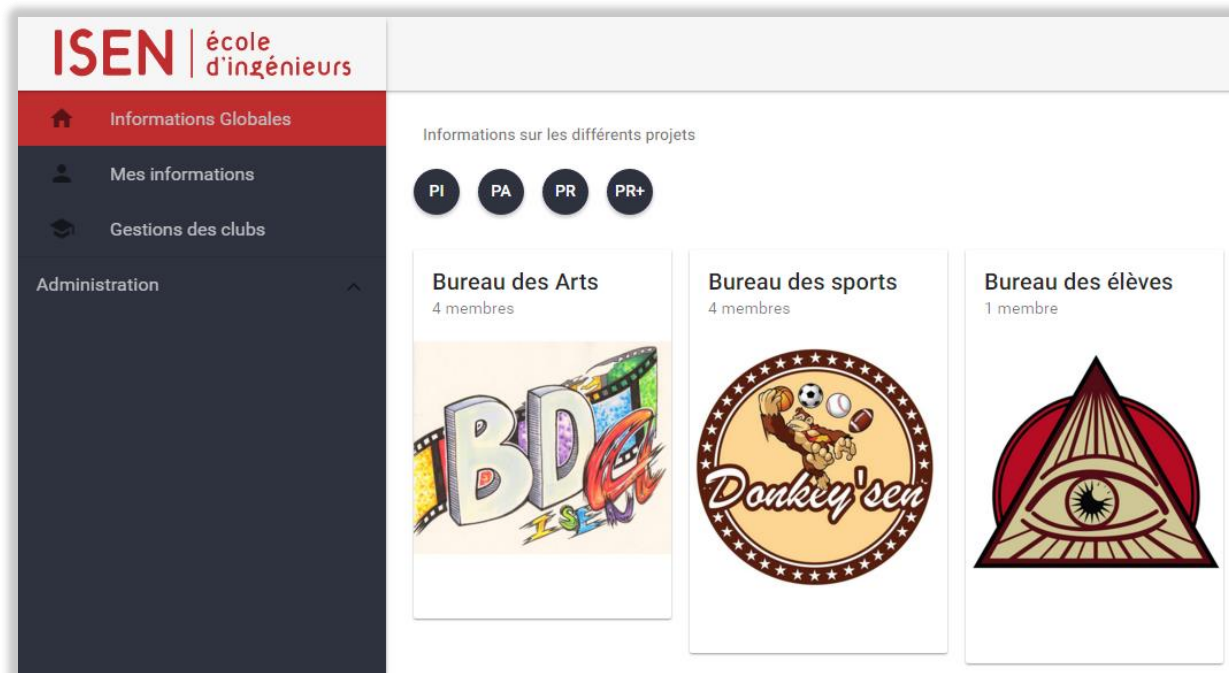


Rapport de projet M1 Interface de gestion des projets associatifs



Proposé par :
M. CABON Mikaël,

Responsable Formation Humaine et Sociale

Réalisé par :
M. RIOUALLON Vincent,
M. COLLIGNON Rémi,

Étudiant M1 option Technologie Bio-Médicale
Étudiant M1 option Génie Logiciel

1. Abstract

During the second year of the engineer cycle at ISEN Brest, students are assigned to a project given by professors. This project is meant to simulate an interaction between a customer and the engineers. They will interact with the referred teacher as they would do with a real client and realize the project as if they were a society providing a service.

This project aims to assist in allocating of students in second and third year for their associative project. A particularly important goal in this project is to keep a history of these projects. This historic will be used in later years for graduation.

The realization of this project is done through a web application to facilitate the access to data. The interface is designed for simplicity and aesthetics. A distribution algorithm was created to place the various users in the clubs. This solution has already been proven functionality and a number of results.

In the future this project will be taken up by another team that will continue to develop features and improve performance.

2. Résumé

Le but de ce projet est de garder une trace des différents projets bénévoles (PB) et des projets responsables (PR) jusqu'à l'obtention du diplôme. A l'ISEN, la validation d'un projet à responsabilité est nécessaire pour cette obtention, de plus, le projet donne lieu à une mention sur le diplôme d'ingénieur. Il est donc primordial d'être en mesure de pouvoir connaître la validation de projets passés.

Son deuxième intérêt est de faciliter la notation des clubs par leurs évaluateurs. L'objectif est également d'évaluer les clubs et ses membres, afin de faciliter l'insertion des notes dans WebAurion.

Une répartition automatique des élèves dans les clubs sera mise en place pour un gain de temps et pour une question de simplicité. Cette répartition repose sur de nombreuses conditions et sera par conséquent d'une complexité avancée.

Ce projet met en jeu plusieurs acteurs, chacun ayant des attentes différentes du projet, attentes que la solution s'efforce à combler.

Enfin, le projet a pour but d'informer les étudiants des caractéristiques des différents projets possibles (PI / PA / PR / PR+) et de clarifier les attentes sur les projets.

Le client souhaitant une définition plus précise du besoin, des sondages ont été réalisés. Les résultats seront présentés dans ce dossier.

Dans sa finalité le projet fera ses preuves lors de notation des clubs de 2016 et donnera lieu à l'affectation automatique des élèves.

3. Sommaire

1. Abstract	2
2. Résumé	2
3. Sommaire	3
4. Introduction	5
5. Gestion de projet	6
5.1 Détermination des objectifs	6
5.2 Définition du besoin	7
5.3 Situations de vie du produit	7
5.4 Les milieux extérieurs lors de la phase d'utilisation	9
5.5 Définition plus précise du besoin	10
5.6 Définition du projet mnimum requis	11
6. Cahier des charges	12
6.1 Entrées	12
6.2 Sorties	12
6.3 Contraintes	12
6.4 Risques	12
6.5 Srtructuration et estimation	12
6.6 Planification globale	13
6.7 Definition des étapes du projets	14
6.8 Coût estimé	15
6.9 Logiciels, outils, matériels et technologies utilisés	16
7. Méthodologie	20
7.1 Méthode agile	20
7.2 Outil de gestion	20
8. Développement technique	21
8.1 Démarche du projet	21
8.2 Conception de la base de données	21
8.3 Mise en place d'une architecture	22
8.4 Mise en place d'une API REST	22
8.5 Mise en place de controller et de modele	23
8.6 Mise en place d'un front-end et des fonctionnalités	23
8.7 Déroulement des tests	27
8.8 Mise en production Initiale	28

8.9	Algorithme de répartition	29
8.10	Mise en production Finale	31
8.11	Intégrité des données	32
9.	Conclusion	34
10.	Remerciements	35
11.	Glossaire	36
12.	Table des figures	37
13.	Webographie	37
14.	Annexe	38

4. Introduction

L'étude est proposée lors de la réalisation de projets en fin de 4^{ème} année d'école d'ingénieur. L'exercice se passe comme en entreprise. Un client (ici un professeur) nous soumet son besoin et attend du binôme une solution concrète.

Le client du produit réalisé, M. Cabon est responsable de l'équipe FHS.

Actuellement l'historique des Projets Associatifs et Projets Responsables est conservé sur papier (archives) et dans la mémoire collective. Il peut ainsi arriver que l'on perde la trace d'un projet associatif effectué. Sa recherche pourrait être évitée grâce à une gestion rigoureuse des projets réalisés par les Isenniens. C'est dans une optique de normalisation et de création de processus que le projet est né.

La notation se fait par recopie d'un tableau Excel rendant donc la tâche fastidieuse et propice aux erreurs de saisies.

Cette situation n'est pas propice à l'évolution des processus et à gestion de labels pour le diplôme.

Notre projet contribue à l'évolution du processus en améliorant le système de notation, de répartition des élèves dans les clubs et de conservation des données.

5. Gestion de projet

5.1 DETERMINATION DES OBJECTIFS

5.1.1 Quoi ?

Une solution informatique permettant la gestion des clubs au sein de l'ISEN. Cette interface doit être accessible pour ses utilisateurs afin de leur permettre d'avoir une vision d'ensemble des caractéristiques des clubs. Le système doit aussi être capable de répartir les élèves en fonction de leurs choix dans les clubs, en fin d'année. A tout moment, l'évaluateur doit pouvoir noter, changer, réattribuer les membres d'un club.

5.1.2 Qui ?

Au sein de l'école plus de 30 clubs participent à la vie associative de l'ISEN, ils sont composés d'élèves de deuxième et troisième année (et potentiellement plus) souhaitant s'investir dans un groupe similaire à de l'associatif. Le domaine d'activité de ces clubs est varié, de l'organisation de semaine de ski, à la conception de carte électronique, en passant par la mise en place de tournois sportifs.

Quatre types d'utilisateurs généraux existent :

- Les membres de clubs
- Les présidents
- Les évaluateurs
- Les administrateurs

A ceux-ci se rajoutent des cas particuliers tels que les clubs à recrutement spécifique BDE, BDS et Capisen.

5.1.3 Où ?

La solution a pour vocation d'être mise en place au sein de l'école. Elle doit être accessible depuis un ordinateur ou un smartphone (les terminaux seront choisis en fonction des réponses au questionnaire). Ce service doit être accessible aussi bien de l'école que de l'extérieur.

5.1.4 Quand ?

La solution a pour but d'être utilisée en début d'année pour l'affectation des membres puis en fin d'année pour l'attribution des notes.

Une utilisation réelle de la solution est prévue pour la notation des clubs de l'année 2015-2016, soit aux alentours du 06/05 afin d'éprouver le système.

5.1.5 Comment ?

La solution prendra la forme d'une application web. Via son interface, elle permettra à l'utilisateur d'avoir accès à des fonctions spécifiques selon son groupe.

Dans sa phase de test, l'application sera développée en local, sans impact sur les données réelles.

5.1.6 Pourquoi ?

Depuis la création des clubs, il a été constaté que l'attribution des notes est une tâche fastidieuse et source d'erreurs. Du point de vue des clubs, toutes les fonctionnalités sont réalisées manuellement (papier). L'attribution des élèves dans les clubs en début d'année est aussi une tâche chronophage. Enfin, le stockage d'archives contenant les différents projets de FHS pour chaque étudiant depuis de nombreuses années rend la recherche de données complexe.

