsi en téléchargeant le script (ou CMS Content Management System), il sera très simple de créer un forum, un blog ou tout autre site.

La mise à jour est très simple : une fois le script dynamique en place, on met à jour le site en ligne dans la partie « administration » du site. On peut donc mettre à jour le site de

n'importe quel ordinateur et même depuis certains téléphones mobiles (avec accès Internet naturellement)

Avec un site dynamique il est possible de réaliser une grande interaction avec les visiteurs : les visiteurs peuvent donc rester beaucoup plus longtemps sur vos pages si les fonctionnalités sont intéressantes.

#### 4. Classification des sites Web:

On peut classer les sites Web selon leur but visé en plusieurs catégories :

- **Les sites vitrines:**

Ce sont des sites qui ont pour but de mettre en avant l'image de marque de l'entreprise, en présentant par exemple ses produits ou ses services.

- **Les sites catalogue:**

Sont des sites visant à présenter l'offre de l'entreprise.

- **Les sites d'information:**

Sont des sites fournissant une information particulière à un type d'internautes

- **Les sites marchands:**

Sont des sites vendant directement des produits aux internautes et permettant éventuellement de payer en ligne. Leur activité principale est le e-commerce.

- **Les sites institutionnels:**

Ce sont des sites présentant l'organisation et ses valeurs. Ce type de site décrit généralement l'activité de l'organisation, des chiffres clés et donne les informations nécessaires aux clients ou aux bénéficiaires. En l'occurrence, le site que j'ai développé dans ce travail rentre dans cette catégorie de sites.

- **Les sites personnels (blogs):**

Ce sont des sites réalisés par des particuliers à titre vde loisir, le plus souvent par passion pour un sujet ou une discipline.

- **Les sites communautaires :**

Sont des sites réunissant des internautes autour d'un intérêt commun.

- **Les sites intranet :**

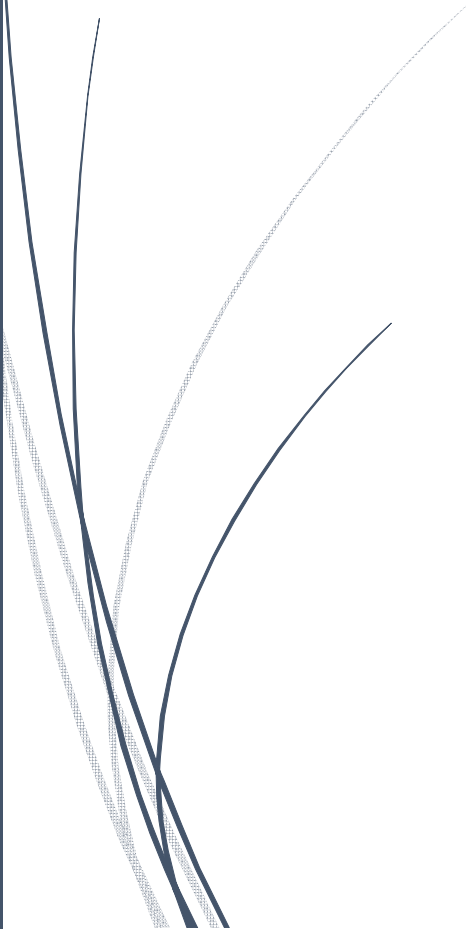
Sont des sites accessibles de l'intérieur d'une entreprise ou d'une direction, ayant pour objet la mise à disposition et le partage d'informations professionnelles entre les membres de cette organisation.

**Conclusion :**

Après avoir détaillé quelques notions concernant l'internet et le web qui constituent les notions de base dont nous avons besoin afin de construire notre site. Nous allons présenter dans le chapitre suivant le langage de modélisation que nous utiliserons dans la modélisation de notre système ainsi que la démarche adoptée.



***Chapitre 02:  
Présentation le langage  
de modélisation UML***



## **Introduction :**

Le développement des applications et des logiciels est une opération très difficile et complexe. Selon leurs tailles les applications peuvent être développées par une seule personne ou un ensemble d'équipes coordonnées. Le développement de grands logiciels.

Par des grandes équipes posent d'importants problèmes de conception et de coordination.

Pour maîtriser la complexité et résoudre ces problèmes plusieurs langages ou méthodes de

Spécification et outils de modélisation ont été créés (telle MERISE) et autre comme Booch, OMT, UML.

Dans notre projet on va utiliser le langage de modélisation UML. Donc le but de ce chapitre est de donner une description détaillée des diagrammes fondamentaux d'UML (Unified, Modeling, Language), qui s'est imposé comme une norme standard de modélisation.

## 1. La modélisation :

### 1.1 C'est quoi un modèle?

Un modèle est une représentation abstraite et simplifiée (qui exclut certains détails), d'une entité (Phénomène, processus, système, etc.) Du monde réel en vue de le décrire, de l'expliquer ou de le prévoir.

Un modèle permet de réduire la complexité d'un phénomène en éliminant les détails qui n'influencent pas Son comportement de manière significative.

Il reflète ce que le concepteur croit important pour la compréhension du phénomène modélisé.

### 1.2 Pourquoi modéliser ?

Modéliser un système avant sa réalisation permet de:

☐☐ **Faciliter la compréhension du système étudié** : Un modeler étui la complexité du système étudié.

☐☐ **Faciliter la communication entre les membres de l'équipe**: Un modèle est un langage Commun, précis, qui est connu par tous les membres de l'équipe et il est donc, à ce titre, un vecteur Privilégié pour communiquer .Cette communication est essentielle pour aboutir à une compréhension Communaux différentes parties prenantes (notamment entre la maîtrise d'ouvrage et la maîtrise d'œuvre Informatique.

☐☐ **Mieux répartir les tâches entre les membres de l'équipe.**

☐☐ **Réduire les coûts et les délais** : Par exemple, les plateformes de modélisation savent maintenant Exploiter les modèles pour faire de la génération de code (au moins au niveau du squelette).

• **Faciliter la maintenance** : une fois mise en production, l'application va devoir être maintenue, Probablement par une autre équipe qui n'est pas nécessairement celle ayant créée l'application.

### 1.3 Définition d'UML :

UML (sigle désignant Unifie Mödling Langage ou « langage de modélisation unifié ») est un Langage de modélisation graphique à base de pictogramme. Il est apparu dans le monde du génie logiciel.

C'est une méthode de modélisation orientée objet développée en réponse à l'appel à propositions lancé Par l'OMG (Object Management Group) dans le but de définir la notation standard pour la modélisation Des applications construites à l'aide d'objets.

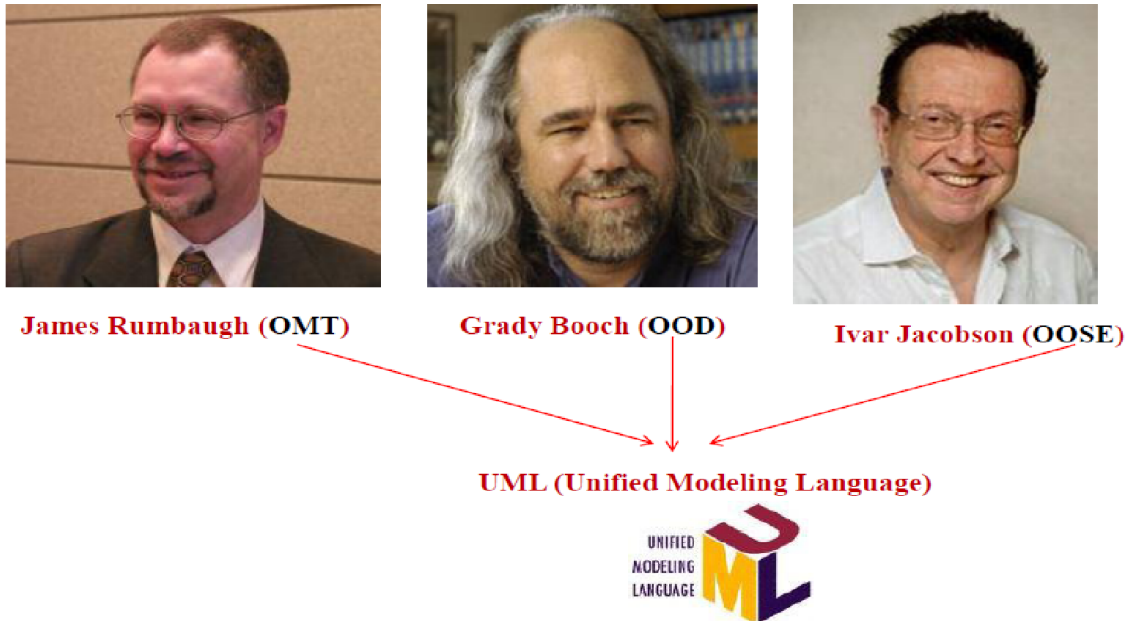


Figure 2.1 : Image développeur UML

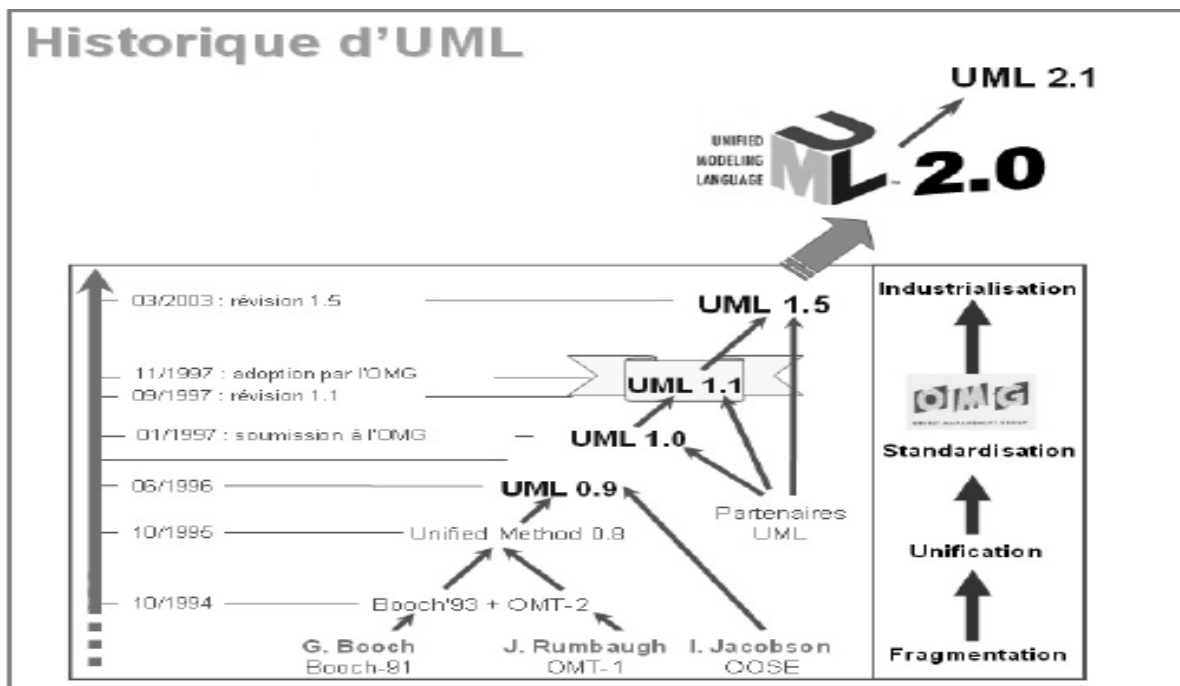


Figure 2.2 : Historique UML [1]

#### 1.4. Caractéristiques d'UML :

UML cadre l'analyse objet en offrant différentes vues (perspectives) complémentaires d'un système, il offre aussi Plusieurs niveaux d'abstraction qui permettent de mieux contrôler la complexité dans l'expression des solutions Objets.

Les principales caractéristiques d'UML sont les suivantes :

- Sa notation graphique permet d'exprimer visuellement une UML solution objet.
- L'aspect formel de sa notation limite les ambiguïtés et les Incompréhensions.
- Son aspect visuel facilite la comparaison et l'évaluation de Solutions.
- Son indépendance (par rapport aux langages d'implémentation, domaine d'application, processus...) en font un Langage Universel.

## 2. Présentation des vues et diagrammes d'UML :

UML dans sa version 2 propose treize diagrammes (9 en UML 1.3) qui peuvent être utilisés dans la description d'un Système.

Ces diagrammes sont regroupés dans deux grands ensembles

### 2.1. Diagrammes structurels ou diagrammes statiques :

Ces diagrammes permettent de visualiser, spécifier, construire et documenter l'aspect statique ou structurel du Système informatisé :

□□ **Diagramme de classes (Class diagram)**: Le but d'un diagramme de classes est d'exprimer de manière générale la Structure statique d'un système, en termes de classes et de relations entre ces classes. Une classe a des attributs, des Opérations et des relations avec d'autres classes.

□□ **Diagramme d'objets (Object diagram)**: Le diagramme d'objet permet d'éclairer un diagramme de classe en l'illustrant par des exemples. Il montre des objets et des liens entre ces objets (les objets sont des instances de classes Dans un état particulier).

□□ **Diagramme de composants (Component diagram)** : il montre les composants du système d'un point de vue Physique, tels qu'ils sont mis en œuvre (fichiers, bibliothèques, bases de données...). Il montre la mise en œuvre Physique des modèles de la vue logique avec l'environnement de développement.

□□ **Diagramme de déploiement (Deployment diagram)** : Ce type de diagramme UML montre la disposition physique Des matériels qui composent le système (ordinateurs, périphériques, réseaux...) et la répartition des composants sur Ces matériels. Les ressources matérielles sont représentées sous forme de nœuds, connectés par un support de Communication.

□□ **Diagramme des paquetages (Package Diagramme)** : un paquetage est un conteneur logique permettant de regrouper Et d'organiser les éléments dans le modèle UML, il sert à représenter les dépendances entre paquetages.

□ **Diagramme de structure composite (Composite Structure Diagram)** : Le diagramme de structure composite Permet de décrire sous forme de boîte blanche les relations entre les composants d'une seule classe.

## 2.2 Diagrammes comportementaux ou diagrammes dynamiques :

Les diagrammes comportementaux modélisent les aspects dynamiques du système. Ces aspects incluent les interactions entre le système et ses différents acteurs, ainsi que la façon dont les différents objets contenus dans le Système communiquent entre eux.

□□ **Diagramme des cas d'utilisation (Use Case Diagram)** : Les cas d'utilisation sont une technique de description du système étudié selon le point de vue de l'utilisateur. Ils décrivent sous la forme d'actions et de réactions le comportement d'un système. Donc, le diagramme des cas d'utilisation, permet d'identifier les possibilités d'interaction entre le système et les acteurs. Il permet de clarifier, filtrer et organiser les besoins.

□□ **Diagramme d'activité (ActivityDiagram)** : Un diagramme d'activité est une variante des diagrammes d'états-transitions. Il permet de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation. Dans un diagramme d'activité les états correspondent à l'exécution d'actions ou d'activités et les Transitions sont automatiques.

□□ **Diagramme états-transitions (State Machine Diagram)** : permet de décrire sous forme de machine à états finis Le comportement du système ou de ses composants. Il est composé d'un ensemble d'états, reliés par des arcs orientés qui décrivent les transitions.

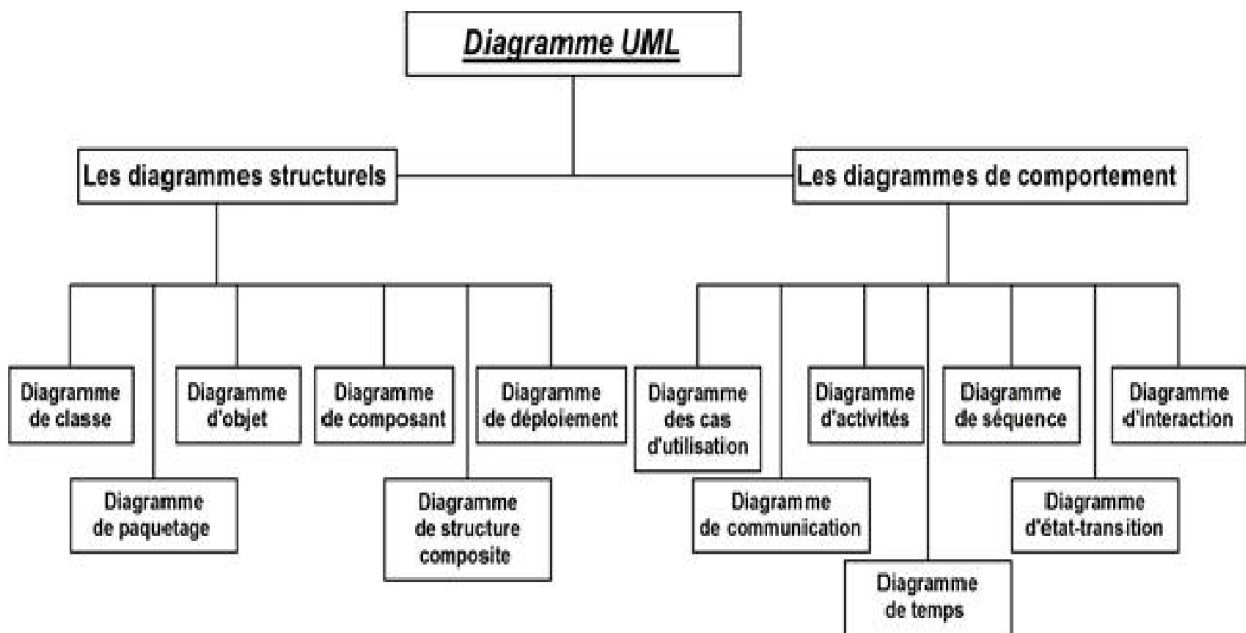
□□ **Diagramme de séquence (SequenceDiagram)** : Il représente séquentiellement le déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs. Le diagramme de séquence peut servir à illustrer un cas d'utilisation.

□□ **Diagramme de communication (Communication Diagram)** : C'est une représentation simplifiée d'un diagramme de séquence, en se concentrant sur les échanges de messages entre les objets.

□ □ **Diagramme global d'interaction (Interaction Overview Diagram)** : permet de décrire les enchaînements possibles entre les scénarios préalablement identifiés sous forme de diagrammes de séquences (variante du diagramme d'activité).

□ □ **Diagramme de temps (Timing Diagram)** : Le diagramme de temps permet de décrire les variations d'une donnée au cours du temps.

Nous décrivons dans ce qui suit les principaux diagrammes d'UML :



**Figure 2.3:** Schéma de la hiérarchie des diagrammes d'UML 2.0

### 2.3. Diagramme de cas d'utilisation :

Le diagramme de cas d'utilisation représente la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. C'est le premier diagramme du modèle UML, celui où s'assure la relation entre L'utilisateur et les objets que le système met en œuvre.

Le diagramme de cas d'utilisation permet :

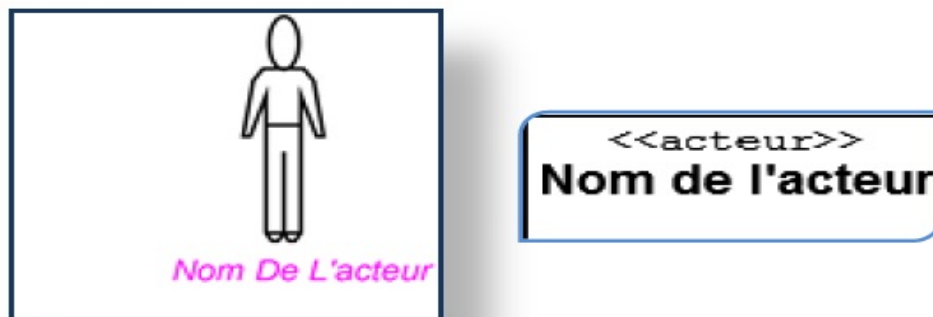
- □ D'exprimer simplement les besoins des utilisateurs
- □ D'analyser les besoins des utilisateurs
- De déterminer les interfaces du système

- □ De connaître le comportement du système sans spécifier comment ce comportement est réalisé,
- □ De définir les limites précises du système,
- De comprendre l'attente des utilisateurs et des experts du domaine. Les éléments d'un diagramme de cas

D'utilisations sont les suivants :

- □ Acteur : Un acteur est l'archétype de l'utilisateur (personne, externe, ...) qui interagit avec le système. Il se

Représente par un petit bonhomme avec son nom ou par la forme d'un classeur.



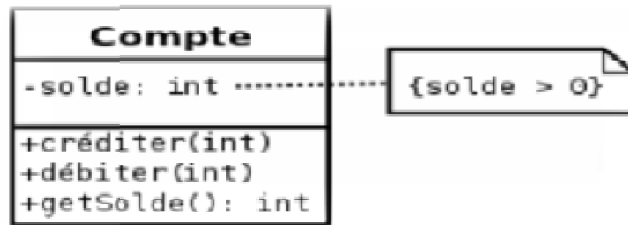
**Figure2.4:** Représentation d'un acteur

\* **Cas d'utilisations:** Les cas d'utilisations modélisent le service rendu par le Système sans en imposer le mode de réalisation. Ils permettent de décrire l'interaction entre l'acteur et le système. Les cas d'utilisations sont représentés par une ellipse sous-titrée par le nom du cas d'utilisation (éventuellement le nom est placé dans l'ellipse). Dans le cas où l'on désire présenter les attributs ou les opérations du cas d'utilisation, il est préférable de le représenter sous la forme d'un classeur stéréotypé



**Figure2.5:** Représentations de cas d'utilisation.

\* **Note:** Une note permet l'ajout d'une information textuelle à un diagramme. Cette information peut être un commentaire, un corps de méthode ou une contrainte. Les notes sont représentées par un rectangle avec le coin supérieur droit replié sur lui-même. On peut relier une note à un élément en utilisant une ligne pointillée.



**Figure2.6 :** Représentations d'une note

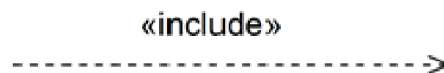
• **Relations**

➤ **Relation d'association :** Une relation d'association est un lien de communication entre un acteur et un cas d'utilisation. Elle est représentée par un trait continu.



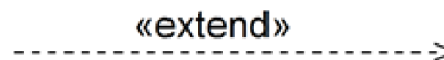
**Figure2.7 :** Représentation de relation d'association.

➤ **Relation d'association :** La relation d'inclusion spécifie qu'un cas d'utilisation est nécessairement une partie d'un autre cas d'utilisation. Elle est représentée par Une flèche discontinue stéréotypée <<inclusion>>



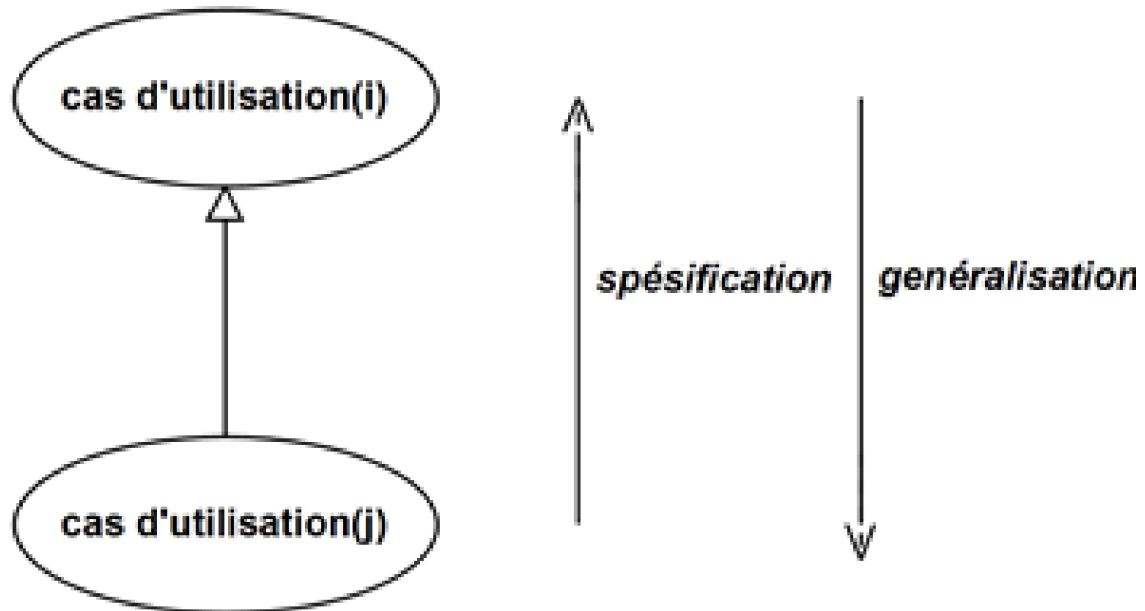
**Figure2.8 :** Représentations de relation d'inclusion.

➤ **Relation d'extension:** La relation d'extension spécifie qu'un cas d'utilisation est éventuellement une partie d'un autre cas d'utilisation. Elle est représentée par Une flèche discontinue stéréotypée <<extension>>.



**Figure2.9 :** Représentations de relation d'extension.

➤ **Relation de généralisation** : La relation de généralisation/spécialisation est la transposition aux cas d'utilisation de la notion d'héritage dans le paradigme objet. Elle est représentée par Une flèche dont la pointe (un triangle fermé) est dirigée vers l'élément le plus général



.Figure 2.10: Relation de généralisation.

## 2.4. Diagramme de classe/

Le diagramme de classe est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Le diagramme de classe est un diagramme entité association identifie la structure de la classe et les interfaces d'un système, y compris les propriétés et les méthodes de chaque classe ainsi que les différentes relations entre celles-ci. Les éléments de diagramme de classe sont les suivants :

- **Classe**: Une classe est une **abstraction** de choses du monde réel possédant des **caractéristiques** (Attributs) et des **Comportements communs** (méthodes) qui sont liées ensemble par un champ sémantique. Une classe est représentée par un rectangle séparé en trois parties : la première partie contient le nom de la classe, la seconde contient les attributs de la classe et la dernière contient les méthodes de la classe

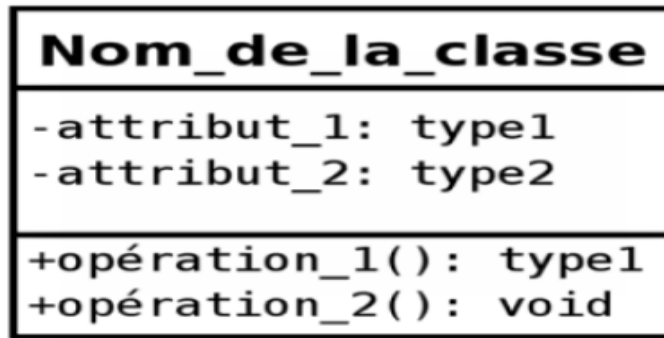


Figure2.11: Représentations d’une classe

• Relations entre classes :

➤ **Relation d’association :** Une association est une relation entre deux classes (association binaire) ou plus (Association aire), qui décrit les connexions structurelles entre leurs instances. Une association indique donc qu’il Peut y avoir des liens entre des instances des classes associées.



Figure2.12 : Relation d’association.

➤ **Relation d’agrégation :** Une agrégation est une **relation tout-partie** entre un agrégat (le tout) et un composant ou agrégée (la partie). L’agrégation est représentée par un losange blanc du côté de l’agrégat



Figure 2.13: Représentation relation d’agrégation.

➤ **Relation de composition:** La composition, également appelée agrégation composite, décrit une contenance structurelle entre instances. Ainsi, la destruction de l’objet composite implique la destruction de ses Composants.

L’agrégation est représentée par un losange noir du côté de l’agrégat composite.



Figure 2.14: Représentation relation de composition.

➤ **Relation de généralisation et héritage:** La **généralisation** est la relation entre une classe et deux autres classes ou plus partageant un sous-ensemble commun d'attributs et/ou d'opérations. La classe qui est affinée s'appelle **Superclasse**, les classes affinées s'appellent **sous-classes**. L'opération qui consiste à créer une superclasse à partir de classes s'appelle la généralisation. Inversement la **spécialisation** consiste à créer des sous-classes À partir d'une classe. Elle est représentée par Une flèche dont la pointe (un triangle fermé) est dirigée vers la superclasse.

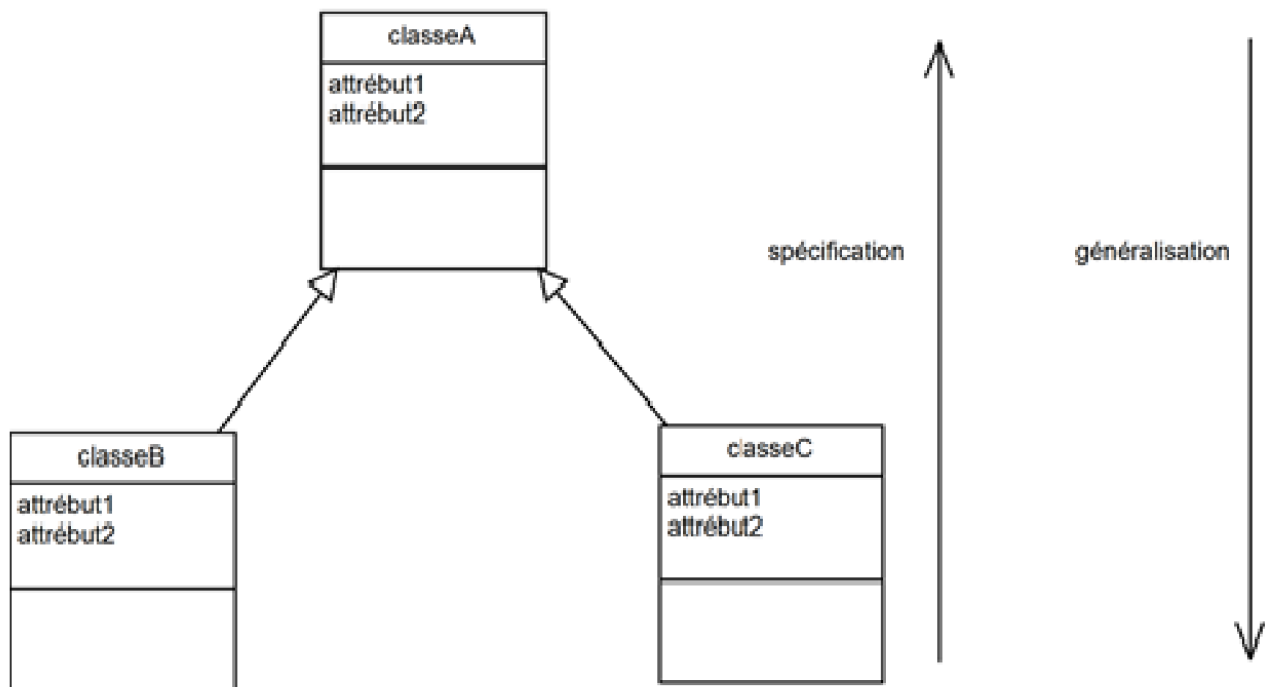


Figure2.15 : Représentation relation de généralisation

## 2.5. Diagramme d'activité :

Le diagramme d'activité est un diagramme états-transitions simplifié pour lequel les états se réduisent à de simples actions ou activités et dont les transitions se déclenchent automatiquement avec éventuellement des gardes.

