

PROJET D'APPLICATION

Rapport Final

Projet 16: Extraction de données de l'API AlumnForce et enregistrement en base de données

Client: Centrale Nantes Alumni

Encadrants: Christelle ROUSSEAU, Stéphane BURET

Responsables Techniques: Larissa ALBUQUERQUE NOGUEIRA, Clara MATTOS MEDEIROS

Nantes
20 Décembre 2024

Sommaire

Sommaire	2
DÉFINITIONS	3
1. SYNTHÈSE DU PROJET	4
2. INTRODUCTION	5
2.1. Contexte	5
2.2. Objectifs	6
2.3. Portée du logiciel	6
3. MÉTHODOLOGIE	7
3.1. Méthodologie de développement	7
3.1.1 Tests unitaires	7
3.3. Outils et logiciels utilisés	7
3.4. Planification	8
4. RÉSULTATS	11
4.1. Rendus Finaux	11
4.2. Adhérence au planning initial	12
4.3. Estimation des coûts	13
5. DIFFICULTÉS RENCONTRÉS	13
6. SUITE DU PROJET	14
7. CONCLUSION	15
8. ANNEXES	15
8.1. Dépôt Github du Projet	15
9. RÉFÉRENCES	15

DÉFINITIONS

API : Interface de Programmation d'Application.

CSV : Comma-separated values. Format de fichier de valeurs séparées par des virgules.

Alumnforce : Plateforme (réseau social) gérant les informations des étudiants et diplômés de Centrale Nantes

CNA: Centrale Nantes Alumni.

DSI: Direction des Systèmes d'Information de Centrale Nantes.

1. SYNTHÈSE DU PROJET

Ce projet s'inscrit dans le cadre de l'amélioration de la gestion des données des alumni de CNA, en vue de faciliter leur accès et leur mise à jour par le personnel concerné. L'objectif principal était de concevoir et de mettre en place une base de données locale, synchronisée avec la plateforme AlumnForce, tout en assurant une documentation technique claire et détaillée pour en garantir l'utilisation et la continuité.

Le projet a été développé en suivant la méthodologie du cycle en V, permettant une validation méthodique des étapes et des fonctionnalités. Les travaux ont inclus la conception d'un modèle relationnel, le développement de scripts pour la création et la population initiale de la base, ainsi que l'extraction et l'insertion des données via l'API de AlumnForce. Les livrables principaux comprennent les scripts de gestion de la base de données, des programmes pour l'insertion et mise à jour des données, et une documentation complète décrivant l'architecture technique, les outils nécessaires et les étapes d'utilisation.

2. INTRODUCTION

2.1. Contexte

Afin de connecter ses diplômés et étudiants, l'École Centrale de Nantes dispose d'une association appelée Centrale Nantes Alumni (CNA). Cette association fédère, rapproche et anime la communauté des centraliens de Nantes afin de promouvoir l'École et ses diplômés, de contribuer au rayonnement de la communauté des centraliens de Nantes et de développer des relations avec le monde économique au bénéfice de ses membres et de Centrale Nantes.

À partir de son site web, les membres ont accès aux profils des étudiants et diplômés, peuvent rejoindre des groupes géographiques, métiers, sportifs ou culturels... créés sur la plateforme (avec des forums de discussions), consulter des actualités et des événements liés à Centrale Nantes notamment (comme la cérémonie de diplomation, des événements, conférences, afterworks, webinaires, visites d'entreprises...), et accéder à une boutique en ligne proposant des articles exclusifs aux couleurs de Centrale Nantes, tels que des T-shirts, pulls et mugs, entre autres.

Ce réseau est fourni par une entreprise tierce : MEVIA. Cette dernière propose la plateforme web AlumnForce avec ses fonctionnalités et est responsable du stockage des données des membres et de leurs informations personnelles, sur leur parcours académique, leur carrière et les groupes rejoints sur le site.

L'extraction des données depuis la plateforme AlumnForce n'est pas un processus direct. Elle nécessite une requête au support technique du prestataire pour obtenir les données, lesquelles sont transmises au format CSV. Par ailleurs, l'insertion des données est également complexe. L'équipe utilise actuellement un dépôt distant géré par le prestataire, où un fichier CSV contenant les informations des étudiants – extraites d'une autre plateforme d'inscription utilisée à Centrale – est transmis pour être intégré chaque jour à la base de données par le tiers entreprise.

