**REPUBLIQUE DU SENEGAL**



**UNIVERSITE CHEIKH ANTA DIOP DE DAKAR**



**ECOLE SUPERIEURE POLYTECHNIQUE**

***DEPARTEMENT GENIE INFORMATIQUE***

**MEMOIRE DE FIN DE CYCLE**

**Pour l’obtention du :**

DIPLOME UNIVERSITAIRE DE TECHNOLOGIE (DUT) - DIPLOME SUPERIEUR DE TECHNOLOGIE (DST)

**SUJET :**

***DEVELOPPEMENT D’UN SITE WEB POUR L’AMICALE DU PERSONNEL DE L’ESP (APESUP)***

**Lieu de stage**: Département Génie Informatique (DGI)

**Période stage: 04/06/2018 – 20/07/2018**

|  |  |  |
| --- | --- | --- |
| ***Présenté et soutenu par*** | ***Professeur encadrant*** | ***Maître de stage*** |
| *Khady MBAYE (DUT)*  *Mamadou Demba BA (DST)* | *Professeur Ibrahima FALL* | *M. Boubacar FATY* |

***DEDICACES***

Khady MBAYE

Au nom d’Allah le très Miséricordieux

Dieureudieuf Serigne Ababacar SY

Je dédie ce travail à :

**A mon père Assane MBAYE** et à **ma mère Ndéye Fatou FAYE**

Vous avez su être pour moi des parents exemplaires qui m’ont toujours entouré d’un grand amour, qui s’évertuent à me guider vers le droit chemin, qui m’ont fourni une éducation de très bonne qualité et qui, en toute circonstance, ne cessent de me rappeler ce qu’est le respect de soi-même et surtout le respect envers les autres.

Je vous dédie ce travail du fond du cœur en espérant que vous pourrez, un jour, être fiers de votre très chère fille.

**A mes très chers frères et sœurs Yague Mar, Moussa, les jumeaux et Ndickou MBAYE**

**A mon meilleur ami Papa Alioune BA**

**A ma grand-mère Fatou Niang BA**

Je ne saurais vous faire part de toute l’estime et de toute l’admiration que je vous porte.

Vous êtes des personnes formidables, chaleureuses et vous n’avez cessé de m’offrir attention et protection, choses lesquelles ont été pour moi une grande source de motivation. Je vous en suis très reconnaissante.

MAMADOU DEMBA BA

Au nom d’Allah le très Miséricordieux

Je dédie ce travail à :

**A mon père Demba Laba BA** et à **ma mère Aminata BARRO**

Vous avez su être pour moi des parents exemplaires qui m’ont toujours entouré d’un grand amour, qui s’évertuent à me guider vers le droit chemin, qui m’ont fourni une éducation de très bonne qualité et qui, en toute circonstance, ne cessent de me rappeler ce qu’est le respect de soi-même et surtout le respect envers les autres.

Je vous dédie ce travail du fond du cœur en espérant que vous pourrez, un jour, être fiers de moi.

**A mes frères et sœurs**

**A mes amis**

**A toute la promotion DSTI2 2017-2018**

**A tous ceux qui m’ont soutenu de près ou de loin**

Je ne saurais vous faire part de toute l’estime et de toute l’admiration que je vous porte.

Vous êtes des personnes formidables, chaleureuses et vous n’avez cessé de m’offrir attention et protection, choses lesquelles ont été pour moi une grande source de motivation. Je vous en suis très reconnaissant.

***REMERCIEMENTS***

Nous remercions tout d’abord ALLAH le tout puissant qui nous a permis de mener à bout ce travail qui a été soumis à notre étude. Ensuite nous adressons nos remerciements les plus chaleureux à:

* Nos parents respectifs qui ne cessent de nous accompagner dans toutes nos activités
* Nos frères et sœurs
* Notre encadreur Professeur Ibrahima FALL et notre maitre de stage Monsieur Boubacar FATY, de par leurs disponibilités et leurs conseils qu’ils nous donnent sans pour autant se lasser
* La famille SALL de Thiaroy, mon oncle Ibrahima FAYE, ma Badiène Awa MBAYE
* Ma grande sœur Marième MBAYE
* Nos camarades de classe
* Nos parrains et marraines
* Nos vifs remerciements vont également à l’endroit des membres du jury pour l’intérêt qu’ils ont portés à notre recherche en acceptant d’examiner notre travail et de l’enrichir par leurs propositions.
* L’ensemble des personnes ayant contribué de près ou de loin à la réalisation de ce document.

***Avant-propos***

L’Ecole Supérieure Polytechnique est une école de formation professionnelle de renommée internationale sous la tutelle du Ministère de l’Education nationale et rattachée à l’Université Cheikh Anta Diop de Dakar. Elle a été créée le 24 Novembre 1994 suite aux recommandations de la concertation nationale sur l’enseignement supérieur qui s’est tenue d’Avril à Août 1993 et qui avait préconisé la reconstruction des écoles d’enseignement technologique de l’UCAD.

L’ESP regroupe deux centres qui sont, l’une basée à Dakar dans l’enceinte de l’université Cheikh Anta Diop (Ex ENSUT (Ecole nationale supérieure universitaire de technologie)) et l’autre à Thiès qui a fait irruption suite à la fusion de l’ESP et de l’EPT (Ecole Polytechnique de Thiès).

Elle compte en son sein cinq(6) départements :

Le département génie chimique et biologie appliquée, le département génie civil, le département génie électrique, le département de Gestion (Ex ISG (Institut supérieur de gestion) Dakar), le département génie informatique et le département génie mécanique.

L’ESP propose les formations suivantes :

### DUT (Diplôme universitaire de technologie);

### DST (Diplôme supérieur de technologie);

### DIT (Diplôme d’ingénieur technologue);

### DIC (Diplôme d’ingénieur de conception);

### DSC (Diplôme supérieur comptable);

### DGC (Diplôme de gestion et de comptabilité);

### DESCAF (Diplôme d’études supérieures en commerce, administration et finance);

### DSECG (Diplôme supérieure d’études comptables et de gestion);

### LP (Licences professionnelles);

### MP (Master Professionnel);

### Formation à la carte etc.

### Dans le cadre du développement de la recherche, l’ESP héberge divers laboratoires répartis dans les domaines suivants :

### CIFRES (Centre internationale de formation et de recherche en énergie solaire);

### LEPM (Laboratoire d’électrochimie et des procédés membranaires);

### LEA (Laboratoire d’énergétique appliquée);

### LER (Laboratoire des énergies renouvelables);

### LERG (Laboratoire d’enseignement et de recherche en géomatique);

### LIMBI (Laboratoire d’imagerie médicale et bio-informatique);

### LIRT (Laboratoire d’informatique, réseaux et télécoms);

### LMAGI (Laboratoire de microbiologie appliquée et génie industriel);

### LPAO-SF (Laboratoire de physique de l’atmosphère et de l’océan – Siméon Fongang);

### LTI (Laboratoire de traitement de l’information);

### CRED (Centre de recherche entreprise et développement) etc.

***Table des matières***

INTRODUCTION

**Chapitre 1 : Présentation générale**

[I. PRESENTATION DU DGI/ESP 12](#_Toc521094856)

[I.1 Missions 12](#_Toc521094862)

[I.2 Formations proposées 13](#_Toc521094867)

[II. PRESENTATION DU SUJET 13](#_Toc521094875)

[II.1 Contexte du sujet 13](#_Toc521094876)

[II.2 Problématique 13](#_Toc521094877)

[II.3 Objectifs 14](#_Toc521094878)

**Chapitre 2 : Analyse des besoins**

[I. PRESENTATION DU LANGAGE UML 16](#_Toc521094879)

[II. IDENTIFICATION DES BESOINS 19](#_Toc521094881)

[III. SPECIFICATION DES FONCTIONNALITES 20](#_Toc521094882)

[III.1 Les fonctionnalités de notre système 20](#_Toc521094883)

[III.2 Acteurs principaux et les fonctionnalités associées 20](#_Toc521094884)

[III.2.1 Pour l’internaute 20](#_Toc521094885)

[III.2.2 Pour le bureau 23](#_Toc521094886)

[III.2.3 Pour le membre](#_Toc521094886) 30

[IV. Entités métiers et leurs relations de notre système 34](#_Toc521094888)

Chapitre 3 : Eléments de conception et de réalisation de la solution

[I. Structure de la solution et technologies utilisés 37](#_Toc521094889)

[I.1 Structure de la solution 37](#_Toc521094890)

[I.1.1 Notion d’application web 37](#_Toc521094890)

I.1.2 Architecture client/serveur 37

I.2 Technologies utilisées 38

[II. Environnement de travail 41](#_Toc521094891)

[III. Présentation des résultats 42](#_Toc521094892)

CONCLUSION

WEBOGRAPHIE

***Table des figures***

[Figure 1 : Historique du langage UML 17](#_Toc521138872)

[Figure 2 : Exemples de diagramme UML 18](#_Toc521138873)

[Figure 3 : La hiérarchie des diagrammes UML 2.0 sous forme d'un diagramme de classes 19](#_Toc521138874)

[Figure 4 : Fonctionnalités pour l’internaute 21](#_Toc521138875)

[Figure 5 : Description graphique de la fonctionnalité « Voir Partenaire » 22](#_Toc521138876)

[Figure 6 : Description graphique de la fonctionnalité « Contacter Bureau » 23](#_Toc521138877)

[Figure 7 : Fonctionnalités pour le bureau 23](#_Toc521138878)

[Figure 8 : Description textuelle de la fonctionnalité « S’authentifier » 25](#_Toc521138879)

[Figure 9 : Description graphique de la fonctionnalité « Convoquer Réunion » 26](#_Toc521138880)

[Figure 10 : Description graphique de la fonctionnalité « Ajouter Adhérent » 28](#_Toc521138881)

[Figure 11 : Description graphique de la fonctionnalité « Enregistrer Ressource » 30](#_Toc521138882)

[Figure 12 : Fonctionnalités pour le membre 30](#_Toc521138883)

[Figure 13 : Description graphique de la fonctionnalité « Voir convocation réunion » 31](#_Toc521138884)

[Figure 14 : Description graphique de la fonctionnalité « Voir PV » 32](#_Toc521138885)

[Figure 15 : Description graphique de la fonctionnalité « Voir état cotisation » 33](#_Toc521138886)

[Figure 16 : Entités métiers et leurs relations 35](#_Toc521138887)

[Figure 17 : Exemple d’architecture client/serveur 38](#_Toc521138888)

[Figure 18 : Page accueil 43](#_Toc521138889)

[Figure 19 : Page de connexion 43](#_Toc521138890)

[Figure 20 : Page accueil du bureau 45](#_Toc521138891)

[Figure 21 : Page enregistrer cotisation 46](#_Toc521138892)

[Figure 22 : Page ajouter membre 47](#_Toc521138893)

[Figure 23 : Page publier PV 47](#_Toc521138894)

[Figure 24 : Page voir état cotisation 48](#_Toc521138895)

***INTRODUCTION***

Ce rapport décrit le stage effectué dans le Département Génie Informatique (DGI) de l’Ecole Supérieure Polytechnique de Dakar dans la période du 04 juin au 20 juillet 2018. Ce stage se situe dans le cadre de la formation de fin de cycle du Diplôme Universitaire de Technologie (DUT).

L’objectif de ce stage était de développer un système dynamique pour l’Amicale du Personnel de l’Ecole Supérieure Polytechnique (APESUP) de Dakar.

