

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

CENTRE UNIVERSITAIRE DE MILA
INSTITUT DES SCIENCES ET DE LA TECHNOLOGIE

Réf. /12

Mémoire de fin d'étude
Présenté pour l'obtention du diplôme de

Licence Académique

Domaine : **Mathématiques et Informatique**
Filière : **Informatique**

Thème

**Conception et réalisation d'un
site web dynamique pour
la gestion et le suivi
des journées scientifiques
au centre universitaire
de Mila**

Présenté par :

- 1- Bouanane Imane
- 2- Mezzahi Sara

Dirigé par :

- Merabet Adil

Année universitaire 2012-2013

Dédicaces

Je dédie ce mémoire :

A mes très chers parents, pour leur soutien, leurs sacrifices, leurs encouragements qui m'ont permis d'arriver

Au bout de ce travail,

A ma mère «Souad » qui m'a éclairée mon chemin et qui m'a encouragé et soutenue toute au long de mes

Études, je prie dieu de vous protège et vous guérit, a Mon père « Abdelaziz » qui s'est sacrifié afin que

Rien n'en déroulement de mes études, a ma tante « Ibtissame » pour son soutien aux moments difficiles,

A ma chère sœur : Farida,

A mon encadreur Mr Merabet Adil,

A mon binôme Sara,

À mes très chers amis : Bouchra, Zina, Siham, Hachem, Haroun, Issam , Lamia , Nora , Hadjer, Manel

A Tous les membres de ma famille, à mes profs et mes collègues de la promotion informatique 2012/2013

A tous ceux et celles qui me sont chers.

IMANE

Je dédie ce mémoire : A mes très chers parents, qui m'ont fourni au quotidien un soutien et une confiance

Sans faille et de ce fait, je ne saurais exprimer ma gratitude seulement par des mots. Que dieu vous protège et

Vous garde pour nous,

A celle qui m'a transmis la vie, l'amour, le courage, à toi chère maman toutes mes joies, et ma reconnaissance,

A mon frère : Amir,

A ma chère sœur : Farida,

A mon encadreur Mr Merabet Adil,

A mon binôme Imane,

À mes très chers amis : Randa ,Imane bousseta, Imane ,Abla, Zina, Siham , haroun et Issam, A Tous les Membres de

Ma Famille, à mes profs et mes collègues de la promotion informatique 2012/2013.

A tous ceux et celles qui me sont chers.

SARA

Remerciements

Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail.

Nous exprimons toute notre gratitude à Monsieur le Professeur Merabet Adil, pour l'effort fourni, les conseils prodigués, sa patience et sa persévérance dans le suivi de notre travail.

Nous adressons également nos remerciements, à tous nos enseignants tout au long du cycle d'étude nous remercions très sincèrement toute personne, à toute personne qui a participé de près ou de loin pour l'accomplissement de ce modeste travail.

SARA & IMANE

Sommaire

Résumé	13
Introduction générale	14
Chapitre I : Technologie web	
Introduction	17
I.1- Web : Les concepts fondamentaux	17
I.1.1- World Wide Web	17
I.1.2- Page web	18
I.1.3- Le site Web	18
I.1.4- Quelle est la différence entre une page web et un site Web	18
I.1.5- L'application Web	18
I.1.6- Navigateur Web	18
I.2- La topologie des sites Web	19
I.2.1- Le site statique	19
I.2.2- Le site dynamique	19
I.2.3- Comment fonctionne un site Statique	19
I.2.4- Comment fonctionne un site Dynamique	20
I.2.5- Quels avantages pour un site Statique	20
I.2.6- Quels avantages pour un site Dynamique	21
I.2.7- Quels types de site pour quelle utilisation	21
I.3- L'architecture client/serveur	22
I.3.1- Définition	22
I.3.2- Client	23
I.3.3- Serveur	23
I.3.4- Requête	23
I.3.5- La réponse	23
I.4- Les avantages de l'architecture Client/Serveur	23
I.5- Les inconvénients de l'architecture Client/Serveur	24
I.6- Le fonctionnement d'un système Client-Serveur	25
I.7- Les différents types de Client/Serveur	25
I.7.1- Client / Serveur de représentation	25
I.7.2- Client Serveur de données	25
I.7.3- Client / Serveur de procédure	25
I.8- Les niveaux de l'architecture Client/ Serveur	25
I.8.1- L'architecture à deux niveaux	25

I.8.2- L'architecture à trois niveaux	26
I.8.3- L'architecture à plusieurs niveaux	27
Conclusion.....	28

Chapitre II : UML et Processus Unifié UP

Introduction	30
II.1-UML	30
II.1.1-Les bases d'UML	30
II.1.2-UML le langage de modélisation unifié	30
II.1.3-UML est un langage graphique	32
II.1.4-UML est un support de communication	31
II.1.5-UML n'est pas une méthode.....	32
II.1.6-UML est basé sur un Meta modèle	32
II.1.7-UML est basé sur un Meta modèle	32
II.1.8-UML est une norme	32
II.1.9-UML est une unification des méthodes	32
II.10-Diagrammes d'UML	32
II.10.a-Diagrammes structurels ou diagrammes statiques (UML Structure).....	33
II.10.b-Diagrammes comportementaux ou diagrammes dynamiques	33
II.2-Processus unifié (UP)	38
II.2.1-Définition.....	38
II.2.2-Catégories	38
II.2.2.a-UP est itératif	38
II.2.2.b-UP est centré sur l'architecture	38
II.2.2.c-UP est piloté par les cas d'utilisation d'UML	39
II.2.3-Les phases.....	39
II.2.3.a-Analyse des besoins	39
II.2.3.b-Élaboration.....	39
II.2.3.c-Construction.....	40
II.2.3.d-Transition.....	40
II.2.4-Une méthode simple et générique	40
II.2.4.1-Identification des besoins et spécification des fonctionnalités	40
II.2.4.1.a-diagramme de cas d'utilisation	40
II.2.4.1.b-diagramme de séquence système	40
II.2.4.1.c-Maquette IHM.....	41

II.2.4.2-Phase d'analyse.....	41
II.2.4.2.a-diagramme de domaine	41
II.2.4.2.b-diagramme des classes participantes.....	42
II.2.4.2.c-diagramme d'activité de navigation	42
II.2.4.3-Phase de conception.....	43
II.2.4.3.a-diagramme d'interaction	43
II.2.4.3.b-Diagramme de classe de conception.....	43
Conclusion.....	44

Chapitre III : Identification des besoins

Introduction	46
III.1-Identification des acteurs	46
III.2-Diagramme de cas d'utilisation.....	47
III.3-Les fiches descriptives des cas d'utilisations.....	49
III.3.1-Fiche descriptive: cas d'utilisation «S'authentifier».....	49
III.3.2-Fiche descriptive: cas d'utilisations «Gérer les thèmes »	49
III.3.3-Fiche descriptive: cas d'utilisations «Gérer les demandes des participants ».....	50
III.3.4-Fiche descriptive: cas d'utilisation «Gérer les membres».....	50
III.3.5-Fiche descriptive: cas d'utilisation «Consulter la liste des journées scientifiques ».	50
III.3.6-Fiche descriptive: cas d'utilisation « Créer le compte »	51
III.3.7-Fiche descriptive: cas d'utilisation « Consulter les listes des participants».....	51
III.3.8-Fiche descriptive: cas d'utilisation « Lancer une demande de participation»	51
III.3.9-Fiche descriptive: cas d'utilisation « Gérer le compte»	52
III.3.10-Fiche descriptive : cas d'utilisation « Gérer les listes des participants».....	52
III.3.11-Fiche descriptive du cas d'utilisation « Gérer la participation.....	52
III.4-Les diagrammes de séquence système	53
III.4.1-S'authentifier.....	53
III.4.2- Gérer les thèmes.....	54
III.4.3- Gérer les listes des participations	56
III.4.4-Gérer les membres.....	57
III.4.5- Créer le compte	58
III.4.6-Lancer une demande de participation.....	59
III.4.7-Gérer la participation.....	60
III.5-Les digrammes d'activité de navigation	61
III.5.1- Créer compte	61

III.5.2-Gérer les listes des participations Gérer les thèmes	62
III.5.3-Gérer les comptes	63
III.5.4-Demande de participation	64
III.5.5-Gérer les membres.....	65
III.5.6-Diagramme de classe	67
Conclusion.....	67

Chapitre IV : Implémentation de site

Introduction	69
IV.1-Les langages utilisés	69
IV.1.1- HTML	69
IV.1.2- CSS	69
IV.1.3- PHP	69
IV.1.4-MySQL	70
IV.1.5-Conception de base données	70
IV.2-Les logiciels utilisés.....	70
IV.2.1-WampServer	70
IV.2.2-Dreamweaver	70
IV.3-Description des tables	70
IV.3.1- Table de membre (users).....	71
IV.3.2- Table de journée.....	71
IV.3.3- Table de participation	71
IV.3.4- Table de jurée.....	71
IV.4- Interface de notre application	72
IV.4.1- Page d'accueil	72
IV.4.2-Connexion	72
IV.4.3- Page d'inscription	73
IV.4.4- Contact	73
IV.5- Les opérations différentes.....	74
IV.5.1- Administrateur	74
IV.5.1- Page d'accueil d'administrateur.....	74
IV.5.2- Page de gestion des participations.....	74
IV.5.3- Page pour affectée des jurées	75
IV.5.2- Membre	75
IV.5.2.1- Page de participation	75

IV.5.3- Jurée	76
IV.5.3.1- Page d'accueil de jurée.....	76
IV.5.3.2- Page de gestion des participations.....	76
Conclusion.....	.77
Conclusion générale78
Bibliographie.....	.79

Liste des figures

Figure 1 : site statique	19
Figure 2 : fonctionnement d'un site dynamique.....	20
Figure 3 : Serveur.....	23
Figure 4 : Historique d'UML.....	31
Figure 5 : Exemple simplifié de diagramme de cas d'utilisation modélisant Une borne d'accès une banque.....	34
Figure 6 : Représentation UML d'une classe.....	34
Figure 7 : Exemple de diagramme de classes et de diagramme d'objets.....	35
Figure 8 : Exemple d'état composite modélisant l'association d'une commande à un client.....	35
Figure 9 : Exemple de diagramme d'activité illustrant l'utilisation de nœuds de contrôle. Ce diagramme décrit la prise en compte d'une commande.....	36
Figure 10 : Exemple Diagramme de communication d'un système de pilotage...	36
Figure 11 : Exemple Diagramme de séquence d'un système de pilotage.....	37
Figure 12 : Exemple de diagramme montrant les dépendances entre composants.....	37
Figure 13 : Exemple de diagramme de déploiement illustrant la communication entre plusieurs nœuds.....	37
Figure 14 : l'itération dans l'UP.....	38
Figure 15 : les différentes perspectives qui définissent un modèle d'architecture.....	38
Figure 16 : Diagramme de cas d'utilisation	40
Figure 17 : Les diagrammes de séquence illustrent la description textuelle Des cas d'utilisation.....	40
Figure 18 : Les besoins donnent lieu à des cas d'utilisation et à une maquette...	41
Figure 19 : La phase d'analyse du domaine permet d'élaborer la première version Du diagramme de classes.....	41
Figure 20 : Le diagramme de classes participantes effectue la entre lescas d'utilisation, le modèle du domaine et les diagrammes de conception logicielle.....	42
Figure 21 : Les diagrammes d'activités de navigation représentent graphiquement l'activité de navigation dans l'IHM.....	42

Figure 22 : Les diagrammes d'interaction permettent d'attribuer précisément les responsabilités de comportement aux classes d'analyse.....	43
Figure 23 : Chaîne complète de la démarche de modélisation du besoin jusqu'au code.....	43
Figure 24.1 : diagramme de cas d'utilisation.....	47
Figure 24.2 : diagramme de cas d'utilisation.....	48
Figure 25 : S'authentifier.....	53
Figure 26.1 : Gérer les thèmes.....	54
Figure 26.2 : Gérer les thèmes.....	55
Figure 27 : Gérer les listes des participations.....	56
Figure 28 : Gérer les membres.....	58
Figure 29 : Créer le compte.....	58
Figure 30 : Lancer une demande de participation.....	59
Figure 31 : Gérer la participation.....	60
Figure 32 :Créer compte.....	61
Figure 33 :Gérer les listes des participations.....	62
Figure 34 :Gérer les thèmes.....	63
Figure 35 :Gérer les comptes.....	64
Figure 36 : Demande de participation.....	65
Figure 37 : Gérer les membres.....	66
Figure 38 :Diagramme de classe.....	67
Figure 39 : Page d'accueil.....	72
Figure 40 : Connexion.....	72
Figure 41 : Page d'inscription.....	73
Figure 42 : contact.....	73
Figure 43 :Page d'accueil d'administrateur.....	74
Figure 44 : Page de gestion des participations.....	74
Figure 45 : Page pour affectée des jurées.....	75
Figure 46 :Page de participation.....	75
Figure 47 :Page d'accueil de jurée.....	76
Figure 48 :Page de gestion des participations.....	76

Liste des Tableaux

Tableau 1 : S'authentifier.....	49
Tableau 2 : Gérer les thèmes	49
Tableau 3 : Gérer les demandes des participants	50
Tableau 4 : Gérer les membres	50
Tableau 5 : Consulter la liste des journées scientifiques	50
Tableau 6 : Créer le compte	51
Tableau 7 : Consulter les listes des participants.....	52
Tableau 8 : Lancer une demande de participation.....	52
Tableau 9 : Gérer le compte.....	53
Tableau 10 : Gérer les listes des participants.....	53
Tableau 11 :Gérer la participation	53
Tableau 12 :Table de membre.....	71
Tableau 13 :Table de journée.....	71
Tableau 14 :Table de participation.....	71
Tableau 15 :Table de jurée.....	71

Liste des Schémas

Schéma 1 : Architecture à deux niveaux.....	26
Schéma 2 : architecture 3 niveaux.....	27
Schéma 3 : architecture plusieurs niveaux.....	28

Résumé

Une journée scientifique réunit des chercheurs, des enseignants, des étudiants universitaires (baccalauréat, maîtrise et doctorat) et même des professionnels hors de l'université. Dans notre projet nous allons réaliser un site web dynamique pour la Gestion et le suivi des journées scientifiques en ligne en utilisant un outil de Modélisation UML et un ensemble de langages de Programmations tels que l' HTML et le PHP...

Introduction générale

Le Web est le service qui a placé l'Internet sous les feux de projecteur. Il est un support bien adapté pour la coopération sur un réseau à grande distance pour permettre par exemple à plusieurs personnes de publier et maintenir un ensemble de documents consultable sur un serveur.

L'Internet offre des possibilités considérables et accroît les chances de communications à travers un lieu virtuel qui regroupe des pages Internet présentées par des personnes ou des organisations, c'est le site web.

Une journée scientifique réunit des chercheurs, des enseignants, des étudiants universitaires (baccalauréat, maîtrise et doctorat) et même des professionnels hors de l'université. Dans notre projet nous allons réaliser un site web dynamique pour la Gestion et le suivi des journées scientifiques en ligne.

Le but est de : Faciliter les échanges entre la Recherche et l'enseignement, Contribuer à la diffusion des résultats et des pistes actuelles de recherche, Renforcer la culture scientifique des enseignants et étudiants, Favoriser les collaborations en matière d'expérimentation... etc.

En utilisant un outil de Modélisation UML « langage de modélisation unifié », et des langages de Programmations tels que l' HTML et le PHP... ce projet est divisé en quatre chapitres :

Le premier chapitre est dédié à présenter des concepts jugés nécessaires sur le principe, l'architecture, les avantages et les inconvénients du modèle client-serveur, ainsi que le Web.

Le deuxième chapitre s'intéresse au langage UML et ses diagrammes, en plus, le processus unifié (UP) qui est un processus de développement logiciel définit une séquence d'étapes.

Le troisième chapitre est consacré à notre conception, par la définition des différents acteurs et les cas d'utilisations, et puis la proposition des différents diagrammes de cas d'utilisations, les fiches descriptives, les diagrammes de séquences et les digrammes d'activité

Introduction générale

Enfin, Le quatrième chapitre présente les outils de développement que nous avons utilisé tels que : PHP, MySQL, HTML,et aussi le résultat de notre implémentation résumé dans quelques interfaces de notre site web de gestion des journées scientifiques.

Chapitre I

Technologie web

Introduction

S'il y a est un phénomène qui a marqué le monde de l'informatique ces dernières années, c'est bien celui d'Internet. Ce réseau mondial crée en 1969 par l'armée américaine puis utilisé par les chercheurs et les scientifiques, a connu une croissance phénoménale auprès du grand public avec l'introduction du World Wide Web en 1989.

Le web, c'est l'application Client/Serveur la plus largement déployée au monde. Cette dernière permet de publier simplement des informations riches, des mises en forme et pouvant même, par la suite contenir des documents multimédia.

Cependant il est nécessaire d'avoir un minimum de notions sur la technologie Internet, l'architecture client/serveur et le Web.

1. Web : Les concepts fondamentaux.

1.1. World Wide Web



Le World Wide Web (ou WWW, ou Web) est un très grand système d'information réparti sur un ensemble de sites connectés par le réseau internet. Ce système est essentiellement constitué de documents hypertextes, ce terme pouvant être pris au sens large : textes, images, sons, vidéos, etc. Chaque site propose un ensemble plus ou moins important de documents qui se transmettent sur le réseau par l'intermédiaire d'un programme serveur. Ce programme serveur dialogue avec un programme client qui peut être situé n'importe où sur le réseau. Le programme client prend le plus souvent

la forme d'un navigateur. Il existe deux types de site Web, qui sont les listes statiques et dynamiques.

1.2. page web

Une page Web est un document du Web, un élément du réseau qui forme le Web. C'est un hypertexte qui contient des liens vers d'autres pages Web ou ressources.

1.3. Le site Web

Un site Web est une collection de pages Web et d'éventuelles autres ressources liées dans une structure cohérente. Un site Web est hébergé sur un ordinateur appelé le serveur Web.

1.4. Quelle est la différence entre une page web et un site Web

Une page Web est un fichier, comme celui-ci qui contient du texte, des images et des liens à d'autres pages. Un site Web est un regroupement de pages sur un sujet, un thème, un commerce, une organisation. Un site Web a aussi une page principale. C'est une page Web qui aide les lecteurs à naviguer sur le site pour trouver l'information voulue.

1.5. L'application Web

En informatique, une application Web est un logiciel applicatif manipulable grâce à un navigateur Web (liens, cliqués, formulaires, cookies...).

De la même manière que les sites Web, une application Web est généralement placée sur un serveur (base de données SQL, fichiers de configuration...). Et se manipule en actionnant des widgets à l'aide d'un navigateur Web, via un réseau informatique (Internet, intranet, réseau local, etc.)[1]

Quelques exemples sur les applications:

Les systèmes de gestion de contenu, les wikis, les blogs, les moteurs de recherches, les logiciels de commerce électronique, les jeux en ligne, les logiciels de forum et peut rajouter les appareils réseau tels par exemple les routeurs.

1.6. Navigateur Web

Un navigateur web est un logiciel informatique qui permet d'utiliser le web. Pour être plus précis, ce type de logiciel permet de consulter le *World Wide Web* (WWW).

L'utilisation la plus répandue de ces logiciels étant de visualiser les pages web et d'utiliser les liens hypertextes dans le but d'aller de pages en pages.

2. La topologie des sites Web

2.1. Le site statique

Ce sont des sites réalisés uniquement à l'aide des langages HTML et CSS. Ils fonctionnent très bien mais leur contenu ne peut pas être mis à jour automatiquement : il faut que le propriétaire du site (le webmaster) modifie le code source pour y ajouter des nouveautés. Ce n'est pas très pratique quand on doit mettre à jour le site plusieurs fois par jour! Les sites statiques sont donc bien adaptés pour réaliser des sites « vitrine » pour présenter par exemple son entreprise, mais sans aller plus loin. Ce type de site se fait de plus en plus rare aujourd'hui, car dès que l'on rajoute un élément d'interaction (comme un formulaire de contact), on ne parle plus de site statique mais de site dynamique.

2.2. Le site dynamique

Plus complexes, ils utilisent d'autres langages en plus de HTML et CSS, tels que PHP et MySQL. Le contenu de ces sites web est dit « dynamique » parce qu'il peut se changer sans l'intervention du webmaster ! La plupart des sites web que vous visitez aujourd'hui, sont des sites dynamiques. Le seul prérequis pour apprendre à créer ce type de sites est de déjà savoir réaliser des sites statiques en HTML et CSS.

2.3. Comment fonctionne un site Statique

Lorsque le site est statique, le schéma est très simple. Cela se passe en deux étapes comme vous le schématise la figure 1 :

- le client demande au serveur à voir une page web,
- le serveur lui répond en lui envoyant la page réclamée.

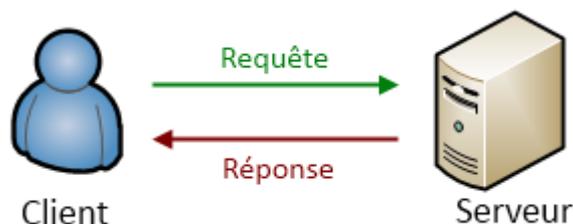


Figure 1 : site statique [2]

La communication est donc plutôt basique :

- « Bonjour, je suis le client, je voudrais voir cette page web. »
- « Tiens, voilà la page que tu m'as demandée. »

Sur un site statique, il ne se passe rien d'autre. Le serveur stocke des pages web et les envoie aux clients qui les demandent sans les modifier.