Ce processus d'extraction, de mise à jour et d'ajout d'informations dans la base de données présente plusieurs inconvénients. Non seulement il est chronophage, mais il expose également l'équipe à des risques liés à la manipulation directe de données sensibles. Ces risques

incluent des erreurs potentielles lors de la transmission et de l'intégration des informations dans la base de données. Le fait d'interroger l'existant grâce à l'API permet d'éviter d'écraser des cursus déjà suivis antérieurement par certains membres.

2.2. Objectifs

Depuis un an, AlumnForce, le prestataire de stockage en base de données de CNA, a mis en place une API de service qui facilite l'interaction avec la base de données, que ce soit pour importation ou exportation.

D'ailleurs, le projet a pour objectif général la mise en place d'une base de données locale qui exploitera les données depuis l'API de la plateforme AlumnForce, afin de permettre au personnel de CNA d'accéder plus facilement aux données des membres. Cette solution sera accompagnée d'une documentation détaillée expliquant son utilisation, garantissant ainsi une adoption fluide et une continuité dans sa gestion.

Les rendus attendus comprennent un script pour la création de la base de données, un autre pour sa population et sa mise à jour, ainsi qu'une documentation technique détaillée.

La documentation inclura la description de l'architecture de la base de données et du programme développé, ainsi que les logiciels et leurs versions nécessaires pour accéder aux données.

2.3. Portée du logiciel

Le projet a pour objectif principal d'améliorer la gestion des données des alumni en récupérant l'ensemble des informations déjà enregistrées sur la plateforme AlumnForce et en les intégrant dans une base de données locale, spécifiquement conçue pour répondre aux besoins du personnel de CNA. Cette nouvelle base de données sera accompagnée d'une documentation technique détaillée, permettant une utilisation efficace et durable par l'équipe en charge.

Il est important de préciser que le projet se limite à la récupération des données existantes et à leur intégration dans une base de données locale. Bien que l'équipe de développement ait envisagé d'inclure le développement d'un système permettant l'insertion de nouvelles données dans cette base, la complexité du projet s'est avérée plus importante que prévu. Par conséquent,

la création d'un outil de visualisation des données et le développement d'un système pour l'insertion de nouvelles informations ont été exclus du périmètre défini pour ce projet. Ces fonctionnalités supplémentaires pourraient néanmoins être considérées dans le cadre d'une phase ultérieure.

3. MÉTHODOLOGIE

3.1. Méthodologie de développement

La méthodologie adoptée pour ce projet a été le cycle en V. À chaque fonctionnalité ajoutée au code, l'équipe a mis en place des tests afin de s'assurer que les spécifications initialement définies étaient respectées et alignées avec les besoins réels du projet, tout en validant la phase précédente.

Le choix du cycle en V pour ce projet s'explique par la clarté des exigences dès le départ, la nécessité de livrer des résultats fiables et robustes, et les contraintes de temps et de ressources.

Contrairement aux méthodes agiles, qui favorisent l'adaptabilité mais peuvent entraîner des délais et des ajustements fréquents, le cycle en V a permis une planification rigoureuse et une validation méthodique à chaque étape.

Cette approche s'est avérée particulièrement adaptée à une équipe avec une expérience limitée, réduisant les risques liés aux malentendus et assurant une correspondance stricte entre les spécifications initiales et les livrables, comme la base de données, les scripts et la documentation technique.

3.1.1 Tests unitaires

à faire

3.3. Outils et logiciels utilisés

Le projet a utilisé plusieurs outils et technologies pour assurer la gestion efficace de la base de données, le développement du logiciel et la communication entre les membres de l'équipe.

Pour la gestion et la conception de la base de données, le principal logiciel utilisé est pgAdmin 4, associé à PostgreSQL 16.4, qui est une base de données relationnelle open-source robuste et fiable, permettant de gérer et manipuler les données de manière efficace. En complément, l'outil SQL Power Architect 1.0.9 a été utilisé pour la conception de schémas de bases de données, facilitant la création et la gestion des modèles relationnels.

Le langage de programmation choisi pour le développement du projet est Python 3, en raison de sa flexibilité et de ses bibliothèques puissantes, qui facilitent la récupération de données via une API et la manipulation des bases de données. Les principales bibliothèques utilisées sont :

- asyncio et aiohttp : pour la gestion des requêtes asynchrones à l'API.
- psycopg2 : pour interagir avec la base de données PostgreSQL.
- unicodecode : pour normaliser les chaînes de caractères.
- json : pour manipuler les données en format JSON.
- Visual Studio Code a été utilisé comme l'environnement de développement intégré (IDE).

