



الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث العلمي



Ministère de L'enseignement Supérieur et de la Recherche Scientifique

Université 20 Aout 1955 Skikda  
Faculté Des Sciences  
Département D'informatique

جامعة 20 أوت 1955 سكيكدة  
كلية العلوم  
قسم الإعلام الآلي

Projet de fin : d'étude dans le cadre de l'article ministériel 1275 en vue de l'obtention du Diplôme Master informatique

Spécialité : Réseaux et Systèmes Distribués

### THÈME

## **CoopEase : Une Application Web Intelligente de Gestion Coopérative des Ressources Pédagogiques**

Réalisé par :

- Laifa Hocine Chiheb
- Boutamine Amani

Encadré par :

- Dr. Boulehouache Soufiane

Co-encadré par :

- Dr. Bougueroua Salah

Année Universitaire 2024 – 2025

## Remerciement

Ce travail n'aurait pas été possible sans l'aide d'Allah, qui nous a accordé la patience et le courage nécessaires tout au long de la préparation de ce mémoire de licence.

Nous tenons à exprimer notre profonde gratitude à **Monsieur Boulehouache Soufiane** et à **Monsieur Bougueroua Salah** pour leur accompagnement précieux, leur soutien constant, leurs conseils avisés et leur grande bienveillance durant l'élaboration de ce travail.

Nous remercions également toute la communauté universitaire pour son soutien et son accompagnement, ainsi que toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce mémoire et à la réussite de cette belle année universitaire.

Enfin, nous adressons nos remerciements les plus sincères à nos parents, nos frères et sœurs pour leurs efforts, leur soutien indéfectible et leurs encouragements. Nous prions Dieu de les protéger et de les préserver de tout mal.

## المخلص

يعرض هذا العمل تصميم وتطوير تطبيق ويب ذكي لإدارة الموارد التربوية المشتركة في المؤسسات التعليمية. استجابةً لقيود الطرق التقليدية (اليديوية أو شبه المؤتمتة)، نقترح حلاً مبتكراً يعتمد على تقنيات حديثة مثل Django (بايثون) و Google OR-Tools. يتيح النظام إنشاء جداول زمنية تلقائياً، ومعالجة التصريحات بالغياب وعدم التوفر، وتسهيل التنسيق بين الأساتذة والطلاب والإداريين. كما يحمل هذا المشروع بعداً ريادياً، حيث يهدف إلى النشر الواسع في المدارس والجامعات والمؤسسات الخاصة. وتكمن القيمة المضافة في الأتمتة الذكية، وسهولة الاستخدام، وإمكانية إدخال بيانات الطلاب مباشرة عبر ملفات Excel، مما يُسهّل عملية بدء التشغيل.

## الكلمات المفتاحية

إدارة الموارد، المؤسسات التعليمية، الجدول الزمني، التحسين، Django، OR-Tools، الذكاء الاصطناعي، تطبيق ويب، الأتمتة.

## Résumé

Ce mémoire présente la conception et la réalisation d'une application web intelligente dédiée à la gestion coopérative des ressources pédagogiques dans les établissements éducatifs. Face aux limites des approches traditionnelles (manuelles ou semi-automatisées), nous proposons une solution innovante s'appuyant sur des technologies modernes comme Django (Python) et Google OR-Tools. L'application permet la planification automatique des emplois du temps, la déclaration d'indisponibilités, la gestion des absences, ainsi qu'une coordination fluide entre enseignants, étudiants et gestionnaires. Ce travail s'inscrit également dans une démarche entrepreneuriale, avec une vision de déploiement à large échelle, notamment dans les écoles, universités et établissements privés. L'innovation repose sur l'automatisation intelligente, la simplicité d'usage et l'importation directe des données à partir de fichiers Excel, facilitant ainsi l'intégration des utilisateurs.

## Mots-clés

Gestion des ressources, établissement éducatif, emploi du temps, optimisation, Django, OR-Tools, intelligence artificielle, application web, automatisation.

## **Abstract**

This thesis presents the design and implementation of an intelligent web application for cooperative resource management in educational institutions. In response to the limitations of traditional (manual or semi-automated) approaches, we propose an innovative solution built with modern technologies such as Django (Python) and Google OR-Tools. The platform automates timetable generation, handles unavailability and absence declarations, and ensures smooth coordination between teachers, students, and administrators. This project also takes an entrepreneurial dimension, aiming at large-scale deployment across schools, universities, and private institutions. The innovation lies in intelligent automation, user-friendly interfaces, and direct Excel file import, streamlining user integration.

## **Keywords**

Resource management, educational institutions, timetable, optimization, Django, OR-Tools, artificial intelligence, web application, automation.

