

DEPARTEMENT INFORMATIQUE - IUT 2 GRENOBLE



Année Universitaire 2022-2023

RAPPORT DE PROJET

RAPPORT DE PROJET
Améliorer et optimiser une application existante

Équipe 03

Présenté par

Nacer BERKANE
Samuel BILLOT
Johaïna BOUDIAF
Gabriel HALUS
Farès MEDJAHED

Résumé long :

Notre projet traite d'une application dont l'objectif est de faciliter la recherche de stage en permettant aux étudiants de consulter les offres disponibles et de suivre leurs candidatures jusqu'à l'acceptation d'une proposition parmi plusieurs réponses positives. Cette application concerne deux parties prenantes : le responsable des stages, chargé de gérer les offres de stage, et les étudiants qui postulent à ces offres. L'ensemble de la solution est composée de quatre modules logiciels. L'application front office pour mobile est destinée aux étudiants. Elle stocke localement des informations comme l'historique des candidatures. Elle permet à l'étudiant de s'authentifier avec un compte, de consulter les offres disponibles, de marquer les offres qui l'intéressent. Il peut également déposer une candidature et la suivre jusqu'à l'obtention de réponses positives. Il met alors à jour l'état de ses candidatures en cours et peut valider l'une d'elles si elle est acceptée. L'application web est utile au responsable des stages qui alimente la base des offres et consulte un tableau de bord de l'avancement des recherches. Un service web gère l'authentification ainsi que l'ensemble des actions. Une base de données est liée au service web, assurant la persistance des données. Ces derniers sont hébergés sur une [machine virtuelle](#) faisant office de serveur. En partant d'une application existante, notre équipe a réalisé une évaluation complète de la conception, des performances, de la qualité du logiciel, de l'ergonomie et de la sécurité, afin d'identifier les points d'amélioration nécessaires dans chacun de ces domaines. Dans cette optique, nous avons procédé à l'optimisation de la base de données, la définition de jeux de tests, la virtualisation des services, la gestion au mieux du cycle de vie de l'application mobile ou encore l'ajout de l'interactivité [JavaScript](#) sécurisé au [back-office Symfony](#). Nous avons obtenu une solution conforme aux attentes des utilisateurs et qui respecte les critères prédéfinis. On constate un meilleur design et des interactions plus fluides avec les différents modules de l'application. Ce projet a nécessité le recours à diverses ressources telles que le développement web et mobile, la qualité de développement, la virtualisation, les mathématiques, la qualité du relationnel, la gestion de projet, la communication ou encore l'anglais. Nous avons dû respecter un planning précis et déposer différents livrables afin de répondre à toutes les attentes vis-à-vis de ce projet en matière de qualité d'analyse des besoins, de la conception, de qualité de développement logiciel, de la mise en fonctionnement de l'application et de qualité relatif au processus de développement.

Résumé court :

Amélioration d'une application de gestion de recherche de stage pour des étudiants grâce à une analyse complète de l'existant

Notre projet traite d'une application dont l'objectif est de faciliter la recherche de stage en permettant aux étudiants de consulter les offres disponibles et de suivre leurs candidatures jusqu'à l'acceptation d'une proposition parmi plusieurs réponses positives. Notre application concerne deux parties prenantes. Elle a pour but de permettre au responsable des stages de gérer les offres de stage et les étudiants qui interagissent avec ces offres. L'application mobile est destinée aux étudiants tandis que l'application web est utile au responsable des stages qui alimente la base des offres et consulte un tableau de bord de l'avancement des recherches. En partant d'une application existante, notre équipe a réalisé une évaluation complète de la conception, des performances, de la qualité du logiciel, de l'ergonomie et de la sécurité, afin d'identifier les points d'amélioration nécessaires dans chacun de ces domaines. Dans cette optique, nous avons procédé à l'optimisation de la base de données, la définition de jeux de tests, la virtualisation des services, la gestion au mieux du cycle de vie de l'application mobile ou encore l'ajout de l'interactivité [JavaScript](#) sécurisé au [back-office Symfony](#). Nous avons obtenu une solution conforme aux attentes des utilisateurs et qui respecte les critères prédéfinis. On constate un meilleur design et des interactions plus fluides avec les différents modules de l'application. Ce projet a nécessité le recours à diverses ressources telles que le développement web et mobile, la qualité de développement, la virtualisation, les mathématiques, la qualité du relationnel, la gestion de projet, la communication ou encore l'anglais.

