

ECOLE D'INGÉNIEURS ISIS
INFORMATIQUE ET SYSTÈMES D'INFORMATION POUR LA SANTE
Année universitaire 2021-2022

Projet Syllabus

Bastien Hatinguais
Anaëlle Jolivet
Paul Massias
Clotilde Salvador

FIE 3 – Promotion 2024

Tuteur : M. Defossez
Commanditaire : M. Bastide

Sommaire

Sommaire	1
Liste des tableaux et figures	2
Remerciements	3
Glossaire	4
Introduction	5
Contexte	5
Objectifs	5
Charte Projet	6
Présentation du projet	6
Public et service cible	6
Conception du projet	7
Acteurs et fonctionnalités	7
Diagramme des cas d'utilisation	8
Maquette	8
Diagramme des classes	9
Point sur l'organisation et outils utilisés	9
Planification	10
Développement	11
Choix techniques	11
Frontend (cf glossaire)	11
Backend (cf glossaire)	12
Base de données	12
Sécurité	13
Connexion / inscription	13
Droits d'accès	14
Sécurisation de l'API	14
Sécurisation des pages de l'application	15
Génération de l'API	15
Résultat	15
Bilan	16
Projet	16
Personnel	16
Perspective d'avenir	17
Webographie	18
Annexes	19

Liste des tableaux et figures

Figure 1 - Tableau des droits selon les rôles.....	7
Figure 2 - Diagramme des cas d'utilisation.....	8
Figure 3 -Diagramme des classes.....	9
Figure 4 - Schéma représentatif des branches et commits du github.....	9
Figure 5 - Extrait de la partie Analyse-Conception de la planification.....	10
Figure 6 - Diagramme d'architecture de l'application.....	11
Figure 7 - Schéma représentatif des langages utilisés au sein de notre projet.....	12
Figure 8 - Modèle conceptuel de données.....	12
Figure 9 - Schéma du fonctionnement de l'authentification.....	13
Figure 10 - Diagramme de séquence de connexion et d'inscription.....	14
Figure 11 - Représentation d'une attaque "Man In the Middle".....	14

Remerciements

Avant toute chose, nous tenons à remercier toutes les personnes nous ayant aidé, d'une manière ou d'une autre, à mener à bien ce projet.

Nous remercions M. Defossez, pour nous avoir proposé ce sujet et pour nous avoir accompagné et conseillé tout au long du projet. Son aide et son expertise nous ont été précieuses, nos remerciements vont aussi au temps qu'il a su trouver pour nous recevoir et répondre à nos questions.

Nous remercions également M. Bastide et M. Pecatte pour les connaissances apportées par leurs cours ainsi que leurs précieux conseils techniques liés au développement de l'application.

Enfin, nous remercions l'école de nous permettre de telles expériences enrichissantes, et de nous avoir offert l'occasion de participer à sa gestion à travers ce projet Syllabus.

Glossaire

API : Application Programming Interface

Interface faisant le lien entre une base de données et une application nécessitant ces données.

Backend

Base fonctionnelle d'une application, partie non visible par l'utilisateur.

BDD : Base De Données

Permet de stocker des données de manière structurée.

BDD relationnelle

BDD dans laquelle l'information dans des tableaux à 2 dimensions appelés "tables" ou "relations".

Commit

Dans le contexte de Git, il s'agit de la validation des modifications apportées avant de pouvoir les publier.

CSS : Cascading Style Sheets

Langage informatique qui décrit la présentation des documents HTML et XML.

DAO : Decentralized Autonomous Organization

Interface qui fait le lien entre les objets manipulables et la base de données.

Framework

Ensemble d'outils et de composants logiciels à la base d'un logiciel ou d'une application.

Frontend

Ensemble des éléments visibles et accessibles directement sur une application web, avec lesquels un utilisateur peut interagir.

HTML : HyperText Markup Language

Langage de balisage conçu pour représenter les pages web en les structurant et mettant en forme leur contenu.

IDE : Integrated Development Environment

Ensemble d'outils qui permet d'augmenter la productivité des programmeurs qui développent des logiciels.

JS : JavaScript

Langage de programmation de script principalement employé dans les pages web interactives.

JWT : JSON Web Token

Jeton d'accès permettant l'authentification de l'utilisateur.

"Man in the middle"

Attaque qui a pour but d'intercepter les communications entre deux parties, sans que ni l'une ni l'autre puisse se douter que le canal de communication entre elles a été compromis.

ORM

Interface permettant de manipuler le contenu d'une base de données en utilisant la programmation orientée objet.

Plugins

Petit logiciel conçu pour être greffé à un autre logiciel à travers une interface prévue à cet effet, et apporter à ce dernier de nouvelles fonctionnalités.

Routeur

Outil redirigeant la requête HTTP vers le code qui doit la gérer. En d'autres termes, il détermine ce qui doit se produire quand un utilisateur visite une certaine page.

SPA : Single Page Application

Application web accessible via une page web unique. Le but est d'éviter le chargement d'une nouvelle page à chaque action demandée, et de fluidifier ainsi l'expérience utilisateur.

SSL : Secure Sockets Layer

Protocole pour navigateurs Web et serveurs qui permet l'authentification, le chiffrement et le déchiffrement des données envoyées sur Internet.

UE : Unité d'enseignement

Regroupement de cours selon une même thématique. Une UE a un nombre de crédits ECTS associés contrairement à un enseignement seul.

WBS : Work Breakdown Structure

Organigramme des tâches du projet.

1. Introduction

1.1. Contexte

Dans le cadre de la refonte de son système administratif, l'école d'ingénieurs ISIS a proposé différents sujets de projets tutorés pour les FIE3. Ces sujets ont pour but de pouvoir, à terme, simplifier la gestion des différents outils proposés par l'établissement. De plus, la possibilité d'une collaboration entre les enseignants et l'administration permettrait de réduire la charge de travail et simplifier les processus administratifs. En plus de la gestion du syllabus, tout un écosystème pourrait être déployé grâce à des outils de recherche de stages ou encore de mobilités internationales. Tout cela pourrait aider le personnel actuel de l'établissement et les élèves, mais aussi, les futures générations d'élèves ingénieurs. Les différents projets tutorés portent donc chacun sur la création d'une application répondant à une de ces problématiques. Notre équipe s'est portée volontaire pour se pencher sur la question de la gestion et création des syllabus. Ce sujet a été proposé par M. Defossez en sa qualité de directeur des études, il a donc été notre référent durant toute la durée du projet.

1.2. Objectifs

L'application relative au syllabus a pour but de simplifier le bon fonctionnement normal des outils administratifs et pédagogiques mais elle a aussi un rôle informatif : elle doit pouvoir être consultée par tous, pour que les élèves, les enseignants ou encore des personnes extérieures à l'école puissent s'informer sur les cours proposés et leur organisation. Cette application doit également permettre la gestion du syllabus : le personnel administratif et enseignant doit pouvoir créer et modifier les cours, les UE (*cf glossaire*) ainsi que les années auxquelles ils appartiennent. Toutes ces fonctions sont nuancées selon des rôles et leurs droits associés.

2. Charte Projet

2.1. Présentation du projet

L'outil de gestion du syllabus, comme dit précédemment, doit permettre la consultation, modification et création des syllabus de chacune des années d'étude de l'école. Nous avons donc développé une application web à laquelle il est possible de se connecter pour les personnes ayant des droits spécifiques, tandis que les simples visiteurs peuvent seulement parcourir les syllabus sans s'authentifier. Le personnel administratif et enseignant ayant une possibilité de connexion vont voir leurs droits varier selon leurs rôles. Par exemple, le directeur des études, à la tête de l'organisation de tous les cours, doit avoir un droit de lecture et d'écriture sur l'entièreté des syllabus. En revanche, un membre sans rôle spécifique n'aura qu'un droit de lecture. Les professeurs et intervenants auront des accès plus ou moins vastes selon s'ils sont responsables d'un cours, d'une UE ou d'une année.

2.2. Public et service cible

Nous pouvons distinguer deux publics distincts. D'une part, il y a les étudiants ou autres personnes extérieures cherchant à se renseigner sur les cours dispensés à l'école. Ils pourront consulter l'application librement et avoir accès à toutes les informations utiles : les UE enseignées dans une année, les cours ainsi que le détail dans ces mêmes UE. D'autre part, il y a l'administration de l'école : cet outil doit lui permettre de réorganiser les cours, de les modifier (voire les créer ou les supprimer) d'une année sur l'autre.

Cette application a pour but de grandement simplifier la procédure de gestion actuelle qui consiste en une multitude de fichiers différents détenus par le directeur des études et les responsables d'année. Actuellement, il y a un document par cours, un document par UE, chacun contenant tous les détails et informations relatifs aux enseignements ou à l'UE en question. Ces fichiers doivent être mis à jour régulièrement par les professeurs sur demande du responsable de l'année concernée. Les professeurs, qu'ils soient responsables d'année ou simplement assignés à un cours, n'ont qu'une visibilité restreinte à leur champ du syllabus. En termes d'affichage public, les syllabus sont disponibles sur le site internet de l'école, mais ils ne présentent que les noms, le nombre d'heures et les crédits ECTS des UE et des enseignements. Aucun détail supplémentaire n'y est décrit.

3. Conception du projet

3.1. Acteurs et fonctionnalités

La gestion d'un syllabus (création, modification, suppression), sous entend différents rôles avec différents droits associés. Après discussion avec notre référent, nous avons identifié plusieurs rôles : le directeur des études, les professeurs, le personnel administratif et les visiteurs. Toutes ces fonctions ont un droit de lecture sur l'ensemble des syllabus (années + semestres + UE + cours) et la liste des enseignants. La différence va être que les visiteurs vont avoir un droit de lecture plus restreint. Certains détails des cours ne doivent pas être visibles pour eux, comme les compétences visées, le plan du cours et les modalités d'évaluation.