## Table des matières

Liste des figures

Liste des tableaux

<u>المخلص</u>	<u>3</u>
<u>Résumé</u>	<u>3</u>
<u>Abstract</u>	<u>4</u>
<u>Introduction Générale</u>	<u>13</u>
<u>Chapitre 1 : Contexte, Problématique et Objectifs</u>	<u>14</u>
<u>1.1 Contexte général</u>	<u>14</u>
<u>1.2 Définitions et concepts clés</u>	<u>14</u>
<u>1.2.1. Ressource partagée</u>	<u>14</u>
<u>1.2.2. Gestion coopérative</u>	<u>14</u>
<u>1.2.3. Conflit d'allocation</u>	<u>15</u>
<u>1.2.4. Optimisation</u>	<u>15</u>
<u>1.2.5. Emploi du temps</u>	<u>15</u>
<u>1.3 Problématique observée</u>	<u>16</u>
<u>1.3.1 Problème de l'emploi du temps</u>	<u>16</u>
<u>1.4 Limites des approches classiques</u>	<u>17</u>
<u>1.5 Problématique de recherche</u>	<u>17</u>
<u>1.6 Vers une solution intelligente et coopérative</u>	<u>18</u>
<u>1.7 Gestion coopérative des ressources</u>	<u>18</u>
<u>1.8 Durabilité et efficacité dans l'utilisation des ressources partagées</u>	<u>19</u>
<u>1.9 Systèmes d'information collaboratifs</u>	<u>19</u>
<u>1.10 Apport de l'intelligence artificielle dans la gestion des ressources</u>	<u>20</u>
<u>1.11 Objectifs du mémoire</u>	<u>20</u>
<u>1.12 Méthodologie adoptée</u>	<u>21</u>
<u>1.13 Conclusion</u>	<u>21</u>

<u>Chapitre 2 : Analyse, Étude des Approches Existantes et Choix Techniques</u>	<u>22</u>
<u>2.1 Introduction</u>	<u>22</u>
<u>2.2 État de l'art des solutions existantes</u>	<u>23</u>
<u>2.2.1 Méthodes traditionnelles</u>	<u>23</u>
<u>2.2.2 Outils semi-automatisés</u>	<u>23</u>
<u>2.2.3 Systèmes basés sur l'intelligence artificielle</u>	<u>24</u>
<u>2.2.4 Vers des plateformes intelligentes et collaboratives</u>	<u>24</u>
<u>2.3 Comparaison entre la Création Manuelle et Automatique d'un Emploi du Temps</u>	<u>24</u>
<u>2.4 Analyse des besoins fonctionnels et non fonctionnels</u>	<u>25</u>
<u>2.4.1 Identification des utilisateurs</u>	<u>25</u>
<u>2.4.2 Besoins fonctionnels</u>	<u>26</u>
<u>2.4.3 Besoins non fonctionnels</u>	<u>27</u>
<u>2.5 Analyse des contraintes du système</u>	<u>27</u>
<u>2.5.1 Contraintes de ressources</u>	<u>28</u>
<u>2.5.2 Contraintes temporelles</u>	<u>28</u>
<u>2.5.3 Contraintes organisationnelles</u>	<u>28</u>
<u>2.5.4 Contraintes techniques</u>	<u>29</u>
<u>2.5.5 Contraintes liées à l'usage collaboratif</u>	<u>29</u>
<u>2.6 Choix méthodologiques et modélisation</u>	<u>29</u>
<u>2.6.1 Approche orientée objet</u>	<u>30</u>
<u>2.6.2 Modèle UML</u>	<u>30</u>
<u>2.6.2.1 Points fort d'UML</u>	<u>30</u>
<u>2.6.2.2 Points faible d'UML</u>	<u>30</u>
<u>2.6.3 Raison du choix de Google OR-Tools</u>	<u>31</u>
<u>2.6.4 Démarche de modélisation du problème de planification</u>	<u>31</u>
<u>2.6.4.1 Langage de modélisation formalise CSP</u>	<u>32</u>
<u>2.7 Architecture technique de la solution</u>	<u>32</u>
<u>2.7.1 Architecture en couches</u>	<u>32</u>

2.7.2 Technologies utilisées	33
2.7.3 Schéma global de l'architecture	33
2.7.4 Sécurité et accessibilité	35
2.7.4.1 Sécurité du système	35
2.7.4.2 Accessibilité de la plateforme	36
2.8 Conclusion	36
Chapitre 3 : Conception du système	38
3.1 Introduction	38
3.2 Présentation du langage UML	39
3.3 Diagramme de cas d'utilisation	40
3.3.1 Objectifs du diagramme de cas d'utilisation	40
3.3.2 Acteurs identifiés dans notre projet	40
3.3.3 Cas d'utilisation principaux	41
3.4 Diagramme de classes	42
3.4.1 Objectifs du diagramme de classes	43
3.4.2 Principales classes identifiées	43
3.4.3 Présentation du diagramme	43
3.5 Diagramme de séquence	44
3.5.1 Objectifs du diagramme de séquence	44
3.5.2 Scénario 1 : Authentification	45
3.5.3 Scénario 2 : Choix du groupe par un étudiant	46
3.5.4 Scénario 3 : Déclaration des indisponibilités par un enseignant	47
3.5.5 Scénario 4 : génération d'un emploi du temps	49
3.5.6 Scénario 5 : Réservation d'une salle par un enseignant	50
3.5.7 Scénario 6 : Annonce d'une absence par un enseignant	52
3.6 Conception de la base de données	54
3.6.1 Objectifs de la base de données	54
3.7 Conception de l'interface utilisateur	54
3.7.1 Principes de conception	55