2.4. Comment fonctionne un site Dynamique

Lorsque le site est dynamique, il y a une étape intermédiaire : la page est générée (figure 2).

- Le client demande au serveur à voir une page web,
- le serveur prépare la page spécialement pour le client,
- le serveur envoie au client la page qu'il vient de générer.

La page web est générée à chaque fois qu'un client la réclame. C'est précisément ce qui rend les sites dynamiques vivants : le contenu d'une même page peut changer d'un instant à l'autre. C'est comme cela que certains sites parviennent à afficher par exemple votre pseudonyme sur toutes les pages.



Figure 2 : fonctionnement d'un site dynamique [2]

Étant donné que le serveur génère une page à chaque fois qu'on lui en demande une, il peut la personnaliser en fonction des goûts et des préférences du visiteur (et afficher, entre autres, son pseudonyme).

2.5. Quels avantages pour un site Statique

On vient de voir qu'un site statique possède beaucoup d'inconvénient : il faut s'y connaître en HTML pour le modifier et l'étape de mise à jour est fastidieuse. Mais il faut aussi reconnaître au site statique des avantages dans plusieurs domaines :

- le site internet est mis à jour en local sur la machine de l'administrateur : il n'y a donc pas de surprise une fois que le site est en ligne.
- le site internet ne fait pas appel aux technologies en perpétuelles évolutions qui permettent la mise en place de sites dynamique (PHP, Ruby, Python, Perl, Java, ASP, etc.) : on gagne donc en sécurité et en veille technologique.
- le site internet statique consomme peu de ressource serveur : le site n'utilisant aucune technologie compliquée (par exemple : PHP + MySQL + Apache), les coûts d'entretien et de maintenance en activité sont très inférieurs à ceux d'un site dynamique.
- le site internet statique se sauvegarde plus facilement : ceux qui ont déjà manipulé les bases de données MySQL utilisées pour la création de sites dynamiques savent que c'est une galère à sauvegarder et à restaurer. Le fait de disposer directement des pages HTML du site facilite la sauvegarde (un simple copier / coller sur une clé USB est c'est bon !).

2.6. Quels avantages pour un 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) qui va bien, 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.

2.7. Quels types de site pour quelle utilisation

On utilisera un site web statique pour une utilisation bien particulière, et ce fonctionnement sera pour un site web nécessitant peu de maintenance, peu de mise à jour et contenant peu de pages.

En effet, comme la mise à jour d'un site Internet statique peut être fastidieuse, on utilisera ce type de site uniquement si les mises à jour sont exceptionnelles. Car, à chaque mise à jour il faudra modifier la page HTML du site et la mettre en ligne en effectuant une copie par FTP.

On utilisera plutôt un site dynamique si on souhaite créer une interaction avec les lecteurs. Le site dynamique permet de se connecter en ligne sur son site pour réaliser sa mise à jour en direct. Une fois la mise à jour du site dynamique effectuée, le résultat apparaît directement aux lecteurs. On privilégiera donc les sites web dynamiques pour les sites permettant aux visiteurs de laisser des commentaires (blogs) ou de converser avec d'autres lecteurs (forums). De même si le site doit être mis à jour très fréquemment (plusieurs fois par semaine) on pourra choisir de créer un site dynamique même si aucune interaction n'est prévue avec les visiteurs.

Par exemple pour réaliser un site vitrine présentant les 10 produits vendus par une entreprise on pourra choisir :

- *Un site statique* si les produits ne sont modifiés qu'une ou deux fois par mois et qu'un ou deux nouveaux produits sont ajoutés au catalogue chaque année.
- *Un site dynamique* si on souhaite que les visiteurs ajoutent des commentaires sur les fiches produits et/ou qu'il faut modifier très souvent (ajout, suppression, modification) les fiches produits.

3. L'architecture : client/serveur

3.1. Définition

L'architecture client-serveur est une architecture de réseau dans laquelle les traitements sont répartis entre les clients qui demandent aux serveurs les informations dont ils ont besoin.

C'est une architecture qui désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs postes clients du serveur.

3.2. Client

Ce sont les ordinateurs des internautes comme vous. Votre ordinateur fait donc partie de la catégorie des clients. Chaque client représente un visiteur d'un site web.

3.3. Serveur

Ce sont des ordinateurs puissants qui stockent et délivrent des sites web aux internautes, c'est-à-dire aux clients. La plupart des internautes n'ont jamais vu un serveur de leur vie. Pourtant, les serveurs sont indispensables au bon fonctionnement du Web.

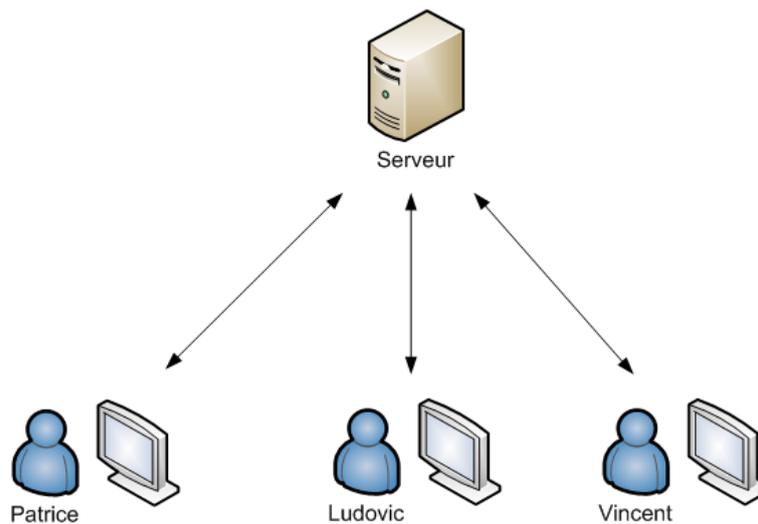


Figure 3 : Serveur

3.4. Requête

C'est un message envoyé par le client au serveur décrivant l'opération à exécuter.

3.5. La réponse

C'est le message envoyé par le serveur à un client suite à l'exécution d'une requête.

4. Les avantages de l'architecteur Client/Serveur

- Elle permet d'accepter une combinaison hétérogène des machines clientes. Dans ces environnements actuels, les utilisateurs sont amenés à employer divers systèmes comme Windows NT, Mac ou une station de travail Unix (Redhat par exemple) qui peuvent tous être intégrés par un même serveur.

- Dans le cadre d'un système utilisant un serveur de BD dédié, les responsabilités de traitement sont réparties entre le client et le serveur, le client étant chargé du contrôle de l'interface utilisateur, de l'affichage des informations et de la validation des entrées utilisateurs et de la délivrance des résultats
- Elle est indépendante du protocole de réseau, du système d'exploitation de serveur et du système d'exploitation du client. Cette indépendance permet de bénéficier d'une certaine souplesse dans le choix des composants matériels et logiciels lors de l'implémentation d'un nouveau système ou de l'installation d'une application client/serveur sur un système existant.
- des ressources centralisées : étant donné que le serveur est au centre du réseau, il peut gérer des ressources communes à tous les utilisateurs, comme par exemple une base de données centralisée, afin d'éviter les problèmes de redondance et de contradiction.
- une meilleure sécurité : car le nombre de points d'entrée permettant l'accès aux données est moins important. [3]

5. Les inconvénients de l'architecteur Client/Serveur

- Si trop de clients veulent communiquer avec le serveur au même moment, ce dernier risque de ne pas supporter la charge (alors que les réseaux pair-à-pair fonctionnent mieux en ajoutant de nouveaux participants).
- Si le serveur n'est plus disponible, plus aucun des clients ne fonctionne (le réseau pair-à-pair continue à fonctionner, même si plusieurs participants quittent le réseau).
- Les coûts de mise en place et de maintenance peuvent être élevés.
- En aucun cas les clients ne peuvent communiquer entre eux, entraînant une asymétrie de l'information au profit des serveurs. [3]

6. Le fonctionnement d'un system Client-Serveur

L'architecture client-serveur fonctionne selon le schéma suivant :

- Le client émet une requête vers le serveur grâce à son adresse IP et le port, qui désigne un service particulier du serveur.
- Le serveur reçoit la demande et répond à l'aide de l'adresse IP de la machine cliente et son port.

7. Les différents types de Client/Serveur

7.1. Client / Serveur de représentation

Ce type d'architecture permet d'assurer une meilleure qualité du dialogue homme machine.

7.2. Client Serveur de donné

C'est l'architecture la plus répandue. Exemple : un PC accède à des données partagées gérées par un serveur SQL. Type d'architecture dans lequel un programme d'application, contrôlé par une interface de présentation sur une machine cliente, accède à des données sur une machine serveur par des requêtes. Cette architecture est qualifiée de première génération.

7.3. Client / Serveur de procédure

C'est une évolution de l'architecture précédente. La base de données intègre des procédures stockées : procédures applicatives recevant des paramètres d'entrée et retournant des paramètres de sortie. [4]

8. Les niveaux de l'architecture Client/ Serveur

8.1. L'architecture à deux niveaux

Les différentes applications dont nous avons parlé précédemment étaient basées sur une architecture à deux niveaux. Une partie des traitements était effectués sur le poste client et la seconde sur le calculateur.

On peut séparer ce type d'application en deux types:

- orienté clients
- orienté serveur.

Dans les applications orienté clients, la plupart des traitements sont effectués sur le poste du client qui possède une Interface Homme Machine dite "lourde". Dans celles orientés serveurs, c'est le serveur qui effectue tous les traitements. Le client, possède une IHM légère.

Qu'elle soit légère ou lourde l'IHM est propriétaire, et nécessite donc une installation sur le poste client. Ceci rend l'installation d'une telle application très lourde si le nombre de clients est important. De plus, on peut très difficilement réutiliser ce type d'application car l'IHM est dédié et composée d'un seul bloc. Enfin les performances réseaux ne sont pas très élevées car de nombreuses requêtes doivent transiter par le réseau.

On peut représenter cette architecture par le schéma ci-dessous.

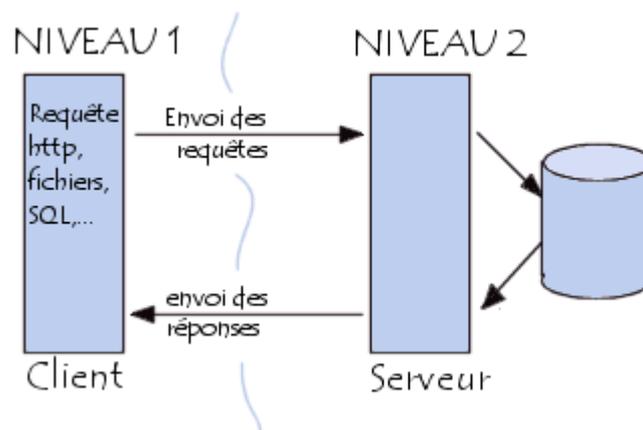


Schéma 1 : Architecture à deux niveaux

8.2. L'architecture à trois niveaux

Dans cette architecture « 3-tier en anglais », aussi nommée « trois tiers » en français, un niveau supplémentaire est ajouté :

- Un client (l'ordinateur demandeur de ressources) équipé d'une interface utilisateur (généralement un navigateur web) chargée de la présentation.

- Un serveur d'application (appelé middleware) qui fournit la ressource, mais en faisant appel à un autre serveur.
- Un serveur de données qui fournit au serveur d'application les données requises pour répondre au client.

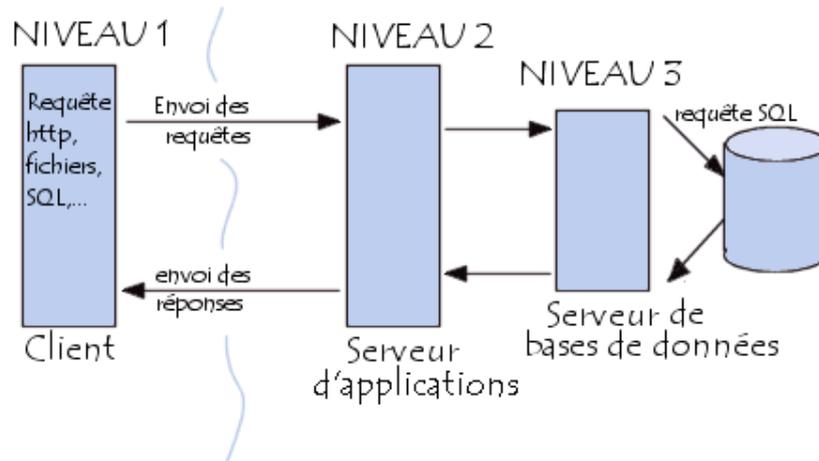


Schéma 2 : architecture 3 niveaux

8.3. L'architecture à plusieurs niveaux

On parle d'architecture à 3 niveaux mais aussi d'architecture plusieurs niveaux. En effet, dans la plupart des applications le niveau intermédiaire est une collection de composants qui sont utilisés dans de nombreux traitements transactionnels. Ces composants peuvent être situés sur un ou plusieurs serveurs physiques. De plus chacun de ces composants effectue une petite tâche et c'est pourquoi on peut séparer cette partie intermédiaire en n parties, d'où le terme architecture à plusieurs niveaux.

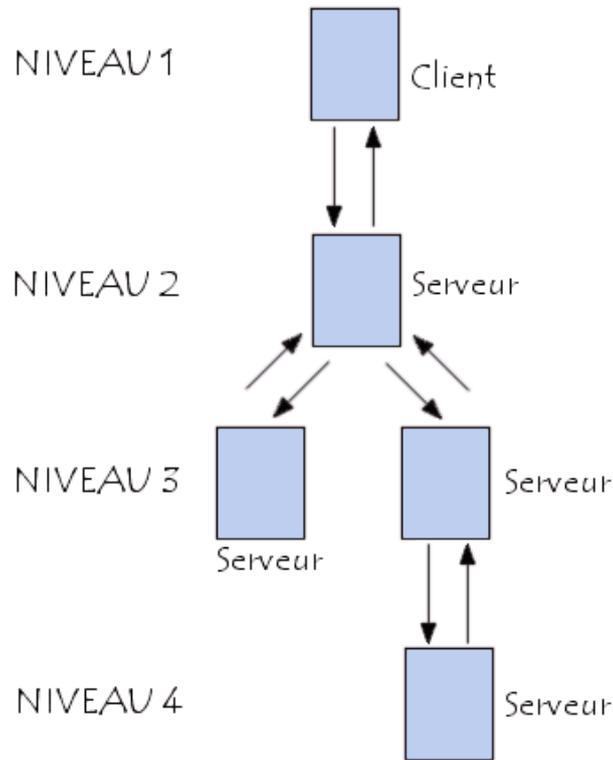


Schéma 3 : architecture plusieurs niveaux

Conclusion

Le Web est en effet un réseau de vigilance, alimenté par les associations, et les gouvernements, accessible à tous les citoyens (au moins dans les pays les plus développés), et qui peut en outre servir de source d'information pour les médias. Et dans ce chapitre, nous avons présenté le mécanisme de transport de l'information sur Internet et la différence entre un site statique et un site dynamique. Ainsi nous allons voir plus en détail dans les chapitres suivants, la présentation des concepts de base du langage de modélisation UML. Ce qui permet une bonne lecture de ce qui va suivre et exprimer de manière uniforme l'analyse et la conception de notre site.

Chapitre II

UML et Processus Unifié UP

Introduction

Dans ce chapitre on propose une présentation générale d'UML, d'UP et ses objectifs et domaines d'application. Principalement ce chapitre vise à reprendre aux questions suivantes :

- Qu'est-ce que c'est UML ?
- Qu'est-ce que c'est UP ?

1. UML

1.1. Introduction

La description de la programmation objet a fait ressortir l'étendue du travail conceptuel nécessaire : définition des classes, de leurs relations, des attributs et méthodes, des interfaces etc. Pour programmer une application, il ne convient pas de se lancer tête baissée dans l'écriture du code : il faut d'abord organiser les idées, les documenter, puis organiser la réalisation en définissant les modules et étapes de la réalisation. C'est cette démarche antérieure à l'écriture que l'on appelle modélisation, son produit est un modèle. L'approche objet permet en principe à la maîtrise d'ouvrage, de s'exprimer de façon précise selon un vocabulaire qui, tout en transcrivant les besoins du métier, pourra être immédiatement compris par les informaticiens.

1.2. Les bases d'UML :

UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue.

UML unifie à la fois les notations et les concepts orientés objet (voir l'historique d'UML sur La figure 1). Il ne s'agit pas d'une simple notation graphique, car les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage.[5]

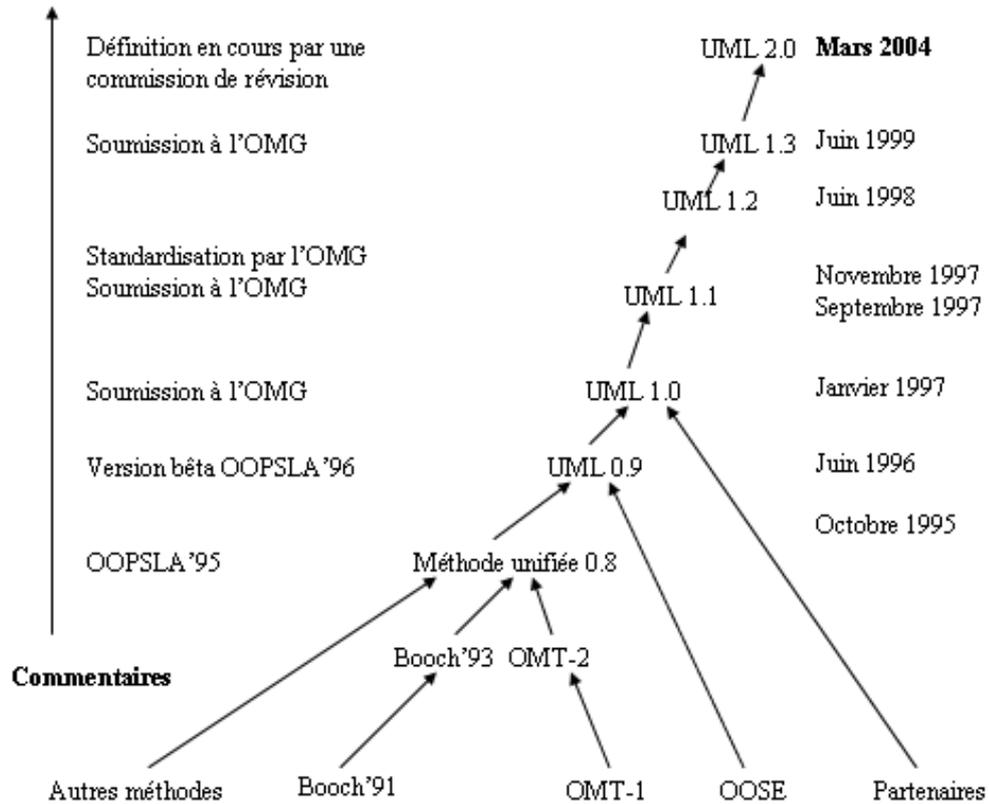


Figure 4 : Historique d'UML[5]

1.3. UML le langage de modélisation unifié :

C'est la forme contractée du langage unifié pour la modélisation. Il fournit les fondements pour spécifier, construire, visualiser et décrire les éléments d'un modèle. Il se base sur une sémantique précise et sur une notation graphique expressive. C'est un moyen d'exprimer des modèles en faisant abstraction de leur implémentation, c'est à dire le modèle fourni par UML est valable pour n'importe quel langage de programmation.

1.4. UML est un langage graphique :

C'est un ensemble de graphisme très complet. Sa notation graphique permet d'exprimer visuellement une solution du problème. L'aspect formel de sa notation limite les ambiguïtés et les incompréhensions. Son aspect visuel facilite la comparaison et l'évaluation des solutions.

1.5. UML est un support de communication :

Il facilite l'expression et la communication de modèles en fournissant un ensemble de symboles (la notation) et de règles qui régissent l'assemblage de ces symboles (la syntaxe et la sémantique).

1.6. UML n'est pas une méthode :

UML n'est pas une méthode ou un processus, UML est développé pour permettre la modalisation des systèmes d'une manière standard et pas pour être une méthode de conception ou d'analyse, pour ça il lui manque la démarche.

1.7. UML est basé sur un Meta modèle :

Un modèle de plus haut niveau qui définit les éléments d'UML (les concepts utilisables) et leur mode d'utilisation (leur signification et leur mode d'utilisation).

1.8. UML est une norme :

UML est devenu une référence en matière de langages de description des systèmes et d'aide à la conception.

1.9. UML est une unification des méthodes :

UML est né de la fusion de trois méthodes :

- OMT de James Rembauche (General Electric) fournit une représentation graphique des aspects statique, dynamique et fonctionnel d'un système.
- OOD de Grade Booth, définie pour le département of défense, introduit le concept de paquetage (package).
- OOSE d'Invar Jacobson (Ericsson) fonde l'analyse sur la description des besoins des Utilisateurs (cas d'utilisation, ou use cases). [5]

1.10. Diagrammes d'UML:

UML 2.0 comporte ainsi 13 types de diagrammes représentant autant de vues distinctes pour représenter des concepts particuliers du système d'information. Ils se répartissent en deux grands groupes :

1.10.1. Diagrammes structurels ou diagrammes statiques (UML Structure)

- Diagramme de classes.
- Diagramme d'objets.
- Diagramme de composants.
- Diagramme de déploiement.
- Diagramme de paquetages.
- Diagramme de structures composites.