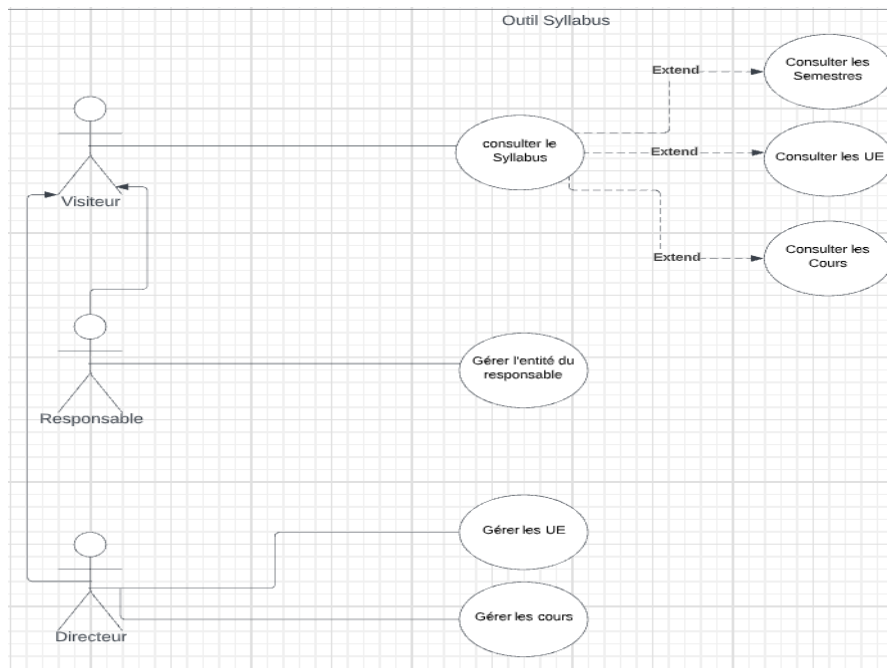
Le directeur des études possède naturellement les droits d'écriture sur l'ensemble des syllabus. Le personnel administratif n'en possède pas mais a un droit de lecture sur tout le contenu (droit non restreint contrairement aux visiteurs). Pour les enseignants (les professeurs et les intervenants), la répartition des droits est plus complexe. Ils peuvent être responsable d'une année, d'une UE ou d'un cours. Leurs droits d'écriture seront en conséquence : un responsable d'année aura un droit d'écriture sur tous les semestres, UE, cours et enseignants de son année. De la même manière, un responsable d'UE pourra le modifier ainsi que les cours et enseignants qui y sont associés. Enfin, un professeur assigné à une matière pourra modifier cette dernière.

Tous ces droits sont résumés dans le tableau ci-dessous.

Droits	année		semestres		ue		cours		profs/intervenants	
	toutes	la sienne	tous	les siens	toutes	les siennes	tous	les siens	tous	"les siens"
Directeur	L+E		L+E		L+E		L+E		L+E	
Responsable année (prof)	L	L+E	L	L+E	L	L+E	L	L+E	L	L+E
Responsable UE (prof)	L		L		L	L+E	L	L+E	L	L+E
Professeur	L		L		L		L	L+E	L	
Intervenant	L		L		L		L	L+E	L	
Reste administration	L		L		L		L		L	
Visiteur (lecture restreinte)	L		L		L		L		L	

Figure 1 - Tableau des droits selon les rôles

3.2. Diagramme des cas d'utilisation



Lors de la définition des acteurs et des rôles nous avons conçu le tableau des rôles présenté ci-dessus. Suite à ce travail nous avons défini ce Diagramme des Cas d'Utilisations.

Nous pouvons y voir que le visiteur a la possibilité de consulter le syllabus, action qui peut se traduire par la consultation des semestres, des UE ou encore des cours.

Nous observons que les deux autres acteurs héritent de ce cas d'utilisation.

Le responsable peut être un responsable de cours, d'UE ou d'une année.

Le cas d'utilisation correspondant est " gérer l'entité du responsable ". En effet, le responsable a des droits (en lecture et en écriture) sur l'entité qu'il supervise . Il va donc pouvoir les modifier ou consulter les informations.

Figure 2 - Diagramme des cas d'utilisation

Enfin le directeur a le cas d'utilisation du visiteur ainsi que la possibilité de gérer (création, lecture, modification, suppression) les Unités d'Enseignement ainsi que les cours.

3.3. Maquette

Durant la période de conception nous avons réalisé une maquette de notre application. Celle-ci vise à avoir une base concernant les interfaces utilisateur avant de commencer le développement. Pour les réaliser, nous avons choisi d'utiliser l'outil de maquettage Web Balsamiq pour son interface simple et nos expériences antérieures.

Vous trouverez les maquettes en annexe A.

3.4. Diagramme des classes

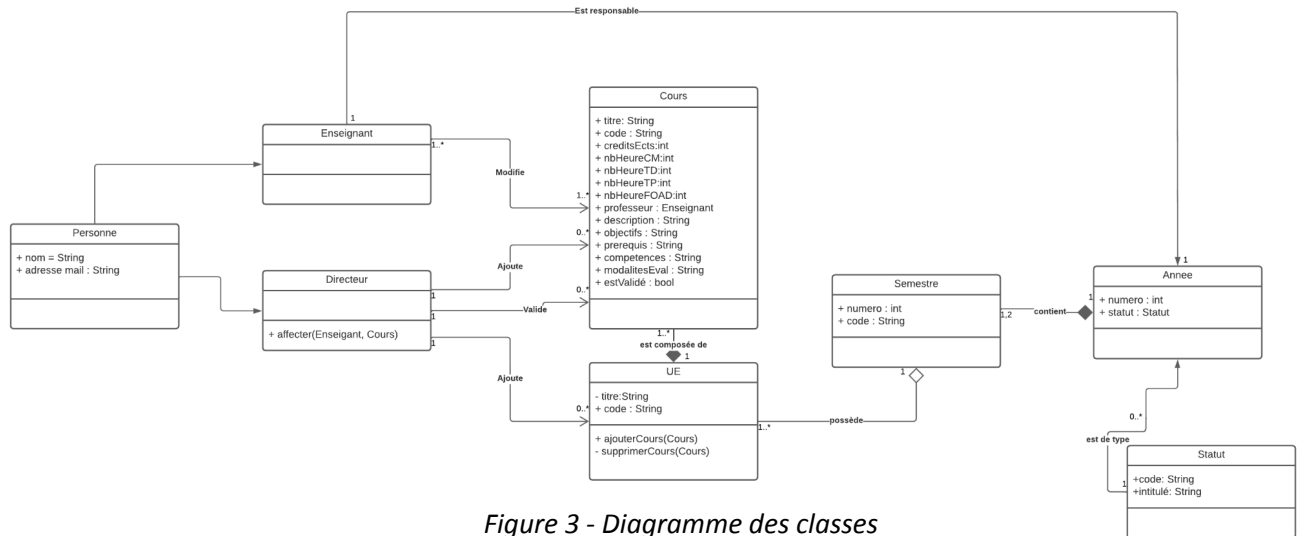


Figure 3 - Diagramme des classes

Comme le montre le diagramme ci-dessus, les cours possèdent un booléen (variable pouvant prendre la valeur "vrai" ou "faux") indiquant leur validation par le directeur. Il déterminera si le cours peut être affiché aux visiteurs ou non. Les années présentent un statut indiquant si elles s'adressent aux étudiants ou aux alternants.

3.5. Point sur l'organisation et outils utilisés

Tout au long du projet, nous avons eu besoin de communiquer, de travailler ensemble sans se gêner, et de mutualiser les documents nécessaires aux autres membres du groupe. Pour ce faire, nous avons choisi pour la discussion instantanée Messenger qui permet aux utilisateurs de créer des groupes de discussions. Nous nous en sommes servis pour décider ensemble de certains points particuliers, pour suivre l'avancement du projet et collaborer. Néanmoins, nous avons aussi besoin de centraliser les différents documents et livrables finaux, nous nous sommes principalement servi de Notions (application de mise en commun de toute forme de données).

Avec Notion, nous avons pu centraliser les différents éléments du projet en les organisant selon différentes catégories. Nous vous proposons un extrait de notre organisation sur Notion en annexe B. Il s'est avéré très utile pour la gestion des éléments du projet. De nombreuses fonctionnalités sont disponibles sur cet outil qui nous a permis d'y intégrer des fonctionnalités extérieures tels que des Google Docs ainsi qu'un lien vers le repository GitHub du projet. GitHub nous a été particulièrement utile pour gérer le développement et le versionning de l'application. Nous avons choisi de travailler chacun sur des branches séparées afin d'éviter les problèmes de conflits et pouvoir avancer chacun de notre côté.

Légende :

— Main

— Dev

— AffichageVisiteurs

— UE

— Dev-personnel

— AffichageVisiteurs.ModifBastien

— Rôles

— Inscription-connexion

— Cours

— Cours.edition

— NavBar

— Cours.suppression

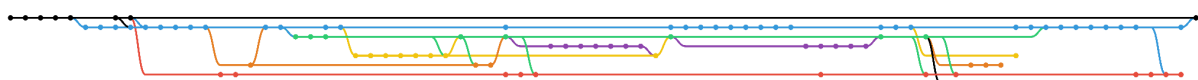


Figure 4 - Schéma représentatif des branches et commits du github

Rapport de projet - Projet Syllabus

Le schéma ci-dessus représente les différentes branches de développement du projet. Les points représentent les commits (*cf glossaire*) et les jointures entre deux branches représentent les merges. La branche “main” est la version fonctionnelle de l’application. La “dev” contient la version fonctionnelle mais non testée de l’application. Les autres sont les branches sur lesquelles de nouvelles fonctionnalités ont été ajoutées. Elles sont merge avec la branche dev lors leur aboutissement .