Le principal but de la solution est d'éradiquer les tâches fastidieuses des professeurs et des élèves. Par ailleurs, l'avantage est de garder une trace numérique et immuable des données concernant les clubs afin d'en conserver un historique

5.2 DEFINITION DU BESOIN

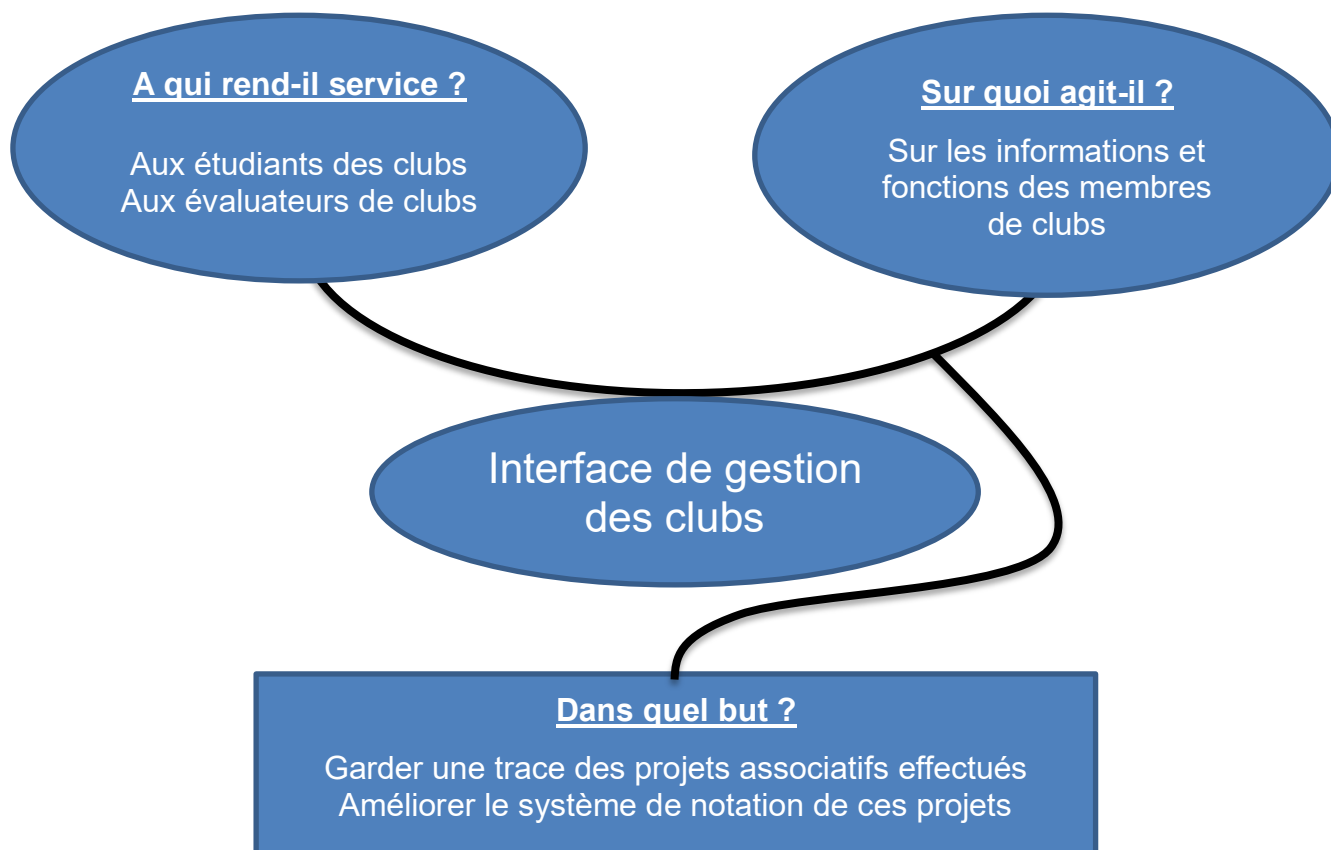


Figure 1 - Bête à cornes

5.3 SITUATIONS DE VIE DU PRODUIT

5.3.1 Installation de l'interface

L'interface devra être installée sur un serveur en ligne afin d'être accessible via internet.

5.3.2 Utilisation de l'interface

L'utilisateur se connecte, via l'authentification de l'ENT, puis accède au « panel » qui lui correspond (Evaluateur, Président de club, Elève, ...). Il pourra ainsi administrer ce qui l'intéresse, ou saisir les informations nécessaires dans le cas des élèves postulant aux clubs. Dans un objectif secondaire, l'interface devra être adaptée aux différentes plateformes (Smartphone, tablette, PC).

5.3.3 Fonctionnement de l'interface dynamique

Le système se régule automatiquement et est construit pour permettre des modifications futures. L'interface est découpée en « objets » créés à la volée permettant ainsi des performances optimales.

5.3.4 Mise à jour de la base de données

La création d'un utilisateur dans la base de données se fait lors de sa première connexion à l'interface. Une sauvegarde automatique de la base est effectuée chaque jour à 12h30 afin de minimiser les impacts de dysfonctionnements.

Il n'y aura aucune connexion de la base de données de l'application vers une autre ; les risques de corruption ou de suppression sont donc évités.

5.4 LES MILIEUX EXTERIEURS LORS DE LA PHASE D'UTILISATION

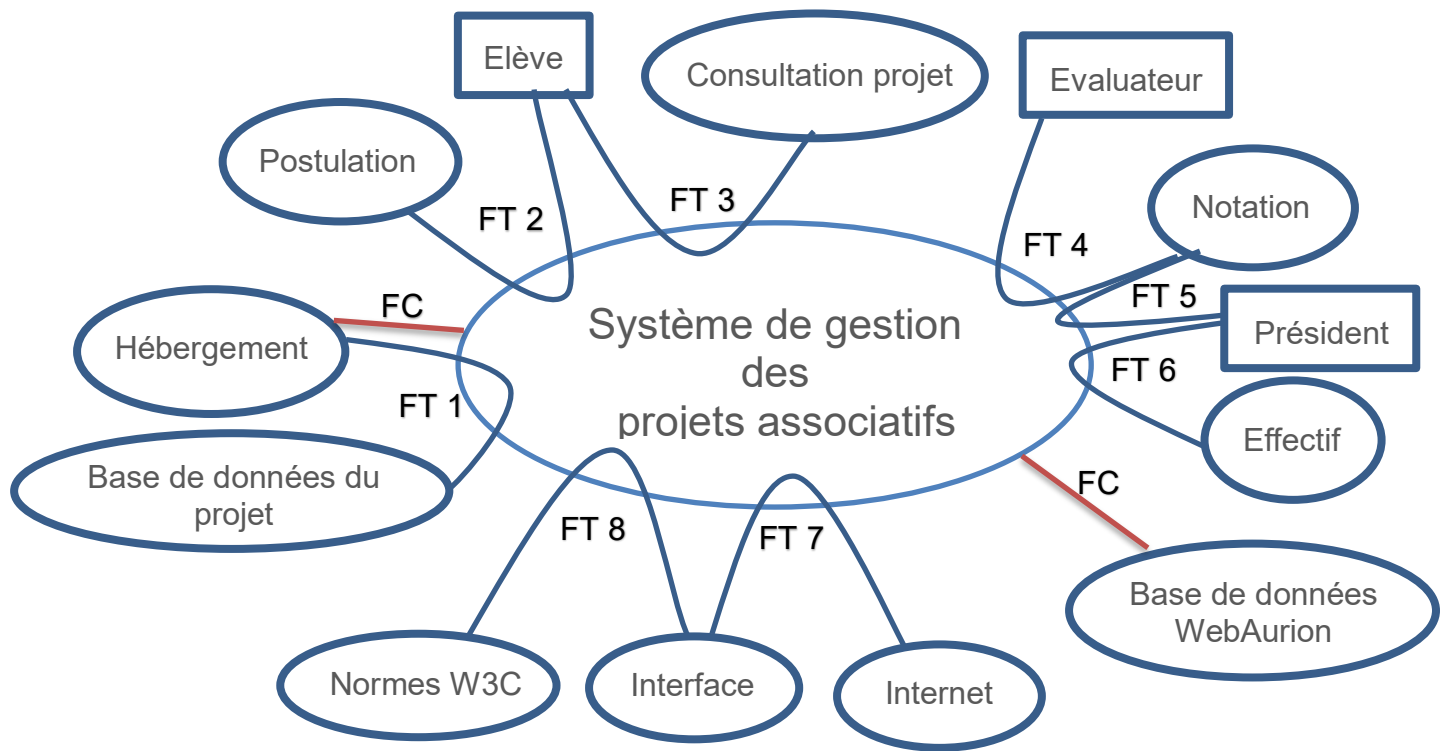


Figure 2 - Diagramme pieuvre

FONCTION Transfert

FT 1 : L'hébergement fournit dispose d'une base de données pour le projet

FT 2 : L'élève peut postuler dans un club

FT 3 : L'élève peut consulter les projets d'un club

FT 4 : L'évaluateur note un club

FT 5 : Le président répartit les points

FT 6 : Le président régule les effectifs

FT 7 : L'interface est accessible par internet

FT 8 : L'interface respecte les normes du web (W3C)

FONCTION Contrainte

FC 1 : L'hébergement permet un accès à l'interface

FC 2 : Le système récupère des informations de la base de données WebAurion

5.5 DEFINITION PLUS PRECISE DU BESOIN

5.5.1 Sondages

Les sondages effectués auprès des différents profils se sont déroulés sous forme de questions-réponses.

Une partie plus libre a permis aux personnes interrogées d'exprimer leurs attentes vis-à-vis du projet. L'objectif est de cerner les différentes fonctionnalités souhaitées afin de favoriser la migration des utilisateurs vers l'interface tout en complétant les attentes principales du client. Le client peut ainsi avoir une liste des fonctions souhaitées par les futurs utilisateurs et se rendre compte des travaux attendus.

5.5.2 Recueil

Ci-dessous le compte rendu rédigé suite aux sondages effectués auprès des professeurs (évaluateurs) et des élèves.

Le compte rendu se concentre sur les informations utiles et en rapport avec le projet.

5.5.3 Recueil pour les évaluateurs

Voici la liste des fonctionnalités demandées par les évaluateurs de club.

Fonctionnalités administratives :

- Une saisie facilitée des notes des clubs en fin d'année
- Une trace des interactions avec un club
- Un historique par étudiant (date d'obtention PI / PA / PR / PR+, note, commentaire du président)
- Une proposition d'exemples de grilles d'objectifs et du dossier de passation
- Un résumé du temps passé à évaluer un club
- Une possibilité pour l'évaluateur d'être noté par les clubs
- Une simplification de la gestion des effectifs dans les clubs

Fonctionnalités informatives :

- Une définition accessible des PI / PA / PR / PR+ et une explication de leurs rôles et objectifs
- Une liste complète des étudiants avec filtres ou un champ de recherche
- Une liste globale par PI / PA / PR / PR+ avec validation et notes
- Une proposition des parcours dans ces labels
- Une interface conviviale, fluide et facile d'utilisation
- Un maintien du contact avec les anciens membres du club
- Un maintien des contacts avec des organisateurs extérieurs (banque alimentaire, ...)