1.10.2. Diagrammes comportementaux ou diagrammes dynamiques

- Diagramme de cas d'utilisation.
- Diagramme d'activités.
- Diagramme d'états-transitions.
- Diagrammes d'interaction.
- Diagramme de séquence.
- Diagramme de communication.
- Diagramme global d'interaction.
- Diagramme de temps.

Les diagrammes les plus utilisés sont les diagrammes d'activités, de cas d'utilisation, de classes, d'objets, de séquence et d'états transitions qui ils permettent de formaliser les contraintes de la réalisation et la solution technique.

a. 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. [6]

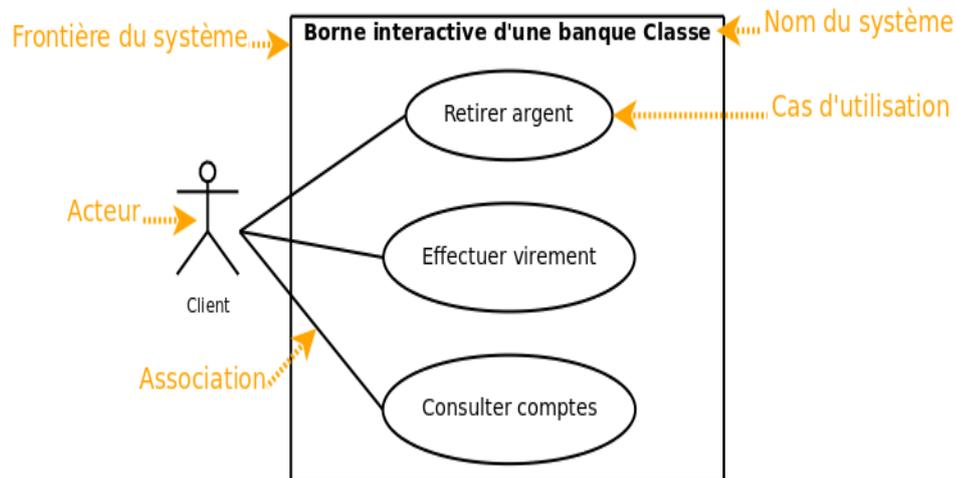


Figure 5 : Exemple simplifié de diagramme de cas d'utilisation modélisant une borne d'accès à une banque.[6]

b. Diagramme de classes

Le diagramme de classes est généralement considéré comme le plus important dans un développement orienté objet. Il représente l'architecture conceptuelle du système : il décrit les classes que le système utilise, ainsi que leurs liens, que ceux-ci représentent un emboîtement conceptuel (héritage) ou une relation organique (agrégation).[6]

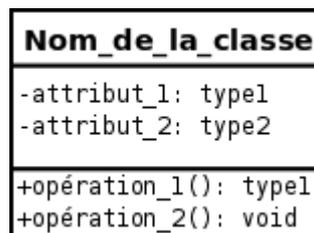


Figure 6 : Représentation UML d'une classe. [6]

c. Diagramme d'objets

Le diagramme d'objets permet d'éclairer un diagramme de classes en l'illustrant par des exemples. Il est par exemple, utilisé pour vérifier l'adéquation d'un diagramme de classes à différents cas possibles.

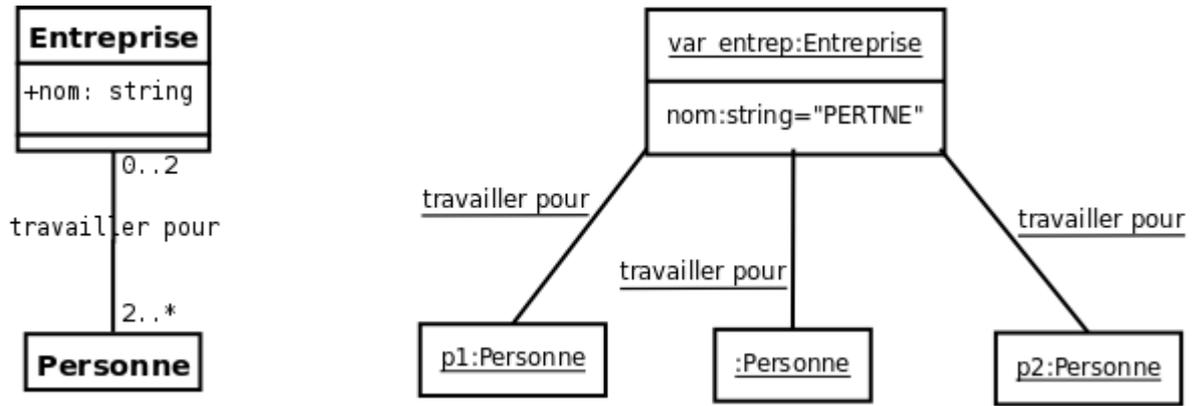


Figure 7 : Exemple de diagramme de classes et de diagramme d’objets.[6]

d. Le diagramme d’états-transitions

Représente la façon dont évoluent (i.e. cycle de vie) les objets appartenant à une même classe. La modélisation du cycle de vie est essentielle pour représenter et mettre en forme la dynamique du système.

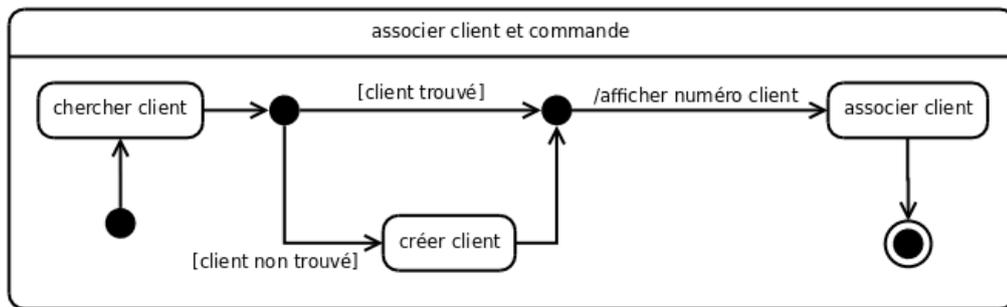


Figure 8 : Exemple d’état composite modélisant l’association d’une commande à un client.[6]

e. Diagramme d’activités

Le diagramme d’activités n’est autre que la transcription dans UML de la représentation du processus tel qu’elle a été élaborée lors du travail qui a préparé la modélisation : il montre l’enchaînement des activités qui concourent au processus.

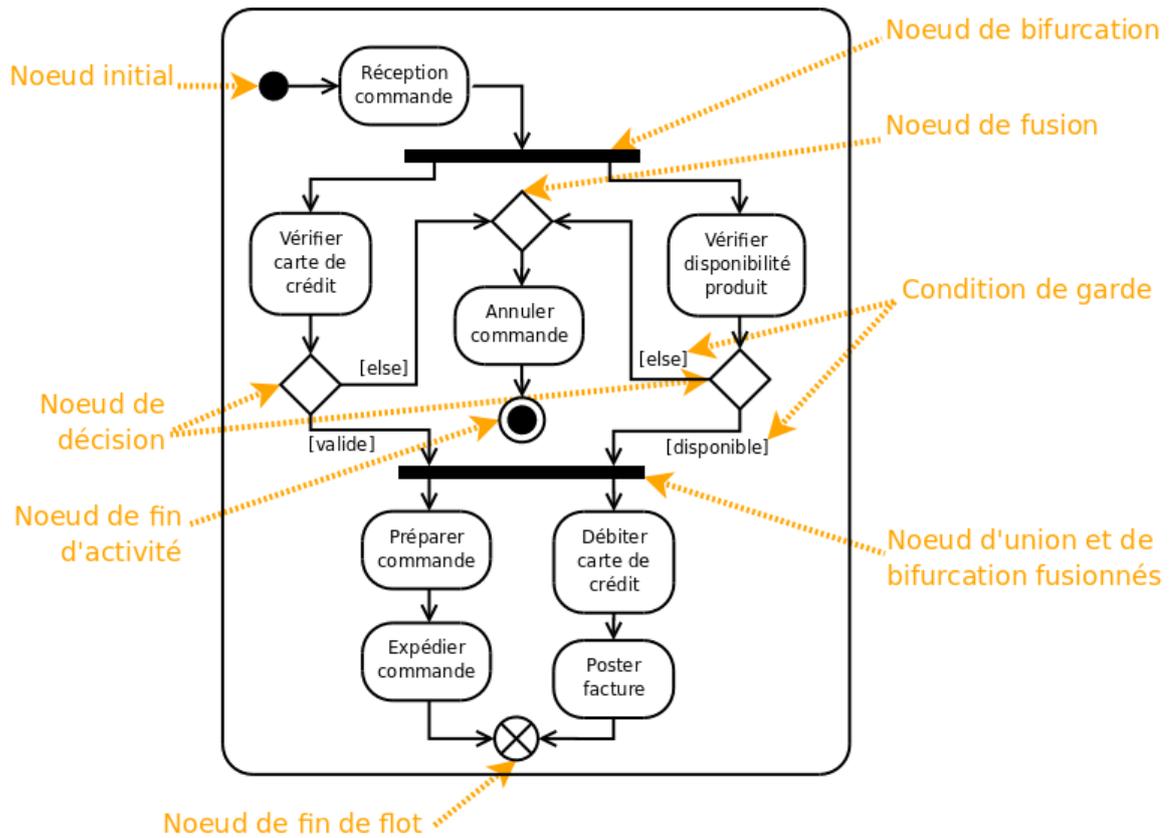


Figure9 :Exemple de diagramme d’activité illustrant l’utilisation de nœuds de contrôle. Ce diagramme décrit la prise en compte d’une commande.[6]

f. Diagramme de séquence et de communication

Le diagramme de séquence représente la succession chronologique des opérations réalisées par un acteur. Il indique les objets que l’acteur va manipuler et les opérations qui font passer d’un objet à l’autre. On peut représenter les mêmes opérations par un diagramme de communication, graphe dont les nœuds sont des objets et les arcs (numérotés selon la chronologie) les échanges entre objets. En fait, le diagramme de séquence et le diagramme de communication sont deux vues différentes mais logiquement équivalentes.

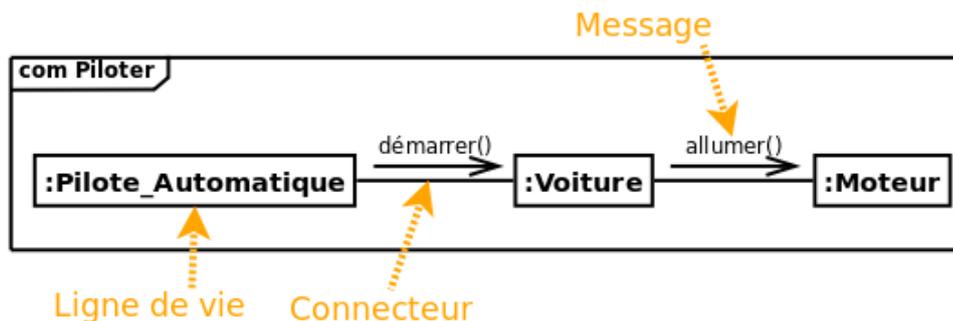


Figure10 : Exemple Diagramme de communication d’un système de pilotage.[6]

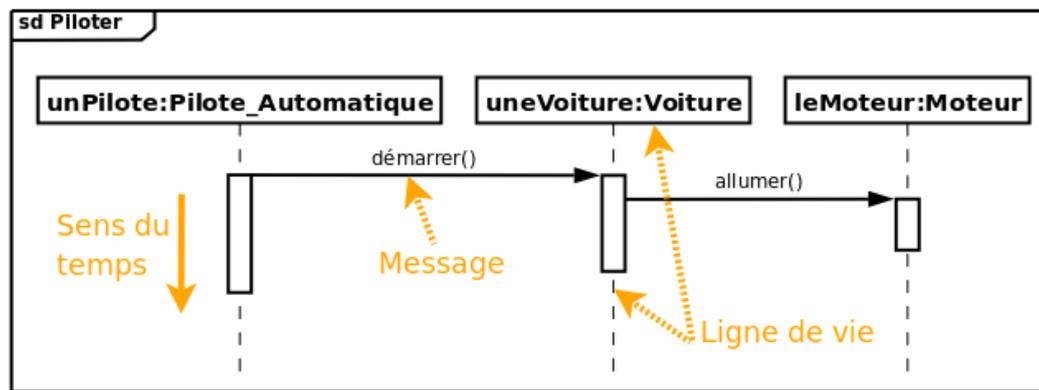


Figure 11 : Exemple Diagramme de séquence d'un système de pilotage.[6]

g. Diagramme de composant

Le diagramme de composants décrit les composants et leurs dépendances dans l'environnement de réalisation ils montrent le choix de réalisation.

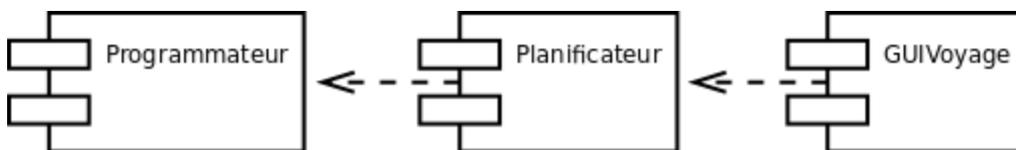


Figure12 : Exemple de diagramme montrant les dépendances entre composants.[6]

h. Diagramme de déploiement

Un diagramme de déploiement décrit la disposition physique des ressources matérielles qui composent le système et montre la répartition des composants sur ces matériels, il précise comment les composants sont répartis sur les nœuds et quelles sont les connexions entre les Composants ou les nœuds. Les diagrammes de déploiement existent sous deux formes :Spécification et Instance.

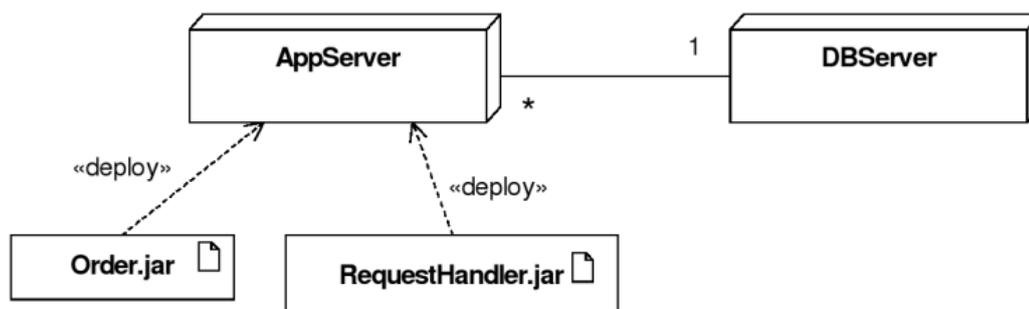


Figure 13: Exemple de diagramme de déploiement illustrant la communication entre plusieurs nœuds.[6]

2. Processus unifié (UP):

2.1. Définition :

Le processus unifié est une méthode de prise en charge du cycle de vie d'un logiciel, et donc, du développement pour les logiciels orientés objets. C'est une méthode générique, itérative et incrémentale, contrairement à la méthode séquentielle Merise (ou SADT).

2.2. Catégories :

a. UP est itératif :

L'itération est une répétition d'une séquence d'instructions ou d'une partie de programme un nombre de fois fixé à l'avance ou tant qu'une condition définie n'est pas remplie, dans le but de reprendre un traitement sur des données différentes.

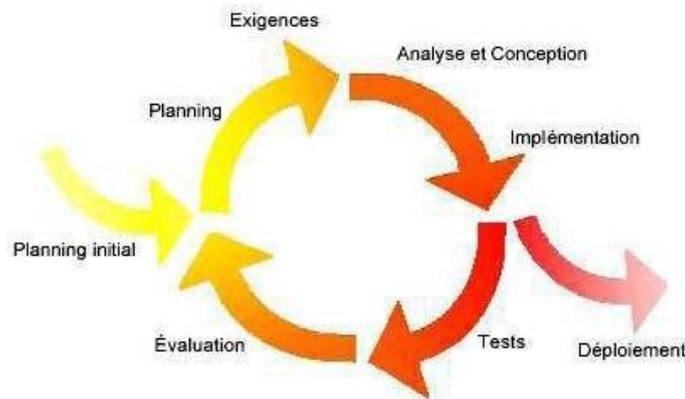


Figure14: l'itération dans l'UP

b. UP est centré sur l'architecture :

Les différentes perspectives, indépendantes et complémentaires, qui permettent de définir le modèle d'architecture, sont proposées par Ph. Kruchten.

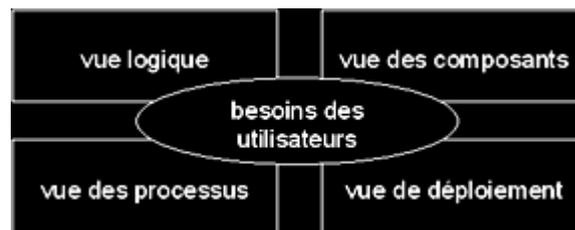


Figure 15: les différentes perspectives qui définissent un modèle d'architecture

c. UP est piloté par les cas d'utilisation d'UML :

La satisfaction des besoins du client c'est le but principal d'un système informatique le processus de développement sera donc accès sur l'utilisateur. Les cas d'utilisation permettent d'illustrer ces besoins. Ils détectent puis décrivent les besoins fonctionnels (du Pointdevue de l'utilisateur), et leur ensemble constitue le modèle de cas d'utilisation dictelesfonctionnalités complètes du système.[6]

2.3. Les phases :

a. Analyse des besoins :

L'analyse des besoins donne une vue du projet sous forme de produit fini. Elle porte essentiellement sur les besoins principaux (du point de vue de l'utilisateur), l'architecture générale du système, les risques majeurs, les délais et les coûts pour mettre en place le projet. Elle répond aux questions suivantes :

Que va faire le système ? Par rapport aux utilisateurs principaux, quels services va-t-il rendre?

Quelle va être l'architecture générale (cible) de ce système ?

Quels vont être : les délais, les coûts, les ressources, les moyens à déployer?[6]

b. Élaboration :

L'élaboration reprend les éléments de la phase d'analyse des besoins et les précise pour arriver à une spécification détaillée de la solution à mettre en œuvre.Elle permet de préciser laplupart des cas d'utilisation, de concevoir l'architecture dusystème et surtout de déterminer l'architecture de référence. Au terme de cette phase, les chefs de projet doivent être en mesure de prévoir les activités et d'estimer les ressources nécessaires à l'achèvement du projet.Les taches à effectuer dans la phase élaboration sont les suivantes :

- Créer une architecture de référence.
- Identifier les risques.
- Définir les niveaux de qualité à atteindre.
- Formuler les cas d'utilisation.

Élaborer une offre abordant les questions de calendrier, de personnel et de budget.[6]

c. Construction :

La construction est le moment où l'on construit le produit. L'architecture de référence

se transforme en produit complet. Le produit contient tous les cas d'utilisation que les chefs de projet, en accord avec les utilisateurs ont décidé de mettre au point pour cette version.[6]

d. Transition :

Le produit est en version bêta. Un groupe d'utilisateurs essaye le produit et détecte les spécificités et défauts.

2.4. Une méthode simple et générique :

C'est une méthode qui se situe à mi-chemin entre UP (Unified Process), qui constitue un cadre général très complet de processus de développement, et XP (eXtreme Programming) qui est une approche minimaliste à la mode centrée sur le code.

