

SOMMAIRE

[Présentation de l'initiative 1](#_Toc196047055)

[PRÉSENTATION DU PROJET 2](#_Toc196047056)

[DESCRIPTION DE L'EXISTANT 4](#_Toc196047057)

[CONCURRENCE 5](#_Toc196047058)

[CIBLE 5](#_Toc196047059)

[OBJECTIFS GRAPHIQUES ET ERGONOMIQUES 6](#_Toc196047060)

[ATTENTES FONCTIONNELLES ET TECHNIQUES 8](#_Toc196047061)

[PRESTATIONS ATTENDUES 10](#_Toc196047062)

[PLANNING PREVISIONEL 11](#_Toc196047063)

[ENVELOPPE BUDGÉTAIRE 12](#_Toc196047064)

[LIENS DES COMPTES 13](#_Toc196047065)

[CONCLUSION 14](#_Toc196047066)

Cahier des Charges

Plateforme de Gestion des Anciens Sujets Universitaires

## Présentation de l'initiative

**Nous sommes deux étudiants en Licence 3 Informatique, spécialité Génie Logiciel, à l'Université de Technologie d'Abidjan (UTA).**

À travers ce projet, nous souhaitons concevoir une **plateforme numérique de gestion et de consultation des anciens sujets d'examens** spécifiquement pour notre université.

Nous avons constaté que **l'accès aux anciens sujets à l'UTA est peu optimisé et difficile pour les étudiants**, car ces documents sont actuellement archivés au format papier dans les départements. Cette situation limite la préparation des étudiants aux examens et n'exploite pas le potentiel des outils numériques.

**Notre objectif** est donc de contribuer à la modernisation et à la digitalisation des ressources pédagogiques de notre établissement à travers cette initiative étudiante.

* Rôle du projet dans la stratégie universitaire :

Cette plateforme de gestion des anciens sujets d'examens s'inscrit dans la volonté de l'université UTA de digitaliser l'accès aux ressources pédagogiques et d'améliorer la préparation des étudiants aux examens. Elle permettra de centraliser les contenus historiques tout en assurant un suivi de leur utilisation

PRÉSENTATION DU PROJET:

1. Contexte

Le contexte actuel à l'UTA montre un accès limité aux ressources pédagogiques antérieures, en particulier aux anciens sujets d'examens. Ces documents sont conservés de manière physique, rendant leur consultation contraignante et souvent incomplète pour les étudiants.

1. OBJECTIFS DU FUTUR SITE WEB

**Créer une plateforme web de type ressource pédagogique permettant aux étudiants d'accéder aux anciens sujets d'examens et corrigés, et aux administrateurs de gérer efficacement ces ressources.**

* Permettre aux étudiants de consulter et télécharger les anciens sujets et corrigés via un espace personnel
* Offrir aux administrateurs la possibilité d'ajouter, gérer et mettre à jour ces ressources facilement
* Sécuriser l'accès via un système d'authentification et de gestion des droits
* Mettre en place une recherche avancée par matière, année, enseignant, type de sujet
* Centraliser et organiser ces sujets sur une seule plateforme consultable en ligne
* Optimiser la préparation académique des étudiants
* Participer à la digitalisation des services universitaires

1. Objectifs quantitatifs

* Volume de trafic visé : 5000 visites mensuelles
* Objectif de téléchargements : 2500 téléchargements par mois
* Taux d'adoption par les étudiants : 80% des étudiants actifs sur la plateforme durant la première année
* Couverture de 90% des matières enseignées dans les différentes filières

1. Objectifs qualitatifs

* Faciliter la préparation aux examens pour les étudiants
* Assurer une traçabilité des documents consultés et téléchargés
* Améliorer la qualité des révisions par l'accès aux anciens sujets et corrigés
* Favoriser l'autonomie des étudiants dans leur apprentissage

## DESCRIPTION DE L'EXISTANT

1. État actuel de la gestion des sujets d'examens

* Absence de plateforme centralisée pour l'accès aux anciens sujets
* Distribution non systématique et inégale des sujets entre étudiants
* Stockage physique ou dispersé sur différents espaces numériques
* Absence de traçabilité des consultations et téléchargements

1. Ressources disponibles

* Base de données existante des sujets d'examens au format PDF
* Serveurs universitaires pour l'hébergement potentiel
* Domaine web institutionnel disponible pour l'intégration
* Logo et charte graphique de l'université utilisables

1. Ressources humaines

* Équipe technique composée de **Dabo Ali (Chef de projet/Front-end) et AGOH CHRIS (Back-end)**
* Personnel administratif pouvant être formé à la gestion de la plateforme

## CONCURRENCE

Quelques universités ivoiriennes et africaines proposent des solutions similaires, mais elles sont souvent limitées à des groupes Facebook ou des archives papiers numérisées sans réelle plateforme dédiée.