3.8 Conclusion du chapitre	56
Chapitre 4 : Implémentation et validation	56
4.1 Introduction	56
4.2 Environnement de développement	56
4.2.1 Langage de programmation	57
4.2.1.1 Python	57
4.2.1.2 HTML	57
4.2.1.3 CSS	58
4.2.1.4 JavaScript	58
4.2.2 Framework web	58
4.2.2.1 Django	59
4.2.2.2 Bootstrap	59
4.2.3 intelligence artificielle	59
4.2.3.1 Moteur d'optimisation : Google OR-Tools	60
4.2.4 Base de données	60
4.2.5 Outils complémentaires	60
4.2.5.1 PyCharm	60
4.2.5.2 Github	61
4.2.5.3 Star UML	61
4.3 Implémentation des fonctionnalités clés	62
4.3.1 Gestion des utilisateurs et des rôles	62
4.3.2 Déclaration des contraintes et indisponibilités	62
4.3.3 Génération automatique de l'emploi du temps	62
4.3.4 Réservation de salles et suivi	63
4.4 Sécurité, accessibilité	63
4.4.1 Sécurité de l'application	63
4.4.2 Accessibilité	64
4.5 Présentation de l'utilisation et de la manipulation du site à travers quelques interface	64
4.5.1 Interface d'authentification	64

<a href="#">4.5.1.1 Interface d'authentification d'administration</a>	<a href="#">65</a>
<a href="#">4.5.1.2 Interface d'inscription d'un gestionnaire établissement</a>	<a href="#">65</a>
<a href="#">4.5.1.2.1 Interface de formulaire à remplir par gestionnaire</a>	<a href="#">66</a>
<a href="#">4.5.2 les pages d'accueil</a>	<a href="#">67</a>
<a href="#">4.5.3 Consultation de l'emploi du temps par les étudiants</a>	<a href="#">68</a>
<a href="#">4.5.4 Consultation de l'emploi du temps par l'enseignant</a>	<a href="#">68</a>
<a href="#">4.5.5 Interface de gestion des disponibilités enseignant</a>	<a href="#">69</a>
<a href="#">4.5.6 Interface des demandes des disponibilités des enseignants</a>	<a href="#">70</a>
<a href="#">4.5.7 Interface d'annonce d'une absence par un enseignant</a>	<a href="#">71</a>
<a href="#">4.5.8 Consultations la disponibilité des salles</a>	<a href="#">72</a>
<a href="#">4.5.9 Réserveation de salle</a>	<a href="#">73</a>
<a href="#">4.5.9.1 Réserveation des salles toujours disponible dans emploi du temps</a>	<a href="#">73</a>
<a href="#">4.5.9.2 Réserveation des salles occasionnellement disponible</a>	<a href="#">73</a>
<a href="#">4.5.9.3 Interface de gestion des Réserveation des salles</a>	<a href="#">74</a>
<a href="#">4.5.10 interface d'historique des réservations des salles</a>	<a href="#">74</a>
<a href="#">4.5.11 interface d'affichages pour étudiant</a>	<a href="#">75</a>
<a href="#">4.5.12 Interface de demande d'intégration à un groupe par un étudiant</a>	<a href="#">76</a>
<a href="#">4.5.13 Interface du profil</a>	<a href="#">77</a>
<a href="#">4.5.14 Interface des modules</a>	<a href="#">78</a>
<a href="#">4.5.15 Interface liste des enseignants</a>	<a href="#">79</a>
<a href="#">4.5.16 Interface d'Equipements disponible</a>	<a href="#">81</a>
<a href="#">4.5.17 Tableau de bord de l'administrateur pour la création des groupes</a>	<a href="#">81</a>
<a href="#">4.5.18 Tableau de bord de l'administrateur génération des emplois du temps</a>	<a href="#">82</a>
<a href="#">4.5.19 Tableau de bord de l'administrateur pour l'affectation des étudiants aux groupes</a>	<a href="#">83</a>
<a href="#">4.5.20 Tableau de bord de l'administrateur</a>	<a href="#">83</a>
<a href="#">4.6 Conclusion</a>	<a href="#">83</a>

