

RAPPORT DE STAGE DE FIN DE TROISIÈME ANNÉE

Développement d'une interface web d'administration qui permet d'industrialiser les processus d'audit d'une partie d'un SI collaboratif Microsoft 365



Préparé par : AMIAR Abd rezak

Tuteurs :

- M. Anas BOURHLEM
- M. Alexis BES

*Filière Informatique
Année universitaire : 2023/2024*

Remerciements

Avant d'aborder la description des parties importantes du projet, je tiens tout d'abord à exprimer ma profonde gratitude à toutes les personnes qui ont contribué de manière significative à l'élaboration et à la réalisation de ce travail. Je commence par remercier chaleureusement l'ensemble des personnes travaillant à l'Université Paris Est Créteil (UPEC). Ces hommes et femmes dévoués ne cessent de nous préparer, moi et mes collègues, à des expériences enrichissantes afin de nous intégrer efficacement dans le marché du travail réel.

*Toute ma reconnaissance va également à mes encadrants de stage : M. **BOURHLEM** Anas, M. **ZORGATI** Mauslem, M. **DAOUDI** Yassir, M. **KRUZIK** Marc, ainsi qu'à toutes les personnes qui ont contribué à la réalisation de ce projet. Je vous remercie sincèrement de m'avoir assuré un encadrement de qualité, permettant de mener à bien ce travail dans les meilleures conditions.*

Résumé

Ce document représente une étude synthétique du travail réalisé dans le cadre de mon stage de troisième année. J'ai effectué mon stage au sein de l'université Paris Est Créteil (**UPEC**) et de l'entreprise Exakis Nelite (**EXN**), collaboratrice de Microsoft. Ce stage a duré trois mois, de fin avril à fin juillet 2024. L'objectif principal de ce projet était de développer un outil ainsi qu'une interface web pour la gestion, la configuration et l'audit des environnements Microsoft et Azure cloud, souvent appelés "tenant" ensemble. La réalisation du projet s'est déroulée en trois étapes principales :

- Analyse détaillée des besoins ;
- Conception des différentes composantes de l'application ;
- Implémentation des objectifs définis.

J'ai donc eu l'opportunité d'exploiter une variété de technologies, outils et concepts qui ont grandement contribué à la réalisation de cette solution. Parmi ces ressources, plusieurs proviennent des cours que j'ai suivis au cours de l'année précédente, consolidant ainsi mes connaissances théoriques. En outre, j'ai découvert et appris à maîtriser de nouveaux outils durant ma formation au sein de l'entreprise, ce qui m'a permis d'acquérir des compétences pratiques supplémentaires. Vous trouverez un récapitulatif détaillé de ces technologies et outils utilisés dans les annexes de ce document.

Le présent rapport est structuré de manière à offrir une vue d'ensemble claire et détaillée de l'ensemble du projet. Le chapitre I est consacré à la présentation de l'organisme d'accueil, y compris l'introduction à l'équipe et l'entreprise, ainsi que la définition des objectifs du stage. Le chapitre II se concentre sur le cadrage fonctionnel et la présentation du projet, détaillant le benchmark des solutions disponibles, et l'étude conceptuelle des besoins et des normes de sécurité. Le chapitre III aborde le cadrage technique, où sont élaborées les différentes composantes techniques de la solution envisagée, incluant l'analyse de l'existant, les premières utilisations et la personnalisation de la solution MONKEY365. Enfin, le chapitre IV se consacre à la partie réalisation, où je vais aborder mes réalisations concrètes pendant le stage, telles que l'obtention de certifications, l'introduction au DevOps, la construction d'applications avec Azure DevOps, la participation à un événement Microsoft sur les nouvelles technologies IA, et le développement et déploiement d'un site web servant de portfolio personnel, mettant en lumière les contributions spécifiques que j'ai apportées au projet.

Mots clés :

Gestion, Configuration, Audit, Microsoft, Azure cloud, Tenant, Conception, Annexe, Technologie.

Abstract

This document represents a synthetic study of the work carried out during my third-year internship. I completed my internship at the University of Paris Est Créteil (**UPEC**) and the company Exakis Nelite (**EXN**), a collaborator with Microsoft. This internship lasted three months, from late April to late July 2024. The main objective of this project was to develop a tool and a web interface for the management, configuration, and audit of Microsoft and Azure cloud environments, often collectively referred to as "tenant." The project was carried out in three main stages :

- Detailed needs analysis;
- Design of the various components of the application;
- Implementation of the defined objectives.

I had the opportunity to utilize a variety of technologies, tools, and concepts that greatly contributed to the realization of this solution. Among these resources, several were drawn from the courses I took over the previous year, thus consolidating my theoretical knowledge. Additionally, I discovered and learned to master new tools during my training within the company, which allowed me to acquire additional practical skills. You will find a detailed summary of these technologies and tools used in the appendices of this document.

This report is structured to provide a clear and detailed overview of the entire project. Chapter I is dedicated to the presentation of the host organization, including the introduction to the team and the company, as well as the definition of the internship objectives. Chapter II focuses on the functional framework and project presentation, detailing the benchmark of available solutions, and the conceptual study of needs and security standards. Chapter III addresses the technical framework, where the various technical components of the proposed solution are developed, including the analysis of the existing setup, initial uses, and customization of the MONKEY365 solution. Finally, Chapter IV is devoted to the realization part, where I will discuss my concrete achievements during the internship, such as obtaining certifications, introduction to DevOps, building applications with Azure DevOps, participating in a Microsoft event on new AI technologies, and the development and deployment of a website serving as a personal portfolio, highlighting the specific contributions I made to the project.

Keywords :

Management, Configuration, Audit, Microsoft, Azure cloud, Tenant, Design, Appendix, Technology.

Table des matières

Remerciements	i
Résumé	ii
Abstract	iii
Introduction générale	vii
Liste des abréviations	viii
Terminologie	ix
1 Présentation des organismes	1
1.1 Présentation de l’UPEC	1
1.2 Présentation d’ Exakis Nelite	2
1.2.1 Présentation de l’entreprise	2
1.2.2 Structure organisationnelle	3
1.2.3 Présentation de l’équipe	4
1.3 Définition des objectifs du stage	4
1.3.1 Description des objectifs globaux du stage	4
1.3.2 Importance et impact attendu sur l’entreprise	4
1.3.3 Compétences à développer ou acquérir	5
1.3.4 Livrables et indicateurs de réussite	5
1.4 Récapitulatif	6
2 Cadrage Fonctionnel	7
2.1 Périmètre du projet	7
2.1.1 Contexte et justification	7
2.1.2 Objectifs principaux	8
2.1.3 Étapes clés	9
2.1.4 Méthodologie de travail	10
2.2 Benchmark des solutions disponibles	11
2.2.1 Définition de Benchmark	11
2.2.2 Exemples de notre cas	11
2.2.3 Résultat du Benchmark	12
2.3 Étude Conceptuelle	13
2.3.1 Nos besoins	13
2.3.2 Référentiels des normes de sécurité (Security Framework)	14
2.4 Récapitulatif	14
3 Cadrage Technique	15
3.1 Analyse de l’existant	15
3.1.1 Examen du code de la librairie tierce Monkey365	15
3.1.2 Développement du Proof of Concept (POC)	16
3.1.3 Évaluation du Proof of Concept	16
3.1.4 Problèmes identifiés	17
3.1.5 Solutions apportées	18
3.1.6 Vérifications de sécurité	19

3.2	Personnalisation de l'outil Monkey365	20
3.2.1	Adaptation du framework de sécurité	20
3.2.2	Modification de l'interface utilisateur	20
3.2.3	Conception de l'architecture de l'application	22
3.3	Gestion du dépôt du projet	23
3.4	Récapitulatif	23
4	Formations et Apprentissages	24
4.1	Certifications	24
4.1.1	Microsoft Azure Fundamentals AZ-900	24
4.1.2	Microsoft365 Fundamentals MS-900	25
4.1.3	Microsoft Azure Developer Associate AZ-204	26
4.2	Formation en DevOps	27
4.2.1	Qu'est-ce que le DevOps	27
4.2.2	Construction d'application avec Azure DevOps	27
4.2.3	Déploiement d'application avec Azure DevOps	27
4.3	Participation à un événement Microsoft sur les nouvelles technologies IA	27
4.4	Formation sur les LLM (Large Language Models) chez Microsoft	28
4.4.1	Azure AI Studio	28
4.4.2	Prompt engineering	28
4.4.3	Fine tuning	28
4.5	Portfolio Website	29
4.6	Résumé des apprentissages	30
	Conclusion et perspectives	32

Table des figures

1.1	UPEC	1
1.2	Groupe Magellan Partners	2
1.3	Exakis Nelite (EXN)	2
1.4	Structure Organisationnelle EXN	3
2.1	Périmètre des services Microsoft à auditer	8
2.2	Processus d’audit	9
2.3	Tableau Kanban sur Microsoft Planner	10
2.4	Logo Microsoft365DSC	11
2.5	Logo Monkey365	11
2.6	Logo Maester	12
3.1	Erreur importation module	17
3.2	Erreur droits Sharepoint	17
3.3	Erreur souscription Azure	18
3.4	Méthodes d’authentification prises en charge par service	19
3.5	Interface Monkey365 avant changement	21
3.6	Interface Monkey365 apres changement (Audit365)	21
3.7	Architecture de déploiement proposée pour l’application	22
4.1	Certification AZ-900	25
4.2	Certification MS-900	26
4.3	Site Vitrine de Amiar Abdrezak	30

Introduction générale

Dans le cadre d'un processus de modernisation de la gestion des services informatiques, et plus spécifiquement pour assurer la conformité et l'optimisation des environnements Microsoft, j'ai eu l'opportunité de développer un outil d'audit des tenants Microsoft au sein de **Exakis Nelite**. Ce rapport de stage présente les différentes étapes de cette mission, depuis la conception jusqu'à la réalisation de l'outil.

En effet, avec l'évolution rapide des technologies et l'importance croissante de la sécurité informatique, il est devenu impératif pour les entreprises de s'assurer que leurs systèmes sont non seulement efficaces mais aussi sécurisés. L'objectif fondamental de ce projet est de fournir un outil qui permet de vérifier et d'optimiser les configurations des tenants Microsoft, en se basant sur les standards et les meilleures pratiques en vigueur.

Actuellement, les entreprises utilisent divers outils pour gérer leurs environnements Microsoft, mais un besoin crucial se fait sentir pour un audit automatisé, systématique et efficace qui peut être effectué en temps réel. Les indicateurs fournis par les différents systèmes doivent être vérifiés et comparés aux standards en vigueur pour garantir une gestion optimale. C'est ainsi que le développement de cet outil d'audit constitue une nécessité pertinente et opportune pour améliorer le suivi et l'optimisation des environnements Microsoft.

Ce rapport est structuré de la manière suivante : le chapitre I est dédié à la présentation de l'organisme d'accueil. Le chapitre II traite du cadrage fonctionnel. Le chapitre III est consacré au cadrage technique et enfin le chapitre IV présente les réalisations.

Liste des abréviations

- **UPEC** : Université Paris Est Créteil.
- **EXN** : Exakis Nelite.
- **IA** : Intelligence Artificielle.
- **IoT** : Internet of Things.
- **SLM** : Service Line Manager.
- **SDM** : Service Delivery Manager.
- **SI** : Systeme d'Information.
- **M365** : Microsoft365.
- **CIS** : Center for Internet Security.
- **SF** : Security Framework/Baseline.
- **POC** : Proof Of Concept.
- **CI** : Continuous Integration.
- **CD** : Continuous Deployment.
- **LLM** : Large Language Models.

Terminologie

- **Tenant** : Un tenant est une instance dédiée d'un service cloud utilisé par une organisation. Il permet de séparer les données et les configurations d'une organisation de celles des autres utilisateurs du service.
- **Azure** : Azure est la plateforme de cloud computing de Microsoft. Elle offre une variété de services cloud, y compris le calcul, l'analytics, le stockage et le réseautage. Azure permet aux entreprises de développer, tester, déployer et gérer des applications et des services à travers des centres de données gérés par Microsoft.
- **Microsoft Entra ID** : Anciennement connu sous le nom d'Azure Active Directory (Azure AD), Microsoft Entra ID est un service de gestion des identités et des accès basé sur le cloud. Il permet aux organisations de gérer les identités et de contrôler l'accès aux ressources.
- **Audit** : Un audit est un examen systématique et indépendant des configurations, des pratiques et des opérations d'un système pour s'assurer qu'elles sont conformes aux politiques, aux procédures et aux exigences réglementaires.
- **CIS** : Le Center for Internet Security (CIS) est une organisation à but non lucratif qui développe des normes et des meilleures pratiques pour sécuriser les systèmes et les données. Les benchmarks CIS sont des configurations de sécurité recommandées pour les systèmes informatiques.
- **Prompt Flow** : Dans le contexte de l'IA et du traitement du langage naturel, le prompt flow se réfère à la manière dont les invites (prompts) sont formulées et enchaînées pour obtenir des réponses cohérentes et pertinentes d'un modèle de langage.