## CIBLE

1. Personne principal

**Nom** : Mariam, 21 ans

**Statut** : Étudiante en 3ᵉ année Informatique

**Besoins** : Retrouver rapidement les anciens sujets pour se préparer aux examens

**Compétences** : Maîtrise basique des outils numériques

**Appareil utilisé** : PC portable, smartphone

**Contraintes** : Connexion internet irrégulière, peu de temps pour chercher les sujets à la BU

**Attentes** : Trouver facilement des sujets triés par matière et année, avec possibilité de téléchargement

1. Personne secondaire

**Nom** : M. Koné

**Statut** : Responsable Pédagogique

**Besoins** : Publier les sujets et consulter les statistiques des téléchargements

## OBJECTIFS GRAPHIQUES ET ERGONOMIQUES

1. Attentes en matière de design

* Interface moderne, épurée et professionnelle
* Navigation intuitive adaptée aux habitudes des étudiants
* Expérience utilisateur fluide sur tous les appareils (responsive design)

1. Éléments graphiques

* Code couleur : Utilisation des couleurs institutionnelles de l'UTA
* Logo : Intégration du logo officiel de l'université
* Police de caractère : [À déterminer selon la charte graphique de l'UTA]
* Éléments visuels : Icônes intuitives pour représenter les fonctionnalités

1. Organisation visuelle

* Dashboard étudiant : Message de bienvenue, accès rapide aux fonctionnalités, statistiques personnelles
* Page de recherche : Filtres avancés clairement visibles et faciles à utiliser
* Visualisation des documents : Lecteur PDF intégré avec options de zoom et navigation

## ATTENTES FONCTIONNELLES ET TECHNIQUES

1. Arborescence du site

A diagram of a structure

AI-generated content may be incorrect.

1. Description fonctionnelle

* Espace étudiant
* Système d'authentification sécurisé via email et matricule
* Génération de mot de passe envoyé par email lors de l'inscription
* Moteur de recherche avancée (filtres par matière, année, type)
* Visualisation intégrée des documents PDF
* Système de téléchargement avec traçabilité
* Gestion du profil utilisateur et historique de téléchargements
* Espace administrateur
* Interface d'ajout et de gestion des sujets avec upload de fichiers
* Système de catégorisation des sujets (matière, année, type)
* Gestion des comptes utilisateurs (validation, suspension)
* Tableau de bord statistique avec indicateurs clés
* Configuration des filières et matières
* Spécifications techniques

|  |  |  |  |
| --- | --- | --- | --- |
| Côté Client (Frontend) | Côté Serveur (Backend) | Base de Données | Outils Divers |
| React JS | Django (Python) | PostgreSQL | Git & GitHub (versioning) | |
| Figma (UI Design) | Django REST Framework |  | Postman (tests API) |
| Tailwind CSS / Bootstrap | JWT pour l’authentification |  | Netlify (front) |
|  |  |  | Railway (back) |

## PRESTATIONS ATTENDUES

1. Liste des prestations

* Développement du backend en Django/DRF
* Développement du frontend en React/Vite
* Conception et intégration responsive de l'interface utilisateur
* Mise en place de la base de données PostgreSQL
* Configuration du système d'authentification JWT
* Mise en place du stockage sécurisé des documents
* Tests de performance et de sécurité
* Documentation technique et utilisateur
* Formation de l'équipe administrative (2-3 personnes)
* Déploiement sur les serveurs de production
* Maintenance et support technique (durée à définir)

## PLANNING PREVISIONEL

👉 [Lien vers le Diagramme de Gantt ou capture d'écran](#)

|  |  |  |  |
| --- | --- | --- | --- |
| Phase | Responsable(s) | Durée estimée | Statut |
| Création du dépôt GitHub | Dabo Ali | 1 jour |  |
| Rédaction du cahier des charges | Dabo Ali | 1 jours |  |
| Conception des maquettes (Figma) | Dabo Ali & Py |  |  |
| Mise en place environnement | Dabo Ali & Py |  |  |
| Définition des modèles et migrations | Py |  |  |
| Développement de l’API (auth, sujets) | Py |  |  |
| Intégration du front React (connexion, tableau de bord) | Dabo Ali |  |  |
| Intégration API / Front | Py |  |  |
| Tests et ajustements | Dabo Ali & Py |  |  |
| Déploiement | Dabo Ali & Py |  |  |

## ENVELOPPE BUDGÉTAIRE

Le projet étant porté bénévolement par deux étudiants, le budget est réduit et basé sur :

* **Nom de domaine** : 10 000 FCFA / an
* **Hébergement web** : environ 20 000 FCFA / an
* **Développement** : réalisé par nous-mêmes
* **Éventuelles extensions** : à définir selon les besoins futurs

