

Tables des matières

Cahier des charges	2
1 - Introduction	2
1.1 - Objectif du document	2
1.2 - Terminologie	2
1.3 - Contexte du projet	3
1.4 - Portée du projet	3
1.5 - Public cible	4
1.6 - Description de l'existant	4
2 - Cahier des charges fonctionnel	4
2.1 - Besoins utilisateurs	4
Personas	4
Scénario d'utilisations	5
Exigences fonctionnelles	7
2.2 - Spécifications fonctionnelles	8
Flux de données	8
Contraintes	9
Détails des écrans	9
3 - Cahier des charges technique	9
3.1 - Contexte et environnement technique	9
3.2 - Architecture du système	10
Diagramme d'architecture	10
Architecture logicielle	11
Composants du système	11
3.3 - Spécifications techniques	11
Langages de programmation	11
API et services tiers	12
Base de données	12
Sécurité	12
Performances	13
Compatibilité et accessibilité	14
Infrastructure et déploiement	15
3.4 - Plan de tests	15
Stratégie de tests	16
Outils	16
3.5 - Plan de maintenance	17
3 - Annexes	17

Cahier des charges

1 - Introduction

1.1 - Objectif du document

L'objectif de ce document est de référencer et définir les exigences et les besoins du projet. Il apporte un cadre théorique sur la conception du projet tant sur le plan technique que sur fonctionnel (architecture, contraintes techniques, besoins fonctionnels ..).

Il va également déterminer des ressources tels que les détails des écrans / maquettes, les interactions utilisateurs, les interfaces externes qui seront utilisées..

1.2 -Terminologie

Database	Collections d'informations utilisées par le système
Responsables pédagogiques	Rang d'administrateur, responsables des étudiants, promotions mais aussi des intervenants et des accompagnateurs
Accompagnateurs	Responsables des étudiants et de leurs suivis durant la formation
Intervenants	Personnes externes à l'organisme responsable le temps de l'intervention de la promotion et des étudiants qui la compose
Intranet	Outil ou réseau local utilisé en interne par l'organisme de formation
Scénario nominal	Scénario de base des cas d'utilisation
Scénario alternatif	Scénario impliquant une potentielle erreur ou un procédé différent de celui attendu de base

1.3 - Contexte du projet

Le directeur d'un organisme de formation souhaite mettre en œuvre une refonte du système d'intranet de l'entreprise. Il précise que cet outil doit être divisé en deux parties distinctes pour les équipes pédagogiques et pour les étudiants.

Les objectifs de ce projet sont multiples. Ils concernent d'un point de vue de l'équipe pédagogique la centralisation de l'administration des étudiants et de leurs parcours (projets, promotions, compétences).

Pour les étudiants, un outil simple permettant de suivre leurs projets, d'en commencer de nouveau ou de consulter leurs informations.

L'entreprise étant un organisme de formation, ils ont des besoins stricts concernant le suivi des taux de présence des étudiants à l'école. Mais également des informations essentielles sur les différents intervenants et accompagnateurs pédagogiques. L'outil devra donc fournir les analyses et informations nécessaires.

1.4 - Portée du projet

L'organisme de formation compte 20 personnes dans l'équipe pédagogique et près de 400 élèves.

La refonte de l'outil intranet vise à faciliter l'administration des étudiants et de l'équipe pédagogique à travers un système centralisé.

L'utilisation de cet outil sera limité au personnel pédagogique ainsi qu'aux étudiants de l'organisme de formation et n'intègre pas d'utilisateurs externes.

Le système comprendra une base de données relationnelle qui aura pour but de stocker des informations concernant les responsables, les accompagnateurs, mais aussi les étudiants, leurs projets, leurs promotions et leurs compétences.

Les versions livrées durant le développement devront intégrer des besoins fonctionnels spécifiques selon les rôles des utilisateurs :

- **Les responsables pédagogiques** doivent être en capacité d'éditer les étudiants, les promotions, les unités de formation, les projets, les groupes d'étudiants pour les projets, les temps de logs des étudiants, des statistiques sur les promotions.
- **Les accompagnateurs** auront accès aux informations des promotions et étudiants desquelles ils sont responsables.
- **Les intervenants externes** pourront consulter des informations sur les promotions dans lesquelles ils interviennent et les étudiants qui les composent.