Chapitre 1

Présentation des organismes

Le chapitre I de ce rapport est consacré à la présentation de l'organisme d'accueil, une étape cruciale pour comprendre le cadre dans lequel le stage s'est déroulé. Cette section commence par une introduction à l'université puis à l'entreprise, détaillant l'historique, la mission, ainsi que les produits et services principaux de l'entreprise. Elle continue par une description de l'équipe, incluant la composition et les rôles des membres, ainsi que les objectifs et missions de l'équipe. Enfin, la structure organisationnelle est examinée, mettant en lumière la position de l'équipe au sein de l'entreprise. Le chapitre se conclut par la définition des objectifs du stage, détaillant les objectifs globaux, l'importance et l'impact attendu sur l'entreprise, les compétences à développer, ainsi que les livrables et les indicateurs de réussite.

1.1 Présentation de l'UPEC

L'Université Paris-Est Créteil ou **UPEC** est une université publique fondée en 1970. Située en Île-de-France, elle est l'une des principales institutions académiques de la région. L'**UPEC** est réputée pour son engagement à fournir une éducation de qualité et à promouvoir la recherche scientifique. Elle offre une variété de formations en sciences humaines et sociales, médecine, sciences économiques et gestion, sciences et technologies, ainsi qu'en droit et sciences politiques. Je suis actuellement en troisième année de licence informatique à l'**UPEC**, où j'acquiers des compétences en programmation, développement de logiciels et gestion de systèmes informatiques.



FIGURE 1.1 – UPEC

1.2 Présentation d'Exakis Nelite

1.2.1 Présentation de l'entreprise

Exakis Nelite, une entreprise de conseil en transformation digitale fondée en 2001, est devenue une entité clé du groupe Magellan Partners depuis 2008. Sa mission est d'accompagner les entreprises dans l'adoption et l'intégration des solutions Microsoft, offrant des services de conseil, d'intégration, de développement et de support. Exakis Nelite s'engage à fournir des solutions innovantes et adaptées, participant activement à la transformation numérique de ses clients grâce à une expertise technique de haut niveau. L'entreprise est spécialisée dans plusieurs domaines clés de la transformation digitale portés par les équipes :

- **Modern Work** : Elle déploie des solutions collaboratives et de productivité comme Microsoft365, met en place des intranets et des applications métiers, et accompagne au changement pour maximiser l'adoption des outils Microsoft.
- **Data & Intelligence Artificielle** : Exakis Nelite propose des solutions de Business Intelligence, d'analytique avancée, et d'Internet des Objets (**IoT**) pour exploiter les données et favoriser des décisions informées.
- **Business Applications** : Les applications développées par l'entreprise optimisent les processus métiers et automatisent les workflows.
- **Application & Infrastructure** : Elle aide à la migration vers le cloud Azure, modernise les applications existantes, et met en place des infrastructures sécurisées et hybrides.
- **Cybersécurité** : Les services incluent la protection des données sensibles, la gestion des identités et des accès, et la défense contre les menaces.

Grâce à son expertise et à une gamme complète de certifications Microsoft, Exakis Nelite se positionne comme le premier pure player Microsoft en France, offrant des solutions complètes pour relever les défis technologiques et réussir la transformation digitale de ses clients.

J'ai eu l'occasion de collaborer directement avec l'équipe de la société **EXN** basée à Paris, une des nombreuses agences de l'entreprise réparties en Europe, notamment en France, en Bulgarie et au Maroc. Cette présence internationale permet à Exakis Nelite d'offrir une expertise locale tout en bénéficiant de ressources globales.



FIGURE 1.2 – Groupe Magellan Partners



FIGURE 1.3 – Exakis Nelite (EXN)

1.2.2 Structure organisationnelle

L'équipe Modern Workplace, dirigée par Anas Bourhlem, Service Line Manager (**SLM**), occupe une position stratégique au sein d'Exakis Nelite. Anas Bourhlem reporte directement à David Deragon, Service Delivery Manager (**SDM**), qui lui-même est sous la direction de Steven Le Morzadec, actuel Directeur de Région. Au sommet de la hiérarchie, Didier Zeitoun, Président de Magellan Partners, supervise les opérations globales du groupe, assurant une direction cohérente et une vision stratégique pour toutes les entités, y compris Exakis Nelite.

Cette structure hiérarchique permet une gestion fluide et efficace des projets et des services offerts par l'équipe Modern Workplace. En plus de Mauslem Zorgati, consultant, Yassir Daoudi, chef de projet, et moi Abdrezak Amiar. Voici une figure qui présente cette structure :

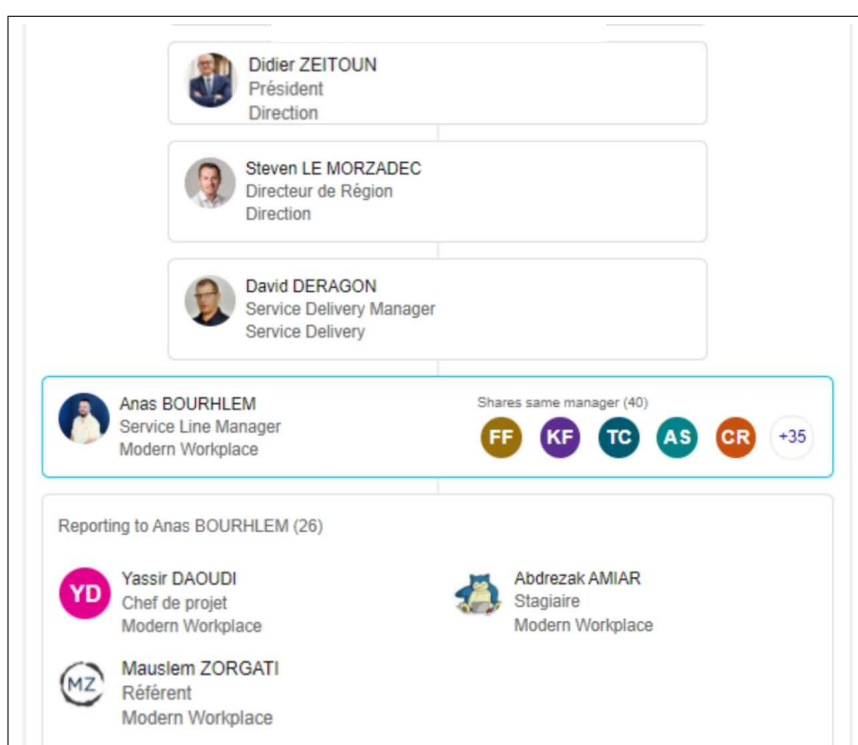


FIGURE 1.4 – Structure Organisationnelle EXN

L'équipe comprend d'autres membres qui ne sont pas sous la supervision directe d'Anas Bourhlem. Dans le prochain paragraphe, je présenterai ces membres supplémentaires et expliquerai le rôle de chacun au sein de l'équipe.

1.2.3 Présentation de l'équipe

L'équipe Modern Workplace se compose de divers membres ayant des rôles spécifiques et complémentaires pour assurer la réussite de nos projets. Voici une présentation de notre équipe :

- **Steven Le Morzadec** : Directeur de Région, il est la voix décisive pour guider le projet et assure la supervision globale des opérations.
- **Anas Bourhlem** : Service Line Manager (Modern Workplace), il est l'un des spécialistes métier et joue un rôle clé dans la direction et la coordination des activités de l'équipe. Il est aussi mon tuteur de stage.
- **Yassir Daoudi** : Chef de projet, il est le chef d'orchestre assurant la coordination de toutes les étapes du projet et garantissant que les objectifs sont atteints de manière efficace.
- **Marc Kruzik** : Consultant Apps & Infra & Data, il aide au niveau technique et assure un rôle de coach pour le développement des compétences techniques.
- **Mauslem Zorgati** : Consultant SharePoint, il apporte son expertise pour les configurations SharePoint spécifiques à notre solution et plus généralement sur Microsoft 365.
- **Pisda Phcar** : Graphiste Designer, il apporte sa créativité et ses compétences en design graphique pour améliorer l'ergonomie de notre projet.
- **Abdrezak Amiar (Auteur)** : Développeur, il est l'artisan du code, responsable de la programmation et du développement des solutions techniques.

1.3 Définition des objectifs du stage

1.3.1 Description des objectifs globaux du stage

Le stage a pour objectif principal de développer une interface d'administration pour les scripts d'audits et de gouvernance des systèmes d'information (SI) Microsoft365. Cette interface vise à industrialiser les processus d'audit de tout ou partie d'un SI collaboratif en utilisant des technologies telles que Microsoft .NET, Azure, M365 et PowerShell.

1.3.2 Importance et impact attendu sur l'entreprise

L'importance de ce projet pour l'entreprise réside dans l'amélioration de l'efficacité et de la précision des audits SI. L'interface permettra de centraliser et d'automatiser les audits, réduisant ainsi les erreurs manuelles et augmentant la rapidité des processus. Cela aidera Exakis Nelite à offrir des services de haute qualité à ses clients, renforçant ainsi sa position sur le marché en tant qu'expert en transformation digitale et gouvernance SI.

1.3.3 Compétences à développer ou acquérir

Au cours de ce stage, plusieurs compétences clés seront développées ou renforcées :

- **Techniques de développement** : Maîtrise de l'environnement Microsoft et Azure pour le Modern Workplace, utilisation de .NET C# et PowerShell pour le développement d'applications, et expertise en Azure DevOps et pipelines CI/CD pour le déploiement.
- **Gestion de projets** : Compréhension et application des méthodologies Agile pour planifier et exécuter des projets.
- **Compétences fonctionnelles** : Analyse des besoins utilisateurs, rédaction des User Stories et conception de solutions adaptées.
- **Sécurité informatique** : Connaissance des bonnes pratiques de sécurité et de conformité pour les systèmes Microsoft 365 et Cloud Azure.

1.3.4 Livrables et indicateurs de réussite

Les livrables attendus du stage incluent :

- **Interface d'administration** : Une application web fonctionnelle pour la gestion des scripts d'audit.
- **Documentation technique** : Guides et manuels pour l'utilisation et la maintenance de l'interface.
- **Rapports d'audit** : Modèles de rapports automatisés générés par l'interface.

Les indicateurs de réussite pour l'entreprise et pour moi sont :

- **Pour l'entreprise** :
 - Réduction du temps nécessaire pour réaliser des audits SI.
 - Augmentation de la précision des audits grâce à l'automatisation.
 - Satisfaction des clients grâce à une meilleure qualité des rapports d'audit.
- **Pour moi** :
 - Acquisition de compétences techniques avancées en développement et sécurité.
 - Compréhension approfondie des processus de gouvernance SI.
 - Réussite dans la gestion et la livraison d'un projet complet en suivant les méthodologies Agile.

1.4 Récapitulatif

Dans ce chapitre introductif, j'ai présenté l'organisme d'accueil, en commençant par une introduction à l'université, puis à l'entreprise et à l'équipe. J'ai détaillé l'historique, la mission, ainsi que les produits et services principaux de l'entreprise. Ensuite, j'ai décrit la composition et les rôles des membres de l'équipe, ainsi que leurs objectifs et missions. J'ai également examiné la structure organisationnelle pour mettre en lumière la position de l'équipe au sein de l'entreprise. Enfin, j'ai défini les objectifs du stage, en précisant les objectifs globaux, l'importance et l'impact attendu sur l'entreprise, les compétences à développer, ainsi que les livrables et les indicateurs de réussite.

Le chapitre suivant abordera le cadrage fonctionnel du projet.

Chapitre 2

Cadrage Fonctionnel

Ce chapitre représente le point de départ de mon travail. Dans ce chapitre, j'analyserai et spécifierai les besoins du projet, puis j'identifierai les différents acteurs impliqués. Ensuite, je procéderai à un benchmark des solutions disponibles et à une étude conceptuelle pour définir les besoins et les référentiels de sécurité nécessaires.