Mots clés : carnet de stage, amélioration d'application existante, performance, qualité logicielle, web, mobile, front office, back office

Abstract :**Improvement of an Internship Management Application for Students by Analyzing Thoroughly the Existing Solution**

Nacer BERKANE, Johaïna BOUDIAF, Samuel BILLOT, Fares MEDJAHED, Gabriel HALUS

Our project consists of an application whose objective is to facilitate the search for an internship by allowing students to consult the available offers and to follow their applications until the acceptance of a proposal among several positive answers. Our application concerns two stakeholders. It aims to allow the internship manager to manage internship offers and the students who interact with these offers. The mobile application is intended for students while the web application is useful for the internship manager who feeds the database of offers and consults a dashboard of the progress of the research. Starting from an existing application, our team performed a complete evaluation of the design, performance, software quality, ergonomics and security, in order to identify the necessary improvement points in each of these areas. To this end, we optimized the database, defined test sets, virtualized services, managed the mobile application's lifecycle as well as possible, and added secure [JavaScript](#) interactivity to the [PHP](#) back office. We obtained a solution that meets the expectations of the users and respects the predefined criteria. We have seen a better design and more fluid interactions with the different modules of the application. This project required the use of various resources such as web and mobile development, quality of development, virtualization, mathematics, relational quality, project management, communication and English.

Keywords : internship book, existing application improvement, performance, software quality, web, mobile, front office, back office

Table des matières :

I. Introduction.....	6
II. Présentation du contexte.....	7
III. Analyse des besoins et spécifications.....	8
A.Objectif.....	8
B.Liste des exigences.....	8
C.Cas d'utilisation.....	9
IV. Conception.....	11
A.Conception de la base de données.....	11
B.Conception de l'architecture.....	12
V. Réalisation.....	13
A.Choix techniques.....	13
B.Gestion de projet.....	14
C.Phase d'implémentation.....	14
VI. Interface et expérience utilisateur.....	15
VII. Conclusion.....	16
VII. Glossaire.....	17
IX. Annexes.....	19
A. Maquettes.....	19
B. Modèle de données :.....	20
C. Application Web.....	22

Table des figures :

IV. Conception.....	12
C.Modèle de données.....	13
Figure 1 - Exemple de Cards concernant les différents états d'une recherche de stage.....	14
VI. Interface et expérience utilisateur.....	18
Figure 2 - Exemple de Cards concernant les différents états d'une recherche de stage.....	18
IX. Annexes.....	22
A. Maquettes.....	22
Figure 3 - Page d'authentification du back-office.....	22
Figure 4 - Page de gestion de stage visualisable par l'étudiant.....	22
B. Modèle de données :.....	23
Figure 5 - Modèle de données de base.....	23
Figure 6 - Modèle de données en étoile.....	24
C. Application Web.....	25
Figure 7 - Page web visible par les admins permettant de consulter le statut de la recherche de stage de chacun.....	25
Figure 8 - Page web visible par les étudiants pour voir le status de leurs demandes, a quelles entreprises ils n'ont pas fait de demandes et autres.....	26

I. Introduction

Notre projet porte sur l'amélioration d'une application existante, nommée "Mon carnet de suivi de recherche de stage", destinée à faciliter la recherche de stage pour les étudiants. Cette application rencontre actuellement des lacunes et des défauts qu'il est nécessaire de corriger, notamment en termes de conception, de développement et de mise en fonctionnement.

Le but de ce projet est de mettre en pratique les connaissances en développement logiciel et en gestion de projet en travaillant en équipe pour améliorer cette application existante. Les améliorations envisagées concernent plusieurs aspects, comme l'expérience utilisateur, la spécification du processus de recherche de stage, la qualité du développement logiciel et la mise en fonctionnement de l'application.

Ce rapport final présente en détail les améliorations apportées à l'application, la répartition des tâches au sein de l'équipe, ainsi que le processus de développement.

II. Présentation du contexte

Le cadre général de travail pour notre projet était bien défini et organisé. Nous étions un groupe de cinq personnes travaillant sur le développement d'une application mobile et nous avons eu accès aux ordinateurs et aux logiciels de l'IUT pour mener à bien notre mission. Nous avons disposé d'environ un mois pour réaliser ce projet, dont une semaine entière dédiée au projet, pendant laquelle nous avons travaillé ensemble de manière intensive pour faire avancer le projet.

Nous avons défini les tâches de chacun de manière claire et précise, afin d'éviter tout chevauchement de responsabilités ou confusion. Certains membres de l'équipe se sont concentrés sur le développement de l'application mobile, d'autres sur les aspects mathématiques, tandis que d'autres travaillaient sur le projet [*Symfony*](#).

Nous avons également utilisé des outils de communication tels que Discord pour rester en contact régulier les uns avec les autres. Nous avons également travaillé en utilisant des plateformes de partage de fichiers telles que Google Drive et Gitlab, ce qui nous a permis de collaborer efficacement et de travailler ensemble en temps réel.

Nous avons défini des objectifs clairs et spécifiques pour chaque étape du projet, ainsi que des échéances et des délais réalistes pour respecter les délais impartis. Nous avons également prévu des outils de suivi et d'évaluation pour mesurer l'avancement du projet et évaluer les résultats obtenus, ce qui nous a permis d'ajuster notre cadre général de travail si nécessaire.

Dans l'ensemble, notre cadre général de travail était solide et bien organisé, ce qui nous a permis de travailler efficacement en équipe et de mener à bien notre projet.