Les diagrammes d'activité sont utilisés pour documenter le déroulement des opérations dans un système, du niveau commercial au niveau opérationnel (de haut en bas). Les éléments de base de diagramme d'activité :

- ✚ **Nœud initial** : Un nœud initial est un nœud de contrôle à partir duquel le flot débute lorsque l'activité enveloppante est invoquée. Une activité peut avoir plusieurs nœuds initiaux.
- ✚ **Nœud final** : Un nœud final est un nœud de contrôle possédant un ou plusieurs arcs entrants et aucun arc sortant.
- ✚ **Nœud de fin d'activité** : Lorsque l'un des arcs d'un nœud de fin d'activité est activé (i.e. lorsqu'un flot d'exécution atteint un nœud de fin d'activité), l'exécution de l'activité enveloppante s'achève et tout nœud ou flot actif au sein de l'activité enveloppante est abandonné.
- ✚ **Nœud de fin de flot** : Lorsqu'un flot d'exécution atteint un nœud de fin de flot, le flot en question est terminé, Mais cette fin de flot n'a aucune incidence sur les autres flots actifs de l'activité enveloppante.
- ✚ **Nœud de décision** : Un nœud de décision est un nœud de contrôle qui permet de faire un choix entre plusieurs flots sortants. Il possède un arc entrant et plusieurs arcs sortants.
- ✚ **Nœud de fusion** : Un nœud de fusion est un nœud de contrôle qui rassemble plusieurs flots alternatifs entrants En un seul flot sortant. Il n'est pas utilisé pour synchroniser des flots concurrents (c'est le rôle du nœud d'union) Mais pour accepter un flot parmi plusieurs.
- ✚ **Le nœud d'action** : Un nœud d'action est un nœud d'activité exécutable qui constitue l'unité fondamentale de Fonctionnalité exécutable dans une activité.
- ✚ **Le nœud d'objet** : Un nœud d'objet permet de définir un flot d'objet (i.e. un flot de données) dans un Diagramme d'activités. Ce nœud représente l'existence d'un objet généré par une action dans une activité et utilisé Par d'autres actions.
- ✚ **La transition** : Quand un état d'activité est accompli, le traitement passe à un autre état d'activité. Les transitions Sont utilisées pour marquer ce passage. Les transitions sont modélisées par des flèches.

## 2.6. Le diagramme de séquence/séquence système :

Le diagramme de séquence représente la succession chronologique des opérations réalisées par un acteur, à savoir :

Saisir une donnée, consulter une donnée, lancer un traitement...etc. Il indique les objets que l'acteur va manipuler, et Les opérations qui font passer d'un objet à l'autre. Un diagramme de séquence se représente globalement dans un Grand rectangle avec indication du nom du diagramme en haut à gauche. Les éléments d'un diagramme de séquences sont les suivants :

- **Messages** : Un message définit une communication particulière entre des lignes de vie. Ainsi, un message est une communication d'un objet vers un autre objet. La réception d'un message est considérée par l'objet récepteur comme un événement qu'il faut traiter (ou pas). Plusieurs types de messages existent, les plus communs sont :

- L'invocation d'une opération : **message synchrone** (appel d'une méthode de l'objet cible).
- L'envoi d'un signal : **message asynchrone** (Typiquement utilisé dans la gestion événementielle).
- La création ou la destruction d'une instance de classe au cours du cycle principal.

- **Fragment d'interaction** : Un fragment d'interaction dit combiner correspond à un ensemble d'interaction auquel on applique un opérateur. Un fragment combiné se représente globalement comme un diagramme de séquence avec indication dans le coin à gauche du nom de l'opérateur.

- **Fragment d'interaction avec opérateur alt.** : L'opérateur alt. correspond à une instruction de test avec une ou plusieurs alternatives possibles possédant plusieursopérandes séparés par des pointillés. Il est aussi permis d'utiliser les clauses de type sinon

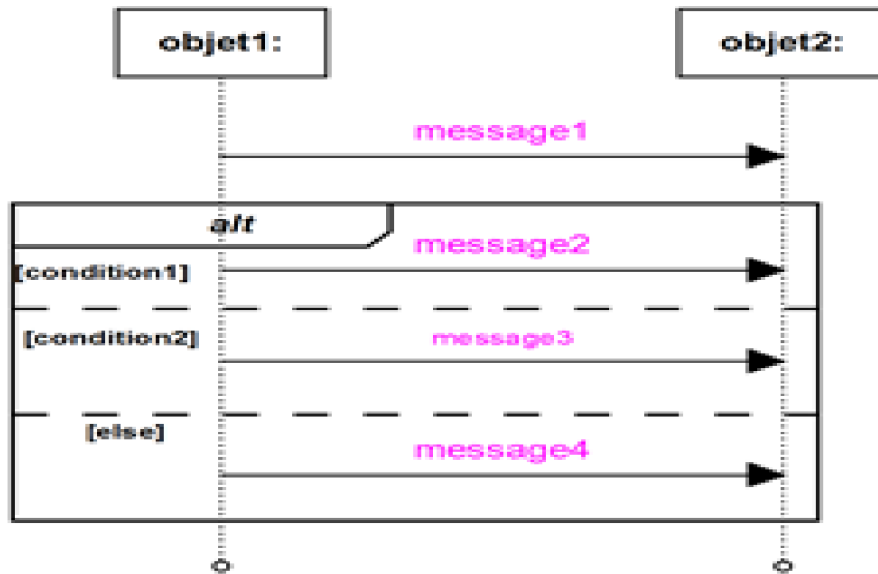


Figure2.16 : Fragment Alt

➤ **Fragment d'interaction avec operateur opt:** L'opérateur *option* (**opt**) comporte un opérande et une condition de garde associée. Le sous-fragment s'exécute si la condition de garde est vraie et ne s'exécute pas dans le cas contraire.

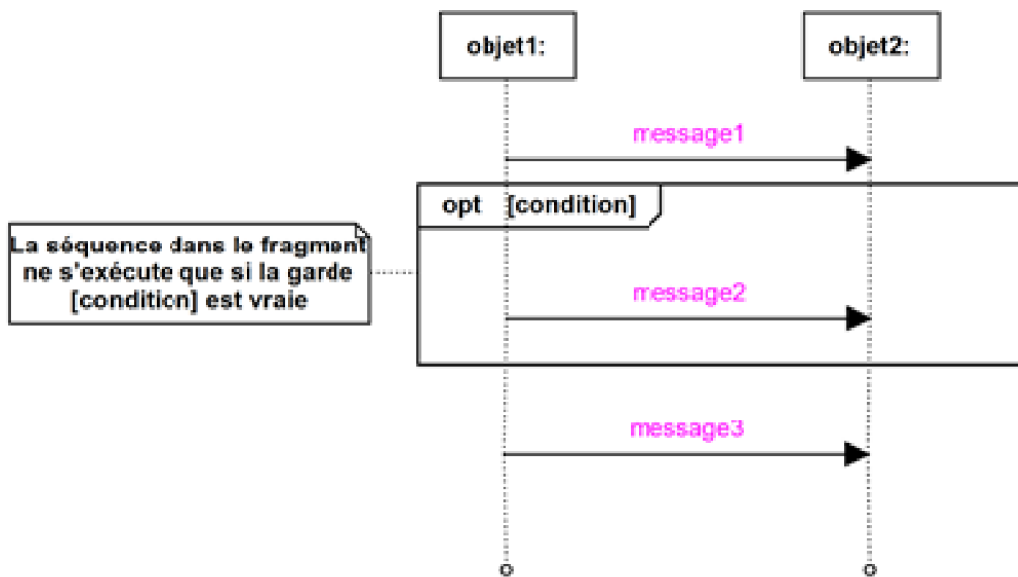


Figure2.17 : Fragment opt

➤ **Fragment d'interaction avec operateur loop:** L'opérateur loop correspond à une instruction de boucle Qui permet d'exécuter une Séquence d'interaction tant qu'une condition est satisfaite

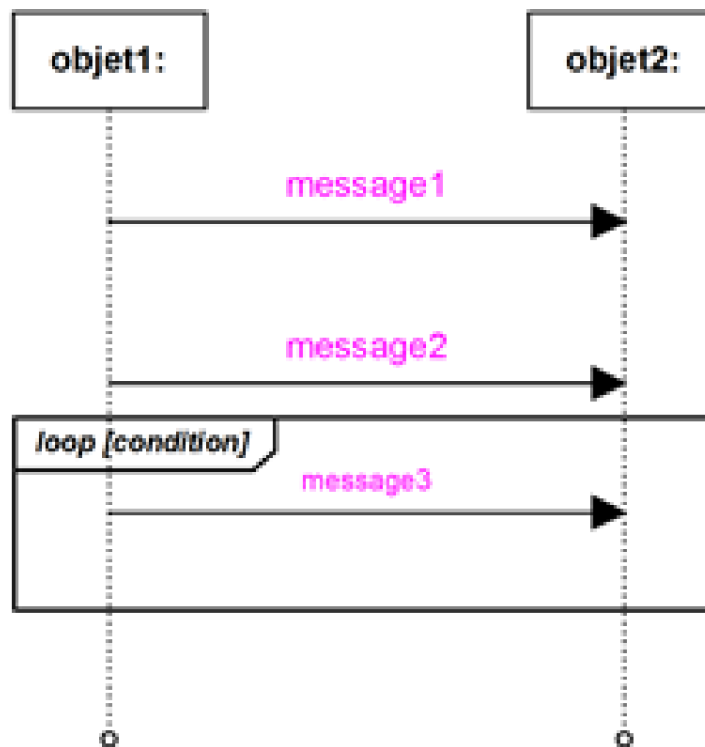


Figure2.18 : Schéma qui présente fragment loop

➤ **Fragment d'interaction avec operateur ref** : permet de faire appel à un autre diagramme de séquence.

## 2.7. Le diagramme de classe participante :

Le diagramme de classes participantes est un diagramme de classes UML qui décrit cas d'utilisation par cas, les trois principales classes d'analyse et leurs relations.

- **Les classes dialogues** : possèdent des attributs et des opérations. Les attributs représentent des champs de saisie Ou des résultats. Les opérations elles, représentent des actions de l'utilisation sur l'Interface Homme Machine (IHM)
- **Les classes contrôles** : contiennent des opérations. Ces opérations représentent la logique applicative de L'application, les règles métiers ou les comportements du système informatique.
- **Les classes entités** : possèdent en général des informations persistantes de l'application.
- **Le processus** : Un processus de développement définit une séquence d'étapes, en partie ordonnée, quiconcure à l'obtention d'un système logiciel ou à l'évolution d'un système existant ; pour produire des logiciels de qualité, qui répondent aux besoins des utilisateurs dans des temps etdes coûts prévisibles.

Processus s'appuyant sur UML :

### 3. Processus Unifié (UP) :

#### 3.1 Définition :

Un processus unifié est un processus de développement logiciel construit sur UML ; il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques. La gestion d'un tel processus est organisée suivant les 4 phases suivantes: Pré-étude (*inception*), élaboration, construction et transition.

Ces activités de développement sont définies par 5 disciplines fondamentales qui décrivent la capture des besoins, l'analyse et la conception, l'implémentation, le test et le déploiement. Le processus unifié doit donc être compris comme une trame commune des meilleures pratiques de développement, et non comme l'ultime tentative d'élaborer un processus universel. La définition d'un processus UP est donc constituée de plusieurs disciplines d'activité de production et de contrôle de cette production. Tout processus UP répond aux principes ci-après. [3]

#### 3.2 Principe :

\* **Il est itératif et incrémental** : La définition d'itérations de réalisation est en effet la meilleure pratique de gestion des risques d'ordre à la fois technique et fonctionnel. On peut estimer qu'un projet qui ne produit rien d'exécutable dans les 9 mois court un risque majeur d'échec. Chaque itération garantit que les équipes sont capables d'intégrer l'environnement technique pour développer un produit final et fournit aux utilisateurs un résultat tangible de leurs spécifications. Le suivi des itérations constitue par ailleurs un excellent contrôle des coûts et des délais.

\* **Il est piloté par les risques** : Dans ce cadre, les causes majeures d'échec d'un projet logiciel doivent être écartées en priorité. Nous identifions une première cause provenant de l'incapacité de l'architecture technique à répondre aux contraintes opérationnelles, et une seconde cause liée à l'inadéquation du développement aux besoins des utilisateurs.

\* **Il est orienté composant** : Tant au niveau modélisation que production, c'est une garantie de souplesse pour le modèle lui-même et le logiciel qu'il représente. Cette pratique constitue le support nécessaire à la réutilisation logicielle et offre des perspectives de gains non négligeables.

\* **Il est orienté utilisateur** : car la spécification et la conception sont construites à partir des modes d'utilisation attendus par les acteurs du système.

\* Il est construit autour de la création et de la maintenance d'un modèle : plutôt que de la production de montagnes de documents. Le volume d'informations de ce modèle nécessite une organisation stricte qui présente les différents points de vue du logiciel à différents degrés d'abstraction. L'obtention de métriques sur le modèle fournit par ailleurs des moyens objectifs d'estimation. [3]

### 3.3 Cycle de Vie d'un Processus Unifié :

La gestion d'un tel processus est organisée suivant les quatre phases suivantes : création (initialisation), élaboration, construction et transition.

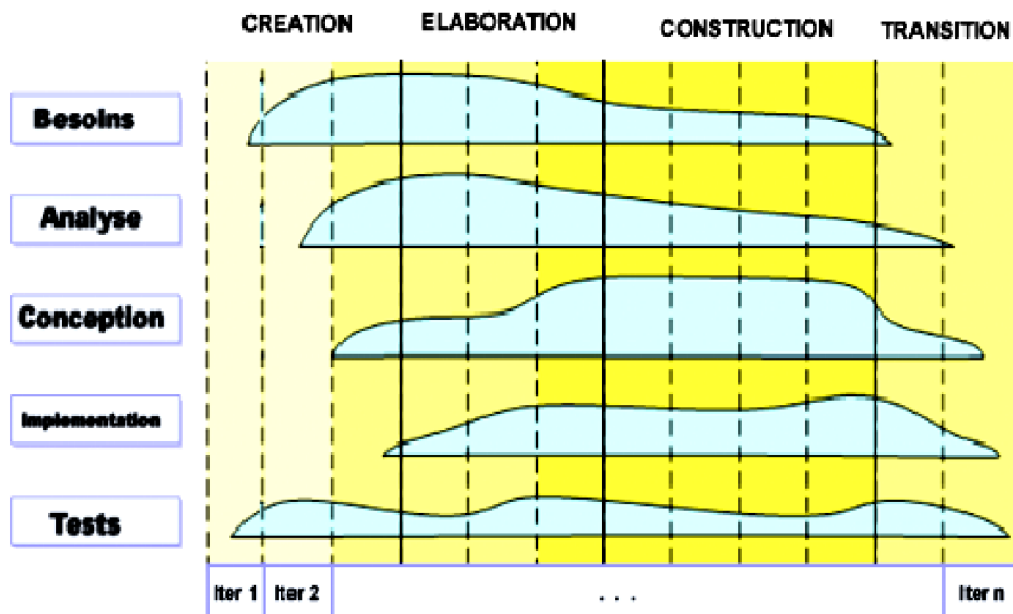


Figure 2.19 : Principe et phase de l'UP

- **Création:** C'est la première phase du processus unifié. Il s'agit de délimiter la portée du système, c'est-à-dire tracer ce qui doit figurer à l'intérieur du système et ce qui doit rester à l'extérieur, identifier les acteurs, lever les ambiguïtés sur les besoins et les exigences nécessaires dans cette phase. Il s'agit aussi d'établir une architecture candidate, c'est à dire que pour une première phase, on doit essayer de construire une architecture capable de fonctionner. Dans cette phase, il faut identifier les risques critiques susceptibles de faire obstacles au bon déroulement du projet.
- **Élaboration:** C'est la deuxième phase du processus. Après avoir compris le système, dégagé les fonctionnalités initiales, précisé les risques critiques, le travail à accomplir maintenant consiste à stabiliser l'architecture du système. Il s'agit alors de raffiner le modèle initial de cas

d'utilisation, voire capturer de nouveaux besoins, analyser et concevoir la majorité des cas d'utilisation formulés, et si possible implémenter et tester les cas d'utilisation initiaux.

- **Construction** : Dans cette phase, il faut essayer de capturer tous les besoins restants car il n'est pratiquement plus possible de le faire dans la prochaine phase. Ensuite, continuer l'analyse, la conception et surtout l'implémentation de tous les cas d'utilisation. A la fin de cette phase, les développeurs doivent fournir une version exécutable du système.

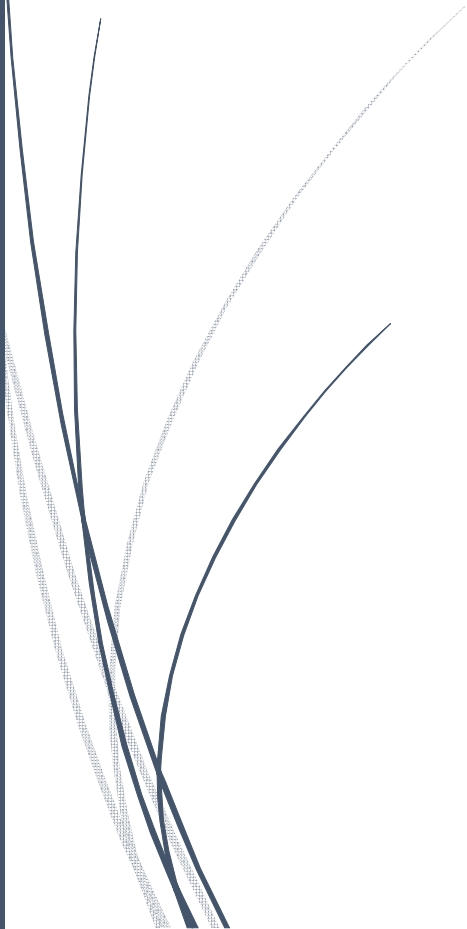
- **Transition** : C'est la phase qui finalise le produit. Il s'agit au cours de cette phase de vérifier si le système offre véritablement les services exigés par les utilisateurs, détecter les défaillances, combler les manques dans la documentation du logiciel et adapter le produit à l'environnement (mise en place et installation).

**Conclusion :**

Dans ce chapitre, nous avons passé en revue le langage de modélisation UML. D'abord, nous avons présenté les différents diagrammes définis par UML, puis nous avons présenté le processus que nous allons utiliser pour modéliser notre application. UML est un langage unifié qui est le résultat d'une opération d'unification d'un ensemble de conceptions pris à partir des méthodes orientées objets dans le but de modéliser d'une manière claire et précise la structure et le comportement d'un système indépendamment de toute méthode et de tout langage de programmation.



# **Chapitre 03 : Analyse des besoins et la conception**



## **Introduction :**

L'identification des besoins est une phase importante dans le processus de développement d'un logiciel. L'objectif de ces phases est de préciser les différentes tâches que le système devra faire. Ce système doit répondre aux besoins de l'utilisateur (dans notre cas l'auto-école).

Dans le cadre de ce chapitre et après une présentation du système à développer, on va mentionner et appliquer les différentes étapes de cette phase :

- Identification des acteurs de ce système ;
- Diagrammes de contexte dynamique
- Identification des cas d'utilisation ;
- Diagramme de cas d'utilisation ;
- Description textuelle du cas d'utilisation ;
- Diagramme de séquence système.

***Partie 01 :***  
***Identification des besoins***

## 1. Elaboration du cahier des charges :

### 1.1. Présentation du système:

Le système à développer est un site web dynamique pour une auto-école .Les objectifs principaux de notre système sont :

- Accéder au site et consulter toutes les informations de l'auto-école.
- Consulter les différentes annonces de l'auto-école.
- Passer des tests concernant le code et le créneau et consulter le résultat.
- Télécharger livres et cours concernant le code et la conduite.
- Consulter résultat de l'examen passé au niveau de l'auto-école.

### 1.2. La définition des grands choix techniques :

Les choix techniques que nous avons faits pour la réalisation de notre site web sont :

- Le processus de développement : UP (unifedprocess).
- Le langage de conception et de modélisation *UML (UnifiedModelingLanguage)*
- L'architecture de l'application : client/serveur.
- Le langage *PHP*, et le système de gestion de base de données relationnel *MySQL*.
- La plate-forme disponible:( dreamweaver....)

### 1.3. Les exigences fonctionnelles :

Ce sont les actions que doit effectuer le système en réponse à une demande, dans notre cas :

- L'internaute peut consulter en ligne les informations de l'auto-école.
- L'internaute peut consulter les différentes actualités de l'auto-école.
- L'internaute peut consulter en ligne les annonces d'auto-école.
- L'internaute peut s'inscrire comme membre.
- Le membre peut télécharger des livres.
- Le membre peut passer des tests en ligne.
- Le candidat peut voir les résultats de test réel.
- Le gérant peut faire des mises à jour sur les comptes (supprimer, ajouter et modifier les comptes).
- Le gérant peut ajouter et modifier les livres.
- Le gérant peut ajouter des résultats.

#### 1.4. Les besoins techniques :

- **Convivialité et simplicité d'utilisation de l'application** : L'application devrapouvoir être utilisée simplement. Elle doit avoir donc une interface graphique claire et bien organisée.
- **La sécurité** : Lors de l'authentification, chaque membre ou candidat doit être reconnue au système à l'aide d'un pseudo et un mot de passe
- **Le temps de réponse** : Le temps de réponse doit être acceptable.

#### 1.5. La description du contexte du système :

##### 1. Identification des acteurs :

*Visiteur* : personne inconnu par le site qui peut s'inscrire dans le site.

*Membre* : La personne qui est déjà inscrit au site.

*Candidat* : Présente la personne inscrit au site et inscrit réellement à l'auto-école, Il s'agit de l'acteur le plus important.

*Gérant* : rôle de l'employé qui prend en charge le fonctionnement et la maintenance du site web.

##### 2. Identification des cas d'utilisation :

L'identification des cas d'utilisation nous donne un aperçu sur les fonctionnalités futures de notre système. A part le visiteur, tous les acteurs doivent s'authentifier avant de consulter leur espace.

- *Visiteur* :
  - Consulter les informations.
  - S'inscrire.
  - Consulter les annonces.
- *Membre* :
  - Passer test.
  - Télécharger cour.
- *Candidat* :
  - Consulter le résultat d'examen.
- *Gérant* :
  - Mise à jour des cours.
  - Mise à jour d'information.

- Mise à jour des annonces.
- Gérer les comptes candidat.
- Afficher les résultats.

### 3- Diagramme de contexte dynamique :

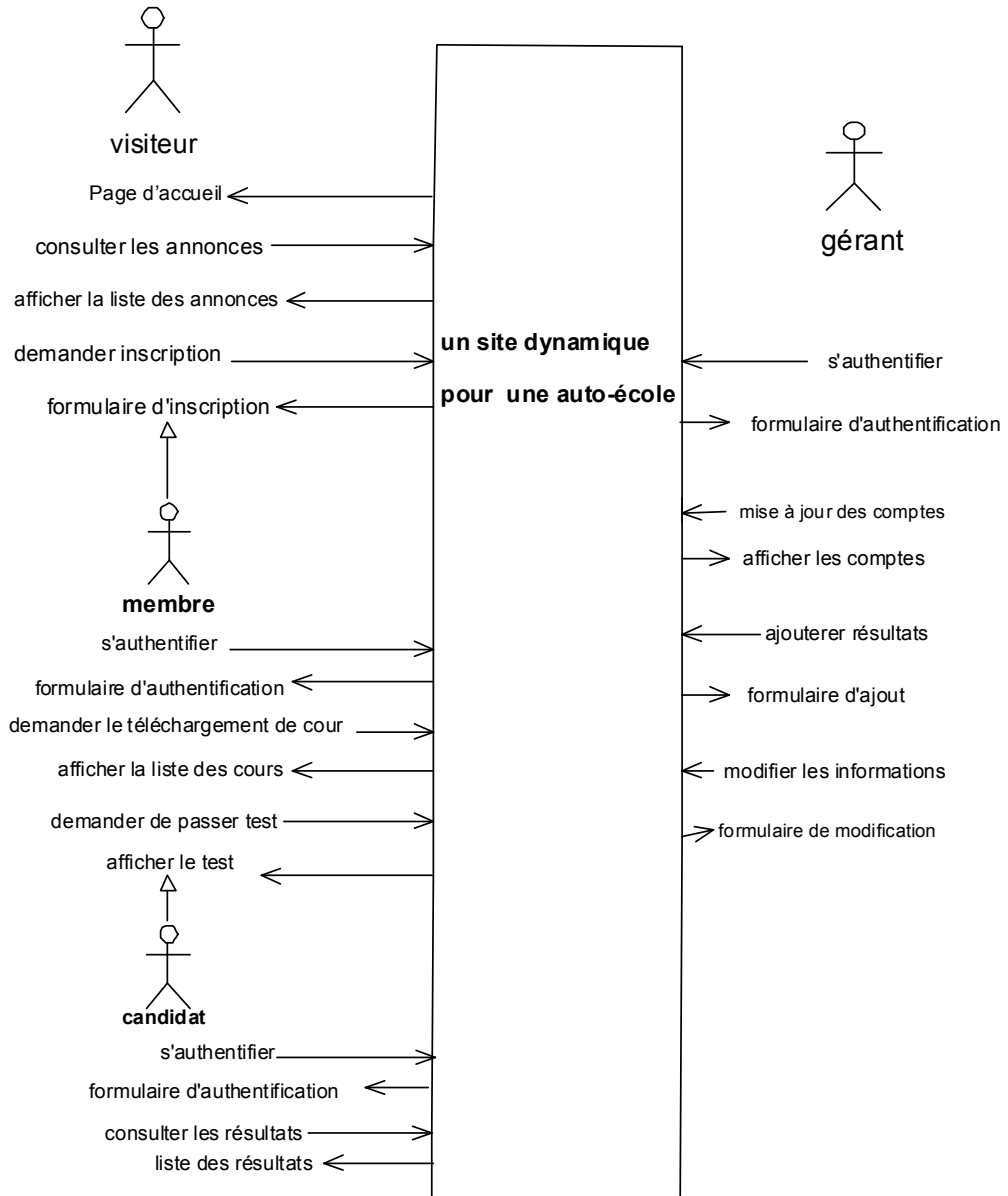


Figure 3.1: Diagramme de contexte dynamique

4. Diagramme de cas d'utilisation :

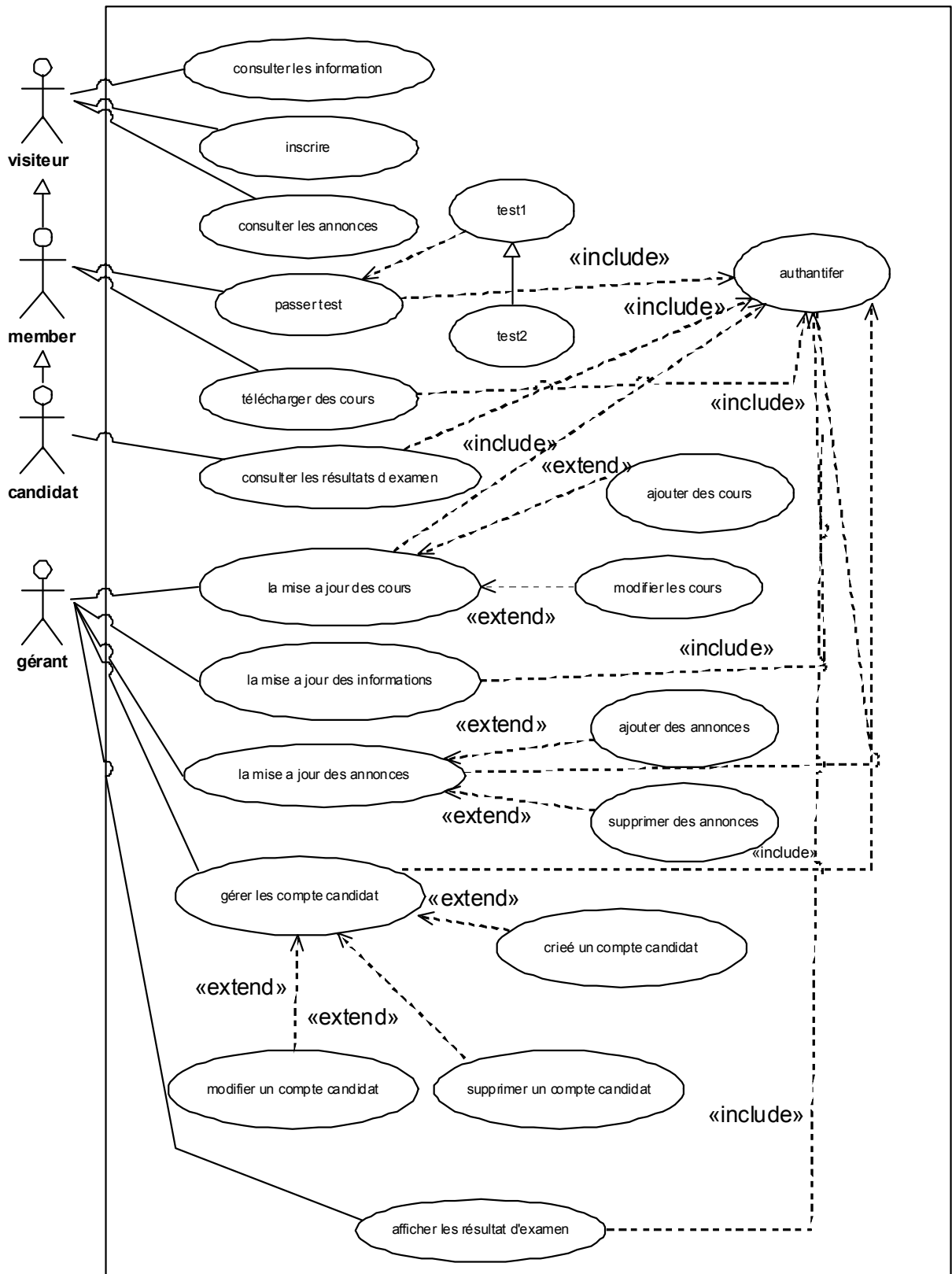


Figure 3.2 : Diagramme de cas d'utilisation

## 5. Les fiches descriptives des cas d'utilisations :

### 5.2.1- Authentifier

CU	Authentifier
acteur	Candidat / gérant / membre
Pré-condition	Chaque utilisateur possède un compte
Post- condition	Vérifié l'autorisation d'accéder au système  L'utilisateur est authentifié par le système
Scénario nominal	1- L'utilisateur demande de connexion au système 2- Le système demande à l'utilisateur d'entrer le mot de passe et son LOGIN 3- L'utilisateur saisie le mot de passe el LOGIN puis valide 4- Le système vérifier la validité des informations puis afficher son espace
Scénario alternatif	1- Les informations saisies sont incorrects ou incomplets

**Tableau 3.1:** Fiche descriptive du cas d'utilisation «Authentifier».

### 5.2.2-Consulter les informations :

CU	Consulter les informations
acteur	Visiteur
Pré-condition	Néant
Post- condition	Néant
Scénario nominal	1- Le visiteur demander la page d'information existe 2- Le système afficher la page

**Tableau 3.2 :** Fiche descriptive du cas d'utilisation «Consulter les informations».

**5.2.3 - s'inscrire :**

CU	S'inscrire
acteur	Visiteur
But	L'inscription au site d'auto-école
Pré-condition	Néant
Post- condition	Le visiteur est inscrire par le système
Scénario nominal	1- Le visiteur demande au système l'inscription 2- Le système envoi un formulaire d'inscription 3- Le visiteur remplit le formulaire 4- Le système vérifier l'inscription
Scénario alternatif	Le formulaire rempli est incorrecte (retour à point 3)

**Tableau 3.3 :** Fiche descriptive du cas d'utilisation «inscrire».

**5.2.4-Consulter les annonces :**

CU	Consulter les annonces
acteur	Visiteur
Pré-condition	Néant
Post- condition	Néant
Scénario nominal	1- Le membre demander la liste des annonces 2- Le système afficher les annonces 3- Le membre choisit l'annonce