Pour le suivi du code et la gestion des versions, Git a été utilisé en combinaison avec GitHub. Le lien du dépôt GitHub était partagé entre les parties prenantes du projet, qui pourraient suivre le projet à distance.

Concernant la gestion de projet et communication interne, Google Drive a été utilisé pour le stockage et le partage de fichiers, notamment des documents Word et des rapports. Pour le suivi des tâches et l'organisation du travail de l'équipe, l'outil Trello était utilisé. Il s'agit d'une application web de gestion de projet visuel basée sur le principe du Kanban, permettant de suivre l'avancement des tâches et d'assigner des responsabilités.

La communication entre les membres de l'équipe s'est faite principalement par email et parfois via l'application Slack, un outil de messagerie instantanée.

3.4. Planification

Le projet était initialement divisé en trois phases. Au fur et à mesure que le projet avançait, l'équipe a ressenti le besoin de diviser la phase 2 en deux pour pouvoir atteindre un

objectif faisable et concret pour la finalisation du projet. C'est pourquoi les phases qui seront détaillées ci-dessous diffèrent légèrement des phases prévues sur le cahier des charges.

Phase 1 - Initial - Familiarisation - Découverte de l'API

Dans un premier temps, l'objectif principal est de se familiariser avec l'API d'AlumnForce, de connaître la syntaxe pour réaliser des extractions et de réussir à visualiser sur le terminal les champs de données envisagées. Le livrable de cette phase est un script dans la langue de programmation choisie qui se connecte à l'API et sort les informations de quelques alumni dans le terminal. Toutes les informations pertinentes pour l'utilisation de l'API seront ajoutées à la documentation du projet.

Phase 2 - Modélisation et création de la base de données

Dans la deuxième phase, l'objectif est de concevoir, de modéliser et de créer la base de données qui récoltera les informations sorties d'AlumnForce. Pour cela, les livrables de cette phase sont le modèle conceptuel de données et le modèle physique de données, ainsi qu'un script pour la création des tableaux. En plus, cette phase comportera une documentation très importante pour la continuation du projet, responsable pour expliquer le fonctionnement de la base de données, sa conception et comment la remplir.

Phase 3 - Final - Insertion des données et mise à jour de la base de données

Dans la troisième et dernière phase, l'objectif est de remplir la base de données locale avec les informations récupérées via l'API d'AlumnForce et de mettre en place un mécanisme de mise à jour automatique de ces données.

Le livrable de cette phase est un programme développé en Python, conçu pour effectuer l'extraction et la mise à jour des données de manière efficace. Ce programme garantit que la base locale reste continuellement synchronisée avec la base de données d'AlumnForce, offrant ainsi une solution fiable et automatisée pour maintenir l'intégrité des informations.

Phase 4 - Application pour faciliter l'utilisateur à accéder aux infos de l'API

Dans le cas idéal, la finalisation de ce projet sera accompagnée de la création d'une application qui donne facilement aux infos de AlumnForce via API sans utilisation des scripts par le personnel de CNA. Cette application aurait une documentation claire et extensive et

permettrait la continuation du projet vers la création d'une base de données partagé entre les différents secteurs de l'école, et pas seulement locale dans l'ordinateur de qui souhaite l'utiliser.

Cependant, comme discuté avec les encadrants du projet, il est improbable que cette phase arrive à terme faute de temps pour le développement du projet. Donc, cette phase sera classifiée comme en 'stand-by' pour qu'un groupe prochain puisse prendre le projet en main et la continuer.

Phase 1: Découverte de l'API	Deadline
Réunion de suivi	03 Octobre
Prise en main de l'API	03 Octobre
Réalisation des tests avec l'API	18 Octobre
Réunion de suivi	17 Octobre
Livrable : Cahier des charges	11 Octobre
Livrable : Script pour accéder les données	18 Octobre

Phase 2 - Modélisation et création de la base de données	Deadline
Tester les manières d'extraire une quantité importante de données via l'API	20 Octobre
Conceptionner une base de données pour garder les informations	20 Octobre
Livrable : Diagrammes UML / MCD / MPD / documents de conception	25 Octobre
Livrable : Script pour la création de la base de données	25 Octobre