III. Analyse des besoins et spécifications

A. Objectif

Notre projet vise à évaluer l'application et à identifier les améliorations nécessaires pour améliorer sa conception, son ergonomie, ses performances et sa qualité logicielle. Pour ce faire, nous avons pour objectif d'améliorer les différentes étapes du cycle de développement logiciel de cette application, en identifiant les améliorations et les optimisations nécessaires pour chaque module, y compris une application mobile, une application web, un service web et une base de données pour la persistance des données.

L'application sera conçue de manière à ce que les étudiants puissent facilement s'authentifier avec leur compte pour consulter les offres de stage disponibles. Ils pourront également marquer les offres qui les intéressent et déclarer une candidature. Le suivi des candidatures sera possible jusqu'à l'obtention d'une ou plusieurs réponses positives en mettant à jour l'état de la candidature en cours. Enfin, la recherche sera finalisée en validant une candidature acceptée parmi l'ensemble des candidatures acceptées.

Notre objectif principal est d'améliorer la qualité globale de l'application et d'optimiser son fonctionnement afin de fournir une meilleure expérience utilisateur et des fonctionnalités plus efficaces. En évaluant chaque aspect de l'application et en identifiant les améliorations nécessaires, nous espérons atteindre cet objectif et fournir une solution plus efficace et conviviale pour les utilisateurs qui répondra aux besoins des étudiants en matière de recherche de stage..

B. Liste des exigences

Gestion des offres de stage : l'application doit permettre au responsable des stages de publier, modifier et supprimer des offres de stage.

Recherche d'offres de stage : les étudiants doivent pouvoir rechercher des offres de stage en utilisant différents critères tels que le domaine, la localisation, etc.

Suivi des candidatures : le responsable doit pouvoir consulter l'état des candidatures de chaque étudiant (en attente, acceptée, refusée, etc.).

Gestion des étudiants : le responsable des stages doit pouvoir gérer les informations relatives aux étudiants (nom, prénom, filière, etc.).

Interface utilisateur conviviale : l'application doit être facile à utiliser et intuitive pour les deux types d'utilisateurs (le responsable des stages et les étudiants).

Sécurité des données : l'application doit garantir la sécurité des données personnelles des utilisateurs (noms, adresses, etc.) et des informations sensibles (CV, lettres de motivation, etc.).

Disponibilité de l'application : l'application doit être accessible à tout moment et disponible sur différents types de dispositifs (ordinateurs, smartphones, tablettes, etc.).

C.Cas d'utilisation

Le scénario décrit de manière détaillée comment un étudiant peut utiliser l'application mobile "Mon carnet de suivi de recherche de stage" pour consulter les offres de stage disponibles, marquer celles qui l'intéressent, déclarer une candidature et suivre son évolution jusqu'à l'obtention d'une réponse positive ou négative.

Pour commencer, l'étudiant doit être authentifié et avoir accès à l'application mobile, il devra donc se connecter avec ses identifiants et son mot de passe . Une fois connecté, il peut accéder à la liste des offres de stage disponibles et sélectionner celles qui l'intéressent en les marquant pour y revenir ultérieurement.

Lorsqu'il souhaite postuler à une offre, l'étudiant peut déclarer sa candidature pour celle-ci. Il peut ensuite suivre l'évolution de sa candidature en consultant régulièrement son état dans l'application. Il est important de noter que l'étudiant peut consulter l'état de toutes ses candidatures en cours dans l'application mobile "Mon carnet de suivi de recherche de stage". Cela lui permet de garder une vue d'ensemble sur l'avancement de ses demandes et de savoir à tout moment quelles offres de stage il a déjà postulées.

Si la réponse à sa candidature est positive, l'étudiant peut finaliser sa recherche en acceptant la proposition. En revanche, si la réponse est négative, l'étudiant peut continuer sa recherche en consultant les autres offres disponibles. Il est important de noter que l'étudiant peut également abandonner une candidature ou modifier les offres de stage marquées comme intéressantes à tout moment.

Enfin, pour faciliter sa recherche de stage, l'étudiant peut également ajouter une date de rappel pour les offres de stage qui l'intéressent, afin de ne pas les oublier et de pouvoir y revenir plus tard si besoin. En résumé, l'application mobile "Mon carnet de suivi de recherche de stage" permet à l'étudiant de consulter les offres de stage, marquer celles qui l'intéressent, déclarer une candidature et suivre son évolution jusqu'à l'obtention d'une réponse positive ou négative, tout en offrant une vue d'ensemble sur l'état de ses candidatures en cours.

IV. Conception

A. Conception de la base de données

Nous avons amélioré la qualité de notre base de données en la normalisant jusqu'à la troisième [forme normale](#) (3FN) pour éviter les [anomalies](#) de mise à jour, de suppression et d'insertion de données. Nous avons commencé par diviser les tables en plusieurs tables plus petites, en veillant à ce que chaque table contienne uniquement des informations relatives à une seule entité ou relation. Cette division nous a permis d'éviter la redondance de données et de garantir l'intégrité des données.

De plus, nous avons créé des relations entre les tables en utilisant des [clés étrangères](#) pour faciliter les [requêtes](#) et assurer une meilleure efficacité dans l'utilisation de la base de données.