- **Les étudiants** eux auront un accès sur diverses informations les concernant, mais aussi sur leur unité de formation actuelle, un suivi de leurs projets (en cours, terminé) avec la possibilité de créer des groupes ou pas pour en commencer de nouveaux. Ils pourront également visualiser un tableau des compétences acquises au cours de leur formation.

1.5 - Public cible

Cet outil est interne et n'est destiné qu'aux étudiants et membres de l'équipe pédagogique de l'organisme de formation.

Cette équipe est composée de plusieurs rôles distincts :

- Les responsables pédagogiques
- Les accompagnateurs
- Les intervenants externes

1.6 - Description de l'existant

Il existe une version de l'intranet actuel. Cette dernière permet seulement aux responsables pédagogiques de visualiser la liste des promotions et leurs étudiants. Elle permet également aux étudiants de se connecter pour réaliser leur temps de logs.

Le site est actuellement hébergé sur ovh (.io)

2 - Cahier des charges fonctionnel

2.1 - Besoins utilisateurs

Personas

Dans la version primaire du projet, quatre types de personas sont définis selon les besoins et exigences.

Parmi eux nous pouvons retrouver (personas simplifié) :

Un étudiant :

Âgé de vingt à cinquante ans généralement.

A l'aise avec le milieu, les outils et équipements informatiques, il consulte plusieurs fois par semaine l'intranet pour commencer et suivre ses projets et sa progression.

Il souhaite pouvoir facilement créer un groupe pour rejoindre un projet, consulter ses compétences et différentes informations de son parcours (promotion, projet rendus ..).

Un responsable pédagogique :

Âgé de vingt à cinquante ans généralement.

Très à l'aise avec le milieu, les outils et équipements informatiques il utilise l'intranet comme outil de gestion et d'administration des étudiants et des autres points.

Il souhaite pouvoir gérer de manière centralisée les promotions et étudiants qui les composent, mais aussi obtenir des statistiques sur les classes, et consulter les informations des intervenants.

Un accompagnateur pédagogique :

Âgé de vingt à cinquante ans généralement.

Très à l'aise avec le milieu, les outils et équipements informatiques il utilise l'intranet comme outil de gestion des étudiants et promotions desquelles il est responsable.

Il souhaite consulter le suivi des étudiants et l'avancement de leurs projets et unités de formation, mais également noter les projets et en ajouter de nouveaux.

Un intervenant externe :

Âgé de vingt à cinquante ans généralement.

A l'aise avec le milieu informatique ou pas, il utilise l'intranet comme un outil permettant de consulter les promotions pour lesquelles il va intervenir.

Il souhaite avoir un outil simple (en cas de connaissances limitées dans le milieu informatique par exemple) pour consulter et obtenir des informations concernant les étudiants pour lesquels il prépare l'intervention.

Scénario d'utilisations

Les différents scénarios d'utilisation peuvent être définis à l'aide d'un diagramme de cas d'utilisation UML (voir annexes).

Voici des cas d'utilisation pour cet outil :

Cas d'utilisation : Authentification via le compte Google La Plateforme

Acteurs : Étudiants

Objectifs : N'autoriser la connexion qu'aux étudiants possédant un compte Google La Plateforme.

Précondition : Posséder un compte.

Postcondition : Utilisateur authentifié, accès à l'interface de l'intranet.

Scénario nominal :

1. L'étudiant accède à la page d'authentification
2. L'étudiant saisit ses identifiants de connexion
3. le système vérifie l'existence et la validité des identifiants
4. Le système donne accès à l'interface de l'intranet pour les étudiants.

Scénario alternatif :

Erreur d'authentification : Login ou mot de passe invalide.

Ce scénario démarre au point 3 du scénario nominal.

4. Le système affiche un message d'erreur.

Le scénario alternatif redémarre au point 1.

Cas d'utilisation : Ajout d'une nouvelle unité de formation

Acteurs : Responsables pédagogique

Objectifs : Créer une nouvelle unité de formation qui accueillera les futurs promotions.

Précondition : Être connecté au rang de responsables pédagogiques.

Postcondition : Création d'une nouvelle unité de formation.

Scénario nominal :

1. Le responsable accède à la page de création d'unité de formation
2. Le responsable renseigne les informations nécessaires à l'ajout de l'unité.
3. Le système vérifie la validité des informations saisies et l'unicité des noms des unités.
4. Le système ajoute en base de données la nouvelle unité de formation.