2.1 Périmètre du projet

2.1.1 Contexte et justification

Le projet **Audit365 Manager** trouve son origine dans le besoin d'Exakis Nelite d'optimiser et de moderniser ses processus d'audit et de gouvernance des **SI Microsoft 365 (M365)**. Actuellement, les audits sont réalisés de manière manuelle, ce qui est non seulement chronophage mais aussi sujet aux erreurs humaines. La problématique principale est donc de développer une solution qui puisse automatiser et industrialiser ces processus d'audit pour garantir une meilleure efficacité et fiabilité.

Audit et gouvernance

L'audit et la gouvernance sont des concepts clés pour assurer la sécurité et le bon fonctionnement des systèmes d'information.

Audit : L'audit consiste à examiner attentivement les systèmes pour vérifier qu'ils respectent les règles et les normes de sécurité. Cela inclut l'inspection des paramètres, la détection des problèmes potentiels et la génération de rapports avec des recommandations pour améliorer la sécurité.

Gouvernance : La gouvernance implique l'établissement de règles et de processus pour gérer les systèmes de manière efficace et sécurisée. Elle assure que les systèmes sont utilisés de manière appropriée et conforme aux objectifs de l'organisation et aux réglementations.

En résumé, l'audit vérifie que tout fonctionne correctement et en toute sécurité, tandis que la gouvernance définit comment les systèmes doivent être gérés pour rester sûrs et efficaces.

2.1.2 Objectifs principaux

- **Développer une interface d'administration** pour les scripts d'audit, facilitant ainsi leur gestion et leur exécution.
- **Industrialiser les processus d'audit** d'une partie ou de la totalité d'un SI collaboratif **M365**, permettant une automatisation des audits et une réduction des erreurs.
- **Améliorer la qualité des audits** en fournissant des rapports détaillés et des recommandations de sécurité basées sur des analyses comparatives avec des baselines reconnues.

Nous travaillons déjà sur une première version de la solution qui se concentre principalement sur les services **M365** Sharepoint et Onedrive, en particulier sur SharePoint et OneDrive. Le périmètre des services Microsoft à auditer est présenté ci-dessous :



FIGURE 2.1 – Périmètre des services Microsoft à auditer

Cette première version (V1) inclut SharePoint et OneDrive sous **M365**. Les versions futures (V2, V3, V4) s'étendront à d'autres services tels que SharePoint Premium, Teams, Exchange, Power Platform, et les services de sécurité et de conformité. En parallèle, l'audit des services **Azure** commencera avec la version V2, incluant Entra ID et les applications.

Pour illustrer ces objectifs, le processus d'audit peut être visualisé ci-dessous où on détaille les étapes clés de la collecte des données jusqu'à la remédiation :

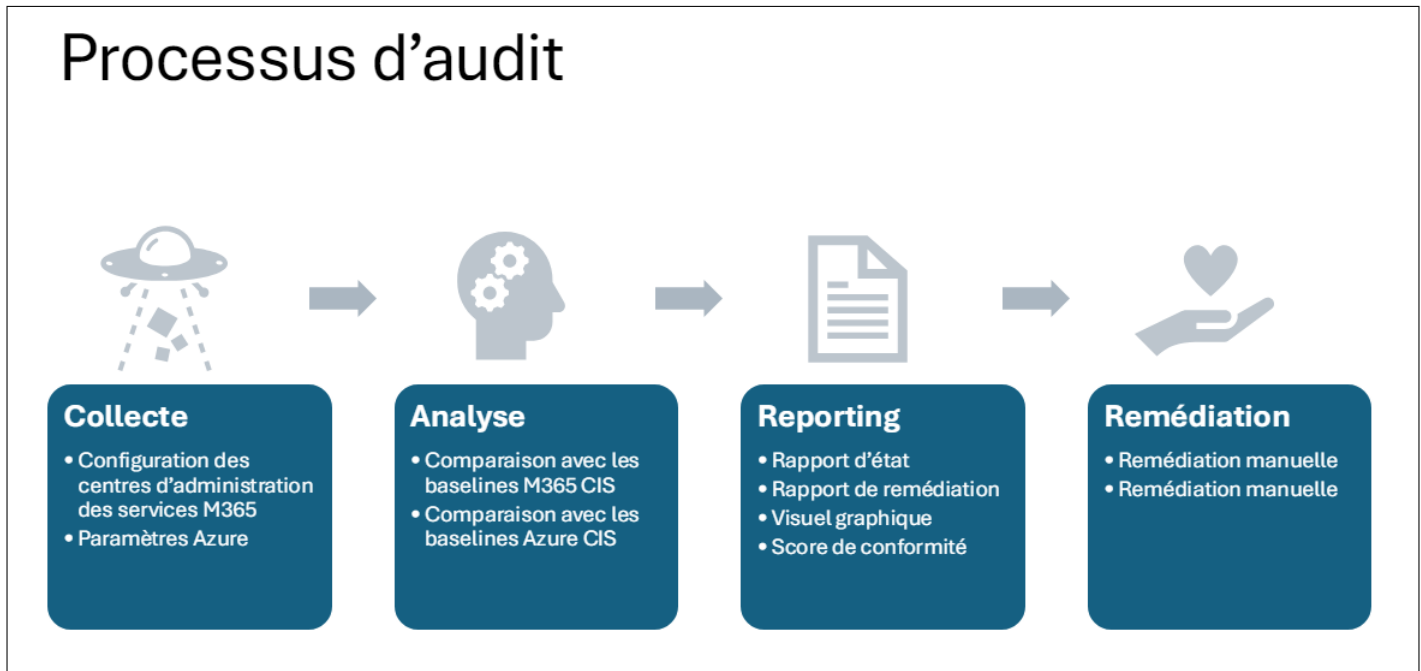


FIGURE 2.2 – Processus d'audit

Ce schéma montre le flux complet du processus d'audit, incluant la collecte des configurations des services **M365**, l'analyse comparative avec les baselines **M365 CIS**(Center for Internet Security) et **Azure CIS**, le reporting sous forme de rapports d'état et de remédiation, et enfin, la remédiation manuelle des écarts identifiés. Ce processus vise à garantir une meilleure efficacité et précision des audits, répondant ainsi aux objectifs principaux du projet.

2.1.3 Étapes clés

Le projet se déroulera en plusieurs phases principales :

1. **Phase de cadrage fonctionnel** : Identification des besoins, analyse des exigences et définition des User Stories.
2. **Phase de développement** : Création de l'interface web d'administration en utilisant des technologies telles que **Microsoft .NET**, **Azure**, **M365** et **PowerShell**.
3. **Phase de tests et validation** : Réalisation des tests unitaires, des tests d'intégration et des tests de régression pour garantir la qualité et la fiabilité de la solution.
4. **Phase de déploiement** : Mise en production de l'application, optimisation des performances et formation des utilisateurs finaux.
5. **Phase de suivi et d'amélioration continue** : Analyse des retours utilisateurs, correction des anomalies et mise à jour de l'application en fonction des besoins évolutifs.

2.1.4 Méthodologie de travail

Pour assurer une progression structurée et efficace du projet, une méthodologie de travail bien définie a été mise en place. Celle-ci repose sur plusieurs éléments clés.

Des points de suivi sont organisés deux fois par semaine avec l'équipe, chaque mardi et jeudi. Ces réunions permettent de faire le point sur l'avancement du projet, de débloquer d'éventuelles difficultés, et d'assurer un suivi rigoureux de la progression.

Un tableau Kanban est utilisé sur Microsoft **Planner** pour la gestion des tâches. Ce tableau permet de visualiser les différentes étapes du projet et les tâches associées, facilitant ainsi la répartition du travail et le suivi de l'avancement. Chaque tâche est attribuée aux membres de l'équipe en fonction de leurs compétences et disponibilités.

Ci-dessous, une image du tableau Kanban utilisé, illustrant les différents types de tâches et leur répartition :

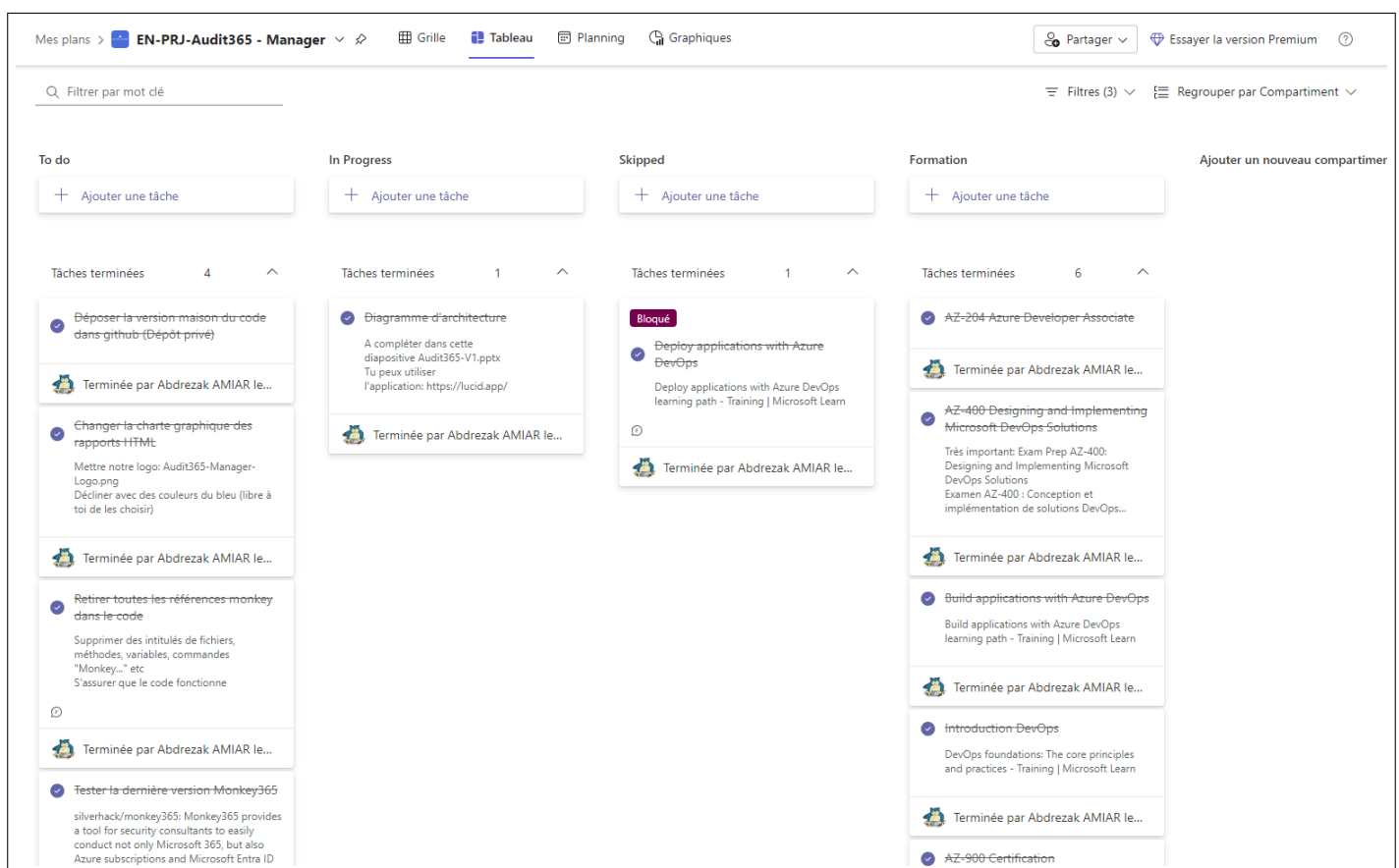


FIGURE 2.3 – Tableau Kanban sur Microsoft Planner

Cette méthodologie de travail collaborative et visuelle permet de maintenir une communication fluide et une organisation claire, garantissant ainsi l'atteinte des objectifs du projet de manière efficace et structurée.

2.2 Benchmark des solutions disponibles

2.2.1 Définition de Benchmark

Dans le contexte de notre projet, le benchmarking consiste à comparer les solutions existantes d’audit et de gouvernance des **SI M365**. En analysant les fonctionnalités, les performances et les résultats obtenus par ces solutions, nous pouvons évaluer leur efficacité et identifier les points forts et les lacunes de notre propre solution. Cette démarche nous aide à définir des objectifs clairs et ambitieux, tout en garantissant que notre solution répond aux attentes élevées du marché et des utilisateurs finaux.