Pour aller plus loin, nous avons étudié les relations et les [dépendances fonctionnelles](#) existantes pour effectuer des modifications spécifiques. Nous avons créé deux nouvelles tables, intitulées Mot_cle et "rôles", pour séparer les [données non-atomiques](#) dans les tables existantes. La première a pour [clé primaire](#) "intitule, mot_cle" et la seconde a pour [clé primaire](#) "login, rôle".

Enfin, nous avons ajouté une nouvelle [colonne](#) "login" à la table "Compte étudiant" pour stocker le login de l'étudiant, et avons supprimé la [colonne](#) "roles" de cette table car elle a été déplacée dans la nouvelle table "roles". Ces modifications ont éliminé les [dépendances fonctionnelles non-atomiques](#) et ont ainsi amélioré la normalisation de la base de données.

En somme, la normalisation de notre base de données jusqu'à la 3FN, ainsi que les modifications spécifiques apportées pour éliminer les [dépendances fonctionnelles non-atomiques](#), ont permis d'améliorer considérablement la qualité de nos données et d'assurer une meilleure efficacité dans leur utilisation.

B. Conception de l'architecture

Nous avons décidé de maintenir l'architecture *MVC* (Modèle-Vue-Contrôleur) déjà présente dans l'application pour plusieurs raisons.

Tout d'abord, cette architecture est largement utilisée dans l'industrie du développement de logiciels, ce qui signifie que de nombreux développeurs la connaissent déjà et sont donc capables de travailler efficacement sur l'application.

En outre, l'architecture *MVC* est particulièrement adaptée aux applications Web car elle permet une séparation claire des différentes responsabilités de l'application. Le modèle représente la logique métier de l'application, la vue représente l'interface utilisateur et le contrôleur gère les interactions entre le modèle et la vue. Cette séparation facilite la maintenance et l'évolutivité de l'application.

Enfin, l'utilisation de l'architecture *MVC* facilite la collaboration entre les membres de l'équipe de développement. Chacun peut travailler sur une partie différente de l'application en même temps, sans affecter le travail des autres membres.

En résumé, nous avons choisi de maintenir l'architecture *MVC* dans l'application en raison de sa popularité, de son adaptation aux applications Web et de sa capacité à faciliter la collaboration entre les membres de l'équipe de développement.

C. Modèle de données

Dimensions : étudiant, offre, parcours, ville, pays, année, mois.

Indicateurs : nb_offres_consultées, nb_offres_dispo, nb_candidatures, moy_candidatures_pour_stage, nb_candidatures_refusées, nb_candidatures_acceptées.

Figure 1 - Exemple de Cards concernant les différents états d'une recherche de stage

	étudiant	offre	parcours	ville	pays	année	mois
nb_offres_consultées	X	X				X	X
nb_offres_dispo		X	X	X	X	X	X
nb_candidatures	X	X				X	X
moy_candidatures_pour_stage	X	X				X	X
nb_stages_pourvus	X	X				X	X
nb_candidatures_refusées	X	X				X	X

Justification des choix des indicateurs :

nb_offres_consultées : Suivre le nombre d'offres qu'un étudiant consulte permet au responsable des stages d'avoir une première idée de l'avancement de la recherche de chacun des étudiants. Dans le cas où ce chiffre est jugé trop faible, le responsable peut avertir, par exemple par mail, l'étudiant en question afin que celui-ci intensifie sa recherche de stage.

nb_offres_dispo : Prendre connaissance du nombre d'offres disponibles, pour chaque parcours, est utile afin de le comparer au nombre d'élèves actuels et d'en déduire si chacun des étudiants a une chance de voir un stage lui être pourvu. On pourra également être amené à comparer ce chiffre à ceux des années précédentes à des fins statistiques.

nb_candidatures : Tout comme pour le nombre d'offres consultées, cet indicateur sert à jauger l'implication de chaque étudiant dans sa recherche de stage. Du point de vue des offres, il est alors possible de savoir lesquelles reçoivent le plus de candidatures et donc de potentiellement proposer plus d'offres similaires à celles-ci.

moy_candidatures_pour_stage : Cet indicateur permet de prendre en compte le nombre total de candidatures, toute offre et tout parcours confondu, et à partir de là d'établir une moyenne de candidatures par offre.

nb_stages_pourvus : Connaître le nombre de stages pourvus permet d'établir un ratio par rapport au nombre de candidatures totale.

nb_candidatures_refusées : Connaître le nombre de candidatures refusées permet d'établir un ratio par rapport au nombre de candidatures totale.

V. Réalisation

A. Choix techniques

Dans les langages que nous utilisons, une partie découle des contraintes liées au projet. En effet, comme nous avions du code qui nous était donné, nous devions manier les langages qui étaient utilisés dans le code, notamment [Symfony](#) et [Twig](#), un moteur de template moderne pour [PHP](#). Ensuite, pour le style des pages, il était fait avec [Bootstrap](#). Nous devions alors choisir entre garder [Bootstrap](#) et essayer d'apprendre cette librairie pour pouvoir modifier le style des pages, ou alors enlever [Bootstrap](#) du projet, et refaire le style des pages en [CSS](#). C'est vers cette deuxième option que nous nous sommes tournés. Puis, pour faire fonctionner le site, nous avons dû utiliser le langage [SQL](#) qui nous était donné dans le code. De plus, l'utilisation de [JavaScript](#) nous était indispensable pour écrire nos tests fonctionnels (en [Cypress](#)). Les langages que nous utilisons sont donc : [Symfony](#), [Twig](#), [CSS](#), [JavaScript](#), [SQL](#).