Phase 3 - Insertion des données et mise à jour de la base de données	Deadline
Insérer et/ou mettre à jour les données sur les tables qui sont indépendantes	29 Novembre
Insérer et/ou mettre à jour les données sur les tables qui dépendent fortement des autres	20 Décembre
Livrable : Squelette de rapport final	06 Décembre

Livrable : Script pour l’insertion des données sur la base de données	06 Décembre
--	--------------------

Phase 4 - Application pour faciliter l’usager à accéder aux infos de l’API	Deadline
Étude - possibilité de faire une interface visuel	20 Décembre
Livrable: Rapport final	20 Décembre
Soutenance	13 Janvier

4. RÉSULTATS

4.1. Rendus Finaux

Todos os elementos da entrega final aos encadrantes do projeto estão disponíveis no repositório github do projeto, cujo o link se encontra nos anexos do relatório.

4.1.1 Base de données

Uma das entregas do projeto é um script em Postgresq para a criação da base de dados. Seu modelo físico mudou muito desde a sua primeira concepção a medida que a dupla foi

populando mais tabelas e se familiarizando mais com os dados da API.

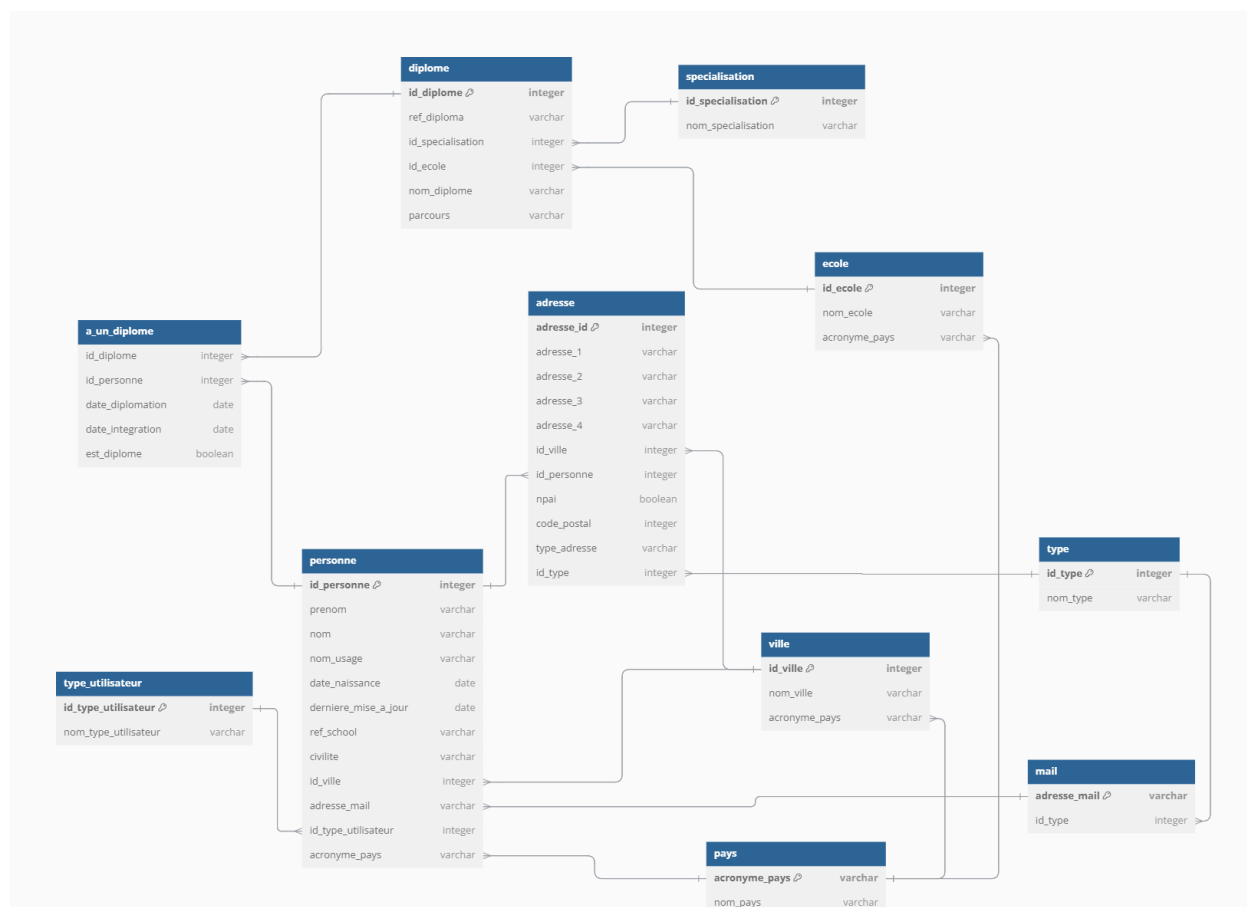


Figure 1 - Modèle Physique des Données

A modelização de dados escolhida visa assegurar a inserção das informações correta na base de dados e tornar facilmente identificável os valores discretizados possíveis para os dados, como tipo de utilizador, tipo de endereço/mail... Com isso, anomalias tornam-se mais facilmente identificáveis e pode-se proibir a inserção de dados inválidos na base.

É importante salientar que todas as tabelas da base (com exceção da tabela pays) são populadas diretamente com os dados provindos da API. Caso um dia a API disponha dados diferentes ou novos, as tabelas serão atualizadas e nenhum script adicional necessitará ser executado. A exceção para essa regra é a tabela pays, que é populada no arquivo de criação da base de dados. Essa tabela é assumida como fixa e não recebe atualizações posteriormente, e esse problema será melhor explorado nas próximas sessões desse relatório

Sur le plan technique, les scripts permettant la création de la base de données et la population initiale des tables moins dépendantes ont été finalisés et sont disponibles sur le dépôt GitHub partagé avec les encadrants. Ces livrables témoignent d'une progression constante dans le projet.

4.1.2 Scripts pour peupler la base de données

No github da equipe são disponibilizados os scripts capazes de se conectar com a API Alumnforce, recuperar as informações necessárias para a inserção na base de dados, tratar esses dados e inserí-los na base de dados.

4.1.3 Documentação

Readme.md completo para explicar como rodar os scripts e os requerimentos no github da equipe. tutorial detalhado para a criação e população de um banco de dados local na máquina do usuário