Fonctionnalités organisationnelles :

- Une planification des activités de chaque club (événements)
- Une intégration de ces événements en base de données et intégration à Google Calendar
- Les retours d'un événement (ex : le nombre de personnes, ...)
- Une proposition de guides pour remplir ces 2 formulaire

5.5.4 Recueil pour les élèves

Fonctionnalités informatives :

- Historique des projets réalisés dans le club
- Tableau récapitulatif des membres de tous les clubs
- Courriel automatique vers les clubs à la fin de la répartition
- Tableau Excel généré avec les membres d'un club
- Mentionner le lien Facebook / YouTube dans la page de présentation d'un club

Fonctionnalités administratives

- Intégrer les différentes listes de BDE pour les élections
- Pouvoir recommander les membres qui semblent intéressés par le club
- Gestion de cas spécifiques avant la répartition (BDE, BDS, Junior Entreprise)
- Possibilité d'être réaffecté à un club
- Gestion des ½ PA et ½ PR (*demande de non implémentation de la part du client*)
- Comparaison des fiches des personnes postulantes pour les clubs fonctionnant sur entretiens (Capisen)
- Définir un nombre minimum/maximum de personnes souhaitées pour chaque projet et chaque club
- Gérer les clubs pour les Rennais
- Gérer les projets et les activités (événements) des clubs
- Définir les rôles des membres d'un groupe

Fonctionnalités de notation et de bilan :

- Formulaire de validation/invalidation des projets associatifs
- Intégrer les fiches d'objectifs
- Gérer la trésorerie d'un club
- Redistribution des points de la note par le président

- Optimiser la consultation sur Smartphone (surtout pour de la consultation, peu pour de réelles manipulations)

5.6 DEFINITION DU PROJET MNIMUM REQUIS

Le MVP a été décidé avec le client (M. Cabon). Il se base sur ses attentes initiales et sur les informations mises en évidence par les sondages.

- Interface disponible sur ordinateur
- Vision globale des clubs pour un évaluateur
- Attribution d'une note à un club
- Vision succincte des membres pour un président
- Export des notes sous forme de fichiers de type Excel
- Attribution d'un club aux étudiants en fonction de plusieurs paramètres (ancien membre, numéro du choix, recommandé ou inversement, type de projet associatif à effectuer, ...)

Le projet comporte un certain nombre d'objectifs primaires et secondaires. Tous les objectifs primaires doivent être réalisés, les objectifs secondaires les plus pertinents pourront également être réalisés.

Ces objectifs primaires sont ceux décrits dans le MVP et représente une charge de travail considérable pour allier simplicité, efficacité et traitement complexe. Le travail fourni doit être **reprenable** par une future équipe afin de continuer l'implémentation des fonctions.

Un impératif est de mettre en production la solution avant la notation des clubs afin qu'elle puisse être immédiatement utilisé.

6. Cahier des charges

6.1 ENTRÉES

- Description et définition du projet
- Etude de l'existant
- Technologies choisies

6.2 SORTIES

- Réunions officielles avec le client
- Source du projet disponible sur le site internet GitHub à l'adresse : <https://github.com/projet-ISEN/projet>
- Mise en production du projet sur un serveur interne de l'ISEN Brest
- Rapport technique écrit
- Rapport de mise en production écrit
- Manuel d'utilisation écrit
- Soutenance et démonstration

6.3 CONTRAINTES

- Interfaçage avec le CAS (récupération des informations d'un utilisateur connecté)
- Acquisition et configuration d'un sous domaine de isen-bretagne.fr (fourni par l'école)
- Un serveur de l'école avec une base de données
- Limite de temps
- Formations sur certaines technologies utilisées

6.4 RISQUES

- Retard lors d'une des phases du projet
- Non obtention d'un des éléments de dépendance critique du projet

6.5 SRTRUCTURATION ET ESTIMATION

Lorsque le cahier des charges est bien défini, il est nécessaire de pouvoir le diviser en étapes et fonctionnalités à réaliser pour le bon accomplissement du projet. En entreprise, ces étapes sont assimilées à des blocs horaire permettant de réaliser une facturation au client. Il est crucial d'avoir une bonne évaluation de la charge de travail pour évaluer les coûts de main d'œuvre.

Dans ce projet, la main d'œuvre est exempte de charges. Il reste cependant important de prendre le temps de bien définir les étapes maîtresse du projet.

Ci-dessous, la planification globale et les différentes phases du projet du point de vue de la gestion de projet.

6.6 PLANIFICATION GLOBALE

Suite à l'analyse fonctionnelle du projet, un planning prévisionnel a pu être établi. Il est constitué des principales activités à mener dans leur ordre chronologique et est décrit par le tableau suivant :

N°	Nom de la tâche	Elongation	Début	Fin	Prédécesseurs
1	Réunion avec le client (M. Cabon)	1 hr	Ven 18/12/15	Ven 18/12/15	
2	Choix et mise en place de la plateforme de gestion de projet	4,5 jours	Lun 21/12/15	Ven 25/12/15	1
3	Définition globale du projet	4,5 jours	Lun 28/12/15	Ven 01/01/16	2
4	Réunion N°1 avec le client (M. Cabon)	1 hr	Lun 04/01/16	Lun 04/01/16	
5	Création d'un sondage	4 hr	Lun 04/01/16	Lun 04/01/16	3
6	Sondage des besoins auprès des profils types	5,5 jours	Mar 05/01/16	Mar 12/01/16	5
7	Définition du cahier des charges globales	6,5 jours	Mar 05/01/16	Mer 13/01/16	
8	Réunion M. Gerval pour base de données et intégration à l'ENT	10 hr	Jeu 14/01/16	Jeu 14/01/16	
9	Réflexions sur les technologies à utiliser	3,5 jours	Jeu 14/01/16	Mar 19/01/16	6;7
10	Rédaction d'une analyse technique pour le choix des technologies	2 jours	Mar 19/01/16	Mer 20/01/16	
11	Réunion avec le client pour validation des technologies	1 hr	Jeu 21/01/16	Jeu 21/01/16	10
12	Première phase de développement	40 jours	Ven 22/01/16	Ven 06/05/16	
13	Essai de la solution initiale	9 jours	Lun 9/05/16	Mer 18/05/16	12
14	Deuxième phase de développement	14 jours	Lun 9/05/16	Mer 18/05/16	13
16	Rédaction du rapport de projet	8 jours	Lun 16/05/16	Ven 20/05/16	
17	Préparation de la présentation du projet	7 jours	Jeu 19/05/16	Mar 24/05/16	

Dans la planification globale du déroulement du projet, des réunions avec le client ont été ajoutées, afin de lui faire un retour de l'état d'avancement et la conformité du travail demandé.

Ces réunions ont permis de remettre en question de la définition de la solution que ce soit par des ajouts de fonctions ou des propositions de solutions pratiques aux fonctionnalités demandées. Le cahier des charges a donc évolué, tout en conservant le MVP définissant le but à atteindre.

6.7 DEFINITION DES ETAPES DU PROJETS

Phase 1 : Avant-projet

Durant cette phase, une étude de l'existant a été nécessaire ainsi qu'une analyse de la demande, afin de détourner et définir le besoin du client dans sa globalité. Cette étude présente une description du besoin et les coûts.

Cette phase permet de réunir toutes les informations nécessaires à la réalisation du projet.

Phase 2 : Phase d'analyse

Afin de développer la solution, une analyse des besoins techniques est nécessaire. Une fois cette analyse effectuée, une étude des outils et solutions existantes sera effectuée afin de définir les technologies/solutions les plus adaptées à satisfaire le client.

Les outils doivent permettre la réalisation de la solution et correspondre aux exigences du client.

Phase 3 : Fonctionnement de la solution

Pour développer l'application (qui est la solution demandée par le client), il est indispensable de bien concevoir son architecture.

Pour que la solution soit dynamique et performante il a été décidé de se baser sur le modèle MVVM.

Phase 4 : Modèle de données

La mise en place d'une base de données est l'un des piliers de la solution car la vue et le modèle se reposeront sur ce pilier. La création du module de données devra donc être suivie avec grande attention.

Phase 5 : Mise en place du moteur de rendu

Le moteur de l'application permettra de faire la liaison entre les données présentes en base et les fonctions de l'application. Ce moteur permettra de faire les vérifications d'actions des utilisateurs. Il devra être capable de protéger les données des utilisateurs afin d'éviter toute corruption de la base, voire la destruction de celle-ci. La sécurité prend donc une part conséquente dans l'attention à apporter dans cette phase.

Phase 6 : Vue des utilisateurs

La vue des utilisateurs devra être logique et fluide. La solution devra être simple d'utilisation et esthétique. Une des spécificités de la demande du client est une interface au « design ¹ » agréable. L'interface de l'utilisateur devra être dynamique et réactive afin de faciliter son intégration et sa vitesse de prise en main.

¹ Le « design » est un concept qui consiste à créer, choisir et utiliser des éléments graphiques pour élaborer un objet

Phase 7 : Tests

Durant la réalisation des différentes phases, des tests seront réalisés. Ils devront pour certains être réalisés automatiquement et d'autres plus critiques nécessiteront une attention particulière. Si l'application passe tous les tests, elle répondra au cahier des charges et donc à la demande.

Phase 8 : Mise en production

Une mise en production en fin de projet sera réalisée afin que l'application puisse être immédiatement utilisée par le client et les autres utilisateurs concernés.

Phase 9 : Rapport et manuels

Une fois l'application réalisée et mise en service, la rédaction d'un rapport de projet ainsi qu'une préparation de soutenance devront être réalisées.

A la suite de cela, la rédaction d'un manuel de mise en production ainsi qu'un manuel d'utilisation seront rédigés pour permettre une pérennité du projet.

Phase 10 : Choix des technologies

Une fois la définition des phases du projet effectué, il est impératif de réaliser la sélection les technologies qui seront employées pour réaliser chacune des phases du développement du produit. Ces différentes technologies seront détaillées dans la partie « Logiciels, outils, matériels et technologies utilisés » de ce rapport.

6.8 COUT ESTIME

Ce produit utilisera une solution informatique. Le seul matériel nécessaire est le serveur sur lequel il sera mis en production. La mise en œuvre se fera par des technologies « open-source ² » permettant de ne pas avoir de dépenses.

L'environnement de développement se fera sur nos machines.

Les coûts de main d'oeuvre développement seront nuls car il s'agit d'un projet réalisé dans le cadre de la formation ISEN.

² La désignation open source, ou « code source ouvert », s'applique aux logiciels dont la licence respecte des critères précisément établis par l'Open Source Initiative, c'est-à-dire les possibilités de libre redistribution, d'accès au code source et de création de travaux dérivés

6.9 LOGICIELS, OUTILS, MATERIELS ET TECHNOLOGIES UTILISES

6.9.1 Logiciels



Figure 3 - Logo de Github

Git et Github seront utilisés en tant que gestionnaire/synchronisateur de version communautaire. Il sera ainsi possible de travailler à plusieurs et revenir à des versions antérieures en cas de besoin.

Le nom GitHub est composé du mot « git » faisant référence à un système de contrôle de version open-source et le mot « hub » faisant référence au réseau social bâti autour du système Git.

GitHub propose l'intégration d'un grand nombre de services externes, tels que l'intégration continue, la gestion de versions, les badges, le service de discussion instantané, basés sur les projets.