2.4.1. Identification des besoins et spécification des fonctionnalités :

a. diagramme de cas d'utilisation :

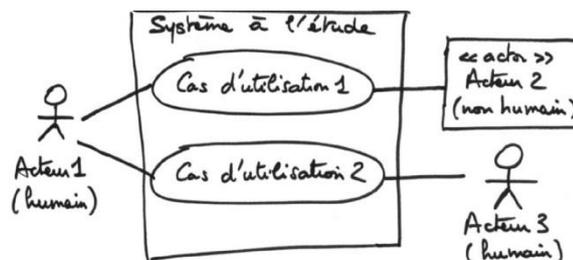


Figure16: Diagramme de cas d'utilisation (bases).

b. diagramme de séquence système :

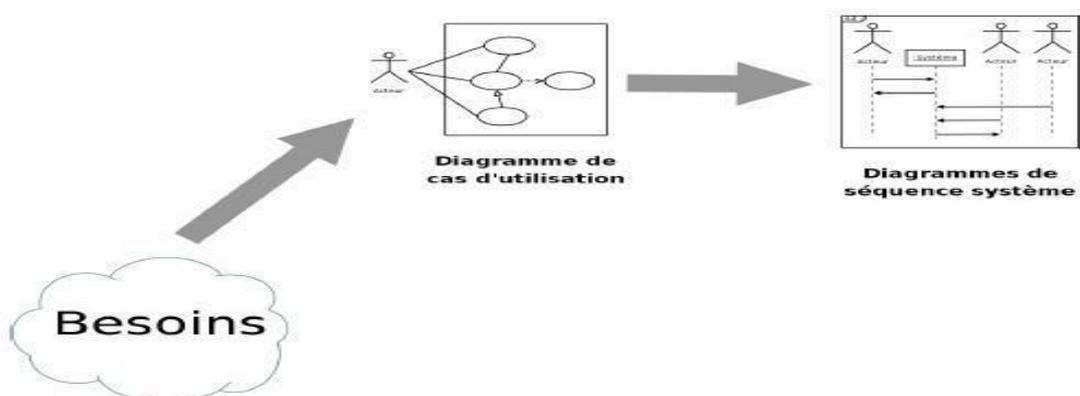


Figure17: Les diagrammes de séquence illustrent la description textuelle des cas d'utilisation.[7]

c. Maquette IHM :

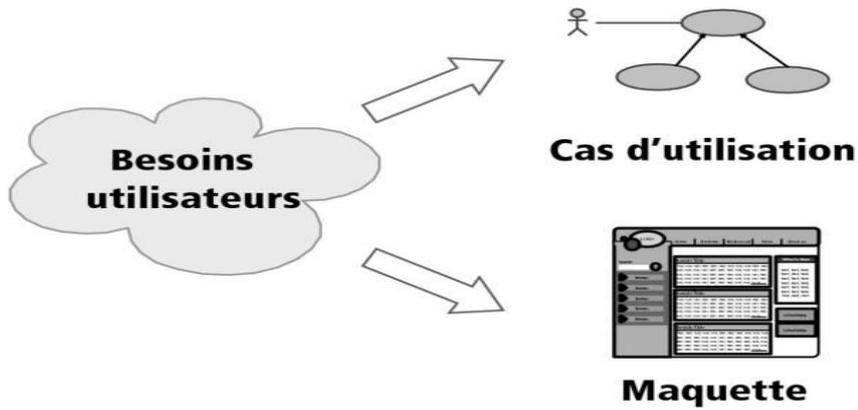


Figure18: Les besoins donnent lieu à des cas d'utilisation et à une maquette.[7]

2.4.2. Phase d'analyse :

a. diagramme de domaine :

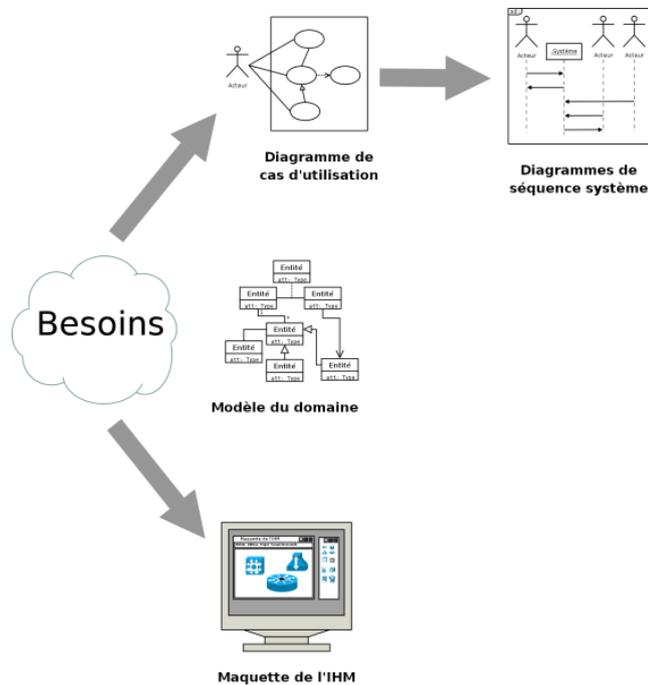


Figure 19: La phase d'analyse du domaine permet d'élaborer la première version du diagramme de classes.[7]

b. diagramme des classes participantes :

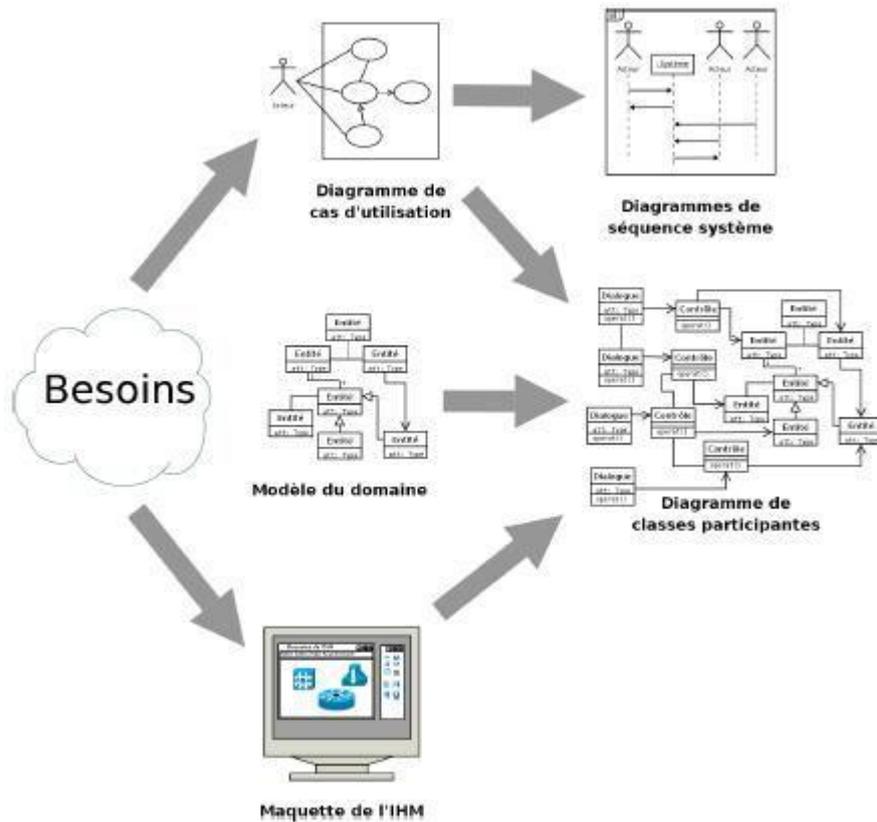


Figure20: Le diagramme de classes participantes effectue la jonction entre les cas d'utilisation, le modèle du domaine et les diagrammes de conception logicielle.[7]

c. diagramme d'activité de navigation :

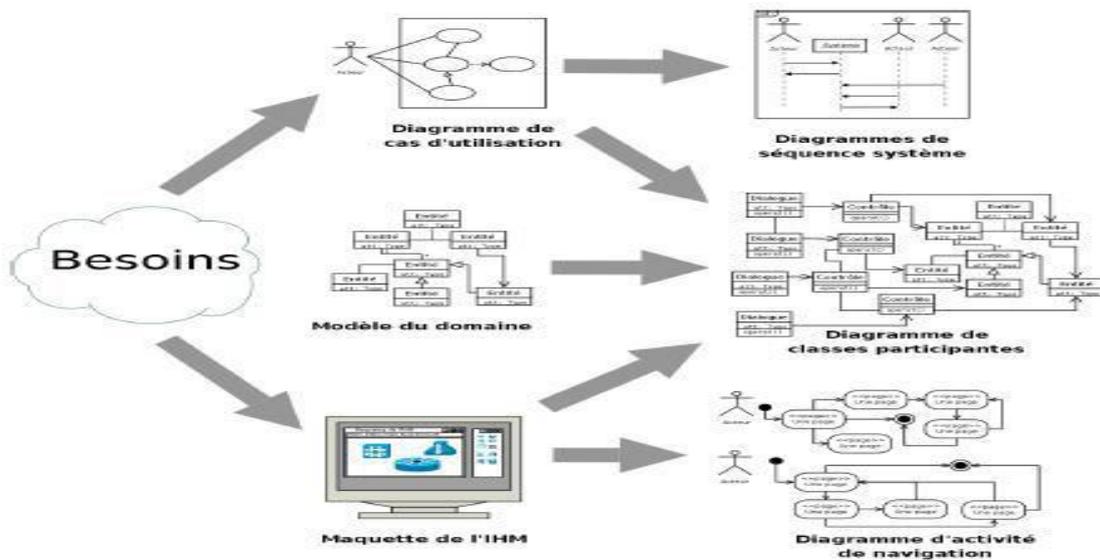


Figure21: Les diagrammes d'activités de navigation représentent graphiquement l'activité de navigation dans l'IHM.[7]

2.4.3. Phase de conception :

a. diagramme d'interaction :

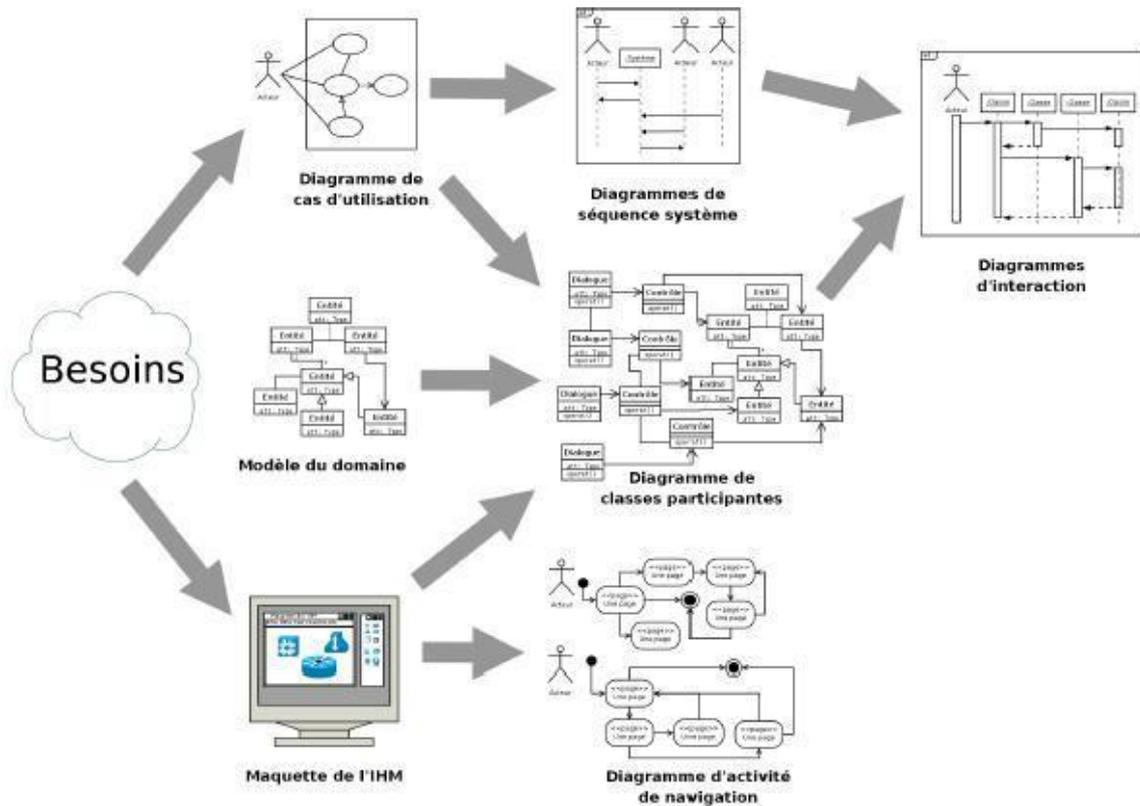


Figure22: Les diagrammes d'interaction permettent d'attribuer précisément les responsabilités de comportement aux classes d'analyse.[7]

b. Diagramme de classe

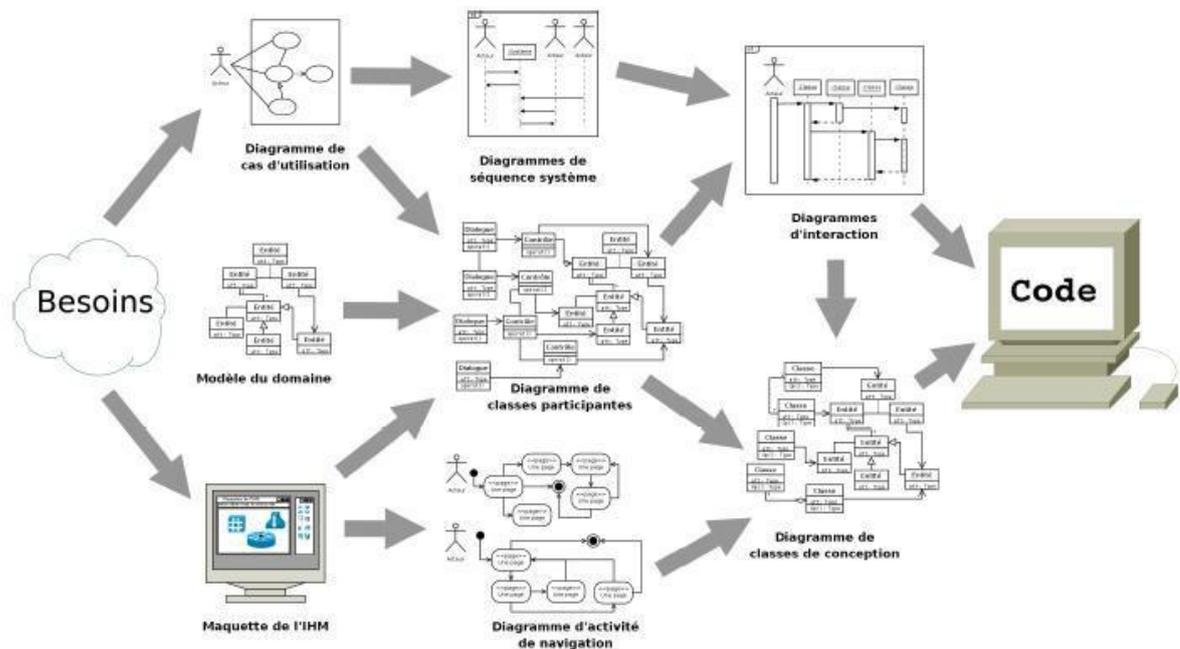


Figure 23 : Chaîne complète de la démarche de modélisation du besoin jusqu'au code.[7]

Conclusion :

UML est un moyen d'exprimer des modèles en faisant une abstraction de leur implémentation, i.e. que le modèle fourni par UML est valable pour n'importe quel langage de programmation. Il offre une manière de représenter le système selon différentes vues complémentaires grâce aux diagrammes. Pour une pertinente conception, UML doit suivre une démarche de développement. La démarche qu'on a suivi c'est UP (une approche simplifiée destinée pour les applications web).

Chapitre III

Identification des besoins

Introduction

Après avoir bien étudié notre domaine , et choisir le langage de modélisation et le processus de développement, nous allons entamer dans ce chapitre la première étape du Processus UP qui est l'identification des besoins.

1. Identification des acteurs :

Un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. Dans notre projet il existe trois acteurs qu'est :

Administrateur : c'est le personne qui responsable de créer les Comptes et faire la mise à jour de site.

Visiteur : le personne qui consulter le site et peut inscrit dans le site.

Membre : le personne qui consulter et participer dans le site.

Participant : le personne qui consulter et modifier sa participation.

Juré : la personne qui peut gérer les listes des participations.

2. Diagramme de cas d'utilisation :

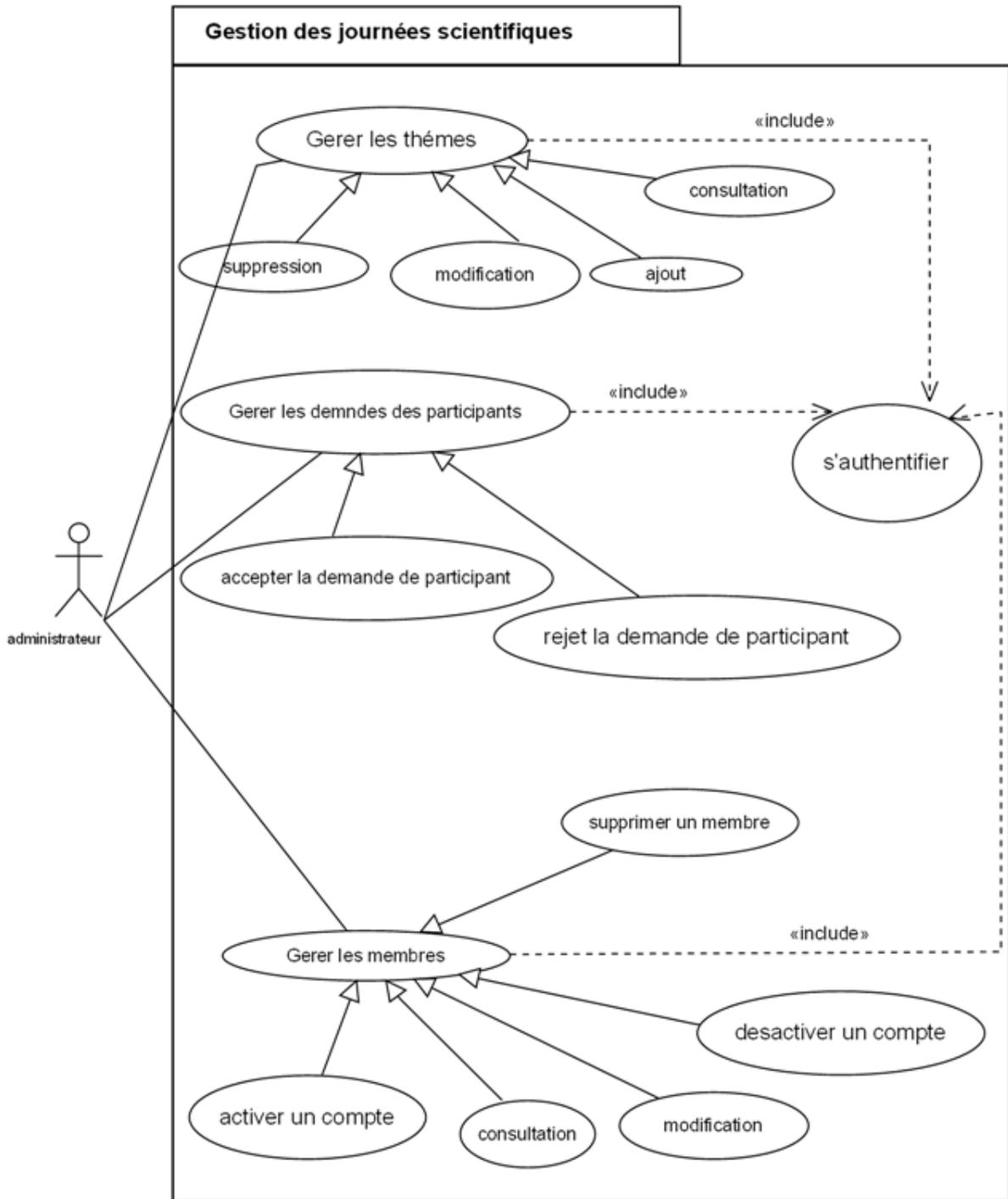


Figure 24.1 : diagramme de cas d'utilisation.

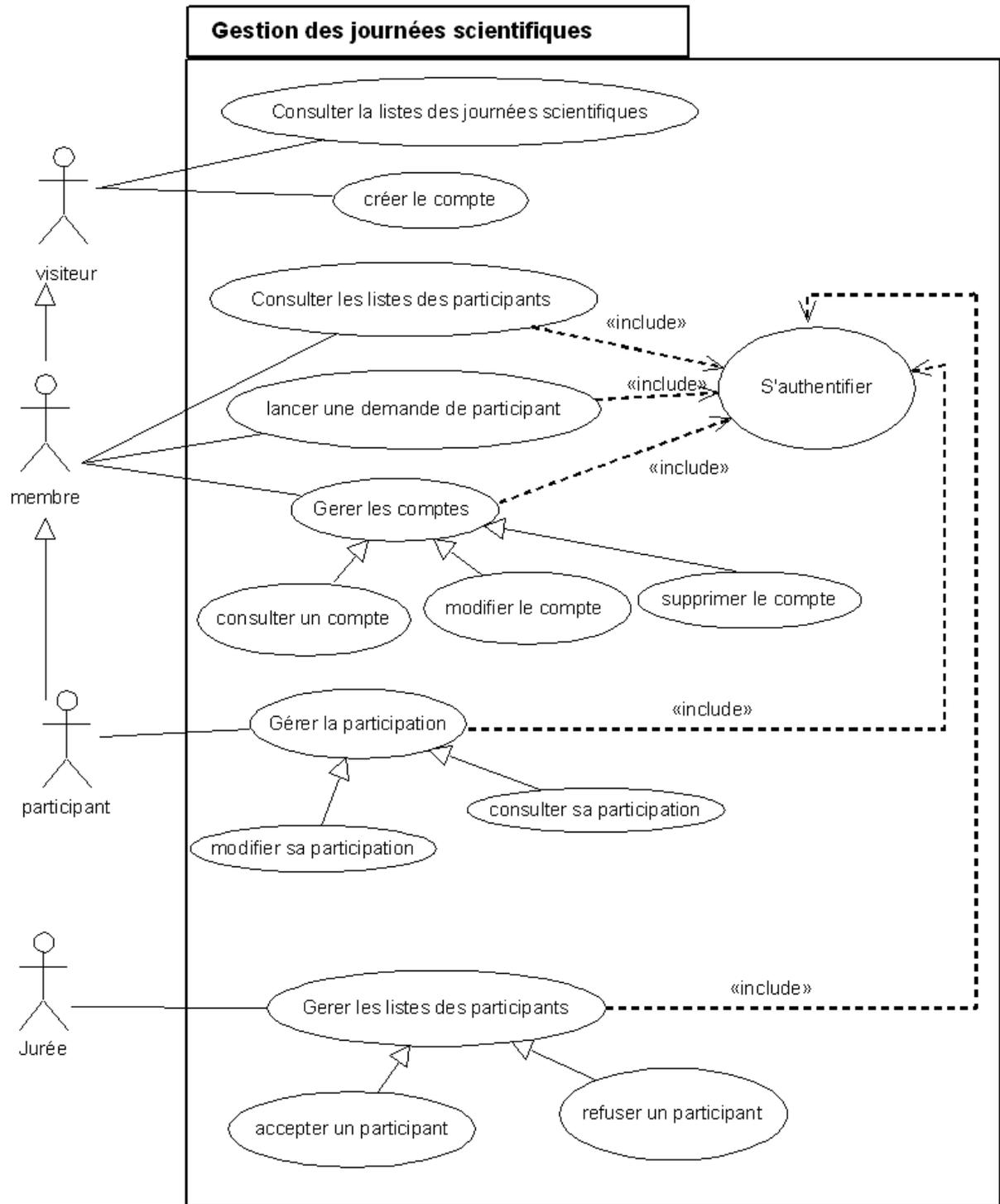


Figure 24.2 : diagramme de cas d'utilisation.