Pour les outils que nous avons utilisés, au niveau de la communication, nous nous sommes tournés vers *Discord*, puisque tout le monde dans le groupe l'utilise et qu'il permet de s'envoyer des fichiers tout en étant simple d'utilisation. Comme autre outil de communication, nous nous sommes aussi servis de Google Drive. En ce qui concerne les outils techniques, nous utilisons [Figma](#), afin de produire des maquettes qualitatives pour pouvoir avoir une idée de ce à quoi va ressembler le site avant de se lancer dans le code. De plus, c'est un outil que certains membres du groupe maîtrisaient particulièrement bien. [PHPStorm](#) a aussi été un autre outil très pratique pour coder en [Symfony](#), puisque, comme son nom l'indique, cet environnement de travail est fait pour le langage [PHP](#). En adéquation avec [PHPStorm](#), [Visual Studio Code](#) a été un autre environnement de travail utilisé pour les tests fonctionnels en [JavaScript](#) et aussi pour coder le [CSS](#) du site. L'utilisation de [ChatGPT](#) a pu s'avérer utile lorsque nous avons rencontré des erreurs dans le code, puisqu'il permet notamment de le déboguer.

B. Gestion de projet

Concernant l'organisation de notre groupe projet, nous avons tout d'abord commencé par élire un chef de projet (Nacer), qui nous guidera pendant toute la durée de celui-ci. Nous avons par la suite décidé des outils que nous allions utiliser, et aussi discuté des compétences et des préférences de chacun pour répartir le travail de manière optimale. La répartition du travail s'est faite aussi petit à petit en fonction des besoins et des contraintes que nous avions.

Nous avons alors réparti le travail de la manière suivante : Gabriel s'est occupé de la [machine virtuelle](#) et des [dockers](#). Fares, quant à lui, a plutôt fait le mobile. Johaïna et Samuel ont fait la partie mathématiques du projet, que nous n'allons pas traiter dans ce rapport, les tests fonctionnels et ont aussi travaillé sur le code [Symfony](#) ([back-office](#)) avec Nacer. De plus, comme ce projet n'était pas extrêmement ambitieux, nous n'avons pas fait de diagramme de GANTT, ou de définition de critères de découpage.

C. Phase d'implémentation

L'implémentation est d'abord passée par une assez longue phase de compréhension du code fourni avec un travail de documentation assez conséquent. A chaque fois que l'on repérait une erreur dans le code, nous la répertorions sans y toucher pour que l'on puisse d'abord identifier un maximum d'erreurs et de mauvaises pratiques. Une fois répertoriées, nous sommes allés voir si on avait les compétences et assez compris le code pour pouvoir corriger ces erreurs. Sinon, on passait à une autre étape pour mieux comprendre le code. De plus, au niveau de la base de données, nous avons décidé de faire un travail de traduction de la base fournie en [PostgreSQL](#) pour la passer en [SQLite](#). Cela permet de travailler de manière optimale depuis chez nous puisque l'on avait pas tous [PostgreSQL](#) de configuré sur nos machines et nous n'avions pas envie d'attendre que la [machine virtuelle](#) soit en place pour pouvoir coder. Nous avons alors écrit les tests fonctionnels et en parallèle, nous avons réalisé une ébauche de la maquette du [back-office](#) pour la rendre plus ergonomique.

Concernant les tests, nous avons utilisé des outils comme Wave afin d'évaluer la qualité de notre interface. Nous avons également . Puis, nous avons décidé de retirer [Bootstrap](#) du code pour avoir une base plus saine et pouvoir appliquer le style avec du [CSS](#) que nous réaliserons. Nous sommes ensuite allés plonger dans le code en détail pour refaire les interfaces et continuer de corriger les erreurs. Nous avons notamment amélioré grandement la gestion des erreurs et la sécurité des formulaires. Dans le même temps, les [dockers](#) ont été mis en place.

VI. Interface et expérience utilisateur

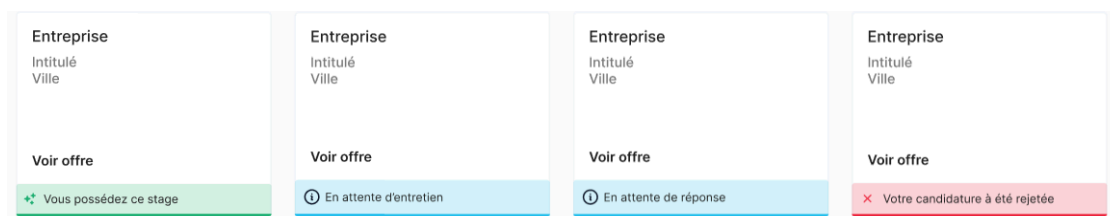
Dans l'optique de proposer la meilleure expérience utilisateur, nous avons étudié les différents points faibles présents dans les applications données pour tenter de remédier au plus grand nombre, c'est durant cette phase de conception que nous en sommes venu à nous fixer des objectifs clairs. Et pour fixer ces derniers nous avons eu à sélectionner différents critères qualités, qui nous ont par la suite permis de maquetter les différentes applications.