Figure 4 – Logo de MySQL

Pour gérer la base de données, un serveur MySQL a été mis en place. Ce système permet de bonnes performances ; il est également utilisé lors de projets scolaires. MySQL est un système de gestion de bases de données.

Un Système de Gestion de Base de Données (SGBD) est un logiciel (ou un ensemble de logiciels) permettant de manipuler les données d'une base de données. Manipuler, c'est-à-dire sélectionner et afficher des informations tirées de cette base, modifier des données, en ajouter ou en supprimer.



Figure 5 - Logo de PowerAMC

L'élaboration de la base de données pour le projet s'est faite au travers du logiciel PowerAMC. Il a ainsi été possible de réaliser le MCD puis de générer un MPD. Si les prochaines équipes veulent modifier cette base ou l'exporter vers un autre système de données que celui-ci, ce sera possible.

Ce logiciel de création de modèle de données est complet, facile d'utilisation et sa licence gratuite grâce à l'ISEN.

PowerAMC est un logiciel de conception, qui permet de modéliser les traitements informatiques et leurs bases de données associées.

6.9.2 Outils



Figure 6 - Logo de Travis

Travis CI a permis de tester le code à chaque nouvelle version mise en ligne sur le serveur. Son utilisation a été limitée à la « transpilation ³ » du projet afin de vérifier son bon déploiement. Si le projet devient plus important en terme de taille et de complexité, l'utilisation de Travis CI deviendra impérative afin de garantir une intégration continue.

Travis CI est un logiciel libre et un service en ligne utilisé pour compiler, tester et déployer le code source des logiciels développés, notamment en lien avec le service d'hébergement du code source GitHub.

³ Action de retranscrire un langage informatique en un autre, notamment utilisé pour les pseudo-langages



Figure 7 - Logo de Gulp

Gulp a été utilisé afin de faciliter le développement du produit. Il a servi à exécuter les tâches rébarbatives et chronophages.

Gulp est un automatisateur de tâches ; il est capable de lancer des séquences de scripts sur certains événements. Il peut par exemple « transpiler » tout le code afin de vérifier les erreurs de syntaxes et visualiser directement le résultat de la programmation.

6.9.3 Matériels

Pour ce qui est du matériel seul un serveur est nécessaire (il peut être virtualisé) afin de faire tourner notre application web et de la rendre disponible.

6.9.4 Technologies

Plusieurs technologies vont être utilisées durant ce projet. Elles ont été choisies de manière à avoir de bonnes performances, une facilitation de programmation et sont issues du vivier des technologies mais aussi en fonction des technologies enseignées à l'ISEN. Le projet pourra ainsi être plus facilement compréhensible par les futures équipes en charge de son développement.

6.9.4.1.1 Back-end

Pour les langages utilisés en « back-end ⁴ », les technologies présentées ci-dessous ont été employées.



Figure 8 - Logo de SQL

Les requêtes exécutées sur les données seront réalisées en SQL. Ce langage est enseigné à l'ISEN lors de l'initiation à la gestion de SGBDR.

SQL (Structured Query Language, en français langage de requête structurée) est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données relationnelles.



Figure 9 - Logo de PHP

Les traitements côté serveurs seront effectués en PHP car c'est un langage propice à ce type d'utilisation et il est enseigné à l'ISEN.

PHP: Hypertext Preprocessor[5], plus connu sous son sigle *PHP* (acronyme récursif), est un langage de programmation libre[6], principalement utilisé pour produire des pages Web dynamiques ou renvoyer des données via un serveur HTTP[5], mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté objet.

⁴ En informatique, un back-end (parfois aussi appelé « arrière-plan ») est un terme désignant l'étage de sortie d'un logiciel devant produire un résultat.



Figure 7 - Logo de Node JS

Node est une technologie nécessaire à la bonne exécution de npm et Bower permettant d'inclure des librairies. Cette inclusion est simple, optimisée, automatisée et permet de ne pas encombrer le dépôt Git. L'utilisation d'un tel « package manager 5 » permet une gestion plus fine et efficace des versions de librairies. Les modules tels que npm et Bower facilitent la mise en production (seulement trois commandes pour installer toutes les librairies nécessaires). En outre, ces modules permettent de gérer les dépendances des scripts.



Figure 9 - Logo de Bower



Figure 10 - Logo de npm

Node.js est une plateforme logicielle libre et événementielle en JavaScript orientée vers les applications réseau qui doivent pouvoir monter en charge.

npm (Node Package Manager) est le gestionnaire de paquets officiel pour Node.js. Bower est un gestionnaire de paquets pour le web.

Pour permettre une liaison fluide et dynamique entre la gestion des données et les interactions de l'utilisateur, il a été décidé que AngularJS ferait cette liaison.



Figure 11 - Logo de AngularJS

6.9.4.1.2 Front-end

Pour les technologies utilisées en « front-end ⁶ », deux familles de langages sont employées. La première est la famille des « langages transpilés ⁷ ». La seconde famille correspond aux langages qui seront utilisés pour le résultat final du produit.

Les langages « transpilés » utilisés :



Figure 12 - Logo de Jade

Jade est un langage permettant de produire du code HTML. Il possède certaines logiques de programmation. Ce langage se base sur l'indentation ce qui permet une logique dans le code et une présentation claire.

CoffeeScript est un langage de programmation, qui se compile en JavaScript. Les programmes peuvent être optimisés en réduisant le nombre de lignes de code écrits (typiquement -30% de lignes en moins) sans impact sur la vitesse d'exécution.



Figure 13 - Logo de CoffeeScript



Figure 14 - Logo de Sass

Sass un métalangage de feuilles de style en cascade. C'est un langage de script qui est interprété en CSS. De la même manière que Jade, il permet un gain de temps et une uniformité de l'indentation de code suivant les programmeurs.

⁵ Un gestionnaire de paquets est un (ou plusieurs) outil(s) automatisant le processus d'installation, désinstallation, mise à jour de logiciels installés sur un système informatique

⁶ En informatique, un frontal peut désigner une interface de communication entre plusieurs applications hétérogènes ou un point d'entrée uniformisé pour des services différents.

⁷ Les langages transpilés sont des langages qui seront traduits en un autre langage dans le but d'être exécutés

Les langages de sortie sont les langages classiquement utilisés en web c'est-à-dire :

- Le HTML pour l'architecture de la page
- Le CSS pour l'aspect visuel
- Le JavaScript pour donner du dynamisme et de la réactivité aux pages



Figure 17 - Logo de HTML5



Figure 16 - Logo de CSS3



Figure 15 - Logo de JavaScript

7. Méthodologie

7.1 METHODE AGILE

La méthode de travail utilisée durant ce projet peut être apparentée à une méthode Agile.

Le projet est de courte durée (moins de six mois), le temps attribué n'est pas du plein temps et l'équipe est extrêmement réduite (deux personnes).

La méthode du projet s'est donc basée sur le principe de la méthode agile mais exemptée de certaines caractéristiques qui semblaient plus convenir pour des équipes et projets plus importants.

Nous avons conservé le principe de création d'un « backlog ⁸», de « user story ⁹» et de « sprint ¹⁰».

Les sprints ont été fixés à une durée d'une semaine permettant ainsi de se rendre compte de l'avancée et par conséquent d'une potentielle prise de retard sur le projet.

Au travers des réunions hebdomadaires, la méthode Agile a permis au client de l'impliquer dans le projet. Le résultat du sprint précédent lui était exposé. Les fonctionnalités étaient testées et approuvées ou alors une demande de modification était apportée.

Le backlog a été mis à jour tout au long du projet et pourra être transmis dans un dossier spécifique.

Les itérations ont été construites en fonctions de l'ordre de priorité d'urgence des tâches ainsi que leur importance. En fin d'itération, la suivante était construite et cela jusqu'à la rédaction de ce rapport.

La méthode Agile a permis une très grande réactivité et adaptabilité au projet notamment par rapport aux modifications et demandes effectuées durant le développement du produit.

7.2 OUTIL DE GESTION

Pour suivre l'évolution du projet et avoir un outil simple d'utilisation et synchronisable, l'utilisation de « Asana » a été décidée. Asana est une application web et mobile conçue pour permettre le travail d'équipe sans courriel. ».

L'application web susnommée a permis de gérer les fonctionnalités à implémenter et avoir une plateforme facile d'utilisation pour consulter les « deadline ¹¹». Son utilisation a également permis une répartition des tâches et une fluidité dans le développement du projet.

⁸ Un "backlog" est une liste de fonctionnalités ou de tâches, jugées nécessaires et suffisantes pour la réalisation satisfaisante du projet

⁹ Un récit utilisateur ou user story est une phrase simple dans le langage de tous les jours permettant de décrire avec suffisamment de précision le contenu d'une fonctionnalité à développer

¹⁰ Un sprint est une boîte de temps (time box) contenant des user stories

¹¹ Deadline est un mot anglais pouvant signifier en français date limite, échéance

8. Développement technique

8.1 DEMARCHE DU PROJET

Le point de départ du projet a été une étude de l'existant et du souhaité. Le résultat de ces recherches et d'une étude de faisabilité technique a permis de définir les frontières et les limites à se fixer durant ce projet.

Cette recherche est une partie intégrante du projet. Elle a permis de structurer la demande du client. A partir de ce cahier des charges, les différentes parties du produit ont été développées.

Certaines fonctionnalités du site ne seront pas implémentées à cause de la contrainte temporelle imposée. Cependant le développement de la solution a pris en compte la nécessité de réaliser ces évolutions.

Le développement de tous les modules constituant le projet a été réalisé pour permettre des modifications et améliorations futures. Du fait de cette logique de développement, il sera tout à fait possible de changer le front-end indépendamment du backend et inversement.

8.2 CONCEPTION DE LA BASE DE DONNEES

La base de données est d'une taille importante du fait du nombre de fonctions qui sont demandées par le client. La priorité a été donnée à la clarté, l'optimisation ainsi qu'à l'efficacité plutôt qu'au nombre de table. La base de données est constituée de 17 tables permettant de supporter toutes les fonctionnalités qui ont été demandées par le client. Ces fonctionnalités ne sont pas toutes implémentées mais la base de données permettra leur développement.

Lors de la fin du projet et après une utilisation de deux semaines environ on peut faire le postulat suivant :

- 160 utilisateurs
- 300 choix
- Une quarantaine de clubs
- Une centaine membres de clubs répertoriés et sauvegardé

Le projet se référant à un historique d'actions et de projets associatifs effectués une table pour la temporalité a été intégrée permettant ainsi de garder un historique avec une précision annuelle.

Il a alors fallu gérer les informations des différents clubs en ajoutant en supplément une dimension temporelle. Cette dimension supplémentaire à ajouter un niveau de difficulté qui a pu être résolu par une profonde réflexion et recherche sur la base de donnée.