3. Les fiches descriptives des cas d'utilisations

3.1. Fiche descriptive du cas d'utilisation «S'authentifier» :

S'authentifier	
Description	Ce cas permet à un membre, un participant, ou une jurée de s'authentifier pour avoir accédé à leur espace.
Acteur	Participant, membre, jurée.
Pré conditio	Chaque utilisateur (participant, membre, jurée) possède un compte.
Post condition	Le système connaît l'utilisateur et son type d'utilisation.
Nominal	1-Le système demande à l'utilisateur de saisir le mot de passe et le pseudonyme. 2-L'utilisateur saisit ses données. 3-Le système vérifier les données de l'utilisateur et les validés. 4-Le système permet à l'utilisateur d'ouvrir son compte et entrer dans son espace. 2-1-Le système lui demande de remplir un formulaire. 2-2-L'utilisateur saisie les informations nécessaire. 2-3-Le système vérifier les informations saisies.
Alternative	1-L'user Name ou le mot de passe est erroné « retour au scénario nominal 1 » 2-Mot de passe oublié.
Exception	L'utilisateur quitter le système.

Tableau 1 : « S'authentifier »

3.2. Fiche descriptive du cas d'utilisations « Gérer les thèmes » :

« Gérer les thèmes »	
Description	Ce cas permet à l'administrateur de gérer les thèmes.
Acteur	Administrateur
Pré conditio	L'administrateur est déjà authentifier.
Post condition	Le système valide l'enregistrement.
Nominal	1-1-L'administrateur demande au système d'afficher les listes es thèmes. 1-2-Le système affiche les listes des thèmes. 1-3-L'administrateur fait la consultation des thèmes. 1-4-L'administrateur demande au système de modifier un thème. 1-5-Le système permet à l'administrateur de modifier un thème. 1-6-L'administrateur fait la modification et enregistré. 1-7-Le système affiche le message d'enregistrement. 2-1- L'administrateur demande au système de supprimer un thème. 2-2-Le système permet à l'administrateur de supprimer un thème. 2-3-L'administrateur supprimer le thème et enregistré. 2-4-Le système affiche le message d'enregistrement et affiche le thème supprimé. 3-1- L'administrateur demande au système d'ajouter un thème. 3-2-Le système permet à l'administrateur d'ajouter un thème ou plusieurs. 3-3-L'administrateur ajouter le thème et enregistré. 3-4-Le système fait d'enregistrement.
Alternative	Erreur en l'information.
Exception	L'administrateur annuler la modification/suppression /ajout.

Tableau 2 : « Gérer les thèmes »

3.3. Fiche descriptive du cas d'utilisations « Gérer les demandes des participants » :

« Gérer les demandes des participants »	
Description	Ce cas permet au l'administrateur de Gérer les demandes des participants
Acteur	L'administrateur
Pré condition	L'administrateur est déjà authentifier
Post condition	Le système valide l'enregistrement
Nominal	1-L'administrateur demande au système d'afficher les listes des Participants. 2-Le système affiche les listes des demandes des participants. 3- L'administrateur fait (accepter /rejeter) les demandes de participants. 4- Le système enregistre le nouvel état.
Alternative	L'administrateur annulé la demande d'affichage des listes
Exception	La demande de participant est déjà accepter

Tableau 3 : « Gérer les demandes des participants »

3.4. Fiche descriptive de cas d'utilisation « Gérer les membres»

« Gérer les membres »	
Description	Ce cas permet au l'administrateur de Gérer les membres
Acteur	L'administrateur
Pré condition	L'administrateur est déjà authentifier
Post condition	Le système valide l'enregistrement
Nominal	1-L'administrateur demande au système d'activer /désactiver un compte d'un membre. 2-Le système permet à l'administrateur d'activer/désactiver un compte d'un membre 3-L'administrateur demande au système de modifier (Suppression, consultation) d'un compte 4-Le système permet à l'administrateur de modifier un compte d'un membre 5- L'administrateur fait la modification (suppression, consultation). 6-Le système à enregistre les modifications (suppression).
Exception	L'administrateur quitte le site.

Tableau 4 : « Gérer les membres »

3.5. Fiche descriptive du cas d'utilisation « Consulter la liste des journées scientifiques »

« Consulter la liste des journées scientifiques »	
Description	Ce cas permet à visiteur de consulter la liste des journées scientifiques
Acteur	Visiteur
Pré condition	Connexion est existe
Post condition	Le système afficher la liste des journées scientifiques
Nominal	1-Le visiteur demande au système de consulter la liste des journées scientifiques. 2-Le système afficher la liste des journées scientifiques
Exception	Visiteur quitter le site.

Tableau 5 : « Consulter la liste des journées scientifiques »

3.6. Fiche descriptive du cas d'utilisation « Créer le compte »

« Créer le compte »	
Description	Ce cas permet au visiteur d'inscrire dans le site
Acteur	Visiteur
Pré condition	Connexion est existe
Post condition	Le système enregistre un nouveau membre
Nominal	1-Le visiteur demande d'inscrire dans le site. 2-Le système donne le formulaire d'inscription au visiteur. 3-Le visiteur saisit les informations. 4-Le système affiche que l'inscription est réussite.
Alternative	L'administrateur annulé la demande d'affichage des listes.
Exception	La demande de participant est déjà accepter.

Tableau 6 : « Créer le compte »

3.7. Fiche descriptive du cas d'utilisation « Consulter les listes des participants » :

« Consulter les listes des participants »	
Description	Ce cas permet à un membre de consulter les listes des participants
Acteur	Membre
Pré condition	Connexion est existe
Post condition	Le système affiche les listes des participants.
Nominal	1-Un membre demandé de consulter les listes des participants. 2-Le système affiche les listes des participants.
Alternative	
Exception	Membre quitter le site.

Tableau 7 : « Consulter les listes des participants »

3.8. Fiche descriptive du cas d'utilisation « Lancer une demande de participation » :

« Lancer une demande de participation »	
Description	Ce cas permet à un membre de lancer une demande de Participation
Acteur	Membre
Pré condition	Membre est déjà s'authentifier
Post condition	Le système sauvegarde la nouvelle demande de participation
Nominal	1-Le membre lancer une demande de participation 2-Le système demande au membre d'insérer son exposé. 3-Le membre inséré l'exposé. 4-Le système sauvegarde la nouvelle demande de participation.
Exception	Le membre annuler la participation dans une journée scientifique

Tableau 8 : « Lancer une demande de participation »

3.9. Fiche descriptive du cas d'utilisation « Gérer le compte » :

« Gérer le compte »	
Description	Ce cas permet au membre de gérer son compte.
Acteur	Membre
Pré condition	Le membre est authentifié.
Post condition	Le système valide l'enregistrement.
Nominal	1- Le membre demandé au système de consulter son compte. 2-Le système permet à un membre de consulter son compte. 3-Le membre demandé au système de modifier son compte. 4-Le système permet à un membre de modifier son compte. 5-Le membre modifier son compte et enregistré. 6-Le système afficher le message d'enregistrement. 7-Le membre demandé au système de supprimer son compte. 8-Le système permet à un membre de supprimer son compte. 9-Le membre supprimer son compte et enregistré. 10-Le système fait l'enregistrement.
Alternative	Erreur en l'information (retour au scénario nominal 3)
Exception	Le membre annuler la modification /suppression.

Tableau 9 : « Gérer le compte »

3.10. Fiche descriptive du cas d'utilisation « Gérer les listes des participants » :

« Gérer les listes des participants »	
Description	Ce cas permet à une jurée de gérer les listes des participants.
Acteur	Jurée
Pré condition	Jurée est authentifier.
Post condition	Le système affiche les listes des participants.
Nominal	1-Jurée demande au système d'afficher les listes des participants. 2-Le système affiche les listes des participants. 3-Jurée choisi d'accepter /refuser une participation. 4-Le système enregistre le nouvel état.
Alternative	
Exception	Jurée annulé d'accepter/refuser une participation.

Tableau 10 : « Gérer les listes des participants »

3.11. Fiche descriptive du cas d'utilisation « Gérer la participation » :

« Gérer la participation »	
Description	Ce cas permet au participant de gérer sa participation.
Acteur	Participant
Pré condition	Le participant est déjà authentifier
Post condition	Le système enregistre la modification du participant

Nominal	1-1-Le participant demande au système de consulter la participation. 1-2-Le système permet au participant de consulter la participation. 2-1- Le participant demande au système de modifier sa participation 2-2- Le système permet au participant de modifier sa participation 2-3- Le participant fait la modification et enregistre. 2-4-Le système affiche le message d'enregistrement. 3-1- Le participant demande au système de supprimer sa participation 3-2- Le système permet au participant de supprimer sa participation 3-3- Le participant fait la suppression et enregistre. 3-4-Le système affiche le message d'enregistrement.
Alternative	
Exception	Le participant quitter le système

Tableau 11 : « Gérer la participation »

4. Les diagrammes de séquence système

4.1. S'authentifier

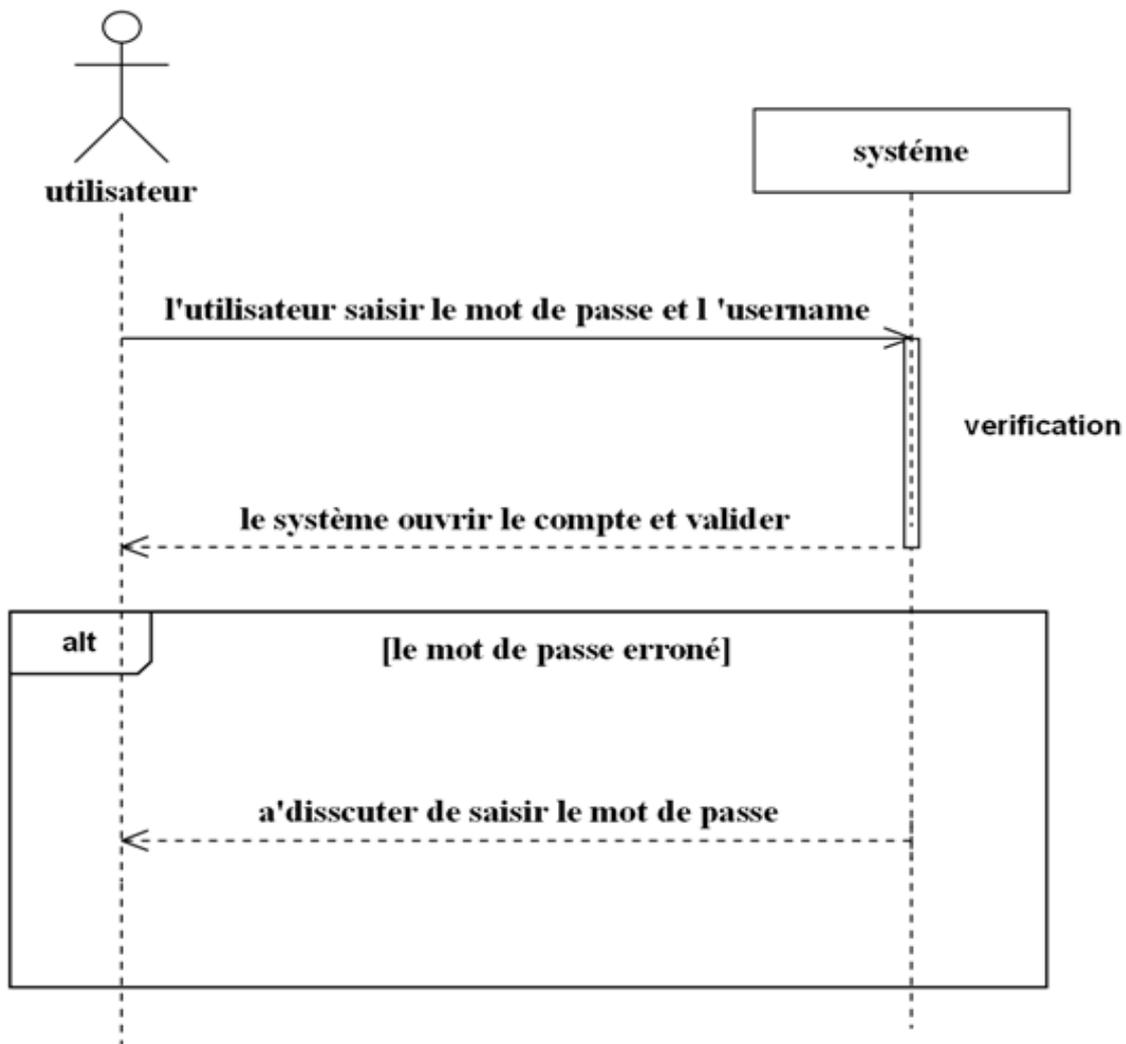


Figure 25 : « S'authentifier »

4.2. Gérer les thèmes

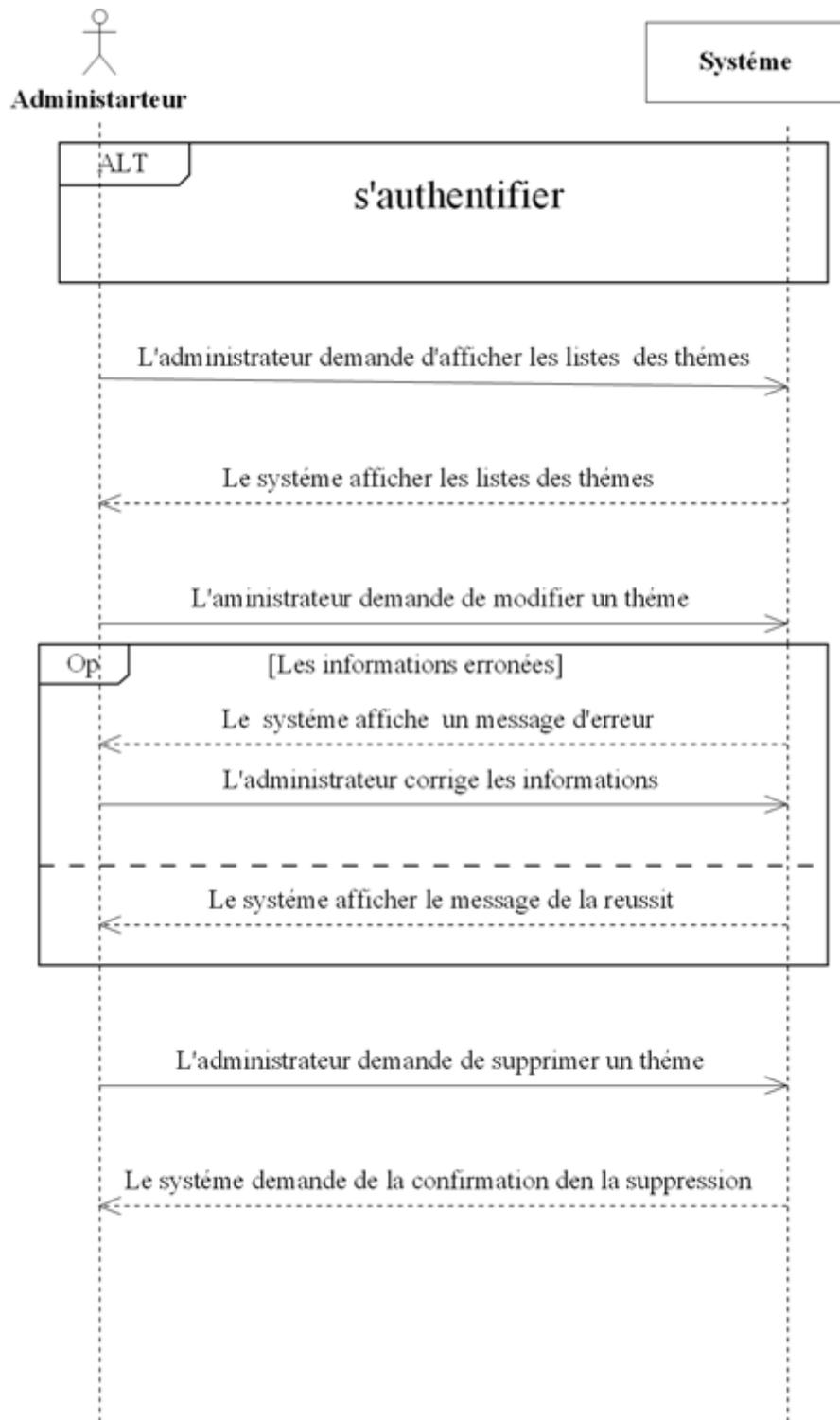
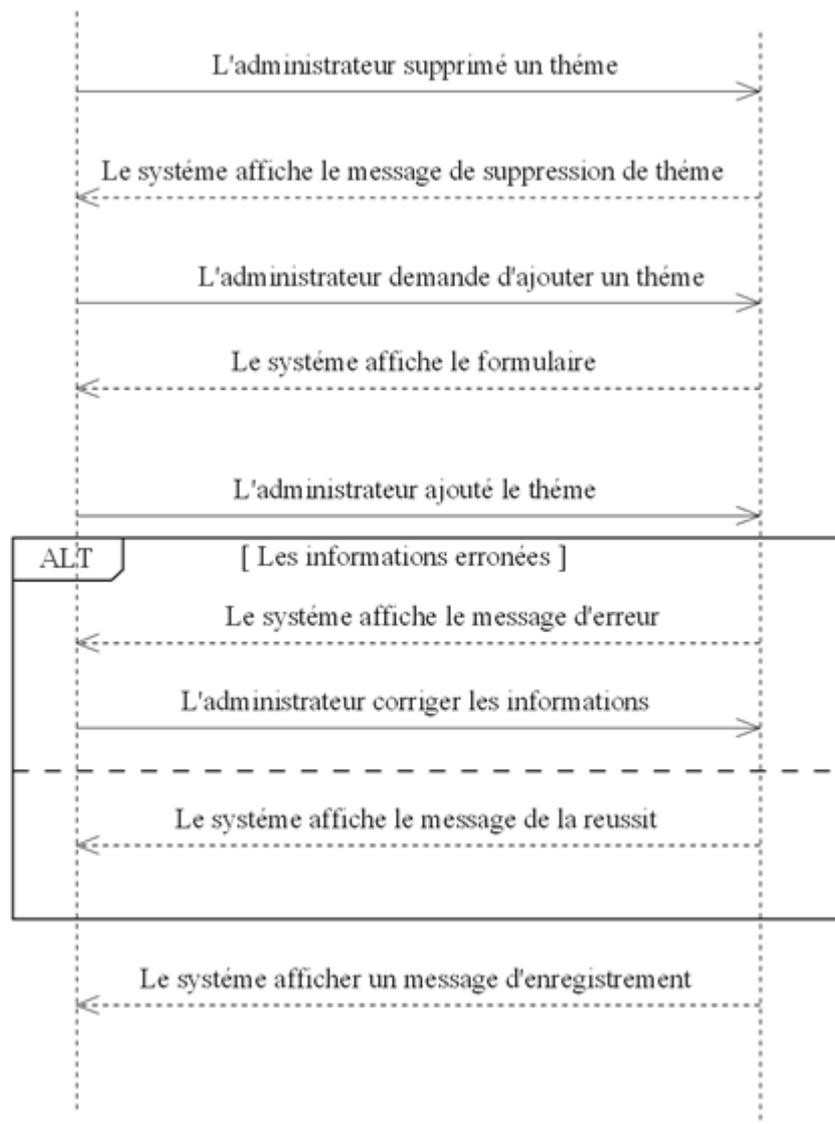


Figure 26.1 : « Gérer les thèmes »