En effet, en mettant en place un système permettant la gestion de l’APESUP, les difficultés que rencontre le bureau pour partager les informations et les activités de l’amicale aux autres membres s’amélioreront. Outre cet aspect, le système permettra une meilleure communication entre les différents utilisateurs qui s’y trouvent.

Dans le but de mener à bien notre étude, nous allons décomposer ce rapport de stage en trois parties :

* Présentation générale : comprenant la présentation du lieu de stage (DGI), la description du sujet, sa problématique et son objectif.
* Analyse des besoins : décrivant la collecte d’informations sur les attentes des utilisateurs finaux du système de gestion de l’APESUP, la présentation d’UML, la spécification des différentes fonctionnalités du système, la spécification des entités statiques et leurs relations.
* Eléments de conception et de réalisation de la solution : comprenant la structure de notre solution, les outils et technologies utilisés pour développer notre système, notre environnement de travail et la présentation des résultats attendus.

**CHAPITRE**

**1**

**PRESENTATION GENERALE**

# PRESENTATION DU DGI/ESP

L’Ecole supérieure polytechnique a pour vocation de former pour les entreprises des techniciens supérieurs, des masters, des licences professionnelles et des ingénieurs. Les dispositions statuaires de l’établissement et plus particulièrement du département génie informatique (DGI) prévoient un stage obligatoire d’un mois et quinze (15) jours pour l’obtention du DUT ou DST suivi d’une présentation d’un rapport de stage.

Le Département Génie Informatique, structure au sein de l’Ecole Supérieure Polytechnique de Dakar, comprend plusieurs niveaux dont 5 pour la formation payée par l’Etat et 5 pour la formation payée par les tiers. Or chaque niveau regroupe deux options à savoir l’informatique et les réseaux et télécommunications.

### L’enseignement, réparti en deux modules principaux rassemblant les matières de spécialisation dites informatiques et les matières générales proposées par le Département Génie Informatique, est :

### Fondamental pour acquérir des connaissances, des concepts de base et des méthodes de travail

### Appliqué pour faciliter l’apprentissage de ces concepts et déployer des savoir-faire personnels

### Evolutif pour intégrer les progrès technologiques et les exigences du monde professionnel

### Ouvert pour développer les facultés de communication indispensables aux informaticiens dans l’exercice de leur métier

## I.1 Missions

### Le DGI a pour mission de :

### Former des Techniciens Supérieurs, des Assistants Ingénieurs, des Ingénieurs, des masters et des licences en informatique et en réseaux et télécommunications.

### Mener des activités de recherche dans les domaines susmentionnés visant au perfectionnement permanent, à l’adaptation et à la participation, à l’évolution scientifique et technologique.

### Procéder à des expertises dans le cadre de la formation à l’intention des entreprises privées ou publiques.

## I.2 Formations proposées

### Il propose les formations suivantes :

### DUT (Diplôme Universitaire de Technologie);

### DIC (Diplôme d’Ingénieur de Conception);

### DIT (Diplôme d’ingénieur Technologue);

### DST (Diplôme Supérieur de Technologie);

### Licences professionnelles en Systèmes Réseaux et Télécoms et Génie logiciel;

### Master professionnel en Système Réseaux et Télécoms et Génie logiciel.

# II. PRESENTATION DU SUJET

Ce sujet soumis à notre analyse partira d’une observation rudement menée. Ainsi nous présenterons tout d’abord le contexte du sujet, suivi de sa problématique et enfin les objectifs visés.

## II.1 Contexte du sujet

L’Ecole Supérieure Polytechnique, plus connu sous l’acronyme ESP est un établissement de formation professionnelle faisant partie intégrante de l’Université Cheikh Anta Diop de Dakar (UCAD).

Elle regroupe en son sein deux catégories de personnels : le Personnel d’Enseignement et de Recherche (PER) et le Personnel Administratif, Technique et de Service (PATS). Les agents du PER et du PATS se sont regroupés autour d’une amicale dénommée l’Amicale du Personnel de L’Ecole Supérieure Polytechnique (APESUP). Cette amicale est apolitique et n’a aucune coloration religieuse ou syndicale.

Elle a été créée en mars 2016 et compte jusqu’à la date du 30 Avril 2018, 173 membres.

L’APESUP organise des activités qui se déroulent chaque année et à travers ces activités, l’amicale cherche à développer les liens interpersonnels entre ses membres ainsi que l’esprit de fraternité et de solidarité. L’amicale dispose des ressources matérielles et financières. Ces ressources financières proviennent des cotisations mensuelles par membre et des activités à but lucratif.

## II.2 Problématique

Les membres du personnel plus particulièrement les contractuels sont souvent confrontés à des problèmes d’accès aux informations concernant l’Amicale du personnel de l’ESP. En effet, pour adhérer les agents sont obligés de se rapprocher du bureau pour avoir des renseignements sur les modalités d’adhésion. Il arrive aussi au bureau d’avoir des difficultés à partager les informations de l’amicale. En effet, la diffusion des informations se fait par envoi d’emails et des messages textes (sms). Parfois, certains membres ne reçoivent pas ces informations du fait qu’ils ne consultent pas régulièrement leurs comptes émails et aussi d’autres qui changent de numéro de téléphone sans en informer le bureau. Aussi, il arrive que certains membres ne disposent pas de comptes émails pour recevoir les procès-verbaux de réunion et les comptes rendu des assemblées générales de réunion (AG).

## II.3 Objectifs

Etant donné les différentes tâches que fait face le bureau de l’APESUP dans la gestion de cette structure, notre objectif principal sera la mise en place d’un système de gestion de l’Amicale du Personnel de l’Ecole Supérieure Polytechnique (APESUP).

Ce système devra permettre l’accès et le partage des informations de l’amicale. Il doit aussi permettre à chaque membre de consulter les informations le concernan

**CHAPITRE**

**2**

**ANALYSE DES BESOINS**

1. **PRESENTATION DU LANGAGE UML**

Le **Langage de Modélisation Unifié**, de l'anglais Unified ModelingLanguage (**UML**), est un [langage](https://fr.wikipedia.org/wiki/Langage) de troisième génération normalisé par l’**OMG** (Object Management Group) en début 1997. Il est conçu pour fournir une méthode normalisée, pour visualiser la conception d'un système en fonction de ses méthodes objets avec lesquelles il a été construit.

UML est un moyen d'exprimer des modèles objet en faisant abstraction de leur implémentation, c'est-à-dire que le modèle fourni par UML est valable pour n'importe quel langage de programmation. Il est couramment utilisé en [développement logiciel](https://fr.wikipedia.org/wiki/D%C3%A9veloppement_logiciel) et en [conception orientée objet](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet).

Cependant, il faudra noter qu’UML est le résultat de la fusion de précédents langages de modélisation objet comme : Booch, OMT, OOSE, qui s’imposaient dans le domaine de la modélisation dans les années 90.

Le langage UML ne préconise aucune démarche, ce n’est donc pas une méthode. De très nombreuses entreprises de renommé ont adopté UML et participe encore aujourd’hui à son développement.

UML est un langage qui s'appuie sur un méta modèle, un modèle de plus haut niveau qui définit les éléments d'UML (les concepts utilisables) et leur sémantique (leur signification et leur mode d'utilisation).

Le méta modèle permet de se placer à un niveau d'abstraction supérieur car il est étudié pour être plus générique que le modèle qu'il permet de construire.

Le méta-modèle d'UML en fait un langage formel possédant les caractéristiques suivantes :

* Un langage sans ambiguïtés
* Un langage universel pouvant servir de support pour tout langage orienté objet
* Un moyen de définir la structure d'un programme
* Une représentation visuelle permettant la communication entre les acteurs d'un même projet
* Une notation graphique simple, compréhensible même par des non informaticiens.

Le méta modèle permet de donner des bases solides et rigoureuses à ce langage graphique, dont les représentations graphiques ne sont là que pour véhiculer des concepts de réalisation.

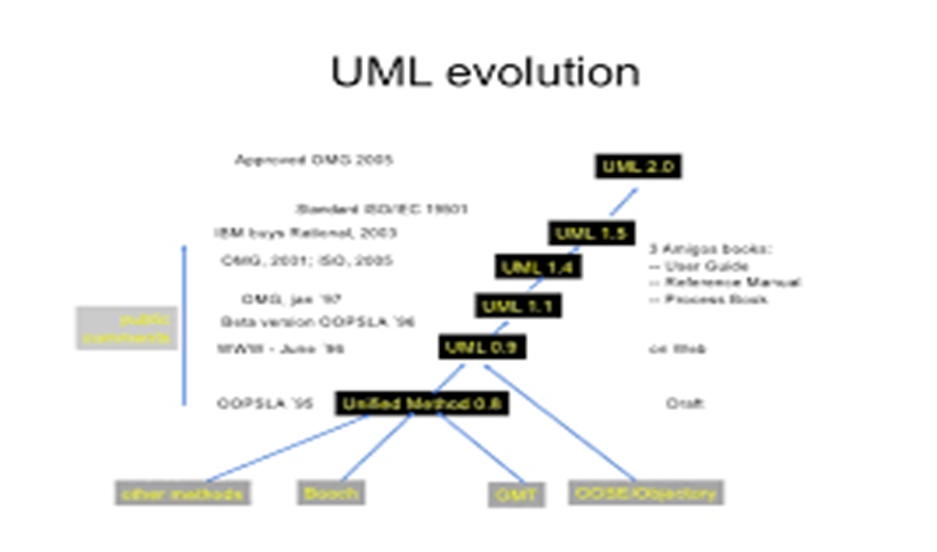


Figure 1 : Historique du langage UML

**Les diagrammes d’UML**

Le méta modèle UML fournit une panoplie d'outils permettant de représenter l'ensemble des éléments du monde objet ainsi que les liens qui les relie. Les versions UML proposent aujourd’hui quatorze diagrammes. Ces quatorze types de diagrammes représentent autant de vues distinctes pour représenter des concepts particuliers du système d'information. Ils se répartissent en deux grandes vues :

**Les vues statiques**, c'est-à-dire représentant le système physiquement :

* Diagrammes d'objets;
* Diagrammes de classes;
* Diagrammes de cas d'utilisation;
* Diagrammes de composants;
* Diagrammes de déploiement;
* Diagramme de paquetages (Package diagram);
* Diagramme de structures composites (Composite structure diagram);
* Diagramme de profils.

**Les vues dynamiques**, montrant le fonctionnement du système :

* Diagrammes de séquence;
* Diagrammes de collaboration;
* Diagrammes d'états-transitions;
* Diagrammes d'activités;
* Diagramme de temps (Timing diagram);
* Diagramme global d'interaction (Interaction overview diagram).

Ces diagrammes, d'une utilité variable selon les cas, ne sont pas nécessairement tous produits à l'occasion d'une modélisation. Les plus utiles pour la maîtrise d'ouvrage sont les diagrammes d'activités, de cas d'utilisation, de classes, d'objets, et de séquence. Les diagrammes de composants, de déploiement et de communication sont surtout utiles pour la maîtrise d'œuvre à qui ils permettent de formaliser les contraintes de la réalisation et la solution technique. Dans la suite nous allons présenter les diagrammes utilisés dans notre modélisation.