Le modèle conceptuel de la base de données se trouve en annexe. C'est à partir d'un modèle conceptuel comme celui-ci que l'on peut construire une base de données.

8.3 MISE EN PLACE D'UNE ARCHITECTURE

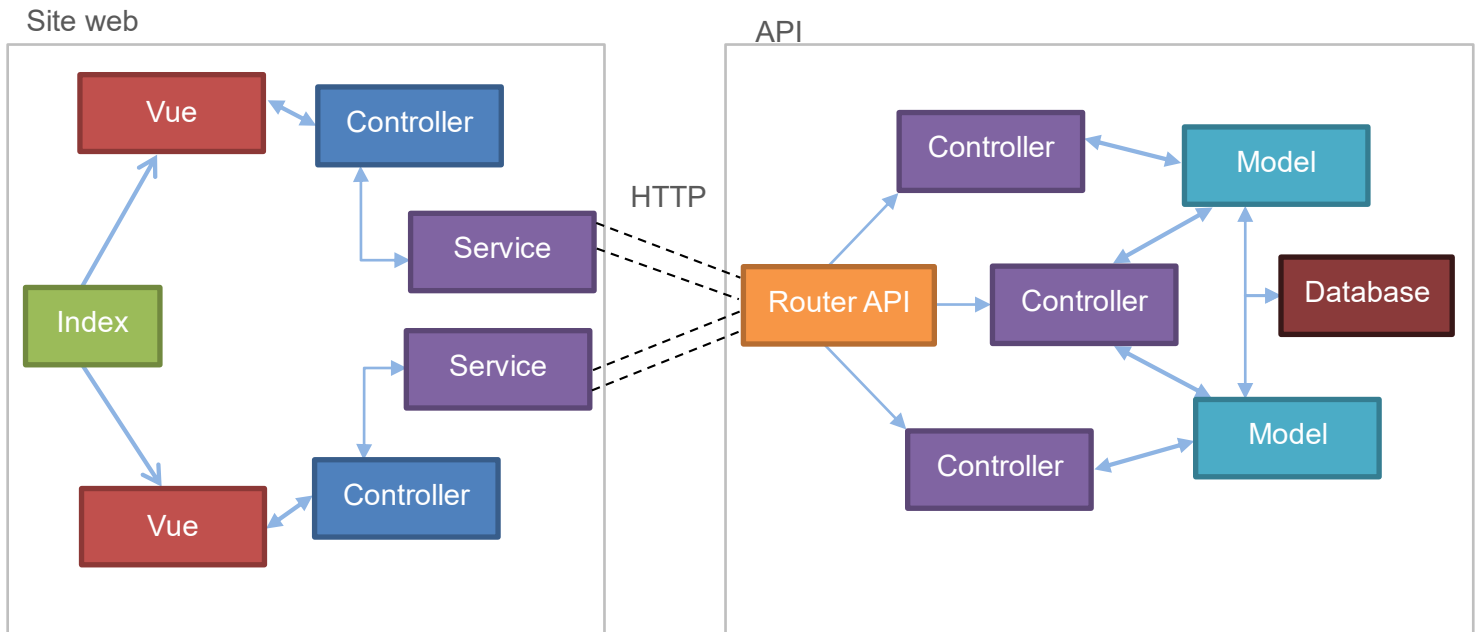


Figure 18 - Schéma de l'architecture de l'application

8.4 MISE EN PLACE D'UNE API REST

Une API compatible REST est une interface de programmation d'application qui fait appel à des requêtes HTTP existante pour obtenir, modifier, créer et supprimer des données. L'action demandée est basée sur l'entête comportant un mot du type GET, PUT, POST ou DELETE.

8.4.1 Justification technologique

Les avantages d'une telle solution est la segmentation des différents blocs constituant un site web. Cela permet de garder une indépendance entre front-end et back-end. L'utilisation d'URI permet un système universel d'identification des éléments de l'application.

8.4.2 Implémentation

Les quatres actions citées précédemment sont les principales implémentées dans une API REST. Ce sont elles qui ont été mises en place dans le projet.

Les requêtes sont utilisées comme ceci :

- GET permet de récupérer une information
- PUT permet de modifier une information
- POST permet de créer une nouvelle information
- DELETE permet de supprimer une information

D'un point de vue sécurité chaque requête donne lieu à une fonction vérifiant les autorisations de la personne. Cette pratique permet d'avoir l'architecture la plus simple possible. Durant le projet il a été primordiale de tenir cette ligne directrice lors de la conception de l'application, par exemple

GET /api/clubs/

Va retourner la liste des identifiants de tous les clubs renseignés dans l'application.

L'API est constituée d'un ensemble de routes de différents types et classée en fonction de la ressource avec laquelle elle interagit. Il sera ainsi plus simple de reprendre le projet.

Une API REST est facile à utiliser car les outils de test peuvent être un navigateur web ou un logiciel équivalent à curl pour envoyer différents types de requêtes.

Du côté client, AngularJS, le framework utilisé pour concevoir le site, permet une utilisation native de ce type d'API via des services. L'application a été conçue pour que 90 % des requêtes vers le serveur soient effectuées par des services. Toutes modifications au niveau de l'API ne nécessiteront donc que la modification du service correspondant.

8.4.3 Format de réponse

Il a été convenu que chaque requête, même non autorisée ou défailante, donnera lieu à une réponse au format JSON. L'intégration des informations renvoyées est ainsi plus facile en JavaScript.

8.5 MISE EN PLACE DE CONTROLLER ET DE MODELE

L'intégration d'une architecture orientée MVVM, préconise la création de Modèles et Contrôleurs.

Les modèles

L'application incorpore des objets PHP servant de modèle, et chaque modèle correspond à une structure (table) dans la base de données. Chaque attribut de la classe correspond à un champ de la base de données. Des méthodes génériques permettant de sauvegarder les modifications dans la base, de charger des informations et de récupérer une liste d'objets existants. Ces modèles font appel à un « Singleton »¹², celui-ci permet de garantir un nombre limité de connexion avec la base de données.

8.5.1 Les contrôleurs

Les contrôleurs permettent de segmenter les fonctionnalités de l'API, par exemple, toutes les actions relatives aux clubs se trouvent dans le contrôleur « Club.php ».

Chaque contrôleur est un ensemble de fonctions publiques ou privées. Le routeur, en fonction de la requête, appellera la fonction correspondante en lui transmettant les potentiels paramètres de l'URL.

C'est le contrôleur qui est en charge de :

- Journaliser les actions importantes (création de club, lancement de l'affectation automatique, etc.)
- Écrire la réponse formater en JSON au client
- Bloquer la requête de l'utilisateur si l'action demandée ne lui est pas permise.

8.6 MISE EN PLACE D'UN FRONT-END ET DES FONCTIONNALITÉS

Le choix de front-end s'est reposé sur les demandes du client qui voulait une interface claire, facile à prendre en main, dynamique et dotée d'un « design » élégant.

8.6.1 Aspect graphique

Le choix du design a été une question primordiale car il fallait quelque chose d'esthétique et facilement compréhensible. Un design structuré et si possible connu est donc nécessaire. Subséquemment, la possibilité de visualiser l'application sur plate-forme de type téléphone ou tablette est un objectif secondaire. Plusieurs choix de graphisme ont été identifiés et observés. Le choix graphique conservé est le « Material Design ». Vous pouvez voir la vue de notation pour l'évaluateur ci-dessous.

¹² Patron de conception utilisé afin d'obtenir une instanciation unique d'un objet.






Gestion des clubs

Cette page permet

- Nommer un p
- Valider ou no
- Rendre les cl
- Noter les club

TÉLÉCHARGER TOUTES MES NOTES

Mes clubs actifs

- 
Asso. sociale A.F.E.V.
3 membre(s)
- 
Bureau de l'international
0 membre(s)
- 
Bureau des Arts
4 membre(s)
- 
Bureau des sports
4 membre(s)
- 
Bureau des élèves
1 membre(s)

Capisen

Nommer un président

Président

Maxime Goanvic

AJOUTER

Noter les clubs ?

Validation des projets

Note du Club

14

<input checked="" type="checkbox"/>	PR+	Erwan Joly	
<input checked="" type="checkbox"/>	PR+	Maxime Goanvic	Note
<input checked="" type="checkbox"/>	PA	Antoine Le GUERN	14
<input checked="" type="checkbox"/>	PA	Loïs Demeocq	Note
			14

SAUVEGARDER

DÉVERROUILLER LES PROJETS VERROUILLER LES NOTES

Figure 19 - Page de notation et de validation de projet pour évaluateurs

Le Material Design est le design développé par Google. Ce design est implémenté sur tous les produits de google que ce soit Youtube ou tout simplement sur l'OS Android. De ce fait, il est connu par de nombreuses personnes et sera rapidement pris en main.

Les utilisateurs de ce design seront les élèves de l'ISEN ou les évaluateurs (professeurs), tous dont une connaissance au moins succincte de ce type de présentation et de logique de navigation.

Il a tout de même été décidé de procéder à des modifications mineures dans le design. Een effet les textes descriptifs ont été privilégiés à l'utilisation d'icônes. Ainsi même un utilisateur néophyte d'application web ou de Material Design pourra comprendre le site. Les boutons déclencheurs d'actions sont clairement identifiés et la logique d'actions y a été réfléchie.

8.6.2 Framework

Un framework de développement a été utilisé afin de produire un résultat de qualité en un temps minimum. L'équipe de développement ne possédant pas de compétences poussées en web design, il s'agit là d'un choix visant à maximiser la production.

La contrepartie est l'apprentissage de ce framework coutant un temps raisonnable par rapport à un création complète de design.

8.6.3 L'Expérience utilisateur

L'utilisateur est guidé dans les différentes actions qu'il voudra exécuter. Les pages qui semblaient peu explicites ont été décrites dans un en-tête. On y retrouve les actions possibles sur la page actuelle et parfois un avertissement sur les fonctions critiques ou définitives. Pour rendre l'application claire et rapide d'utilisation. Une attention particulière a été portée à la qualité des messages de l'IHM.

Cette solution est une nouveauté puisque la solution en place jusqu'à présent était purement humaine et manuscrite. Cette application web vient prendre la suite de la méthode utilisée jusqu'à aujourd'hui, la solution doit donc s'inscrire comme référence. Pour faciliter cette transition et uniformiser les changements et méthodes, des boutons d'informations ont été mis à disposition dans le produit. L'utilisateur sera en mesure de se renseigner sur les différentes étapes d'affectations de club ou encore sur les clubs.

L'interface est fluide, au design épuré et dirige l'utilisateur à l'aide de mise en avant d'informations.