Scénario alternatif :

Erreur dans les informations renseignées : Le nom de l'unité est déjà utilisé.

Ce scénario démarre au point 3 du scénario nominal.

4. Le système affiche un message d'erreur.

Le scénario alternatif redémarre au point 2 en préservant les informations précédemment renseignées sauf le nom de l'unité et précise le champ responsable de l'erreur.

Cas d'utilisation : Consulter la liste des étudiants d'une promotion

Acteurs : Responsables pédagogique, accompagnateurs, intervenants

Objectifs : Consulter et visualiser les étudiants d'une promotion

Précondition : Être connecté au rang de responsable pédagogique / accompagnateur / intervenant. Être responsable de la promotion en cas d'accompagnateur ou intervenant.

Postcondition : Aucune.

Scénario nominal :

1. L'utilisateur accède à la liste des promotions auxquelles il a accès
2. L'utilisateur choisit une promotion
3. Le système affiche les étudiants de cette promotion
4. le système vérifie le type d'utilisateur connecté
5. Si l'utilisateur est un responsable pédagogique, un bouton permettant d'ajouter un élève sera disponible

Scénario alternatif :

Erreur dans les droits d'accès : Pas de droits pour cette promotion.

Ce scénario démarre au point 1 du scénario nominal.

2. Le système vérifie le rôle de l'utilisateur en question

3. Le système affiche un message d'erreur d'accès.

Le scénario alternatif redémarre au point 1.

Exigences fonctionnelles

Elles représentent la définition du processus du système pour répondre aux besoins.

Les exigences fonctionnelles ont été précisées par le directeur de l'organisme lors de la définition des besoins.

Les **étudiants** doivent avoir la possibilité de :

- Se connecter uniquement avec leur compte Google La Plateforme
- S'inscrire au projet
- Rendre un projet
- Consulter les projets disponibles
- Renseigner leurs informations personnelles
- Consulter leurs compétences

Pour l'**équipe pédagogique** :

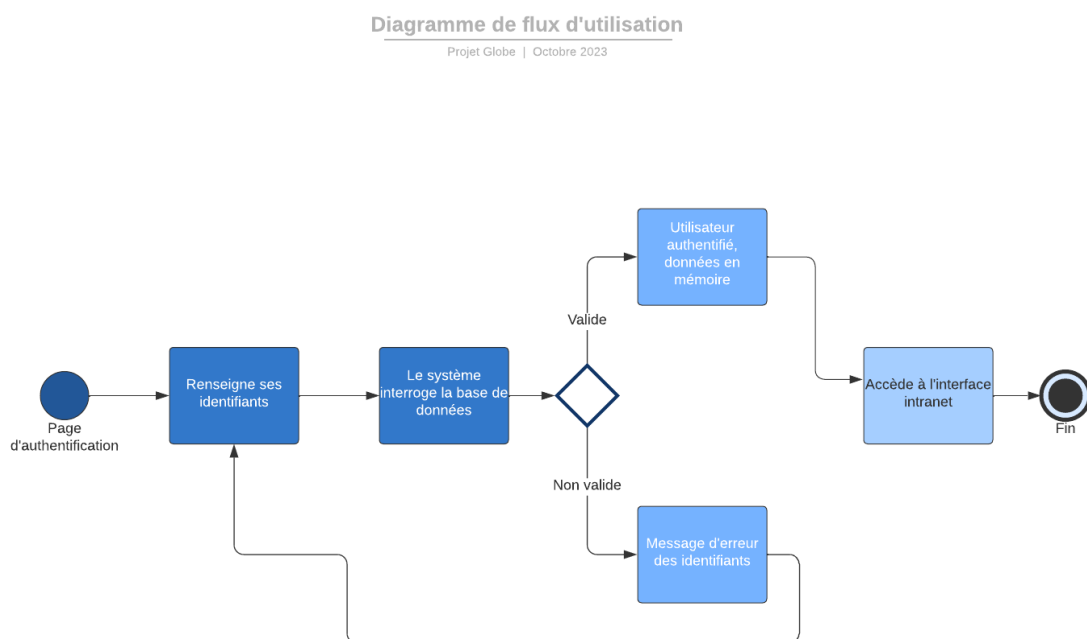
- Ajouter, Modifier, Récupérer et Supprimer une promotion
- Ajouter, Modifier, Récupérer et Supprimer un étudiant
- Ajouter, Modifier, Récupérer et Supprimer une unité de formation
- Ajouter, Modifier, Récupérer et Supprimer un projet
- Créer et valider des groupes pour les projets
- Suivre les projets en cours et les projets rendus
- Suivre les temps de log des étudiants
- Avoir des statistiques sur les promotions :
 - Les temps de logs
 - Les projets en cours, rendus
 - La moyenne dans chacune des compétences

2.2 - Spécifications fonctionnelles

Flux de données

Il est possible de mettre en place des diagrammes de flux d'utilisation ou d'état pour identifier le chemin des données et le comportement du système pour une fonctionnalité choisie.