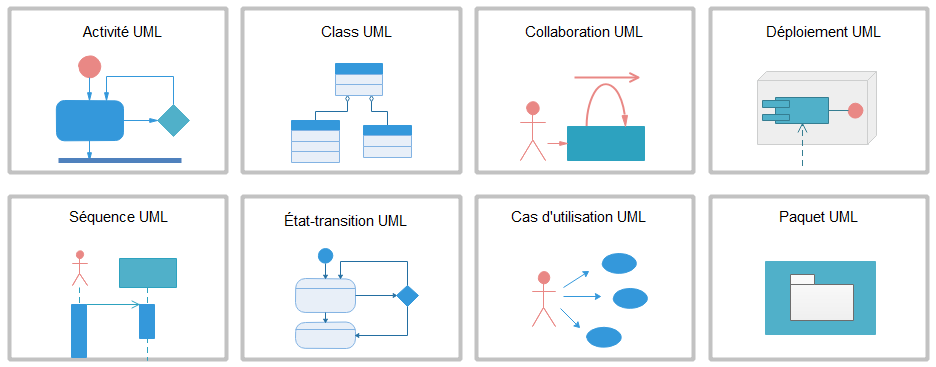


Figure 2 : Exemples de diagramme UML

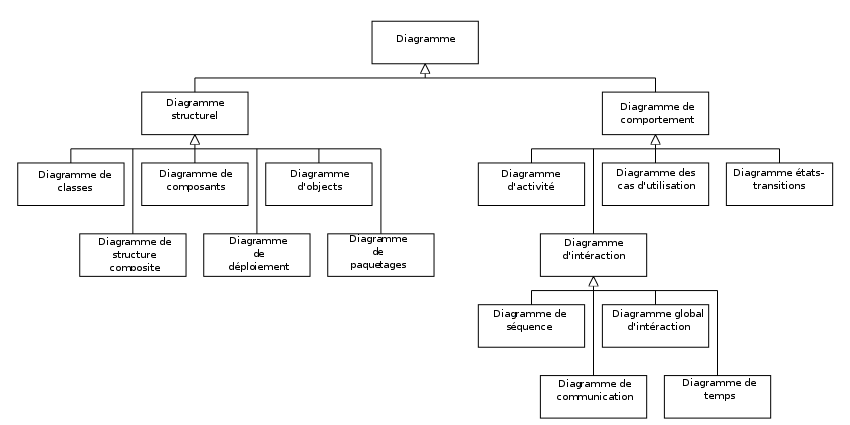


Figure 3 : La hiérarchie des diagrammes UML 2.0 sous forme d'un diagramme de classes

1. IDENTIFICATION DES BESOINS

L’analyse de notre sujet nous amène à définir trois types d’acteurs intervenant sur le système : l’internaute, le bureau et le membre.

**L’internaute**: c’est un acteur qui n’est pas membre de l’amicale. Il peut chercher à avoir des informations sur l’APESUP. Il peut demander à devenir membre ou partenaire de l’amicale en contactant le bureau.

**Le bureau**: il gère l’administration de l’APESUP. Il peut être amené à convoquer des réunions, envoyer des informations, rédiger des comptes rendus et de des procès-verbaux (PV), enregistrer les recettes de l’amicale (don, subvention etc.), ajouter des agents du personnel, enregistrer les cotisations mensuelles des membres.

**Le membre**: cet acteur représente un membre de l’APESUP. Il peut consulter les informations envoyées par le bureau, voir les PV et convocations de réunion, contacter le bureau, voir état cotisation.

1. SPECIFICATION DES FONCTIONNALITES
2. Les fonctionnalités de notre système

Les fonctionnalités de notre système se présentent comme suit :

* **S’authentifier** : c’est un processus permettant au système de s’assurer de la légitimité de la demande d’accès faite par un acteur du système afin d’autoriser l’accès à cette entité à des ressources du système.
* **Diffuser information** : c’est la fonctionnalité qui permet au bureau de partager les informations de l’amicale. Le bureau peut envoyer au membre des convocations de réunion, des PV à l’occasion des AG et des activités de l’amicale.
* **Gérer adhésion** : le bureau peut être amené à ajouter un agent du personnel de l’ESP dans l’amicale. Il peut éventuellement le supprimer de l’amicale.
* **Gérer caisse :** cette fonctionnalité représente la gestion de la comptabilité de l’APESUP. Lebureau se chargera d’enregistrer toutes les ressources (subventions, dons, les cotisations du membre) reçues par l’amicale, les dépenses effectuées durant les activités de l’APESUP.
* **Consulter information :** cette fonctionnalité permet au membre de l’amicale d’accéder aux informations (de voir les PV et convocation de réunion etc.) publiées par le bureau.
* **Contacter bureau :** l’internaute peut joindrele bureaupour avoir des renseignements sur l’APESUP et devenir éventuellement membre ou partenaire de l’amicale en utilisant les coordonnées de l’APESUP.
* **Voir état cotisation**: Le membre de l’APESUP peut voir l’état de ses cotisations en ayant la liste des mois de l’année qu’il n’a pas encore cotisés.
* **Consulter plateforme :** l’internaute peut visiter le système pour voir les réalisations de l’amicale, les événements à venir, voir les partenaires de l’amicale et les actualités de l’amicale.

1. Acteurs principaux et les fonctionnalités associées
2. Pour l’internaute

La figure 4 représente les différentes fonctionnalités du système pour l’internaute sous la forme d’un diagramme de cas d’utilisation. Les tableaux 1 et 2 quant à eux représentent les descriptions textuelles des cas d’utilisation « Voir Partenaire » et « Contacter bureau ». Ils sont complétés par les figures 5 et 6 qui correspondent aux descriptions respectives des mêmes cas d’utilisation sous forme graphique avec l’aide de diagrammes de séquences.



Figure 4 : Fonctionnalités pour l’internaute

**Tableau1 : Description textuelle de la fonctionnalité « Voir Partenaire »**

|  |  |
| --- | --- |
| Titre du cas | **Voir Partenaire** |
| Descriptif et objectif | Ce cas d’utilisation permet à l’internaute (bureau ou membre) de connaître les différents partenaires qui soutiennent l’APESUP. |
| Acteur (s) | L’internaute |
| Précondition | Accéder à la page de l’amicale |
| Scénario nominale | 1. L’internaute demande à accéder à la fonctionnalité « voir partenaires ». 2. Le système lui affiche la liste des partenaires. 3. L’internaute consulte la liste des partenaires. |
| Scénario alternative | - |
| Postcondition | - |

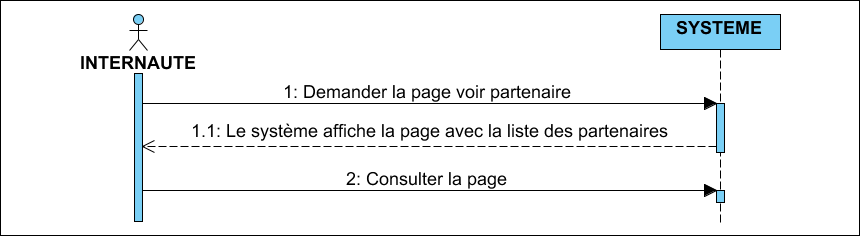


Figure 5 : Description graphique de la fonctionnalité « Voir Partenaire »

**Tableau2 : Description textuelle de la fonctionnalité « Contacter bureau »**

|  |  |
| --- | --- |
| Titre du cas | Contacter bureau |
| Descriptif et objectif | Ce cas d’utilisation permet à l’internaute et au membre de communiquer avec le bureau. Pour l’internaute, il peut contacter le bureau s’il a besoin de nouer un partenariat ou subventionner une activité du bureau. Pour le membre, en cas de besoin d’informations supplémentaires sur les activités de l’APESUP, il peut contacter le bureau pour plus de renseignements. |
| Acteur(s) | Membre, internaute |
| Précondition | - |
| Scénario nominal | 1. L’utilisateur demande la page « contacter bureau ». 2. Le système lui affiche la page avec les coordonnées de l’APESUP (téléphone, mail, page Facebook etc.). 3. L’utilisateur choisi une coordonnée de l’APESUP |
| Scénario alternatif | - |
| Postcondition |  |

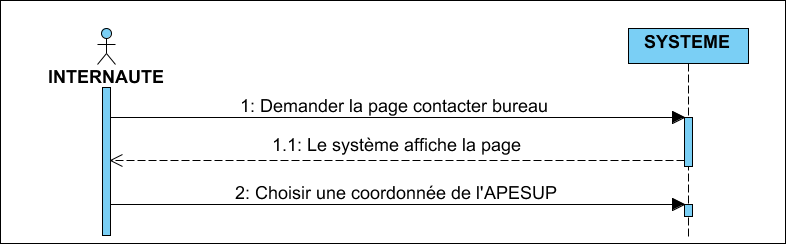


Figure 6 : Description graphique de la fonctionnalité « Contacter Bureau »

III.2.2 Pour le bureau

La figure 7 représente les différentes fonctionnalités du système pourle bureausous la forme d’un diagramme de cas d’utilisation. Les tableaux 3, 4, 5 et 6 quant à eux représentent les descriptions textuelles des cas d’utilisation « S’authentifier », « Convoquer réunion », « Ajouter adhérent », « Enregistrer ressource ». Ils sont complétés par les figures 8, 9, 10 et 11 qui correspondent aux descriptions respectives des mêmes cas d’utilisation sous forme graphique avec l’aide de diagrammes de séquences.

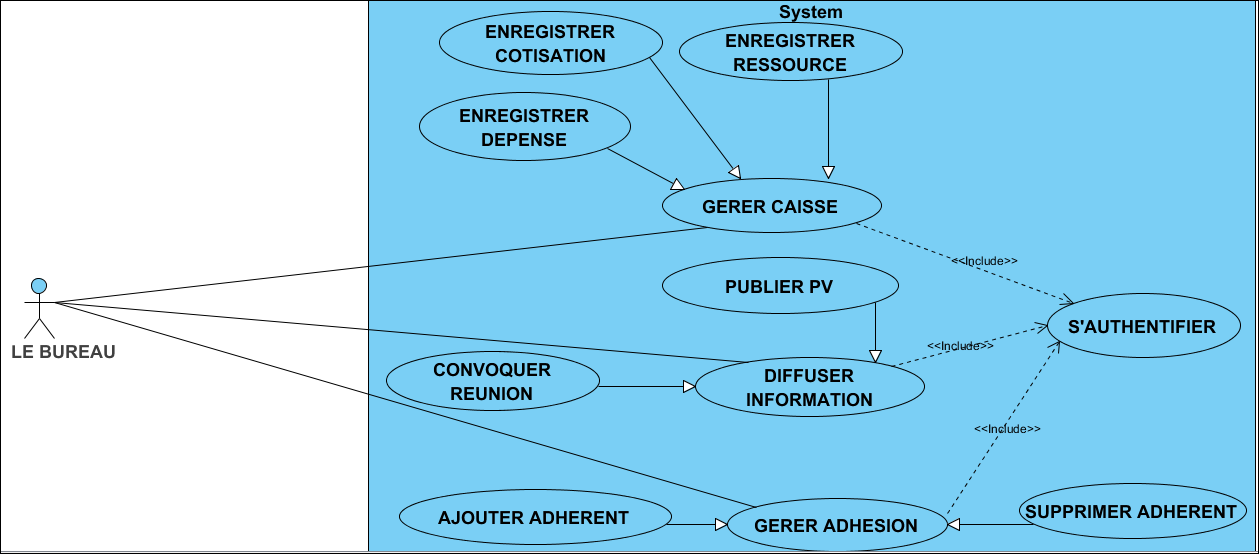


Figure 7 : Fonctionnalités pour le bureau

**Tableau3 : Description textuelle de la fonctionnalité « S’authentifier »**

|  |  |
| --- | --- |
| Titre du cas | S’authentifier |
| Descriptif/objectif | Le site devra procéder à une authentification sécurisée du bureau et du membre afin d'assurer la traçabilité de leurs actions, ainsi que l'intégrité et la non-falsification des informations. |
| Acteur (s) | Le bureau et le membre. |
| Précondition | Accéder à la page d’authentification |
| Scénario nominale | 1. L’utilisateur demande à se connecter ou à accéder à une fonctionnalité. 2. Le système vérifie si l’acteur s’est authentifié. 3. Si oui, le système le connecte ou lui donne un accès à la fonctionnalité considérée. 4. Sinon, le système lui demande ses informations de connexion (email et carte membre). 5. L’acteur entre ses informations de connexion et valide. 6. Le système effectue une vérification des données saisies puis, si elles sont correctes, connecte l’utilisateur ou lui donne accès à la fonctionnalité considérée. |
| Scénario alternatif | * Après l’action 6 : si les données de connexion sont incorrectes, le système affiche un message d’erreur. * L’acteur va reprendre l’authentification au point5. |
| Postcondition | Ouverture espace d’occupation selon acteur |