**LISTE DES FIGURES**

<a href="#">Figure 1.1 Emploi du temps Université skikda master 2 RSD [4]</a>	16
<a href="#">Figure 1.2 Conflit de réservation simultanée</a>	17
<a href="#">Figure 1.3 Processus de gestion des ressources</a>	19
<a href="#">Figure 2.1 Schéma global de l'architecture</a>	33
<a href="#">Figure 3.1 diagramme de cas d'utilisation utilisateurs</a>	42
<a href="#">Figure 3.2 diagramme de classes</a>	44
<a href="#">Figure 3.3 diagramme de séquence d'authentification</a>	45
<a href="#">Figure 3.4 diagramme de séquence d'inscription d'un gestionnaire</a>	46
<a href="#">Figure 3.5 diagramme de séquence Choix du groupe par un étudiant</a>	47
<a href="#">Figure 3.6 diagramme de séquence Déclaration des indisponibilités par un enseignant</a>	48
<a href="#">Figure 3.7 diagramme de séquence génération d'un emploi du temps</a>	49
<a href="#">Figure 3.8 diagramme de séquence Réservation d'une salle par un enseignant</a>	51
<a href="#">Figure 3.9 diagramme de séquence Annonce d'une absence par un enseignant</a>	53
<a href="#">Figure 4.1 pages d'authentification enseignant, étudiant, responsable département</a>	65
<a href="#">Figure 4.2 pages d'authentification administrateurs</a>	65
<a href="#">Figure 4.3 pages d'inscription d'un gestionnaire</a>	66
<a href="#">Figure 4.4 pages de formulaire d'inscription</a>	66
<a href="#">Figure 4.5 pages d'accueil enseignant</a>	67
<a href="#">Figure 4.6 pages d'accueil responsable département</a>	67
<a href="#">Figure 4.7 pages d'accueil étudiant</a>	68
<a href="#">Figure 4.8 l'emploi du temps d'étudiant</a>	68
<a href="#">Figure 4.9 l'emploi du temps de l'enseignant</a>	69
<a href="#">Figure 4.10 l'interface de gestion d'indisponibilités d'enseignants</a>	70
<a href="#">Figure 4.11 l'interface de confirmation d'indisponibilités d'enseignants</a>	71
<a href="#">Figure 4.12 l'interface d'annoncer une absence d'un enseignant</a>	71
<a href="#">Figure 4.13 l'interface pour consulte les salles disponibles</a>	72
<a href="#">Figure 4.14 Interface de réservation des salles toujours disponibles</a>	73
<a href="#">Figure 4.15 Interface de réservation des salles occasionnellement disponible</a>	73
<a href="#">Figure 4.16 Interface de la gestion des réservations des salles</a>	74
<a href="#">Figure 4.17 Interface de l'historique des réservations de salle</a>	75

<u>Figure 4.18 Tableau d'affichage pour étudiant</u>	<u>76</u>
<u>Figure 4.19 Interface de choix du groupe par un étudiant</u>	<u>77</u>
<u>Figure 4.20 Interface du profil d'étudiant</u>	<u>77</u>
<u>Figure 4.21 Interface du profil enseignant</u>	<u>78</u>
<u>Figure 4.22 Interface modules par d'étudiant</u>	<u>78</u>
<u>Figure 4.23 Interface modules par enseignant</u>	<u>79</u>
<u>Figure 4.24 Interface liste des enseignants</u>	<u>79</u>
<u>Figure 4.25 Interface liste des enseignants par étudiant</u>	<u>80</u>
<u>Figure 4.26 Interface liste d'équipements</u>	<u>81</u>
<u>Figure 4.27 Interface de tableau de bord de l'administrateur et déclenchement de création des groupes</u>	<u>82</u>
<u>Figure 4.28 Interface de tableau de bord de l'administrateur et déclenchement de la génération automatique des emplois du temps</u>	<u>82</u>
<u>Figure 4.29 Interface de l'administrateur pour faire affectation des étudiants a des groupes</u>	<u>83</u>

## **LISTE DES TABLEAUX**

<u>Tableau 2.1 Comparaison entre la méthode Manuelle et Automatique</u>	<u></u>
<u>Tableau 2.2 Synthèse des Technologies Utilisées par Composant du Système</u>	<u></u>

## **Introduction Générale**

Dans un contexte mondial marqué par une croissance continue des besoins et une limitation des ressources disponibles, la gestion efficace de ressources partagées devient un enjeu crucial. Qu'il s'agisse d'équipements, de salles, de données ou de services, ces ressources sont souvent limitées, utilisées par plusieurs entités, et soumises à des contraintes temporelles et spatiales. Leur coordination simultanée, leur allocation équitable, ainsi que leur exploitation optimale représentent des défis majeurs. À ces contraintes s'ajoutent des exigences croissantes en matière de mobilité, de coordination multi-acteurs et de coopération continue. Ces facteurs rendent les approches classiques de gestion souvent rigides, coûteuses et inadaptées aux réalités complexes des organisations actuelles.

Face à ces problématiques, l'émergence des technologies web et des outils d'intelligence artificielle permet d'envisager des solutions intelligentes et collaboratives. Celles-ci offrent de nouvelles perspectives pour automatiser, optimiser et centraliser les processus de gestion des ressources. Grâce à des systèmes dynamiques, accessibles en temps réel et capables de s'adapter aux besoins évolutifs des utilisateurs, il devient possible de renforcer la traçabilité des ressources, de simplifier leur réservation, et d'améliorer leur utilisation globale.