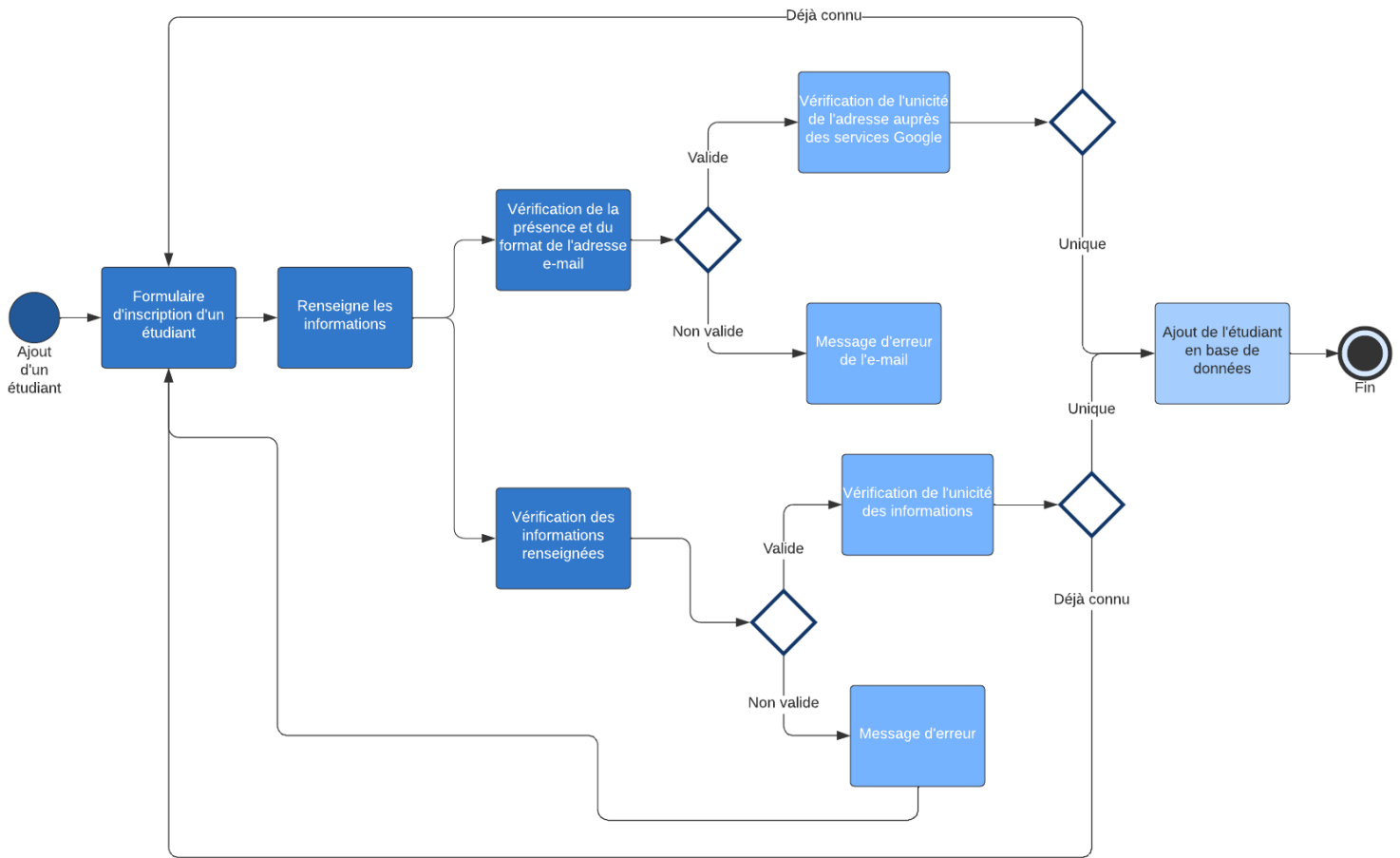
Authentification



Ajout d'un étudiant

Diagramme de flux d'utilisation

Projet Globe | Octobre 2023



Contraintes

Plusieurs types de contraintes sont à prendre en compte dans la réalisation du cahier des charges :

Les **contraintes fonctionnelles** sont définies dans les exigences fonctionnelles et les cas d'utilisation.