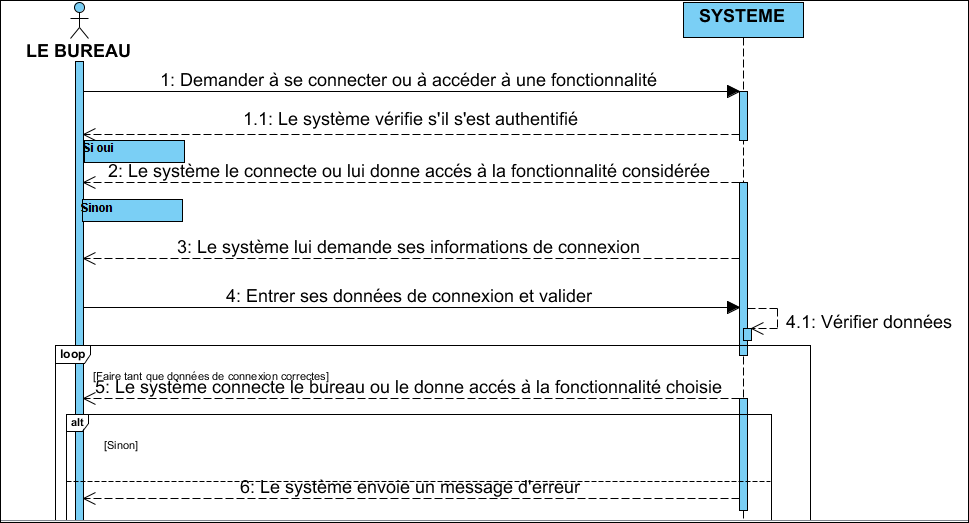


Figure 8 : Description textuelle de la fonctionnalité « S’authentifier »

**Tableau4 : Description textuelle de la fonctionnalité « Convoquer Réunion »**

|  |  |
| --- | --- |
| Titre du cas | **Convoquer Réunion** |
| Descriptif et objectif | Ce cas d’utilisation permet au bureau de l’APESUP d’informer le membre des AG, des activités de l’amicale etc. |
| Acteur (s) | Le bureau |
| Précondition | S’authentifier au préalable |
| Scénario nominal | 1. Le bureau demande à accéder à la fonctionnalité « Convoquer Réunion ». 2. Le système affiche la page. 3. Le bureau saisit les informations (libellé réunion, date et lieu) de la réunion et valide les données. 4. Le système enregistre les informations de la réunion et envoie un message de confirmation |
| Scénario alternatif | - |
| Postcondition | Convocation envoyée!! |

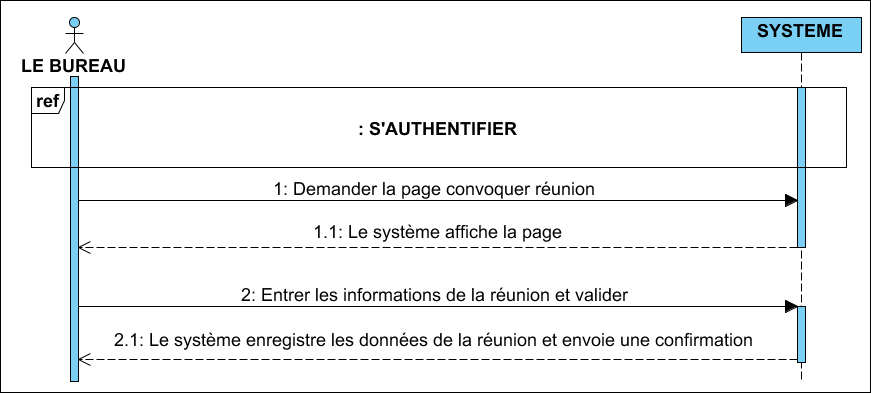


Figure 9 : Description graphique de la fonctionnalité « Convoquer Réunion »

**Tableau5 : Description textuelle de la fonctionnalité « Ajouter Adhérent »**

|  |  |
| --- | --- |
| Titre du cas | **Ajouter Adhérent** |
| Descriptif et objectif | Le bureau se chargera aussi de gérer les adhésions en ajoutant des agents du personnel dans l’amicale. |
| Acteur (s) | Le bureau |
| Précondition | S’authentifier au préalable |
| Scénario nominale | 1. Le bureau demande à ajouter un agent. 2. Le système affiche la page. 3. Le bureau saisit les informations (nom, prénom, mail, carte membre, adresse, structure, téléphone etc.) de l’agent à ajouter et valide. 4. Le système vérifie si les données (mail, carte membre) sont correctes. Si oui, le système enregistre les données et envoie un message de confirmation d’ajout. |
| Scénario alternatif | Lors de la vérification des données, s’il y’a une erreur ou des champs non renseignés, le système envoie un message d’erreur. L’acteur reprend au point3. |
| Postcondition | Agent ajouté!! |

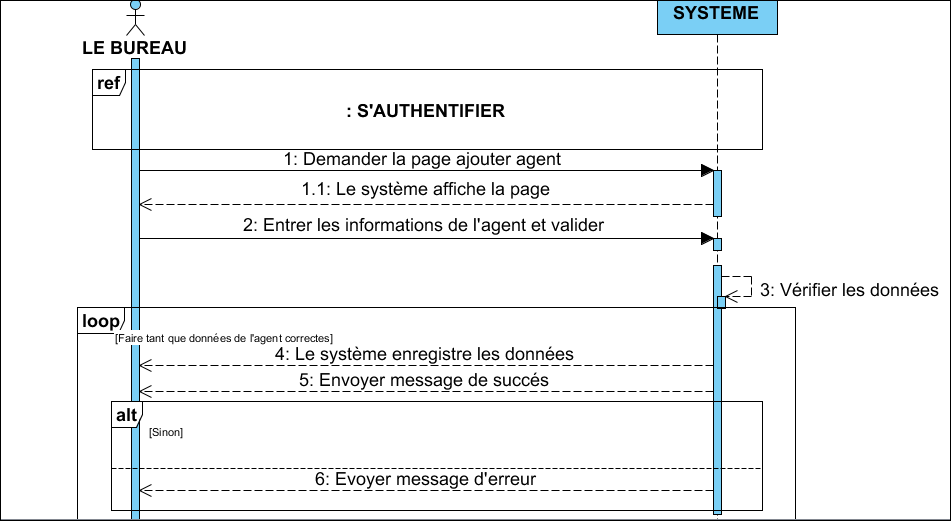


Figure 10 : Description graphique de la fonctionnalité « Ajouter Adhérent »

**Tableau6 : Description textuelle de la fonctionnalité « Enregistrer Ressource »**

|  |  |
| --- | --- |
| Titre du cas | **Enregistrer Ressource** |
| Descriptif/Objectif | Le membre de bureau s’occupe de la gestion de la caisse de l’APESUP. Il enregistre toutes les recettes (les dons, les subventions, les cotisations de membres etc.) de l’amicale. |
| Acteur (s) | Le bureau |
| Précondition | S’authentifier |
| Scénario nominal | 1. Le bureau demande la page enregistrer ressource. 2. Le système affiche la page. 3. Le bureau saisit les informations (libellé ressource, montant, date reçue) concernant la ressource et valide. 4. Le système vérifie si tous les champs sont renseignés. Si oui il enregistre la ressource 5. Le système envoie un message de confirmation au bureau. |
| Scénario alternatif | Lors de la vérification des données, s’il y’a un champ non renseigné, le système envoie un message « Veuillez remplir tous les champs ». L’acteur reprend au point3. |
| Postcondition | Ressource enregistrée!! |

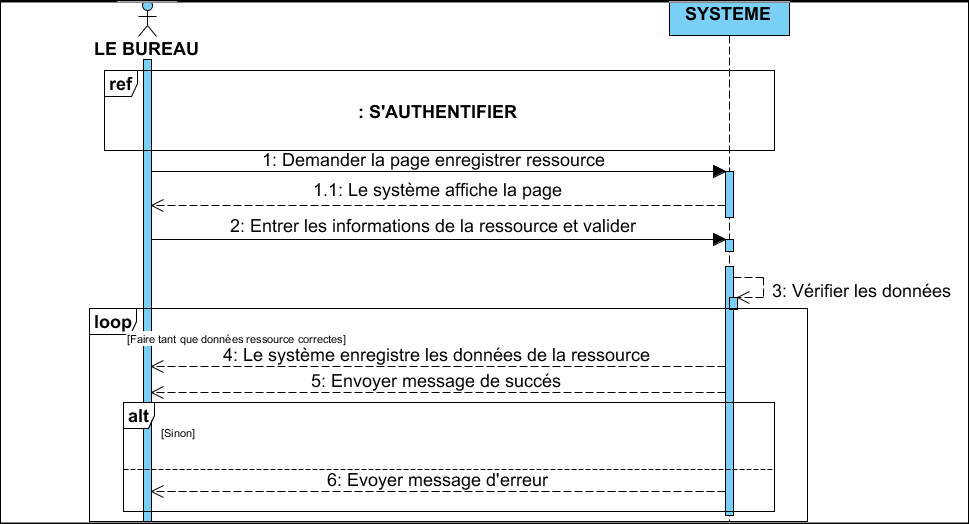


Figure 11 : Description graphique de la fonctionnalité « Enregistrer Ressource »

III.2.3 Pour le membre

La figure 12 représente les différentes fonctionnalités du système pour le membre sous la forme d’un diagramme de cas d’utilisation. Les tableaux 7, 8 et 9 quant à eux représentent les descriptions textuelles des cas d’utilisation « Voir Convocation », « Voir PV » et « Voir état caisse ». Ils sont complétés par les figures 13, 14 et 15 qui correspondent aux descriptions respectives des mêmes cas d’utilisation sous forme graphique avec l’aide de diagrammes de séquences.