2.2.2 Exemples de notre cas

Pour illustrer le benchmarking dans le contexte de notre projet, nous avons analysé plusieurs solutions existantes d’audit et de gouvernance des **SI M365**. Voici quelques exemples de ces solutions :

Microsoft365DSC

Microsoft365DSC est une solution open source qui permet de gérer et de configurer les paramètres de **M365** de manière déclarative. Elle utilise des scripts PowerShell pour déployer et auditer les configurations, facilitant ainsi la gestion de la conformité et des changements dans l’environnement **M365**.



FIGURE 2.4 – Logo Microsoft365DSC

Monkey365

Monkey365 est un outil spécialisé dans l’audit et la surveillance des environnements **M365** et **Azure**. Il offre des fonctionnalités avancées pour la collecte et l’analyse des données de configuration, permettant aux administrateurs de suivre les modifications et de garantir la conformité aux politiques de sécurité.

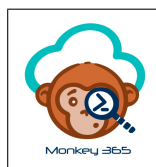


FIGURE 2.5 – Logo Monkey365

Maester

Maester est un outil de gouvernance et d'audit pour **Azure**, conçu pour simplifier la gestion des configurations et la conformité. Il offre des fonctionnalités pour l'audit des modifications, la gestion des permissions et l'analyse des configurations, aidant ainsi les administrateurs à gérer efficacement leurs environnements Cloud.



FIGURE 2.6 – Logo Maester

EasyGovernance

EasyGovernance est une plateforme qui propose des solutions de gouvernance pour **M365**. Elle inclut des modules pour l'audit, la gestion des accès et des permissions, ainsi que la conformité réglementaire. EasyGovernance aide les organisations à maintenir un contrôle strict sur leurs environnements **M365**.

2.2.3 Résultat du Benchmark

Après avoir analysé plusieurs solutions, nous avons sélectionné l'application de référence la plus alignée à nos besoins : **MONKEY365**. Voici les conclusions tirées de notre benchmark :

- **Microsoft365DSC** est une solution très avancée qui permet de gérer et configurer les paramètres de Microsoft 365 de manière déclarative. Cependant, sa complexité et ses nombreuses fonctionnalités dépassent largement nos besoins actuels, rendant cette solution surdimensionnée pour notre projet.
- **EasyGovernance** propose des fonctionnalités de gouvernance pour **M365**, mais elle est encore en développement. Elle présente plusieurs bugs et certaines de ses fonctionnalités ne fonctionnent que partiellement, ce qui la rend moins fiable pour notre utilisation.
- **Maester** est une solution efficace pour la gouvernance et l'audit, mais elle se concentre uniquement sur la partie **Azure**. Étant donné que notre projet se concentre sur les deux environnements **M365** et **Azure**, cette solution ne répond pas entièrement à nos besoins actuels.
- **MONKEY365** est la solution qui se rapproche le plus de nos objectifs. Elle offre des fonctionnalités avancées pour l'audit et la surveillance des environnements **M365** et **Azure**, ce qui correspond parfaitement à nos besoins. Sa capacité à collecter et analyser les données de configuration, ainsi que sa fiabilité et sa conformité aux politiques de sécurité, en font l'application de référence idéale pour notre projet.

Ainsi, **MONKEY365** a été sélectionnée comme l'application de référence alignée à nos besoins, nous permettant d'atteindre nos objectifs de manière efficace et fiable.

2.3 Étude Conceptuelle

2.3.1 Nos besoins

Pour mener à bien le projet **Audit365 Manager**, il est crucial de bien définir les besoins spécifiques. Ces besoins seront articulés autour de deux aspects principaux : les personas et les user stories, qui permettent de comprendre les utilisateurs cibles et les fonctionnalités attendues.

Personas

Les personas représentent les profils types des utilisateurs finaux de notre solution. Ils aident à mieux cerner leurs attentes, leurs objectifs et leurs comportements. Voici quelques exemples de personas identifiés pour notre projet :

- **Auditeur général** : Responsable de la surveillance et de l'évaluation de l'ensemble des services et fonctionnalités de la plateforme **M365**. Son objectif est de s'assurer que les politiques et les normes de l'organisation sont respectées.
- **Auditeur des services SharePoint** : Spécialisé dans la surveillance des environnements SharePoint Online. Il veille à ce que les sites, les bibliothèques de documents et les permissions respectent les normes de l'organisation. (Persona Visé pour notre V1)
- **Auditeur des services de collaboration Teams** : Chargé d'assurer le bon fonctionnement des fonctions de collaboration dans Teams. Il surveille la gestion des canaux, des équipes et des politiques associées.

User stories

Les user stories sont des descriptions simples et concises des fonctionnalités que les utilisateurs finaux souhaitent obtenir. Elles sont formulées du point de vue de l'utilisateur et servent de base pour le développement des fonctionnalités. Voici quelques user stories pour notre projet :

Catégorie	User Story
Collecte	En tant qu'auditeur des services SharePoint, je souhaite avoir accès à un backoffice pour configurer les paramètres de collecte de données de SharePoint afin de m'assurer que toutes les configurations nécessaires sont correctement auditées.
Analyse	En tant qu'auditeur des services SharePoint, je souhaite comparer les configurations collectées avec les baselines M365 CIS afin d'identifier les écarts de conformité spécifiques à SharePoint.
Reporting	En tant qu'auditeur des services SharePoint, je souhaite générer des rapports d'état détaillés sur la conformité de SharePoint afin de fournir une vue claire de la posture de sécurité et des domaines nécessitant des améliorations.
Remédiation	En tant qu'auditeur des services SharePoint, je veux pouvoir effectuer des remédiations manuelles ou automatiques des écarts de conformité identifiés afin de corriger les configurations de sécurité de SharePoint.
Backoffice	En tant qu'auditeur des services SharePoint, je souhaite configurer les paramètres d'audit tels que les services spécifiques à auditer via le backoffice afin de m'assurer que les audits sont effectués de manière ciblée.

TABLE 2.1 – User Stories

Ces personas et user stories permettent de structurer le développement du projet en se concentrant sur les besoins réels des utilisateurs, garantissant ainsi une solution efficace et adaptée.

2.3.2 Référentiels des normes de sécurité (Security Framework)

Pour assurer la conformité et la sécurité des **SI M365**, il est essentiel de se baser sur des référentiels de normes de sécurité reconnus, souvent nommés "Baselines". Ces référentiels fournissent des lignes directrices et des meilleures pratiques pour garantir que les configurations et les opérations des **SI** répondent aux exigences de sécurité et de conformité. Parmi les principaux référentiels utilisés, nous avons :

CIS (Center for Internet Security) : Le CIS propose des benchmarks et des guidelines de sécurité spécifiques à différents environnements, dont **M365** et **Azure**. Ces benchmarks fournissent des configurations de sécurité recommandées pour protéger contre les menaces courantes et assurer la conformité aux meilleures pratiques de sécurité. Dans notre projet, nous avons décidé d'utiliser le référentiel CIS comme référence principale, car il est le plus conforme à nos besoins. Toutefois, il est possible que nous fassions des ajustements en fonction des situations spécifiques, tout en conservant CIS comme notre première référence.

SCuBA (Secure Cloud Business Applications) : SCuBA est un ensemble de recommandations et de pratiques pour sécuriser les applications cloud. Il propose des solutions pour gérer les risques de sécurité liés à l'utilisation des services cloud, en mettant l'accent sur la protection des données et la gestion des accès.

BluePrint : BluePrint offre des modèles et des configurations sécurisées pour divers services **M365** et **Azure**. Il aide à déployer des environnements conformes aux normes de sécurité, facilitant ainsi la mise en place de mesures de protection robustes et cohérentes.

En plus de ces référentiels, il existe d'autres normes et frameworks de sécurité, tels que NIST (National Institute of Standards and Technology), ISO/IEC 27001, et GDPR (General Data Protection Regulation), qui peuvent être utilisés pour renforcer la sécurité et la conformité des systèmes d'information. Ces référentiels fournissent des lignes directrices supplémentaires pour la gestion des risques, la protection des données et la gouvernance des **SI**.

L'adoption de ces référentiels de sécurité permet de garantir que notre solution **Audit365 Manager** est alignée avec les meilleures pratiques de l'industrie, assurant ainsi une protection optimale des environnements **M365** et **Azure**.

2.4 Récapitulatif

*Ce chapitre a permis de définir le périmètre du projet **Audit365 Manager**. Nous avons examiné le contexte, les objectifs principaux, les étapes clés, et la méthodologie de travail. Un benchmark des solutions existantes a été réalisé, menant à la sélection de **MONKEY365** comme application de référence. Enfin, une étude conceptuelle a été menée pour définir les besoins et les référentiels de sécurité.*

Le prochain chapitre traitera du cadrage technique du projet.

Chapitre 3

Cadrage Technique

Ce chapitre est consacré au **cadrage technique** du projet **Audit365 Manager**. Il vise à analyser en profondeur les aspects techniques nécessaires pour la réalisation du projet, en se concentrant sur l'existant et les personnalisations à apporter. Nous débuterons par une analyse de l'existant, incluant une revue de code de la librairie tierce partie Monkey365, le développement d'un proof of concept (**POC**), et les tests associés. Nous examinerons également les problèmes rencontrés, les solutions mises en place et les vérifications de sécurité effectuées.

Ensuite, nous aborderons la personnalisation de l'outil Monkey365, en modifiant le framework de sécurité et le front-end. Enfin, nous concevrons l'architecture de déploiement nécessaire pour intégrer la solution dans l'environnement de production. Ce chapitre détaillera chaque étape technique pour assurer une mise en œuvre réussie du projet.

3.1 Analyse de l'existant

3.1.1 Examen du code de la librairie tierce Monkey365

Dans cette section, j'ai entrepris une analyse détaillée du code de la librairie tierce Monkey365 pour mieux comprendre son fonctionnement et évaluer son adéquation avec notre projet. Monkey365 est principalement écrit en PowerShell, un langage qui m'était relativement nouveau. J'ai donc dû me familiariser avec ses spécificités avant de pouvoir évaluer efficacement le code.

J'ai commencé par explorer la structure du code source, qui est bien organisée en modules et fonctions distincts. Chaque module couvre des aspects spécifiques de l'audit et de la gouvernance des environnements Microsoft 365, ce qui facilite la compréhension et la navigation.

Ensuite, j'ai examiné les fonctionnalités principales offertes par Monkey365. L'outil fournit une gamme de commandes pour collecter des données, analyser des configurations et générer des rapports de plusieurs type de sorties. Ces fonctionnalités sont pertinentes et alignées avec les besoins de notre projet, ce qui m'a rassuré quant à sa capacité à répondre à nos exigences.

Concernant la qualité du code, j'ai constaté que le code PowerShell de Monkey365 est "très" bien structuré et commenté, ce qui en facilite la lisibilité et la maintenabilité. La documentation disponible sur le site officiel de Monkey365 m'a également été très utile pour comprendre les différentes commandes et leur utilisation.

Enfin, j'ai examiné les dépendances externes utilisées par Monkey365. Ces dépendances sont courantes dans l'écosystème PowerShell et ne posent pas de problèmes majeurs, mais nécessitent une surveillance continue pour s'assurer qu'elles restent à jour et sécurisées.

En résumé, l'examen du code de Monkey365 m'a permis de conclure que cet outil est bien conçu, fonctionnellement riche et aligné avec les objectifs de notre projet. Cependant, quelques ajustements et renforcements de la sécurité seront nécessaires pour l'intégrer pleinement et en toute confiance dans notre solution Audit365 Manager.

3.1.2 Développement du Proof of Concept (POC)

Après avoir examiné le code de Monkey365, j'ai entrepris le développement d'un Proof of Concept (**POC**) pour valider l'utilisation de cette librairie dans notre projet Audit365 Manager. L'objectif de ce **POC** était de tester les fonctionnalités principales de Monkey365 et d'évaluer son intégration avec notre environnement.

Pour commencer, j'ai suivi les instructions détaillées fournies dans la documentation officielle de **Monkey365** pour installer l'outil. Les prérequis incluent une version récente de PowerShell et l'accès à un environnement Microsoft 365 avec les permissions nécessaires pour exécuter les audits.

L'installation a été relativement simple. J'ai utilisé les commandes PowerShell fournies pour télécharger et configurer Monkey365. La documentation, disponible sur le site de Monkey365, m'a guidé à travers chaque étape du processus d'installation. Voici un résumé des étapes suivies :

1. Mise à jour de PowerShell à la version requise.
2. Téléchargement de Monkey365 depuis le Git Bash.
3. Installation des modules et des dépendances nécessaires.
4. Configuration des paramètres initiaux pour permettre à Monkey365 d'accéder aux services Microsoft 365.