Les **contraintes de sécurité** :

- Un étudiant ne peut pas accéder aux formulaires d'édition et de création de projet, de promotion, d'étudiant. Un système de rôle et de permission doit être mis en place pour garantir la sécurité sur l'accès aux fonctions de l'outil.
- Concernant les étudiants, la connexion n'est autorisée qu'à partir d'un compte Google La Plateforme.

Les **contraintes ergonomiques**, l'outil doit être facile et agréable à utiliser. Il doit regrouper les principales fonctionnalités.

Les **contraintes économiques**, la mise en place et le développement de l'outil doivent respecter les coûts et prévisions.

Les **contraintes esthétiques**. Le projet doit respecter les besoins et exigences visuels (charte graphique potentielle) et plaire aux utilisateurs.

Les **contraintes de délai** : Pas précisées dans ce contexte.

Détails des écrans

Arborescence (à voir dans flux de navigation) , maquettes graphiques.

3 - Cahier des charges technique

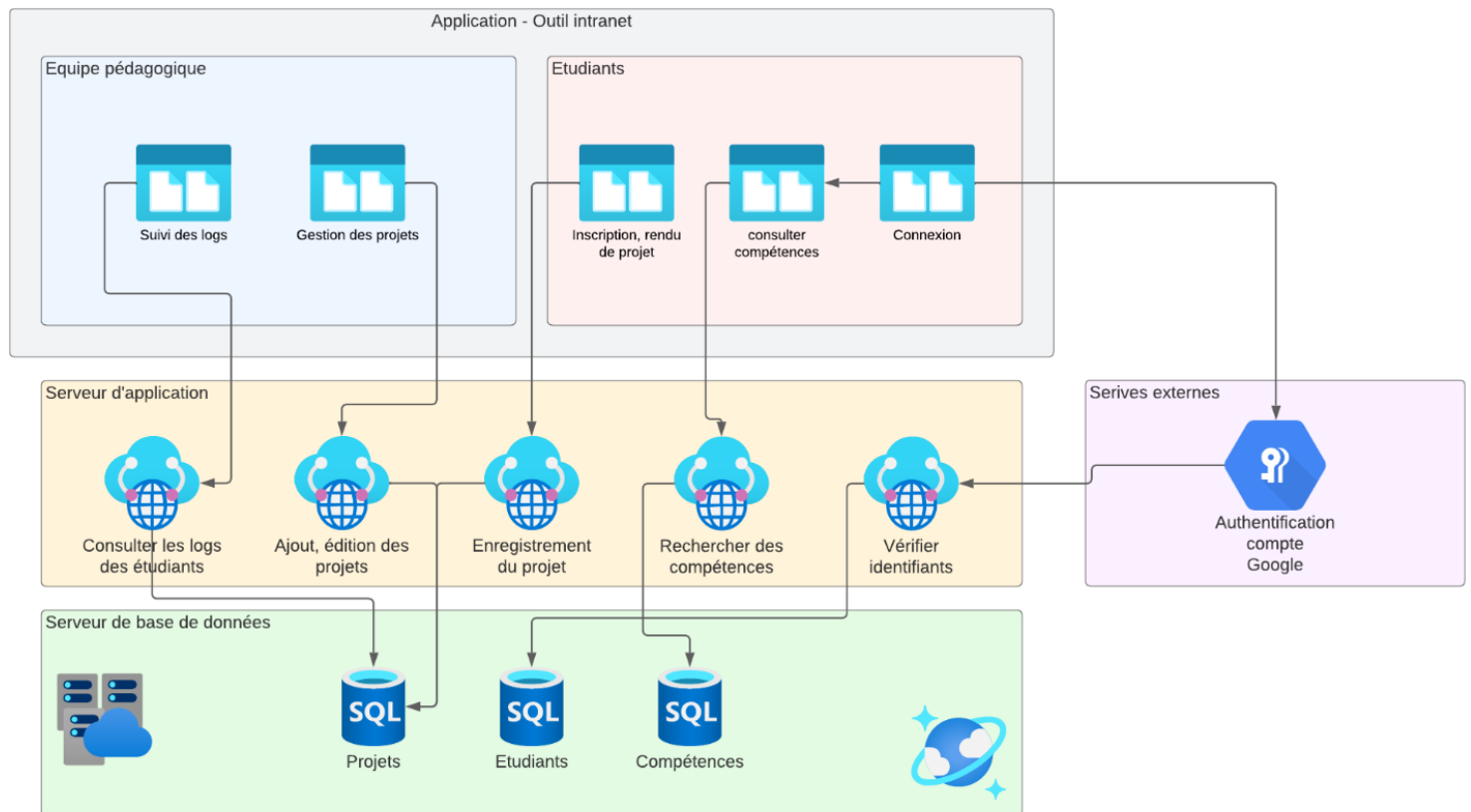
3.1 - Contexte et environnement technique