Nous avons fait le choix de tous utiliser le même IDE (*cf glossaire*) pour simplifier notre collaboration. Nous avons opté pour VS Code qui est largement répandu dans le développement web. Les plugins mis à disposition par la communauté nous ont beaucoup aidé pendant la programmation. Nous avons utilisé “Spring Boot Dashboard” qui permet de lancer et de visualiser facilement le serveur Spring Boot. Le plugin “Live Share” fut également utile pour notre collaboration. Il nous a permis de nous entraider en visualisant le code de chacun depuis un autre ordinateur à distance.

3.6. Planification

Pour planifier ce projet, nous avons fait appel à des outils vus en cours. Pour un premier découpage des tâches, nous nous sommes servis d’une WBS (*cf glossaire*) avec un faible niveau de détail, de “grandes tâches”. Puis les parties ont été identifiées et les sous-tâches définies. A partir de ces dernières, nous avons rétro-planifié suivant la date de rendu du rapport. Nous avons défini trois grandes parties : analyse-conception, développement et rendus finaux.

Pour la répartition des tâches de ce projet, nous nous sommes basés sur les compétences et les intérêts de chacun. Notre collaboration est restée tout au long du projet sur une volonté d’entraide, d’écoute et de partage. Nous avons collaboré afin d’aider la réalisation des tâches d’une personne en cas de difficultés.

Tâches/ Infos	Responsable	Durée estimée	Date début estimée	Date de fin estimée	Date de début effective	Date de fin effective	Durée effective (j)	Delta début / durée	Etat
I-Analyse/Conception	Paul	10 jours		21/02	06/03	21/02	28/02	6 0 /	Fin
a-Diagramme des Cas d'utilisation	Paul	2	21/02	23/02	21/02	22/02	1	0 /-1 jours	Fin
b-Modèle Conceptuel de Données	Bastien-Clotilde	4	21/02	25/02	22/02	27/02	5	1 / 2	Fin
c-Diagramme de Classes	Bastien-Clotilde	5	26/02	03/03	22/02	25/02	3	-4 jours / -2 jours	Fin
d-Maquetage	Anaëlle	7	21/02	28/02	23/02	27/02	4	2 / -1	Fin
e-WBS/Planification	Paul	5	23/02	28/02	23/02	28/02	5	0 / 0	Fin

Figure 5 - Extrait de la partie Analyse-Conception de la planification

Vous pouvez observer ici un extrait de la planification du projet. Elle a été réalisée avec un tableur car bien que nous ayons vu des outils comme Gantt aidant à planifier, l’avantage que garde le tableur est la possibilité infinie de personnalisation de cette dernière. Ici nous avons gardé une trace des éléments planifiés au préalable (en gris sur l’image) et des déroulements réels des tâches avec les dates et durées effectives.

De plus cette méthode nous permettait de suivre l’avancement grâce à des éléments visuels comme des couleurs particulières ou encore des états associés aux tâches. Pour permettre la collaboration évoquée précédemment nous avons défini des responsables de grandes parties. Ici le responsable de la partie Analyse Conception est Paul. Cela nous permet d’avoir toujours un point d’ancrage et de pouvoir avancer rapidement en cas de blocages.

Vous trouverez la planification complète en annexe C.

4. Développement

4.1. Choix techniques

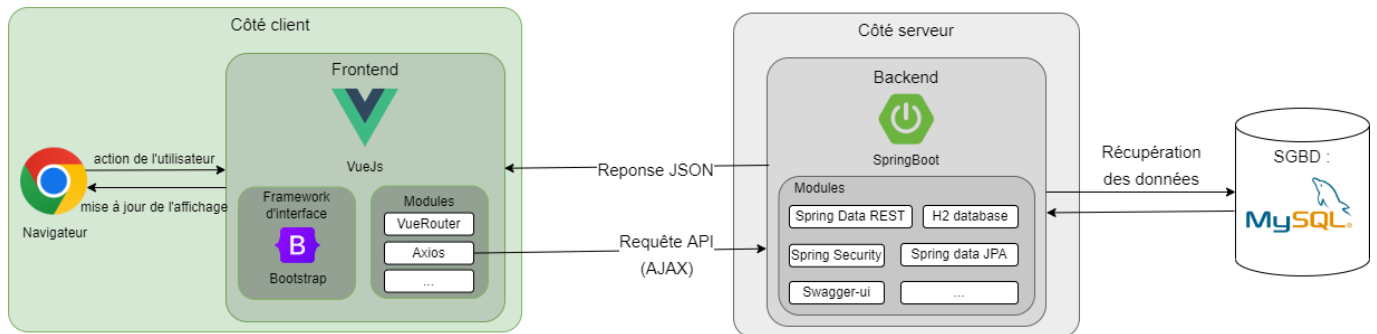


Figure 6 - Diagramme d'architecture de l'application

4.1.1. Frontend (cf glossaire)

Pour développer notre frontend, nous avons choisi d'utiliser VueJs. C'est un framework (cf glossaire) qui permet de facilement créer une interface utilisateur à l'aide de composants réutilisables à toutes étapes du projet. Cette solution s'est présentée naturellement, étant donné que nous l'avons déjà utilisé dans nos cours de Technologies Web. De plus, VueJs est plutôt simple à prendre en main pour les moins familiers d'entre nous. Aussi, VueJs est un logiciel très répandu dans la communauté des développeurs (3e framework le plus utilisé au monde), ce qui signifie qu'il existe une très grande documentation et de nombreux tutoriels accessibles en ligne.

VueJs était également un choix adapté vis à vis de notre volonté de créer une SPA (cf glossaire). Une SPA, étant une application continue sur une seule page, utilise un procédé dans lequel le JS (cf glossaire, code de l'application) envoie des requêtes à l'API (cf glossaire) pour récupérer les données. L'alternative à la SPA est le SSR : Server Side Rendering. Dans ce cas là, le serveur génère la page en commençant par envoyer les requêtes à la base de données (BDD, cf glossaire) puis ensuite construit le code (en HTML, CSS, JS). Cela demande plus de ressources au serveur mais présente des avantages en termes de référencement (donc non indispensable de notre côté). L'avantage de la SPA, plus appropriée dans notre cas, est que le serveur va être moins saturé car il ne va gérer que les requêtes de la BDD vers l'API. Il ne va pas s'occuper de tout le code HTML/JS (cf glossaire) qui lui, est généré directement par la machine du client. La SPA permet d'avoir une meilleure expérience utilisateur puisque la page ne se recharge pas.

Ce logiciel est d'autant plus pratique qu'il comprend de nombreux plugins (cf glossaire) permettant de simplifier l'utilisation du framework. Certains permettent de gagner un temps précieux en mettant en place des raccourcis. Ils vont parfois même jusqu'à générer des parties de codes entières. Nous avons utilisé de nombreux plugins dans notre projet, mais afin de ne pas encombrer le rapport, nous ne citerons ici que les 2 principaux : Axios et Vue Router.

Axios est un outil qui permet d'exécuter plus facilement les requêtes vers l'API. Vue Router permet, quant à lui, de créer différents chemins d'accès aux pages de l'application. Il limite également leur accès selon les droits de la personne connectée.

Pour compléter et alimenter notre frontend, nous avons utilisé la bibliothèque d'outils Bootstrap. Ce site web regroupe une très grande quantité d'extraits de code HTML et CSS (cf glossaire) correspondant à toutes sortes d'éléments visuels (boutons, barres de navigations, formulaires...). Il ne reste plus qu'à insérer dans un projet, permettant d'avoir plus rapidement un rendu visuel plus professionnel.

Languages



Figure 7 - Schéma représentatif des langages utilisés au sein de notre projet

4.1.2. Backend (cf glossaire)

Le framework choisi pour développer notre backend est SpringBoot : ce logiciel permet de créer des applications conformes au modèle MVC : Modèle Vue Contrôleur. Ce modèle permet de réunir 3 aspects du backend en un seul framework : les entités manipulées (modèle), l'affichage (vue) et l'intelligence de l'application (contrôleur).

Dans le cadre de notre projet, nous n'avons utilisé SpringBoot uniquement comme interface permettant d'accéder à une API et donc à notre BDD. Nous avons ce framework notamment puisque c'est celui que nous avons utilisé en classe, mais également puisqu'il utilise le langage de programmation Java, qui est le seul avec lequel nous étions tous familiers. SpringBoot permet aussi de gérer la sécurité de l'application et dispose de modules très utiles tel que Swagger-ui qui nous a permis de visualiser et tester l'API. Il fournit également un système très pratique permettant de manipuler les données de la BDD avec l'ORM (cf glossaire) Hibernate.

4.1.3. Base de données

Pour la gestion de la base de données, nous avons utilisé l'outil MySQL. MySQL est un système de gestion de BDD relationnelle (cf glossaire), modèle plus adapté dans notre cas car nous ne manipulons que peu de données. Cet outil utilise donc le langage de requêtes SQL, langage qui permet d'exploiter les BDD telles que la nôtre. Une des raisons pour lesquelles nous avons choisi SQL, en plus d'être le langage étudié en cours et adapté à notre utilisation, est qu'il permet de facilement concevoir les relations entre nos entités.

