

ENSIIE 2019

# Rapport - Projet Web

Géraldine BAILLEUL - Mathis BOITEAU

Dorian DOAN - Alexandra GAUTHIER - Romain TIRBISCH



---

<b>I. Présentation du sujet</b>	<b>2</b>
<b>II. Limites et contraintes</b>	<b>2</b>
Base de données	2
Structuration du code	3
Incompatibilité des technologies	3
<b>III. Travail en équipe et répartition des tâches</b>	<b>4</b>
Outils de travail	4
Répartition des tâches	4
<b>IV. Difficultés rencontrées</b>	<b>5</b>
Technologies employées	5
Redéfinition de nos priorités	5
<b>V. Conclusion</b>	<b>5</b>

---

# I. Présentation du sujet

Notre groupe étant composé en majeure partie de membres de l'association IImagE, qui a pour vocation de promouvoir l'ENSIIE en dehors de ses murs, c'est tout naturellement que nous avons choisi d'en faire le sujet de notre projet web. Cette promotion se fait principalement par la participation de nos élèves aux forums d'orientation de leurs établissements d'origine ainsi qu'à divers événements en lien direct avec l'informatique, mais également par la réalisation d'une plaquette Alpha ainsi que la production et la vente de goodies.

L'association possède à l'heure actuelle un site web hébergé à l'adresse <http://iimage.iens.net/>, mais il présente de très nombreux défauts. En effet, la connexion n'est pas opérationnelle, ce qui empêche par la même occasion de faire une distinction entre les différents niveaux d'utilisateurs.

De fait, nous avons pour idée de réaliser un nouveau site pour IImagE, permettant de présenter l'association mais également de :

- Consulter la liste des forums précédemment visités ou requérant la présence d'élèves (explicitement demandée par la réception de messages sur notre mailing list) ;
- Donner la possibilité à un élève de s'inscrire sur l'un de ces forums ainsi que de nous envoyer les pièces justificatives nécessaires au remboursement de son trajet ;
- Contacter l'équipe d'IImagE en cas de problème ou pour nous suggérer d'ajouter un forum ou événement auquel il aimerait participer en notre nom.

Nous avons également pour ambition de créer un site pour l'InédItE, la boutique de l'école, afin de présenter les différents produits disponibles à la vente et éventuellement de fournir un système de paiement en ligne. Toutefois, nous savions déjà que cette partie du projet serait probablement trop longue à implémenter et donc impossible à livrer dans le temps imparti.

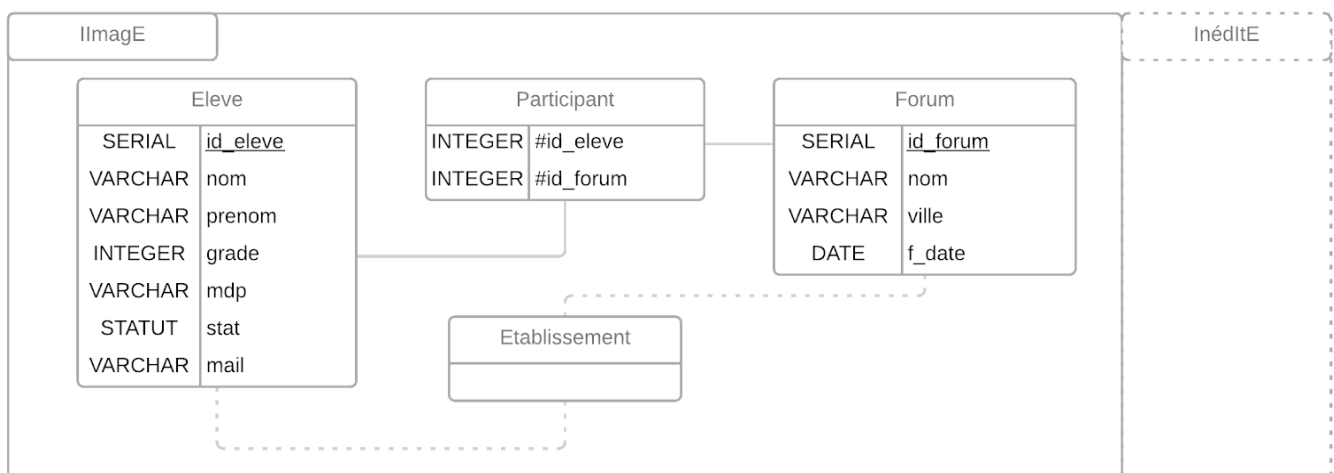
Pour réaliser ce site, nous souhaitons avoir recours à la base de données d'Arise, ce qui nous aurait permis de mettre notre site en production dès la fin de son implémentation. En effet, les utilisateurs du site que nous voulions développer étant tous des élèves de l'école, créer un système d'authentification compatible avec l'architecture mise en place par Arise nous assurait du fait que notre projet serait utilisable et utilisé par le public que nous visions.

## II. Limites et contraintes

### Base de données

À l'origine, nous avions prévu d'utiliser un schéma de table relativement simple, avec en autres :

- Une table **Eleve** contenant toutes les informations relatives aux utilisateurs, notamment leurs identifiants ainsi que leur grade (membre ou non de l'association) ;
- Une table **Forum** précisant entre autres son lieu et sa date ;
- Une table **Etablissement**, reliée à la fois à Forum et à Élève permettant de dresser une liste des anciens élèves par lycée, université ou IUT ;
- Une table de liaison **Participant** permettant de lister chaque participation d'un ou plusieurs élèves à un forum ;
- Des tables supplémentaires en fonction de l'ajout effectif ou non de l'InéditE à notre projet.