**Tableau 3.4 :** Fiche descriptive du cas d'utilisation «Consulter les annonces».

**5.2 5- passé test :**

CU	Passer test
----	-------------

acteur	Membre/candidat
Pré-condition	Authentifié
Post- condition	Néant
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le Membre/candidat demander de passer un teste</li> <li>2- Le système afficher un formulaire</li> <li>3- Transmettre les réponses             <ol style="list-style-type: none"> <li>3.1-la note est &lt; à 80 %                 <ol style="list-style-type: none"> <li>3.1.1 le système affiché les résultats</li> </ol> </li> <li>3.2-la note est&gt;à 80%                 <ol style="list-style-type: none"> <li>3.2.1 le système affiché le formulaire de test 2</li> <li>3.2.3 le candidat transmettre les réponses</li> <li>3.2.4 le système affiché les résultats</li> </ol> </li> </ol> </li> </ol>

**Tableau 3.5 :** Fiche descriptive du cas d'utilisation «passer test»

### 5.2.6-Télécharger cours :

**Tableau 3.6 :** Fiche descriptive du cas d'utilisation «télécharger cours».

<b>CU</b>	<b>Télécharger cours</b>
Acteur	Membre/candidat
Pré-condition	Membre s'authentifier Cour existe déjà
Post- condition	Le système permet au membre de télécharger une cour
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le membre demande de télécharger une cour</li> <li>2- Le système affiche la liste des cours existants</li> <li>3- Le membre sélectionne une cour</li> <li>4- Le système lance le téléchargement et affiche le message « cour téléchargé »</li> </ol>

### 5.2.7. Consulter les résultats de d'examen :

CU	Consulter les résultats de test
----	---------------------------------

Acteur	Candidat
Pré-condition	Authentifier
Post- condition	Néant
Scénario nominal	1- Le candidat demander la liste de résultat 2- Le système afficher la liste

**Tableau 3.7 :** Fiche descriptive du cas d'utilisation «Consulter les résultats d'examen»

### 5.2.8. Mise à jour des cours :

#### 5.2. 8.1. Ajouter cours :

CU	Ajouter cours
acteur	Gérant
but	Ajouter un cours
Pré-condition	Le gérant s'est authentifié
Post- condition	Le système afficher un message d'enregistrement
Scénario nominal	5- Le gérant demander l'ajout d'un cours 6- Le système afficher le formulaire 7- Le gérant saisie les informations d'accès au cours 8- Le système affiché « le cours est ajouté avec succès»
Scénario alternatif	1- Les informations est incorrecte 2- cour est existée déjà

**Tableau 3.8 :** Fiche descriptive du cas d'utilisation «ajouter cour»

#### 5.2.8.2. Modifier un cours :

CU	Modifier cours
----	----------------

acteur	Gérant
Pré-condition	Cours déjà existé / authentifier
Post- condition	Le système afficher un message de modification
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le gérant demandé de modifier un cours</li> <li>2- Le système afficher la liste des cours</li> <li>3- Le gérant sélectionné un cours</li> <li>4- Le système affiche r un formulaire de modification</li> <li>5- Le gérant fait la modification</li> <li>6- Le système affiché « le cours est modifié avec succès »</li> </ol>

**Tableau 3.9 :** Fiche descriptive du cas d'utilisation «modifier cour»

### 5.2. 9. Modifié les informations :

CU	MAJ d'information
acteur	Gérant
but	Modifier l'information d'école
Pré-condition	Le gérant s'est authentifié
Post- condition	Le système afficher un message de modification
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le gérant demandé au système de modifier les informations</li> <li>2- Le système afficher la liste d'information</li> <li>3- Le gérant sélection l'information</li> <li>4- Le système envoie un formulaire</li> <li>5- Le gérant saisie les nouvelles informations</li> <li>6- Le système afficher l'information modifier</li> </ol>
Scénario alternatif	L'information saisie est incorrecte ou incomplet (retour a point 5)

**Tableau 3.10 :** Fiche descriptive du cas d'utilisation «modifier les informations»

### 5.2.10. Mise à jour des annonces :

#### 5.2.10 .1. Ajouter annonce :

CU	Ajouter une annonce
Acteur	Gérant
Pré-condition	
Post- condition	Le gérant s'est authentifier
Scénario nominal	1- Le gérant demander l'ajout d'une annonce 2- Le système afficher le formulaire 3- Le gérant saisie les informations d'annonce 4- Le système affiché « l'annonce est ajouté»
Scénario alternatif	1- l'annonce existe déjà

**Tableau 3.11** : Fiche descriptive du cas d'utilisation «ajouter annonce»

### 5.2.10.2. Supprimer une annonce :

CU	Supprimer l'annonce
Acteur	gérant
Pré-condition	authentifier
Post- condition	L'annonce sera supprimé de la base de donné
Scénario nominal	1- Le gérant demande au système de supprimer l'annonce 2- Le système afficher la liste des annonce 3- Le gérant choisi l'annonce et valider la suppression 4- Le système supprimer l'annonce est affiché le message de suppression

**Tableau 3.12** : Fiche descriptive du cas d'utilisation «supprimer annonce»

### 5.2.11. Gérer les comptes candidat :

#### 6.2.11.1. Ajouter un compte candidat :

CU	<b>Supprimer un compte candidat</b>
Acteur	Gérant
Précondition	Le gérant s'authentifier le compte existe
Post- condition	Le compte est supprimé
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le gérant demandé de supprimer un compte candidat</li> <li>2- Le système afficher la liste des comptes</li> <li>3- Le gérant sélectionné le candidat</li> </ol> <p>Le système supprimer le candidat est affiché un message de suppression succès</p>

**Tableau 3.13** : Fiche descriptive du cas d'utilisation «ajouter un compte candidat »

### 5.2.11.2. Supprimer un compte candidat :

**Tableau 3.13** : Fiche descriptive du cas d'utilisation «Supprimer un compte candidat »

CU	<b>Crée un compte candidat</b>
acteur	Gérant
Pré-condition	Ajouter un candidat
Post- condition	Le candidat est ajouté
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le gérant demander de crée un compte</li> <li>2- Le système affiche un formulaire</li> <li>3- Le gérant saisie les informations nécessaire</li> <li>4- Le système crée le compte</li> </ol>

### 5.2.11.3. Modifier un compte candidat :

CU	<b>modifier un compte candidat</b>
----	------------------------------------

Acteur	Gérant
Précondition	Le gérant s'authentifier le compte existe
Post- condition	Le compte est modifié
Scénario nominal	<ol style="list-style-type: none"> <li>1- Le gérant demandé de modifier un compte candidat</li> <li>2- Le système afficher la liste des comptes</li> <li>3- Le gérant sélectionné le compte</li> <li>4- Le système affiche un formulaire</li> <li>5- Le gérant saisie les informations nécessaire</li> <li>6- Le système modifier le compte</li> </ol>

**Tableau 3.14 :** Fiche descriptive du cas d'utilisation «modifier un compte candidat »

**5.2.13-ajouter résultat :**

CU	Ajouter résultat
acteur	Gérant
Pré-condition	Le gérant s'est authentifié
Post- condition	Le système afficher un message d'enregistrement
Scénario nominal	<ol style="list-style-type: none"> <li>9- Le gérant demander l'ajout d'un résultat</li> <li>10- Le système afficher le formulaire</li> <li>11- Le gérant saisie les résultats</li> <li>12- Le système ajouter le résultat</li> </ol>

**Tableau 3.15 :** Fiche descriptive du cas d'utilisation «ajouter résultat »

**6. Diagrammes de séquence:**

**6.1. S'authentifier :**

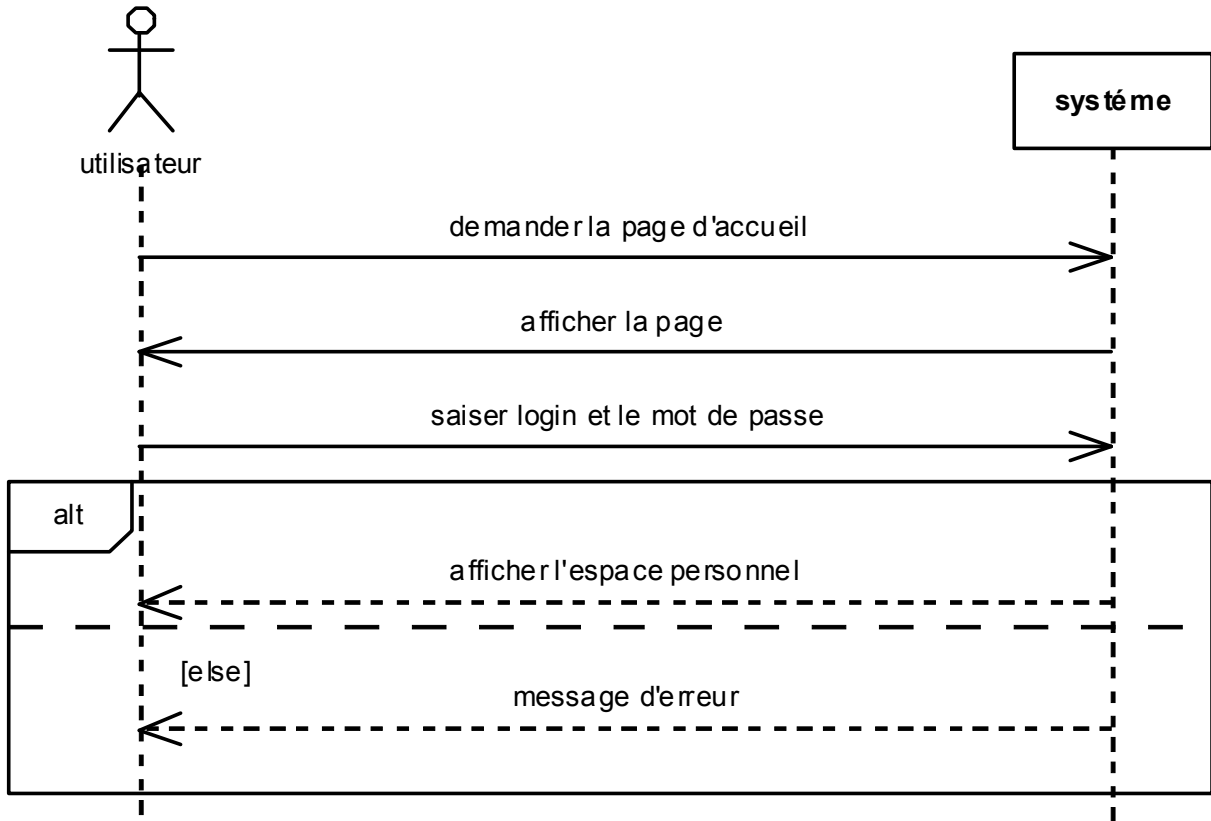


Figure 3.2 : Diagramme de séquence du cas d'utilisation « S’authentifier ».

**6.2. Consulter les informations :**

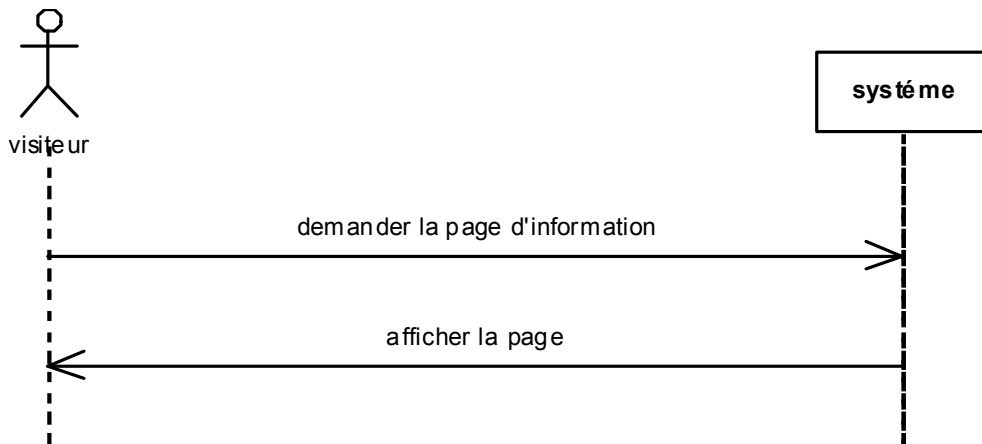


Figure 3.3 : Diagramme de séquence du cas d'utilisation «consulter les informations».

**6.3. Consulter les annonces :**

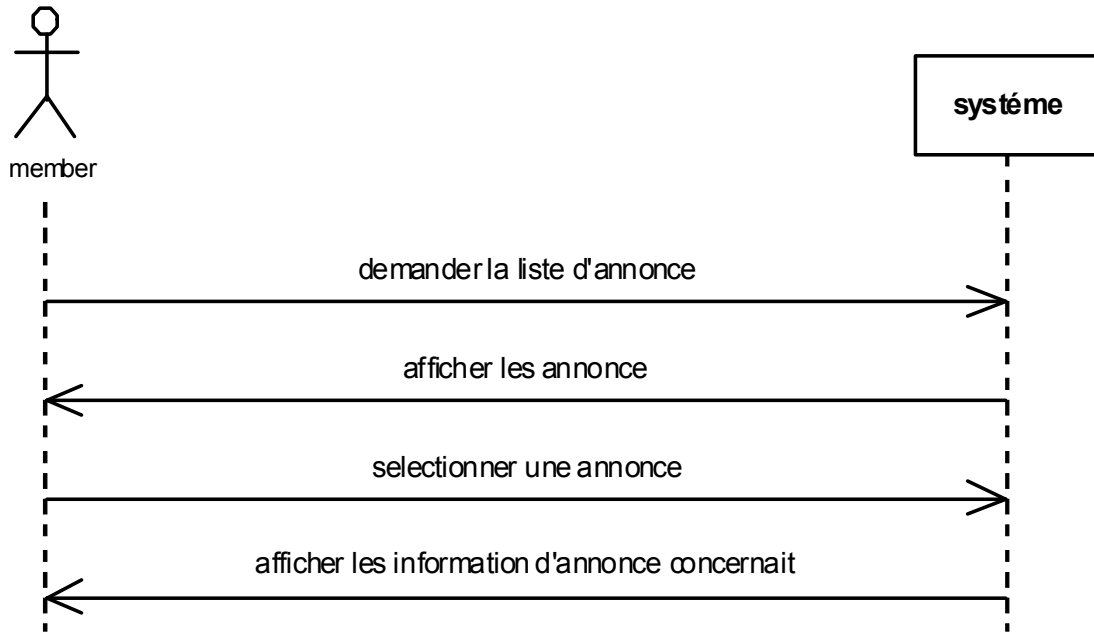


Figure 3.4 : Diagramme de séquence du cas d'utilisation «consulter les annonces».

6.4. S’inscrire :

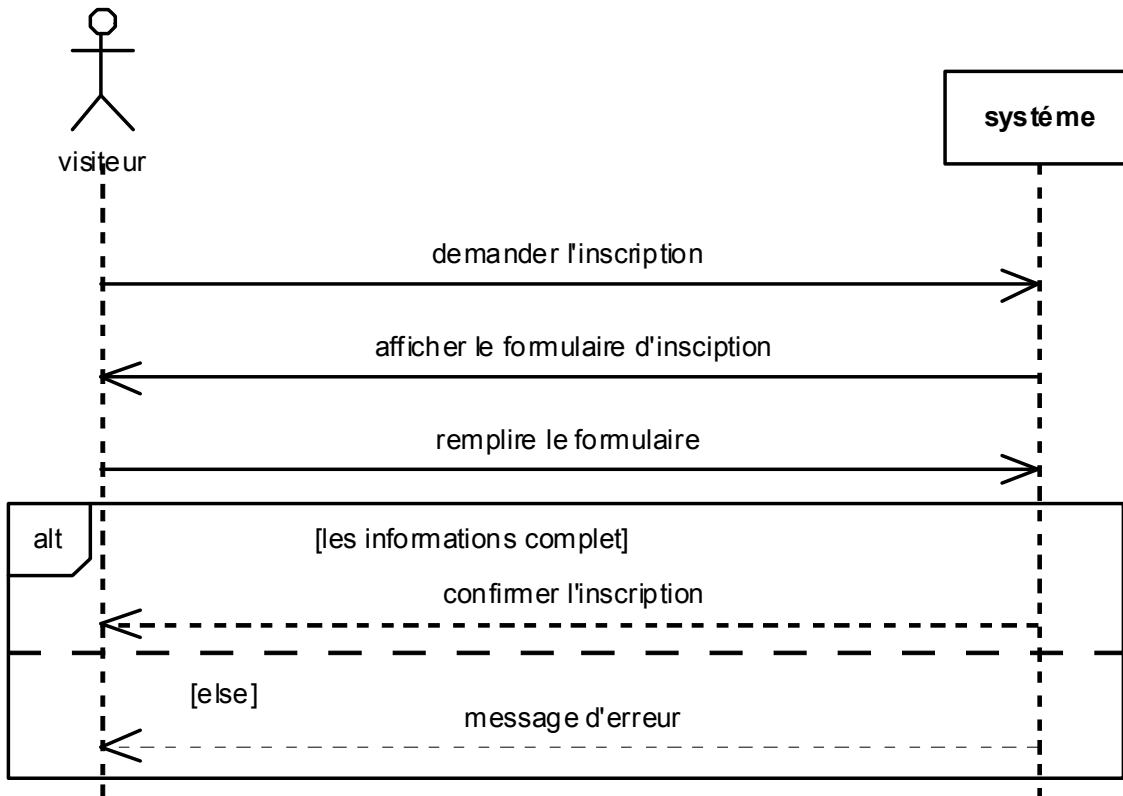


Figure 3.4 : Diagramme de séquence du cas d'utilisation « S’inscrire».

3.3.5. Passer test :

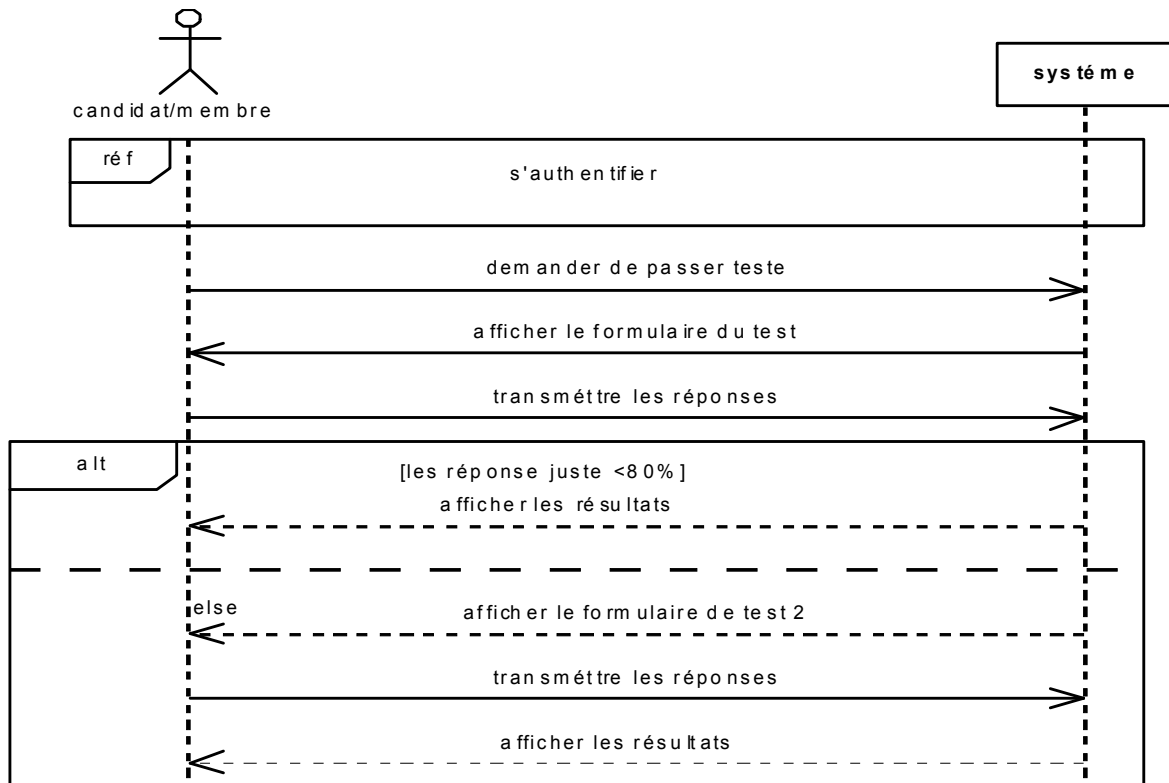


Figure 3.5 : Diagramme séquence système «passer test»

### 3.3.6. Télécharger cour :

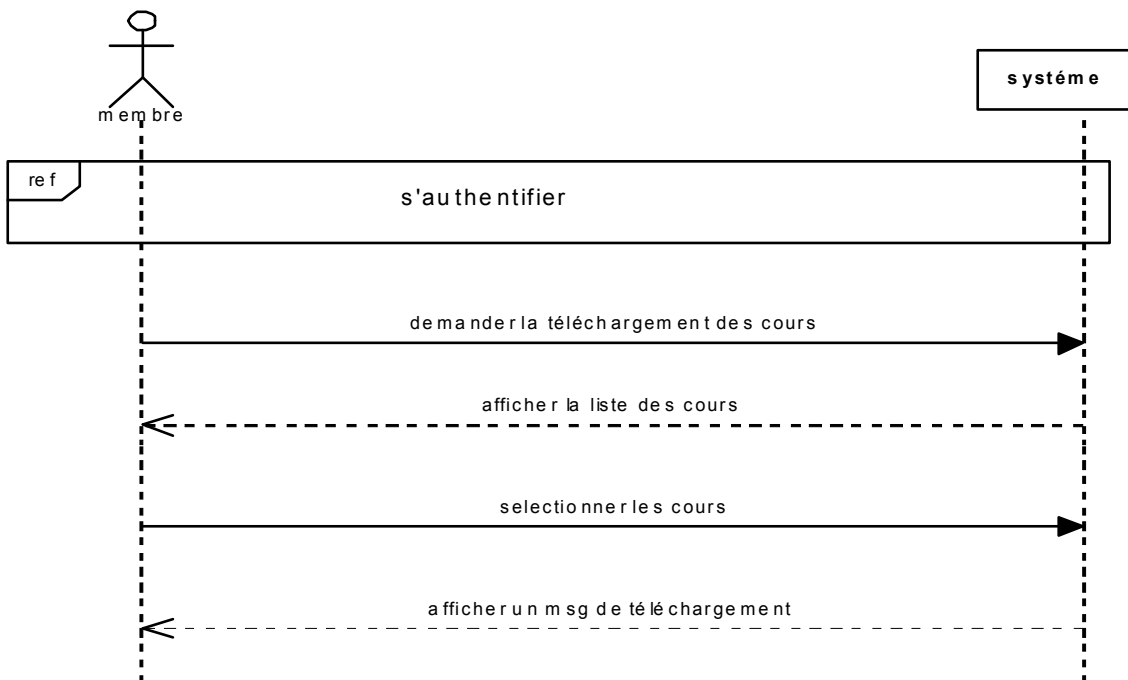


Figure 3.6 : Diagramme de séquence système «télécharger cour»

### 3.3.7. Consulter les résultats d'examen :

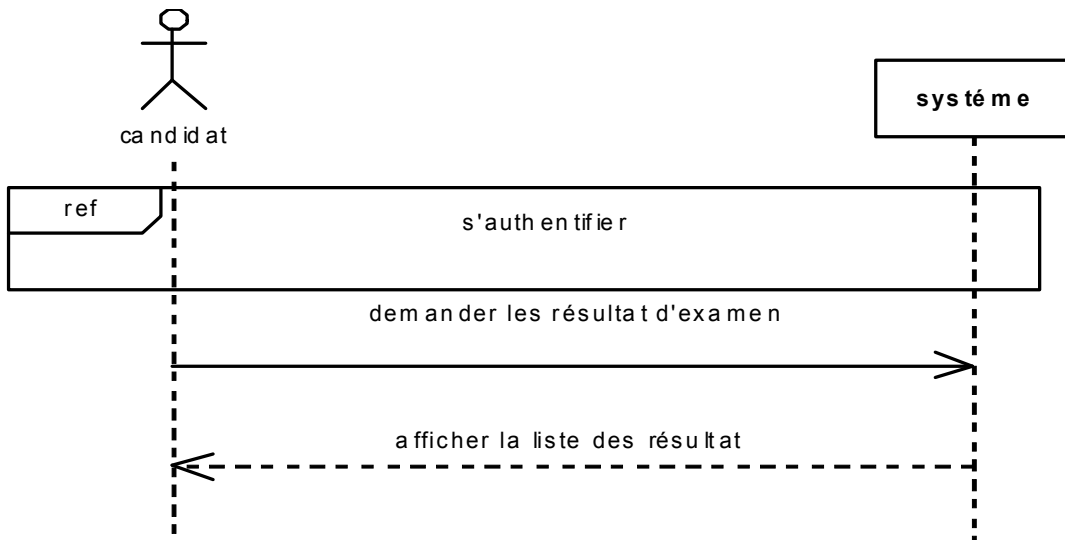


Figure 3.7 : Diagramme de séquence système «consulter les résultats d'examen».

### 3.3.8. La mise à jour des cours :

#### 3.3.8.1. Ajouter des cours :

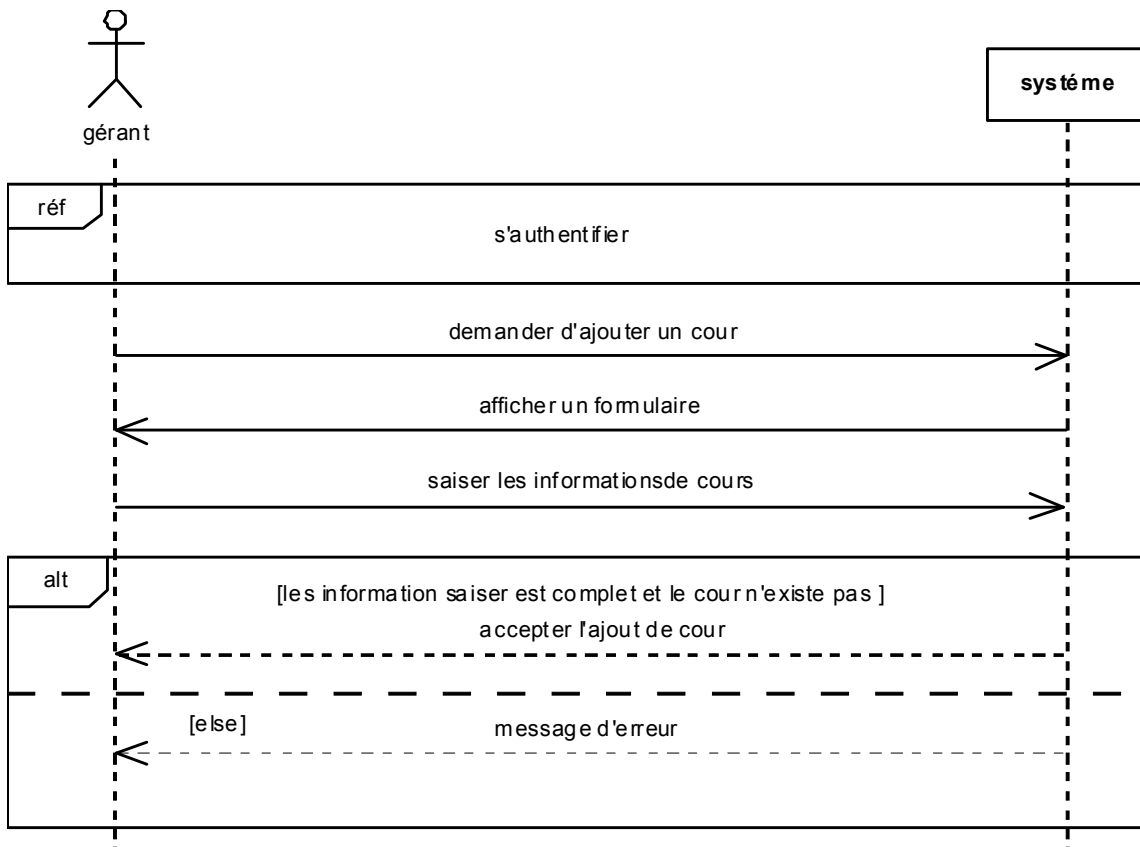


Figure 3.8 : Diagramme de séquence système «ajouter des cours»

3.3.8.2. Modifier cour :

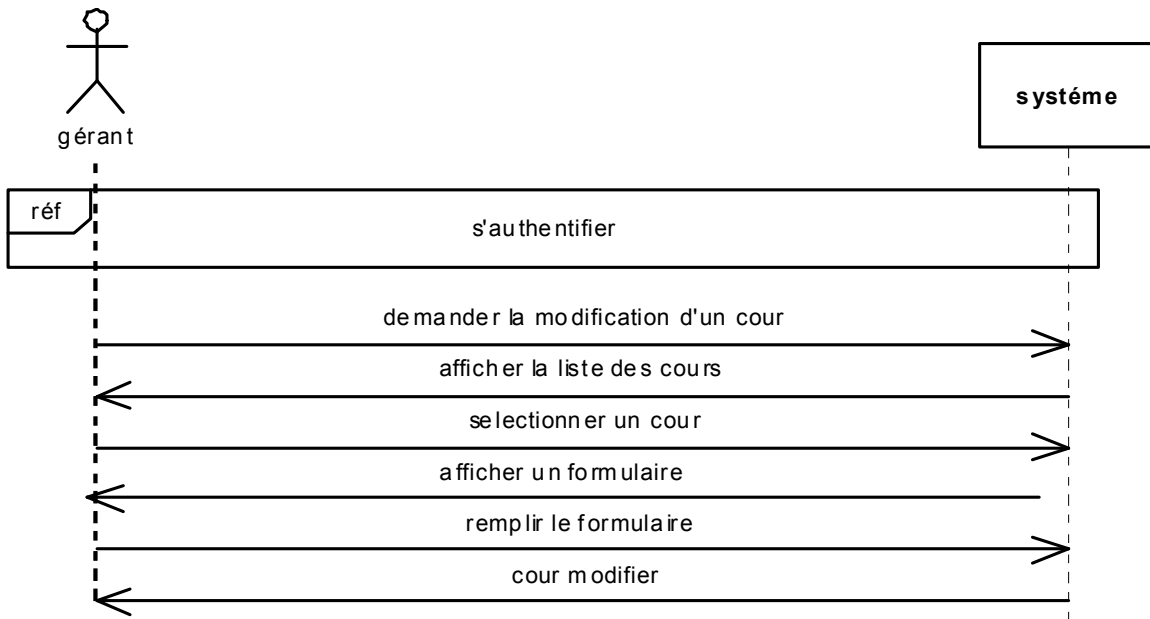


Figure 3.9 : Diagramme de séquence système «modifier cour».