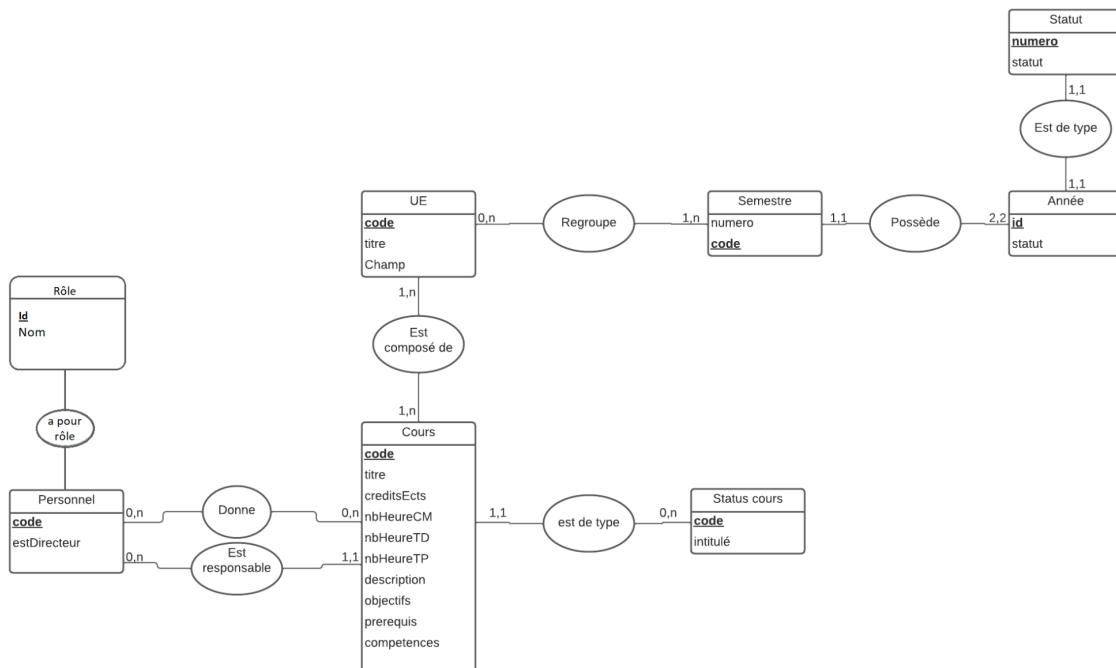


Figure 8 - Modèle conceptuel de données

La modélisation de la base de données nous a permis de réaliser le modèle conceptuel ci-dessus. Nous n'avons pas représenté les informations relatives à l'authentification car elles ont été générées par Spring Security.

4.2. Sécurité

Étant donné que l'application traite des données importantes sur l'organisation des cours d'ISIS, il est nécessaire de verrouiller son accès. C'est pourquoi nous avons mis en place des mesures afin de sécuriser au mieux l'application.

4.2.1. Connexion / inscription

La connexion et l'inscription sont des fonctionnalités primordiales dans notre projet. Leur intérêt est de pouvoir donner accès à certaines fonctionnalités en fonction du rôle de l'utilisateur connecté. Par exemple, comme nous l'avons expliqué, le directeur d'étude doit pouvoir avoir des droits d'écriture et de lecture sur (quasiment) l'ensemble des données de l'application.

Pour permettre l'authentification, nous avons utilisé Spring Security qui est un module créé par Spring. Il permet de préconfigurer et de personnaliser des fonctions de sécurité au sein d'une application Java.

Le contrôleur d'authentification contient le code permettant de se connecter et de s'inscrire en utilisant les DAO (*cf glossaire*) qui permettent d'accéder aux données de la BDD. Lors de l'inscription, les données de l'utilisateur sont enregistrées après vérification de leur validité (longueur de mot de passe, email non utilisé...). Le mot de passe utilisateur n'est bien sûr pas enregistré en clair (donnée brute non encodée) dans la BDD. En effet, si c'était le cas, une personne malveillante ayant accès aux données aurait toutes les informations utilisateurs. Les mots de passe sont encodés grâce à Bcrypt. C'est une fonction de hachage, fonction qui transforme la valeur d'entrée en une chaîne de caractère (un hash) et qui ne peut pas restituer la valeur d'origine. Lors de la connexion, l'encodeur de Spring Security va vérifier que le mot de passe proposé correspond à la valeur encodée dans la BDD.

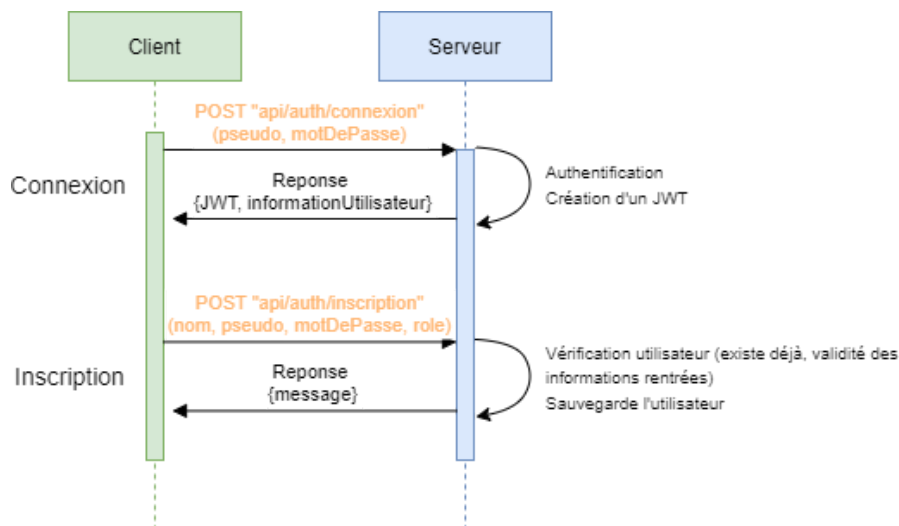


Figure 9 - Diagramme de séquence de connexion et d'inscription

Comme vous pouvez l'observer dans le schéma, la connexion d'un utilisateur lui renvoie un token (une chaîne de caractères) qui sera le moyen d'identifier l'utilisateur à chaque requête. Ce token est ajouté dans l'entête de chaque requête de l'utilisateur (champ Authorization), afin de vérifier si ses droits permettent l'exécution d'une telle requête. Si le token n'est pas renseigné, erroné, ou encore que l'utilisateur ne possède pas le droit nécessaire, le serveur renvoie une erreur 401 "Unauthorized".

Puisque le mot de passe de l'utilisateur est envoyé en clair dans les requêtes HTTP, il est nécessaire de les chiffrer. Si un hacker réalise une attaque "Man in the Middle" (*cf glossaire*) en se plaçant en tant qu'intermédiaire entre le client et le serveur, il pourra récupérer les informations de connexion de l'utilisateur. Pour cela, l'application doit prendre en charge le protocole SSL (*cf glossaire*), qui permet de chiffrer les données. L'url du site commencera donc par HTTPS et non HTTP, puisque c'est maintenant sécurisé.

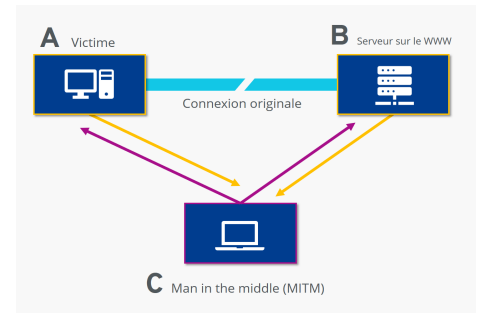


Figure 10 - Représentation d'une attaque "Man In the Middle"

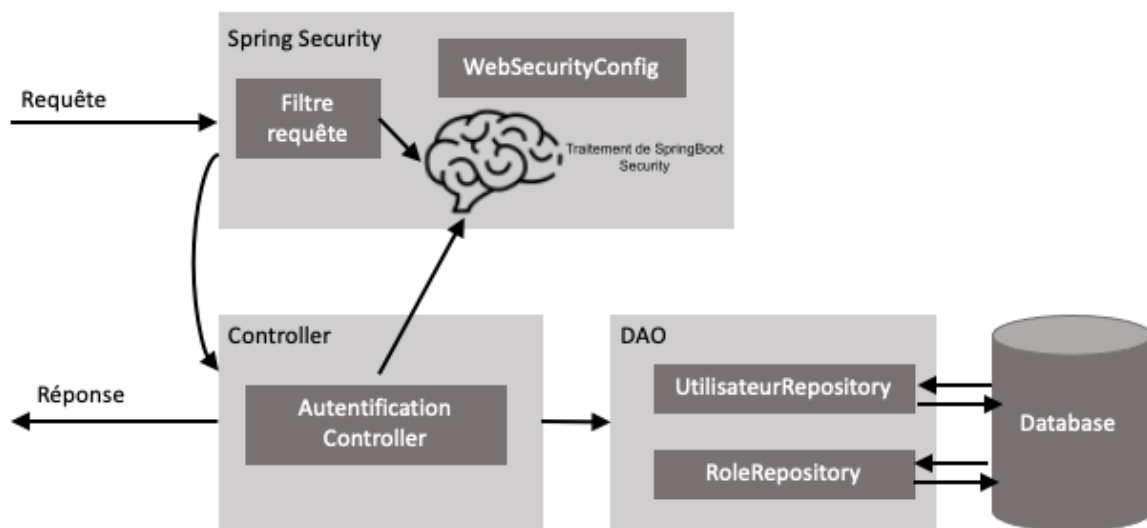


Figure 11 - Schéma du fonctionnement de l'authentification

4.2.2. Droits d'accès

Comme précisé plus haut, les utilisateurs ont différents droit d'accès aux pages de l'application et à l'API. Par exemple, le directeur d'étude doit pouvoir modifier les cours, UE, années... Alors que le professeur responsable d'un cours, ne peut modifier les informations que de son cours. Ces droits ont été développés plus amplement dans la partie 3.1.