**Figure 26.2 : « Gérer les thèmes »**

4.3. Gérer les listes des participations :

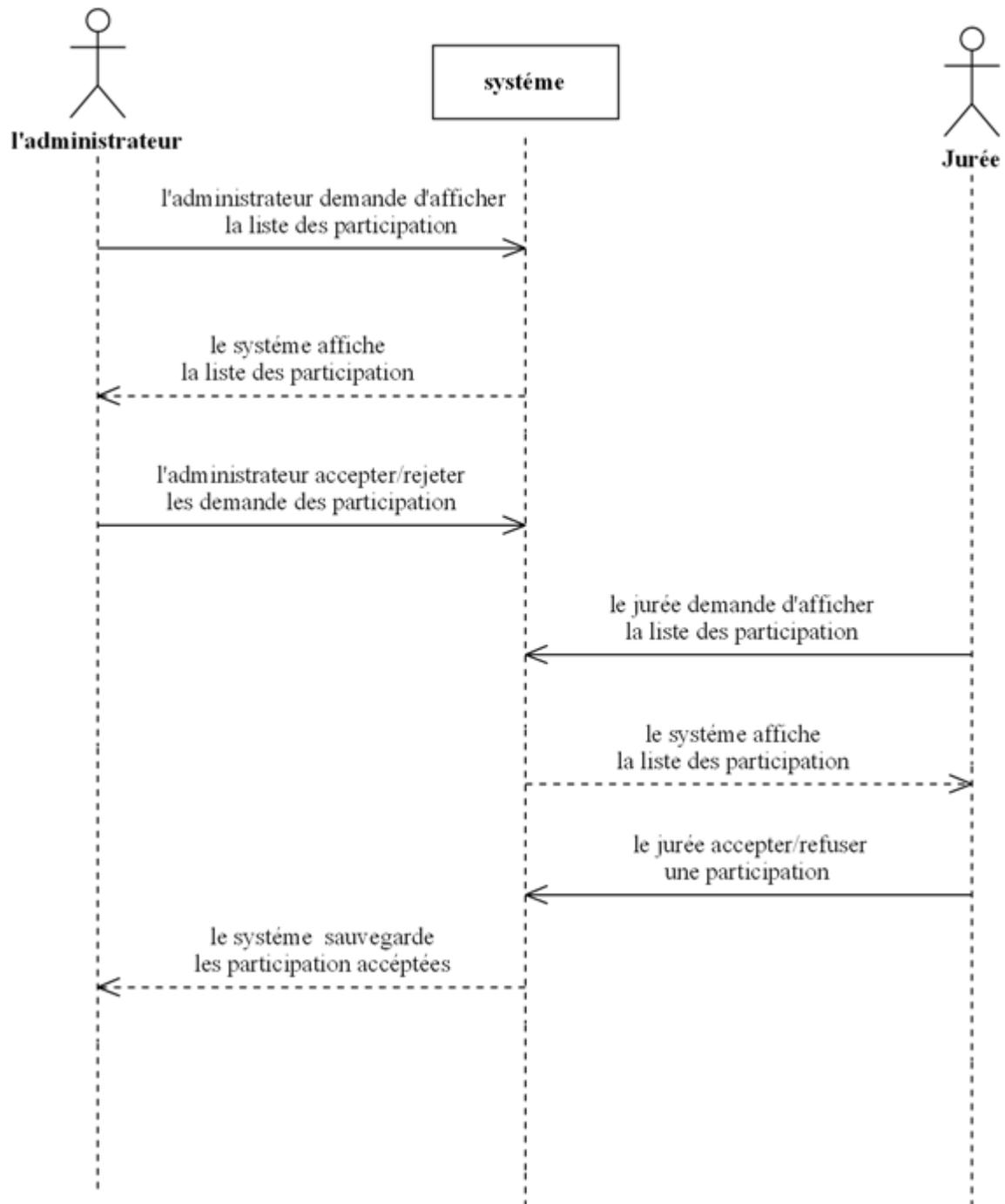
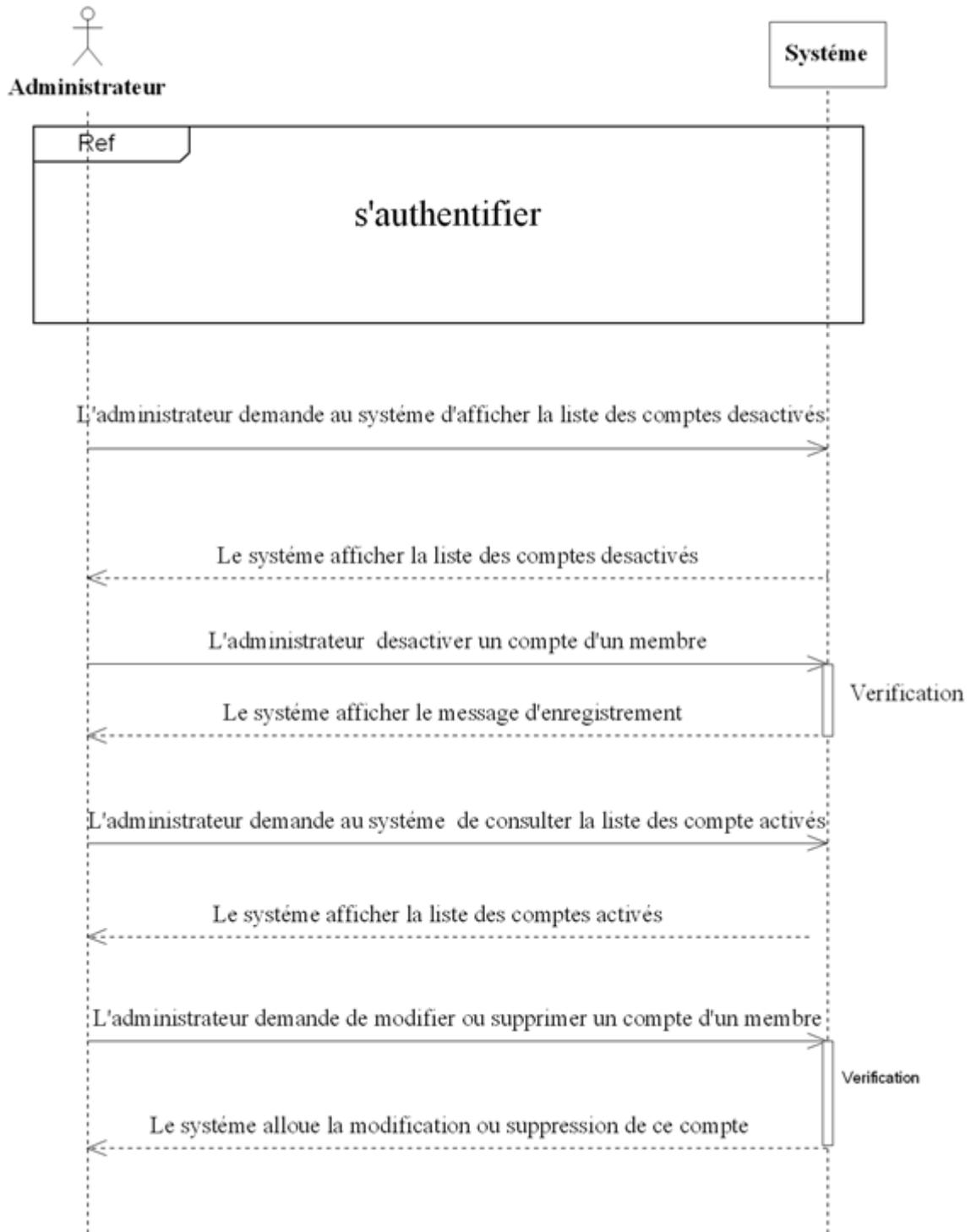


Figure 27 : « Gérer les listes des participations »

4.4. Gérer les membres :



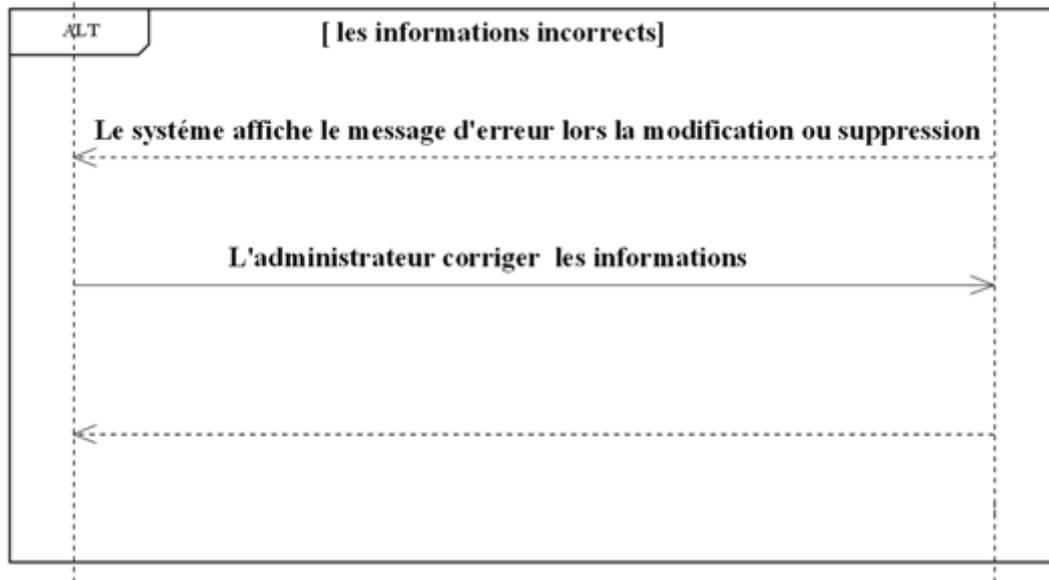


Figure 28 : « Gérer les membres »

4.5. Créer le compte

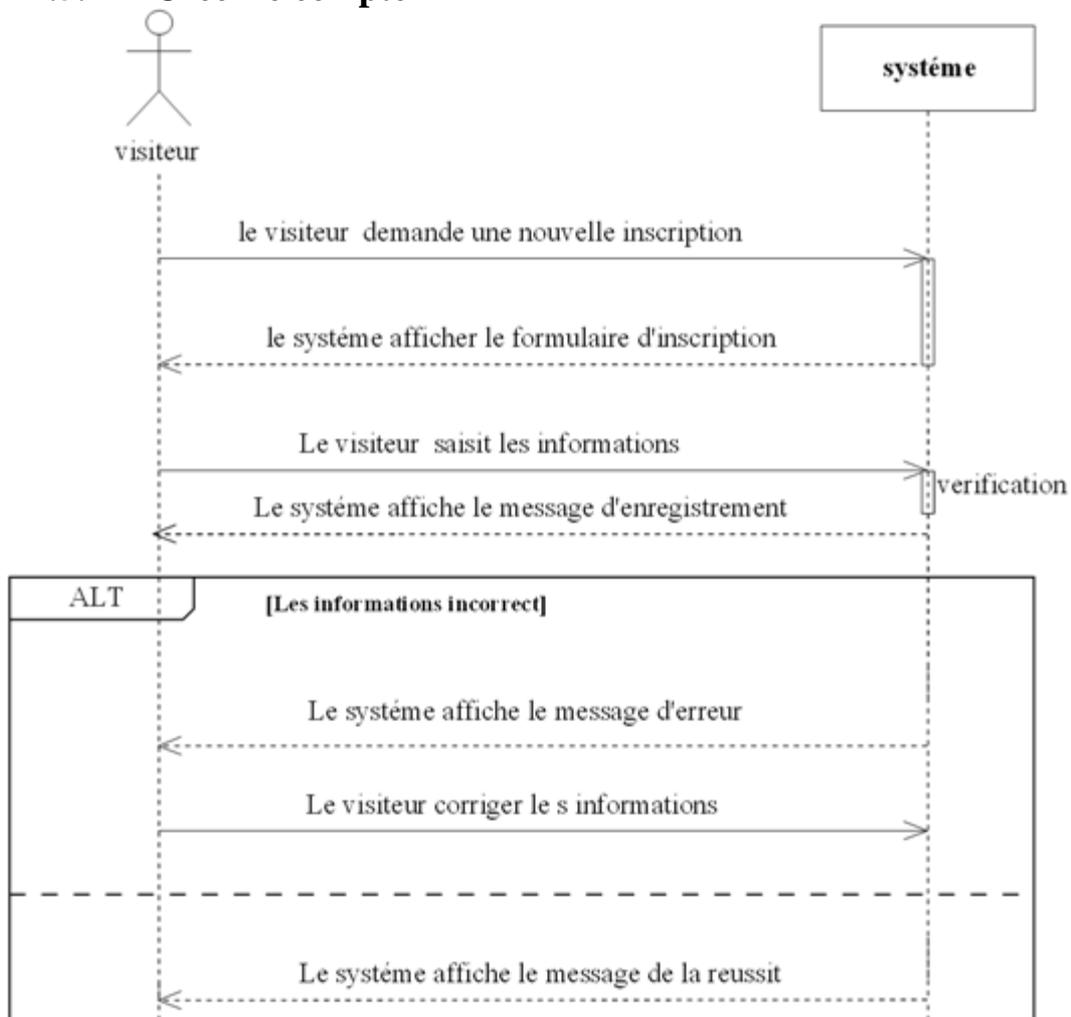


Figure 29 : « Créer le compte »

4.6. Lancer une demande de participation

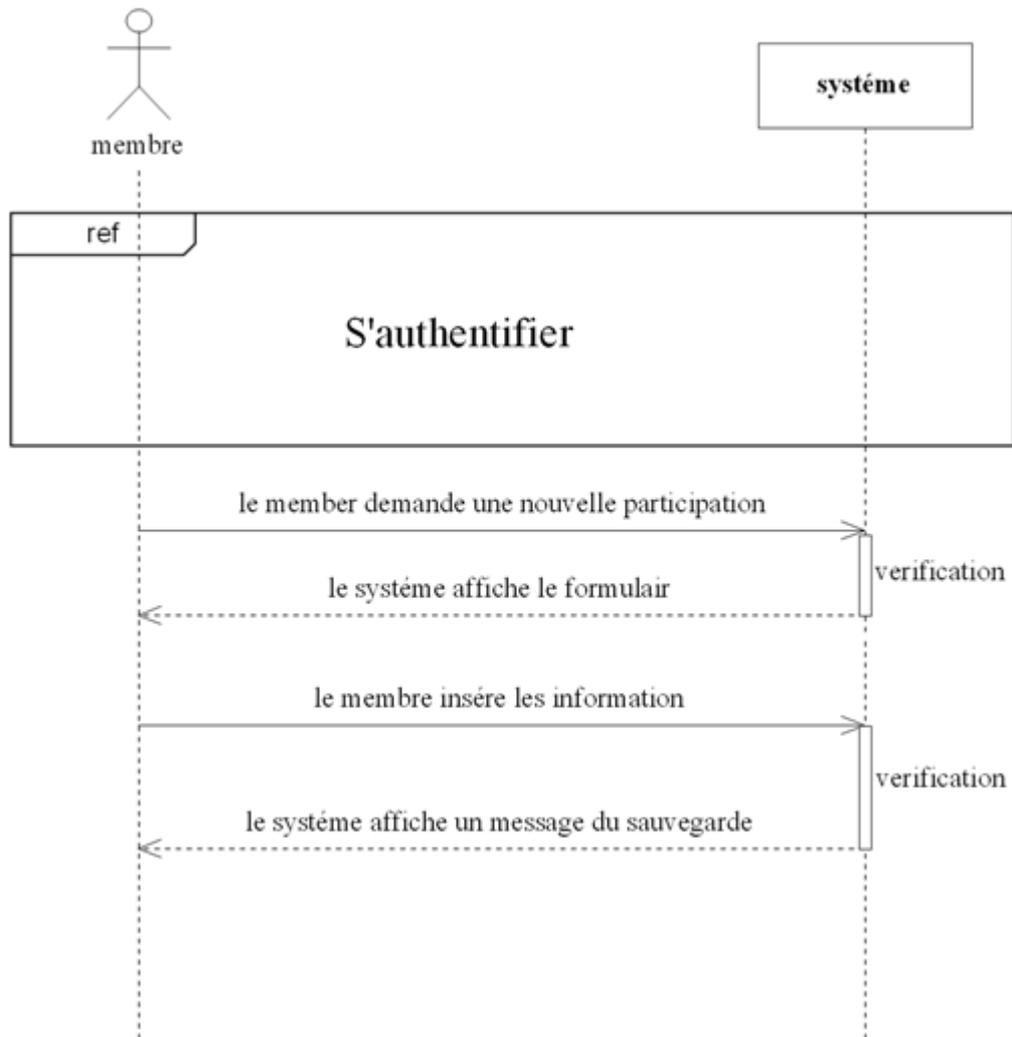


Figure 30: « Lancer une demande de participation »

4.7. Gérer la participation

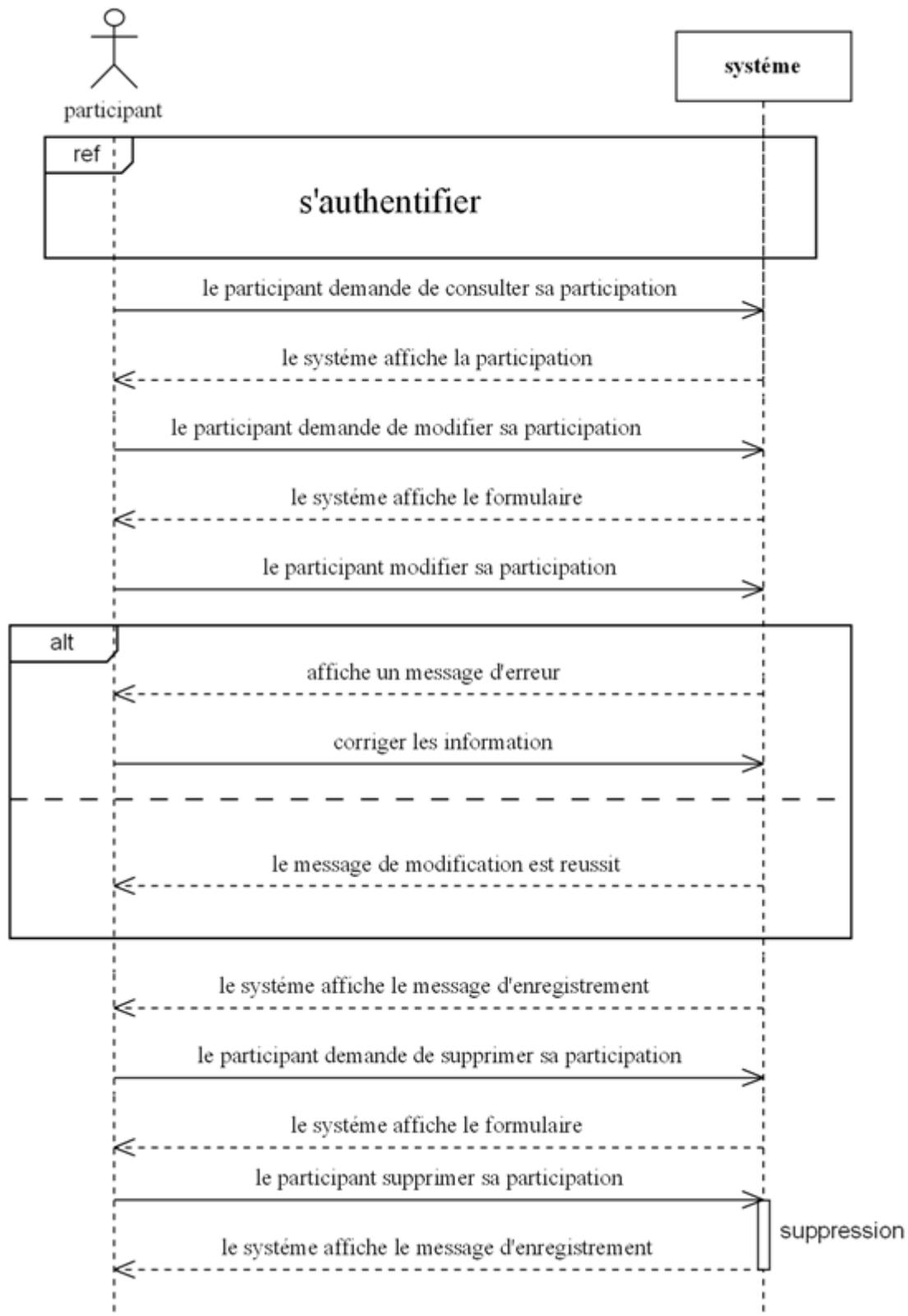


Figure 31 : « Gérer la participation »

5. Les digrammes d'activité de navigation

5.1. Créer compte

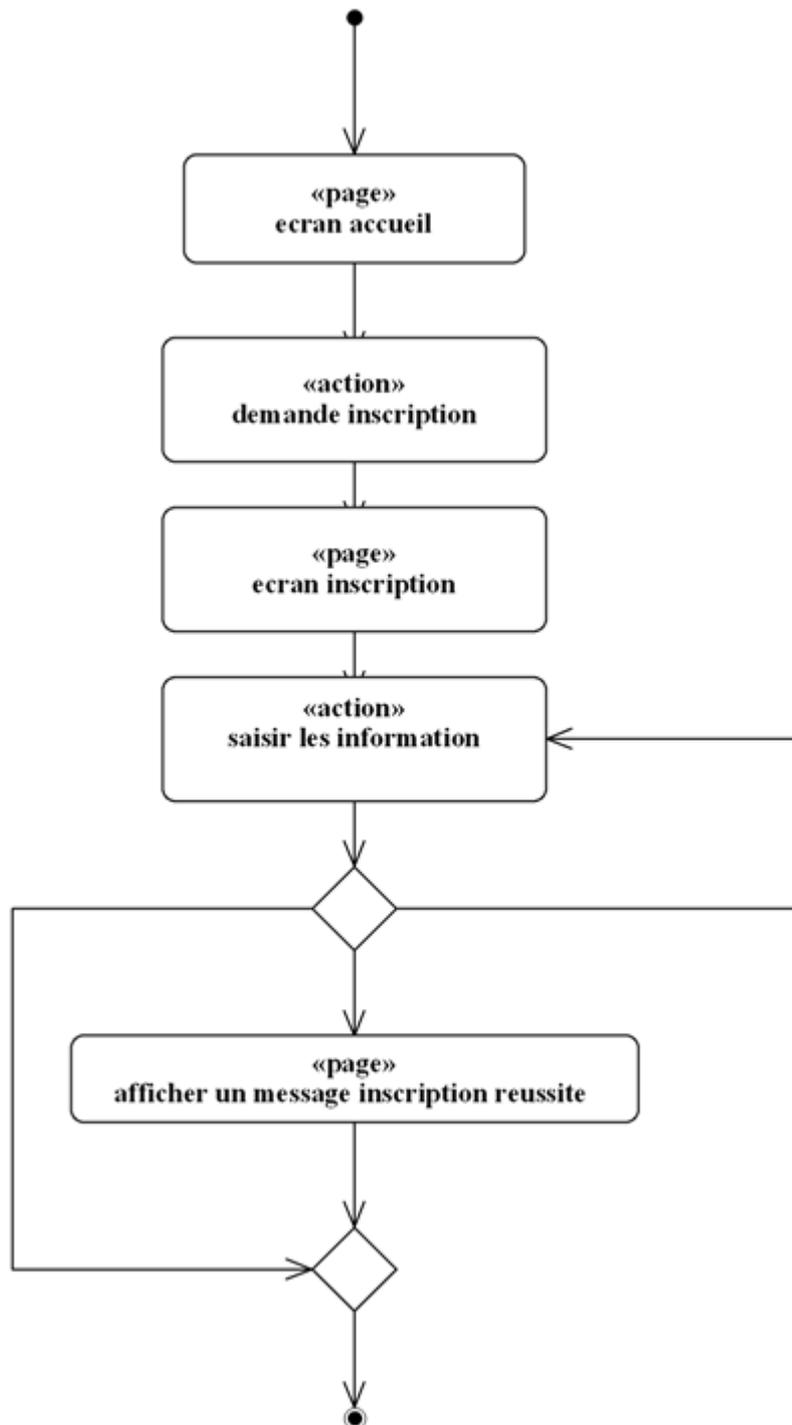


Figure 32 : Créer compte.