3.3.9. Mise à jour des informations :

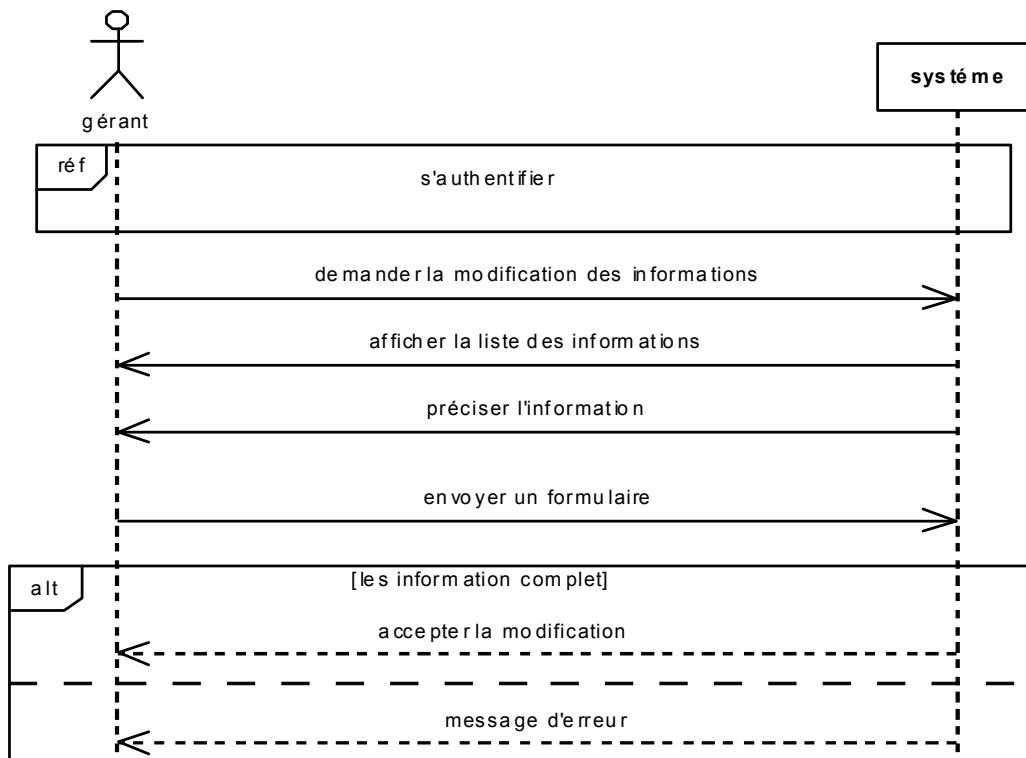


Figure 3.10 : Diagramme de séquence système «modifier les informations».

### 3.3.10. Mise à jour des annonces :

#### 3.3.10.1. Ajouter annonce :

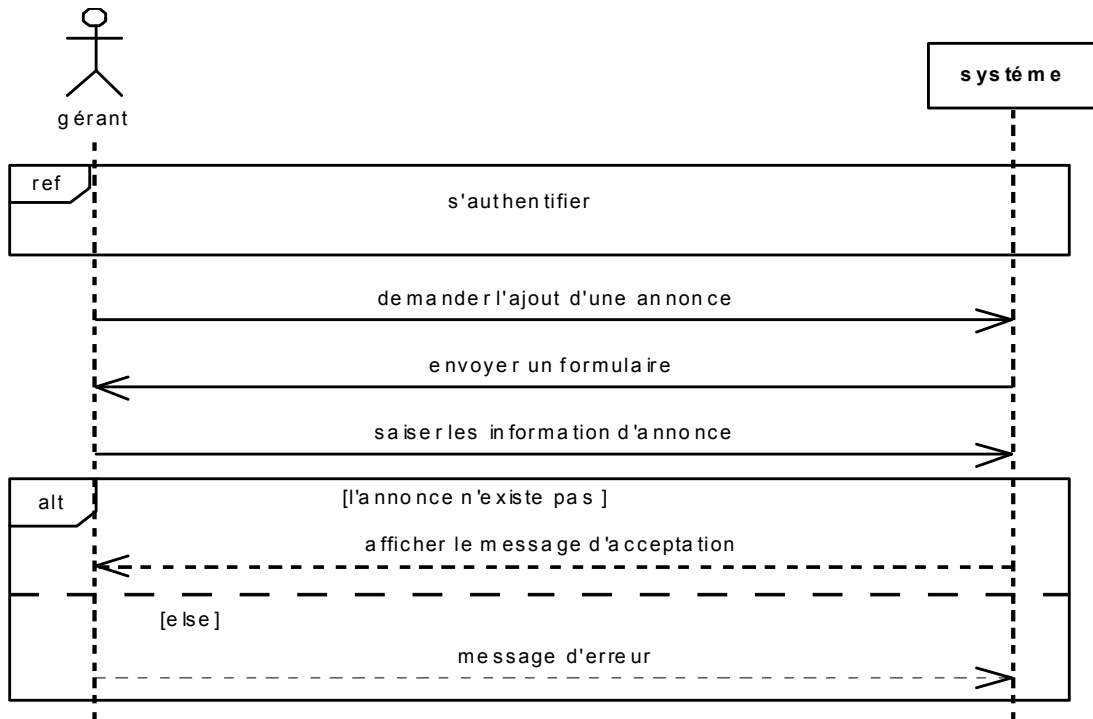


Figure 3.11 : Diagramme de séquence système «ajouter des annonces»

#### 3.3.10.2. Supprimer annonce :

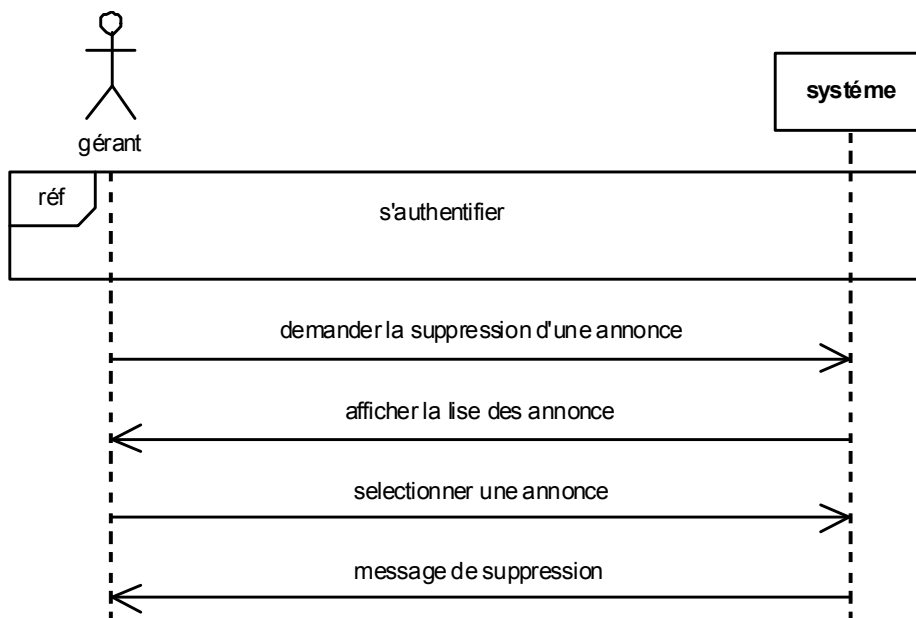


Figure 3.12 : Diagramme de séquence système «supprimer des annonces».

### 3.3.11. Gérer les comptes candidat :

#### 3.3.11.1. Créé un compte candidat :

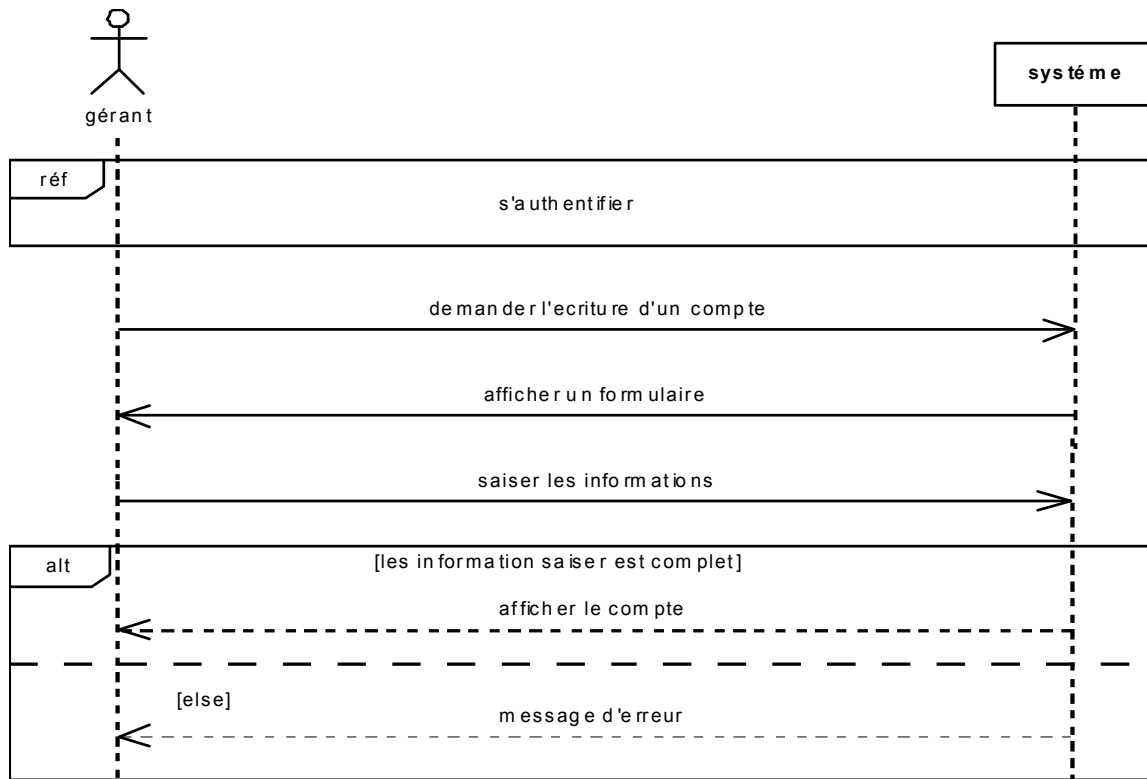


Figure 3.13 : Diagramme de séquence système «créé un compte candidat».

#### 3.3. 11.2. Supprimer un compte candidat :

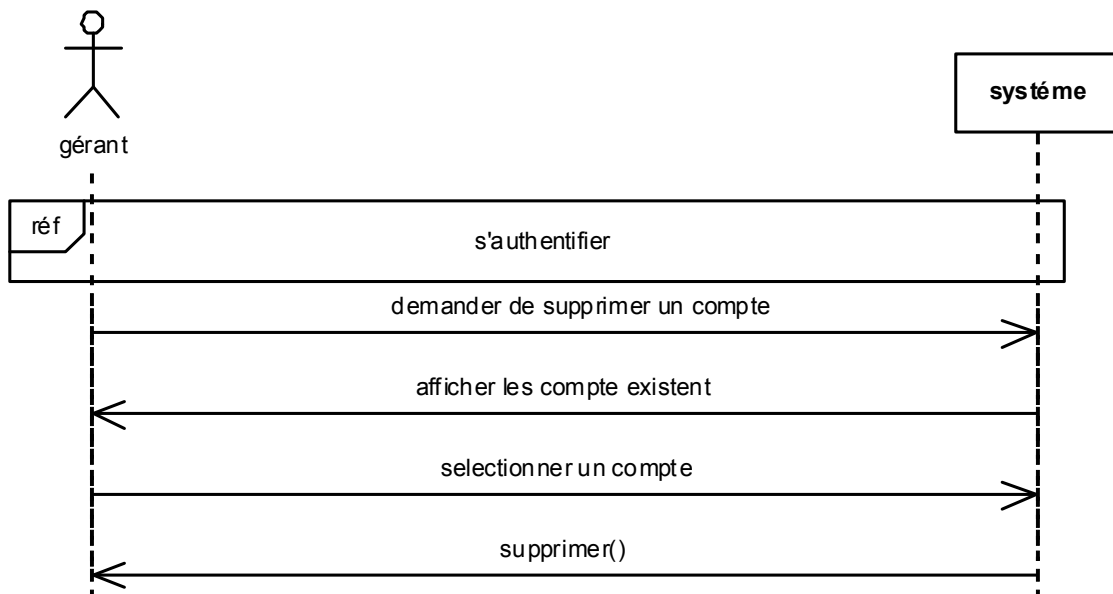


Figure 3.14 : Diagramme de séquence système «supprimer un compte candidat».

### 3.3.11.3. Modifier un compte candidat :

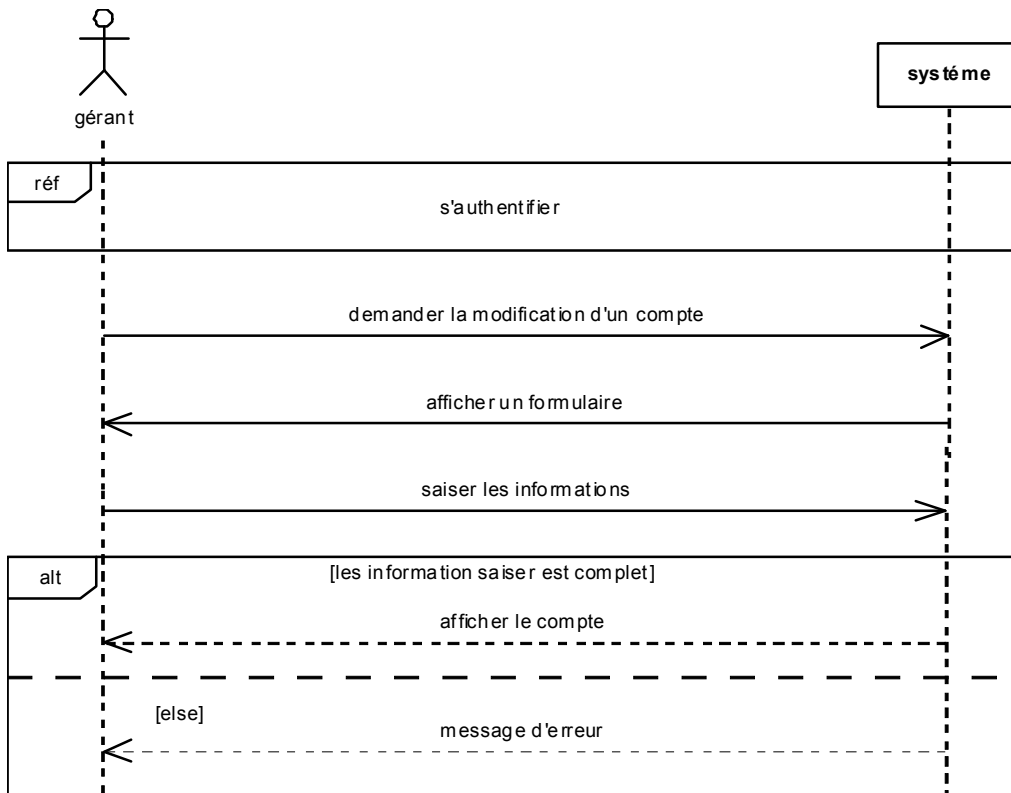


Figure 3.15 : Diagramme de séquence système «modifier un compte candidat».

### 3.3.12. Ajouter résultat:

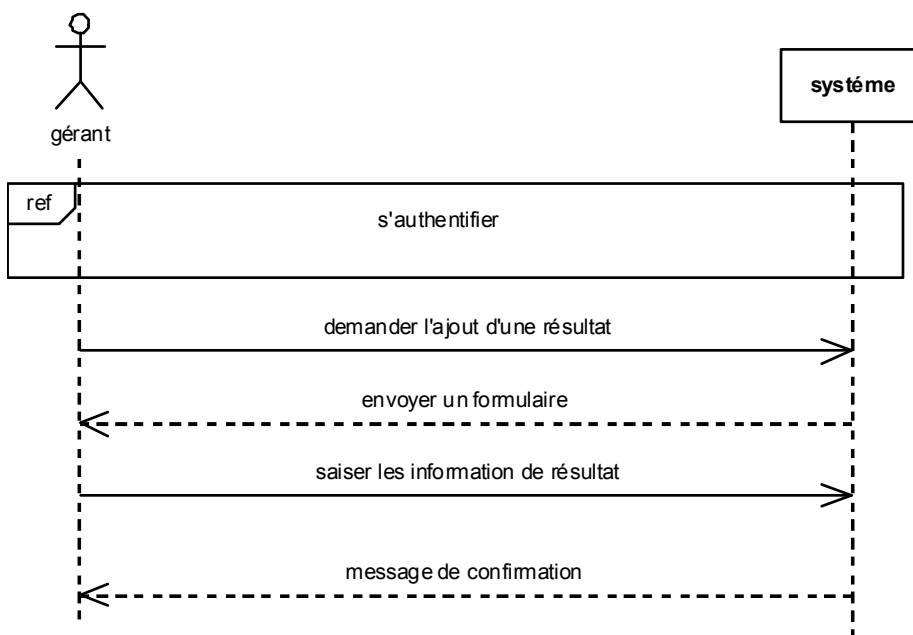


Figure 3.16 : Diagramme de séquence système «Ajouter résultat».

*Partie 02 :*  
*Phase d'analyse*

### 1. Modèle de domaine :

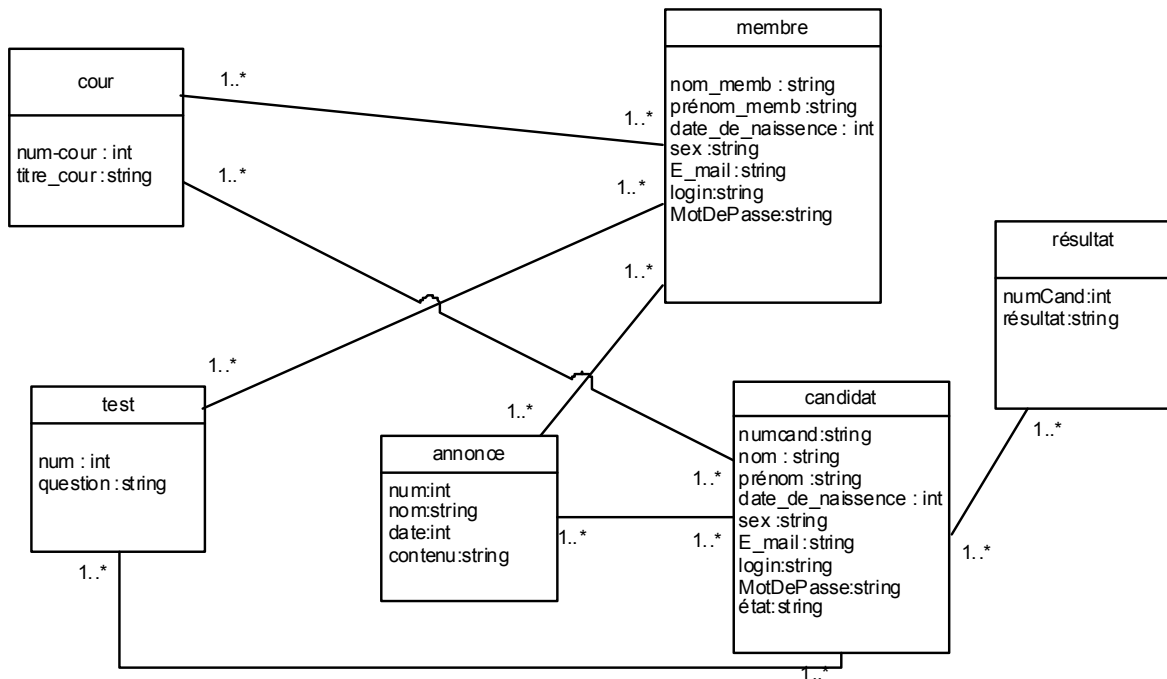


Figure 3.17 : Modèle de domaine

### 2. Diagramme de classe participante :

#### 2.1. S’authentifier :

##### 2.1.1 : S’authentifier gérant :

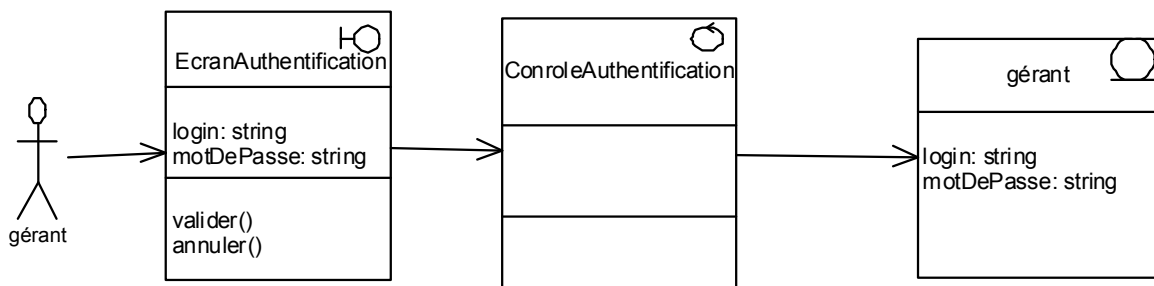
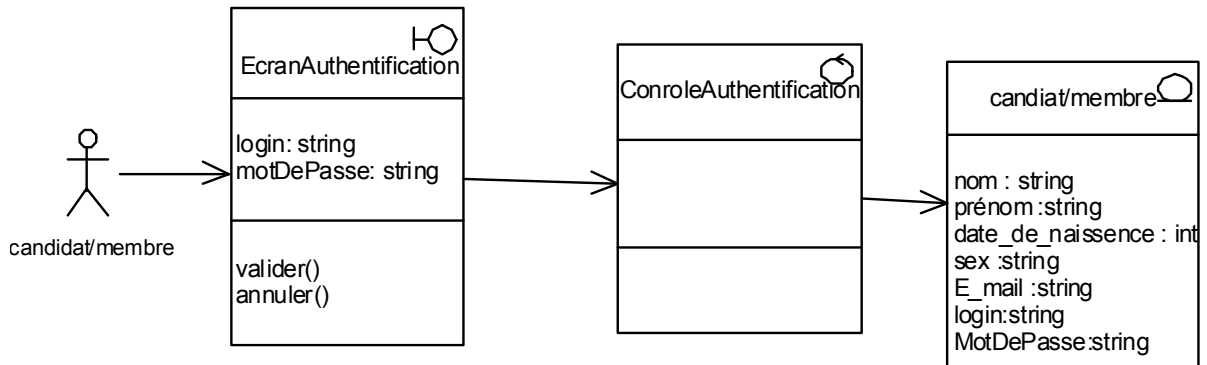


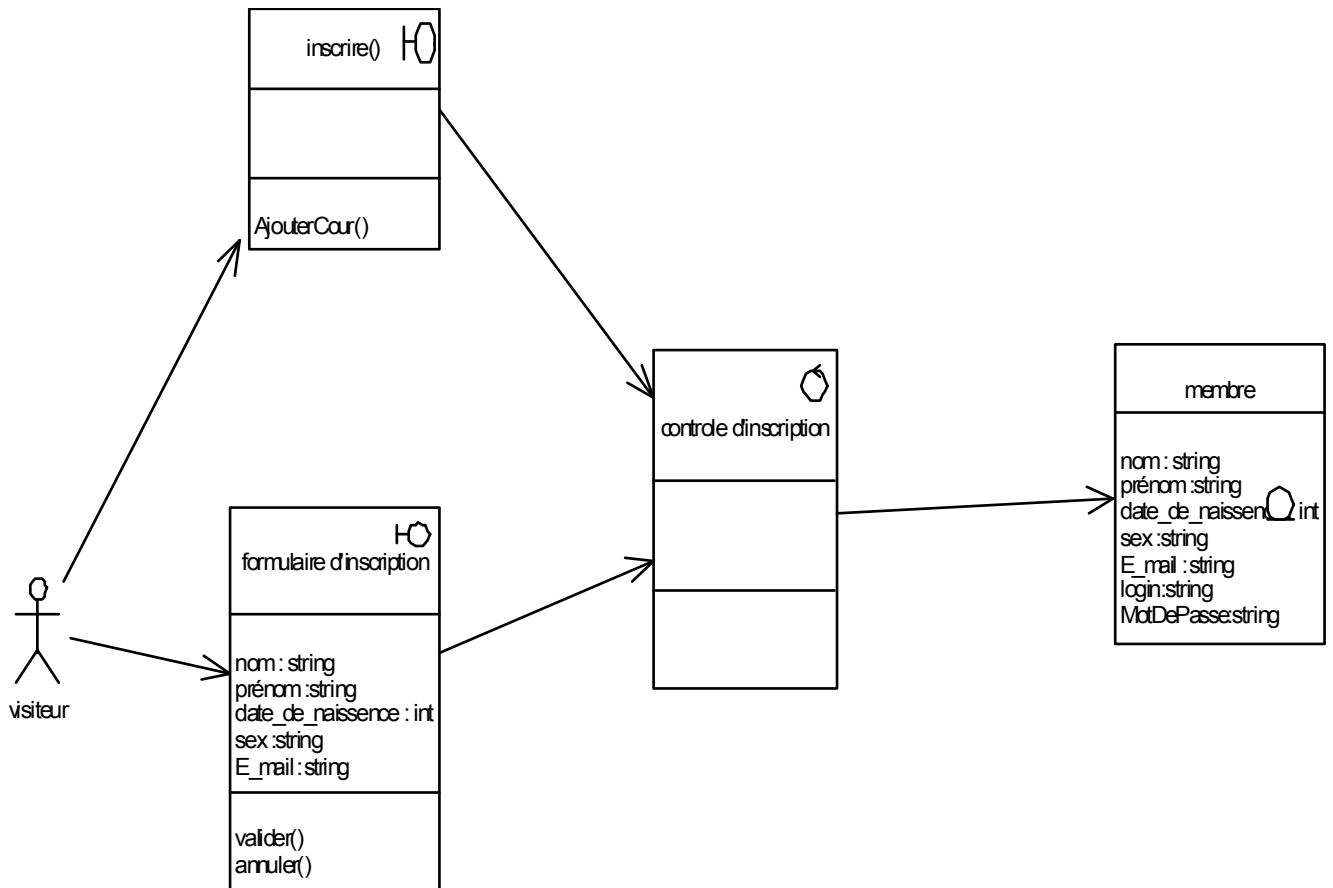
Figure 3.18 : Diagramme de classes participantes du cas d’utilisation « S’authentifier gérant ».

**2.1.2. S’authentifier candidat/membre :**



**Figure 3.19 :** Diagramme de classes participantes du cas d’utilisation «S’authentifie candidat/membre».

**2.2-inscrire :**



**Figure 3.20 :** Diagramme de classes participantes du cas d’utilisation «S’inscrire».

### 2.3-passer test :

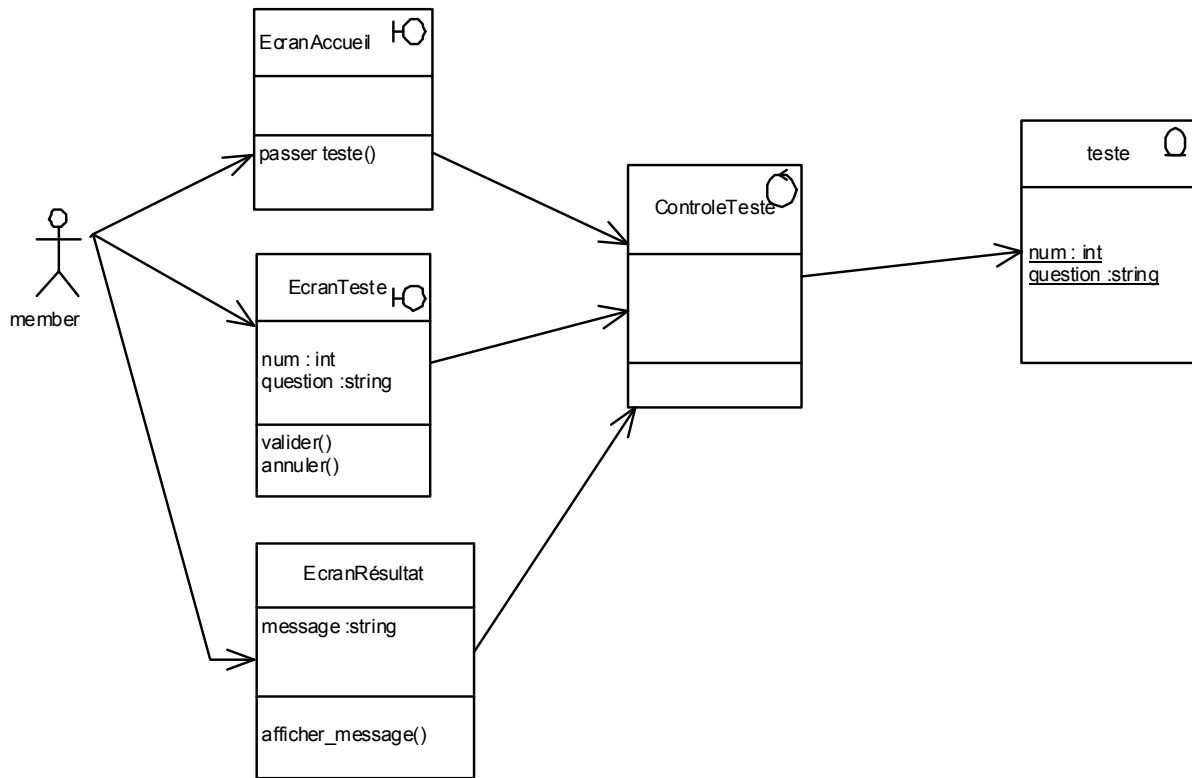


Figure 3.21 : Diagramme de classes participantes du cas d'utilisation «passer test».

### 2.4- ajouter cour :

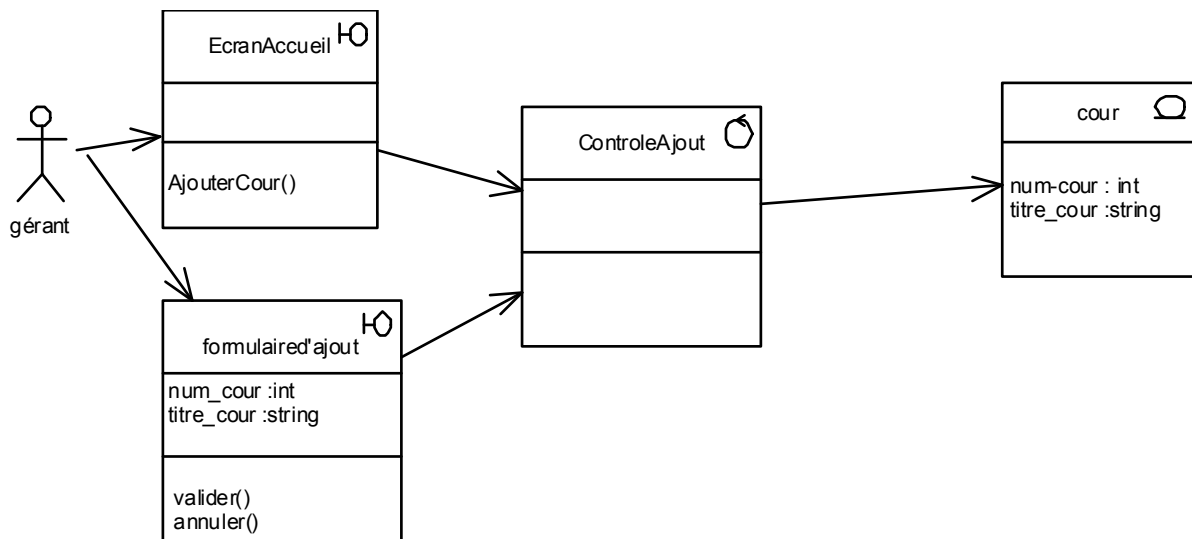


Figure 3.22 : Diagramme de classes participantes du cas d'utilisation «ajouter cour».