Une fois l'installation terminée, j'ai configuré l'environnement de test en définissant les paramètres de connexion à Microsoft 365. J'ai vérifié que toutes les dépendances étaient satisfaites et que l'outil pouvait accéder aux données nécessaires pour les audits.

Cette préparation a permis de s'assurer que Monkey365 était opérationnel et prêt pour les tests fonctionnels et d'intégration dans le cadre de notre projet Audit365 Manager.

3.1.3 Évaluation du Proof of Concept

Avec Monkey365 installé et configuré, j'ai entrepris l'évaluation du **POC** pour tester ses fonctionnalités et vérifier son adéquation avec nos besoins. Voici les étapes que j'ai suivies :

- **Configuration des scénarios de test** : J'ai mis en place des scénarios d'audit pour différents services Microsoft 365, tels que SharePoint, OneDrive, Teams et autres, en utilisant les commandes de Monkey365.
- **Exécution des audits** : J'ai exécuté les scripts d'audit fournis par Monkey365 pour collecter des données sur les configurations actuelles des services. Cela m'a permis de vérifier que l'outil pouvait récupérer des informations précises et complètes.

- **Analyse des résultats** : J'ai comparé les données collectées par Monkey365 avec les configurations réelles de notre environnement Microsoft 365 pour m'assurer de leur exactitude.
- **Génération de rapports** : J'ai utilisé les fonctionnalités de reporting de Monkey365 pour créer des rapports détaillés dans divers formats tels que HTML, Excel, JSON, et autres. Ces rapports contiennent des recommandations de remédiation pour les configurations non conformes.

L'évaluation du **POC** a démontré que Monkey365 est capable de répondre efficacement à nos besoins d'audit. Les données collectées étaient précises et les rapports générés fournissaient des informations utiles pour améliorer la sécurité et la conformité de notre environnement Microsoft 365. Toutefois, il n'a pas été facile d'arriver à ce point, car j'ai rencontré plusieurs problèmes en cours de route. J'ai réservé une discussion de ces problèmes pour la section suivante.

3.1.4 Problèmes identifiés

Au cours du développement et de l'évaluation du Proof of Concept (POC) avec Monkey365, j'ai rencontré plusieurs problèmes qui ont entravé le bon déroulement des tests.

Le premier problème s'est manifesté lors de l'importation de Monkey365 dans PowerShell. Une erreur indiquant que le module "watcher" n'avait pas été trouvé est apparue. En examinant les fichiers, j'ai constaté que le dossier "watcher" était effectivement manquant, empêchant ainsi l'outil de fonctionner correctement. La prochaine image illustre cette erreur rencontrée.

```
PS C:\Users\amiar> Import-Module C:\Users\amiar\Desktop\monkey365
Exception calling "EnumerateFiles" with "3" argument(s): "Could not find a part of the path
'C:\Users\amiar\Desktop\monkey365\core\watcher'."
At C:\Users\amiar\Desktop\monkey365\monkey365.psm1:35 char:5
+ [System.IO.Directory]::EnumerateFiles("{0}\1}" -f $PSScriptRoot, ...
+ ~~~~~
+ CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
+ FullyQualifiedErrorId : DirectoryNotFoundException
```

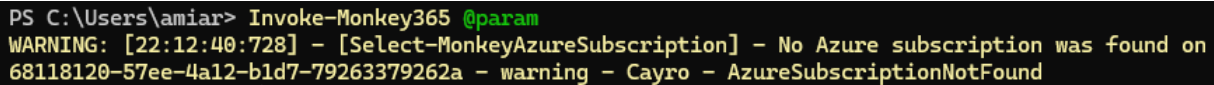
FIGURE 3.1 – Erreur importation module

Ensuite, lors de l'exécution des audits sur SharePoint, Teams et OneDrive, des erreurs de droits d'accès ont été rencontrées. Ces services dépendent de SharePoint pour certaines configurations, et les erreurs de droits ont empêché la collecte des données nécessaires pour les audits, compliquant ainsi l'évaluation des configurations. La prochaine image montre l'erreur de droits d'accès.

```
PS C:\Users\amiar> Import-Module 'C:\Users\amiar\Desktop\Monkey365'
PS C:\Users\amiar> $param = @{
>> ClientId = '8bcd0361-c72f-48b5-84ad-37c877162dc7';
>> ClientSecret = ('il-N8Q-O5uu.1-N-FPdHZkxbrWt-74WhZd33Qkbh9' | ConvertTo-SecureString -AsPlainText -Force)
>> Instance = 'Microsoft365';
>> Analysis = 'SharepointOnline';
>> TenantID = '68118120-57ee-4a12-b1d7-79263379262a';
>> ExportTo = @("HTML");
>> }
PS C:\Users\amiar> Invoke-Monkey365 @param
WARNING: [22:10:27:883] - [Get-MonkeyMSALToken] - Acquire token failed. A configuration issue is preventing
authentication - check the error message from the server for details. You can modify the configuration in the
application registration portal. See https://aka.ms/msal-net-invalid-client for details. Original exception:
AADSTS7000215: Invalid client secret provided. Ensure the secret being sent in the request is the client secret value,
not the client secret ID, for a secret added to app '8bcd0361-c72f-48b5-84ad-37c877162dc7'. Trace ID:
35ea90cd-774d-498a-b0ec-400a642a0100 Correlation ID: d50c69e0-43f5-4ddd-b3ba-f3451c3a901f Timestamp: 2024-06-21
22:10:39Z - warning - Cayro -
WARNING: [22:10:27:925] - [Get-MSALTokenForGraphV2] - Unable to get token for Microsoft Graph - warning - Cayro -
```

FIGURE 3.2 – Erreur droits Sharepoint

Enfin, lors de l'exécution de l'audit Azure, j'ai rencontré une erreur en raison de l'absence de souscription Azure dans notre environnement de test. Cette absence de souscription a empêché l'audit de récupérer les informations requises pour évaluer les configurations Azure. La prochaine image démontre cette erreur liée à l'absence de souscription.