The screenshot displays the ISEN web application interface. On the left is a dark sidebar with the ISEN logo and a menu including 'Informations Globales', 'Candidature', 'Mes informations', 'Club Elec', 'Administration de "Club Elec"', 'Notation', 'Année prospective' (highlighted in red), and 'Gestion du club'. The main content area has a top navigation bar with 'Liste des clubs', 'Ce club', and 'Evaluation'. Below this, a section titled 'Recommander un postulant non ancien membre du club pour 2016/2017' contains a search bar and a '151 Utilisateurs' indicator. The primary section is 'Demande d'effectifs pour 2017', which features a table for setting minimum and maximum values for different roles. A 'SAUVEGARDER' button is located at the bottom of this section.

	minimum	maximum
PR+	10	12
PR	8	10
PA	8	10
PI	0	2

Figure 20 - Page de demande d'effectifs pour les présidents de club

Fonctionnement de la notation d'un club

La notation d'un club se fait en 5 temps:

Le président du club va annoncer qui, selon lui, a validé ou non son projet

Vous allez pouvoir modifier ses choix s'ils ne vous conviennent pas, puis les verrouiller afin qu'ils ne puissent plus être changés

Vous allez également donner une note au club

Le président va pouvoir redistribuer les points aux membres du club

Vous allez ensuite pouvoir modifier ces notes si elles ne vous semblent pas justifiées. Vous pouvez aussi les verrouiller tout de suite après avoir donné une note au club, afin que tout le monde ait la même note

Les PA valident obligatoirement leur projet, peu importe leur note. Un PR validé a forcément au moins 10 tandis que s'il est invalidé il a forcément moins de 10. Enfin, un PR+ n'a pas de note, seule la validation rentre en compte.

J'AI COMPRIS

Figure 21 - Exemple d'explication du fonctionnement de la page

8.6.4 Une portée informative

La communication d'informations sur les clubs est d'une grande importance, notamment pour les Isennniens de Rennes qui ne connaissent pas les clubs de Brest avant d'arriver dans les locaux Brestoï. Ainsi, ils seront capables de se renseigner sur les activités d'un club, son domaine de compétences et postuler pour les clubs en même temps que les brestoï. De ce fait, cette application joue un rôle important dans l'intégration des Rennais sur le site de Brest. Au-delà de l'intégration de nouvelles personnes dans les clubs, l'interface devra pouvoir permettre un historique des projets associatifs.

Nous avons mis en place des vues permettant de rechercher des données d'années passées. Les clubs nous ont demandé cette fonctionnalité afin de pouvoir retrouver les membres de certaines années et pouvoir les contacter. C'est de cette même manière que les administrateurs ou évaluateurs seront capables de visualiser les informations en suivant une certaine temporalité. Ces informations et notamment les informations personnelles ont été traitées avec la plus grande attention.

8.6.5 Traitement des données personnelles et sécurité

Dans l'optique d'une transparence et une explication du système de la solution, il est spécifié que les données sont utilisées dans un cadre pédagogique et en aucun cas dans un but de profit quelconque.

L'administration des informations personnelles est disponible et permet à l'utilisateur de les supprimer s'il ne souhaite pas que ses données soient communiquées. La gestion des données est un point clef de la solution et passe par plusieurs étapes : la base de données, le backend, puis par le front-end.

Ces étapes permettent non seulement un traitement des données mais également la vérification de leur intégrité et détecter les tentatives de fraudes.

La vérification s'exécute donc aussi bien du côté client que du côté serveur. Subséquemment, toutes les fonctions ne sont pas disponibles pour tous les utilisateurs. La vérification de droit se fait une nouvelle fois du côté client comme du côté serveur. Cette vérification est indispensable du fait de la sensibilité des données traitées.

Il n'est pas concevable que n'importe qui puisse modifier sa note ou celle d'une autre personne. Seul l'évaluateur du club et son président ont le pouvoir de valider/invalider des projets associatifs et de noter. Le président propose des validations de projet et par la suite la répartition de points internes au club tandis que l'évaluateur validera ou non et donnera une note globale au club. L'évaluateur reste la force de décision puisque la validation est de son ressort. Ce système-là est une retranscription

informatique du processus physique qui était jusqu'à présent en place. Le système convenait à tous les utilisateurs, c'est pour cela qu'il a été conservé et que les seules démarches réalisées ont été de le rendre plus dynamique et fluide.

8.6.6 Prise en compte des exceptions

Une demande spécifique a été traitée pour les clubs dont le recrutement est spécifique (Bureau des élèves, Capisen, ...) ; les affectations se font aux travers de recrutement ou de liste d'élus. La Junior Entreprise a la particularité de choisir parmi la liste de candidatures pour son club les étudiants qui rejoindront le club.

L'administrateur peut affecter des élèves à des clubs et même leur spécifier le type de projet associatif qu'ils vont réaliser. Les personnes ainsi placées seront assurées de faire partie du club l'année suivante. Cependant, placer manuellement un élève lui enlèvera ses choix précédents de façon définitive, ainsi il ne sera pas traité par l'affectation automatique.

L'algorithme de répartition automatique sera par la suite déclenché. Son fonctionnement est expliqué dans la partie « Algorithme de répartition » ci-après.

8.7 DEROULEMENT DES TESTS

Les tests ont été réalisés suivants plusieurs « process ¹³ » et paramètres.

Tout d'abord, les tests de vérifications syntaxiques sont automatiques et ce déclenchent à chaque sauvegarde d'un des fichiers du code source de l'application. Ils ont été réalisés à l'aide de Gulp. Les langages utilisés se basent sur un schéma d'indentation et de syntaxe très particulier. Il est nécessaire de vérifier le respect des règles pour éviter tout dysfonctionnement résultant d'une erreur provenant de là.

Un autre type de test automatique mis en place avec Travis CI a permis de tester les bons paramètres de l'application pour la mise en production. En effet, lorsque le projet est sauvegardé sur le serveur Github une liste de tests unitaires s'exécute et une alerte est envoyée en cas de problèmes. Ainsi à chaque sauvegarde, il est possible de savoir si le déploiement sera réalisable.

Le test des fonctionnalités en revanche est fait manuellement. Pour chaque implémentation de fonction une liste de critères et limitations a été établie. Une fois cette liste confirmée, la fonction est développée. Une fois la fonction réalisée, la liste est réutilisée pour confirmer le bon résultat de la fonction. Chaque cas de figure est imaginé et testé. En cas de mauvais résultats, un correctif est apporté et la liste de tests est reprise depuis le départ. Il est ainsi possible de s'assurer que, dans une phase de test spécifiquement consacré à un module, la fonction produit le bon résultat. Une fois cette validation réalisée, le module est intégré au projet complet. Grâce à cette méthode, il est possible de limiter les dysfonctionnements à des problèmes d'intégration.

Avant toute mise en production une succession de tests mettant en place une grande quantité de scénarii possibles a été réalisée afin de garantir le bon fonctionnement du produit. Les tests se déroulent suivant plusieurs plans.

Il faut tester toutes les fonctionnalités pour les différents profils d'utilisateurs :

- Administrateur
- Evalueur
- Président de Club
- Membre de Club

¹³ Ensemble d'étapes permettant d'aboutir à un résultat

- Elève demandeur de club
- Profils spéciaux (BDE, Capisen)

Pour chacun des profils ci-dessus chaque vue est testée une à une avec une vérification du bon comportement de l'interface utilisateur, des fonctions côté serveur et de la prise en compte dans la base de données. Faire cela permet de valider les fonctions dans leurs contextes mais pas dans leur situation de vie. C'est pourquoi des tests **en simulation de situations** réels ont été pratiqués avec des modifications en fait par plusieurs utilisateurs différents en parallèle. Une fois ce type de test réalisé et validé, on peut se permettre de faire une mise en production.

8.8 MISE EN PRODUCTION INITIALE

Cette mise en production va permettre d'observer le comportement des utilisateurs mais également de l'interface en situation réelle. C'est à partir des résultats que de futures évolutions du projet pourront être mis en place.

Déployer tôt lors de la phase de développement, la première version permettait seulement aux élèves de sélectionner trois choix, correspondants aux clubs qu'ils souhaitaient intégrer l'année suivante.

8.8.1 L'objectif

Le but de cette mise en production est de remplir la base de données avec des informations réelles pour nous permettre de tester la suite du projet.

Une vérification de la sauvegarde des informations utilisateurs sera nécessaire.

Une vérification complète de la solution pourra être effectuée dans son environnement final.

8.8.2 La plate-forme

L'application a vocation à être déployée sur une machine de type UNIX, comme c'est le cas sur la machine virtuelle fournie par l'école. L'application étant développée sur Windows peut être considérée comme cross-plateforme¹⁴.

8.8.3 La méthode de déploiement

Le déploiement est effectué à l'aide de « Dploy », un utilitaire fourni par NodeJS. Son principal atout est d'utiliser le protocole SFTP, il s'agit d'un protocole sécurisé pour téléverser les nouveaux fichiers sur le serveur. Il ne téléverse que les fichiers modifiés, la masse de fichiers en transit est donc diminuée. Cette utilitaire permet de déployer notre application de façon continue (Continuous Integration). L'avantage de cette méthode est de permettre de mettre à jour le serveur de production en fonction des « commits ¹⁵ » effectués sur Git. Actuellement, il est donc possible de diffuser n'importe quelle version du projet, permettant ainsi de déployer une version spécifique de l'application.

8.8.4 L'environnement sécurisé

Le serveur actuel de distribution Debian possède un serveur web sous Apache2, un serveur MySQL en version 5.7.9 et l'utilitaire PHP v5.6.20. L'installation est identique sur nos postes personnels pour le développement et sur le serveur afin de faciliter l'intégration de la solution. L'accès au serveur s'exécute par SSH, donc par connexion sécurisée.

Pour la gestion de la base de données un utilisateur a été créé sur le serveur. Cet utilisateur peut exécuter des modifications mais uniquement sur la base de données de l'application. Grâce à cet

¹⁴ Se dit d'une application pouvant être utilisée sur plusieurs plateformes (LINUX, Windows, Mac, ...).

¹⁵ Désigne l'enregistrement effectif d'une transaction

utilisateur il est possible de se connecter à la base de données via une application externe (DataGrip a été ici utilisé). Au travers de cette gestion il a été possible de gérer les informations du projet.

Sur nos machines de développement, l'application se réfère à un fichier de configuration. Par souci de sécurité ce fichier n'est pas envoyé sur le serveur GitHub pour ne pas diffuser ces informations confidentielles. L'application se base sur des variables d'environnement, si celles-ci ne sont pas présentes le fichier de configuration sera utilisé.

Cette pratique permet de ne pas diffuser d'informations sensibles. De plus, cela permet de ne pas avoir de modification de configuration entre les phases de développement et de mise en production.

8.8.5 L'application en elle-même

Comme énoncé précédemment, l'application utilise des langages de plus haut niveaux que ceux utilisés en pratique (Jade, Coffee, Sass). Gulp permet donc de transpiler ces langages en HTML CSS et JS. Ce sont ces fichiers une fois « transpilés » qui sont envoyés et utilisés. Les sources de développement de l'application ne sont donc pas présentes sur le serveur, cela évite la présence de fichiers non essentiels et limite le transit de fichier. Ne pas diffuser et maintenir des fichiers source sur le serveur présente divers intérêts. L'un d'entre eux étant de ne pas utiliser de l'espace disque supplémentaire. Sur des solutions professionnelles, l'utilisation de quelques méga octets peut avoir d'importantes répercussions.