Néanmoins, cette structure de données a dû être redéfinie pour s'adapter au projet que nous avons finalement développé. Ainsi, les tables que nous avons choisies de conserver sont représentées en trait plein sur la figure suivante, et celle que nous n'avons finalement pas implémentées en trait discontinu.

### Structuration du code

Afin de structurer notre projet, nous avons opté pour une architecture suivant le motif MVC. Néanmoins, dans la mesure où notre code ne comportait pas un grand nombre de fichiers, nous avons regroupé les vues et les contrôleurs. La liaison avec la base de données (le modèle) reste toutefois séparée du reste de nos fichiers, de même que les instructions SQL qui se situent dans un répertoire data.

## Incompatibilité des technologies

Nous avons très vite réalisé que l'utilisation de Docker ne permettait pas d'exploiter l'architecture mise en place par Arise. Nous avons donc dû repenser la conception de notre projet, ce qui nous a toutefois permis de nous affranchir des contraintes liées à la continuité entre notre code et le leur.

## III. Travail en équipe et répartition des tâches

### Outils de travail

Afin d'optimiser notre méthode de travail en équipe, nous avons choisi d'utiliser plusieurs outils tels que :

- un git commun, hébergé sur [git.iiens.net](https://git.iiens.net) ;
- une To-Do List, régulièrement mise à jour en fonction des disponibilités et compétences de chacun ;
- un serveur Discord consacré au projet.

### Répartition des tâches

En ce qui concerne la répartition des tâches, elle s'est faite de la manière suivante :

Membre du groupe	Travail attribué
Géraldine BAILLEUL	<ul style="list-style-type: none"><li>• Rédaction de toutes les vues</li><li>• Rédaction des contrôleurs (authentification, profil modifiable, gestion des rangs d'utilisateurs, implémentation du CRUD, etc.)</li><li>• Rédaction des scripts en javascript (validation des formulaires)</li></ul>
Mathis BOITEAU	
Dorian DOAN	
Alexandra GAUTHIER	<ul style="list-style-type: none"><li>• Gestion de la base de données</li><li>• Rédaction des modèles</li></ul>
Romain TIRBISCH	<ul style="list-style-type: none"><li>• Conception graphique</li><li>• Gestion du CSS</li></ul>

En pratique néanmoins, nous avons passé la majeure partie du développement de notre projet web à travailler ensemble (physiquement), ce qui nous a permis d'interagir directement les uns avec les autres. Cela nous a également permis de répartir le travail très efficacement, en fonction de l'avancement de chaque membre du groupe. Typiquement, si la tâche qui nous était affectée dépendait d'une fonction en cours d'implémentation par un

autre membre du groupe, il nous était possible de la mettre en attente et de nous concentrer sur un autre aspect du développement.

De plus, cela nous permettait de nous entraider lorsque nous étions confrontés à des difficultés. Notamment, lorsqu'une étape importante du développement posait problème et bloquait donc complètement un membre du groupe, nous nous mettions à plusieurs afin de le résoudre, lui permettant ainsi de reprendre rapidement son avancement.

## IV. Difficultés rencontrées

### Technologies employées

Nous avons rencontré énormément de difficultés techniques vis-à-vis des technologies utilisées dans le développement de notre projet. Cela venait d'une part du fait que n'avions que peu d'expérience en programmation web, et s'expliquait d'autre part par les outils qui ne permettaient pas une prise en main intuitive. Plus particulièrement, le déploiement du projet sur Docker a été la source d'erreurs la plus récurrente, car nous ne savions pas comment l'utiliser de manière fonctionnelle (tout en ayant conscience du fait qu'un emploi correct de cette technologie pouvait nous apporter beaucoup).

En pratique, un seul membre de notre groupe a pu faire fonctionner Docker sur sa machine personnelle. Alexandra GAUTHIER a dû donc assurer seule la cohésion entre les différentes pages que nous codions et l'architecture fournie par Docker. Beaucoup de temps a été perdu à écrire les fonctions permettant d'interroger la base de données, alors que les vues et les contrôleurs étaient souvent fonctionnels.

Plus particulièrement, de ce problème a découlé une mise en place tardive de l'authentification et de la gestion des sessions, ce qui a été bloquant pour le développement de toutes les fonctionnalités liées à l'ajout, l'édition et la suppression d'éléments donnés.

### Redéfinition de nos priorités

Nous avons rapidement réalisé que le retard accumulé ne nous permettrait pas d'implémenter tout ce que nous avons en tête. Nous avons donc dû sacrifier beaucoup de fonctions qui nous tenaient à coeur, et reléguer à un rang de priorité plus bas certains aspects de notre développement.

Cela explique notamment pourquoi nous avons dû renoncer à décomposer notre projet de manière optimale afin de respecter le motif MVC, et également pourquoi nous n'avons pas pu développer l'intégralité de notre architecture de base de données. Par ailleurs, nous avons également abandonné la fonctionnalité permettant d'héberger ses pièces justificatives directement à partir du site.

## V. Conclusion

Finalement, nous avons su surmonter les difficultés auxquelles nous avons été confrontés, à la fois en faisant des concessions vis-à-vis de nos prévisions initiales mais également en apprenant à maîtriser les technologies employées. Cet apprentissage a souvent été complexe, mais nous avons réussi à mettre à profit une partie des facilités promises par Docker, ce qui nous a permis de compléter les fonctions essentielles de notre site. Par ailleurs, la collaboration et l'entraide constante au sein de l'équipe ont été un moteur important de notre motivation.

Enfin, nous aurions aimé aller jusqu'au bout de nos ambitions en peaufinant tous les détails qui nous tenaient à cœur, mais sommes de tout même très satisfaits du résultat. De plus, nous avons énormément appris au cours du développement de notre site, et pouvoir créer un projet du début à la fin s'est révélé être hautement gratifiant.