```
PS C:\Users\amiar> Invoke-Monkey365 @param
WARNING: [22:12:40:728] - [Select-MonkeyAzureSubscription] - No Azure subscription was found on
68118120-57ee-4a12-b1d7-79263379262a - warning - Cayro - AzureSubscriptionNotFound
```

FIGURE 3.3 – Erreur souscription Azure

Ces problèmes ont nécessité une attention particulière, ainsi que l'intervention des membres de mon équipe et de plusieurs experts. Dans la section suivante, je discuterai des solutions que nous avons mises en place pour surmonter ces obstacles et permettre la poursuite du projet.

3.1.5 Solutions apportées

Pour résoudre le problème lié au module "watcher", ma première solution a été de créer un dossier vide nommé "watcher" dans le chemin dédié. Bien que cela ait fonctionné, je savais que ce n'était pas la meilleure façon de traiter ce problème. J'ai ensuite supprimé ce dossier et cherché les fichiers ou lignes de code appelant le module, afin de modifier ces derniers pour enlever la dépendance en supprimant la ligne qui appelait "watcher". Cette approche a également fonctionné, mais elle n'était pas satisfaisante à long terme. **Mauslem Z.** m'a suggéré de consulter les issues sur GitHub, et effectivement, j'ai trouvé que quelqu'un avait déjà signalé ce problème. Une réponse de l'auteur de Monkey365 indiquait qu'il y avait effectivement un problème en cours de correction et qu'il fallait utiliser la branche "dev" au lieu de la branche "main" pour obtenir une version fonctionnelle. En utilisant la branche "dev", nous avons trouvé le module "watcher" complet, ce qui nous a semblé plus sécurisé. Quelques jours plus tard, le créateur de l'outil a publié une nouvelle version sans ce problème, que nous avons adoptée pour nos modifications.

Pour les problèmes de droits d'accès lors des audits **SharePoint**, **Teams** et **OneDrive**, je me suis référé à la documentation de **Monkey365**, qui explique qu'il existe trois méthodes d'authentification : directement avec mon compte, avec un **certificat**, et avec un **code secret**. Les deux dernières méthodes nécessitent que l'application soit enregistrée sur notre tenant. Pour des raisons de sécurité, j'ai d'abord évité d'utiliser mon compte directement. En utilisant mes connaissances sur **Azure** et les ressources en ligne telles que StackOverFlow, reddit et autres blogs, j'ai créé une application dans **Microsoft Entra ID** (le répertoire de tout le tenant), généré un code secret et configuré l'outil pour se connecter. Cependant, cette méthode nécessitait des droits d'accès spécifiques. En tant qu'administrateur, j'ai pu attribuer les droits nécessaires conformément à la documentation, mais j'ai tout de même rencontré des erreurs lors de l'exécution des audits. En essayant avec un certificat, j'ai rencontré le même problème. Après avoir relu la documentation, j'ai trouvé un tableau qui présente les types d'authentification et les services auxquels l'outil peut accéder. Ce tableau est présenté ci-dessous. J'ai découvert que l'authentification par code secret est très limitée, mais l'authentification par certificat devrait fonctionner. En discutant avec mon équipe, nous avons appris que Microsoft met à jour très fréquemment ses politiques de sécurité, ce qui pourrait expliquer les problèmes rencontrés. La seule solution qui restait était l'authentification directe avec mon compte, qui prend les droits "délégés". Pour cette solution, comme l'application va se voir déléguer des droits, il est

nécessaire que le compte utilisé pour lancer un audit soit administrateur du service concerné. Par exemple, pour lancer un audit de SharePoint, le compte doit être administrateur SharePoint, et de même pour les autres services. Un administrateur global, en revanche, peut lancer des audits de tous les services ou de plusieurs services simultanément, ce qui est mon cas. En utilisant mon compte administrateur global, l'outil a fonctionné correctement. Cependant, nous devons rester vigilants car la source de l'outil n'est pas totalement fiable. Je discuterai des vérifications de sécurité dans la prochaine sous-partie.

Authentication method	Azure AD Portal	Azure AD GraphV2	Azure services	Exchange Online	Purview	SharePoint Online
Interactive browser authentication	✓	✓	✓	✓	✓	✓
Service Principal Authentication (Client Secret)	✗	✓	✓	✗	✗	✗
Service Principal Authentication (Certificate Secret)	✗	✓	✓	✓	✓	✓

FIGURE 3.4 – Méthodes d'authentification prises en charge par service

Pour le problème de souscription **Azure**, j'ai remonté le problème à mon équipe, et on m'a conseillé de ne pas trop me focaliser dessus pour l'instant, car la première version du projet ne nécessite pas l'analyse des services cloud donc pas d'abonnements **Azure**. On m'a recommandé de me concentrer sur notre objectif principal afin de progresser, car les problèmes précédents nous ont déjà coûté beaucoup de temps.

Ces solutions, bien que variées et parfois complexes, m'ont permis de surmonter les obstacles rencontrés et de faire avancer le projet de manière significative. Elles ont également permis d'utiliser la connaissance des experts et de favoriser la communication et la **coopération** au sein de l'équipe.

3.1.6 Vérifications de sécurité

En ce qui concerne la sécurité, nous avons effectué uniquement le minimum nécessaire pour nous assurer que l'utilisation de Monkey365 ne présente pas de risques majeurs. Pour ce faire, nous avons utilisé Fiddler pour vérifier les flux de données. Bien que Fiddler soit un nouvel outil pour moi, le concept ne m'était pas étranger, car j'avais déjà utilisé Wireshark à l'université.

Tout d'abord, j'ai lancé PowerShell et identifié l'ID du processus à suivre pour surveiller ses flux. J'ai ensuite configuré Fiddler pour écouter spécifiquement ce processus. En lançant un audit et en suivant tous les flux de données dans Fiddler, nous n'avons détecté aucun élément suspect, ce qui nous a quelque peu rassurés.

De plus, j'ai essayé d'examiner le code source de Monkey365. Bien que le code soit immense et difficile à analyser entièrement,

une lecture partielle n'a révélé rien de suspect. Il est également important de noter que les audits ont été initialement exécutés dans des environnements de test et non sur des systèmes critiques ou officiels.

Après avoir été rassurés par ces vérifications, nous avons exécuté des audits sur d'autres tenants. Les résultats ont été positifs et l'outil a fonctionné correctement. Nous avons ainsi conclu ce Proof of Concept (POC) avec succès.

3.2 Personnalisation de l'outil Monkey365

Après avoir vérifié la sécurité de Monkey365 et validé son fonctionnement de base, nous avons entamé la phase de personnalisation de l'outil pour qu'il réponde spécifiquement à nos besoins. Cette personnalisation vise à transformer Monkey365 en une application adaptée à nos exigences particulières.

3.2.1 Adaptation du framework de sécurité

Pour adapter Monkey365 à nos besoins spécifiques, j'ai d'abord cherché dans le code jusqu'à ce que je trouve le fichier contenant le framework de sécurité CIS. J'ai comparé ce fichier avec le véritable benchmark CIS, et il s'est avéré correct. Cependant, le framework était en format JSON, ce qui nécessitait des ajustements pour correspondre à notre situation spécifique. J'ai modifié certaines règles, en supprimé quelques-unes et ajusté d'autres.

Cette adaptation m'a donné l'idée de développer, à l'avenir, un générateur de JSON avec une interface où nous pourrions configurer les paramètres comme nous le souhaitons, ou importer une baseline déjà établie comme le CIS. Nous pourrions alors apporter des modifications ou conserver la configuration telle quelle et générer le JSON correspondant. Cette idée a été bien accueillie par l'équipe et a été prise en compte pour une utilisation future.

3.2.2 Modification de l'interface utilisateur

Pour commencer, j'ai supprimé toutes les références à "monkey" dans le code. Étant donné la taille du code, cela risquait de casser l'outil. Pour faciliter cette tâche, j'ai utilisé mes connaissances et des recherches en ligne pour créer un **script PowerShell** qui parcourt les fichiers et les lignes de code afin de remplacer toutes les occurrences de "monkey" par "**audit**". Le script a bien fonctionné, mais le code s'est tout de même cassé. Pourquoi ? Parce que le script ne respectait pas les majuscules. Par exemple, "**Monkey**" devait être transformé en "**Audit**" et non en "audit". Il y a eu de nombreux problèmes liés à cette subtilité.

Même si mon script fonctionnait bien, il ne s'adaptait pas parfaitement à la situation. J'ai cherché une solution alternative et trouvé que Visual Studio Code permettait de remplacer des mots en respectant les majuscules et minuscules. J'ai donc pu effectuer le changement de manière semi-manuelle. Après avoir vérifié que mon changement n'affectait pas le fonctionnement du code, j'étais satisfait. Désormais, pour lancer un audit, nous utilisons "**Invoke-Audit**" au lieu de "**Invoke-Monkey**".

Ensuite, je me suis attaqué au front-end et j'ai découvert quelque chose de très étrange : il n'y avait aucun fichier **HTML**, **CSS**, ni **JavaScript**, seulement du **PowerShell**. Comment alors le **HTML** est-il généré ? J'ai passé du temps à découvrir qu'une variable PowerShell cachée au fond du code contenait un template **HTML** en string, en une seule ligne, qui n'était qu'un appel aux styles, scripts et librairies. Mais l'énigme n'était pas encore résolue. En traçant un peu plus, j'ai trouvé un fichier zip nommé "assets" conte-

nant tous les logos, librairies externes comme **Bootstrap**, **CSS** et **JS**. Ce qui m'a impressionné, c'est que tout était contenu dans un fichier zip. Le code générait le **HTML** à partir du template en extrayant le zip dans le dossier de sortie.

Avec cette découverte, j'ai pu faire les modifications nécessaires au niveau du **front-end**, en changeant les styles et scripts dans le fichier "assets.zip". J'ai continué à chercher jusqu'à trouver que chaque fois qu'une analyse d'audit se lançait, elle appelait des scripts PowerShell pour générer des éléments HTML, comme "New-SideBar.ps1" pour la barre de navigation ou "New-HtmlCard.ps1" pour les cartes HTML. Grâce à toutes ces découvertes, j'ai modifié du **PowerShell** et du **CSS** dans "assets.zip" pour donner de nouvelles couleurs à notre outil et changer le logo, afin de prouver ma capacité à effectuer des modifications au niveau du **front-end**.

Les deux images suivantes montrent l'édition officielle de Monkey365 et notre petite modification de personnalisation. Après avoir prouvé ces changements, nous avons sollicité notre coéquipier, **Pisda**, graphiste designer, pour créer un nouveau design pour notre outil. Nous avons donc temporairement arrêté les modifications **front-end** en attendant sa maquette.

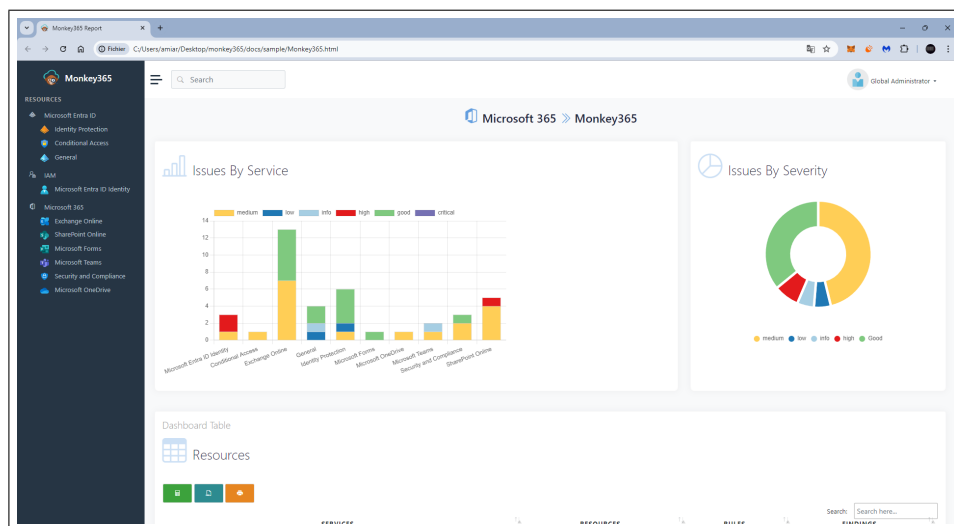


FIGURE 3.5 – Interface Monkey365 avant changement

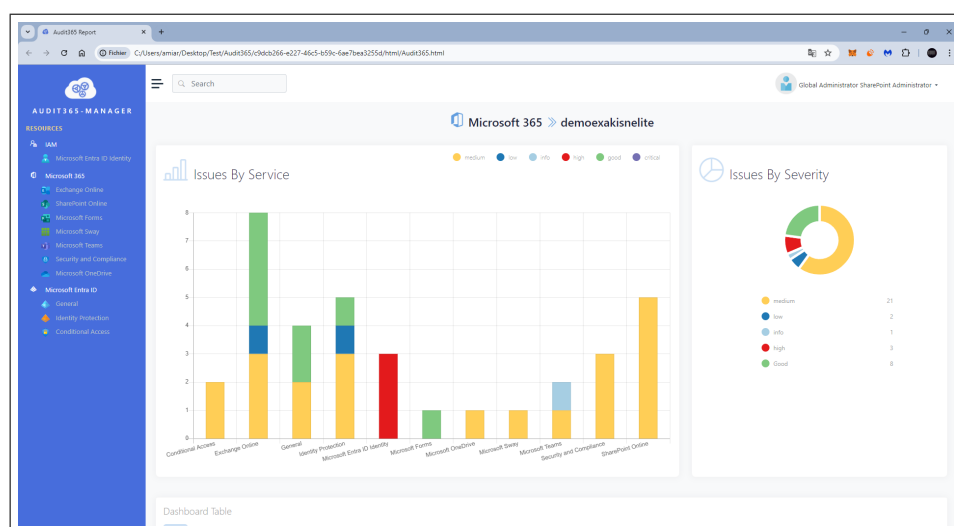


FIGURE 3.6 – Interface Monkey365 après changement (Audit365)

3.2.3 Conception de l'architecture de l'application

On m'a confié la tâche de concevoir une architecture pour notre application. Bien que cette mission dépasse largement mon niveau actuel, l'objectif était que je puisse apprendre de mes erreurs après validation par un expert. Voici la conception que j'ai réalisée.

L'architecture proposée repose sur plusieurs composants clés pour assurer une gestion efficace et sécurisée des audits :

- **Utilisateurs et Entra ID** : Les utilisateurs s'authentifient via Entra ID, qui gère l'authentification et l'autorisation des accès.
- **Web App** : Cette application web, déployée via un App Service Plan, est le cœur du système. Elle héberge le backend (Rest API), le frontend (interface utilisateur), ainsi qu'un conteneur de l'application.
- **Function Apps** : Ces applications de fonctions, qui sont des services serverless, exécutent les scripts PowerShell nécessaires à l'importation des modules et à l'invocation des audits.