### 2.5-ajouter annonce

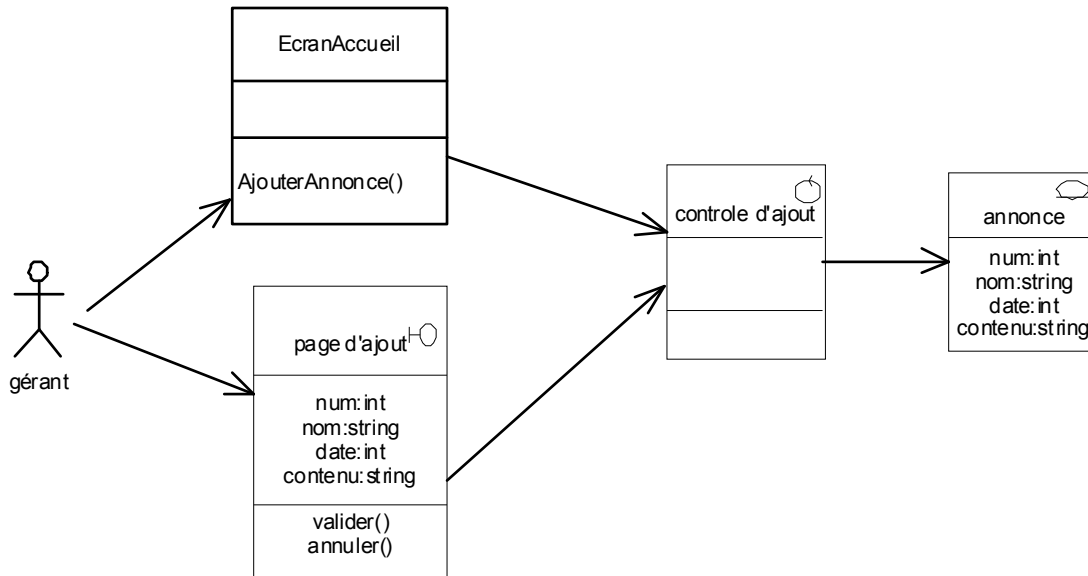


Figure 3.23 : Diagramme de classes participantes du cas d'utilisation «ajouter annonce».

### 2.6-supprimer annonce :

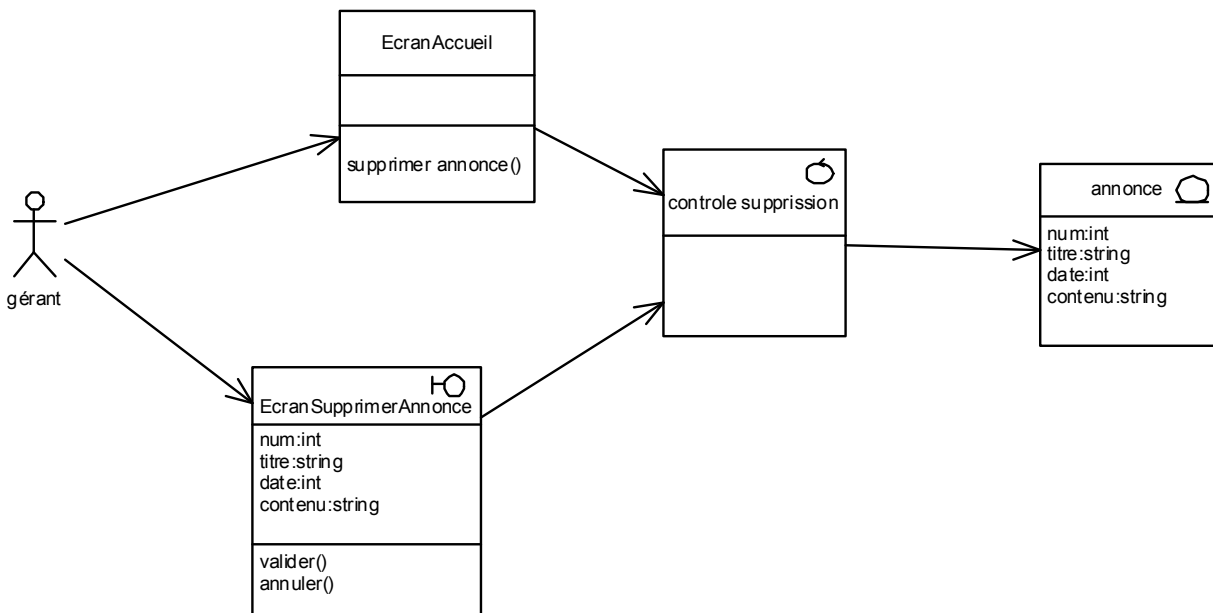


Figure 3.24 : Diagramme de classes participantes du cas d'utilisation «supprimer annonce».

### 2.7- Créé un compte candidat :

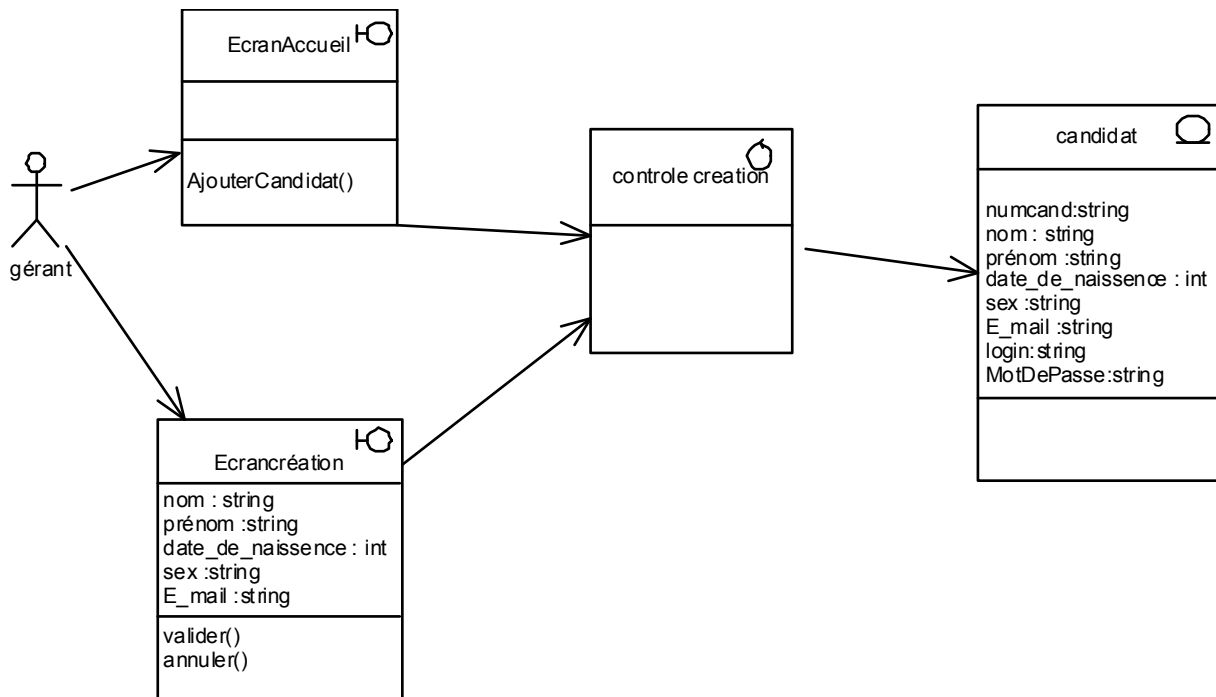


Figure 3.25 : Diagramme de classes participantes du cas d'utilisation «créé un compte candidat»

### 2.8- Supprimer un compte candidat :

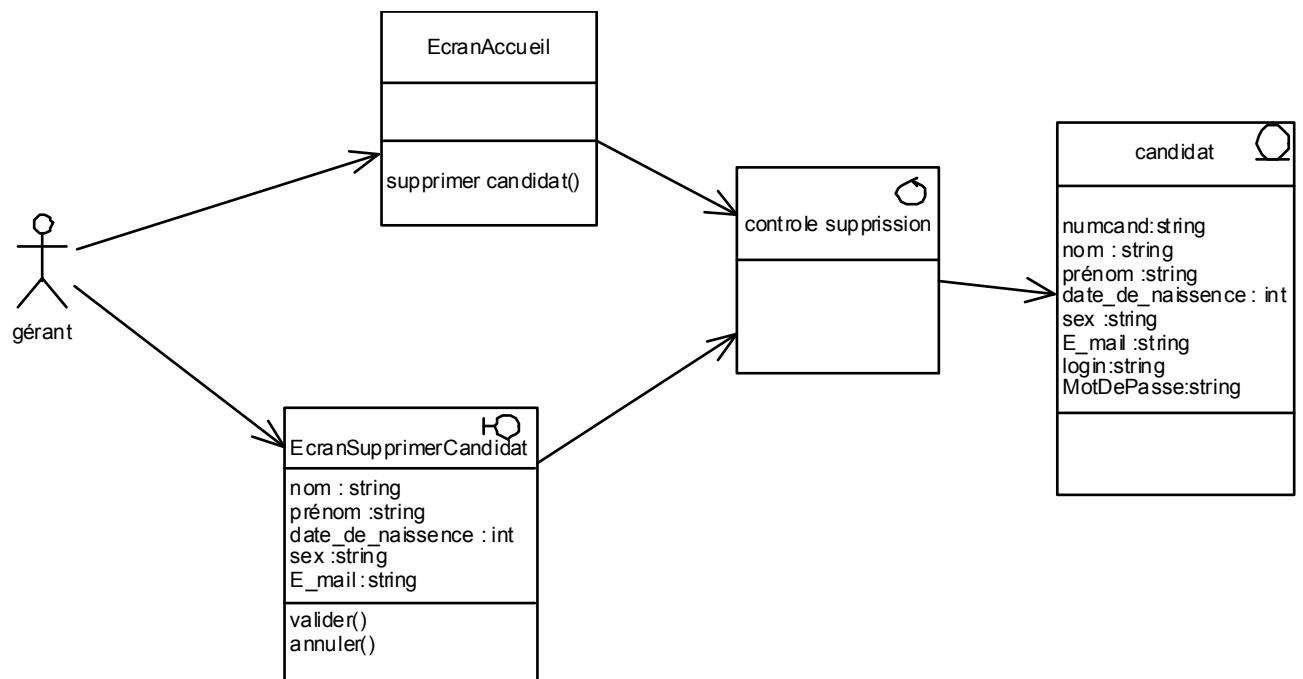


Figure 3.26 : Diagramme de classes participantes du cas d'utilisation «supprimer un compte candidat».

2.9- Modifier candidat :

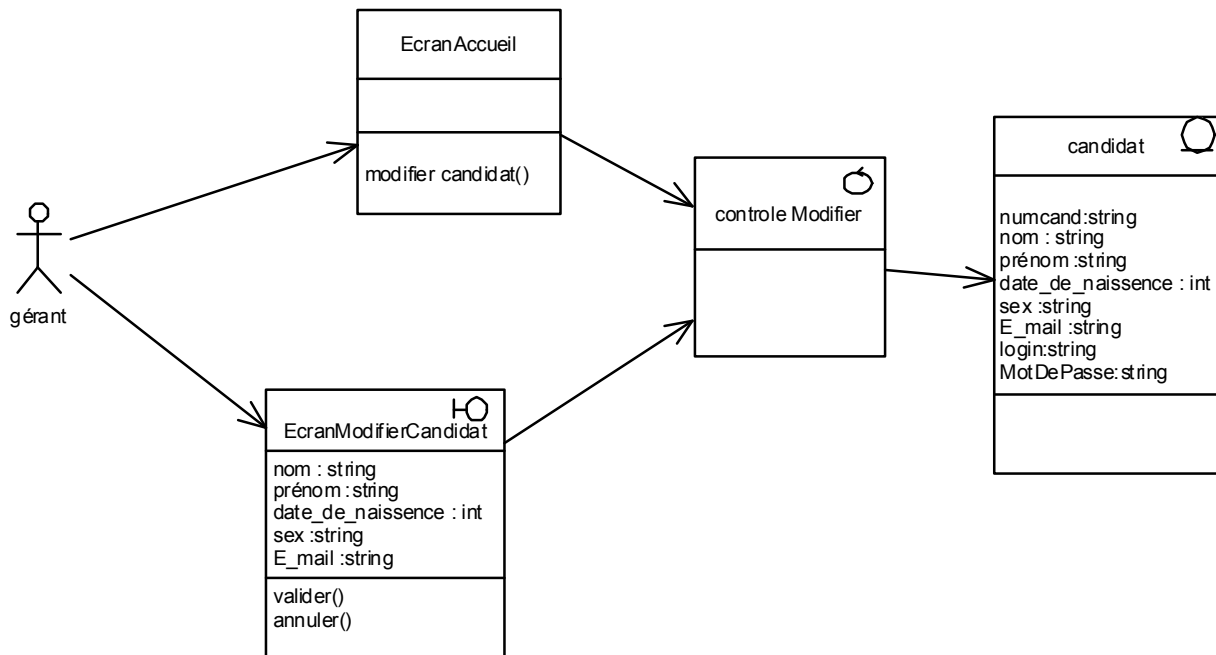


Figure 3.27 : Diagramme de classes participantes du cas d'utilisation «modifier un compte candidat»

2.10-Ajouter les résultats :

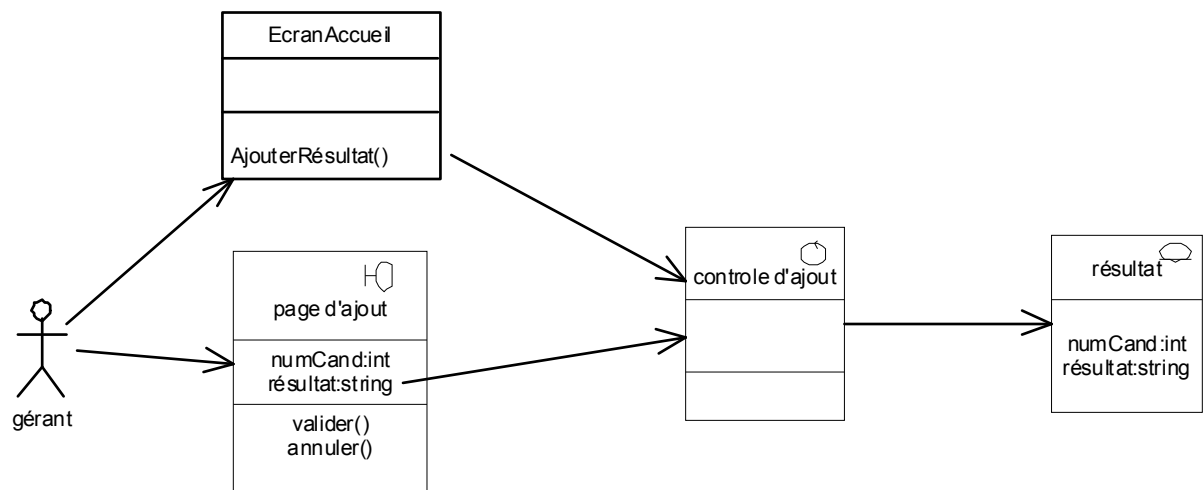


Figure 3.28 : Diagramme de classes participantes du cas d'utilisation «ajouter résultat».

### 3-Diagramme d'activité navigation :

#### 3.1-Authentifier :

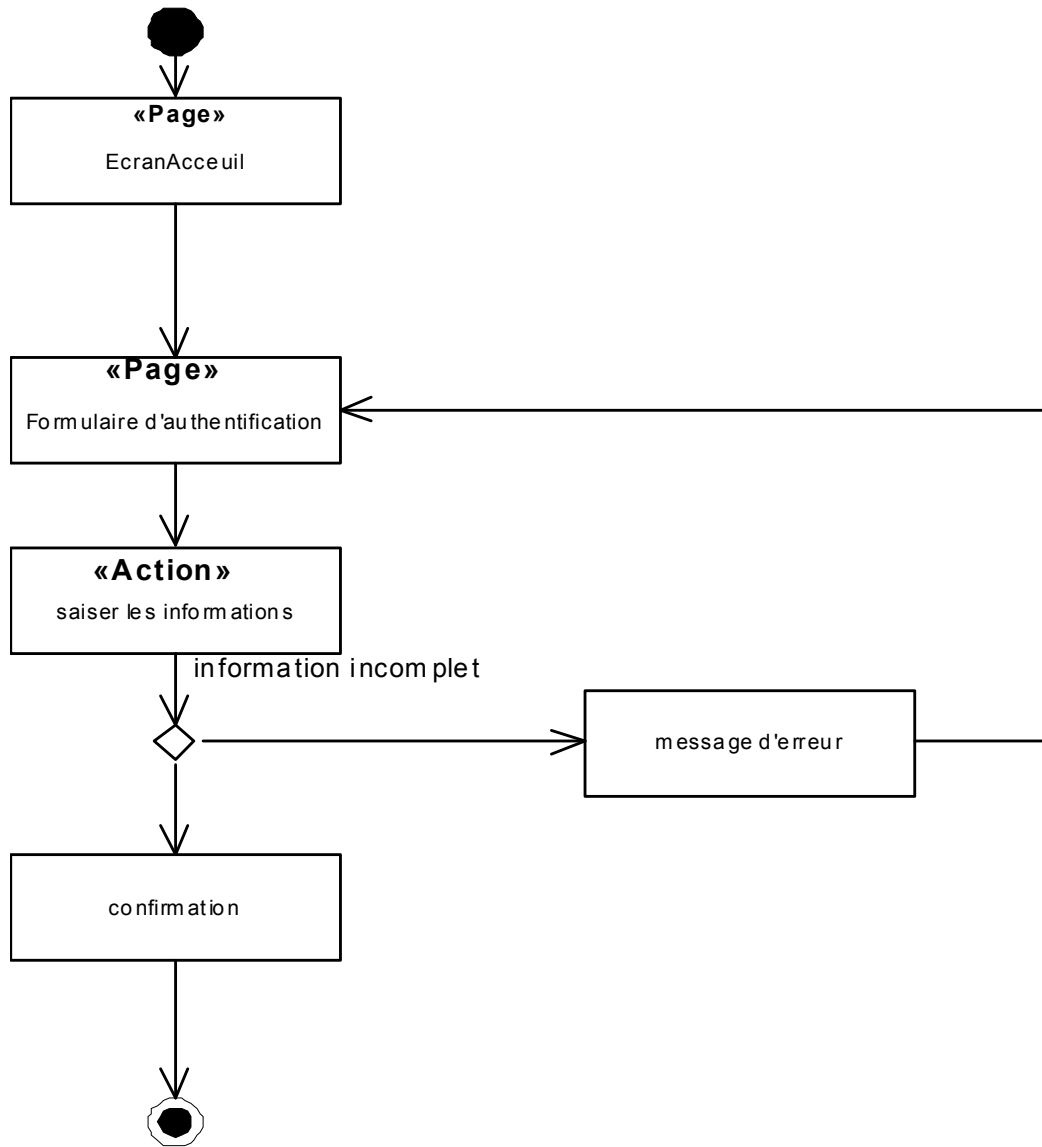


Figure 3.29 : Diagramme d'activité du cas d'utilisation « s'authentifier ».

3.2-Inscrire :

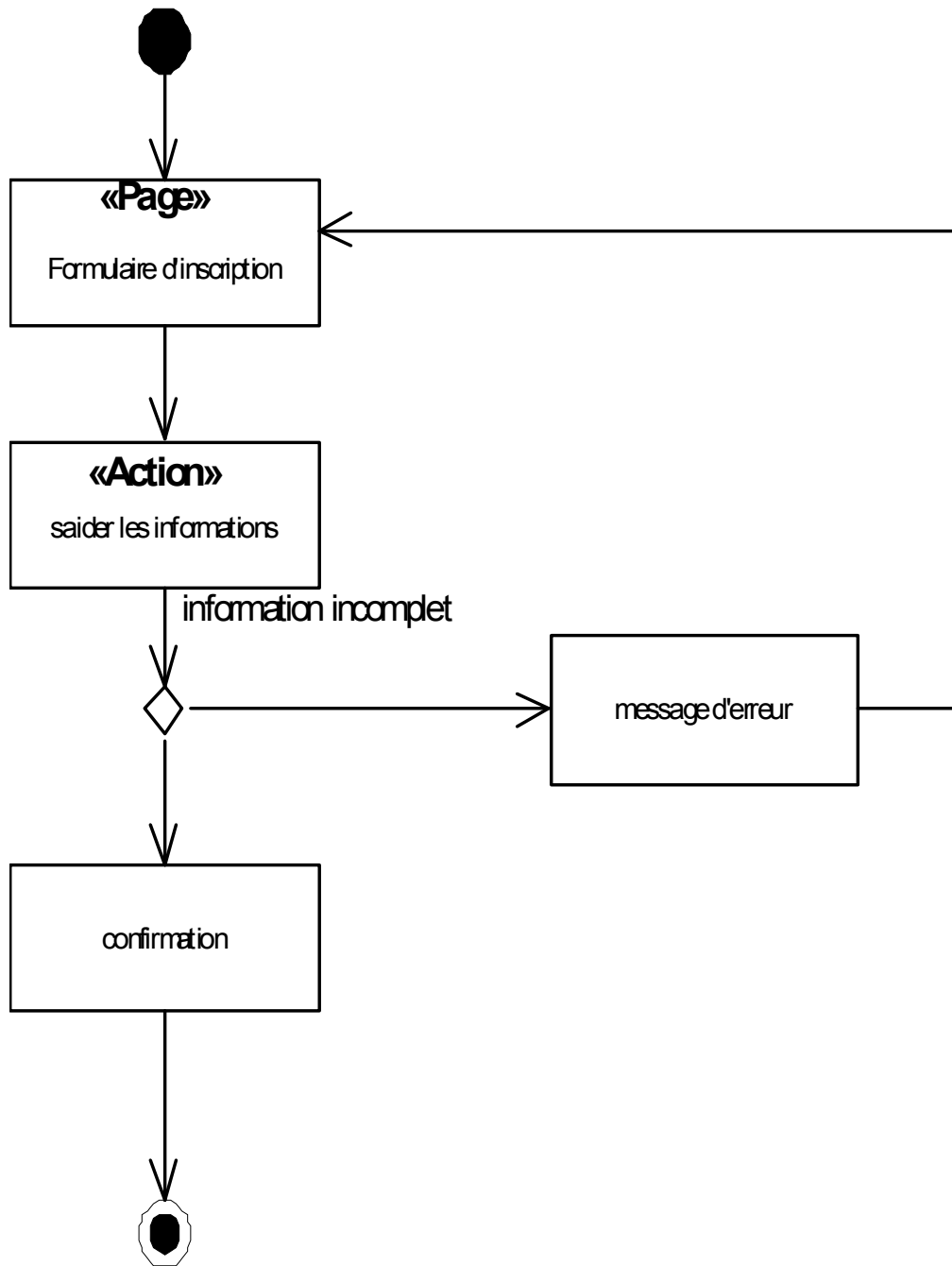


Figure 3.30 : Diagramme d'activité du cas d'utilisation « s'inscrire ».

3.3-Passer test :

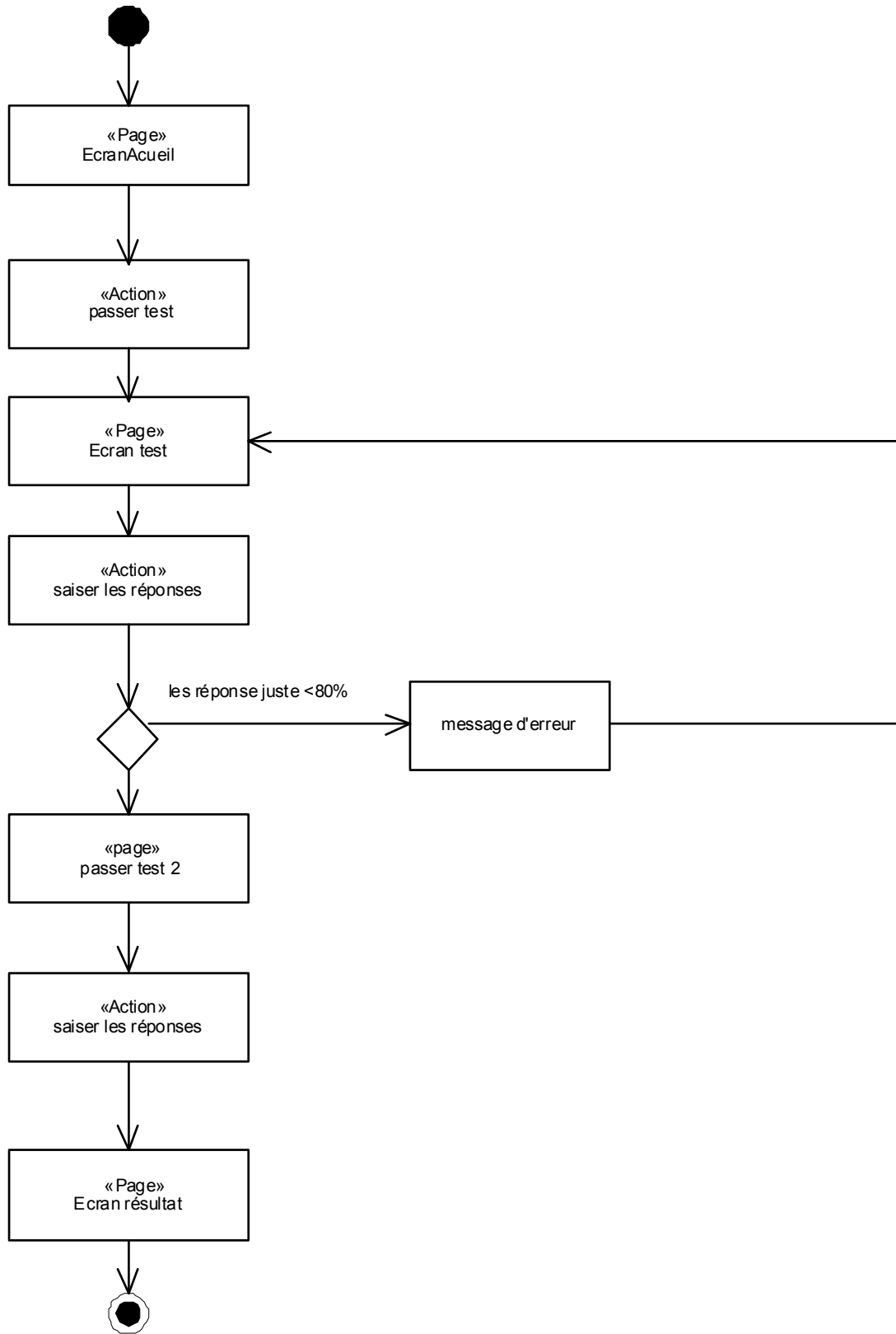


Figure 3.31 : Diagramme d'activité du cas d'utilisation « passer test ».

3.4-Ajouter cour :

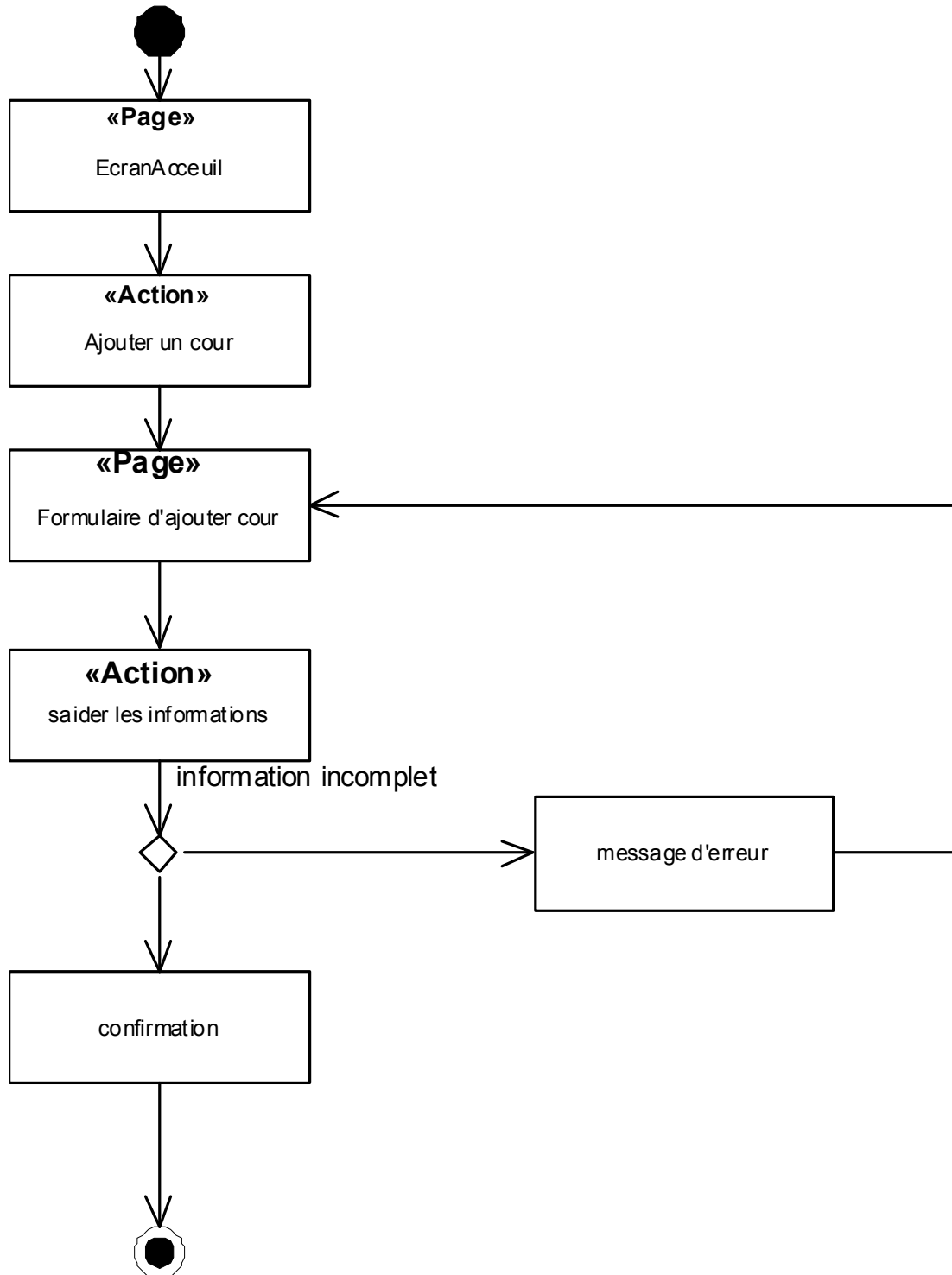


Figure 3.32 : Diagramme d'activité du cas d'utilisation « ajouter cour ».