5.2. Gérer les listes des participations

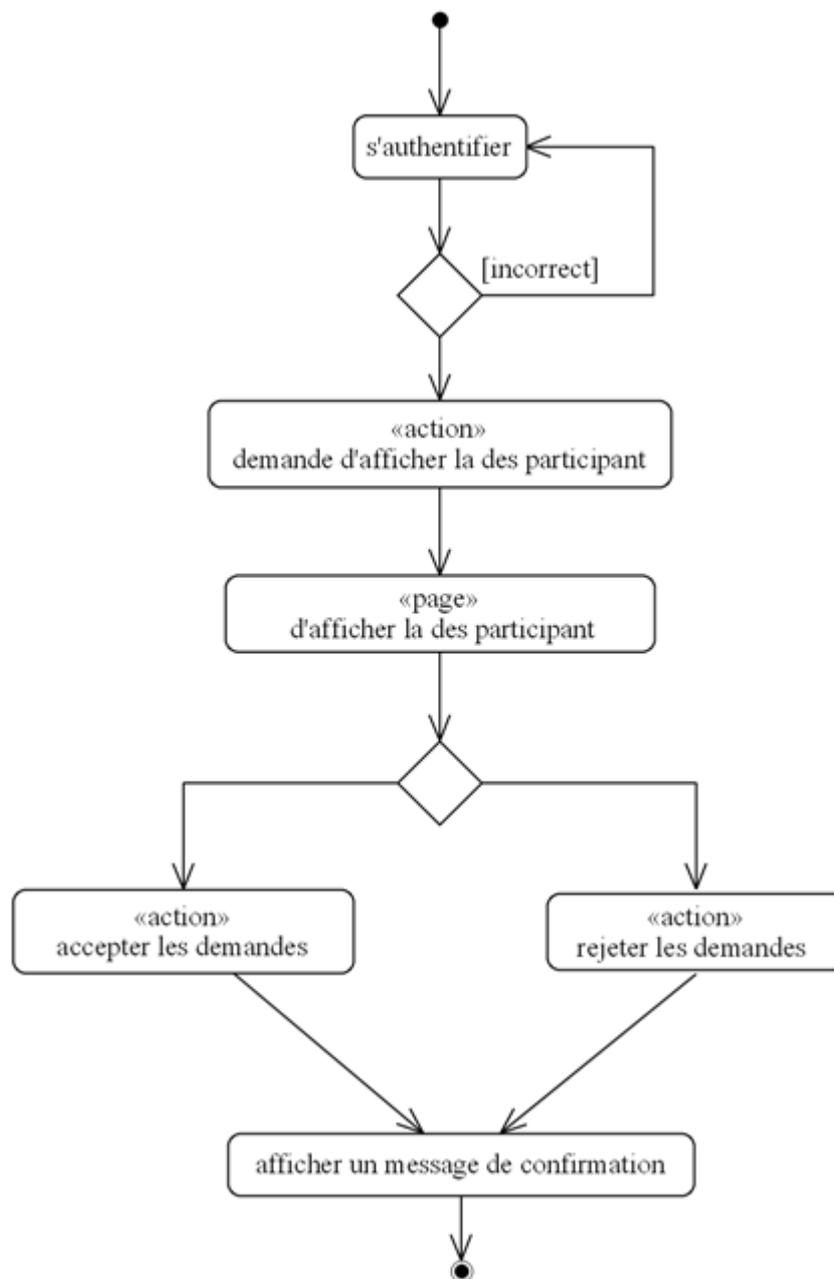


Figure 33 : Gérer les listes des participations

5.3. Gérer les thèmes

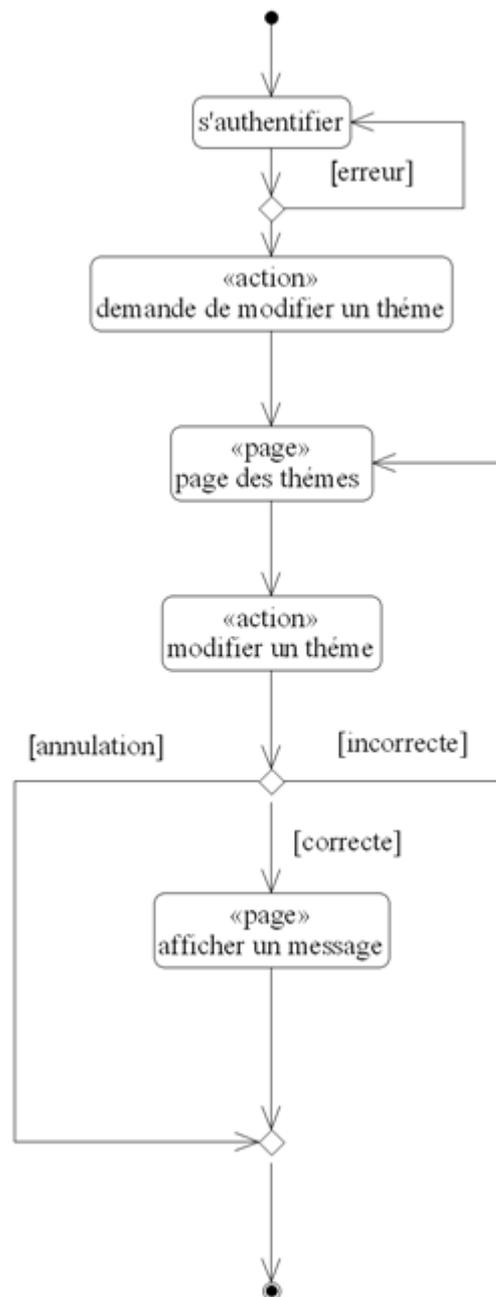


Figure 34 : Gérer les thèmes

5.4. Gérer les comptes

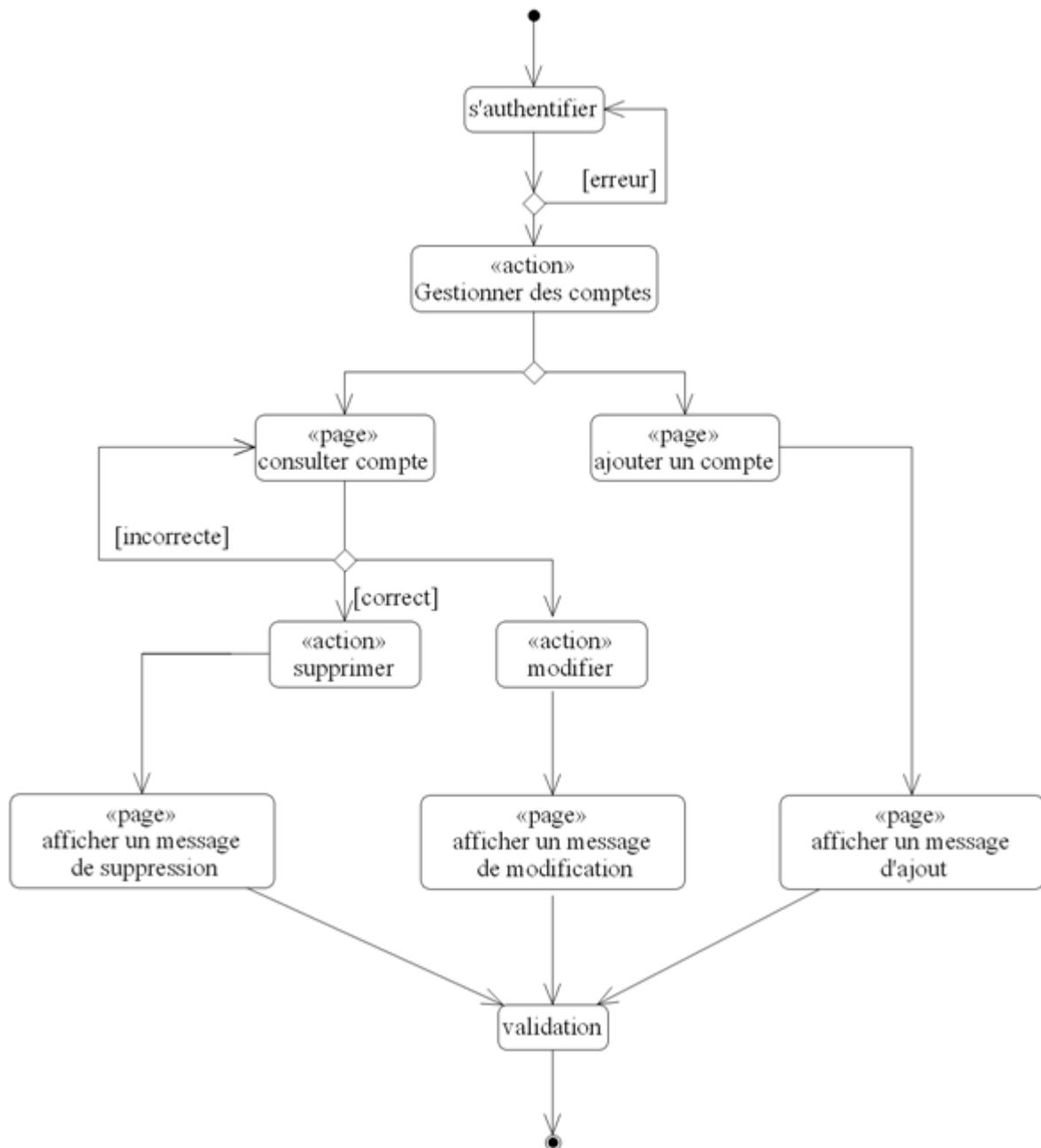


Figure 35 : Gérer les comptes

5.5. Demande de participation

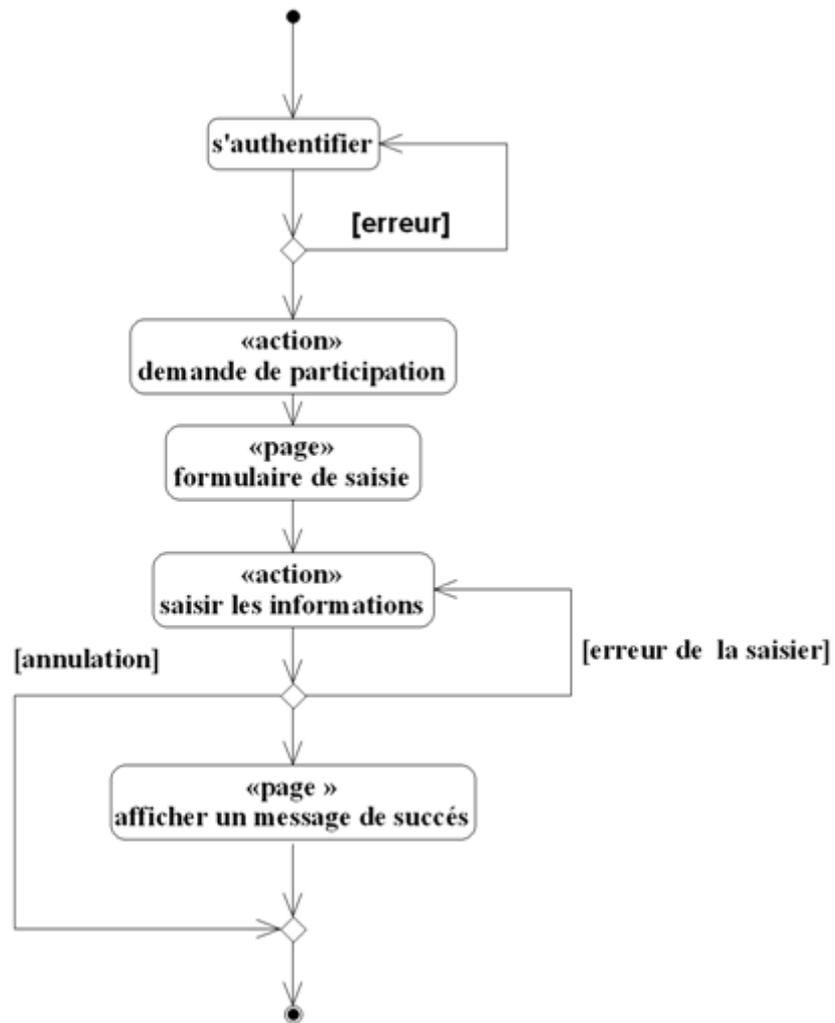


Figure 36 : Demande de participation

5.6. Gérer les membres

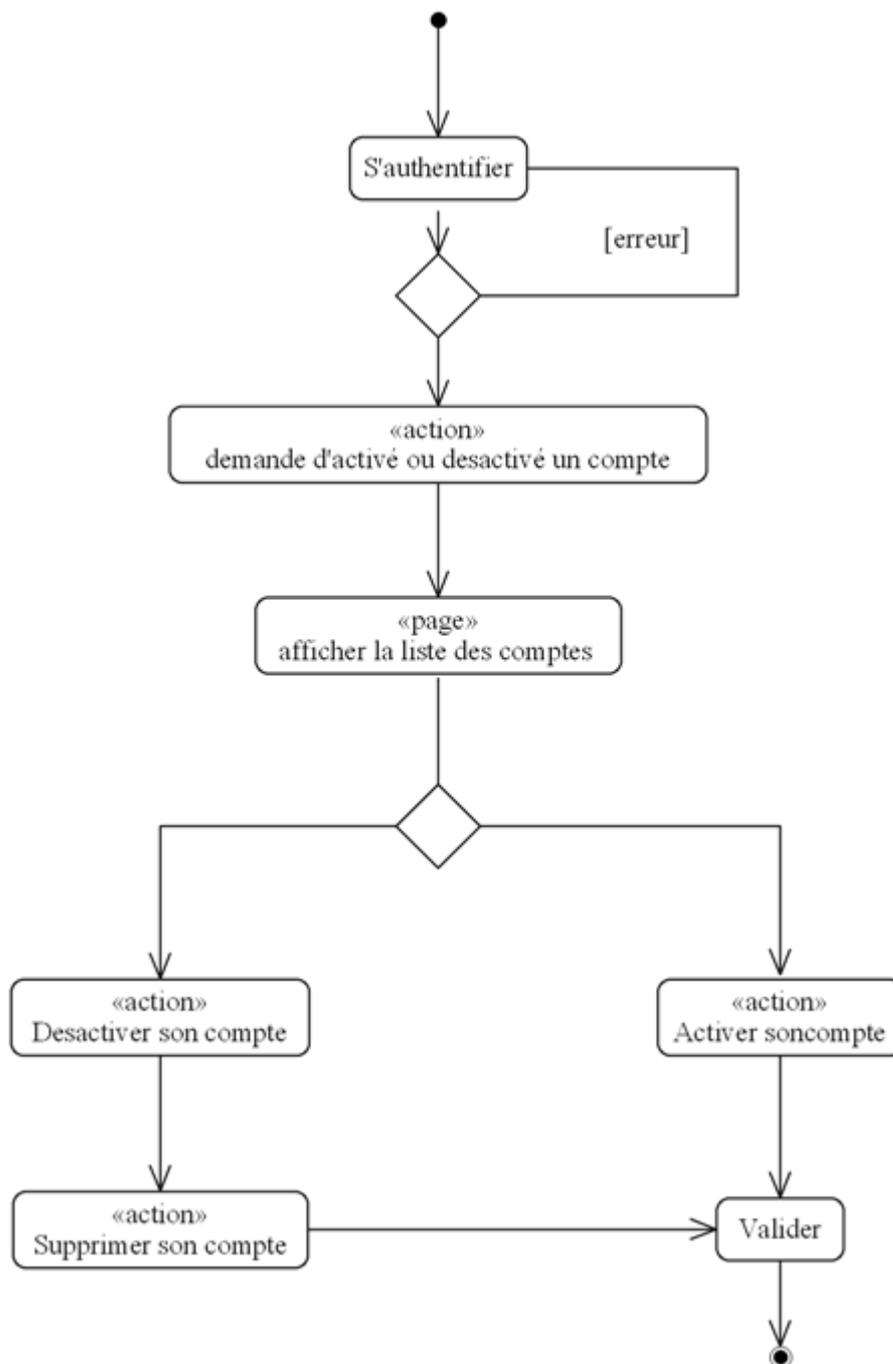


Figure 37 : Gérer les membres

6. Diagramme de classe

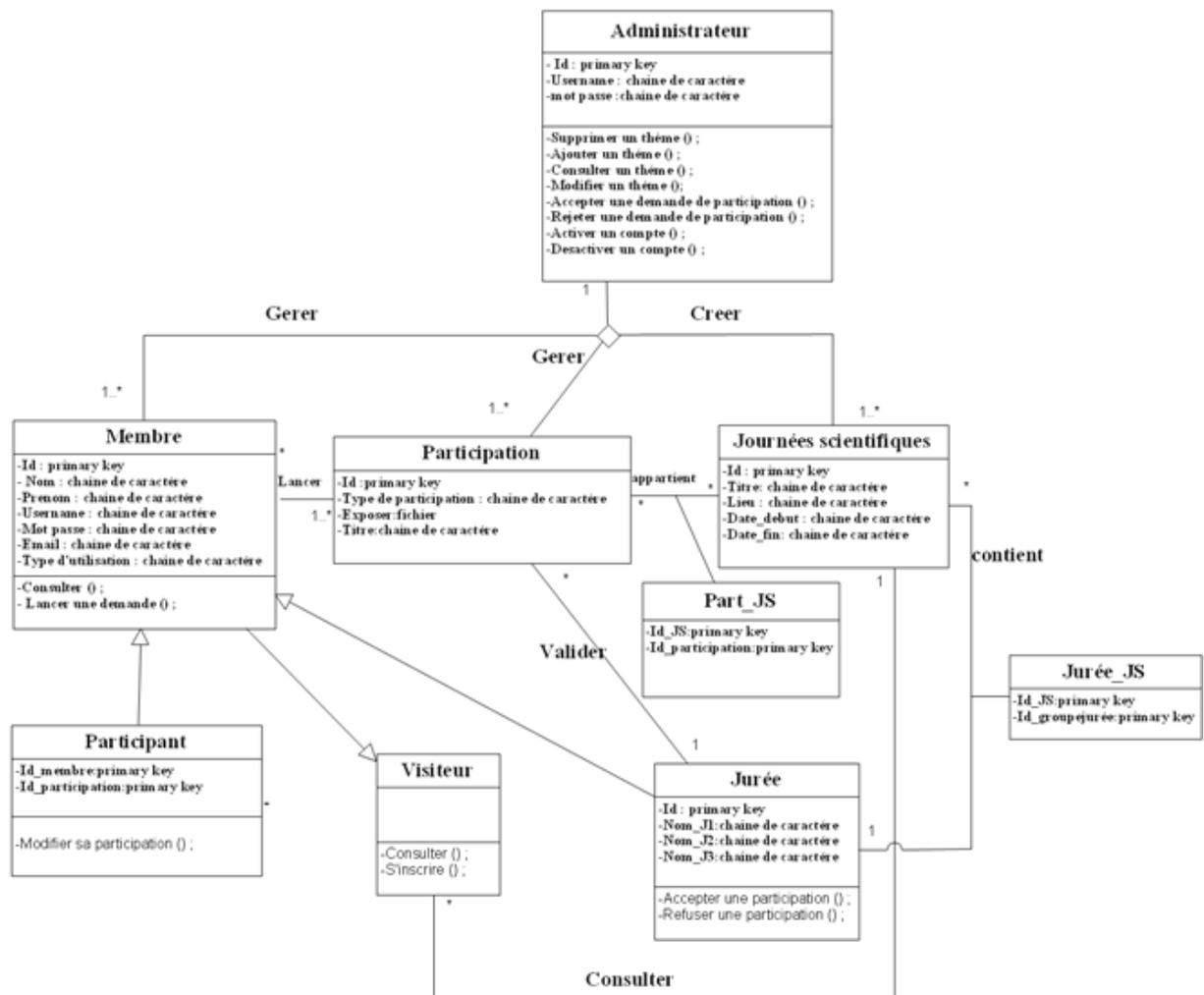


Figure 38 : Diagramme de classe

Conclusion

Dans ce chapitre, nous avons utilisé le langage de modélisation UML pour les applications web, et la planification des informations d'entrée nécessaire, l'analyse nous a permis de recueillir les besoins fonctionnels, de les modéliser, et de spécifier leur réalisation grâce aux diagrammes de cas d'utilisation, d'activité et de séquence. Le chapitre suivant nous présentons la réalisation de site.