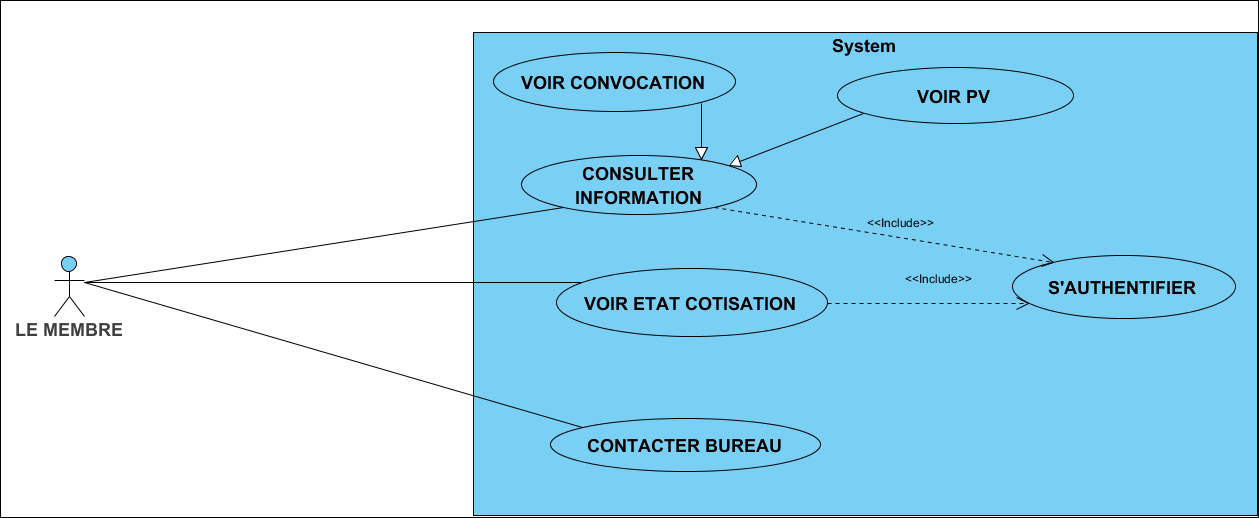


Figure 12 : Fonctionnalités pour le membre

**Tableau7 : Description de la fiche textuelle de la fonctionnalité « Voir convocation»**

|  |  |
| --- | --- |
| Titre du cas | **Voir convocation** |
| Descriptif et objectif | Ce cas d’utilisation permet au membre de l’APESUP de disposer de l’information concernant les AG de réunion, les activités envoyée par le bureau. |
| Acteur(s) | Le Membre |
| Précondition | S’authentifier |
| Scénario nominal | 1. Le membre demande la page voir convocation. 2. Le système lui affiche la page avec une liste des convocations. 3. Le membre consulte la convocation ou télécharge la convocation. |
| Scénario alternatif | - |
| Postcondition | - |

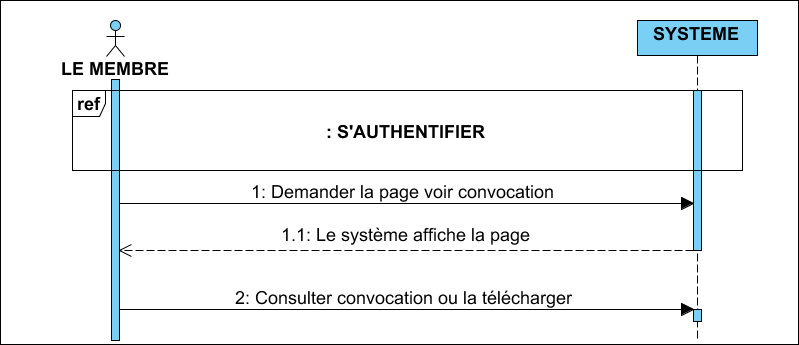


Figure 13 : Description graphique de la fonctionnalité « Voir convocation réunion »

**Tableau8 : Description de la fiche textuelle de la fonctionnalité « Voir PV »**

|  |  |
| --- | --- |
| Titre du cas | **Voir PV** |
| Descriptif et objectif | Avec ce cas d’utilisation, le membre peut consulter les PV des réunions, des AG etc. |
| Acteur(s) | Le Membre |
| Précondition | S’authentifier |
| Scénario nominal | 1. Le membre demande la page voir PV. 2. Le système lui affiche la page. 3. Le membre consulte les PV ou le télécharge. |
| Scénario alternatif | - |
| Postcondition | - |

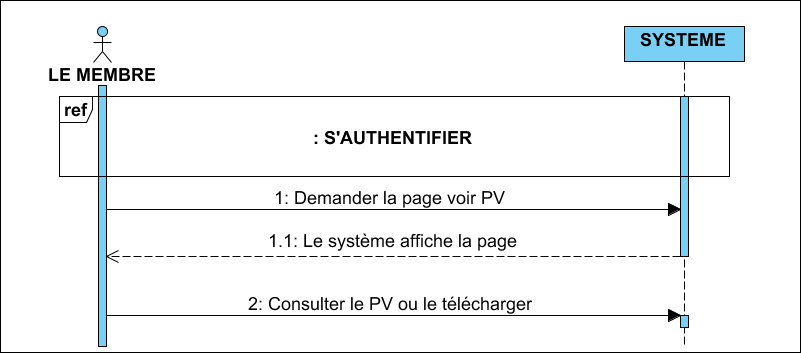


Figure 14 : Description graphique de la fonctionnalité « Voir PV »

**Tableau9 : Description de la fiche textuelle de la fonctionnalité « Voir état caisse »**

|  |  |
| --- | --- |
| Titre du cas | **Voir état caisse** |
| Descriptif et objectif | Ce cas d’utilisation permet au membre de l’APESUP de consulter les états de ses versements. Le membre choisit une année et les mois qu’il n’a pas encore cotisé s’affichent. |
| Acteur(s) | Le Membre |
| Précondition | S’authentifier |
| Scénario nominal | 1. Le membre demande la page voir état cotisation. 2. Le système lui affiche la page. 3. Le membre choisit une année. 4. Le système lui affiche l’ensemble des mois non versés. 5. Le membre consulte la page. |
| Scénario alternatif | - |
| Postcondition | - |

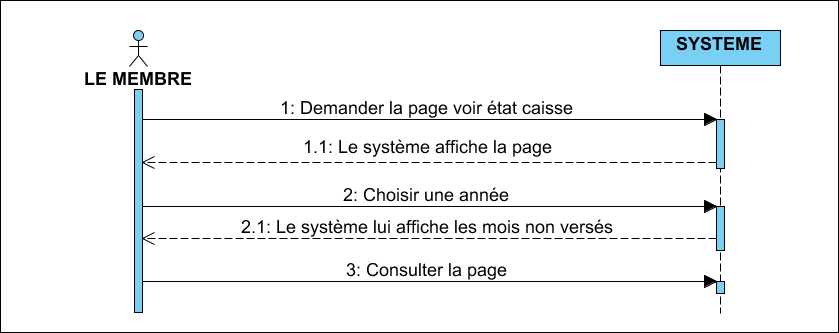


Figure 15 : Description graphique de la fonctionnalité « Voir état cotisation »

1. Entités métiers et leurs relations de notre système

La figure ci-dessous représente les entités métiers de notre système sous la forme d’un diagramme de classe. Ainsi, nous avons huit (8) classes reliées entre elles.

* La classe « Bureau » permet d’enregistrer les différents dirigeants que l’amicale a eu à avoir, la date de début et de fin de leurs mandats.
* La classe « Membre » permet d’enregistrer les agents adhérents de l’amicale avec leurs noms, prénoms, date d’adhésion, téléphone, email, carte membre etc.
* La classe « Partenaire » concerne les partenaires qui soutiennent l’amicale à l’occasion de ses activités. Elle comporte le nom du partenaire, ses coordonnées et son adresse.
* La classe « Ressource » enregistre les différentes recettes (dons, subventions, cotisations etc.) de l’amicale. Elle contient l’identifiant de la ressource, le libellé de la ressource, le montant et la date reçue.
* La classe « Activité » représente les activités que le bureau organise durant chaque année. Les activités peuvent être des réunions, des conférences, des activités à but lucratif, des ndogou durant le ramadan, arbre de noël etc. La date et le lieu sont enregistrés dans la classe « Activité » pour chaque activité que le bureau effectue.
* La classe « Dépense » comptabilise les achats que le bureau effectue durant ses activités. Pour chaque dépense, on enregistre l’identifiant de la dépense, la date, le montant, le libellé de la dépense et l’activité concernée.
* La classe « Structure » représente les différents services dans lesquels les agents du personnel de l’ESP travaillent. Ainsi, pour chaque structure, on donne son identifiant et son nom (département, direction, laboratoire etc.).

Un membre peut faire partie du bureau ou être un membre simple de l’amicale. Il assiste à une ou plusieurs activités de l’amicale. Il appartient à une structure donnée. Il verse une ou plusieurs cotisations.

Un partenaire contacte le bureau. Il peut sponsoriser une ou plusieurs activités de l’amicale.

Le bureau a une ou plusieurs ressources.

Une ressource peut être une cotisation.

Une activité est assurée par aucune ou plusieurs dépenses.

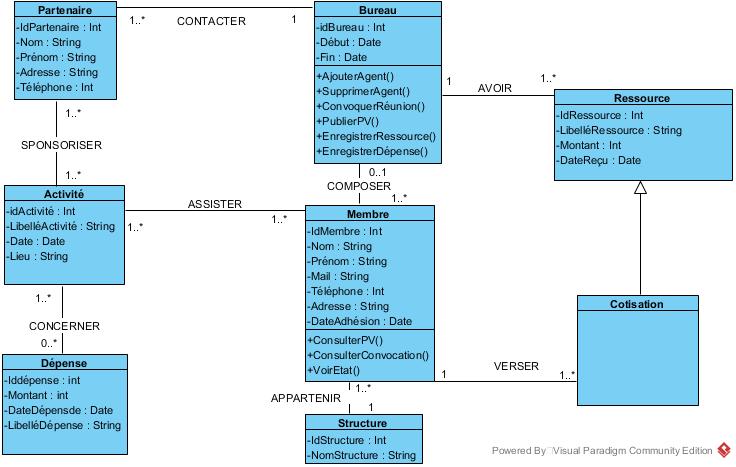


Figure 16 : Entités métiers et leurs relations

**CHAPITRE**

**3**

**ELEMENTS DE CONCEPTION ET DE REALISATION**

La réalisation et la mise en œuvre de la solution seront faites à travers divers technologies que nous présenterons dans les lignes suivantes.

1. **Structure de la solution et technologies utilisés**

**I.1 Structure de la solution**

**I.1.1 Notion d’application web**

Notre choix pour une application web s'explique par le fait que l'annuaire gère une importante quantité de données. Du moment où toute modification sur le serveur est d'ordre général c'est- à-dire que tous les postes clients seront notifiés sur les mises à jour effectuées, il serait donc préférable pour centraliser et mieux contrôler l'accès aux données d'utiliser une application web. Contrairement à un autre type d'application ou toute modification est faite sur chaque poste client, ce qui n'est pas souhaitable car pouvant entrainer des retards ou même des oublis.

Son but ? Offrir une expérience à l’utilisateur et des fonctionnalités équivalentes aux logiciels habituellement installés sur les ordinateurs, en utilisant de technologies de développement identiques à celles employées habituellement dans la création des sites internet.

Une application WEB se divise en deux parties : FRONT et ADMIN. La partie FRONT sera celle vue sur le NET avec accès publique, l’autre partie et comme son nom l’indique concerne les administrateurs de l’application pour gérer le contenu et les différentes fonctionnalités et qui nécessite une identification.