4.2.3. Sécurisation de l'API

Les droits d'accès aux requêtes API sont vérifiées grâce au token précisé plus haut. Spring Security met à disposition un outil configurable permettant de vérifier le rôle de l'utilisateur à partir de son token à chaque requête faite. A la réception de la requête http, SpringSecurity utilise un filtre de requête qui traite et valide le JWT (*cf glossaire*) charge les données utilisateur et vérifie les autorisations.

4.2.4. Sécurisation des pages de l'application

Pour limiter les droits d'accès aux pages de l'application, nous avons utilisé le routeur (*cf glossaire*) Vue Router. Cet outil met à disposition un Navigation Guard qui permet de donner l'accès ou non d'une page à un

utilisateur en fonction d'une condition. Cette dernière dépendra du rôle de l'utilisateur connecté. S'il n'a pas les droits d'accès, il est redirigé vers une page lui signalant qu'il n'a pas les autorisations nécessaires.

4.3. Génération de l'API

Une API, pour définir autrement que dans le glossaire, est une boîte à outils mise à disposition des développeurs permettant d'accéder à des données de la BDD. Nous aurions pu la créer nous-même mais c'est un travail assez laborieux et répétitif. Afin de gagner du temps, nous avons utilisé le module de Spring Data (plugin de SpringBoot) qui permet de générer l'API automatiquement. En plus d'être un gain de temps conséquent, elle peut être combinée avec Swagger-Ui (plugin ajouté à notre framework) qui va nous permettre de voir et tester les requêtes émises.

Pour éviter un surchargement de l'API avec de nombreuses requêtes, nous avons mis en place des contrôleurs personnalisés qui permettent de récupérer toutes les requêtes et de les envoyer en une seule fois. Pour ce faire, nous avons également utilisé JPA (Java Persistence API), un plugin de Spring Data qui permet de simplifier les fonctions utilisées dans ces contrôleurs personnalisés.

4.4. Résultat

Le résultat, à l'heure où nous rédigeons ce rapport, est qu'un utilisateur peut commencer par se connecter à l'application. Si cette connexion n'est pas valide, il ne peut pas accéder aux fonctionnalités du syllabus. Lorsque l'utilisateur est connecté, il peut accéder à la page d'accueil avec les différentes années et leurs semestres. En cliquant sur un semestre, il accède à la liste de ses UE avec leur code, leur nom et leur crédits ECTS. Sur cette page, en cliquant sur une UE, il a la possibilité de dérouler la liste des cours qu'elle possède, et cliquer sur un de ces cours permet d'arriver sur sa page détaillée. Sur cette page, il retrouve les points clefs, les objectifs et la description du cours.

Selon les droits de l'utilisateur connecté, comme nous l'avons déjà décrit, l'utilisateur a la possibilité de modifier certains cours ou UE. Il existe des formulaires d'ajout et de modification pour les UE et les cours, ils sont également supprimables. Il y a également une page avec la liste du personnel que l'utilisateur peut modifier et supprimer s'il en a le droit. Lorsqu'un membre du personnel est ajouté, il est possible de lui attribuer des cours.

5. Bilan

5.1. Projet

Le projet Syllabus a été un projet plus conséquent que nous l'imaginions. A la fin de ce projet, nous avons pu constater aussi bien les différences entre le produit imaginé et le produit final que notre manque d'expérience en "conditions réelles". Dans le cadre d'un exercice, la manœuvre est aisée, mais mises bout à bout les difficultés ont tendance à s'accumuler rapidement, nécessitant parfois des ajustements de parcours.

Notre principale difficulté sur la partie gestion de projet a été la gestion du temps. En effet, nous n'avions que 2 mois pour réaliser ce projet. Beaucoup d'autres projets et de partiels se sont ajoutés à notre emploi du temps, ce qui nous a fait prendre du retard sur le projet Syllabus. Nous avons également rencontré des difficultés techniques au moment du développement. Nous n'avions jamais réalisé d'authentification. Ce fût un challenge enrichissant qui nous a demandé beaucoup de recherches et de documentations.

Au terme de ce projet, nous pouvons dire que nous avons capitalisé une bonne somme d'expériences dans différents domaines, aussi bien analytiquement qu'en termes de développement. Nous pouvons également constater notre progression sur le plan relationnel puisque notre équipe a fonctionné correctement du début à la fin du projet, et la communication a été la clé de voûte de cette expérience.

5.2. Personnel

5.2.1. Bastien Hatinguais

Cette expérience fut très enrichissante sur plusieurs niveaux. J'ai été heureux de pouvoir approfondir mes connaissances techniques avec spring boot et spring security. Ces dernières pourront m'être grandement utiles pour ma potentielle future alternance. Sur le plan humain, le déroulement du projet s'est très bien passé, car nous avons tous su être à l'écoute les uns des autres. Tout le monde a fait avancer le projet avec ses compétences et en apprenant de nouvelles. Je suis personnellement heureux du résultat qui représente bien les efforts que nous avons tous fournis pour être au final fier de notre rendu.

5.2.2. Anaëlle Jolivet

Ce projet tutoré m'a permis de consolider mes compétences techniques et organisationnelles mais aussi d'en développer de nouvelles. J'ai pu mettre en application les compétences que j'ai acquises dans les cours de technologie web, de gestion de projet, de génie logiciel et de base de données. J'ai pu approfondir ma compétence de travail en équipe. En effet, nous nous sommes beaucoup entraides lorsque nous avions des difficultés, notamment lors du développement. Nous avons également fait de nombreuses réunions en équipe et avec notre référent, ce qui nous a permis de bien structurer l'avancée du projet et de nous donner un aperçu de ce que peut être un projet dans le monde professionnel. L'ensemble du projet s'est bien passé et je suis satisfaite du résultat que nous avons réussi à avoir en si peu de temps.

5.2.3. Paul Massias

Ce projet m'a beaucoup apporté, aussi bien en qualité de chef de projet qu'en tant que membre de l'équipe. J'ai pu constater, non sans une certaine déception, que la planification n'était pas une tâche aisée. De plus, j'ai appris que le moindre petit problème peut avoir de grosses répercussions sur l'ensemble de l'équipe et du projet. Pour conclure ce bilan personnel je dirais que le poste de chef de projet n'est pas toujours enviable, il n'est pas facile de faire partie du projet tout en ayant comme responsabilité de le faire avancer/le mener à terme, mais grâce à cette équipe motivée et liée nous y sommes parvenu.

Rapport de projet - Projet Syllabus

5.2.4. *Clotilde Salvador*

De mon point de vue, ce projet a été très enrichissant tout autant sur le plan personnel que professionnel. En effet, cette expérience m'a permis d'appliquer les principes et méthodes vus en cours à un projet concret, ce qui m'a permis de mieux les appréhender. Grâce à mes camarades, j'ai même pu apprendre de nouvelles choses, l'entraide a été un point phare tout au long de notre mission. Sur le plan personnel justement, nous avons tous fait preuve d'empathie les uns envers les autres, et nous avons communiqué le plus ouvertement possible pour éviter tout quiproquo. Tout s'étant bien passé, j'en retire que les efforts que nous avons fait ont porté leurs fruits.

5.3. *Perspective d'avenir*

Au moment où nous rédigeons ce rapport, certaines fonctionnalités ont été revues à la baisse ou abandonnées principalement par manque de temps. Néanmoins, nous aurions pu imaginer dans un futur proche développer un poste de supervision (façon tableau de bord) pour le directeur ou encore des envois de mails personnalisés. Nous avons également pensé à faire valider chaque ajout de cours par le directeur, afin que n'importe qui ne puisse pas publier des informations potentiellement fausses, ce qui n'est pas le cas aujourd'hui. Pour aller plus loin, notre application aurait pu permettre d'exporter les syllabus en format PDF pour pouvoir les publier ou les imprimer plus facilement.

Webographie

<https://www.ionos.fr/digitalguide/serveur/securite/attaque-de-lhomme-du-milieu-aperçu-du-modele/>
<https://www.bezkoder.com/spring-boot-jwt-authentication/>
<https://vuejs.org/guide/>
<https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>
<https://getbootstrap.com/docs/4.6/getting-started/introduction/>
<https://github.com/jonashackt/spring-boot-vuejs>