Chapitre IV

Implémentation de site

Introduction

La réalisation du site consiste à créer des pages (statiques ou dynamiques) ainsi que des éléments graphiques (images, animation, ...) d'une part, et la création de la base de données d'autre part. Nous allons présenter dans ce chapitre la partie utile de notre projet.

1. Les langages utilisés

1.1. HTML

HTML « Hyper Texte Mark up Langage » est un langage permettant de décrire les différents composants d'un document : définir des titres, construire des tableaux, mettre en forme des textes. Le HTML est un langage simple rédigé sous forme de texte et qui demande un simple éditeur de texte. Le lien Hypertexte est un des principaux atouts du langage HTML, un lien permettant à l'utilisateur d'accéder rapidement à un autre emplacement du document ou à une autre page sur internet[8].

1.2. CSS

CSS : Cascading Style Sheets, crée par le CSS Working groupe et un système destiné à mettre en forme les contenus de pages Web. La partie CSS d'un document Web se contente de définir les différents styles de textes ou de blocs qui seront utilisés pour la mise en forme tandis que la partie HTML ne contient que le texte encadré de quelques balises[9].

1.3. PHP

Le PHP (hypertexte préprocesseur), est un langage de script généraliste et open source, spécialement conçu pour le développement d'applications web il peut être intégré facilement au html.

```
Un exemple du traditionnel Hello World
<? php
    Echo 'Hello World' ;
?>
ou
<? php
Print ('Hello World');
?>
```

1.4. MySQL

MySQL est un serveur de bases de données relationnelles Open Source. Un serveur de bases de données stocke les données dans des tables séparées plutôt que de tout rassembler dans une seule table. Cela améliore la rapidité et la souplesse de l'ensemble. Les tables sont reliées par des relations définies, qui rendent possible la combinaison de données entre plusieurs tables durant une requête. Le SQL dans "MySQL" signifie "StructuredQueryLanguage» et il est le langage standard pour les traitements de bases de données.

1.5. Conception de la base de données

Définition technique d'une base de données : désigne un fichier ou un groupe de fichiers contenant des données réelles. Ces informations sont accessibles au moyen d'un ensemble de programmes appelé SGBD, qui sont presque tous relationnels.

2. Les logiciels utilisés

2.1. WampServer

WampServer est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un Serveur externe) des scripts PHP.

WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi qu'une administration pour les deux bases SQL PhpMyAdmin et SQLiteManager.

2.2. Dreamweaver :

Dreamweaver est un éditeur WYSIWYG (what you see is what you get = ce que vous voyez est ce que vous obtenez) destiné à la conception, au codage et au développement des sites et des pages web. Quel que soit l'environnement de travail utilisé, Dreamweaver propose des outils qui nous aideront à créer des applications web. Ce logiciel est édité par Macromedia. L'éditeur de page html "Dreamweaver" appartient au monde de l'édition professionnelle [10].

3. Description des tables :

Dans notre application nous avons créé 6 tables dans la base de données et ses tables sont :

3.1. Table de membre (users) :

id	username	nom	prenom	password	email	date	pays	universite	signup_date	adress	sexe	telephone	etat
1	Admin	malika	hani	azerty	hani.pol@yahoo.com	12/05/1978	Tunnis		1362760117	citÃ© des 130 logots	FEMME	0765554321	actif
2	imane	mezzahi	ilham	azerty	zidhbcn@yahoo.ber	imane	Tunnis	Annaba	1362760450	citÃ© des 300 logots	FEMME	0781045501	jure
4	adill	merabet	adil	poiuyt	merabet.adil@gmail.com	22/08/1985	AlgÃ©rie	bÃ©jaya	1364049044	Ã jjel	homme	0770102200	
6	mouhamed	laalou	mouhamed	aaaaaa	mido_cnep@live.fr	22/08/1985	Tunnis	bÃ©jaya	1365975076	citÃ© des 500 logots	homme	0765554333	
8	moufida	farida	narjah	aaaqqq	moufidaTTTT@live.fr	03/04/1989	France	biskra	1365976797	citÃ© des 130 logots	FEMME	0669873567	non actif
7	hicham	torchi	housam	aqwaqw	hichamcnep@live.fr	12/11/1989	Algerie	Annaba	1366486665	wxcccvcvbn	HOMME	0979654321	jure

Tableau 12 :Table de membre

3.2. Table de journée :

← T →				id	titre	lieu	date_debut	date_fin	id_juree			
<input type="checkbox"/>		Modifier		Copier		Effacer	1	azerty	s	12/11/3	12/11/3	5
<input type="checkbox"/>		Modifier		Copier		Effacer	2	informatique	milano	12/11/3	2013-04-03	6
<input type="checkbox"/>		Modifier		Copier		Effacer	3	application	mila	12-12-2013	17-12-2013	7
<input type="checkbox"/>		Modifier		Copier		Effacer	4	site web	milano	12/11/3	2015-03-04	0
<input type="checkbox"/>		Modifier		Copier		Effacer	5	site web dynamique	manchester	2013-05-16	2013-06-05	8
<input type="checkbox"/>		Modifier		Copier		Effacer	6	site web dynamique	manchester	2013-05-16	2013-06-05	0
<input type="checkbox"/>		Modifier		Copier		Effacer	7	delphi	constantine	2013-05-23	2013-05-25	9

Tableau 13 :Table de journée

3.3. Table de participation :

← T →				id	titre	type_participation	exposer	id_journe	etat	vpariure	userid			
<input type="checkbox"/>		Modifier		Copier		Effacer	1	sss	orale	FtpFileZilla_17.08.2011.15-40.pdf	1	1		
<input type="checkbox"/>		Modifier		Copier		Effacer	2	sss	orale	227.doc	0	valide	1	2
<input type="checkbox"/>		Modifier		Copier		Effacer	3	azerty	orale	227 (1).doc	0	valide	1	3
<input type="checkbox"/>		Modifier		Copier		Effacer	4	oracle	poster	oracle.doc	0	non valide	0	4
<input type="checkbox"/>		Modifier		Copier		Effacer	5	mot	poster	Doc9.doc	2	non valide	0	5
<input type="checkbox"/>		Modifier		Copier		Effacer	6	motde	poster	Doc9.doc	4	non valide	0	6
<input type="checkbox"/>		Modifier		Copier		Effacer	7	ddkh	orale	Doc9.doc	1	valide	1	7

Tableau 14 :Table de participation

3.4. Table de jurée :

← T →				id	Nom_j1	Nom_j2	Nom_j3			
<input type="checkbox"/>		Modifier		Copier		Effacer	1	farida	benabd	farida
<input type="checkbox"/>		Modifier		Copier		Effacer	2	farida	benabd	ser
<input type="checkbox"/>		Modifier		Copier		Effacer	3	farida	benabd	ser
<input type="checkbox"/>		Modifier		Copier		Effacer	5	farida	benabd	ser
<input type="checkbox"/>		Modifier		Copier		Effacer	6	ser	farida	benabd
<input type="checkbox"/>		Modifier		Copier		Effacer	7	ss	benabd	dd

Tableau 15 :Table de jurée

4. Interface de notre application :

4.1. Page d'accueil :

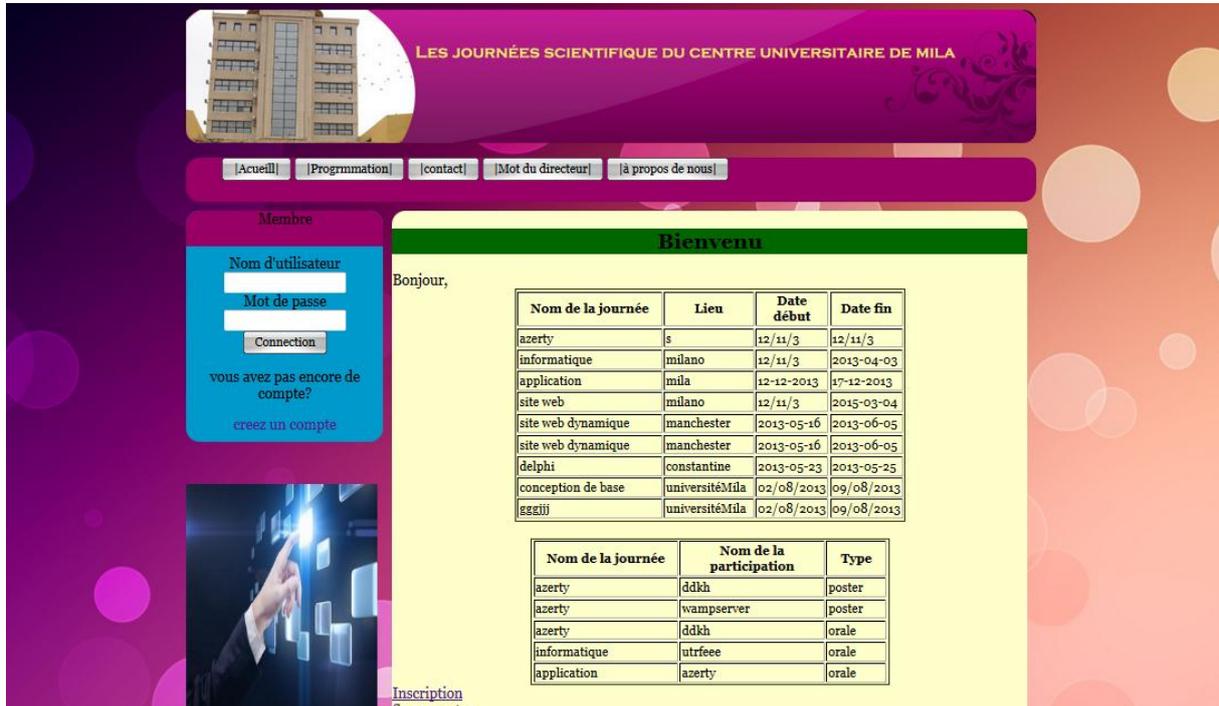


Figure 39 : Page d'accueil.

4.2. Connexion :



Figure 40 : Connexion

4.3. Page d'inscription :

LES JOURNÉES SCIENTIFIQUE DU CENTRE UNIVERSITAIRE DE MILA

[Accueil] [Programmation] [contact] [Mot du directeur] [à propos de nous]

Membre

Nom d'utilisateur

 Mot de passe

vous avez pas encore de compte?
 creez un compte

Veuillez remplir ce formulaire pour vous inscrire:

username

 Nom

 Preom

 Sexe
 HOMME FEMME
 Mot de passe(6 caractères min.)

 Mot de passe(vérification)

 Email

 Date de naissance

 Pays
 Arabie saoudite
 Université
 Université de Biskra, Algérie
 Adress

 Telephone

Figure 41 : Page d'inscription.

4.4. Contact

Contact INFO

Responsable du projet JS 2013

Frédéric BENHAMOU, vice-président du conseil scientifique de centre universitaire de Mila
 Responsable@centre-univ-mila.dz

Organisation des JS 2013

Isabelle AUGEREAU, chargée d'événementiel de centre Universitaire de Mila
 journees.scientifiques@centre-univ-mila.dz
 Tél.213(0)31-57-01-23

Relations presse des JS 2013

Cécile ESTRADÉ, attachée de centre Universitaire de Mila
 Presse@centre-univ-mila.dz
 Tél.213(0)31-57-01-23

Écrivez-nous !

Nom:

Adresse Email:

Votre message:

Figure 42 : contact.

5. Les différentes opérations

5.1. L'administrateur

5.1.1. Page d'accueil de l'administrateur :

LES JOURNÉES SCIENTIFIQUE DU CENTRE UNIVERSITAIRE DE MILA

[Accueil] [Participant] [Programmation] [contact] [Mot du directeur] [à propos de nous]

Membre

Bonjour Admin

Se déconnecter

Modifier mes informations personnelles

Espace d'administration

- * Gestion des membres
- * Ajouter une journées
- * gestion des participants
- * affecter des jurées

Bienvenu

Bonjour Admin,

Nom de la journée	Lieu	Date début	Date fin
azerty	s	12/11/3	12/11/3
informatique	milano	12/11/3	2013-04-03
application	mila	12-12-2013	17-12-2013
site web	milano	12/11/3	2015-03-04
site web dynamique	manchester	2013-05-16	2013-06-05
site web dynamique	manchester	2013-05-16	2013-06-05
delphi	constantine	2013-05-23	2013-05-25
conception de base	universitéMila	02/08/2013	09/08/2013
ggzjjj	universitéMila	02/08/2013	09/08/2013

Nom de la journée	Nom de la participation	Type
azerty	ddkh	poster
azerty	wampserver	poster
azerty	ddkh	orale
informatique	utrftee	orale
application	azerty	orale

Figure 43 : Page d'accueil d'administrateur.

5.1.2. Page de gestion des participations :

LES JOURNÉES SCIENTIFIQUE DU CENTRE UNIVERSITAIRE DE MILA

[Accueil] [Participant] [Programmation] [contact] [Mot du directeur] [à propos de nous]

Membre

Bonjour Admin

Se déconnecter

Modifier mes informations personnelles

Espace d'administration

- * Gestion des membres
- * Ajouter une journées
- * gestion des participants
- * affecter des jurées

Voici la liste des participations:

code	Nom de participation	type	nom	prénom	fichier
9	webmaster	poster	baouche	zina	
6	motde	poster	laalou	mouhamed	
4	oracle	poster	merabet	adil	
16	bhh	orale	mezzahi	mounb	
17	bhh	orale	ferrat	radia	
14	ffff	orale	bellahsan	mouhamed	
11	azerty	orale	askri	lamia	
10	wampserver	poster	samir	amir	
8	ddkh	poster	farida	narjah	
7	ddkh	orale	torchi	bousam	
2	sss	orale	mezzahi	ilham	
1	sss	orale	malika	hani	

Figure 44 : Page de gestion des participations.

5.1.3. Page pour affectée des jurées :

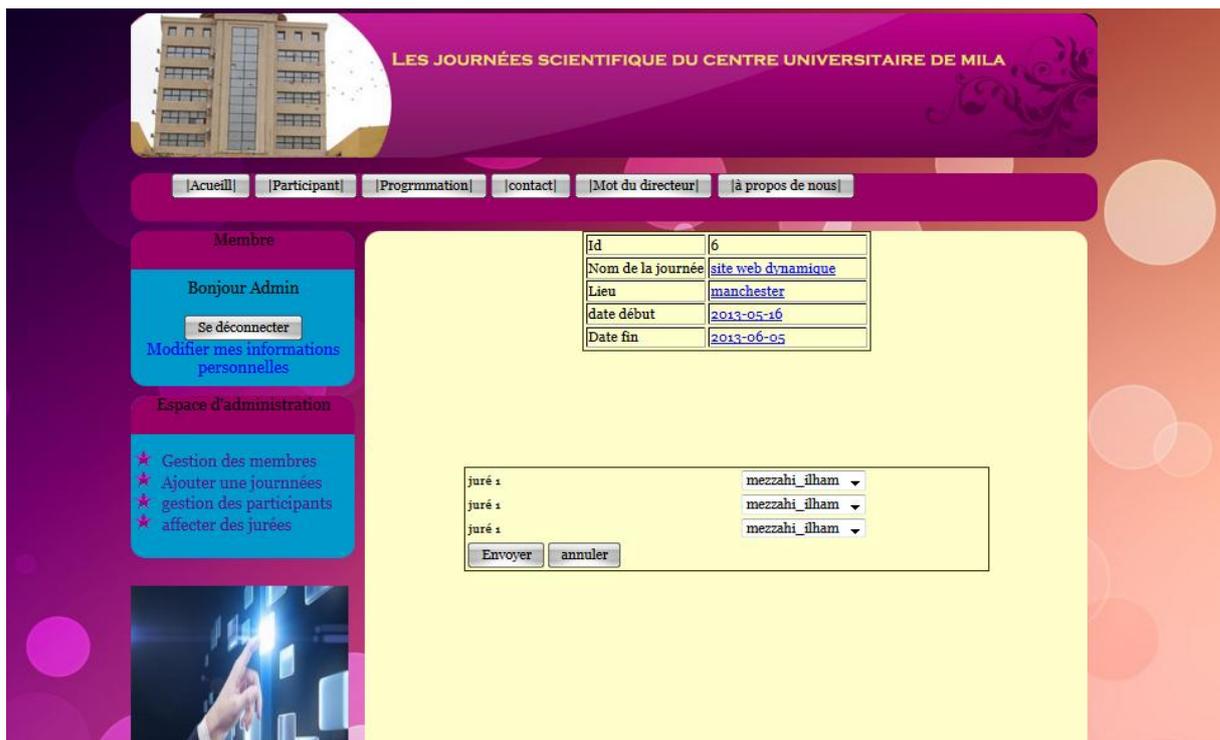


Figure 45 : Page pour affectée des jurées.

5.2. Membre :

5.2.1. Page de participation :

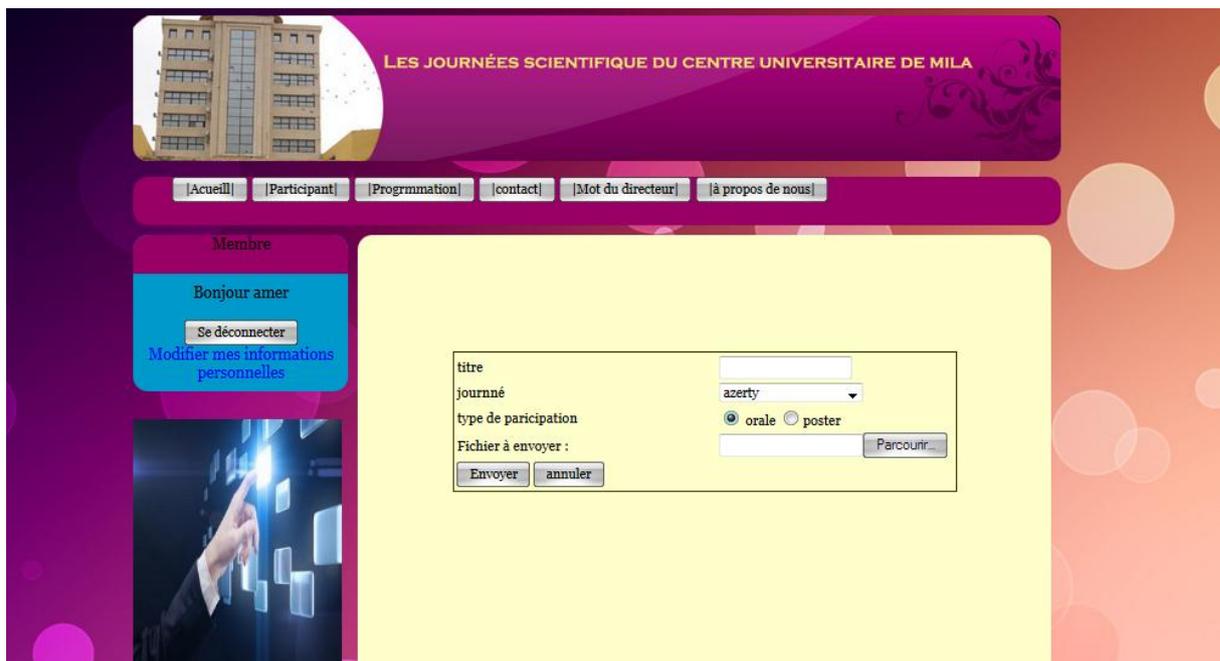


Figure 46 : Page de participation.

5.3. Jurée :

5.3.1. Page d'accueil de jurée :



Figure 47 : Page d'accueil de jurée.

5.3.2. Page de gestion des participations :



Figure 48 : Page de gestion des participations.

Conclusion :

Dans ce chapitre, nous avons proposé la réalisation de notre site web de gestion des journées scientifiques en précisant surtout les pages des différentes fonctionnalités importante du site. Nous avons essayé de le simplifier pour une meilleure utilisation.

Conclusion générale

Conclusion générale

Le travail présenté dans ce mémoire, se situe dans le contexte de développement des sites web dynamique au centre universitaire de Mila dans le but faciliter les taches de gestion des journées scientifiques au centre.

Ce travail consiste à concevoir et réaliser un site webdynamique pour la gestion et le suivi des journées scientifiques en ligne.Nous avons réussi à développer ce site web qui permetaux visiteurs de lancer une demande de participation une journée scientifique choisie. Etpour la réalisation nous avons suivi une méthode conduite par les cas d'utilisation et composée de trois étapes : Analyse, conception et réalisation.

Dans l'analyse nous avons spécifié plusieurs cas d'utilisations. Et puis la conception des différents diagrammes. Et en fin dans la réalisation nous avons développé notre site web en utilisant : PHP,HTML EtMySQL (pour la gestion de la base de données).Le site développé est entièrement opérationnel concernant les fonctionnalités citées dans ce mémoire.

Références bibliographiques

Références bibliographiques

Sites web

[1] www.wikipédia.org

[2] www.siteduzero.com

[3] www.memoireonline.com

[10] http://www.wikipedia.org/wiki/Adobe_Dreamweaver

Ouvrages

[5]. Historique d'UML

[6]. Livre UML 2 , Auteur Laurent Audibert

[7]. UML 2 Modéliser une application web , Auteur Pascal Roques

[8]. Grand livre HTML, Daniel Koch, Oliver Kurten, Florian Harm

1er edition 2000.

Mémoires de fin d'étude :

[9]. Conception et réalisations d'un site web dynamique pour l'enseignant à distance.