C'est dans cette optique que s'inscrit le présent travail, qui a pour objectif la conception et la réalisation d'une application web intelligente dédiée à la gestion coopérative des ressources dans les établissements éducatifs. L'application vise à faciliter la planification, l'optimisation et la coordination de ressources telles que les salles, les équipements et les emplois du temps, en tenant compte des contraintes pédagogiques et des besoins des différents acteurs (enseignants, départements, administration). L'approche repose sur l'intégration de mécanismes d'optimisation basés sur l'outil OR-Tools de Google, combinée à une interface web intuitive et accessible.

Ce mémoire s'articule autour de trois chapitres complémentaires :

- Le **chapitre 1** présente le cadre général du projet, en exposant le contexte, la problématique, les objectifs, ainsi que les concepts théoriques fondamentaux liés à la gestion coopérative des ressources.
- Le **chapitre 2** est consacré à l'analyse de l'existant, à l'état de l'art des technologies disponibles, et aux choix technologiques retenus pour la mise en œuvre de la solution.
- **Le chapitre 3** porte sur la conception fonctionnelle du système, à travers la modélisation des cas d'utilisation, des classes et des interactions. Il comprend les diagrammes UML, l'identification des acteurs, et l'architecture technique globale de la solution.
- **Le chapitre 4** décrit la mise en œuvre et l'évaluation de l'application web, en détaillant les technologies utilisées, le développement des composants clés, ainsi que les résultats obtenus en matière de planification et d'utilisation des ressources.

Une conclusion générale viendra clore ce travail en mettant en lumière les contributions réalisées, les limites rencontrées et les perspectives d'évolution du système proposé.

# **Chapitre 1 : Contexte, Problématique et Objectifs**

## **1.1 Contexte général**

Dans les établissements éducatifs tels que les universités, les écoles supérieures ou les instituts de formation, la gestion des ressources partagées constitue un défi majeur. Ces ressources incluent notamment les salles de cours, amphithéâtres, laboratoires, équipements informatiques, ainsi que les ressources humaines telles que les enseignants. Elles sont généralement en nombre limité et font l'objet de sollicitations multiples, souvent simultanées, de la part de plusieurs départements ou filières. Cette situation impose une coordination fine et rigoureuse dans le temps et dans l'espace afin d'éviter les conflits d'utilisation et d'assurer une distribution équitable et efficiente.

Toutefois, l'évolution des structures pédagogiques, la diversification des programmes, et la complexité croissante des contraintes logistiques rendent les méthodes traditionnelles de gestion de plus en plus inadéquates. Le recours à des processus manuels ou semi-automatisés devient non seulement fastidieux, mais également source d'erreurs et d'inefficacités. Ce constat souligne la nécessité de solutions plus intelligentes, automatisées et collaboratives pour répondre aux enjeux actuels.

## **1.2 Définitions et concepts clés**

Afin de bien circonscrire le cadre de ce travail, il est important de définir les concepts centraux mobilisés :

### **1.2.1. Ressource partagée**

Toute entité, matérielle ou humaine, dont l'usage est commun à plusieurs utilisateurs dans un contexte spatio-temporel défini. [1]

### **1.2.2. Gestion coopérative**

Au cours des vingt dernières années, les entreprises ont recherché de nouvelles sources d'efficacité face à la saturation des marchés. L'essor

des technologies de l'information a intensifié les interactions entre les clients et les entreprises, impactant également les coopératives, qui doivent repenser leur identité et leur pertinence. Dans ce contexte, deux modèles théoriques sont proposés : le Nouveau Paradigme Coopératif (NPC) et le modèle d'équilibre coopératif. Le NPC repose sur quatre stratégies clés la congruence des valeurs, la propriété psychologique, la loyauté et l'engagement, et la cocréation de valeur visant à générer des avantages concurrentiels durables. Le modèle d'équilibre coopératif, quant à lui, propose un cadre de gestion cohérent avec les principes coopératifs. Enfin, l'ouvrage met en lumière le risque de perte d'identité chez certaines coopératives et suggère que ces modèles peuvent contribuer à renforcer leur spécificité et leur efficacité face aux mutations du marché. [2]

### **1.2.3. Conflit d'allocation**

La gestion des conflits découlant d'écarts dans l'affectation des ressources est un défi courant dans la coordination des programmes. Il est crucial de naviguer dans ces situations avec une approche stratégique qui assure l'équité et maintient la cohésion d'équipe. L'allocation des ressources, c'est-à-dire le processus de répartition des ressources disponibles entre divers projets ou départements, conduit souvent à des désaccords lorsque les ressources sont limitées. Donc il faut comprendre les besoins de chaque membre de l'équipe, en hiérarchisant les tâches et en favorisant un environnement de communication ouverte. En corrigeant ces écarts de manière proactive. [3]

### **1.2.4. Optimisation**

L'optimisation est le processus de rendre quelque chose aussi bon en recherchant la meilleure solution possible, selon un ensemble de contraintes et d'objectifs prédéfinis.

### **1.2.5. Emploi du temps**

Agencement planifié des cours et les TD et TP dans le temps et dans l'espace, en tenant compte des contraintes humaines et matérielles.