Annexes

A. Maquettes

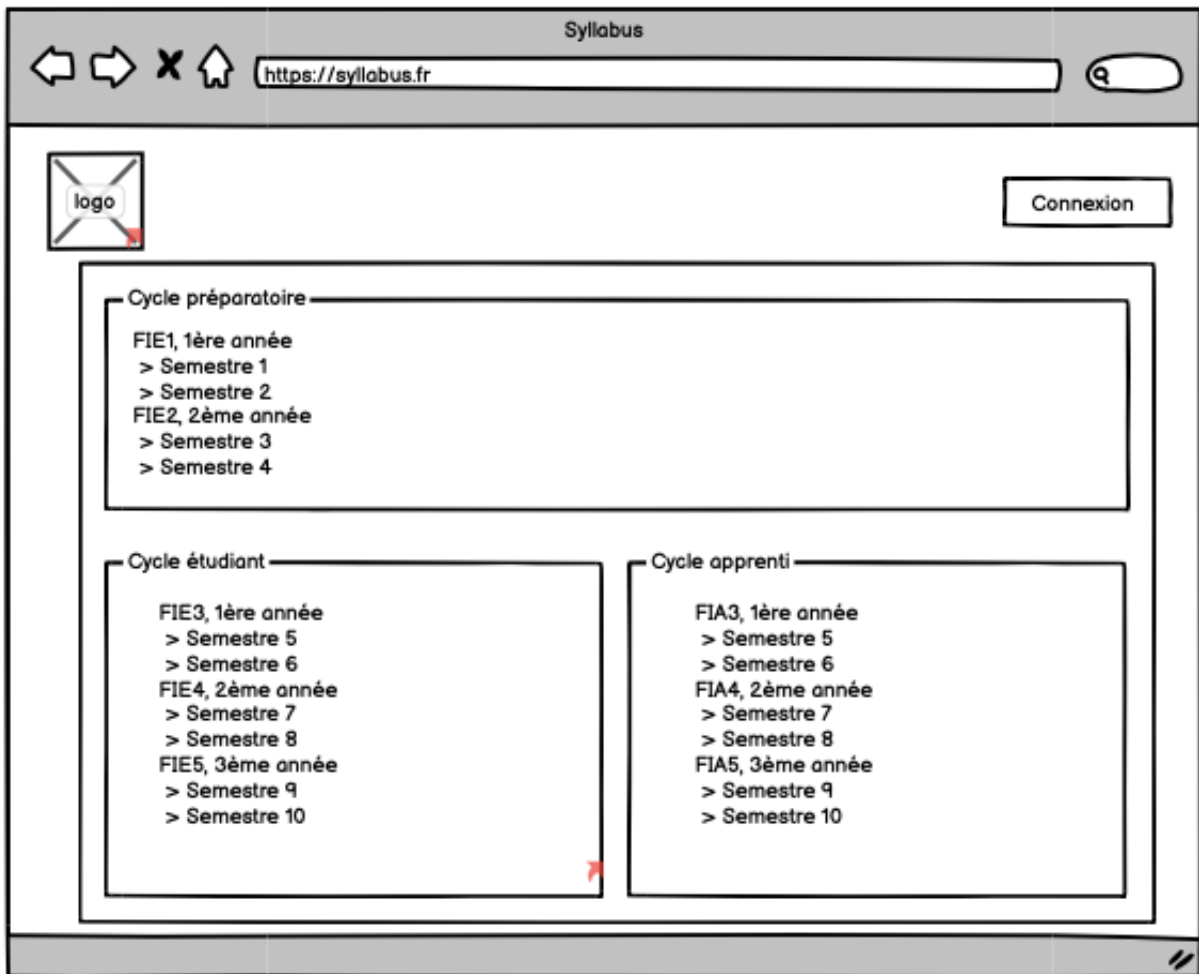
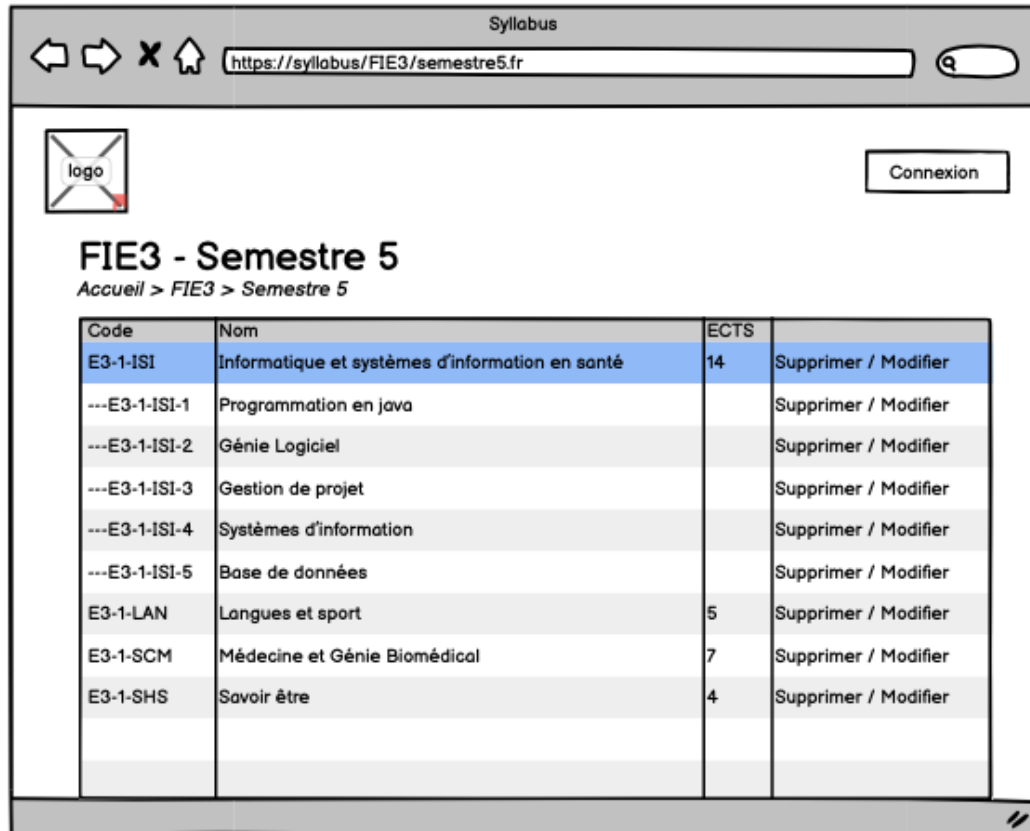


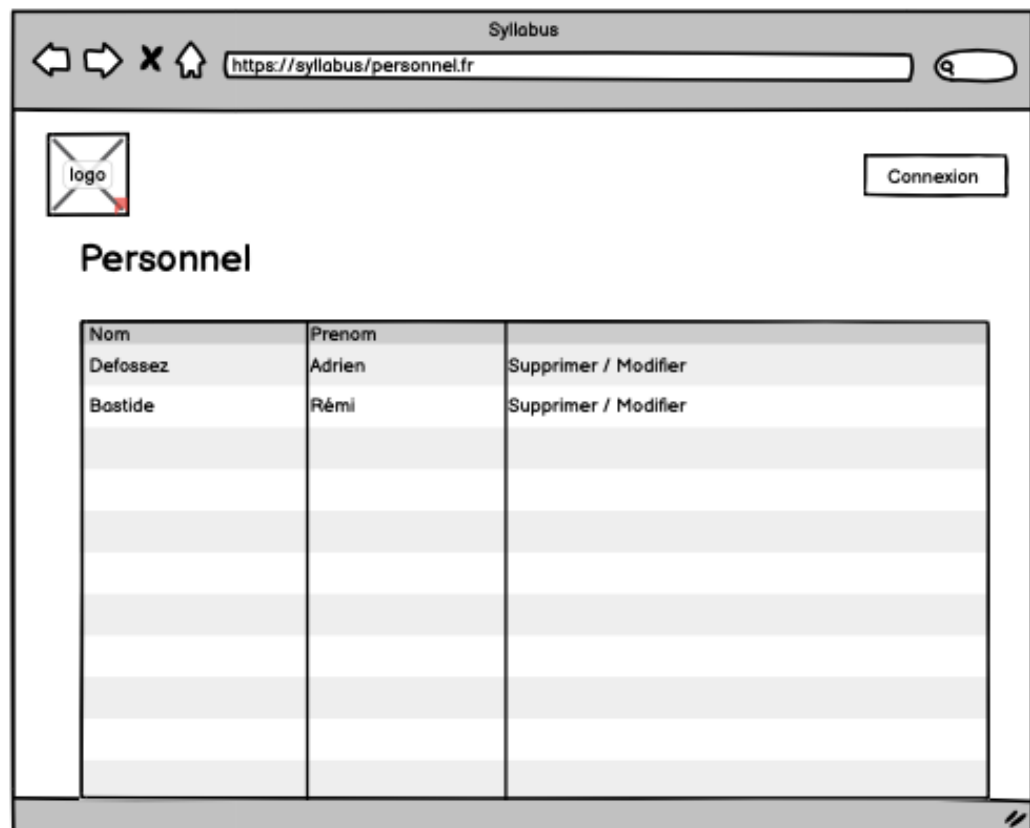
Figure A.1 - Page accueil avec lien vers les semestres



FIE3 - Semestre 5
Accueil > FIE3 > Semestre 5

Code	Nom	ECTS	
E3-1-ISI	Informatique et systèmes d'information en santé	14	Supprimer / Modifier
---E3-1-ISI-1	Programmation en java		Supprimer / Modifier
---E3-1-ISI-2	Génie Logiciel		Supprimer / Modifier
---E3-1-ISI-3	Gestion de projet		Supprimer / Modifier
---E3-1-ISI-4	Systèmes d'information		Supprimer / Modifier
---E3-1-ISI-5	Base de données		Supprimer / Modifier
E3-1-LAN	Langues et sport	5	Supprimer / Modifier
E3-1-SCM	Médecine et Génie Biomédical	7	Supprimer / Modifier
E3-1-SHS	Savoir être	4	Supprimer / Modifier

Figure A.2 - Affichage des UE et cours d'un semestre



Personnel

Nom	Prenom	
Defossez	Adrien	Supprimer / Modifier
Bostide	Rémi	Supprimer / Modifier

Figure A.3 - Liste du personnel

Rapport de projet - Projet Syllabus

The screenshot shows a web browser window titled 'Syllabus'. The address bar contains the URL `https://syllabus/fie3/semestre5/informatiqueetsystème d'informationpourlasanté`. The page layout includes a top navigation bar with a 'logo' icon and a 'Connexion' button. The main content area is divided into a left sidebar and a right main panel. The sidebar contains a list of links: 'Sommaire', 'Points clefs', 'Objectifs', and 'Description'. The main panel displays the course details for 'PROGRAMMATION EN JAVA', including the professor 'M. Bastide', the unit 'UE : Informatique et systèmes d'information en santé', and the semester 'Semestre 5'. Below this, there is a section titled 'Détail des heures' which lists 'Objectifs', 'Compétences visées', 'Description', 'Plan du cours', and 'Modalités d'évaluation'.