Contexte	Choix
Front-end	Outil actuel : HTML / CSS / JavaScript. Technologies préservées pour la refonte.
Back-end	PHP, préservé pour la refonte.
Base de données	Mysql
Serveur de développement	XAMPP
Administration de base de données	PhpMyAdmin
Environnement de développement intégré	Visual Studio Code
Design et intégration	Figma
Versionning	Git / Github

La version actuelle de l'outil est hébergée sur OVH et utilise une base de données relationnelle.

3.2 - Architecture du système

Diagramme d'architecture



Architecture logicielle

Il est possible de penser l'architecture du système en subdivisant les différents modules qui pourraient le composer :

Partie pédagogie	Ajout d'étudiants, de promotions, de membres de l'équipe, de projets. Suivi des logs. Suivi analytique des étudiants et compétences.
Partie étudiante	Suivi de ses projets et compétences. Système de logs
Module d'authentification	Module de connexion externe utilisant l'API Google Authentification

Composants du système

Diagramme de composants UML ?

3.3 - Spécifications techniques

Langages de programmation

Comme défini dans le contexte et l'environnement du projet, les langages de programmation sont les mêmes que ceux utilisés dans la version actuelle de l'outil intranet.

Pour la partie **front-end** : HTML / CSS / Javascript.

Ce choix s'explique par le souci de mettre en place une longue et périlleuse transition de langage de programmation et par l'expérience que possède l'équipe de développement pour ces technologies.

Pour la partie **back-end** : Php

Il est déjà en place sur le module actuel. Il s'adapte bien aux technologies Web (API, services tiers). L'équipe est à l'aise avec cette technologie.

Base de données : SQL

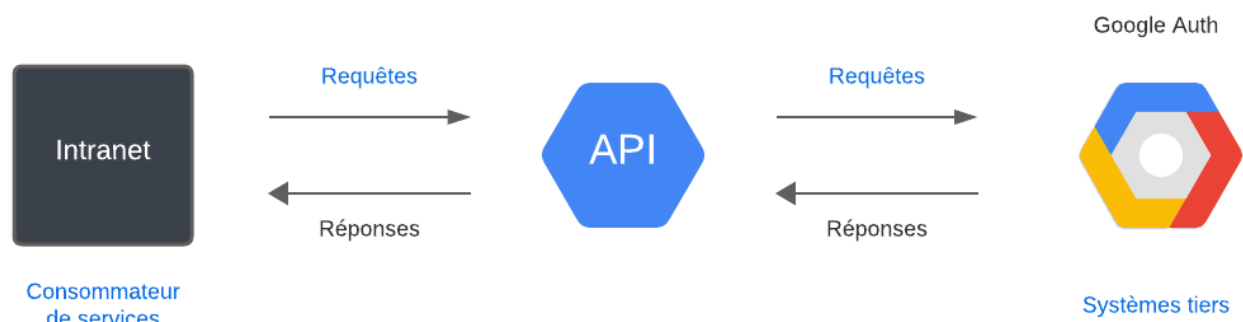
Fonctionne très bien avec Php. Équipe formée à SQL.

API et services tiers

Ce système nécessite la mise à jour de l'API Rest existante. Cette dernière utilise les requêtes HTTP/ HTTPS (get, post, put, delete) et retourne les réponses en format Json.

Elle interagit avec la service de base de données ou couche d'accès aux données afin de répondre à la demande envoyé depuis la couche de présentation. (interface visuel)

Elle communique également avec des services tiers notamment Google Authentification permettant d'assurer une connexion sécurisée et de filtrer seulement les étudiants inscrits.