Timetable

Date: 25/10/2024

M2-RSD S1

	8h:00-9h:30	9h:30-11h:00	11h:00-12h:30	12h:30-14h:00	14h:00-15h:30	15h:30-17h:00
Samedi			D-Option3/G2/S13/Seddari	D-Option3/G1/S13/Seddari		
Dimanche	C-SysEmMob Bourriche S19	C-CSysCoop Boulaiche S19	D-SysEmMob/G1/S18/Bourriche D-CSysCoop/G2/S16/Boulaiche	D-CSysCoop/G1/S24/Boulaiche D-SysEmMob/G2/S16/Bourriche		
Lundi	P-RésAdhoc/G1/SP1/Boulaiche	P-JMXGAD/G1/SP4/Kara P-RésAdhoc/G2/SP1/Boulaiche	C-RésAdhoc Boulaiche S24	P-JMXGAD/G2/SP4/Kara		
Mardi	C-JMXGAD Boulehouache S23	D-JMXGAD/G2/TPA3/Kara	C-ErgoWeb Foughali S15	D-ErgoWeb/G1/S15/Foughali	D-JMXGAD/G1/SP-B3/Kara D-ErgoWeb/G2/S16/Foughali	
Mercredi		D-MFormsDist/G2/S24/Zebiri		C-MFormsDist Layadi S24		
Jeudi		C-Anglais LalaouaY Enligne				

Figure 1.1 Emploi du temps Université skikda master 2 RSD [4]

### 1.3 Problématique observée

Dans de nombreux établissements, les outils traditionnels utilisés pour la planification des emplois du temps et l'allocation des ressources pédagogiques montrent leurs limites. Plusieurs problématiques sont fréquemment observées :

- Une insuffisance ou une mauvaise répartition des ressources disponibles.
- Des conflits fréquents dans l'attribution des créneaux horaires et des lieux d'enseignement.
- Une surcharge administrative lors de la planification semestrielle.
- Un manque de transparence dans les processus de décision.
- Une insatisfaction générale des enseignants et des étudiants vis-à-vis des plannings proposés.

Ces constats appellent à un changement de paradigme dans la manière dont sont planifiées et gérées les ressources pédagogiques.

#### 1.3.1 Problème de l'emploi du temps

La gestion des emplois du temps consiste à affecter les enseignants

et les salles à des cours dans l'espace de temps en satisfaisant un ensemble de contraintes. [5]

## 1.4 Limites des approches classiques

Les outils couramment employés, comme les feuilles Excel, les calendriers manuels ou les logiciels généralistes, ne permettent pas de faire face efficacement à la complexité croissante des besoins. Ces méthodes restent rigides, peu évolutives, et ne garantissent ni l'optimisation globale ni la coordination automatique entre les services. Elles s'avèrent par ailleurs peu adaptées à une gestion collaborative, dynamique et transparente des ressources.

## 1.5 Problématique de recherche

Ainsi, la problématique centrale à laquelle ce mémoire tente de répondre et formulée comme suit :

**Comment concevoir une application web intelligente permettant de gérer de manière coopérative et optimisée les ressources partagées d'un établissement éducatif, tout en respectant les contraintes spécifiques des différents acteurs impliqués ?**

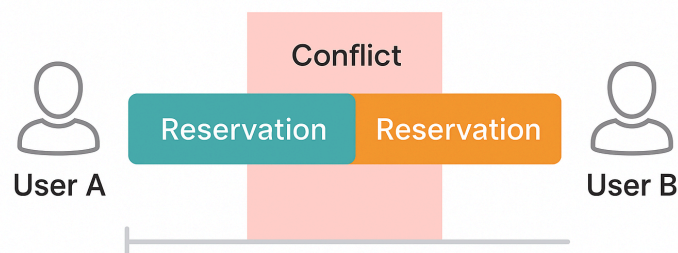


Figure 1.2 Conflit de réservation simultanée

## **1.6 Vers une solution intelligente et coopérative**

Pour dépasser les limites des méthodes traditionnelles, une solution basée sur l'optimisation algorithmique intégrée à une application web centralisée constitue une approche efficace. L'utilisation d'outils comme Google OR-Tools permet de gérer automatiquement les contraintes de planification tout en réduisant les conflits d'allocation. Une telle plateforme favorise une coordination fluide entre les acteurs grâce à une interface collaborative et accessible.

## **1.7 Gestion coopérative des ressources**

La gestion coopérative des ressources implique la collaboration de plusieurs acteurs pour gérer des ressources partagées, telles que des salles de classe, du matériel pédagogique, ou encore des enseignants et des emplois du temps. Dans le cadre des établissements éducatifs, cette gestion se doit d'être fluide et adaptable aux besoins de chaque acteur tout en respectant des contraintes de disponibilité. [6]

L'utilisation d'une application web intelligente permet de centraliser les informations, d'optimiser la planification et de gérer en temps réel les disponibilités de chaque ressource. Les algorithmes d'optimisation et d'intelligence artificielle sont utilisés pour minimiser les conflits et maximiser l'utilisation des ressources partagées.