La séparation entre les couches permet un "confort" de développement et surtout de maintenance, avec plus d'expérience le développeur sera capable de localiser rapidement le fichier à modifier.

**I.1.2 Architecture client/serveur**

L'environnement **client-serveur** désigne un mode de communication à travers un [réseau](https://fr.wikipedia.org/wiki/R%C3%A9seau_informatique) entre plusieurs [programmes](https://fr.wikipedia.org/wiki/Programme_(informatique)) : l'un, qualifié de [client](https://fr.wikipedia.org/wiki/Client_(informatique)), envoie des requêtes ; l'autre ou les autres, qualifiés de [serveurs](https://fr.wikipedia.org/wiki/Serveur_informatique), attendent les requêtes des clients et y répondent.

L'interface entre ces deux parties est classiquement le langage SQL. Dans ce type d'architecture on constate une certaine indépendance du client par rapport au serveur. La programmation du client peut s'effectuer sans se préoccuper de la mise en place de la base de données. En particulier, les questions concernant l'administration des données ne concerneront pas le développeur du client.

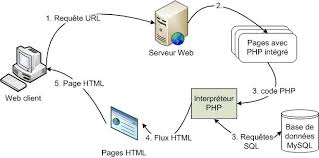
****

Figure 17 : Exemple d’architecture client/serveur

**I.2 Technologies utilisées**

* **Langages de programmation**

Un langage de programmation est une notation conventionnelle destinée à formuler des algorithmes et produire des programmes informatiques qui les appliquent. D'une manière similaire à une langue naturelle, un langage de programmation est composé d'un alphabet, d'un vocabulaire, de règles de grammaire, et de significations. Ils servent de moyens de communication par lesquels le programmeur communique avec l'ordinateur, mais aussi avec d'autres programmeurs; les programmes étant d'ordinaire écrits, lus, compris et modifiés par une communauté. Les langages se différencient de par leur niveau d’utilisabilité, leurs performances, leur extensibilité, leur portabilité et leur pérennité. C’est sur la base de ces critères que notre choix s’est porté sur les langages suivants qui s’exécutent tous du coté client sauf le langage PHP :

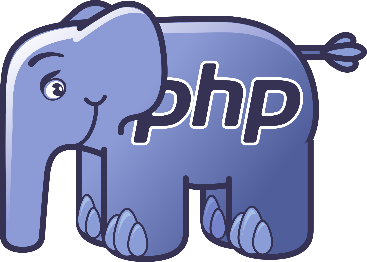
* **HTML et CSS**



**HTML** (HyperText Markup Language): il a fait son apparition dès 1991 lors du lancement du Web. Son rôle est de gérer et organiser le contenu. C'est donc en HTML que vous écrirez ce qui doit être affiché sur la page: du texte, des liens, des images etc. Vous direz par exemple : « Ceci est mon titre, ceci est mon menu, voici le texte principal de la page, voici une image à afficher, etc. ».

**CSS** (Cascading Style Sheets, aussi appelées Feuilles de style) : le rôle du CSS est de gérer l'apparence de la page web (agencement, positionnement, décoration, couleurs, taille du texte, etc.). Ce langage est venu compléter le HTML en 1996. HTML pour le fond et CSS pour la forme.

* **PHP**



**PHP** (HyperText Preprocessor) est un langage interprété (un langage de script) exécuté du côté serveur (comme les scripts CGI, ASP, etc.) et non du côté client (un script écrit en JavaScript ou une applet Java s'exécute sur votre ordinateur). La syntaxe du langage provient de celles du langage C, du Perl et de Java.

Pour contenir la masse d'informations collectées, PHP s'appuie généralement sur un SGBD, généralement MySQL mais aussi SQLite avec PHP5, et sur des serveurs Apache20. PHP, MySQL et Apache forment d'ailleurs le trio ultra dominant sur les serveurs Internet.

Ce langage, qui au début servait pour les traitements des formulaires et l'accès aux bases de données, est d'ores et déjà un langage de Programmation Orientée Objet depuis sa version cinq.

* **JavaScript**



**JavaScript** est un [langage de programmation de](https://fr.wikipedia.org/wiki/Langage_de_programmation) [scripts prin](https://fr.wikipedia.org/wiki/Langage_de_script)cipalement employé dans [les pages web](https://fr.wikipedia.org/wiki/Pages_web) interactives mais aussi pour les serveurs[.](https://fr.wikipedia.org/wiki/JavaScript#cite_note-2) C’est un langage [orienté](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet) objet à [prototype,](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_prototype) c’est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas des [instances de classes,](https://fr.wikipedia.org/wiki/Instance_(programmation)) mais qui sont chacun équipés de [constructeurs](https://fr.wikipedia.org/wiki/Constructeur_(programmation_informatique)) permettant de créer leurs propriétés, et notamment une propriété de prototypage qui permet d’en créer des objets [héritiers](https://fr.wikipedia.org/wiki/H%C3%A9ritage_(informatique)) personnalisés. En outre, [les fonctions](https://fr.wikipedia.org/wiki/Fonction_informatique) sont de[s objets de première classe.](https://fr.wikipedia.org/wiki/Objet_de_premi%C3%A8re_classe)

Le langage a été créé en 1995 par [Brendan Eich](https://fr.wikipedia.org/wiki/Brendan_Eich) (Brendan Eich étant membre du conseil d'administration de la fondation Mozilla à cette époque) pour le compte de [Netscape Communications Corporation.](https://fr.wikipedia.org/wiki/Netscape_Communications_Corporation) Le langage, actuellement à la version 1.8.2, est une implémentation de la troisième version de la norme [ECMA-262](https://fr.wikipedia.org/wiki/ECMAScript) qui intègre également des éléments inspiré[s du langage Python. L](https://fr.wikipedia.org/wiki/Langage_Python)a version 1.8.5 du langage est prévue pour intégrer la cinquième version du standard ECMA.

De plus, le JavaScript est très avantageux car une très grande majorité des visiteurs l’acceptent. Il est relativement simple et facile à apprendre. Accompagné d’une certaine vitesse, le JavaScript est un langage versatile. JavaScript constitue ainsi une technologie vraiment incontournable.

* **JQuery et DataTable JQuery**



**JQuery** est une [bibliothèque](https://fr.wikipedia.org/wiki/Biblioth%C3%A8que_logicielle) [JavaScript](https://fr.wikipedia.org/wiki/JavaScript) [libre](https://fr.wikipedia.org/wiki/Logiciel_libre) et [multiplateforme](https://fr.wikipedia.org/wiki/Multi-plateforme) créée pour faciliter l'écriture de [scripts côté client](https://fr.wikipedia.org/wiki/Client_Side_Scripting) dans le code [HTML](https://fr.wikipedia.org/wiki/HTML) des [pages web](https://fr.wikipedia.org/wiki/Page_web).

**DataTable JQuery** est un widget Tableau jQuery léger construit pour remplacer facilement vos tableaux HTML. Il peut lire et afficher les données de votre tableau HTML, mais il peut aussi afficher des données provenant de diverses sources de données.

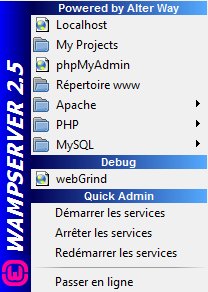
* **BootStrap**



**BootStrap** est un Framework ou boîte à outils. Il embarque des composants CSS, HTML et JavaScript. Il comporte un système de grille simple et efficace pour mettre en ordre l'aspect visuel d'une page web. Il apporte du style pour les boutons, les formulaires, la navigation etc. Il permet ainsi de concevoir un site web rapidement et avec peu de lignes de code ajoutées. Il s’agit d’un Framework coté client ou encore front end (terme anglais).

1. **Environnement de travail**

* **WampServer**

**WampServer** est une plate-forme de développement Web sous Windows. Il vous permet de développer des applications Web dynamiques à l'aide du serveur http Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin et SQLite Manager pour gérer plus facilement vos bases de données.

Son installation est très facile et son utilisation très intuitive permet de le configurer très rapidement (sans toucher aux fichiers de configuration). WampServer vous permet de reproduire fidèlement votre serveur de production. Une fois la base installée, vous avez la possibilité d'ajouter autant de versions d’Apache, MySQL et PHP que vous le souhaitez.

* Apache : Il s’agit d’un serveur http ou serveur web permettant à des clients

d’accéder à des pages web, c’est à dire en réalité des fichiers au format HTML à partir d’un navigateur (aussi appelé browser) installé sur leur ordinateur distant. C’est le serveur le plus répandu dans le web.

* MySQL : Le plus populaire des serveurs de bases de données SQL Open

Source, MySQL est un système de gestion de bases de données (SGBD) robuste et rapide. Il s’exécute sur un serveur sous forme de service ou démon. IL fonctionne sur de nombreuses plateformes, disposant d’un système d’allocation de mémoire très rapide grâce à l’exploitation des threads. Ce SGBD offre des jointures très rapides, exploitant un système de jointures multiples en une seule passe optimisée. Le serveur de bases de données MySQL est très rapide, fiable et facile à utiliser.

* **Sublime Text**



**Sublime Text** est un [éditeur de texte g](https://fr.wikipedia.org/wiki/%C3%89diteur_de_texte)énérique codé en [C++ e](https://fr.wikipedia.org/wiki/C%2B%2B)t [Python,](https://fr.wikipedia.org/wiki/Python_(langage)) disponible sur [Windows,](https://fr.wikipedia.org/wiki/Windows) [Mac e](https://fr.wikipedia.org/wiki/Mac_OS)t [Linux.](https://fr.wikipedia.org/wiki/Linux) Le logiciel a été conçu tout d'abord comme une extension pour [Vim,](https://fr.wikipedia.org/wiki/Vim) riche en fonctionnalités. Depuis la version 2.0, sortie le [26 juin 2012,](https://fr.wikipedia.org/wiki/26_juin) l'éditeur prend en charge quarante-quatre langages de programmation majeurs, tandis que des plugins sont souvent disponibles pour les langages plus rares. Sublime Text intègre la plupart des fonctionnalités de base d'un [éditeur de texte,](https://fr.wikipedia.org/wiki/%C3%89diteur_de_texte) dont la [coloration syntaxique p](https://fr.wikipedia.org/wiki/Coloration_syntaxique)ersonnalisable.

* **VISUAL PARADIGM**



**Visual Paradigm** est unemultiplateforme riche en fonctionnalités puissantes qui dispose de tous les diagrammes UML et des outils DRE essentiellement dans le système et la conception de base de données. Visual Paradigm Community Edition fournit un environnement de modélisation visuelle plus facile à utiliser et intuitif avec un riche ensemble d'[importation et d'exportation](https://www.visual-paradigm.com/features/output-export-import/) des capacités telles que XMI, XML, PDF, JPG.

1. **Présentation des résultats**

La réalisation du système de gestion de l’APESUP est l’un des objectifs principaux de la tâche qui nous a été confiée. Elle a ainsi pris forme et a abouti à un site web dynamique assez sobre et évoquant le sérieux qu’il est sensé inspirer.

Les figures qui vont suivre montreront ce qui a été réalisé.

* La page d’accueil

Elle s'affiche une fois que le site est lancé. La page d’accueil présente une description de l’APESUP. On y trouve la fonctionnalité « Se connecter » qui concerne le bureau et le membre et le menu (Qui sommes-nous?, nous contacter, partenaires, événements, actualités) que l’internaute pourra consulter.