Figure A.4 - Page d'un cours

The screenshot shows a web browser window titled 'Syllabus'. The address bar contains the URL `https://syllabus/ajoutUE.fr`. The page layout is similar to the previous one, with a top navigation bar and a sidebar. The main panel is titled 'Ajouter une UE' and contains a form with the following fields: 'Titre' (text input), 'Code' (text input), 'Crédits ECTS' (text input), 'Année' (dropdown menu), and 'Semestre' (dropdown menu). Below these fields is an 'Ajouter' button.

Figure 5 - Formulaire d'ajout d'une UE

The image shows a web browser window with the title 'Syllabus'. The address bar contains the URL 'https://syllabus/modifierUE.fr'. The page features a 'logo' placeholder on the left and a 'Connexion' button on the right. The main content area is titled 'Modifier une UE' and contains a form with the following fields:

- Titre**: A text input field containing 'Informatique et systèmes d'information en santé'.
- Code**: A text input field containing 'E3-1-IS1'.
- Crédits ECTS**: A text input field containing '14'.
- Année**: A dropdown menu with 'FIE3' selected.
- Semestre**: A dropdown menu with '5' selected.

At the bottom of the form is a 'Modifier' button.

Figure 6 - Formulaire de modification d'une UE

Syllabus

https://syllabus/ajoutCours.fr

logo

Connexion

Ajout d'un cours

Titre

Code

UE

Personnel responsable

Description

Modalités d'évaluation

Nombre d'heures CM Nombre d'heures TD Nombre d'heures TI Nombre d'heures FOAD

0 0 0 0

Objectifs

Compétences visées

Plan du cours

Ajouter

Figure 7 - Formulaire d'ajout d'un cours

Syllabus

https://syllabus/modifierCours.fr

logo

Connexion

Modification d'un cours

Titre

Programmation en java

Code

E3-1-ISI-1

UE

Informatique et systèmes d'information en santé

Personnel responsable

Rémi Bastide

Description

lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate

Modalités d'évaluation

lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate

Nombre d'heures CM

3

Nombre d'heures TD

7

Nombre d'heures TP

10

Nombre d'heures FOAD

0

Objectifs

lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate

Compétences visées

lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate

Plan du cours

lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate

Modifier

Figure 8 - Formulaire de modification d'un cours

The screenshot shows a web browser window titled 'Syllabus' with the address bar displaying 'https://syllabus/ajoutPersonnel.fr'. The page features a 'logo' icon and a 'Connexion' button in the top right. The main content area is titled 'Ajout Personnel' and contains four input fields: 'Prenom', 'Nom', 'Dispense les cours', and 'Responsable des cours'. The 'Dispense les cours' and 'Responsable des cours' fields are dropdown menus. An 'Ajouter' button is located at the bottom of the form.

Syllabus
https://syllabus/ajoutPersonnel.fr

logo Connexion

Ajout Personnel

Prenom
[Text Input Field]

Nom
[Text Input Field]

Dispense les cours
[Dropdown Menu]

Responsable des cours
[Dropdown Menu]

Ajouter

Figure 9 - Formulaire d'ajout du personnel

The screenshot shows a web browser window titled 'Syllabus' with the address bar displaying 'https://syllabus/modifierPersonnel.fr'. The page features a 'logo' icon and a 'Connexion' button in the top right. The main content area is titled 'Modification Personnel' and contains four input fields: 'Prenom', 'Nom', 'Dispense les cours', and 'Responsable des cours'. The 'Dispense les cours' and 'Responsable des cours' fields are dropdown menus. A 'Modifier' button is located at the bottom of the form.

Syllabus
https://syllabus/modifierPersonnel.fr

logo Connexion

Modification Personnel

Prenom
Rémi

Nom
Bastide

Dispense les cours
Programmation en java

Responsable des cours
Informatique et systèmes d'information en santé

Modifier

Figure 10 - Formulaire de modification du personnel

B. Organisation et outils utilisés

Voici un extrait de l'organisation de nos pages sur Notion pour mieux rendre compte de ce que nous avons mis en place.

The screenshot shows a Notion workspace titled "Planification". The sidebar on the left contains a search bar and a list of navigation options: "Sujet", "Dead Line", "Planification" (selected), "Avancée", "Gantt", "Plannification", "Détail tâches développ...", "Avancée", "Etat individuel", "Réunions", "Diagrammes", "Dev", "Base de données", "Questions", "Consignes de rendu", and "Charte graphique". The main content area features a header image of a biplane and a table titled "Planification". The table has columns for "Name", "Assign", "Propriété", "Durée p...", "Dates prévues", "Dates effectives", and "Du". The table lists four tasks: "Diagramme de cas d'utilisation" (assigned to Darius massia, status "Fini"), "Développement semestre" (assigned to Anaëlle Jolive, status "En cours"), "Développement prof" (assigned to Darius massia, status "En cours"), and "Environnement" (assigned to bastien hatinguais, status "En cours"). The table is filtered by "Avancée" and "Gantt" views. The bottom of the page shows a "DÉCOMPTÉ 17" indicator.

Name	Assign	Propriété	Durée p...	Dates prévues	Dates effectives	Du
Diagramme de cas d'utilisation	Darius massia	Fini	2	21/02/2022 → 22/02/2022	21/02/2022 → 22/02/2022	
Développement semestre	Anaëlle Jolive	En cours				
Développement prof	Darius massia	En cours				
Environnement	bastien hatinguais	En cours				

C. Planification

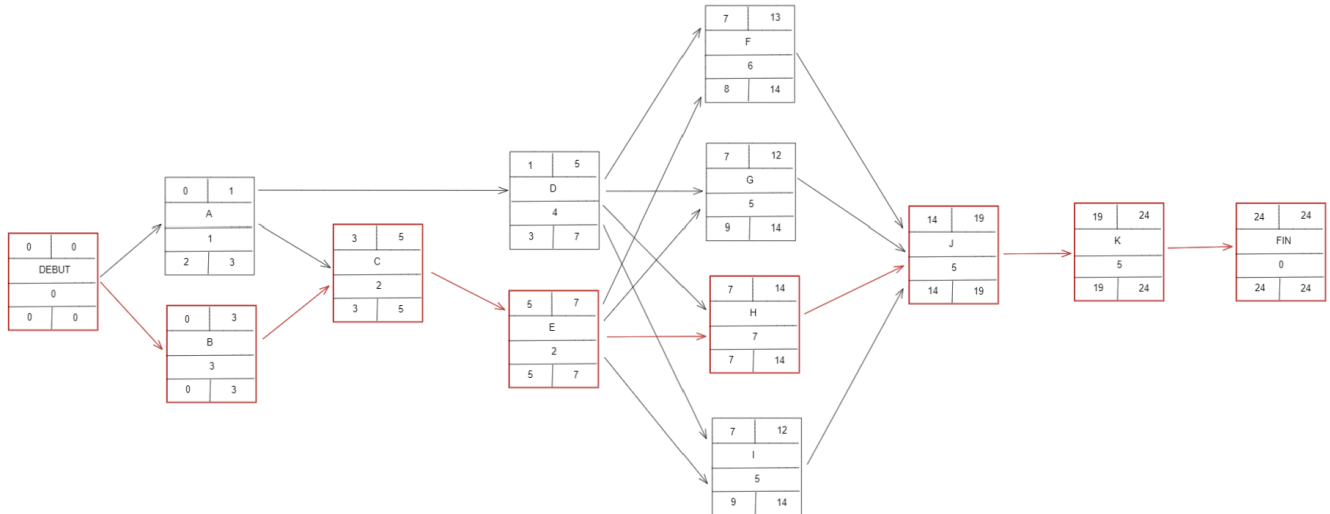


Figure 1 - CPM de notre projet à l'origine

Ce schéma a fait partie de la phase d'analyse préalable au projet, il n'est qu'au statut de première ébauche mais il exprime bien la complexité des liens entre les tâches.