Les données retirées de cette mise en production initiale permettront de constituer une base de données de test plus réaliste que ce qui était utilisé jusqu'à présent. À l'aide de ces informations récupérées le développement de l'algorithme de répartition devient possible et pourra se reposer sur des informations concrètes.

8.9 ALGORITHME DE REPARTITION

Le développement de l'algorithme de répartition des élèves a représenté une partie importante du projet. La tâche est composée d'un grand nombre de variables à prendre en compte.

8.9.1 Les conditions

Il a fallu définir les différentes conditions à prendre en compte dans cet algorithme, proposer des solutions et faire valider notre proposition. Une fois ce choix effectué, une réalisation de la logique de la répartition a été réalisée.

Les conditions à prendre en compte sont :

- Validation du précédent projet associatif
- Définition du projet à effectuer
- Ancienneté dans le club demandé
- Recommandation de l'élève pour ce club
- Elève non voulu pour l'année suivante
- Affectation faite par un administrateur
- Affectation réalisée par le président de Capisen (Junior Entreprise)
- Le numéro du choix du club

Toutes ces conditions sont à prendre en compte pour le bon déroulement de la répartition automatique.

Le schéma simplifié du fonctionnement de la répartition automatique se trouve en annexe 1.

8.9.2 Calcul d'un score

Il a été décidé que l'attribution de points à chaque élève pour chacun des clubs qu'il a demandé sera utilisé. Un tableau d'ordre de points a donc été réalisé et représente la base du fonctionnement de

l'algorithme. Un cas spécifique est à prendre en compte, la demande de clubs à recrutement spécifique doit s'effectuer en premier choix et les élèves ne bénéficient d'aucun bonus/malus de points. Voici le tableau récapitulant l'attribution des points :

Assignment de points		
	Clubs classiques	Club à répartition spécifique
1^{er} choix	5	Ø
2^{ème} choix	3	
3^{ème} choix	1	
Ancien membre	3	
Recommndation	2	
Malus(non recommandation)	-2	

Des subtilités sont à prendre en compte. Un ancien membre ne peut pas être recommandé et lorsqu'il a un malus il perd en plus ses points d'ancienneté et passe donc de +3 à -2 (soit moins cinq points). La recommandation est mise en place pour les élèves qui ne sont pas membre du club mais qui voudraient le rejoindre.

S'ils ont témoigné un intérêt suffisant, le président peut alors les recommander pour augmenter leur chance de rejoindre le club en question.

Le calcul de ce score repose sur la capacité du code à détecter ces critères. La fonction de répartition va donc intégrer des fonctions qui vont permettre de savoir pour chaque élève, pour chaque club s'ils ont été recommandés, si ce sont des anciens ou s'ils ont un malus de points.

Ces points seront ajoutés au score attribué au choix du club. Ce score est renvoyé par une simple fonction, avec « i » représentant le numéro du choix : $-2*i+7$. Le résultat renvoyé, additionné aux conditions, donne un score pour un élève, pour un club, qui servira pour sa répartition.

8.9.3 Effectif demandé par les clubs

Chaque club doit fournir un effectif minimum et maximum pour chaque type de projet associatif (PI, PA, PR, PR+). Si le club ne fournit aucun effectif, l'algorithme en déduira que le club ne souhaite pas de membre dans pour ce type de projet.

L'effectif demandé est donc important, il permet d'avoir un club de la taille souhaitée et non surdimensionné par rapport aux attentes et objectifs du club.

8.9.4 Effectif de club calculé

Afin de ne pas avoir une demande plus forte que l'offre, un système de ratio est utilisé. En effet si le nombre maximum d'effectif demandé pour un certain projet associatif (par exemple 50) est inférieur à la demande, un système d'équilibrage est utilisé. Un coefficient multiplicateur est calculé (ici il sera de 2), une multiplication de la demande d'effectif de chaque club pour ce type de projet sera réalisée avec un arrondi au supérieur.

Le même type de calcul est fait pour le minimum afin d'assurer une répartition optimale pour chaque club.

8.9.5 Détection du projet

Afin de répartir les élèves dans les bons clubs avec les bons projets il faut que l'algorithme puisse définir le projet demandé par l'utilisateur. Un ordre dans les projets associatifs existe, il s'agit de :

PA suivi de PR lui-même suivi de PR+

Une fonction permet donc de trouver quel est le dernier projet réalisé et validé. Ainsi, s'il n'en trouve aucun, l'utilisateur doit effectuer un PA, s'il trouve un PA ce sera un PR, etc. En revanche, s'il trouve que le dernier projet validé est un PR+, l'élève ne sera affecté dans aucun club.

Le choix de clubs pouvant s'effectuer avant la validation des projets il est nécessaire de faire cette vérification. Un ancien PR+ pourra en revanche toujours échanger avec le club mais ne sera pas automatiquement réparti car il n'existe pas de projet type PR++.

8.9.6 Optimisation du code

La logique de programmation est faite pour optimiser au maximum le temps de calcul des différentes informations.

Lorsque l'on pratique des boucles imbriquées pour rechercher une information, une fonction d'interruption quitte ces boucles lorsque l'information souhaitée est trouvée. Le temps d'exécution est donc réduit.

De plus, le calcul des points se fait pour tous les premiers choix des élèves mais ne sera calculé pour les choix suivants que si les utilisateurs ne sont pas admis au premier tour. De cette manière les ressources à allouer sont moins importantes, tout comme le temps d'exécution.

8.9.7 Ordre de répartition

Les membres placés par l'administrateur sont prioritaires et sont assurés d'être placés là où l'administrateur le demande.

L'administrateur devra placer les membres du BDE et du BDS. Les membres de Capisen seront placés par le président du club.

L'algorithme a été testé le 20 mai pour s'assurer de son bon fonctionnement sur des données réelles.

Pour faire la répartition automatique de 151 personnes (nombre de personnes demandant un club à l'heure actuelle) il a fallu 3.1596s. Sur les 151 demandes 147 ont été placées avec succès. En temps normale le BDE et le responsable des clubs se réunissent pour faire cette répartition. La constitution des effectifs des clubs dure environ deux heures et demi, avec cette fonctionnalité 97.3 % des personnes sont placées en 0.03 % du temps habituel.

8.9.8 Attente de validation

Les personnes attribuées à des clubs sont enregistrées en base de données mais ne sont pas définitivement enregistrées tant que l'administrateur n'a pas validé ces affectations. S'il valide le placement, les choix faits par les élèves et les effectifs demandés par les clubs sont supprimés. Ces données ne sont pas des informations utiles à conserver et sont volumineuses. Ainsi, la base est en état pour l'année suivante.

8.9.9 Retour de la fonction de répartition

La fonction de répartition retourne un objet JSON contenant les informations sur les personnes qui n'ont pas été placées. L'administrateur devra les placer manuellement puis confirmer la répartition. Il peut également décider d'attendre de prochaines modifications et refaire une répartition automatique. La fonction est capable de se réinitialiser pour refaire une répartition complète en prenant donc en compte de nouvelles variables.

8.10 MISE EN PRODUCTION FINALE

Une mise en production a été faite le 19/05 pour la livraison finale du produit. Les fonctionnalités développées pour chacun des groupes d'utilisateur ont été implémentées en grande partie.

8.10.1 La migration de la base

D'une taille plus importante que dans la précédente mise en production, la base de données a nécessité la mise en maintenance du site.

La première étape a été d'importer les données de la base de production afin de les intégrer dans notre fichier SQL. Ainsi toutes les données ont pu être conservé en l'état.

Dans un second temps, un travail a été nécessaire pour associer chaque élève à sa classe dans la base de données. Ce travail est maintenant automatique.

Enfin, cette année étant la première où un support numérique est utilisé pour gérer l'effectif des clubs, les différents membres de club ont été affectés manuellement dans leur club. Cela permettra d'effectuer la notation d'ici la fin de l'année.

8.10.2 Mise à jour des fichiers

La phase finale de mise en production s'articule autour du téléversement des nouveaux fichiers. Grâce à Dploy, cette tâche fut rapide, en quelques secondes seulement les nouveaux fichiers et ceux modifiés ont été mis en place.

Les tests en fin de maintenance ont révélé quelques problèmes en rapport avec les chemins d'accès à certains fichiers. Une autre erreur venant de la production de fichier Excel sous Linux a également dû être réglée. Cette erreur provenait de la compatibilité de la librairie avec Linux, il a pu être réglé à l'aide d'une fonction de nettoyage du « buffer ¹⁶».

Il a fallu résoudre ces erreurs avant de finaliser la mise en production.

8.10.3 Le suivi

Depuis cette dernière action, quelques correctifs ont été apportés de façon transparente pour l'utilisateur.

Une analyse manuelle des logs d'Apache2 est effectuée chaque matinée, ainsi qu'une revue des logs de l'application.

Cela a permis de révéler une erreur pour un utilisateur. Il est alors possible de le contacter pour avoir plus d'informations. Une activité suspect a également été constaté, permettant ainsi de vérifier les actions des utilisateurs en cas de tentatives de fraudes.

8.11 INTEGRITE DES DONNEES

La sécurité et l'intégrité des données sont des aspects primordiaux du projet. Les informations issues du CAS de l'ISEN doivent rester confidentielle ainsi que les données que l'utilisateur fournira dans l'application. Cette ligne directive s'est exprimé sous différentes formes.

Au niveau applicatif, très peu de nos modèles intègrent une fonctionnalité de suppression. Seule les données secondaires, tel que les trois choix de chaque utilisateur peuvent être supprimés depuis l'interface.

Parmi les possibilités de suppression, des sécurités ont été ajoutées, en fonction du groupe de l'utilisateur. Cette vérification réduit considérablement les possibilités de corruption de données.

Enfin au niveau système, la base de données est sauvegardée tous les jours à 12 h 30. Le format des sauvegardes est un fichier SQL. Plusieurs intérêts à ce type de fichier, il permet de retourner sur une

¹⁶ Mémoire tampon, une zone de mémoire virtuelle ou de disque dur utilisée pour stocker temporairement des données

base de données saine en cas de dysfonctionnement. La migration de l'application sera simplifiée en cas de changement pour une architecture orientée micro-service ou en cas de migration du serveur de base de données sur un serveur dédié.

L'archivage de ses sauvegardes se fait sous forme d'archives compressée de type GunZip. Stocker les sauvegardes dans un format compressé évite une utilisation excessive d'espace disque.

Il conviendra aux futurs administrateurs de l'application de récupérer manuellement ou automatiquement les sauvegardes. Ces sauvegardes concernent les données utilisateur et le dossier contenant les dossiers de passation, le but étant d'éviter une perte ou une corruption des données.