Base de données

Le système exploite une **base de données relationnelle** à l'aide du langage SQL.
Elle est déjà composée de tables principales telles que les utilisateurs, et les logs.

Elle devra désormais contenir des informations portant sur les différentes unités de formation et leurs promotions. Mais aussi sur les compétences des étudiants, et définir des permissions en fonction des rôles.

Pour l'administration de cette base de données, nous utiliserons PhpMyAdmin.

Les fichiers SQL doivent correspondre strictement à la base de données locale, en cas de changement il doit être effectué via ces scripts.

Sécurité

Un premier point sécurité concernant l'aspect de **rôles et des permissions** :

Les utilisateurs sont séparés en deux principaux groupes, les étudiants et l'équipe pédagogique. L'équipe pédagogique est elle-même supervisée par plusieurs niveaux de rôles (responsables, accompagnateurs ..).

Les actions de chaque type d'utilisateur doivent être limitées à leur niveau de permission seulement.

Il est important de respecter la **restriction d'accès aux composants** pour lesquels les droits sont manquants.

Nous pourrions également soulever le fait qu'il s'agit de **données personnelles** sur les temps d'activités des étudiants, ces données entrent dans un cadre scolaire ou de formation et il est nécessaire qu'elles **soient protégées** de manière adéquate.

Pour ce qui est des accès à l'outil pour les étudiants, ces derniers ne peuvent y accéder qu'à l'aide d'un **compte Google La Plateforme**. Aucune autre adresse électronique différente de celles de La Plateforme n'ont d'accès sur la partie étudiante du projet.

Contre un dommage grave de la base de données, menant à la perte de données sensibles, il est possible de mettre en place des méthodes de **récupération et de restaurations de la base de données** à sa version fonctionnelle antérieure par exemple.

Il faut également veiller à utiliser les protocoles de sécurisation des échanges par réseau informatique. (TLS)

Performances

Les exigences de performance concernent la mise en œuvre des modèles de base de données.

Elles se composent des éléments suivants :

Temps de réponse d'une transaction (moyen et maximum)

Les temps de réponse des transactions visent à être optimisées et analysées dans le but d'être les plus performantes possibles. (Exemple, limité le FullScan)

Débits pour de données ou de transactions par secondes

En raison d'un potentiel haut débit de transactions sur l'outil en vue du nombre d'étudiants présent dans l'organisme de formation, les temps de réponse des transactions doivent être analysés et optimisés.

La capacité en nombre d'utilisateurs simultanés ou de transactions que le système peut supporter

Le système doit pouvoir supporter la totalité des transactions potentielles des étudiants. Le nombre d'étudiants doit être pris en compte ainsi que les analyses des transactions.

Les dégradations possibles si le système manque de ressources

Que se passerait-il pour les étudiants si le système ne supporte pas la charge ?

Certains critères de performances peuvent être mesurés comme :

La fiabilité :

- Probabilité d'un échec
- Taux d'occurrence de fautes
- Précision des calculs

La disponibilité :

- Temps moyen entre pannes: MTTF (Mean Time To Failure)
- Temps moyen de réparation: MTTR (Mean Time To Repair)
- Disponibilité = $MTTF / (MTTF + MTTR)$

Compatibilité et accessibilité

Pour l'**accessibilité** du système, il existe des normes appelées Web Content Accessibility Guidelines (WCAG).

Il faut viser à respecter ces normes au maximum afin de minimiser la distinction entre les utilisateurs encore plus en tant qu'outil pédagogique.

Selon les 4 principes de l'accessibilité (perceptible, utilisable, compréhensible et robuste) plusieurs règles ont été définis :

Principes	Règles	Niveau A	Niveau AA	Niveau AAA
Perceptible	Texte alternatif	✓		
	Media temporel	✓	✓	✓
	Adaptable	✓		
	Distinguable	✓	✓	✓
Utilisable	Accessible au clavier	✓		✓
	Assez de temps	✓		✓
	Crise d'épilepsie	✓		✓
	Navigable	✓	✓	✓
Compréhensible	Lisible	✓	✓	✓
	Prévisible	✓	✓	✓
	Input	✓	✓	✓
Robuste	Compatible	✓		

Pour ce qui est de la **compatibilité**, le système devra fonctionner sur les navigateurs principaux (Mozilla - Chrome - Edge).