4.1. Demonstração de funcionamento

Readme completo para explicar como é populada a base de dados

4.2. Adh rence au planning initial

Au cours du d veloppement du projet, il est rapidement apparu que le plan initial  tait ambitieux pour le d lai imparti. La complexit  li e   l'extraction et au traitement des donn es s'est av r e plus grande que pr vu par les d veloppeurs. Les donn es pr sentes dans AlumnForce sont r cup r es au format JSON avec des structures de crochets imbriqu s, rendant impossible l'extraction de toutes les informations n cessaires avec une seule requ te.

Face   cette situation, l' quipe a d cid  d'ajouter une  tape suppl mentaire au projet : la population progressive de la base de donn es, accompagn e d'une v rification pr alable pour s'assurer que ces donn es n'ont pas d j   t  ins r es. Cela permet de simplifier les op rations futures en cr ant un programme capable d'ins rer et de mettre   jour les donn es en une seule ex cution, optimisant ainsi le processus.

Cette révision du plan initial a été communiquée aux encadrants, avec une explication des ajustements nécessaires. Nous avons notamment confirmé qu'il ne serait pas possible, dans le temps imparti, de livrer une interface de visualisation des données avec la qualité attendue. Nous avons choisi de prioriser la livraison d'une base de données fiable et robuste, constituant une base solide pour d'éventuels développements futurs. Cette décision vise à permettre à un prochain groupe de poursuivre le projet dans de bonnes conditions, en disposant d'une structure claire et de données correctement organisées.

En somme, ces ajustements témoignent de notre volonté d'assurer la qualité et la continuation du projet, en faisant des choix stratégiques alignés sur les contraintes rencontrées.

4.3. Estimation des coûts

À faire

5. DIFFICULTÉS RENCONTRÉS

Au cours du projet, plusieurs difficultés ont été rencontrées, nécessitant des ajustements pour garantir l'avancement et la qualité des livrables.

La première difficulté majeure concernait la définition du modèle physique de données. Bien que le modèle initial ait semblé adéquat, des problèmes sont apparus lors de l'insertion des premières données. Par exemple, il était attendu que chaque spécialisation dispose d'un identifiant propre dans la base de données de AlumnForce. Cependant, cette information n'étant pas disponible, il a fallu adapter le modèle local pour générer des identifiants séquentiels spécifiques à chaque spécialisation. Pour résoudre ce problème, l'équipe a consacré du temps à corriger ces incohérences, en révisant le modèle physique et en s'ajustant aux besoins réels.