- **Container Image** : Une image de conteneur est utilisée pour déployer l'outil Monkey365 et les composants nécessaires de manière cohérente et reproductible.
- **Monkey365** : Cet outil interroge les environnements Azure et Microsoft 365 pour réaliser les audits.
- **Azure SQL Database** : La base de données stocke les résultats des audits et autres données nécessaires au fonctionnement de l'application.
- **Outputs** : Les résultats des audits sont initialement diffusés en page HTML, mais peuvent être transformés en divers formats, y compris Excel, JSON, CSV et CLIXML.
- **Diagnostic logs and metric data** : Les données de diagnostic et les métriques sont envoyées à Log Analytics et Azure Monitor pour surveiller la santé de l'application.

L'image suivante montre l'architecture conçue par moi sur l'application Lucidchart :

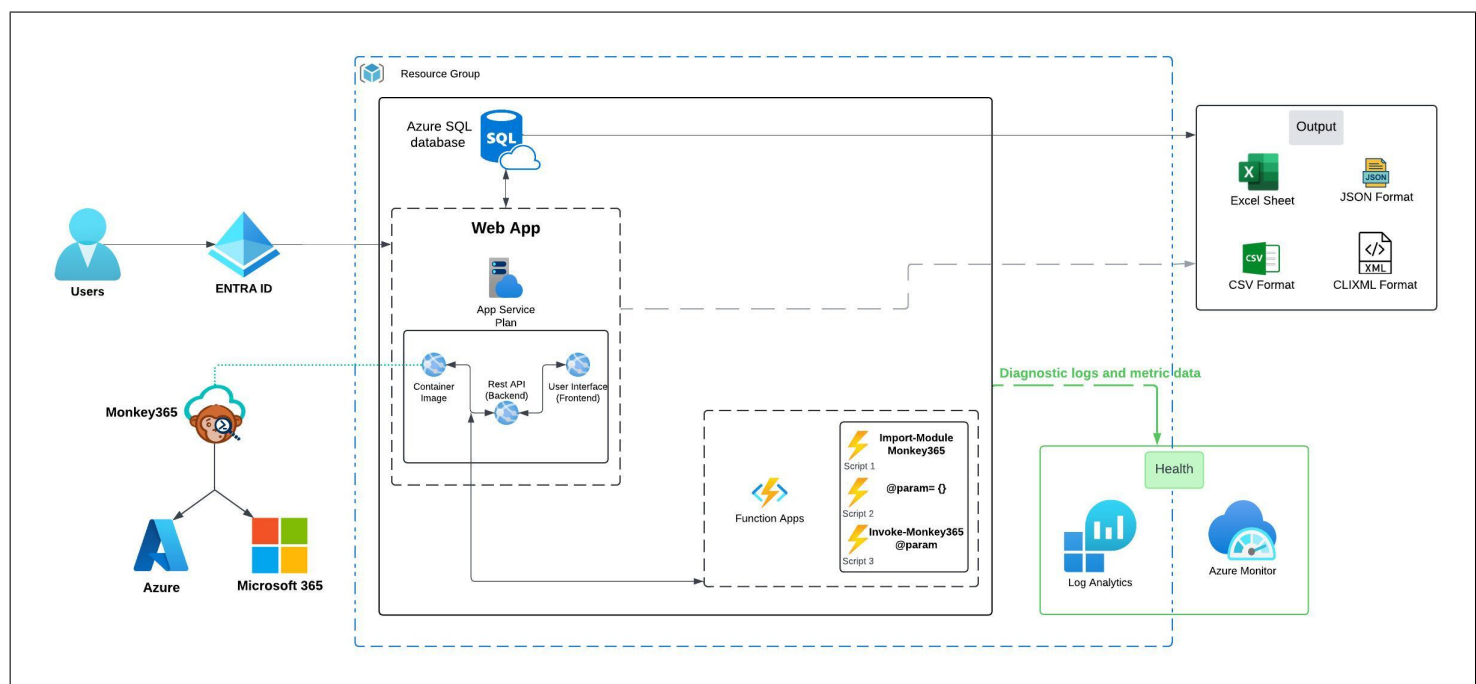


FIGURE 3.7 – Architecture de déploiement proposée pour l'application

Cette architecture permet une intégration fluide entre les différents composants et assure que les audits peuvent être exécutés efficacement et en toute sécurité. L'architecture a été partiellement validée par l'équipe, mais nécessite toujours l'approbation finale d'un architecte cloud, que nous attendons pour valider totalement ou apporter de petites modifications.

3.3 Gestion du dépôt du projet

Pour faciliter la gestion du code source et assurer une collaboration efficace, nous avons décidé d'utiliser un dépôt Git pour notre projet. Actuellement, le code du projet **Audit365 Manager** est hébergé sur mon dépôt personnel GitHub en visibilité privée. Cette solution temporaire nous permet de suivre les modifications et de travailler de manière collaborative.

Cependant, pour une gestion plus sécurisée et professionnelle, nous prévoyons de migrer le projet vers un dépôt professionnel d'entreprise. Cette migration nécessitera l'intervention d'un expert en infrastructure et DevOps, que nous attendons pour configurer et sécuriser le dépôt d'entreprise. Ce dépôt centralisé permettra de mieux gérer les versions, de suivre les modifications apportées par chaque membre de l'équipe, et d'assurer une meilleure intégration continue et déploiement continu (CI/CD).

3.4 Récapitulatif

Ce chapitre a été consacré au **cadrage technique** du projet **Audit365 Manager**. Nous avons d'abord effectué une **analyse de l'existant**, en examinant le code de la librairie tierce Monkey365, et en développant et testant un **Proof of Concept (POC)**. Cette analyse nous a permis de vérifier que Monkey365 répondait à nos besoins, mais a également révélé plusieurs problèmes.

Ensuite, nous avons discuté des solutions apportées pour surmonter ces obstacles. Nous avons également effectué des **vérifications de sécurité** en utilisant Fiddler pour s'assurer qu'aucun flux de données suspect n'était présent.

La deuxième partie du chapitre a porté sur la **personnalisation de l'outil Monkey365**. Nous avons adapté le framework de sécurité pour l'aligner avec nos besoins spécifiques. J'ai également entrepris des modifications de l'interface utilisateur. Ces modifications ont été validées par l'équipe, mais un travail de design supplémentaire a été confié à notre graphiste.

Puis, j'ai conçu une **architecture de déploiement** pour notre application, décrivant comment les différents composants s'intègrent pour assurer une gestion efficace des audits. Cette architecture a été partiellement validée par l'équipe, mais nécessite toujours l'approbation finale d'un architecte cloud.

Enfin, nous avons discuté de la gestion du dépôt du projet, qui est actuellement hébergé sur mon dépôt personnel GitHub, en attendant la mise en place d'un dépôt d'entreprise pour une gestion plus professionnelle et sécurisée.

Ce chapitre marque la fin de la progression de mon travail à la date de rédaction de ce rapport. Cependant, le projet **Audit365 Manager** continue d'évoluer, et mon stage n'est pas encore terminé. Le prochain chapitre abordera les formations et les apprentissages dont j'ai pu bénéficier au cours de ce stage.

Chapitre 4

Formations et Apprentissages

Dans ce chapitre, je vais détailler les différentes formations et apprentissages que j'ai suivis au cours de mon stage. Ces formations m'ont été recommandées par mon équipe pour améliorer mes compétences techniques et surmonter certains obstacles rencontrés dans le projet. D'autres initiatives ont été prises de mon propre chef pour mieux m'intégrer dans le domaine que je vise et démontrer mes capacités. L'objectif de ces formations était non seulement de résoudre les défis rencontrés durant le projet Audit365 Manager, mais aussi de renforcer ma compréhension globale des technologies et des pratiques pertinentes dans les domaines du **DevOps** et de l'**architecture cloud**, avec un accent particulier sur **Azure**.

4.1 Certifications

Durant mon stage, j'ai eu l'opportunité de me former et de passer des certifications essentielles pour approfondir mes connaissances dans les domaines de l'Azure et de Microsoft 365. Ces certifications, certaines déjà obtenues, ont non seulement renforcé ma compréhension des fondamentaux, mais ont également prouvé ma capacité à maîtriser des concepts complexes et à les appliquer de manière pratique. Dans cette section, je vais détailler les certifications que j'ai suivies, notamment Microsoft Azure Fundamentals AZ-900, Microsoft 365 Fundamentals MS-900, ainsi que la formation pour l'AZ-402. Chacune de ces certifications a été choisie pour répondre à des besoins spécifiques de mon projet et pour soutenir mon développement professionnel dans le domaine de l'architecture cloud et du DevOps.

4.1.1 Microsoft Azure Fundamentals AZ-900

Pour obtenir la certification **Microsoft Azure Fundamentals AZ-900**, j'ai suivi le parcours de formation proposé par Microsoft Learn. Ce parcours couvre les concepts fondamentaux du cloud computing, les services de base fournis par Azure, la gestion des coûts et la gouvernance. Les modules abordent également la sécurité, la confidentialité, la conformité et la confiance.

Voici un résumé des principales thématiques couvertes dans le parcours de Microsoft Learn :

- **Concepts de base du cloud** : Introduction aux principes fondamentaux du cloud computing, tels que l'élasticité, l'évolutivité et la résilience.

- **Services principaux d’Azure** : Exploration des services Azure de base, y compris les machines virtuelles, le stockage, les bases de données et les réseaux.
- **Sécurité et gouvernance** : Présentation des fonctionnalités de sécurité, des outils de gestion des identités et des stratégies de gouvernance pour garantir la conformité.
- **Modèles de coût et de tarification** : Compréhension des différents modèles de tarification et des outils disponibles pour gérer les coûts sur Azure.

Ce parcours de formation m’a été particulièrement utile pour le projet **Audit365 Manager**, notamment pour la configuration de **Microsoft Entra ID**, où j’ai pu enregistrer des applications et gérer les identités et les accès de manière sécurisée.

En complément de ce parcours, je me suis également entraîné en ligne sur le site de Microsoft, qui propose des évaluations pratiques pour se préparer à l’examen. De plus, j’ai utilisé ExamTopics pour m’entraîner avec des exemples de vraies questions d’examen, ce qui m’a aidé à mieux comprendre le format de l’examen et les types de questions posées.

Après avoir terminé la formation et les entraînements, j’ai passé l’examen et obtenu la certification **Microsoft Azure Fundamentals AZ-900**. Je suis très fier de cette réalisation, et l’entreprise l’a également bien accueillie. Étant en collaboration étroite avec Microsoft, l’entreprise encourage fortement ce type d’engagement et valorise les certifications qui renforcent nos compétences et notre expertise dans les technologies Microsoft.



FIGURE 4.1 – Certification AZ-900

4.1.2 Microsoft365 Fundamentals MS-900

j’ai suivi un parcours de formation similaire à celui de la première certification pour obtenir la **Microsoft 365 Fundamentals MS-900**. Ce parcours explore en profondeur les principaux concepts des services M365, les avantages de leur utilisation, ainsi que la gestion des utilisateurs et des identités. Les modules traitent également des aspects cruciaux tels que la sécurité, la conformité, la confidentialité et la gestion des abonnements.

Voici un résumé des principales thématiques abordées dans le parcours de Microsoft Learn :

- **Introduction à Microsoft 365** : Aperçu des services et des abonnements **M365**, avec un focus sur leurs avantages pour les entreprises.
- **Productivité et collaboration** : Exploration des outils de collaboration comme **Teams**, **SharePoint** et **OneDrive**, ainsi que leur rôle dans le **Modern Workplace**.
- **Gestion des identités et des accès** : Présentation des fonctionnalités de gestion des identités et des accès avec **Microsoft Entra ID** et les politiques de sécurité associées.
- **Sécurité, conformité et confidentialité** : Discussion des outils et stratégies pour assurer la sécurité des données, la conformité aux normes et la confidentialité des informations dans Microsoft 365.

Grâce à ce parcours, j'ai acquis les compétences nécessaires pour auditer et gérer les services Microsoft 365, ce qui s'est avéré essentiel pour le projet **Audit365 Manager**. De plus, il m'a permis de m'intégrer efficacement à l'équipe en utilisant des outils de collaboration tels que **Teams** pour les communications, **Planner** pour la gestion des tâches, et **SharePoint** pour l'intranet d'entreprise.

Avec quelques entraînements, j'ai également pu obtenir ma deuxième certification, la **MS-900**.



FIGURE 4.2 – Certification MS-900

4.1.3 Microsoft Azure Developer Associate AZ-204

J'ai récemment commencé la formation pour la certification **Microsoft Azure Developer Associate AZ-204** et je prévois de passer l'examen dans un futur proche. Cette formation s'est déjà avérée utile pour la conception de l'architecture du projet, en m'aidant à comprendre des concepts clés tels que les **web apps**, les **containers** et les **functions**.

4.2 Formation en DevOps

Dans cette section, je vais résumer la formations en **DevOps** que j'ai suivi pour renforcer mes compétences en **développement, intégration et déploiement** d'applications. Cette formation m'a permis de mieux comprendre et maîtriser les outils et pratiques **DevOps**, en particulier avec Azure DevOps. Bien que je n'aie pas encore eu l'occasion d'utiliser ces compétences dans le cadre du projet **Audit365 Manager**, car nous n'avons pas encore atteint la phase de déploiement et d'automatisation, je considère ces connaissances comme un atout précieux qui nous sera très utile dans un avenir proche.

4.2.1 Qu'est-ce que le DevOps

La formation **DevOps Foundations: Core Principles and Practices** couvre les concepts fondamentaux du DevOps, un ensemble de pratiques visant à automatiser et à intégrer les processus entre les équipes de développement et les opérations. Les principaux sujets abordés incluent les principes de collaboration, l'automatisation, l'intégration continue, la livraison continue, et la surveillance continue. Cette formation m'a aidé à comprendre comment ces pratiques peuvent améliorer la rapidité et la qualité des livraisons de logiciels, tout en réduisant les risques.

4.2.2 Construction d'application avec Azure DevOps

La formation **Build Applications with Azure DevOps** se concentre sur l'utilisation d'**Azure DevOps** pour planifier, développer et gérer des applications. Les principaux sujets incluent la gestion de projets avec Azure Boards, le contrôle de version avec Azure Repos, et la mise en place d'**intégrations continues (CI)** avec Azure Pipelines. Cette formation m'a permis de comprendre comment structurer et gérer efficacement des projets de développement en utilisant les outils **Azure DevOps**, même si je n'ai pas encore eu l'occasion de mettre ces compétences en pratique dans le cadre du projet actuel.