3.5-Ajouter annonce :

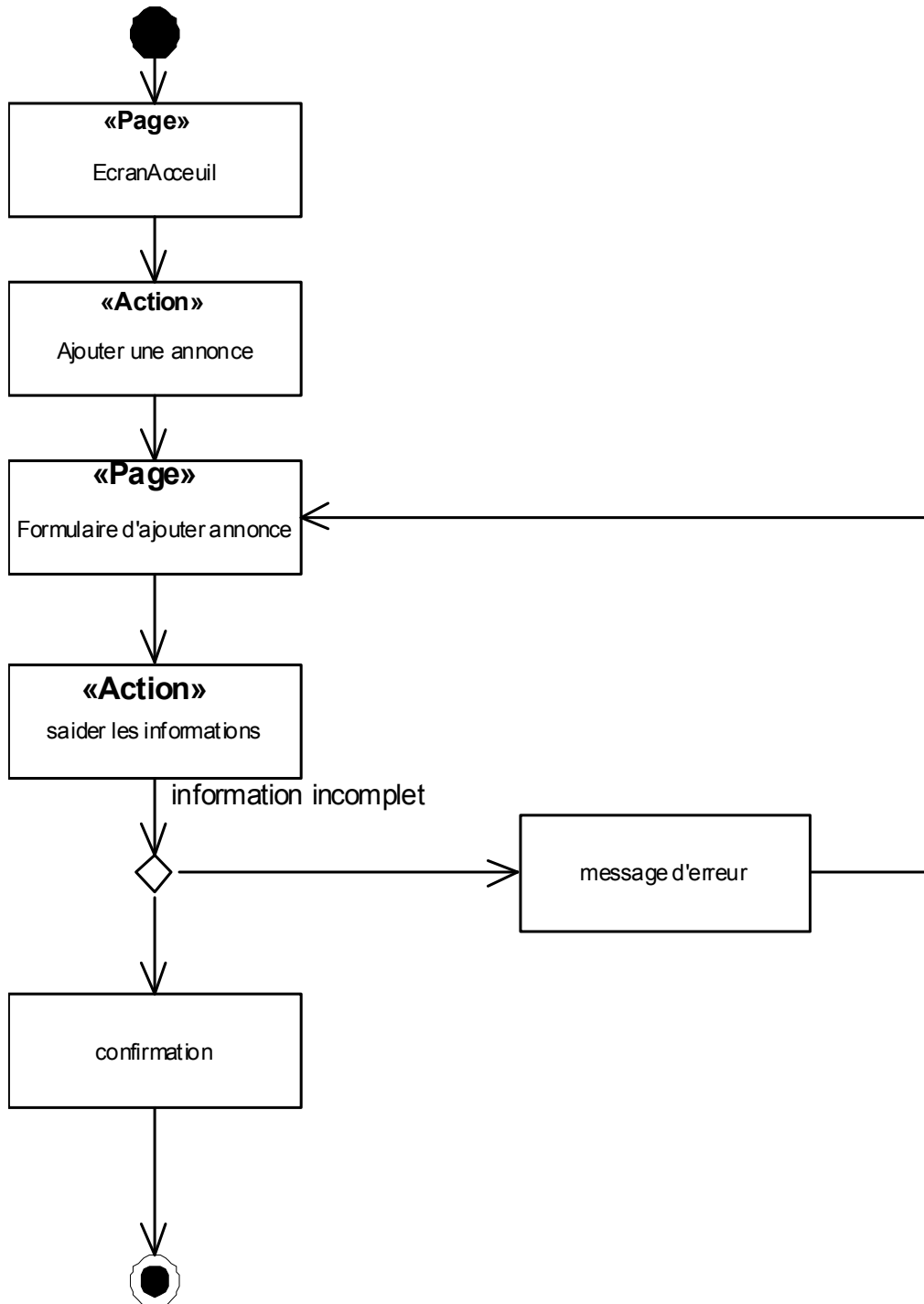


Figure 3.33 : Diagramme d'activité du cas d'utilisation « ajouter annonce ».

### 3.6-Supprimer annonce :

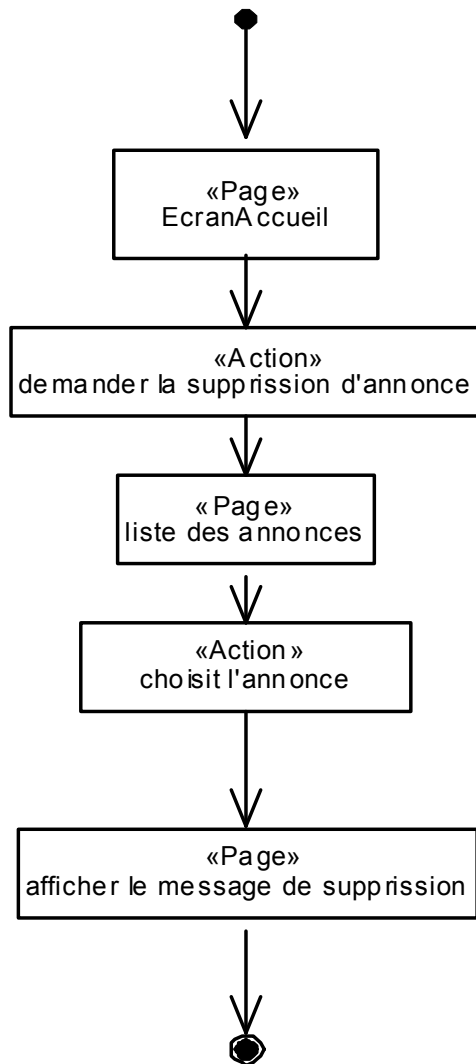


Figure 3.34 : Diagramme d'activité du cas d'utilisation « supprimer annonce ».

3.7- Crée un compte candidat :

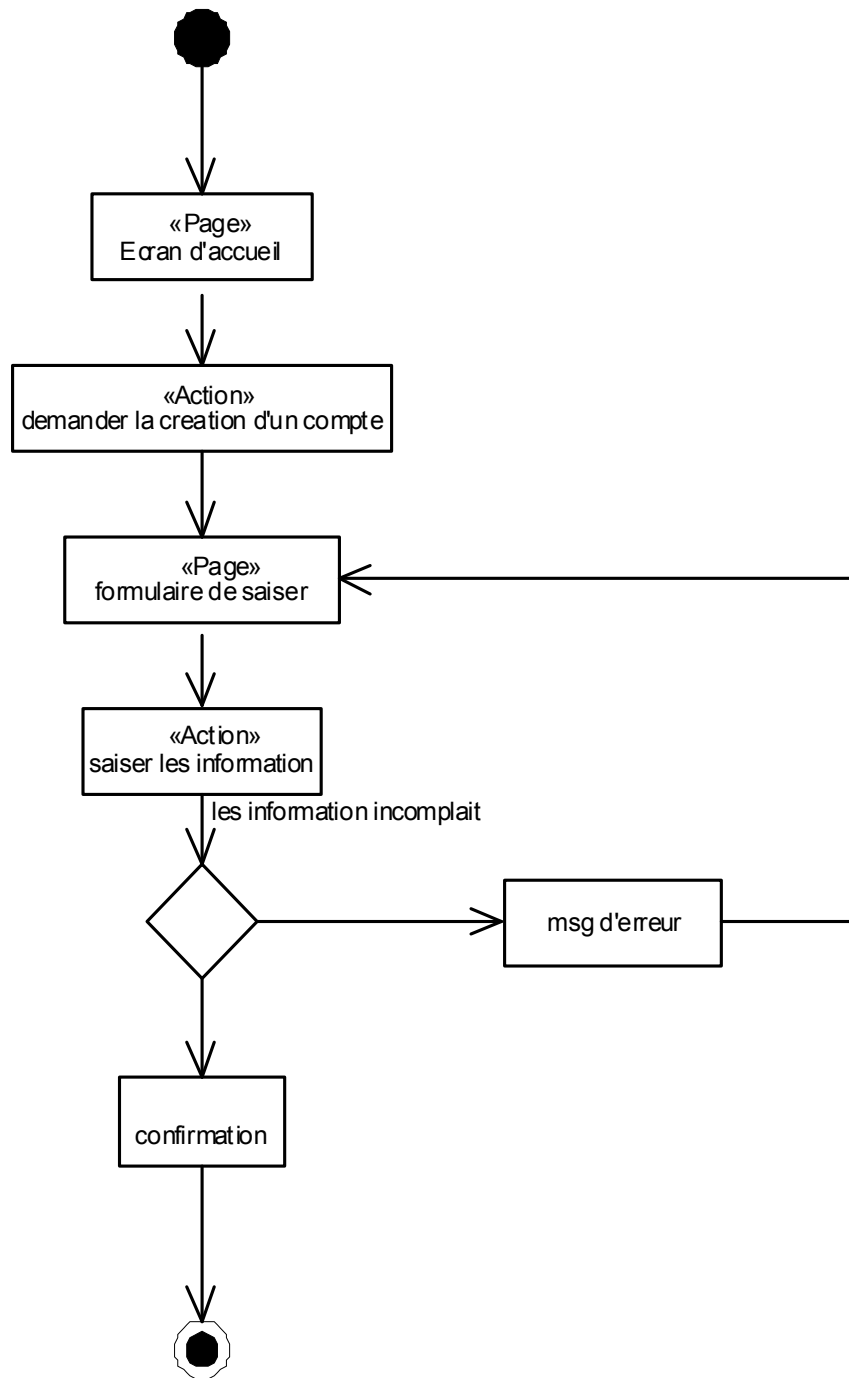


Figure 3.35 : Diagramme d'activité du cas d'utilisation « crée un compte ».

### 3.8-Supprimer un compte candidat :

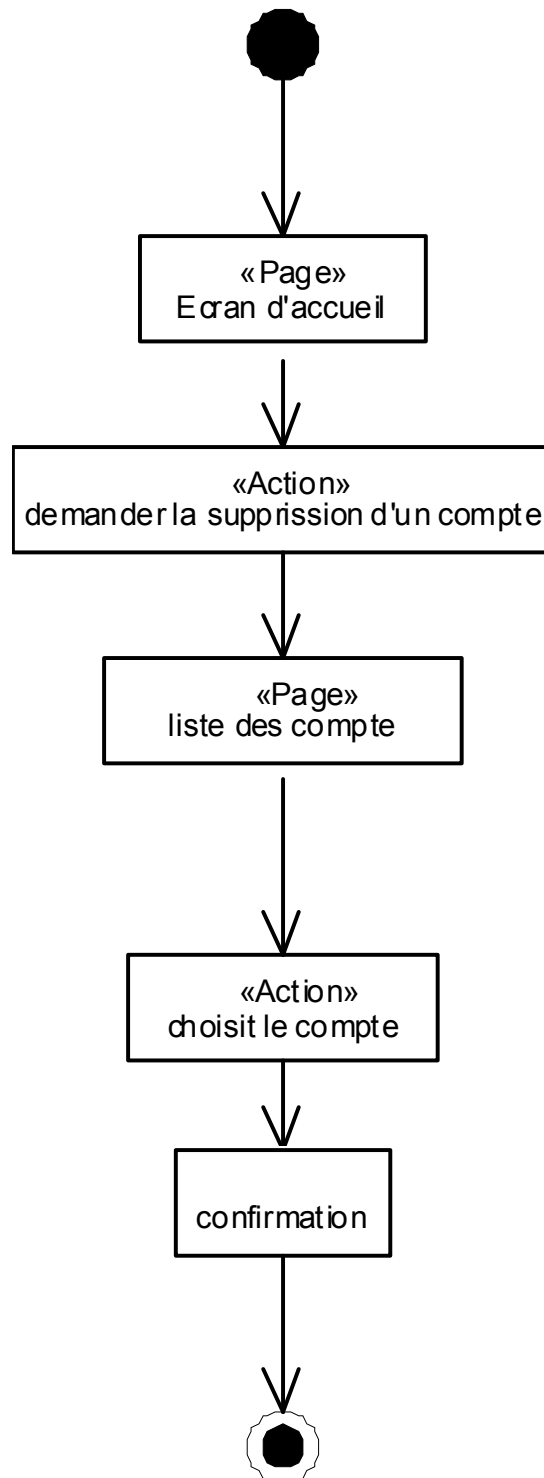


Figure 3.36 : Diagramme d'activité du cas d'utilisation « supprimer un compte ».

### 3.9- Modifier un compte candidat :

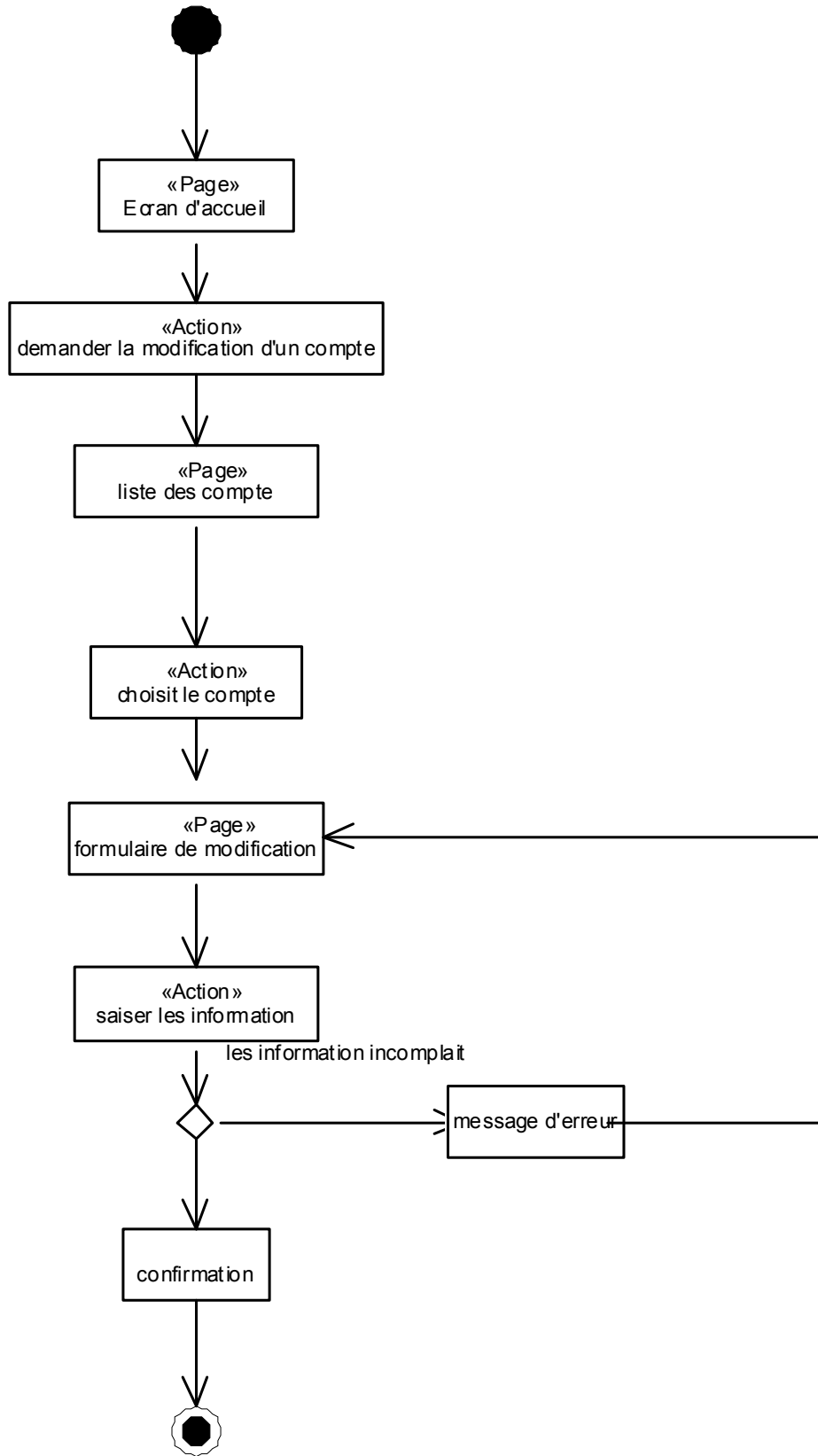


Figure 3.37 : Diagramme d'activité du cas d'utilisation « modifier un compte ».

3.10-Ajouter résultat :

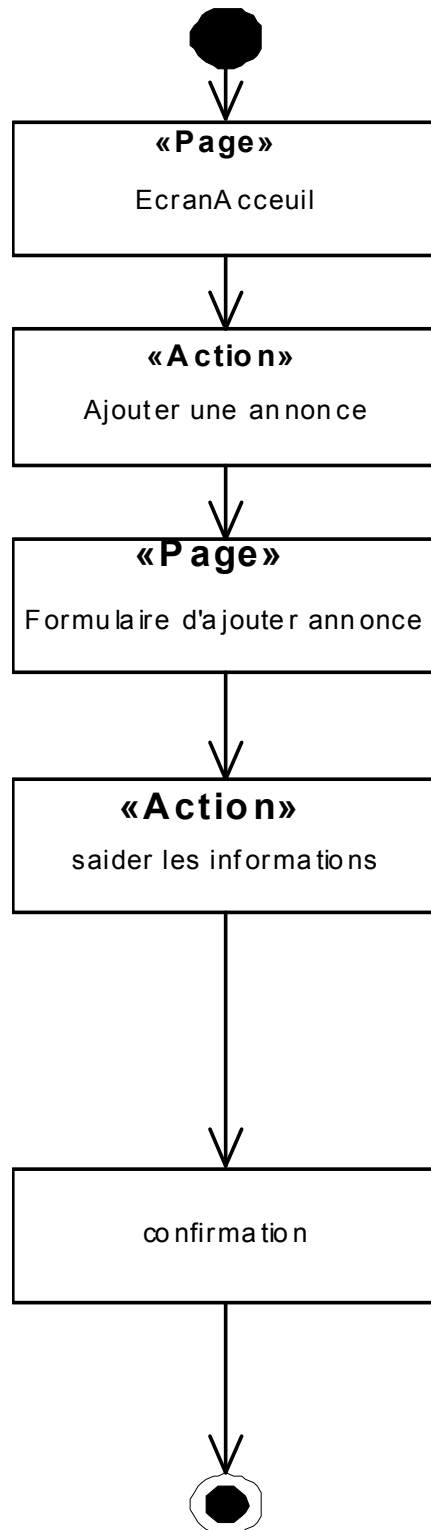


Figure 3.38 : Diagramme d’activité du cas d’utilisation « ajouter résultat».

***Partie 03 :***  
***Phase conception***

### 1. Introduction :

La conception des applications vient après l’analyse pour enlever l’abstraction qui caractérise la phase d’analyse et pour présenter clairement la conception du système.

Dans ce chapitre nous expliquons l’architecture des différentes fonctionnalités de notre application, nous allons élaborer les diagrammes d’interaction qui sont basés sur les diagrammes de séquence décrits dans le chapitre précédent.

### 2. Diagrammes d’interactions :

#### 2.1-S’authentifier:

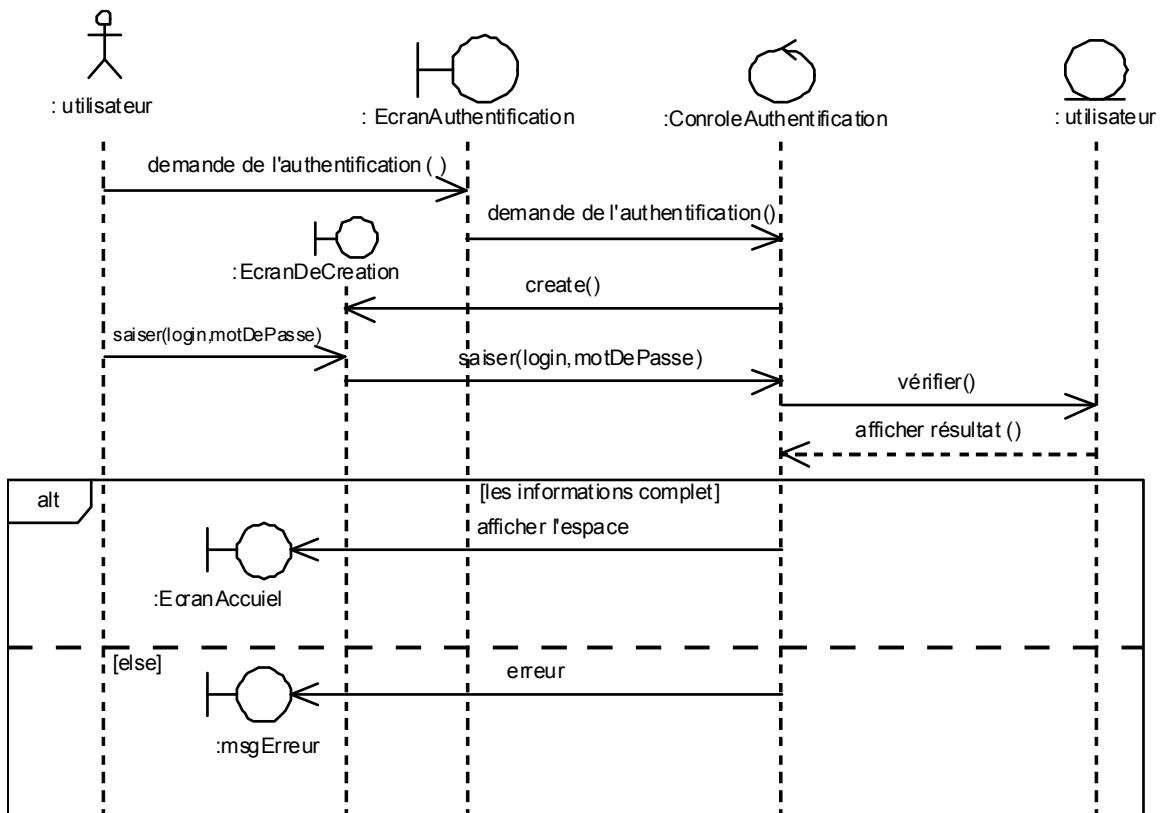


Figure 3.39 : Diagramme d’interactions du cas d’utilisation « S’authentifier»

2.2- Inscrire :

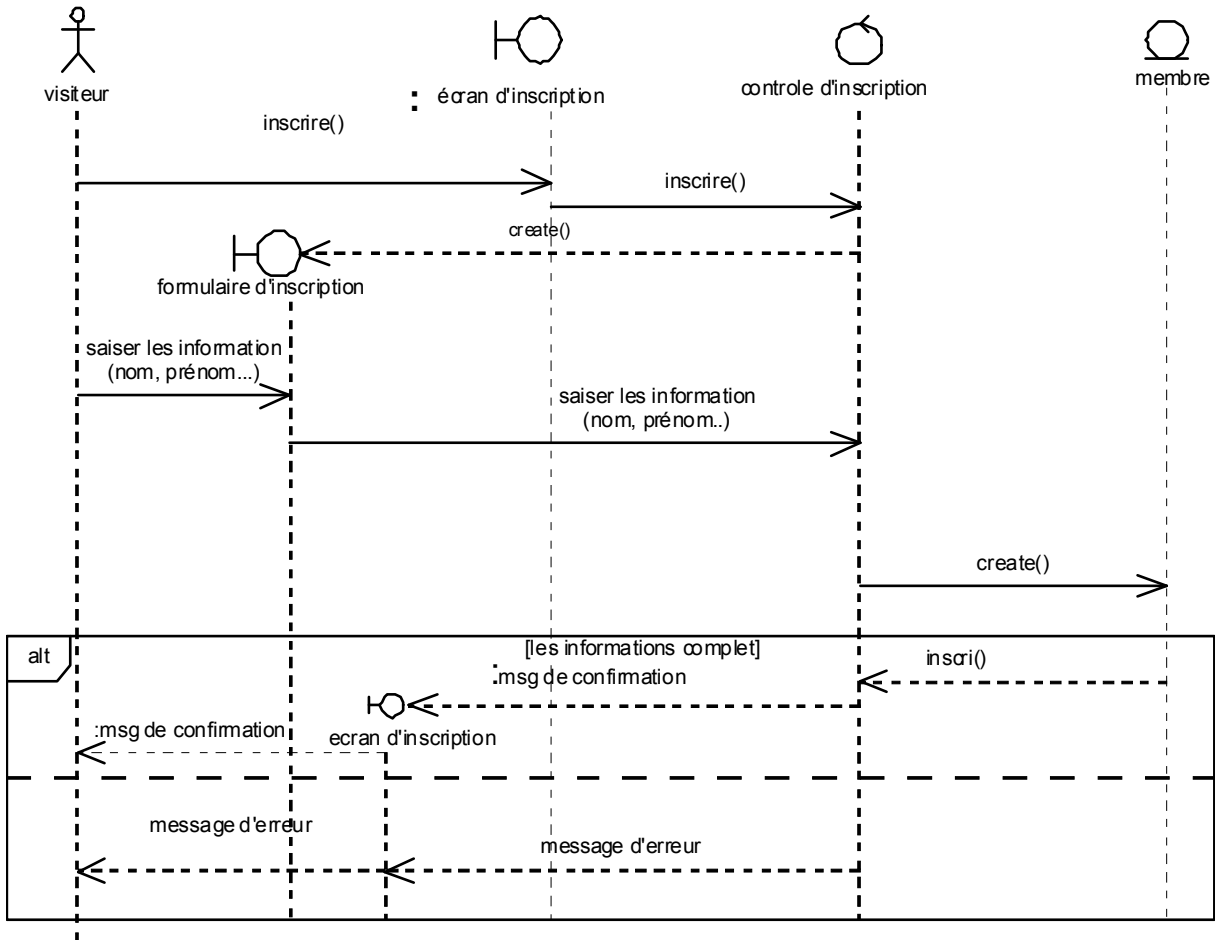


Figure 3.40 : Diagramme d'interactions du cas d'utilisation « inscrire »

2.3- Passer test :

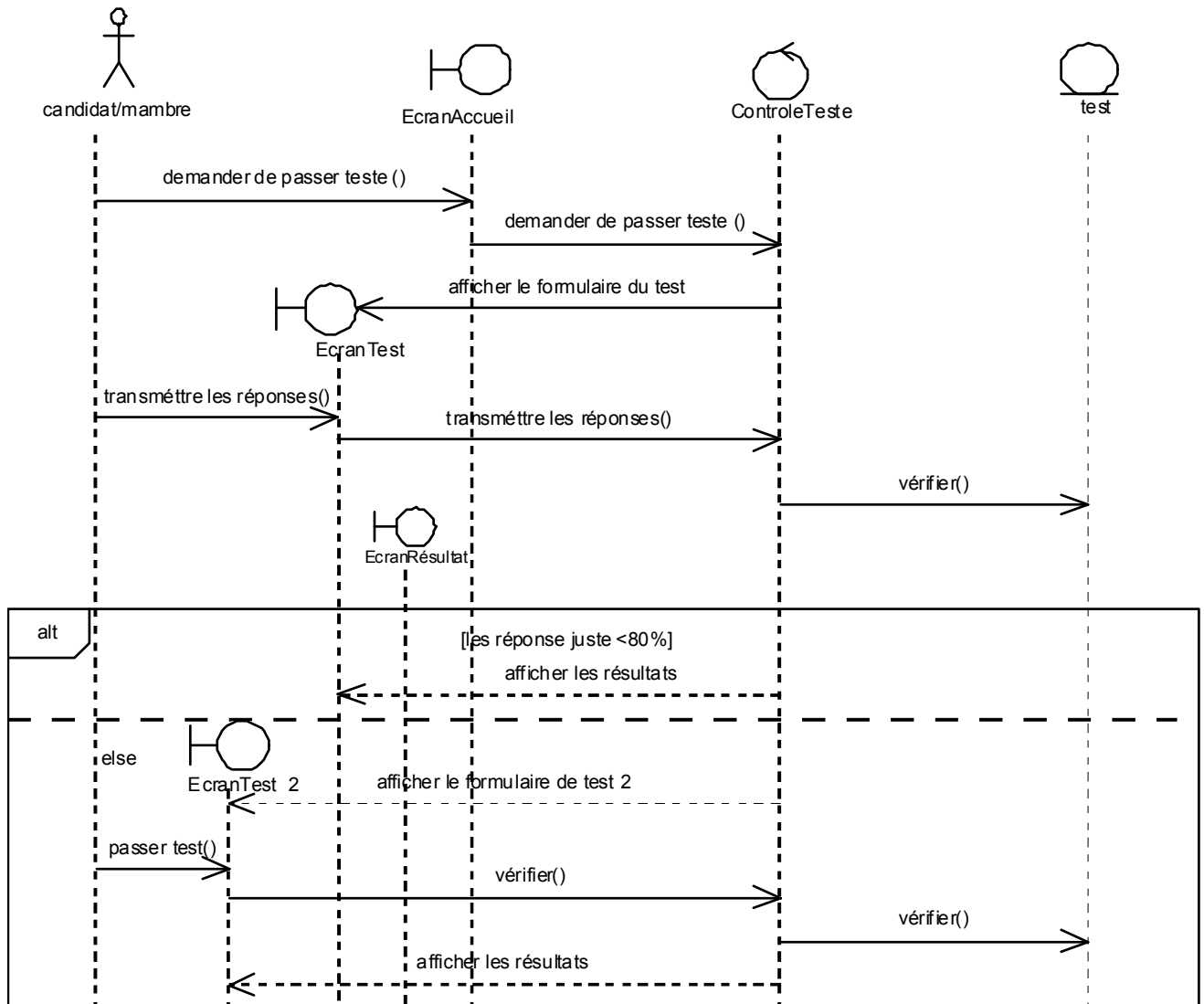


Figure 3.41 : Diagramme d'interactions du cas d'utilisation « passer test»

### 2.4-Ajouter cour :

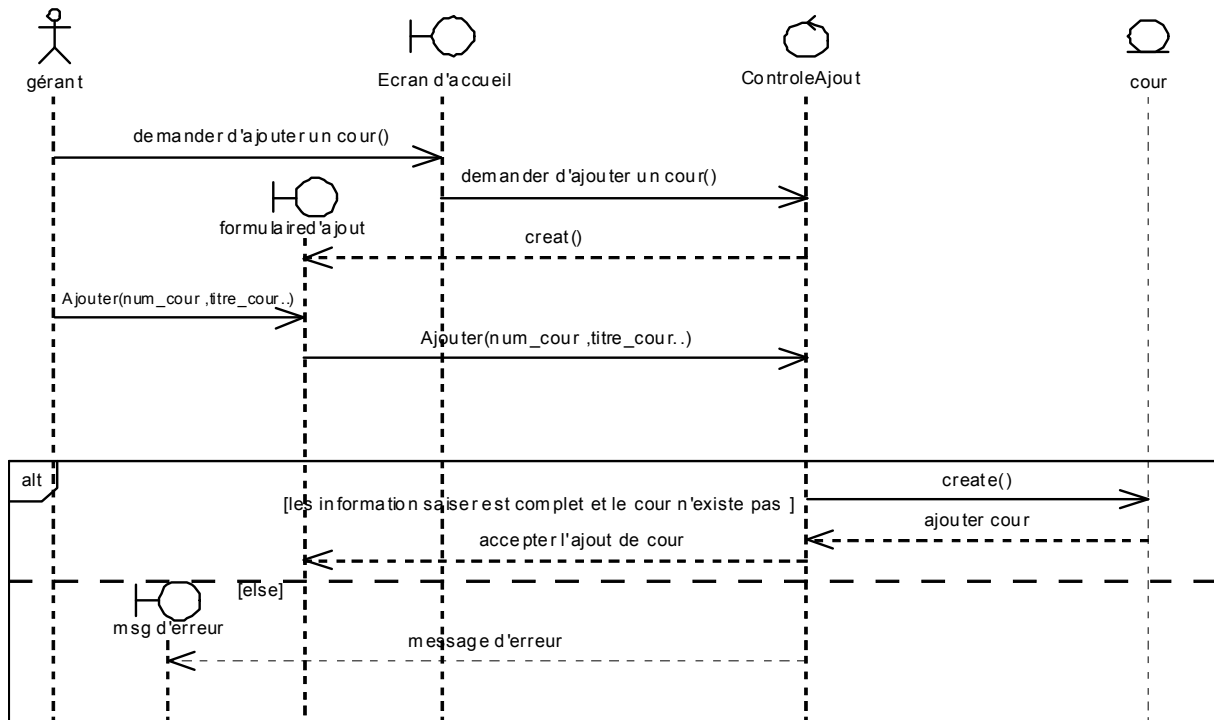


Figure 3.42 : Diagramme d'interactions du cas d'utilisation « ajouter cour»