### PROCESSUS DE GESTION D'UNE RESSOURCE PARTAGÉE

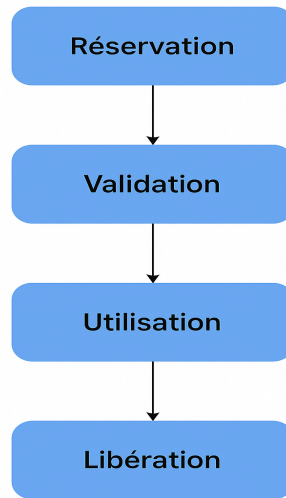


Figure 1.3 Processus de gestion des ressources

## 1.8 Durabilité et efficacité dans l'utilisation des ressources partagées

Dans un établissement éducatif, la durabilité des ressources est primordiale. La gestion doit éviter les gaspillages, réduire les périodes d'inoccupation et garantir que chaque ressource soit utilisée de manière optimale. Par exemple, les salles de cours doivent être allouées de façon à ce qu'aucune ne reste inutilisée durant les horaires clés, tandis que les équipements pédagogiques doivent être mis à disposition des enseignants selon les besoins.

L'introduction de systèmes intelligents permet de prendre des décisions basées sur des données collectées, en utilisant des algorithmes pour ajuster les horaires, optimiser les ressources et proposer des solutions équilibrées pour tous les acteurs.

## 1.9 Systèmes d'information collaboratifs

Les systèmes d'information collaboratifs sont essentiels dans un environnement où plusieurs utilisateurs doivent interagir avec un même système pour partager et gérer des ressources. [7]

Dans le cadre de notre projet, le système collaboratif sera conçu pour permettre aux enseignants, administrateurs et étudiants de consulter, réserver et gérer les ressources en temps réel.

Des fonctionnalités comme la réservation de salle, l'affichage des emplois du temps et la gestion des équipements pédagogiques seront proposées via une interface centralisée. La collaboration sera facilitée par des outils de suivi des ressources disponibles et de notification.

## **1.10 Apport de l'intelligence artificielle dans la gestion des ressources**

L'intelligence artificielle, au sens large, joue un rôle fondamental dans la transformation des méthodes de gestion des ressources éducatives. En particulier, l'usage de bibliothèques d'optimisation comme **Google OR Tools** s'inscrit dans le domaine de l'intelligence artificielle, exploitant des algorithmes avancés de résolution de contraintes pour trouver les meilleures solutions possibles.

Grâce à Google OR-Tools, il devient possible de modéliser de manière fine les multiples contraintes inhérentes aux emplois du temps (disponibilités des enseignants, types de salles, durée des cours, etc.) et de générer automatiquement des plannings optimisés. Cette approche permet non seulement de réduire les conflits et les erreurs humaines, mais aussi d'améliorer l'équité et l'efficacité dans l'utilisation des ressources.

En intégrant ces mécanismes dans une plateforme web coopérative, notre projet illustre l'un des apports concrets de l'IA dans la modernisation des processus administratifs éducatifs.

## **1.11 Objectifs du mémoire**

Dans le cadre de ce projet, les objectifs suivants sont visés :

- Étudier les pratiques actuelles de gestion des ressources au sein des établissements éducatifs.
- Identifier les insuffisances et limites des solutions existantes.
- Proposer une architecture web intelligente adaptée à la gestion coopérative.

- Intégrer un moteur d'optimisation basé sur Google OR-Tools pour automatiser la génération des emplois du temps.
- Évaluer l'efficacité de la solution proposée à travers des cas d'usage concrets.

### **1.12 Méthodologie adoptée**

La méthodologie suivie dans ce travail repose sur une démarche projet, structurée en plusieurs phases :

1. Revue de littérature et étude des approches existantes.
2. Définition des besoins fonctionnels et techniques.
3. Conception de l'architecture logicielle de l'application.
4. Développement du système, intégrant le moteur OR-Tools.
5. Tests et évaluation de la solution à travers des scénarios réels.

### **1.13 Conclusion**

Ce premier chapitre a permis d'introduire le contexte général du projet, en soulignant les enjeux et les difficultés liées à la gestion des ressources partagées dans les établissements éducatifs. Il a également formulé clairement la problématique de recherche et présenté les objectifs ainsi que la méthodologie adoptée.

L'introduction des technologies d'optimisation basées sur l'intelligence artificielle ouvre des perspectives prometteuses pour répondre aux besoins actuels en matière d'efficacité, de transparence et de coopération.

Le chapitre suivant sera consacré à l'étude des approches existantes et aux choix techniques effectués pour la conception de notre application.

# Chapitre 2 : Analyse, Étude des Approches Existantes et Choix Techniques

## 2.1 Introduction

La conception d'un système de gestion coopérative des ressources partagées, notamment dans le cadre des établissements éducatifs, nécessite une analyse approfondie des approches existantes ainsi qu'une réflexion rigoureuse sur les choix techniques à adopter. Ce chapitre vise à établir les fondations méthodologiques et technologiques de notre projet.