4.2.3 Déploiement d'application avec Azure DevOps

La formation **Deploy Applications with Azure DevOps** traite du déploiement et de la gestion des applications avec **Azure DevOps**. Les principaux sujets abordés incluent la création et l'exécution de pipelines de **déploiement continu (CD)**, la gestion des configurations de déploiement, et l'utilisation d'Azure Pipelines pour automatiser les déploiements. Cette formation m'a apporté une compréhension approfondie de la manière d'automatiser les processus de déploiement, de surveiller les applications en production, et de gérer les versions des applications de manière efficace et sécurisée. Ces compétences seront importantes pour la phase de déploiement de notre projet **Audit365 Manager**.

4.3 Participation à un événement Microsoft sur les nouvelles technologies IA

J'ai eu l'opportunité de participer à un événement organisé par Microsoft sur les nouvelles technologies en **intelligence artificielle (IA)**. Cet événement, intitulé "Créer des applications intelligentes avec Azure", s'est tenu au siège de Microsoft à Issy-les-Moulineaux. J'y ai passé la journée, profitant d'une session couvrante d'une variété de sujets sur les dernières avancées et applications

de l'IA, offrant des perspectives précieuses sur l'impact de l'IA dans divers secteurs. La participation à cet événement m'a permis de mieux comprendre comment les technologies IA peuvent être intégrées dans les solutions d'entreprise pour améliorer l'efficacité et l'innovation.

4.4 Formation sur les LLM (Large Language Models) chez Microsoft

Au cours de cet événement, j'ai également suivi une introduction sur les **Large Language Models (LLM)** proposée par Microsoft. La formation s'est déroulée en deux parties : une session théorique le matin et une session pratique l'après-midi, guidée par des experts leaders dans le domaine de l'IA.

4.4.1 Azure AI Studio

La première partie de la formation portait sur l'**Azure AI Studio**, un outil puissant pour créer, entraîner et déployer des modèles d'intelligence artificielle. **Azure AI Studio** offre une interface intuitive pour travailler avec des modèles de langage, facilitant ainsi leur intégration dans diverses applications. J'ai appris à utiliser cet outil pour développer des solutions AI personnalisées, optimiser les performances des modèles et déployer des solutions AI à grande échelle.

4.4.2 Prompt engineering

La deuxième partie de la formation était axée sur le **prompt engineering**, une technique essentielle pour interagir efficacement avec les **LLM**. Le **prompt engineering** consiste à concevoir des entrées (**prompts**) optimisées pour obtenir des réponses précises et pertinentes des modèles de langage. J'ai acquis des compétences en création et ajustement de **prompts** pour différents cas d'utilisation, améliorant ainsi la qualité et la fiabilité des réponses générées par les **LLM**.

4.4.3 Fine tuning

Enfin, la formation a couvert le **fine tuning** des modèles de langage, une étape cruciale pour adapter les **LLM** aux besoins spécifiques d'une organisation. Le **fine tuning** implique l'ajustement des modèles pré-entraînés sur des données spécifiques à un domaine ou à une application, afin d'améliorer leur performance et leur pertinence. J'ai appris les techniques de **fine tuning** et comment les appliquer pour optimiser les **LLM** pour des tâches précises, augmentant ainsi leur utilité et leur efficacité dans des contextes réels.

L'après-midi, j'ai eu l'occasion de mettre en pratique ces connaissances sous la direction des leaders experts présents. Durant cette session, j'ai eu l'opportunité de mettre en œuvre concrètement les concepts appris le matin. J'ai commencé par déployer des modèles IA avancés tels que **GPT-3.5** et **GPT-4** en utilisant Azure AI Studio. Cela m'a permis de comprendre les nuances et les paramètres nécessaires pour optimiser le déploiement de ces modèles. J'ai ensuite exploré les techniques de **prompt engineering**, où j'ai appris à structurer et formuler des prompts pour obtenir des réponses plus précises et pertinentes des modèles IA. Un aspect particulièrement enrichissant de la session pratique a été le **prompt flow**. En ajustant les prompts et en expérimentant avec différents scénarios, j'ai

pu voir comment de petites modifications pouvaient grandement influencer les résultats produits par les modèles. Cette expérience m'a également sensibilisé à l'importance du contexte et de la clarté dans la formulation des prompts.

Bien que cette formation n'était qu'une introduction au monde de l'**IA**, elle m'a équipé des connaissances et des compétences de base nécessaires pour commencer à exploiter le potentiel des **Large Language Models** dans des environnements professionnels. Cela renforce ma capacité à contribuer à des projets innovants dans le domaine de l'**intelligence artificielle**.

4.5 Portfolio Website

Au cours de mon temps libre, notamment les week-ends et plusieurs jours fériés du mois de mai, j'ai développé mon propre site web personnel, **www.amiar.fr**. Ce projet m'a permis de mettre en pratique les connaissances acquises en développement web et en déploiement sur le cloud **Azure**.

Grâce aux compétences acquises durant les différentes formations et expériences de stage, notamment en **Azure**, j'ai pu créer et déployer mon site web avec une grande facilité. Le site présente mon parcours, mes compétences et mes projets, offrant une vitrine professionnelle de mon travail.

Le développement du site m'a permis d'utiliser diverses technologies apprises pendant le stage, telles que :

- **HTML, CSS, et JavaScript** : Pour construire et styliser les pages web.
- **Azure Storage et Azure Custom Domain Name (CDN)** : Pour héberger et distribuer les contenus de manière efficace.
- **Azure Front Door** : Pour sécuriser et optimiser l'accès au site web.

En outre, les connaissances acquises à l'université ont également joué un rôle important dans ce projet. Les bases solides en programmation et en développement web que j'ai obtenues durant mes études m'ont permis d'aborder ce projet avec confiance et compétence.

Le processus de déploiement sur Azure a été grandement facilité par les connaissances acquises lors de mes formations et de mon travail sur le projet Audit365 Manager. J'ai déployé et configuré un **Azure Storage** puis activé une page web statique. J'ai ensuite déposé les fichiers de la page web dans un conteneur du stockage, rendant mon site disponible. Pour améliorer la sécurité et permettre un nom de domaine personnalisé (**CDN**), j'ai configuré un **Azure Front Door**, hébergeant ainsi mon site web sur **www.amiar.fr**.

Ce projet personnel m'a non seulement permis de consolider mes compétences techniques, mais aussi de créer un outil précieux pour ma carrière professionnelle. Il démontre ma capacité à appliquer les technologies apprises dans des projets concrets et à contribuer de manière significative à des initiatives technologiques.

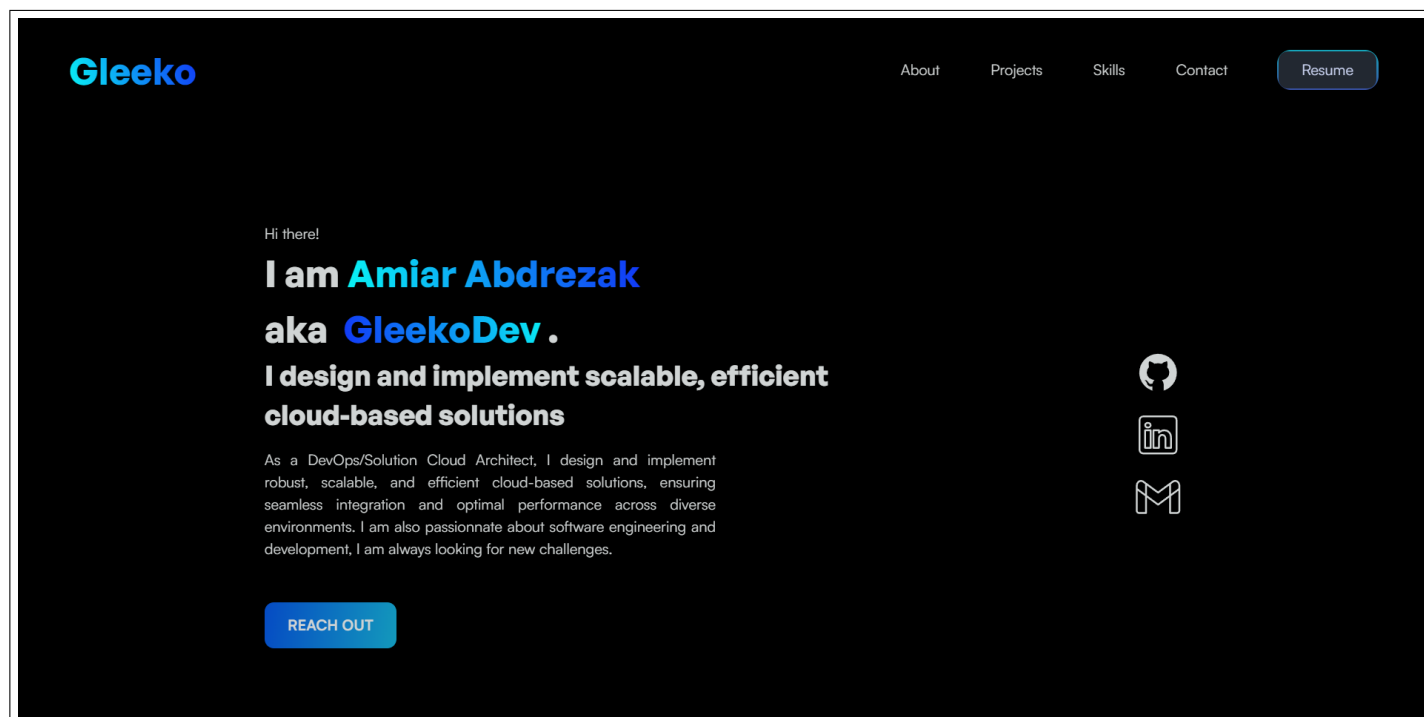


FIGURE 4.3 – Site Vitrine de Amiar Abdrezak

4.6 Résumé des apprentissages

Au cours de mon stage, j'ai eu l'opportunité d'acquérir et de consolider un large éventail de compétences techniques et pratiques. Voici un résumé des principales technologies et outils que j'ai appris et utilisés :

- **Azure** : Utilisation des services Azure tels que Azure Storage, Azure CDN, Azure Front Door, Web Apps, Functions et la gestion des ressources dans le portail Azure.
- **PowerShell** : Développement de scripts pour automatiser les tâches, gestion des configurations et réalisation des audits.
- **C#** : Programmation en C# pour le développement d'applications.
- **Microsoft 365** : Utilisation et gestion des services comme SharePoint, Teams, OneDrive et la suite Microsoft 365 dans son ensemble.
- **Planner et méthodes Agile** : Gestion des tâches et des projets, suivi des progrès à l'aide des tableaux Kanban, et application des principes Agile pour une gestion de projet efficace.
- **DevOps** : Compréhension des principes de DevOps, création et gestion des pipelines CI/CD, et utilisation de Azure DevOps pour le déploiement des applications.
- **HTML, CSS et JavaScript** : Développement et stylisation de pages web.
- **Fiddler** : Utilisation pour la surveillance et l'analyse des flux de données pour assurer la sécurité des applications.
- **Azure AI Studio** : Utilisation pour les projets impliquant des modèles de langage étendus (LLM).
- **LLM (Large Language Models)** : Formation sur les grands modèles de langage et leur application ainsi que sur le Prompt Engineering et le Fine Tuning.

Ces apprentissages m'ont permis de développer une compréhension approfondie des technologies cloud et des pratiques de DevOps, de renforcer mes compétences en développement web et d'acquérir une expérience précieuse en gestion de projets et en collaboration au sein d'une équipe. L'application des méthodes Agile, en particulier, a été essentielle pour structurer notre travail et garantir une communication fluide et une adaptation rapide aux changements. Ces compétences seront essentielles pour ma carrière future et m'aideront à contribuer efficacement à des projets technologiques complexes.

Conclusion et perspectives

La réalisation de ce stage au sein de l'Université Paris Est Créteil (**UPEC**) et de l'entreprise Exakis Nelite (**EXN**) m'a permis de développer des compétences techniques et professionnelles significatives. L'objectif principal de ce stage est de concevoir et de déployer une interface web pour l'audit et la gestion des environnements Microsoft 365 et Azure.

Pendant ce projet, j'ai eu l'opportunité de travailler avec diverses technologies avancées, notamment Microsoft Azure pour la configuration des ressources et le déploiement d'applications, PowerShell pour l'automatisation des tâches, C# pour le développement d'applications, Microsoft 365 pour la gestion de services comme SharePoint, Teams, et OneDrive, ainsi que DevOps pour la mise en place de pipelines CI/CD. Ces compétences seront des atouts précieux pour ma carrière future.

Les principales réalisations de ce stage incluent l'analyse des besoins, la compréhension des exigences et des objectifs du projet, le développement d'un Proof of Concept (POC) pour tester et valider les concepts, la personnalisation de l'outil avec l'adaptation du framework de sécurité et la modification de l'interface utilisateur, ainsi que la mise en place d'une architecture de déploiement sécurisée sur Azure.

Pour l'avenir, plusieurs axes d'amélioration et de développement peuvent être envisagés. Il est essentiel de poursuivre l'optimisation de l'outil pour améliorer ses performances et sa convivialité, tout en adoptant une approche de développement en itération. Chaque itération permettra d'intégrer des retours d'expérience et d'ajuster les fonctionnalités en conséquence. L'extension des fonctionnalités pour couvrir un plus large éventail de services Microsoft sera également une priorité. De plus, il est crucial de renforcer les mesures de sécurité pour s'assurer que l'outil reste conforme aux standards les plus stricts. Enfin, la participation à des formations continues permettra de rester à jour avec les dernières évolutions technologiques, garantissant ainsi que le projet évolue de manière agile et réactive face aux changements du secteur.

Au terme de ce travail, je peux affirmer que ce stage a été plein d'intérêt et d'apprentissage. J'ai pu travailler sur de nouveaux logiciels et découvrir le déroulement de la vie professionnelle à travers des réunions avec mon encadrant et les autres membres de l'entreprise. J'ai appris comment une équipe travaille pour réussir un projet, ce qui m'a donné une idée des situations que je vais confronter dans un futur proche en tant que DevOps/Architecte Solutions Cloud. Les compétences et les expériences acquises seront des fondations solides pour ma future carrière professionnelle.