Il devra être responsive ou du moins adaptés à plusieurs appareils :

- Desktop
- Mobile

La partie tablette n'étant pas privilégiée. Les étudiants sont amenés à travailler sur leurs ordinateurs personnels, tout comme l'équipe pédagogique. Seuls les intervenants extérieurs peuvent être amenés à utiliser l'outil depuis leurs téléphones.

En cas d'ajout d'extension dans l'outil, il faut vérifier la compatibilité de ces dernières avec les navigateurs.

Il faut également prendre en compte le souhait de mettre en place un système de fonctions analytiques afin d'obtenir des informations de suivi des étudiants.

Infrastructure et déploiement

diagramme d'infrastructure
diagramme déploiement uml ?

3.4 - Plan de tests

Le plan de test est un document décrivant l'étendue, l'approche, les ressources et le planning des activités de test prévues.

Il contient :

- Une fiche descriptive qui permet de comprendre pourquoi exécuter des tests et quel fonctionnement va être testé.
- Une description du scope ou out of scope.
- Une analyse des risques, qui permet de couvrir les tests exécutés.
- Les ressources matérielles et temporelles nécessaires.
- Les outils et environnements nécessaires.

Ce plan découle de trois documents :

- La stratégie de test / analyse des risques
- L'analyse fonctionnelle / User stories ou fonctionnalités
- Les documents de critères et exigences (techniques, sécuritaires ..)

Exemple de scénarios de tests :

Type de test	Description	Etape test	Résultat attendu	Statut
Sécurité	Check des règles de mot de passe	Renseigner un mot de passe respectant les règles	Le mot de passe doit être accepté	Réussi ou échec
Usabilité	Vérifier que les liens sont fonctionnels	Cliquer sur les liens de redirection en tant qu'utilisateur	Redirection sur la page cible	Réussi ou échec

Quatres étapes pour mettre en place le plan de tests :

- Mise en place des conditions préliminaires
- Exécution des tests

- Génération des rapports
- Evaluation du risque résiduel

Stratégie de tests

La stratégie de test a pour but de couvrir les périmètres suivants :

- Mesure de la performance de la solution : indicateurs
- Analyse de la performance de la solution : campagnes, scénarios, cas et jeux de données de l'ensemble des tests.
- Identification des limites de la solution : périmètres technique et organisationnel.
- Définition et application des actions correctives et évolutives.

Elle doit définir les raisons pour lesquelles les tests doivent être effectués et qui est chargé de les effectuer.

Une phase de test est suivie d'une validation (souvent appelée GO/NoGo) que ce soit pour passer à la suite du projet ou pour valider la fin d'un sprint et ainsi le passage en production de la version testée. Cette validation peut être définie dans la stratégie de tests.

Outils

Selenium est l'outil d'automatisation des tests open-source le plus populaire pour les applications web. Cet outil permet aux utilisateurs d'interagir et de manipuler des éléments HTML, d'ajuster et d'écrire des tests qui répondent à leurs besoins.

Il est compatible avec plusieurs navigateurs, flexible et accepte un grand nombre de langages de programmation.

JEST

Créée et maintenue par Facebook en 2014, Jest, rendu open-source, est une librairie de test JavaScript ayant énormément gagné en popularité depuis sa mise en libre circulation. Conçu pour fonctionner aussi bien sur du JavaScript côté navigateur (frontend) que côté serveur (backend), Jest a su convaincre par sa rapidité d'exécution des tests, son API complète et sa facilité d'installation. Sa documentation complète et bien maintenue en fait la librairie la plus populaire pour tous les différents tests js.

POSTMAN

Cet outil permet de tester les routes de son API en direct. Elle permet d'effectuer toutes sortes de requêtes vers des routes et de mettre en place des tests.

Ces tests sont effectués directement depuis l'interface du logiciel et son simple à mettre en œuvre. Ils offrent un premier aperçu de tests rapidement.

SmartBear Ready API

SmartBear Ready API est conçu pour tester les API. Ready API fonctionne avec les méthodologies agiles et DevOps. En plus des tests fonctionnels d'API, cet outil offre la virtualisation Web d'API, les tests de performance d'API et les tests de sécurité d'API.

3.5 - Déploiement

3.6 - Plan de maintenance

3 - Annexes

A faire : Sprint Scrum / Présentation / Finir maquette / Diagramme de déploiement et composants