### 2.5-Ajouter annonce :

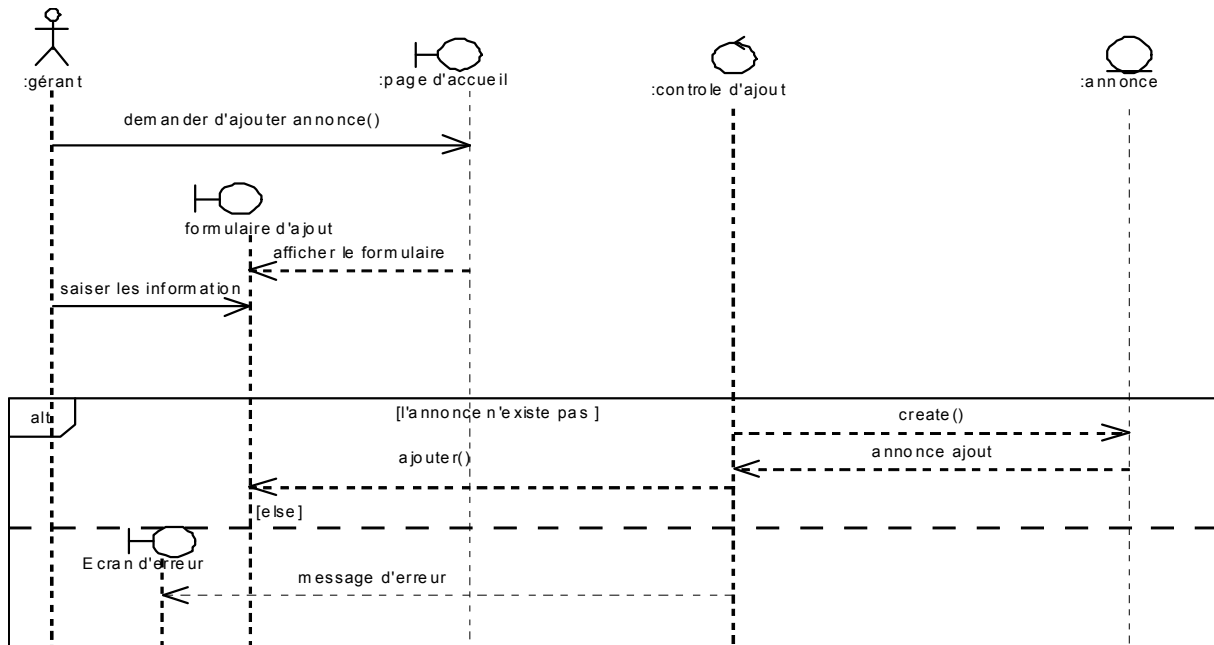


Figure 3.43 : Diagramme d'interactions du cas d'utilisation « ajouter annonce»

### 2.6-Supprimer annonce :

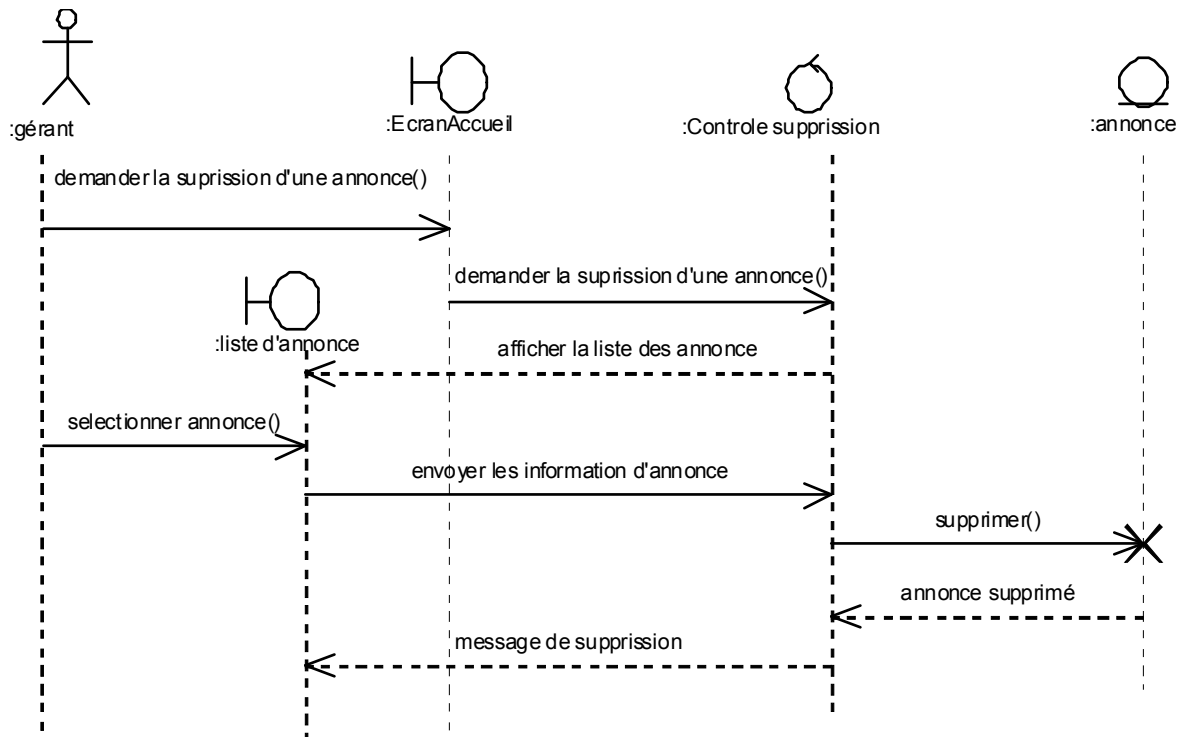


Figure 3.44 : Diagramme d’interactions du cas d’utilisation « supprimer annonce»

### 2.7- Créé un compte candidat :

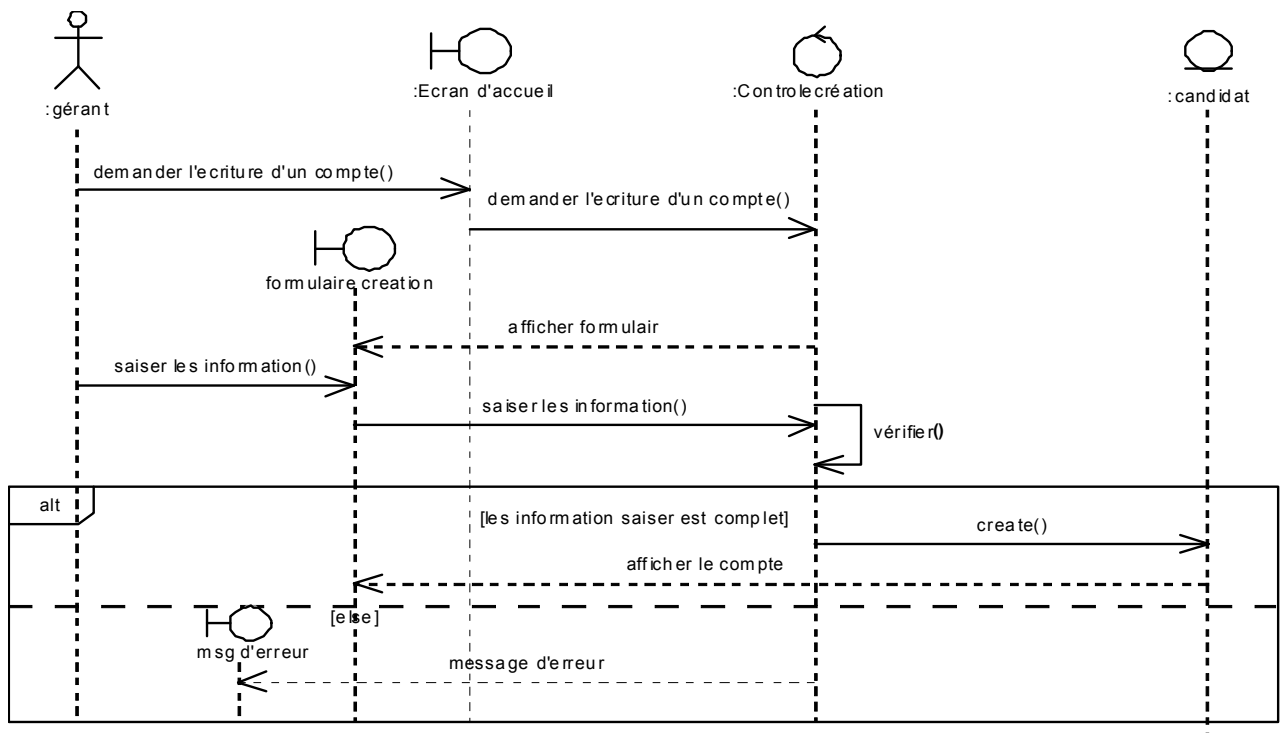


Figure 3.45 : Diagramme d’interactions du cas d’utilisation « créée un compte candidat»

2.8- Supprimer un compte candidat :

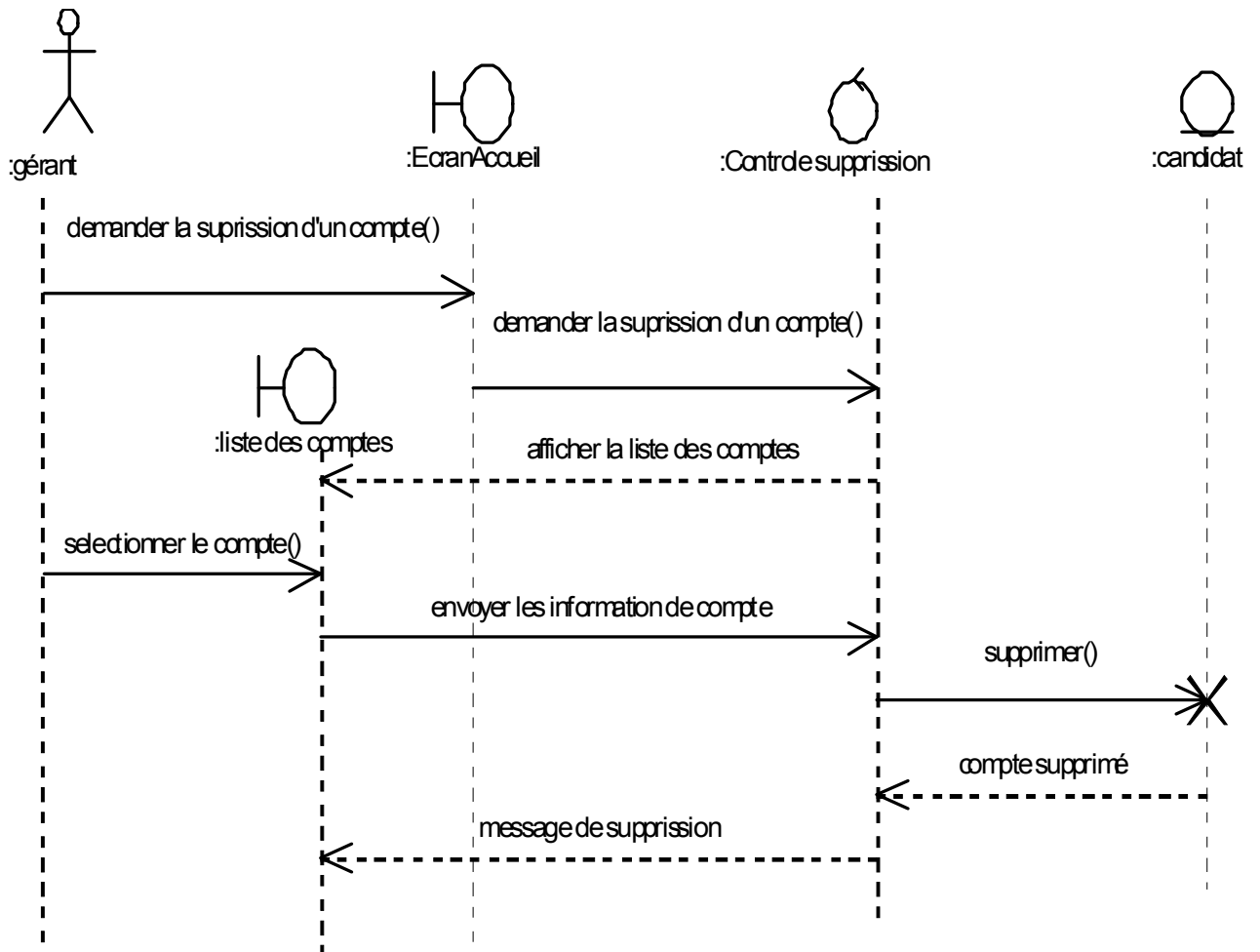


Figure 3.46 : Diagramme d’interactions du cas d’utilisation « supprimer un compte candidat»

2.9-Modifier un compte candidat :

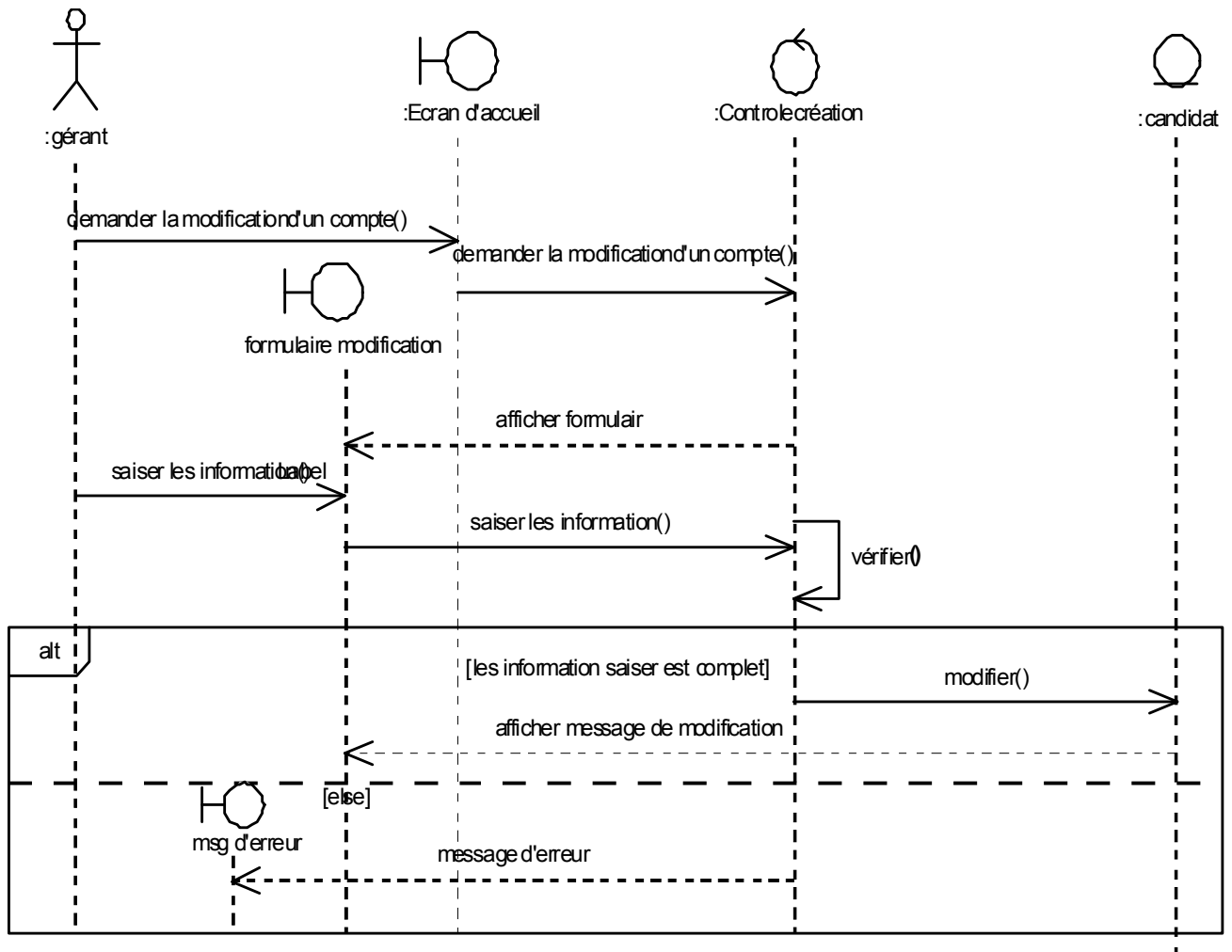


Figure 3.47 : Diagramme d’interactions du cas d’utilisation « modifier un compte candidat »

2.10-Ajouter résultat :

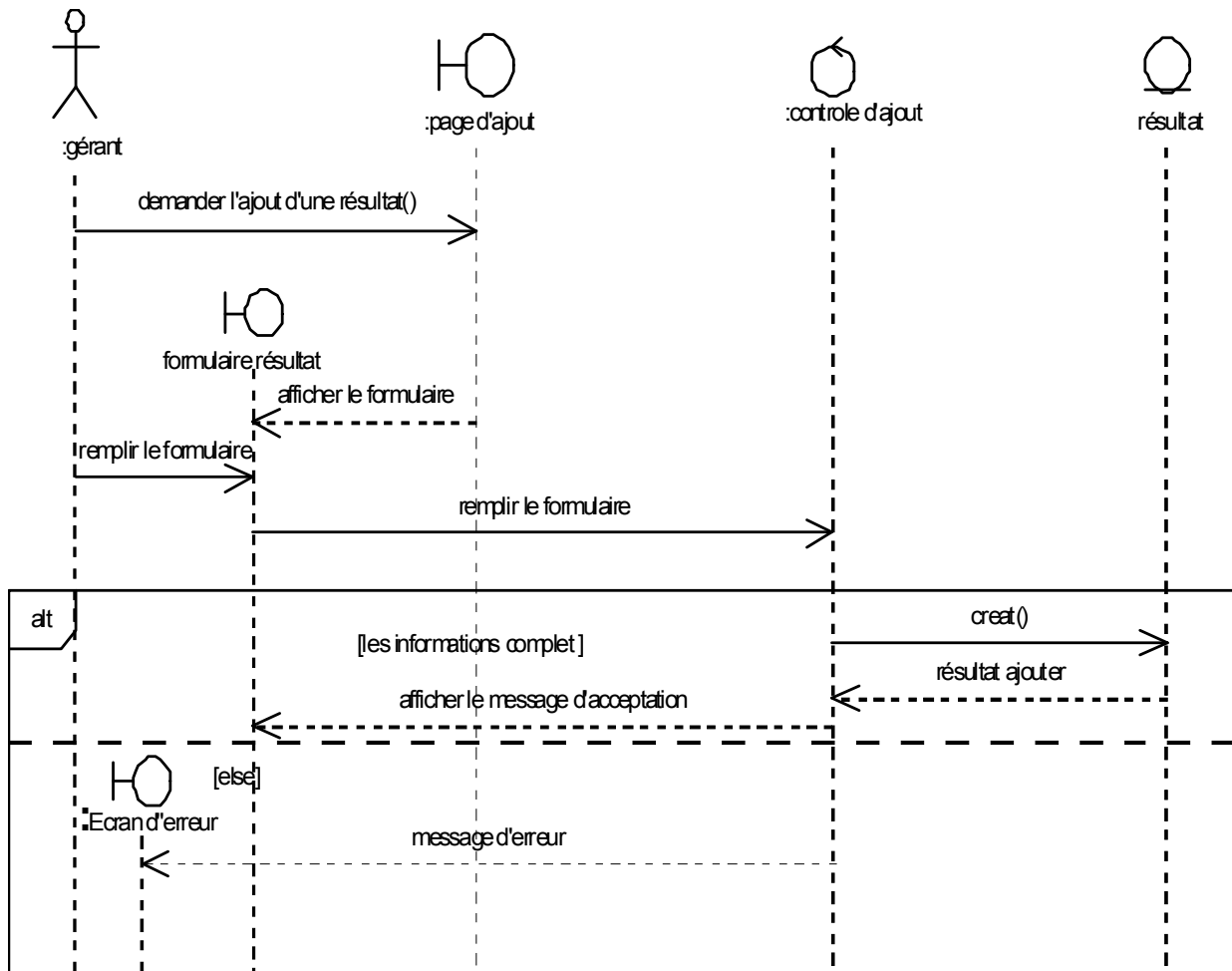


Figure 3.48 : Diagramme d'interactions du cas d'utilisation « ajouter résultat»

2. Diagramme de classes métiers :

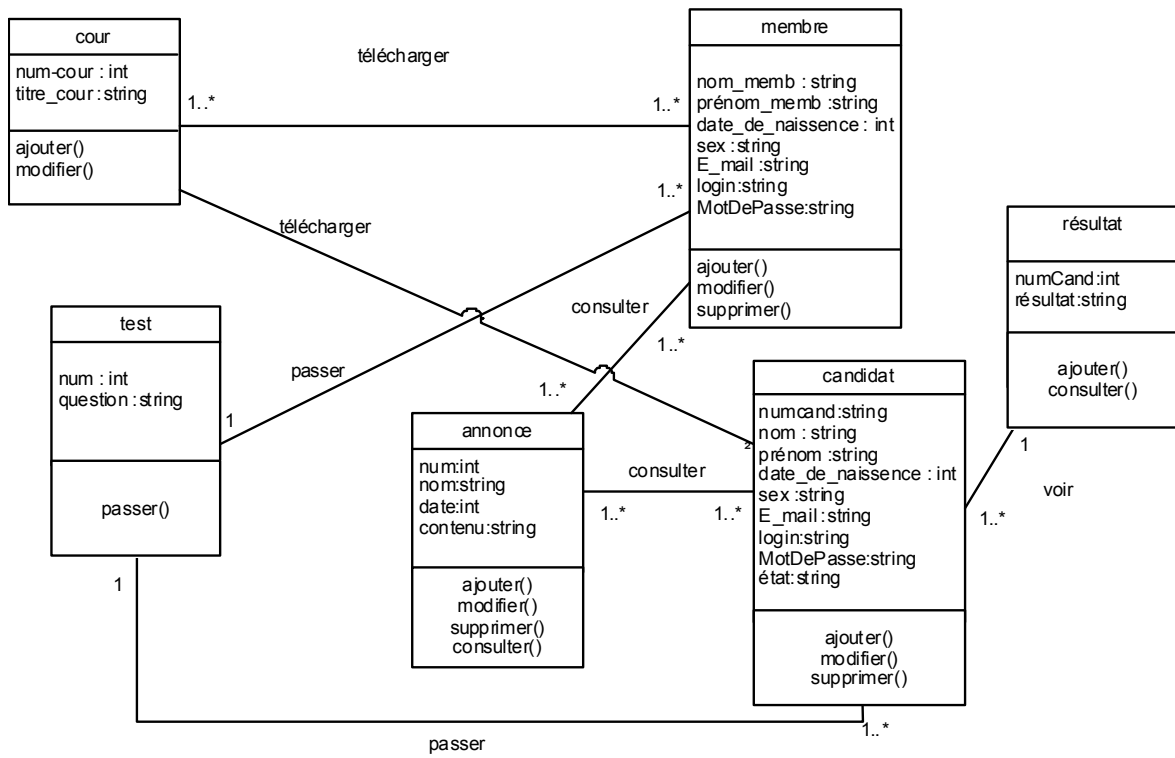


Figure 3-49 : Le diagramme de classes métiers

**Conclusion:**

Pour la réalisation de ce chapitre, nous avons utilisé un langage de modélisation UML pour les applications web, en suivant une démarche de développement logiciel qui est le processus unifié, l'analyse et la conception détaillée nous permettent de bien représenter le système à réaliser ainsi qu'elles nous facilitent la tâche de la programmation. Pour présenter la manière dont notre système est réalisé, vient la dernière étape qui est l'implémentation du site; en utilisant des outils techniques pour la programmation.

Le chapitre suivant, qui va être le dernier, va nous montrer comment le site sera réalisé



# *Chapitre 04 : Implémentation*

## **Introduction :**

La réalisation d'un site web dynamique a besoin de plusieurs logiciels de développements.

Pour cela, nous avons choisi quelques outils et langages parmi ceux existants et qui répondent à nos besoins.

Dans ce chapitre, nous allons citer la liste des outils utilisés pour la réalisation du site tout en expliquant le rôle de chacun. Ensuite, nous allons faire le passage du diagramme de classe- résultant à la fin de l'étape de conception- au modèle relationnel afin de définir la base de données nécessaire pour réaliser notre site.

Enfin, nous allons présenter quelques interfaces du site.

## 1. Outils de travail :

### 1. Les Langages de programmation :

#### 1.1. Le langage PHP (Personnel Home Page) :

Le PHP, Personnel Home Page ou HyperText préprocesseur, est un langage de programmation Web. Le code *PHP* est directement inclus dans les pages Hypertexte, il doit être placé entre les balises « < ? *PHP* » et « ? > ». On appelle ce genre de langage, un langage de script. [4]

PHP permet de créer des pages interactives. Une page interactive permet à un visiteur de saisir des données personnelles. Ces dernières sont ensuite transmises au serveur, où elles peuvent rester stockées dans une base de données pour être diffusées vers d'autres utilisateurs.

Un visiteur peut, par exemple, s'enregistrer et retrouver une page adaptée à ses besoins lors d'une visite ultérieure. Il peut aussi envoyer des e-mails et des fichiers sans avoir à passer par son logiciel de messagerie.

En associant toutes ces caractéristiques, il est possible de créer aussi bien des sites de diffusion et de collecte d'information que des sites d'e-commerce, de rencontres ou des blogs.

PHP est un langage de programmation, très proche du langage C dont il reprend l'essentiel de la syntaxe et destiné à être intégré dans des pages HTML. Contrairement à d'autres langages,

PHP est exclusivement dédié à la production de pages HTML générées dynamiquement.

Les codes du PHP sont appelés « scripts », et ils sont inclus dans le code HTML.

A l'origine du *PHP*, il y a les CGI (Common Gateway Interface) et plus particulièrement des langages tels que le PERL ou C. On trouve dans le *PHP* un grand nombre de similitudes avec ces derniers et plus particulièrement au niveau de la syntaxe. Généralement, Le *PHP* est caractérisé par le fait qu'il est:

- \*  **Un langage polyvalent** : PHP s'inscrit dans le mode des logiciels libres, cela signifie que la communauté des programmeurs qui s'axent autour du *PHP* participe activement au développement de ce langage.

- \*  **un langage Simple** : PHP propose un langage et un modèle de développement très simple. La vocation historique de PHP était de permettre à n'importe que l'informaticien de développer rapidement et sans formation préalable une application web dynamique. Elle s'avère tellement simple que tous les hébergeurs grand public ont retenu PHP.

\* □ **Un langage exécuté coté serveur** : Lorsqu'on effectue une requête sur une page écrite en *PHP*, celle-ci est interprétée et exécutée par le serveur. Le résultat qui apparaît au client est totalement dépourvu de code et celui-ci reste confidentiel.

\* □ **Un langage très simple à assimiler** : La programmation en *PHP* est très libre, pas besoin, par exemple, de déclarer les variables ou redimensionner les tableaux manuellement...etc.

\* □ **Un langage de script** : Le *PHP* n'est pas compilé à chaque fois qu'une requête est effectué sur une page *PHP*, le serveur interprète celui-ci et le traduit en langage machine exécutable.[4]

## 1.2. MYSQL :

Le langage SQL est un langage normalisé de gestion de base de données. MYSQL, le plus populaire des serveurs de bases de données SQL Open Source, est développé, distribué et supporté par MYSQL AB (MYSQL AB est une société commerciale, fondée par les développeurs de MySQL, qui développent leur activité en fournissant des services autour de MySQL).

De nombreux fournisseurs d'accès Internet, et hébergeurs de sites proposent un accès gratuit à une plate-forme PHP/MySQL, ceci soutient le choix de MySQL. D'autres facteurs d'ordre technique existent :

- **Vitesse de traitement** : MySQL se distingue par une vitesse de traitement accrue résultant de la réécriture de routine ISMA de gestion des entrées/sorties, et de l'adoption d'algorithmes d'appariement de tables optimisés et organisés autour de threads.
- **Fiabilité** : Des listes de diffusion active contribuent à la correction rapide d'éventuels bogues de MySQL. De plus, des utilitaires de teste de cohérence automatique existent, comme MYIAMCHK.
- **Compatibilité SQL** : MySQL est compatible avec SQL, il respecte la norme ANSI SQL 92.
- **Sécurité** : L'administrateur d'une base de données contrôle finement les catégories de personnes et les machines autorisées à se connecter, et l'authentification repose surdes mots de passe cryptés. [7]

## 2. Les outils de développement :

Les outils de développements qu'on a utilisés pour développer notre site sont les suivants :

### 2.1. Dreamweaver :

Adobe Dreamweaver est le logiciel leader de la création graphique de sites web depuis quelques années. Mis au point en 1998 par la société américaine Macro media, rachetée en 2004 par Adobe, ce programme est au départ un simple éditeur HTML Wysiwyg.

Il est utilisé pour la création graphique de sites web, et la génération automatique du code HTML à partir d'éléments visuels. La création d'un site basique à l'aide de Dreamweaver ne demande pas plus de connaissances techniques que la rédaction d'une lettre dans un traitement de texte comme Microsoft Word.

Au cours des années, ses fonctionnalités de base ont fortement évolué. On trouve désormais l'ensemble des fonctions indispensables à la création d'un site web moderne.

Depuis sa version 2.0, Dreamweaver permet la génération automatique de programmes JavaScripts qui ajoutent beaucoup d'interactivité. Depuis sa version MX, on peut également créer des sites web dynamiques architecturés autour de langage serveur tels PHP, ASP, JSP ou Cold Fusion 5

#### Avantage :

- Mise en page aisée.
- Publication assistée.
- Intégration des autres produits.



Figure 4.1: Dreamweaver version 8.0

## 2.2. WampServer :

Wamp Server (anciennement WAMP5) est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un serveur externe).

Des scripts PHP. Wamp Server n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que PHPMyAdmin pour l'administration Web des bases MySQL.

Il dispose d'une interface d'administration permettant de gérer et d'administrer ses serveurs au travers d'un trayicon (icône près de l'horloge de Windows) [6].

## 2.3. Adobe Photoshop :

Photoshop est un logiciel de retouche, de traitement et de dessin assisté par ordinateur et édité par Adobe. Il est principalement utilisé pour les traitements de photographies numériques.

Photoshop travaille sur les images matricielles (également appelées "bitmap", à ne pas confondre avec le format d'enregistrement Windows bitmap) car les images sont constituées d'une grille de points appelés pixels. L'intérêt de ces images est de reproduire des graduations.

Photoshop possède son propre format de projet (PSD), qui est plus qu'un simple format de fichier. Le programme accepte également d'importer et d'exporter des fichiers d'image dans les formats les plus courants (GIF, JPEG, TIFF, PNG, ILBM, etc.).



Figure 4.2: Adobe Photoshop version:8.0

## 2.4. PHPMYAdmin :

PhpMyAdmin est une interface conviviale qui permet de gérer très facilement une base de données, sans nécessiter d'une connaissance avancée des requêtes SQL. Elle utilise conjointement une base MySQL et un moteur de scripts PHP. Cette interface peut donc fonctionner directement sur le serveur Web et être accessible par le biais d'un simple navigateur. Elle permette la:

- gestion des bases de données (Créer, copier, supprimer et modifier des tables) ;
- gestion des utilisateurs et leurs droits d'accès ;
- exécution des requêtes SQL.