Tâches/ Infos	Responsable	Durée estimée	Date début estimée	Date de fin estimée	Date de début effective	Date de fin effective	Durée effective (j)	Delta début / durée	Etat
I-Analyse/Conception	Paul	10 jours	21/02	06/03	21/02	28/02	6	0 /	Fin
a-Diagramme des Cas d'utilisation	Paul	2	21/02	23/02	21/02	22/02	1	0 /-1 jours	Fin
b-Modèle Conceptuel de Données	Bastien-Clotilde	4	21/02	25/02	22/02	27/02	5	1 / 2	Fin
c-Diagramme de Classes	Bastien-Clotilde	5	26/02	03/03	22/02	25/02	3	-4 jours / -2 jours	Fin
d-Maquetage	Anaëlle	7	21/02	28/02	23/02	27/02	4	2 / -1	Fin
e-WBS/Planification	Paul	5	23/02	28/02	23/02	28/02	5	0 / 0	Fin
II-Développement	Bastien	25 jours	07/03	01/04	05/03				En cours
a-Mettre en place environnement de travail	Bastien	1	07/03	08/03	05/03	06/03	1	-2 jours / 0	Fin
b-Développement gestion personnel	Paul	7	08/03	15/03	21/03	11/04	21	très important	Fin
c-Développement gestion UE	Anaëlle	7	08/03	15/03	08/03	21/03	13	0 / 7	Fin
d-Développement cours	Bastien	7	08/03	15/03	08/03	13/03	5	0 / -3	Fin
e-Développement semestre	Anaëlle	7	08/03	15/03	10/03	16/03	6	2 / 1	Fin
f-Développement graphique	Tous		15/03		08/03			////////////////////	En cours
g-Connexion	Bastien	10	15/03	25/03	20/03	26/03	6	5 / 1	Fin
h-Gestion des droits d'accès	Bastien	17	15/03	01/04	26/03	05/04	10	11 / 4	Fin
i-Exportation pdf Syllabus	Bast - Paul	17	15/03	01/04					Pas commencé
III-Tests et livrables	Anaëlle-Clotilde	10 jours	04/04	15/04	06/04/2022				En cours
b-Rapport de projet	Tous	10	04/04	15/04	06/04	15/04	9		En cours
c-Soutenance de projet	Tous	8	15/04	22/04					Pas commencé

Figure 2 - Répartition et durée des tâches du projet

Rapport de projet - Projet Syllabus

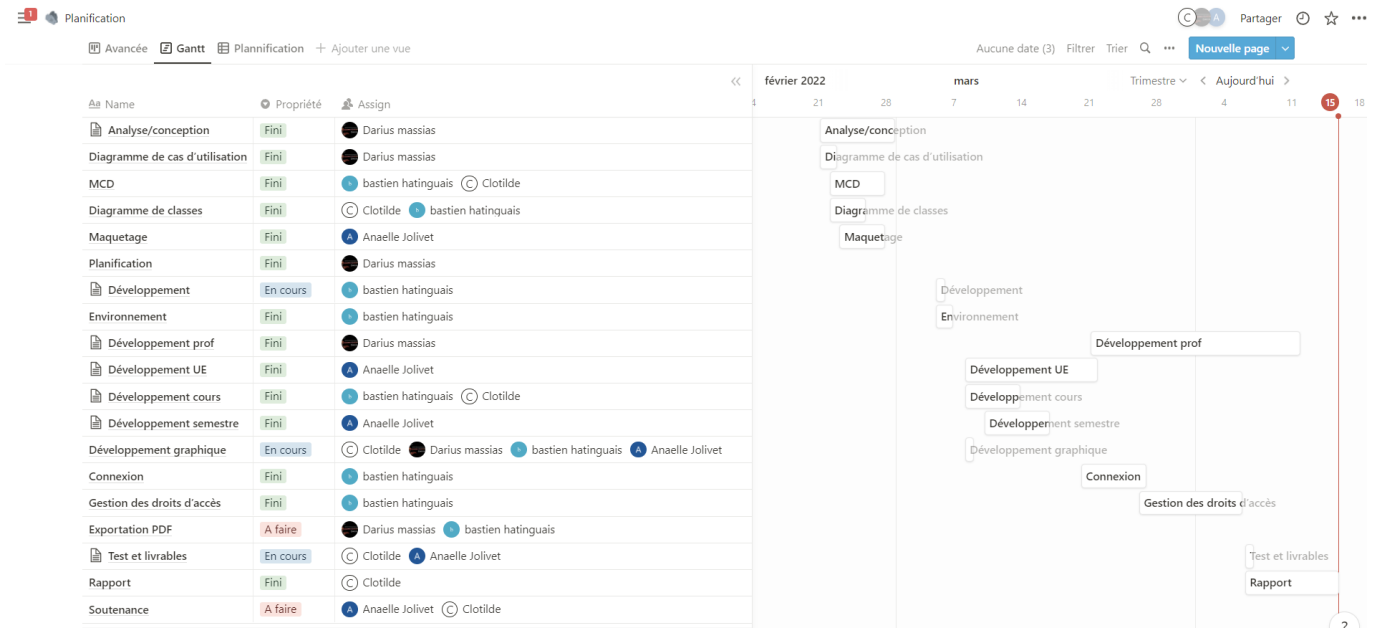


Figure 3 - Diagramme de Gantt de notre projet

D. Manuel utilisateur

Manuel Utilisateur

Projet Syllabus

Introduction	1
Visiteur :	1
Page d'accueil	1
Consulter UE	2
Consulter cours	2
Administrateur :	3
Connexion	3
Navbar	3
Gérer les ue	3
Gérer les cours	4

Introduction

Ce manuel a été conçu pour répondre aux doutes et interrogations que vous pourriez avoir et pour vous permettre de prendre en main correctement les différentes fonctionnalités proposées aussi bien pour les utilisateurs externes que pour les utilisateurs enregistrés. C'est d'ailleurs dans cet ordre que seront traitées les fonctionnalités. Nous considérerons que vous vous trouvez déjà sur le site.

Visiteur

Sans s'identifier, les visiteurs ont accès à une partie des fonctionnalités et des informations. Lors de l'arrivée sur le site la première page est la page d'accueil.

Page d'accueil

The screenshot displays the homepage layout with three distinct navigation sections, each enclosed in a box:

- Cycle Préparatoire**
 - FIE1, 1ère année
 - [Semestre 1](#)
 - [Semestre 2](#)
 - FIE2, 2ème année
 - [Semestre 3](#)
 - [Semestre 4](#)
- Cycle Etudiant**
 - FIE3, 3ème année
 - [Semestre 5](#)
 - [Semestre 6](#)
 - FIE4, 4ème année
 - [Semestre 7](#)
 - [Semestre 8](#)
 - FIE5, 5ème année
 - [Semestre 9](#)
 - [Semestre 10](#)
- Cycle Apprenti**
 - FIA3, 3ème année
 - [Semestre 5](#)
 - [Semestre 6](#)
 - FIA4, 4ème année
 - [Semestre 7](#)
 - [Semestre 8](#)
 - FIA5, 5ème année
 - [Semestre 9](#)
 - [Semestre 10](#)

A l'arrivée sur cette page, il vous est d'ores et déjà possible d'accéder aux différents semestres des années proposées. En cliquant directement dessus vous accéderez à la liste des Unités d'Enseignements de ce Semestre.

Consulter UE

Depuis la page d'accueil il vous faudra cliquer sur le semestre de votre choix pour voir les UE correspondantes. Vous arriverez alors sur cette page :

UE

Code	Titre	Crédits ECTS
E3-1-ISIS	informatique et Système d'information en Santé	14

Cette page recense les différentes UE du semestre, qui sont aussi cliquables et qui permettent de faire dérouler la liste des cours associés à cette UE.

Titre	
maths	

En cliquant sur le cours on accède à la consultation du cours.

Consulter cours

En ayant cliqué sur le cours on arrive sur la page descriptive de ce dernier qui liste toutes les infos concernant le cours cliqué.

maths

Sommaire

Points clefs

[Objectifs](#)

[Description](#)

Points clefs

Professeur en charge : Adrien Defossez

UE concernée : (à implémenter)

Semestre concerné : (à implémenter)

Détail des heures

Type	Nombre
CM	4
TD	5
TP	5
FOAD	0

Objectifs

le but c'est de faire des maths youpi

Compétences visées

le theoreme de pythagore c'est déjà pas mal

On peut voir ici le professeur en charge, le détail des heures et en dessous les objectifs du cours. Voilà qui conclut la partie visiteur de ce manuel, pour la partie Administrateur, voir ci-dessous.

Administrateur

Connexion

Pour accéder au côté administrateur, il faut utiliser le bouton de connexion situé en haut à droite. Une modale s'ouvrira vous demandant un pseudo et un mot de passe.

Se connecter

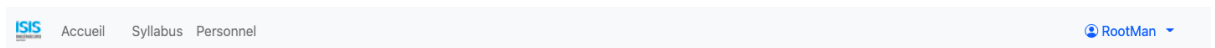
Pseudo

Mot de Passe

Valider

Navbar

Une fois authentifié, vous verrez votre nom en haut à droite de l'écran avec la possibilité de se déconnecter. En haut à gauche les boutons de navigation peuvent renvoyer vers le Syllabus (page d'accueil visiteurs) ou vers la liste du personnel. Depuis ces rubriques vous pourrez modifier les instances demandées.



Gérer les UE

En accédant aux UE depuis le syllabus en étant connecté en tant qu'administrateur, vous verrez la page suivante avec la possibilité d'ajouter via le bouton vert, de supprimer via le bouton rouge ou encore de modifier via le bouton bleu.

UE

Code	Titre	Crédits ECTS		
E3-1-ISIS	informatique et Système d'information en Santé	14	Supprimer	Modifier

Ajouter UE

En cliquant sur "modifier" on arrive sur la page suivante :

Modifier une UE

Titre
informatique et Système d'information en Santé

Code
E3-1-ISIS

Crédits ECTS
14

Année
FIE3

Semestre
5

[Modifier](#)

Pour valider les modifications , il suffit d'ajouter sur "Modifier" en bas de la page.

Gérer les cours

Pour accéder aux outils de gestion de cours il faut naviguer jusqu'aux cours puis, en étant authentifié vous verrez apparaître pour un cours particulier un bouton "modifier" et un bouton "supprimer" dont les fonctions sont relativement explicites.

Titre	
maths	Supprimer Modifier

Edition de cours 1

Titre
maths

UE
▼

Personnel responsable
Adrien Defossez ▼

Description
cours de maths trop super on adore ça en plus

Modalités d'évaluation
un qcm pitié

Nombre d'heures CM
4

Nombre d'heures TD
5

Nombre d'heures TP
5

Comme pour les UE, pour valider la modification d'un cours il faut appuyer sur le bouton modifier en bas de la page.