Un des premiers points qui nous tenait à cœur était de reprendre l'interface présente pour le [back-office](#) et d'en faire quelque chose de plus personnel et plus visuel, la précédente était principalement basée sur des tableaux. C'est pourquoi nous avons choisi un système de "Cards" (c.f. annexe 2), ces dernières comprendront les différentes informations nécessaire pour les utilisateurs, l'utilisateur pourra trier ces différentes cards en fonction d'informations souhaitées (e.g : sur la page contenant toutes les offres proposées aux étudiants, ces derniers pourront trier les offres en fonction des langages de programmations et technologies demandées). Nous avons pris le soin de garder toutes les fonctionnalités importantes comme la recherche et le tri de certains critères.

La notion de guidage de l'application [back-office](#) donnais se trouvait être optimisable, c'est pourquoi nous avons également travaillé sur son amélioration, les différentes pages ayant similairement la même apparence et mode de fonctionnement, l'utilisateur ne pourra alors pas se trouver perdu lors de l'utilisation de notre application.

Quant à l'application mobile, en dehors du fix de nombreux bugs, nous avons également refait une grande partie de l'interface, intégrant notamment une barre de navigation présente en bas de l'application, permettant une meilleure navigation entre les différentes offres et candidatures.

Figure 2 - Exemple de Cards concernant les différents états d'une recherche de stage



VII. Conclusion

Améliorer ces applications s'est avéré être une tâche riche en savoirs que nous avons pu extraire; nous avons eu à faire appel à nos connaissances tout en en acquérant de nouvelles, nous avons appris à nous connaître pour certains et à travailler ensemble. Avec plus de temps et surtout une meilleure gestion des différentes tâches, nous aurions pu accomplir l'ensemble des objectifs que nous nous étions fixés, mais cela n'enlève en rien que nous sommes fiers du travail accompli et de l'effort collectif, chacun a su donner du sien et apprendre à chacun.

Hormis le fait que nous avons déjà des connaissances quant aux technologies utilisées, ce projet nous a permis de les approfondir et de les mettre en action dans un cadre réel, et le l'ensemble du groupe est d'accord pour dire que l'expérience était véritablement profitable à tous.

VII. Glossaire

Symfony	Cadre de travail pour le langage PHP .
PHP	Langage principalement utilisé pour produire des pages Web.
JavaScript	Langage permettant principalement de rendre des pages Web interactives.
PostgreSQL	Système de gestion de base de données relationnelle et objet.
CSS	Langage permettant la restructuration et la personnalisation des pages HTML.
Bootstrap	Collection d'outils contenant du code HTML et CSS pour faciliter la création de pages web.
Forme Normale	Processus d'organisation de données en tables pour éviter les redondances.
Anomalies	Incohérences de données résultant d'opérations de mise à jour, suppression ou insertion.
Clé étrangère	Attribut d'une table reliant à un attribut d'une autre table.
Requêtes	Instructions envoyées à une base de données pour récupérer des données.
Données non-atomiques	Données contenant des informations qui peuvent être séparées en parties distinctes.
Clé primaire	Attribut d'une table permettant d'identifier de manière unique chaque enregistrement.
Dépendances fonctionnelles	Relations entre les attributs d'une table qui déterminent d'autres attributs.

Colonne	Dans une table, ensemble des valeurs pour une propriété.
Dépendances fonctionnelles non-atomiques	Dépendances fonctionnelles qui impliquent des données non-atomiques .
Back-office	Service permettant d'alimenter et gérer les informations concernant une plateforme.
Figma	Application web permettant la création de design et autres ressources nécessaires à la création de projets (dans notre cas, maquettes).
Twig	Moteur d'affichage de "templates" utilisé par le langage PHP .
SQL	Langage normalisé servant à exploiter des bases de données relationnelles.
Visual Studio Code	Logiciel d'édition de texte, permettant de coder des logiciels.
PHPStorm	Logiciel d'édition de texte, permettant de coder des logiciels, principalement spécialisé pour PHP.
Cypress	Librairie utilisée pour tester des sites directement depuis un navigateur
ChatGPT	Application d'intelligence artificielle utilisant un réseau neuronal pour répondre à des questions.
Docker	Plateforme permettant la simplification du lancement d'applications dans des conteneurs logiciels.
Machine Virtuelle	Environnement entièrement virtualisé qui fonctionne sur une machine physique.
SQLite	Système de gestion de base de données relationnelle et objet. Les données sont stockées dans un fichier db.