Figure 4.3 : PHPMYAdminversion 5.3.5.0

### 3. Le passage du diagramme de classe au modèle relationnel :

Nous présentons ci –après quatre règles de (R1 à R4) pour traduire le modèle du domaine en un schéma relationnel équivalent. Il existe d’autre solution de transformation mais ces règles sont les plus simples et les plus opérationnelles : [8]

#### 1) Transformation des entités /classes :

La règle est simple **R1** :

\* Chaque entité devient une relation, identifiant de l’entité devient clé primaire pour la relation.

\* chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant jouer le rôle identifiant.

#### 2) Transformation des associations :

Les règles de transformation que nous allons voir dépendent des cardinalités / multiplicités maximale des associations. Nous distinguons trois familles d’association :

**a. Association 1 vers plusieurs (1…\*):** La règle est la suivante :

**R2 :** Il faut ajouter un attribut de type clé étrangère dans la relation fils de l’association.

L'attribut porte le nom de la clé primaire de la relation père de l'association.

**b. Association plusieurs vers plusieurs (\*...\*) :** La règle et la suivante :

**R3 :** association/classe-association devient une relation. La clé primaire de cette relation est la concaténation des identifiants des entités connecté à l'association. Chaque attribut devient clé étrangère si entité/classe connectée dont il devient une relation en vertu de la règle **R1**.

Les attributs d'association/classe-association doivent être ajoutés à la nouvelle relation. Ces attribut ne sont ni clé primaire, ni clé étrangère.

**c. Association 1 vers 1 (1...1) :** La règle et la suivante :

**R4 :** il faut ajouter un attribut de type clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimale égale à zéro. Dans le cas de diagramme UML il faut ajouter un attribut de type clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un.

L'attribut porte le nom de la clé primaire de la relation dérivée d'entité classe connectée à l'association. Si les deux cardinalités minimales égales à zéro, le choix est donné entre les deux relations dérivées de la R1.

Si les deux cardinalités minimales égales à un, il est préférable de fusionner les deux entités/classe en une seule.

Après application des quatre règles de passage précédentes, nous avons obtenus les tables suivantes :

- membre(id\_me, nom, prénom, date\_naissance, Email, num\_tél, password, sex, pseudo, catégorie, score).
- Candidat(id\_can, nom, prénom, date\_naissance, Email, num\_tél, password, sex, pseudo, type, catégorie, score).
- Annonce (id\_an, nom, contenu, date).
- Livre (code\_livre, nom livre, catégorie, fichier).

## 2. Implémentation de la base de données :

Pour implémenter notre base des données baptisée « auto-école », nous avons utilisé PHPMyAdmin et le système de gestion de base des Données MySQL.

La figure ci-dessous présente un exemple d'une base de données.

The screenshot shows the phpMyAdmin interface for a table named 'livree' in the 'centre' database. The table structure is as follows:

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra	Action
1	code_livre	int(7)			Non	Aucune	AUTO_INCREMENT	Modifier, Supprimer, Primaire, Unique, Index, Spatial, plus
2	nom_livre	varchar(50)	latin1_swedish_ci		Non	Aucune		Modifier, Supprimer, Primaire, Unique, Index, Spatial, plus
3	categorie	varchar(3)	latin1_swedish_ci		Non	Aucune		Modifier, Supprimer, Primaire, Unique, Index, Spatial, plus
4	fichier	varchar(200)	latin1_swedish_ci		Non	Aucune		Modifier, Supprimer, Primaire, Unique, Index, Spatial, plus

Below the table structure, there is an 'Information' section with a table of statistics:

Espace utilisé		Statistiques	
Données	16 Kio	Format	Compact
Index	0 0	Interclassement	latin1_swedish_ci
Total	16 Kio	Prochain index automatique	5
		Création	Mar 13 Mai 2014 à 15:10

The URL at the bottom of the page is: localhost/phpmyadmin/tbl\_structure.php?db=centre&table=livree&token=b0f582d97af2c35b77a8b926a1c358e

Figure 4.4 : Structure de la base de données.

## 2. Organigramme générale du site:

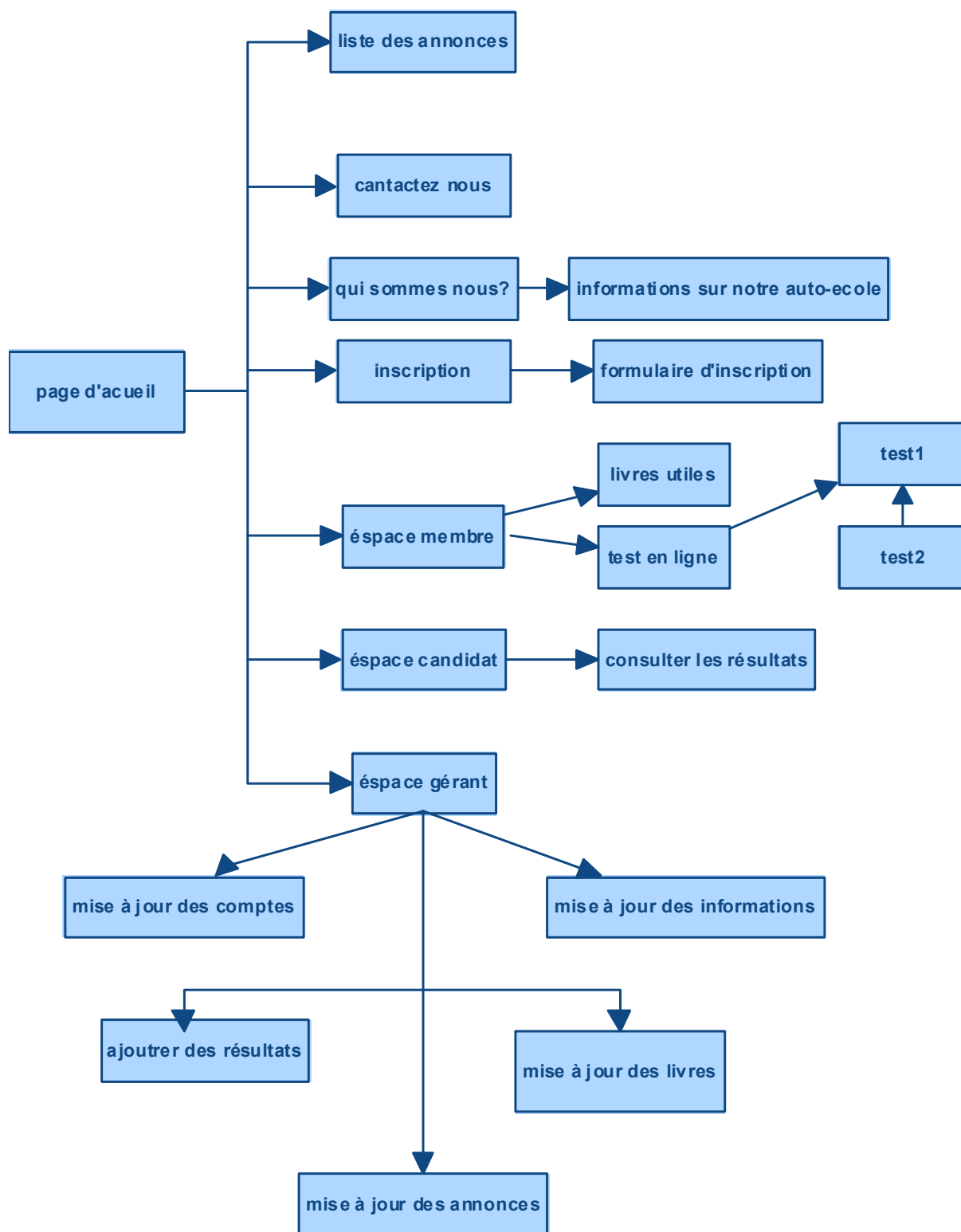


Figure 4.5 : Organigramme du site

### 3. Présentation de quelques pages de l'application :

#### 3.1. La page d'accueil :

La page d'accueil est la première page consultée par un utilisateur. Elle est conçue dans le but de permettre à l'utilisateur d'accéder aux autres pages.



Figure 4.6 : Page d'accueil

### 3.2. Page contactez-nous :

La page qui permet aux utilisateurs de consulter les informations de l'auto-école

**membre**

Bienvenu user1  
Se déconnecter

**SAMI AUTO-ECOLE**

Vos coordonnées

Nom :

Email :

Votre message :

Objet :

Message :

Envoyer le formulaire !

**Coordonnées**

École de conduite SAMI	
Adresse :	Cité 500 logts - mila
Tel:	0661-87-81-02
Fax:	031-57-99-11

**Heures d'ouverture:**

<b>Samedi</b>	12h00 à 17h00
<b>Dimanche</b>	09h00 à 18h00
<b>Lundi</b>	09h00 à 18h00
<b>Mardi</b>	09h00 à 18h00
<b>Mercredi</b>	09h00 à 18h00
<b>Jeudi</b>	09h00 à 18h00
<b>Vendredi</b>	12h00 à 17h00

**Trouvez-nous**

à pied 100 m

à pied 2 min

auto-ecole SAMI

Figure 4.7 : Page contactez-nous

Page d'accueil gérant : le gérant peut accéder à la page après l'authentification

The screenshot displays the manager's dashboard for Sami Auto Ecole. At the top, there is a navigation menu with links for 'Accueil', 'Qui sommes nous?', 'teste en ligne', 'resultat', 'Livres utiles', and 'Contactez nous'. The main content area is dominated by a large promotional banner for a driving program. The banner includes the text 'Maintenant en promotion' and 'Programme complet de la nouvelle loi'. It lists the program components: '24 heures de théorie', '15 heures de pratique', and 'Évaluation et attestations'. A price tag indicates '12000DA prix choc'. A blue car is shown with 'SAMI AUTO-ECOLE' branding and 'Meilleur prix en ville'. Below the banner, there are three smaller sections: 'CONDUITE ACCOMPAGNEE' with a logo, a group of people at a presentation, and 'Horaire des cours et inscriptions'. On the left side, there is a 'membre' section with a login area showing 'Bienvenu user1' and a 'Se déconnecter' button. At the top right, there is an 'Actualités' section with a table containing 'promotion 20%' and 'date concours'.

Figure 4.8 : Page d'accueil gérant

A partir de cette page, l'administrateur peut ajouter des annonces, modifier les comptes, afficher les résultats des examens réels.

**Page ajouter annonces** : à partir de cette page le gérant peut ajouter des annonces.

The screenshot shows the 'Sami Auto Ecole' website interface. At the top, there is a banner with the text 'Sami Auto Ecole' and 'sur la route Conduire demande toute ton attention!' along with illustrations of a man and a woman and traffic signs. Below the banner is a navigation menu with links: 'Accueil', 'Qui sommes nous?', 'teste en ligne', 'resultat', 'Candidat', 'Livres utiles', and 'Contactez nous'. On the right side, there is an 'Actualités' section. The main content area is titled 'SAMI AUTO-ECOLE' and contains a form for adding an announcement. The form has a 'titre' field with the value 'date concours', a 'le contenu' field with the value 'vous allez passé le concours le 15-05-2014 bon courage', and two buttons: 'ajouter' and 'annuler'. On the left side, there is a sidebar with a 'membre' section containing 'Bienvenu Admin' and a 'Se déconnecter' button, and an 'espace administration' section with links for 'ajouter livre', 'ajouter annonce', and 'mis a jour membre'.

**Figure 4.9 : Page ajouter annonces**

**Ajouter les résultats :** On trouve dans cette page l’affichage du résultat de l’examen réel passé par les candidats.

The screenshot shows the SAMI AUTO-ECOLE website interface. At the top, there is a navigation menu with links: Accueil, Qui sommes nous?, teste en ligne, resultat, Candidat, Livres utiles, and Contactez nous. Below the menu is a banner with a cartoon illustration of a man and a woman, the text "sur la route Conduire demande toute ton attention!", and traffic signs. To the right of the banner is an "Actualités" section with a list of items: inscription2014, promotion 20%, and date concours.

The main content area is titled "SAMI AUTO-ECOLE" and contains several sections:

- membre**: A sidebar with "Bienvenu Admin" and a "Se déconnecter" button.
- espace administration**: A sidebar with links: "ajouter livre", "ajouter annonce", and "mis a jour membre".
- liste des candidats admis en code**: A table with columns: nom, prenom, type, and an action column (X and edit icon).
- liste des candidats admis en creneau**: A table with columns: nom, prenom, type, and an action column (X and edit icon).
- liste des candidats admis en circuit**: A table with columns: nom, prenom, type, and an action column (X and edit icon).

nom	prenom	type	
ahmed	ali	admis en code	X ✎

nom	prenom	type	
youcef	dermouci	admis en creneau	X ✎

nom	prenom	type	
charaf	graiche	admis en circuit	X ✎
zahia	bouaroudj	admis en circuit	X ✎

**Figure 4.10 :** Page ajouter résultats

**Mise à jour des comptes:** Cette page permet au gérant de modifier, supprimer et ajouter un compte candidat ou membre.

Accueil | Qui sommes nous? | teste en ligne | resultat | Candidat | Livres utiles | Contactez nous

**membre**

Bienvenu Admin

[Se déconnecter](#)

**SAMI AUTO-ECOLE**

code	pseudo	nom	prenom	email	type	
3	Admin	aaaaaaa	ddd	latifcbm@live.fr	Administrateur	
17	charaf	charaf	graiche	ya7ia43@live.fr	admis en circuit	
18	zahoo	zahia	bouaroudj	farida.1989@live.fr	admis en circuit	
14	qqqlll	ahmed	ali	ya7ia43@live.fr	admis en code	
6	zzz	youcef	dermouci	zaki_cnep@live.fr	admis en creneau	
5	Adminmmm	benabderrahmane	eeee	farida.1989@live.fr	circuit	
8	aaa	aqw	zakaria	zaki_cnep@live.fr	circuit	
16	user1	abd	sami	sfdg@yahoo.fr	creneau	
15	yahyae	ddd	graiche	ali_algerian@live.fr	membre	

**espace administration**

- [ajouter livre](#)
- [ajouter annonce](#)
- [mis a jour membre](#)

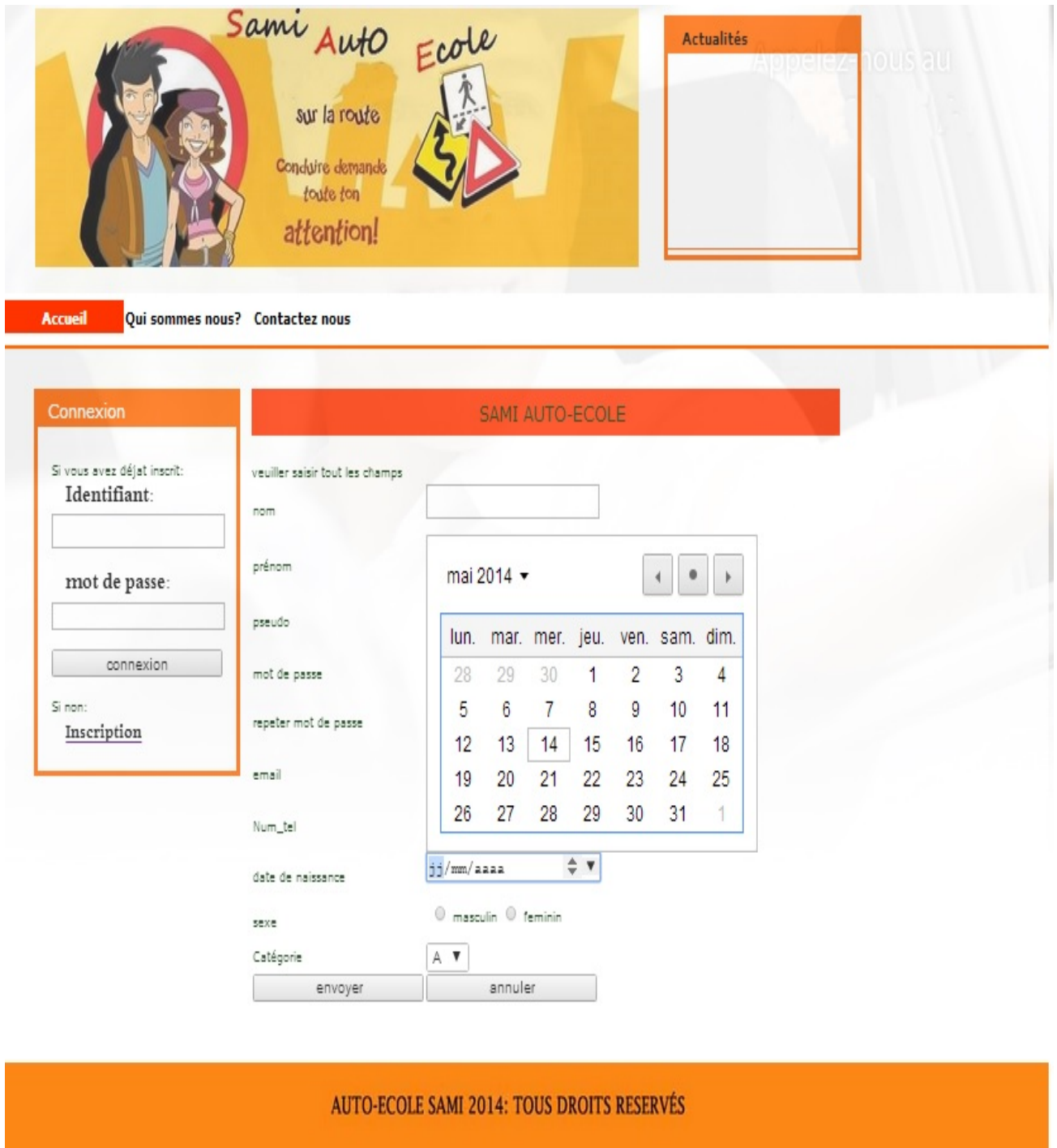
Figure 4.11 : Page Mise à jour des comptes

**Page d'accueil visiteur :** Le visiteur est une personne inconnue par le site. Il peut accéder au site sans condition, pour consulter les informations de l'auto-école et les annonces. Il peut aussi s'inscrire au site.



**Figure 4.12 :** Page d'accueil visiteur

**Page inscription :** Elle permet au visiteur de s’inscrire dans le site.



**Figure 4.13 :** Page d’inscription

Après inscription le visiteur devient un membre.

**Page d'accueil membre :** Après authentification, la page permet à un membre de télécharger des livres ou des cours et de passer un test.

The screenshot displays the member homepage for Sami Auto Ecole. At the top, there is a navigation menu with links for 'Accueil', 'Qui sommes nous?', 'teste en ligne', 'resultat', 'Livres utiles', and 'Contactez nous'. A prominent banner advertises a 'Maintenant en promotion' program, detailing '24 heures de théorie', '15 heures de pratique', and 'Évaluation et attestations'. A blue car is featured with a '12000DA prix choc' badge and 'Meilleur prix en ville' text. On the left, a member sidebar shows a login area with 'Bienvenu yah' and a 'Se déconnecter' button. The page also includes an 'Actualités' section, a 'CONDUITE ACCOMPAGNEE' logo, and a 'Horaire des cours et inscriptions' link.

**Figure 4.14 :** Page d'accueil membre

Page télécharger livre :



Figure 4.15 : Page télécharger livre

Page passer test : Permet à un membre de passer deux types de test :



Figure 4.16 : Page passer test

**Passé test1** : La page permet à un membre de passer un test de code. Le système propose des images et des questions accompagnées de plusieurs réponses. Le membre va sélectionner à chaque fois une réponse. A la fin, le système lui affiche le résultat sous forme d'un pourcentage. Le test change à chaque fois d'une façon aléatoire.

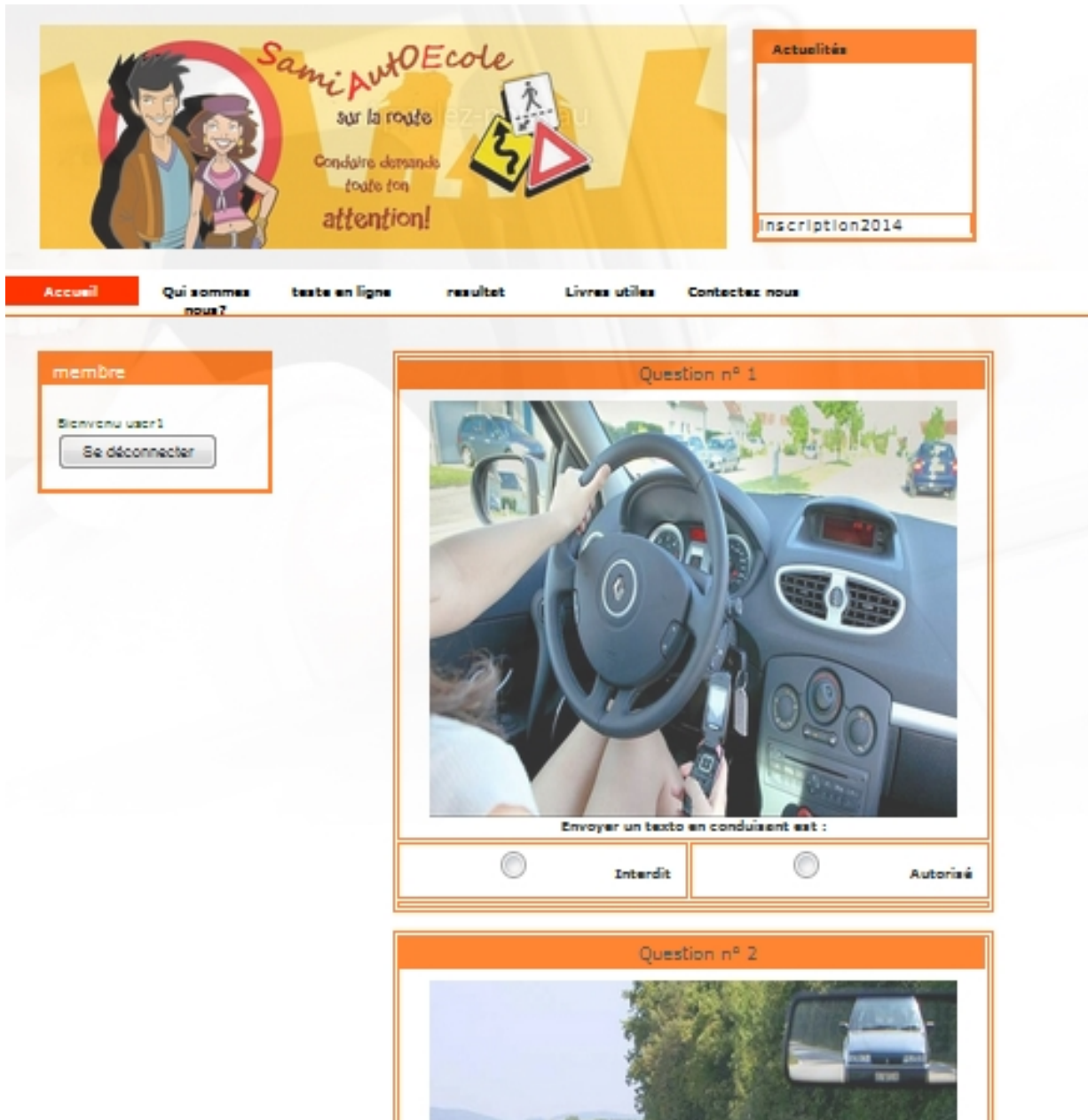


Figure 4.17 : Page passer test1

Le membre ne peut passer le test numéro 2 que si le résultat de test1 est  $\geq 4$



Page Passé test2 :

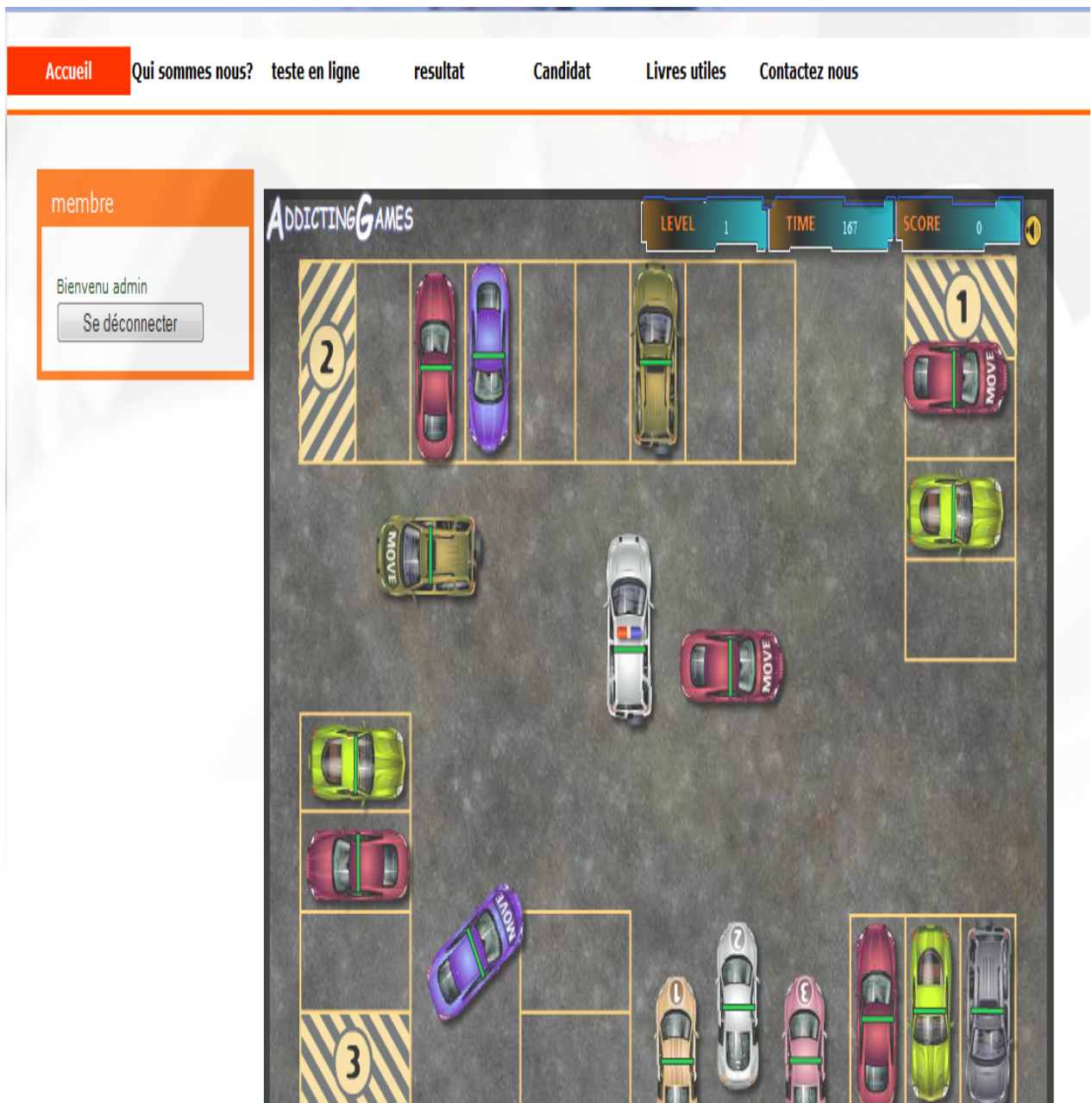


Figure 4.18 : Page passer test2

**Page d'accueil candidat :** Après authentification, la page permet à un candidat de voir le résultat de l'examen réel ainsi que les pages qu'un membre peut accéder.

The screenshot shows the 'Sami Auto Ecole' website interface for a candidate. At the top, there is a navigation bar with links: **Accueil**, **Qui sommes nous?**, **teste en ligne**, **resultat**, **Livres utiles**, and **Contactez nous**. The main content area is dominated by a large promotional banner for a driving course. The banner includes the text 'Maintenant en promotion' and 'Programme complet de la nouvelle loi'. It lists the course components: '24 heures de théorie', '15 heures de pratique', and 'Évaluation et attestations'. A prominent orange starburst graphic displays '12000DA prix choc'. Below this, it notes '(Possibilité de versements sur 3 étapes)' and 'Le prix de test 2000DA+2000DA de timbre'. To the right of the text is an image of a blue car with 'SAMI AUTO-ECOLE' and 'Meilleur prix en ville' branding. A small '1 2 3' indicator is visible at the bottom right of the car image. On the left side of the page, there is a sidebar with a 'membre' section containing a login form with the text 'Bienvenu user1' and a 'Se déconnecter' button. The top left of the page features a graphic with a man and a woman, the text 'Sami Auto Ecole', and the slogan 'sur la route Conduire demande toute ton attention!' along with traffic signs. A 'Actualités' section on the top right lists 'promotion 20%' and 'date concours'. A faint 'Appellez-nous' watermark is visible in the background.

**Figure 4.19 :** Page d'accueil candidat

**Page résultats :** Elle contient le résultat de l'examen réel.

**SamiAutoEcole**  
sur la route  
Conduire demande toute ton attention!

**Actualités**  
promotion 20%  
date concours

**Accueil** | **Qui sommes nous?** | **teste en ligne** | **resultat** | **Livres utiles** | **Contactez nous**

**membre**  
Bienvenu user1

**SAMI AUTO-ECOLE**

liste des candidats admis en code

nom	prenom	type	

liste des candidats admis en creneau

nom	prenom	type	

liste des candidats admis en circuit

nom	prenom	type	
charaf	graiche	admis en circuit	

**Figure 4.20 :** Page résultats.

## **Conclusion :**

Ce chapitre est la phase finale dans notre travail. Il comprend une description générale des langages et des outils utilisés pour le développement du site, ainsi que le passage du diagramme de classe vers le modèle relationnel. Par ailleurs, il comporte les interfaces qui déterminent de façon plus claire les activités réalisées dans ce site.

Finalement, on peut dire que notre application répond à l'objectif souhaité dès le départ.



# *Conclusion générale*

## **Conclusion générale :**

Notre projet consiste à concevoir et réaliser un site web dynamique pour une auto-école.

Nous avons réussi à développer un site web qui offre les services suivants :

- \* Inscription en ligne.
- \* Téléchargement des cours et des livres.
- \* Passage des tests gratuits virtuellement.

Pour réaliser ce site web , nous avons utilisé UML, le processus de développement UP et d'autres outils de développement comme : **Adobe Dreamweaver CC, WAMP et Adobe Photoshop.**

Ce projet nous a permis d'acquérir de nouvelles connaissances à savoir **UML, UP** et les langages **HTML, PHP, CSS** et **SQL**.

Le site développé est entièrement opérationnel concernant les fonctionnalités citées dans ce mémoire.

A la fin de notre réalisation, nous avons constaté que le site peut être amélioré par d'autres scénarios concernant le test niveau deux qui nécessite plus de temps et de professionnalisme.

Et nous souhaitons que des étudiants des prochaines promotions aient le plaisir de se charger à développer cette application.

## *Références bibliographique*

- [1]. [EYROLLES] UML2 Modeliser une application web  
4edition
- [2] .[http:// www.lecompagnon.info/internet/](http://www.lecompagnon.info/internet/)
- [3] .Pascal Roques,Franck Vallée , UML 2 en action: De  
l'analyse des besoins à la conception.
- [4] .Philippe Rigaux, Pratique de MySQL et PHP
- [5] .[www.mti.epita.fr/blogs/wp.content/uploads/dreameaver-  
cs3.pdf](http://www.mti.epita.fr/blogs/wp.content/uploads/dreameaver-cs3.pdf)
- [6] .[www. . wikipedia.org/WAMP](http://www.wikipedia.org/WAMP)
- [7] .l'essentiel pour concevoir un site web dynamique
- [8] .Laurent AUDIBERT UML 2 Édition 2007-2008