9. Conclusion

Le projet fût complexe est très long du fait de la charge de travail importante.

La prise en compte des demandes des différents types d'utilisateurs afin de construire un outil interactif, facile d'utilisation et pleinement fonctionnel était un défi conséquent, surtout dans le temps imparti.

L'objectif était de proposer une solution rapidement accessible mais surtout de pouvoir être maintenu et poursuivi par une future équipe de M1. Les technologies ont donc été choisies de façon à ne pas être trop complexe pour des néophytes afin d'optimiser la rapidité de compréhension du projet.

Suite à une dernière réunion avec le client, il semblerai que ses attentes principales soient atteintes. La mise en situation pourra confirmer son opinion sur la solution proposée déjà utilisé pour la demande d'affectation.

Le projet a pris une envergure très importante lors de la recherche détaillée du besoin. Le sujet initial paraissait succinct mais une fois le besoin décrit la charge de travail est devenu importante.

Il reste pour notre part deux sentiments, l'un de fierté devant le travail accompli, l'autre de frustration devant les fonctionnalités non implémentées.

L'application est actuellement en ligne et fonctionnel. L'efficacité ainsi que sa sécurité a été et sera mise à l'épreuve lors de la notation des clubs pour l'année 2015/2016 le jeudi 26 mai (date ultérieure à la rédaction de ce rapport).

Nous continuons de gérer le support pour les différents utilisateurs mais il ne semble pas y avoir de problèmes majeurs.

Les fonctionnalités et possibilités encore implémentables lié à une étude d'évolutivité du besoin pourra certainement donner suite à au moins un projet M1 l'an prochain.

10. Remerciements

Nous remercions tous les professeurs, les présidents de club et l'équipe FHS qui nous ont aidés au long de notre projet en testant l'application ou en donnant leurs avis afin de construire l'interface la plus intuitive et la plus adaptée aux besoins de l'école.

Nous remercions M. Cabon pour la mise en situation professionnelle dans laquelle il nous a placés, afin de mieux nous préparer aux contextes difficiles des projets en entreprise. Nous le remercions tout particulièrement pour le temps qu'il a consacré à nous guider dans la réalisation du projet. Il nous a également conseillé pour sa structuration et sa gestion.

Nous remercions également M. Le Foll pour ses conseils sur notre base de données et pour les corrections qu'il nous a signalées. Nous avons pu réutiliser le cours de base de données qu'il nous avait dispensé en seconde année, permettant ainsi une bonne appropriation de ce domaine de compétences.

Pour réaliser une mise en production, nous avons eu besoin du support de MM. Gerval et Le Ru, et des conseils qu'ils nous ont prodigués. Leur soutien et la mise en œuvre des solutions proposées nous ont été d'une grande aide.

11. Glossaire

Abréviation	Désignation	Définition
API	Application Programming Interface	Interface de programmation applicative
BDE	Bureau des élèves	Association ayant pour but de représenter les élèves devant l'administration scolaire
BDS	Bureau des sports	Association sportive de l'ISEN
CSS	Cascading Style Sheets	Feuille de style. Ensemble de règles permettant de mettre en place le design d'un site
ENT	Environnement Numérique de Travail	Ensemble d'outils en ligne disponible pour les élèves, enseignants et l'administration
FHS	Formation Humaine et Sociale	Cours dispensé à l'ISEN de Brest
HTML	Hypertext Markup Language	Format de données conçu pour représenter les pages web
HTTP	Hypertext Transfer Protocol	Protocole de communication client-serveur développé pour le World Wide Web
IHM	Interface Homme-Machine	Ensemble de moyens et outils mis en œuvre afin qu'un humain puisse contrôler et communiquer avec une machine
JSON	Javascript Object Notation	Format de données textuelles dérivé de la notation des objets du langage JavaScript
MVP	Minimum viable product	Stratégie de développement de produit, utilisée pour de rapides et quantitatifs tests de mise sur le marché d'un produit ou d'une fonctionnalité. Ici, il définit les fonctionnalités minimum attendues par le client
MVVM	Model-view-view-Model	Architecture particulière appliquer à la conception d'applications web
OS	Operating System	Logiciel principal d'un ordinateur, il gère les différentes applications lancées par l'utilisateur (exemple : Windows)
PA	Projet d'Accompagnement ISEN	Appuyer l'action d'un PR sur des tâches semi-complexes en autonomie partielle
PI	Projet d'Intégration ISEN	Se mettre au service d'un groupe ou d'une association de façon ponctuelle
PR	Projet à Responsabilité ISEN	Prendre des responsabilités, assurer l'encadrement et la prise de décision dans les activités d'un groupe ou d'une association
PR+	Projet à Responsabilité + ISEN	Transmettre son savoir-faire, son expérience
REST	Representational state transfer	Style d'architecture pour les systèmes hypermédia distribués
SASS	Syntactically Awesome Style Sheets	Pseudo-langage simplifiant la rédaction de règles CSS
SFTP	SSH File Transfer Protocol	Protocol de transfert de fichier basé sur le protocole de communication sécurisé SSH
SGBD	Système de Gestion de Base de Données	Logiciel de gestion de base de données
SGBDR	Système de Gestion de Base de Données Relationnel	Logiciel de gestion de base de données
SQL	Structured Query Language	Langage informatique normalisé servant à exploiter des bases de données relationnelles

12. Table des figures

Figure 1 - Bête à cornes	7
Figure 2 - Diagramme pieuvre	9
Figure 3 - Logo de Github	16
Figure 4 - Logo de MySQL	16
Figure 5 - Logo de PowerAMC	16
Figure 6 - Logo de Travis	16
Figure 7 - Logo de Gulp	17
Figure 8 - Logo de SQL	17
Figure 9 - Logo de PHP	17
Figure 10 - Logo de npm	18
Figure 11 - Logo de AngularJS	18
Figure 12 - Logo de Jade	18
Figure 13 - Logo de CoffeeScript	18
Figure 14 - Logo de Sass	18
Figure 15 - Logo de JavaScript	19
Figure 16 - Logo de CSS3	19
Figure 17 - Logo de HTML5	19
Figure 18 - Schéma de l'architecture de l'application	22
Figure 19 - Page de notation et de validation de projet pour évaluateurs	24
Figure 20 - Page de demande d'effectifs pour les présidents de club	25
Figure 21 - Exemple d'explication du fonctionnement de la page	26

13. Webographie

www.angularjs.org => Site officiel de la documentation d'AngularJS

www.material.angularjs.org => Site officiel de la documentation d'Angular-Material

www.php.net => Site officiel de la documentation de PHP

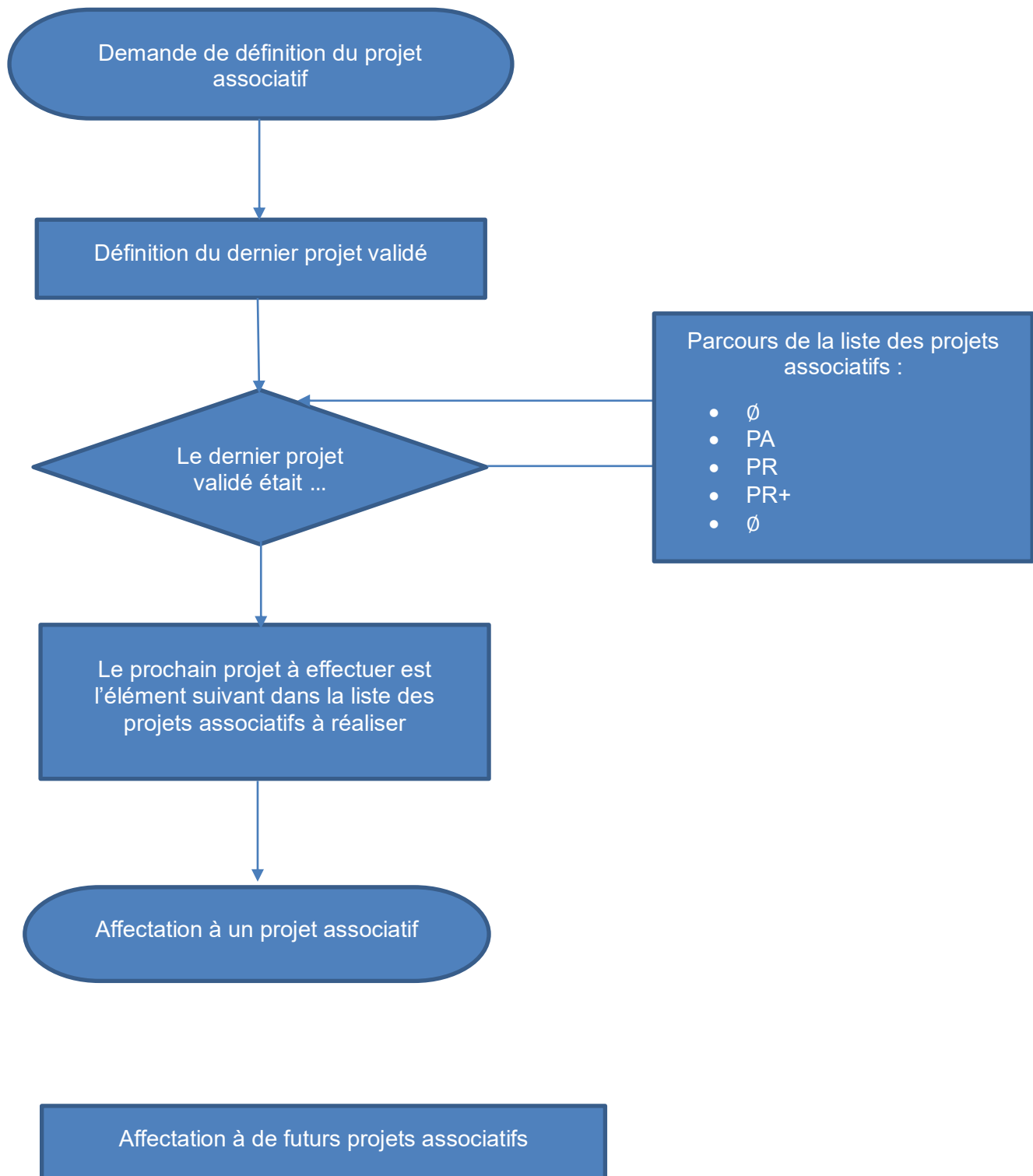
www.stackoverflow.com => Forum d'aide aux développeurs

www.grafikart.fr => Tutoriels de conception de site web

www.github.com => Code distribué et utilisable en Open Source

www.openclassroom.com => Site de cours et d'aide en ligne sur l'informatique

14. Annexe



Détermination du coefficient à appliquer sur les
demande d'effectif

Calcule des demandes d'effectif réelles

Calcul du score en fonction de l'importance du choix

Calcule des points bonus/malus

Affectation dans les clubs

Suppression des effectifs superflus

Mise en base de données