IX. Annexes

A. Maquettes

Figure 3 - Page d'authentification du [back-office](#)

Authentification

Merci d'entrer vos identifiants

Login

Password

Connexion

Connexion étu

Figure 4 - Page de gestion de stage visualisable par l'étudiant

Parcours ▼

Aurore BOREALE - Parcours B

Entreprise Intitulé Ville	Entreprise Intitulé Ville	Entreprise Intitulé Ville	Entreprise Intitulé Ville
Voir offre	Voir offre	Voir offre	Voir offre
✓ Vous possédez ce stage	⌚ En attente d'entretien	⌚ En attente de réponse	✗ Votre candidature a été rejetée

B. Modèle de données :

Figure 5 - Modèle de données de base

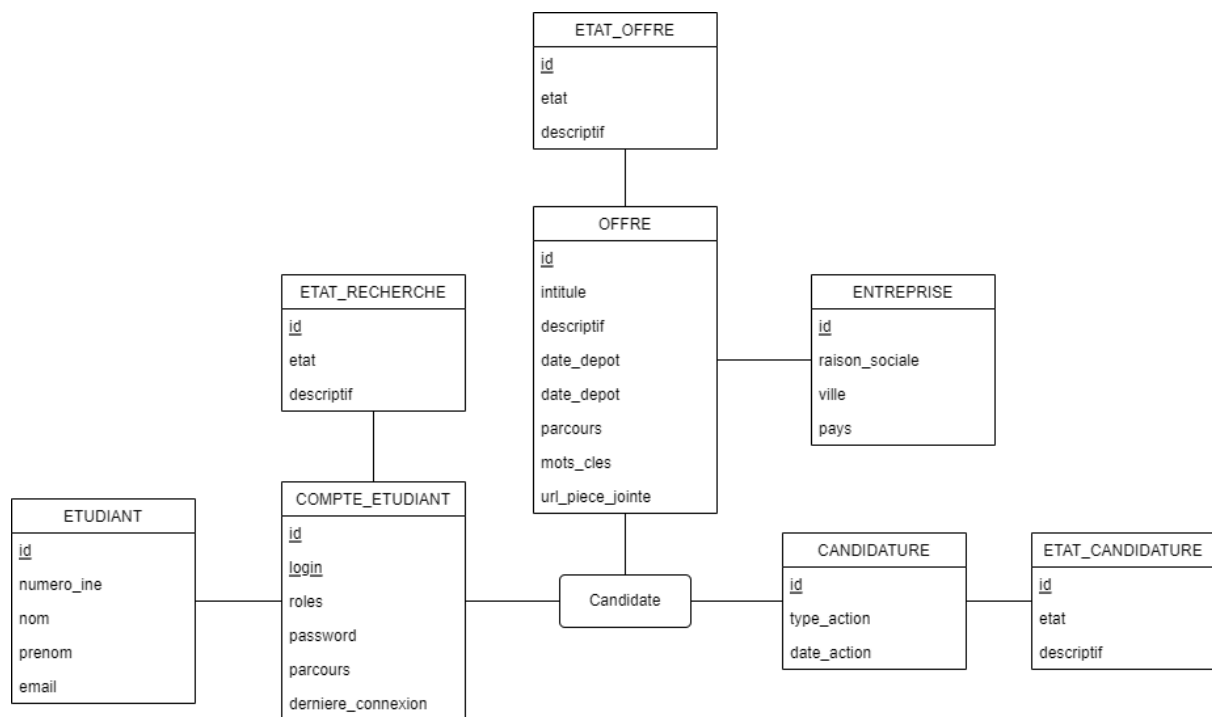
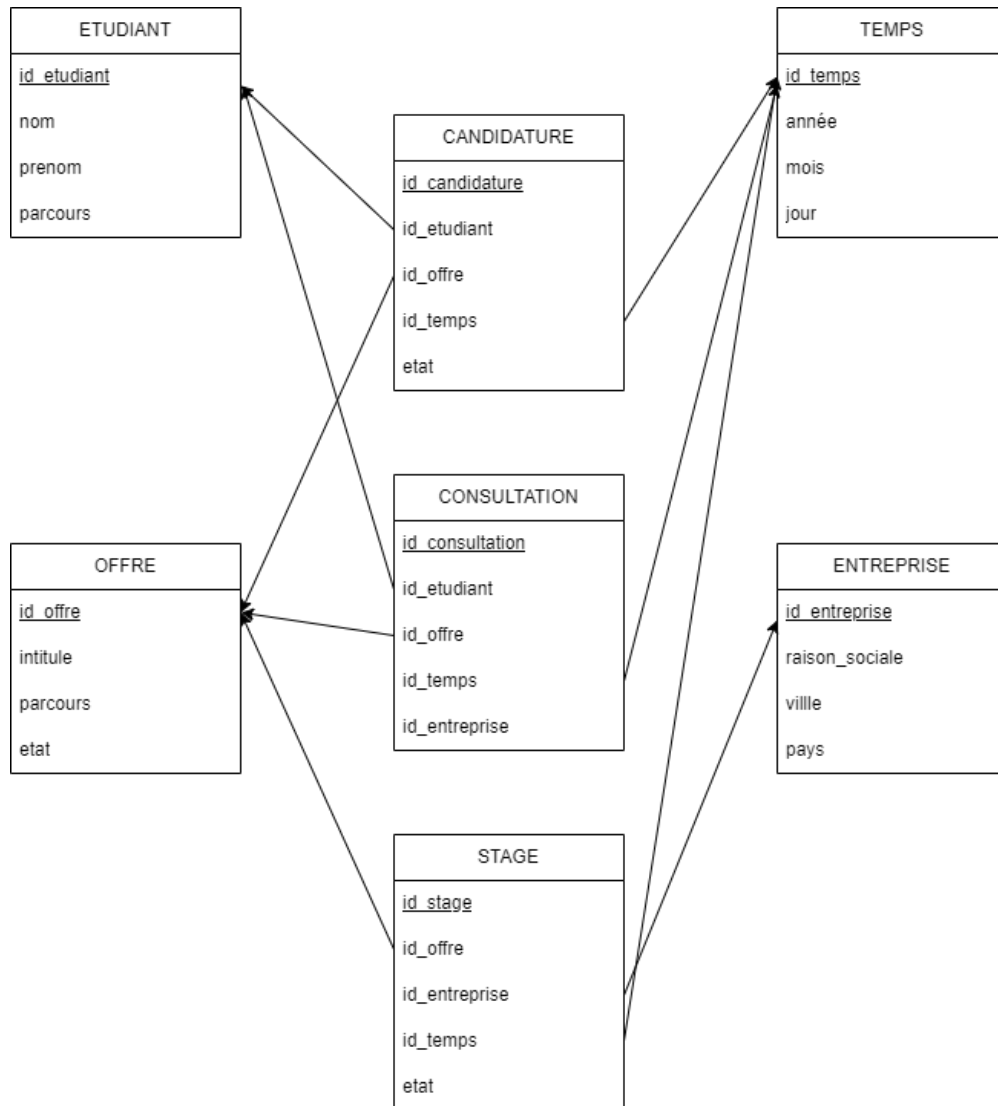