**Budget estimé global : 30 000 FCFA**

## LIENS DES COMPTES

* **Github :**

AGOH CHRIS🡪 <https://github.com/AgohChris>

DABO ALI 🡪 <https://github.com/alsondab>

Dépôt Git du projet 🡪 <https://github.com/AgohChris/OldTopic.git>

GitHub de DABO ALI 🡪 <https://github.com/alsondab>

GitHub de AGOH CHRIS 🡪 <https://github.com/AgohChris>

* **email:**

AGOH CHRIS🡪 [agohchris90@gmail.com](mailto:agohchris90@gmail.com)

DABO ALI 🡪 [alsondab9@gmail.com](mailto:alsondab9@gmail.com)

* **Linkedin:**

[AGOH CHRIS](https://www.linkedin.com/in/chris-agoh-811b6526a/)

DABO ALI

* **Num Tel:**

AGOH CHRIS🡪 0778748602

DABO ALI 🡪 0710145864

# CONCLUSION

Au terme de ce cahier des charges, il apparaît clairement que la "Plateforme de Gestion des Anciens Sujets Universitaires" répond à un besoin réel et urgent au sein de l'Université de Technologie d'Abidjan. Ce projet, porté bénévolement par deux étudiants en Licence 3 Informatique, représente une initiative stratégique pour moderniser l'accès aux ressources pédagogiques de l'établissement.

La plateforme proposée transformera fondamentalement la façon dont les étudiants se préparent aux examens en leur offrant un accès centralisé, organisé et sécurisé aux anciens sujets d'examens. Elle s'inscrit parfaitement dans la démarche de digitalisation des services universitaires tout en répondant aux attentes des utilisateurs principaux que sont les étudiants et le personnel administratif.

Les technologies choisies (React JS pour le frontend, Django pour le backend et PostgreSQL pour la base de données) garantissent une solution robuste, évolutive et conforme aux standards actuels du développement web. Le planning prévisionnel établi permet d'envisager une réalisation efficace du projet, tandis que l'enveloppe budgétaire reste maîtrisée grâce à l'engagement bénévole de l'équipe de développement.

Cette plateforme constituera non seulement un outil précieux pour la communauté universitaire de l'UTA, mais pourra également servir de modèle pour d'autres établissements confrontés à des problématiques similaires. À terme, ce projet contribuera significativement à l'amélioration de la qualité pédagogique et de l'expérience estudiantine à l'UTA.

Les prochaines étapes consistent désormais à mettre en œuvre ce cahier des charges en lançant concrètement le développement selon le planning établi, tout en maintenant une communication régulière avec les parties prenantes pour assurer que la solution finale réponde pleinement aux attentes exprimées.

**ANNEXES 1- Glossaire technique**

***API REST*** *: Application Programming Interface de type Representational State Transfer. Interface de programmation standardisée permettant la communication entre le frontend et le backend via des requêtes HTTP (GET, POST, PUT, DELETE).*

***Back-end*** *: Partie du développement web qui gère le traitement des données, la logique métier et l'accès à la base de données, invisible pour l'utilisateur final.*

***Base de données*** *: Système organisé permettant de stocker, gérer et récupérer des informations de manière structurée.*

***Dashboard*** *: Tableau de bord offrant une vue synthétique des informations et fonctionnalités principales d'une application.*

***Django*** *: Framework web Python de haut niveau encourageant le développement rapide et pragmatique avec une conception propre.*

***Django REST Framework (DRF)*** *: Extension de Django spécialisée dans la création d'API RESTful.*

***Front-end*** *: Partie visible d'un site web ou d'une application avec laquelle l'utilisateur interagit directement.*

***Git*** *: Système de contrôle de version distribué permettant de suivre les modifications du code source.*

***GitHub*** *: Plateforme web d'hébergement et de gestion de développement de logiciels utilisant Git.*

***JWT (JSON Web Token)*** *: Standard ouvert pour la création de tokens d'accès sécurisés utilisés dans l'authentification.*

***PostgreSQL*** *: Système de gestion de base de données relationnelle open-source avancé.*

***Postman*** *: Plateforme API permettant de concevoir, tester et documenter des API.*

***Railway*** *: Plateforme moderne de déploiement et d'hébergement d'applications.*

***React JS*** *: Bibliothèque JavaScript open-source développée par Facebook pour créer des interfaces utilisateur dynamiques.*

***Responsive design*** *: Approche de conception web visant à adapter automatiquement l'affichage d'une page web à la taille de l'écran du terminal utilisé.*

***Serveur*** *: Système informatique offrant des services aux clients (stockage, traitement, etc.).*

***Tailwind CSS*** *: Framework CSS utilitaire permettant de construire rapidement des designs personnalisés.*

***Vite*** *: Outil de build front-end nouvelle génération offrant un serveur de développement plus rapide.*

***Versioning*** *: Gestion des différentes versions d'un logiciel ou d'un document au cours de son développement.*