Figure 18 : Page accueil

* La page de connexion

A travers cette fenêtre le bureau et le membre s’authentifie pour accéder respectivement aux fonctionnalités les concernant. Cette étape met en valeur l'aspect sécurité du système : ils renseignent leurs emails et leurs mots de passe (carte membre) et le système vérifie l’existence de ces données (email et carte membre) dans la base de données. Si leurs identifiants sont corrects, le système les connecte, sinon, le système leurs demande de les renseigner à nouveau.

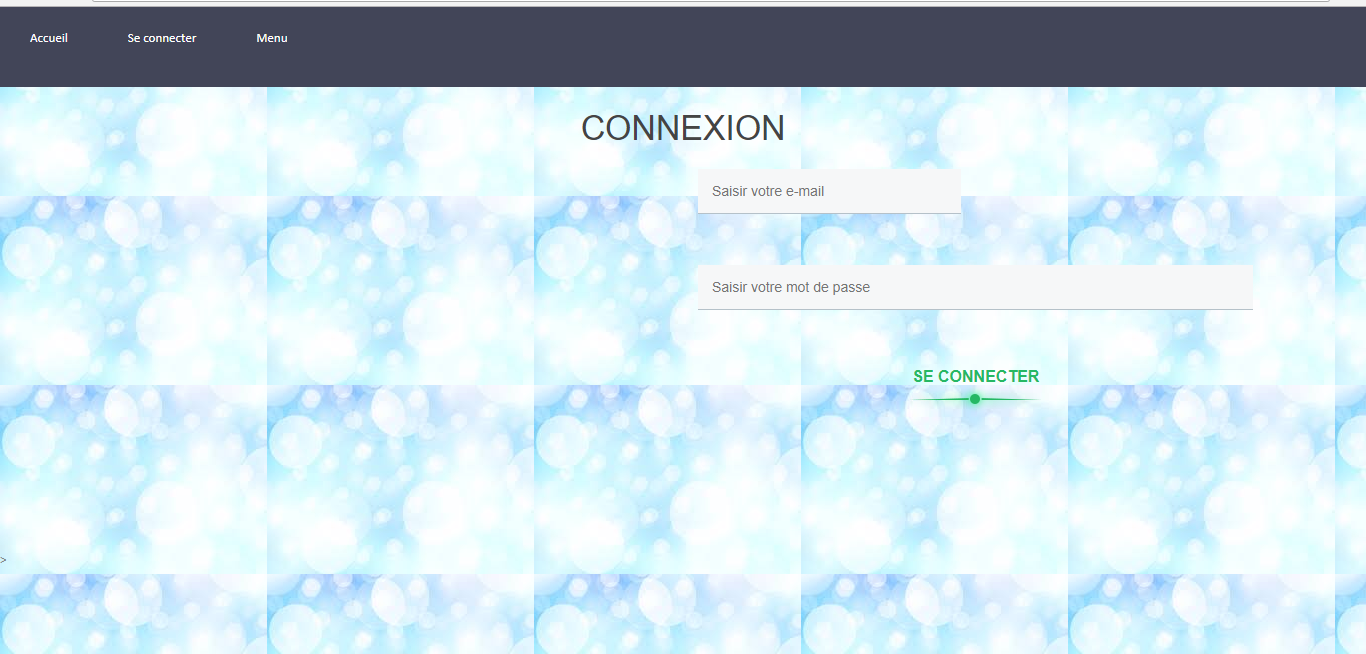
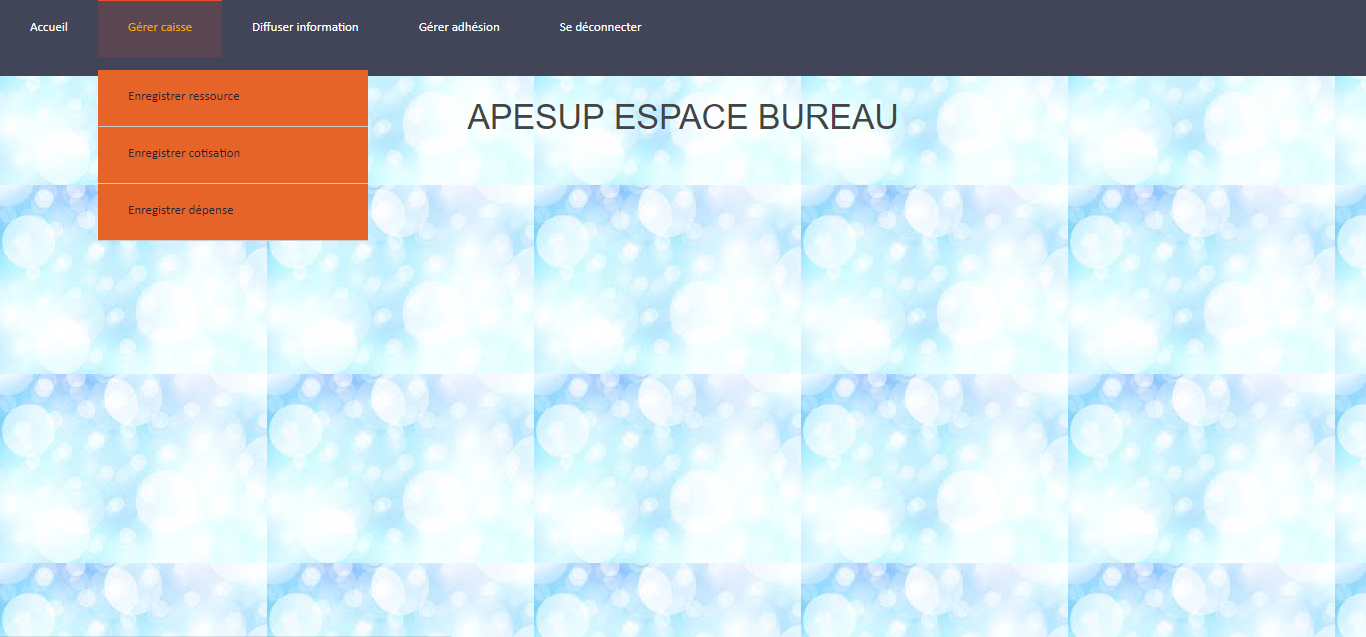
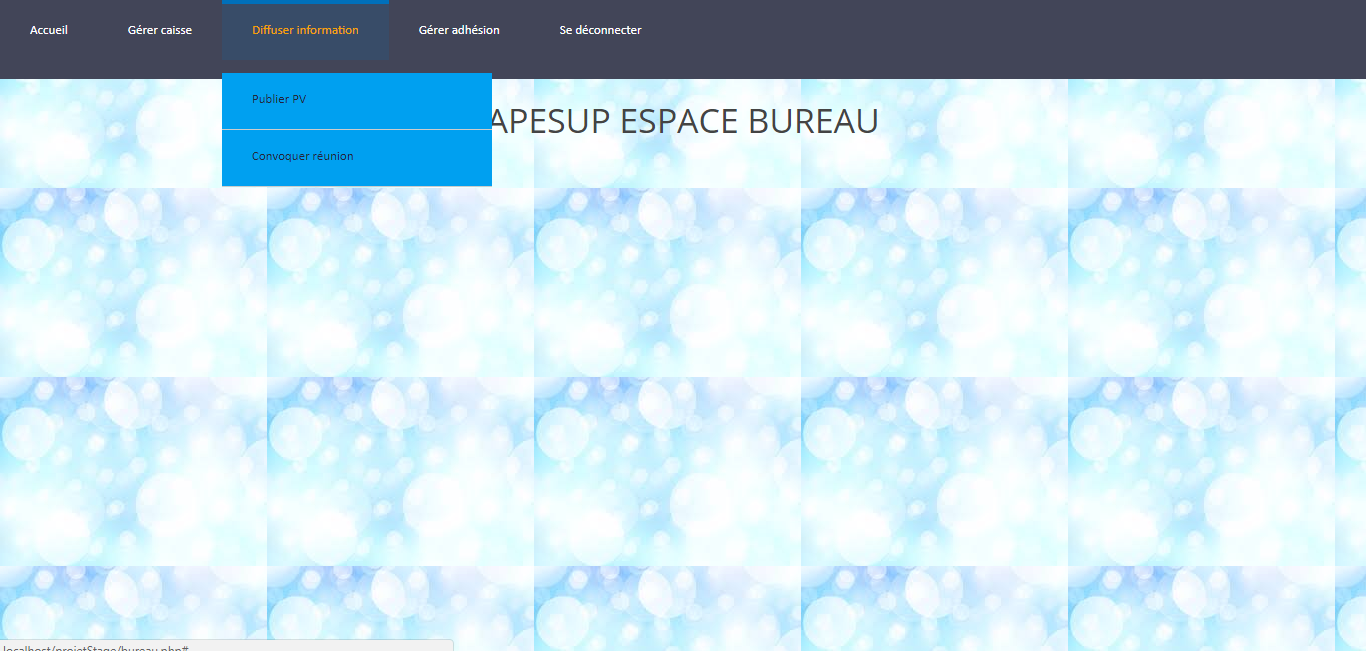


Figure 19 : Page de connexion

* La page d’accueil du bureau

Cette page représente l’espace bureau de l’APESUP. On y voit les fonctionnalités « gérer caisse », « diffuser information », « «gérer adhésion » et le lien de « déconnexion ».





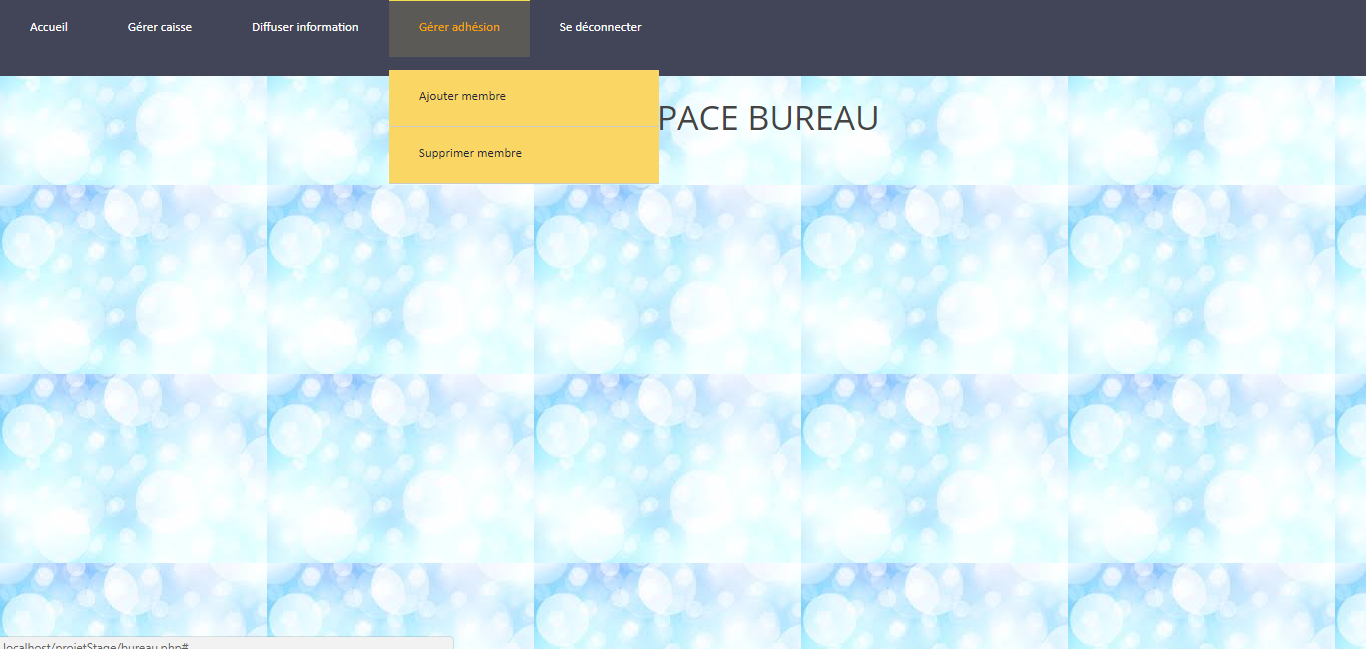


Figure 20 : Page accueil du bureau

* La page « Enregistrer cotisation »

Sur cette page, le bureau enregistre les données (nom, prénom, mail, carte membre, montant, structure, la date et le ou les mois versés) du membre qui a donné son versement.