Figure 6 - Modèle de données en étoile



C. Application Web

Figure 7 - Page web visible par les admins permettant de consulter le statut de la recherche de stage de chacun

Trier ▾

Rechercher...

4567890123E

BOREALE Aurore

aurore.boreale@iut2.fr

borealea

Offres consultées : 2

Offres retenues : 1

Candidatures : 1

B

L'étudiant e a fait une ou plusieurs candidatures, il elle attend encore des réponses

9876543210B

KHONNU Alain

alain.khonnu@iut2.fr

khonnua

Offres consultées : 5

Offres retenues : 1

Candidatures : 1

A

L'étudiant e a trouvé un stage, sa recherche est terminée

2345678901C

ENFAITE Mélusine

melusine.enfaite@iut2.fr

enfaitem

Offres consultées : 2

Offres retenues : 1

Candidatures : 1

B

L'étudiant e a fait une ou plusieurs candidatures, il elle attend encore des réponses

5678901234F

HAPLUTARD Jérémie

jeremy.haplutard@iut2.fr

haplutaj

Offres consultées : 0

Offres retenues : 0

Candidatures : 0

A

L'étudiant e n'a pas encore consulté les offres de stage

3456789012D

HERBIEN Jean-Philippe

jean-philippe.herbien@iut2.fr

herbienj

Offres consultées : 1

Offres retenues : 0

Candidatures : 0

A

L'étudiant e a commencé à consulter les offres de stage mais n'a pas encore candidaté

person

Figure 8 - Page web visible par les étudiants pour voir le statut de leurs demandes, a quelles entreprises ils n'ont pas fait de demandes et autres.

BOREALE Aurore - Parcours B			
Trier ▼			
Rechercher.			
JS/PHP/Windows/Linux/Git, ERP Khrone SAS Romans	Linux debian/centos, scripting bash, office, zabbix, logiciel de monitoring INCOM Gières	CSHarp Jeux UNITY 3D, développement Creative VR3D Cannes	Site web LIG Saint Martin d'Hères
SQL / PYTHON SQL*serveur, SSRS/ POWER BI, analyse de données LYNRED Veuvey-Voroize	Tests fonctionnels, Panther, PHP8, Mysql Parker Lord Pont de l'Isère	Architecture Sharepoint, migration, conception Association Championnet Sallanches	Développement d'application Android, GIT, Kotlin ELYXOFT Gillonay
Administration des Systèmes et Réseaux, configuration serveur Linux INSIDIX SAS Seyssins	Développement d'outils internes, Python, SQL HARDIS Seyssinet-Pariset	C# et VB, tests de non-régression Framatome Ugine	Développement web, full stack, Python, Javascript, Framework ReactJS, Angular, Symphony, D3JS, cytoscapeJS, ajax/tiask, mongoDB Reckonect SAS Grenoble
	+ Vous avez candidaté à cette offre	Vous avez consulté cette offre	L'offre n'est plus disponible, un e étudiant e l'a déjà prise : inutile d'y candidater