Une autre difficulté notable résidait dans le manque d'expérience des développeurs. Malgré une bonne communication, des différences dans les approches logiques de programmation ont parfois conduit à des incohérences dans le code. Cela a probablement ralenti l'avancement du projet par rapport aux prévisions initiales. Pour remédier à cette situation, l'équipe a adopté une série de mesures correctives : l'utilisation de Trello pour le suivi des activités et une meilleure répartition des tâches, des plages horaires distinctes pour éviter les

conflits lors du travail sur le code, ainsi que l'envoi hebdomadaire d'audios détaillés. Ces derniers expliquaient les avancées réalisées, les fonctionnalités développées et les prochaines étapes à suivre, renforçant ainsi la compréhension mutuelle et la coordination.

Bien que ces problèmes aient posé des défis, ils ont également été des opportunités d'apprentissage. En les analysant régulièrement et en mettant en place des actions correctives, l'équipe a pu transformer ces obstacles en éléments constructifs pour le développement du projet et pour éviter leur répétition à l'avenir.

6. SUITE DU PROJET

Pour la suite du projet jusqu'à sa finalisation, l'une des priorités immédiates est de finaliser l'insertion des données dans la base de données locale, en veillant à ce que toutes les informations soient correctement intégrées et accessibles. Parallèlement, une documentation détaillée est en cours de rédaction. Cette dernière inclura non seulement une explication claire de l'architecture et des fonctionnalités du projet, mais également un tutoriel pratique sur l'utilisation des codes disponibles sur le dépôt GitHub. Enfin, l'équipe travaille à adapter le code pour qu'il puisse gérer efficacement d'éventuelles erreurs lors de l'insertion des données, assurant ainsi une robustesse de la solution.

Au-delà de ces étapes finales, le projet ouvre la voie à des améliorations futures qui pourront être guidées par les besoins évolutifs des utilisateurs. Une première amélioration envisagée serait d'automatiser l'exécution du script de mise à jour de la base de données locale à chaque ajout de nouvelles données dans la base AlumnForce. Cela permettrait une synchronisation continue entre les bases locale et distante, réduisant ainsi l'intervention manuelle et les risques d'incohérence.

Une autre perspective de développement serait la création d'une interface de visualisation des données contenues dans la base locale. Cette fonctionnalité offrirait une vue d'ensemble intuitive et exploitable des informations pour le personnel de CNA.

Enfin, une extension ambitieuse consisterait à développer une interface facilitant l'insertion de nouvelles données dans la base AlumnForce via son API. Cela pourrait inclure la

conception d'un script capable de transformer les données des étudiants d'un format CSV en JSON, suivi de l'envoi automatisé d'une requête POST à l'API pour intégrer ces informations.

7. CONCLUSION

Ce projet permet de répondre à l'objectif principal de faciliter la gestion des données des Alumni de Centrale Nantes en mettant en place une base de données locale à partir des données extraites de la plateforme AlumnForce. Malgré les défis rencontrés, tels que la complexité imprévue de l'extraction et du traitement des données, l'équipe a su s'adapter en ajustant le modèle initial et en optimisant les processus de développement.

Désormais, le projet approche de sa phase finale. Les prochaines étapes consisteront à compléter l'insertion des données dans les tables restantes et à finaliser l'organisation de la documentation technique. Cette dernière permettra à l'association d'utiliser efficacement la solution développée et de garantir une éventuelle continuité du projet dans le futur.

Nous exprimons notre gratitude à Mme Christelle Rousseau et à M. Stéphane Buret pour leurs consignes, leurs orientations précieuses tout au long du projet, ainsi que pour leur disponibilité et leur volonté d'apporter leur aide. Leurs contributions ont été essentielles à la progression et à la qualité des résultats obtenus.

8. ANNEXES

8.1. Dépôt Github du Projet

GitHub du Projet: <https://github.com/larianog/projet-application/>

- Ce répertoire contient les codes sources, un tutoriel de comment déployer le projet, les spécifications fonctionnelles du projet et les rapports de suivi.

9. RÉFÉRENCES

- (1) [Centrale Nantes Alumni: C'est quoi un Alumni?](#)
- (2) [Page web Centrale Nantes Alumni](#)
- (3) [Documentation de l'API de AlumnForce](#)