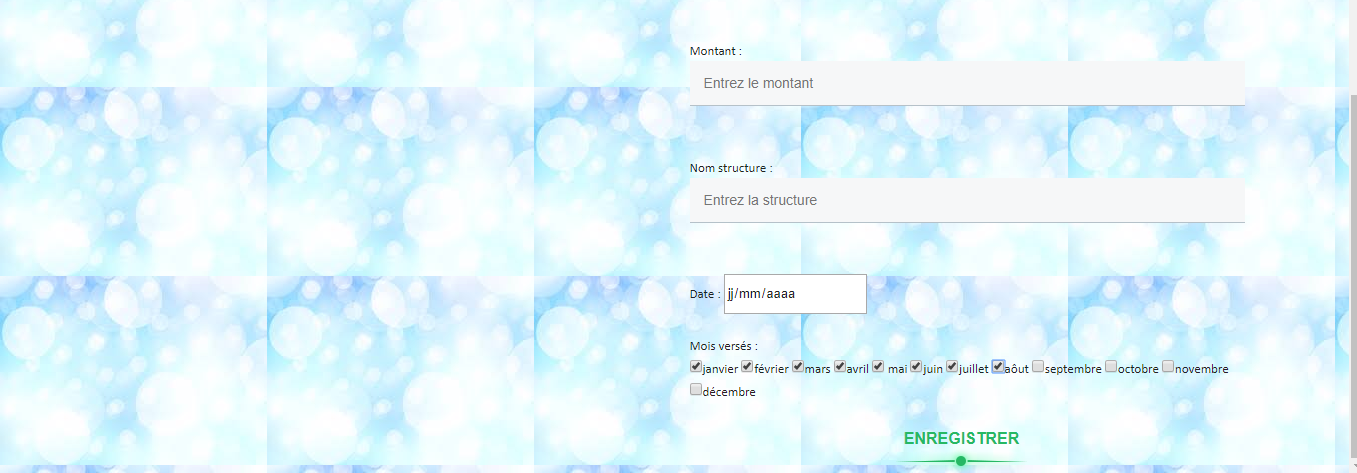
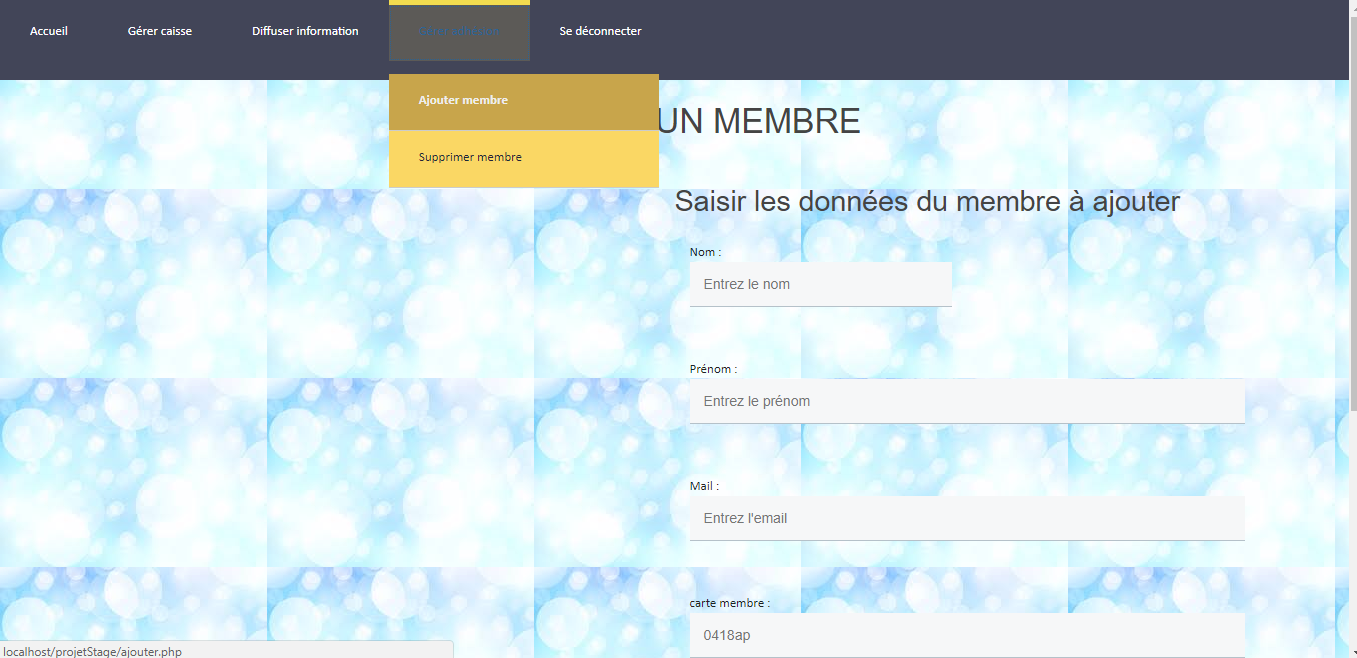


Figure 21 : Page enregistrer cotisation

* La page « Ajouter membre »

Sur cette page, le bureau saisit les données (nom, prénom, mail, adresse, carte membre, téléphone, date adhésion) du membre à ajouter. Le système vérifie si l’email et la carte de membre sont corrects puis enregistre les données dans la base de données.



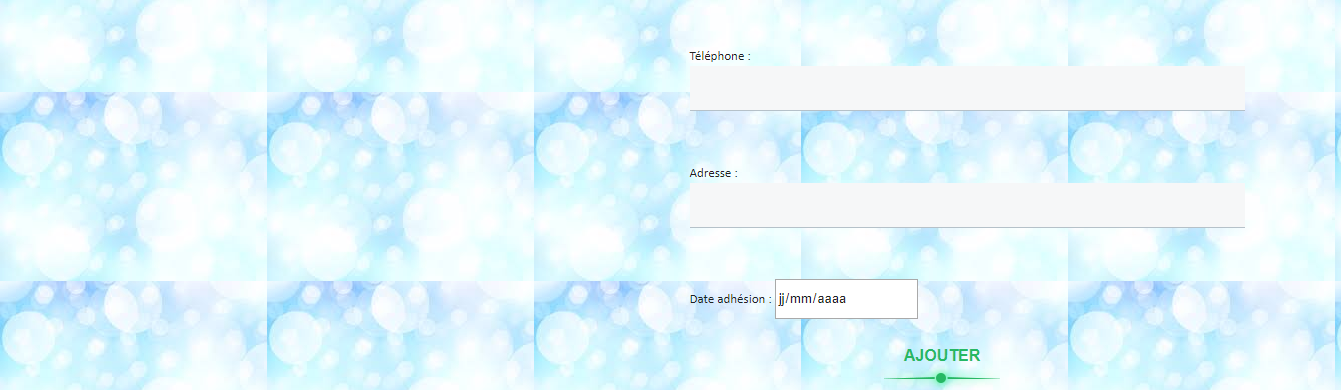


Figure 22 : Page ajouter membre

* La page « publier PV »

Cette page permet au membre du bureau de publier les procès-verbaux des réunions, AG. Le membre de bureau charge un fichier contenant le contenu du compte rendu de réunion, AG pour ensuite le publier.



Figure 23 : Page publier PV

* La page « voir état caisse »

La page « Voir état cotisation » permet au membre de l’APESUP de consulter ses états. C’est-à-dire si le membre accède à cette page, il peut voir pour une année choisie, les mois qu’il n’a pas encore cotisé.



Figure 24 : Page voir état cotisation

***CONCLUSION***

Après un (1) mois et quinze (15) jours de stage au sein du Département Génie Informatique (**DGI**) de l’Ecole Supérieure Polytechnique (**ESP**) de Dakar pour l’obtention du diplôme universitaire de technologie, nous pouvons dire que l’expérience a été très fructueuse, car ce projet nous a permis d’avoir une compréhension plus claire de ce qu’est la gestion de projet informatique. Il nous a aussi donné l’opportunité de travailler en groupe, de suivre une méthodologie de travail bien étudié, d’approfondir nos connaissances dans le monde de développement des sites web.

L’objectif de notre projet de fin de cycle était de développer un système de gestion de l’Amicale du Personnel de l’ESP (APESUP). Ce système permet l’accès et le partage d’informations concernant l’amicale. Il permet aussi au membre de consulter les informations le concernant.

Toutefois, le choix d’une application web s’avère judicieux dans la mesure où il nous a permis de mener à bien notre projet. La phase d’analyse est restée une phase délicate car étant la plus cruciale, c’est pourquoi nous y avons passé beaucoup plus de temps pour pouvoir bien cerner les fonctionnalités de notre système.

Notre choix de méthode de développement s’est porté sur le langage UML car c’est UML qui est conçu pour l'implémentation d’objet avec ses différents détails et sa portabilité (s'adapte à n'importe quelle plateforme) ; elle est donc plus exploitable.

Plusieurs technologies ont été nécessaire pour la réalisation de notre projet, on citera donc le langage HTML pour la réalisation des pages statiques, le langage PHP pour la partie dynamique, SQL pour l'élaboration des requêtes d'interrogation de la base de donnée, et enfin le logiciel sublime Texte pour l'écriture du code.

Après le passage par les différentes étapes de développement, le projet a abouti à un système fonctionnel qui répond globalement aux critères imposés dans ce domaine.

***Webographie***

* <http://fr.openclassrooms.com/informatique/php/cours>
* <http://fr.openclassrooms.com/informatique/php/cours>
* [https://openclassrooms.com/courses/apprenez-a-creer-votre-site-web-avec-](https://openclassrooms.com/courses/apprenez-a-creer-votre-site-web-avec-html5-et-css3) [html5-et-css3](https://openclassrooms.com/courses/apprenez-a-creer-votre-site-web-avec-html5-et-css3)
* <https://fr.wikipedia.org/wiki/Sublime_Text>
* [http://laurent-audibert.developpez.com/Cours-UML/html/Cours-UML.html.](http://laurent-audibert.developpez.com/Cours-UML/html/Cours-UML.html)
* <https://fr.wikipedia.org/wiki/JavaScript>
* <http://www.wampserver.com/>
* <https://www.visual-paradigm.com/features/>
* <https://fr.wikipedia.org/wiki/JQuery>[http://jqwidgets.com/jquery-widgets-demo/demos/jqxdatatable/index.htm](https://www.visual-paradigm.com/features/)