Dans un contexte où les besoins en planification deviennent de plus en plus complexes en raison de la diversité des ressources (salles, enseignants, équipements), de la multiplicité des contraintes (horaires, disponibilités, types d'activités), et de l'exigence croissante en matière d'efficacité et de transparence il est indispensable d'évaluer les outils existants, leurs capacités, ainsi que leurs limites.

Ce chapitre se propose donc de :

- Présenter l'état de l'art des solutions informatiques utilisées dans la gestion des ressources pédagogiques ;
- Comparer les différentes approches (manuelles, semi-automatisées, automatisées par l'IA) ;
- Justifier les choix méthodologiques retenus pour notre application ;
- Expliquer les technologies sélectionnées pour la mise en œuvre de la solution ;
- Décrire l'architecture logique et les aspects liés à la sécurité et à l'accessibilité du système.

À travers cette analyse, nous chercherons à démontrer en quoi une solution web intelligente, s'appuyant sur des outils d'optimisation avancés comme Google OR-Tools, constitue une réponse pertinente et innovante face aux problématiques identifiées dans le chapitre précédent.

## **2.2 État de l'art des solutions existantes**

La gestion des ressources partagées au sein des établissements éducatifs, en particulier la planification des emplois du temps, a longtemps reposé sur des méthodes manuelles ou semi-automatisées. Toutefois, l'évolution des besoins, la complexité croissante des contraintes et la nécessité d'une gestion efficace ont conduit au développement de solutions informatiques spécialisées. Cette section présente un panorama des approches actuellement utilisées, en soulignant leurs principes, leurs avantages et leurs limites. [8]

### **2.2.1 Méthodes traditionnelles**

Les méthodes traditionnelles consistent généralement à établir les emplois du temps à l'aide de tableaux papier ou de feuilles de calcul (comme Microsoft Excel). Bien que flexibles et facilement accessibles, ces outils ne sont pas conçus pour gérer des contraintes complexes telles que les chevauchements, les disponibilités multiples ou l'optimisation des ressources.

Les principales limites observées sont :

- Une forte dépendance à l'expertise humaine ;
- Une difficulté à intégrer plusieurs contraintes simultanément ;
- Un risque élevé d'erreurs et de conflits ;
- Une faible évolutivité.

### **2.2.2 Outils semi-automatisés**

Certains établissements utilisent des logiciels spécialisés permettant d'assister la création des emplois du temps à travers des interfaces conviviales. Ces solutions, comme EDT (Index Éducation) ou ADE Campus, offrent des fonctionnalités intéressantes : définition des contraintes, visualisation des plannings, affectation de ressources, etc. [9]

Cependant, bien que plus performants que les méthodes manuelles, ces outils restent limités en matière d'automatisation complète et d'optimisation algorithmique. Leur flexibilité face à des situations dynamiques ou imprévues est également restreinte.

### **2.2.3 Systèmes basés sur l'intelligence artificielle**

L'essor des techniques d'intelligence artificielle, notamment l'optimisation sous contraintes (Constraint Programming), a permis l'émergence d'outils capables de résoudre automatiquement des problèmes complexes de planification. [10] Des bibliothèques telles que Google OR-Tools, OptaPlanner, ou encore ChocoSolver permettent de modéliser des contraintes multiples et de générer des solutions optimisées.

Les atouts de ces systèmes incluent :

- Une capacité à intégrer un grand nombre de contraintes simultanées;
- Une réduction significative des conflits et erreurs humaines ;
- Une optimisation de l'allocation des ressources ;
- Une adaptabilité à des contextes variés et évolutifs.

Toutefois, leur utilisation requiert des compétences en modélisation et en programmation, ce qui peut constituer un frein pour certains établissements.

### **2.2.4 Vers des plateformes intelligentes et collaboratives**

Une tendance actuelle est l'intégration de ces moteurs d'optimisation au sein de plateformes web collaboratives. Cela permet d'associer puissance algorithmique et accessibilité utilisateur. Ces solutions offrent des interfaces intuitives pour les gestionnaires, tout en assurant un traitement intelligent des contraintes en arrière-plan.

C'est dans cette optique que s'inscrit notre projet :

Proposer une application web intelligente, accessible aux différents acteurs éducatifs, tout en exploitant les capacités d'un moteur d'optimisation tel que Google OR-Tools pour garantir une planification efficace, équitable et évolutive.

## **2.3 Comparaison entre la Création Manuelle et Automatique d'un Emploi du Temps**

Tableau 2.1 Comparaison entre la méthode Manuelle et Automatique

Critère	Création Manuelle	Création Automatique (par algorithme/logiciel)
<b>Temps nécessaire</b>	Long et fastidieux (plusieurs jours)	Très rapide (quelques minutes)
<b>Ressources humaines</b>	Nécessite l'intervention de plusieurs personnes	Intervention humaine minimale après paramétrage
<b>Gestion des conflits</b>	Difficile à détecter (ex : chevauchement de salle)	Gérée automatiquement par l'algorithme
<b>Flexibilité</b>	Faible : chaque modification demande une reprise	Haute : les modifications sont propagées automatiquement
<b>Complexité supportée</b>	Limité à des structures simples	Capable de gérer des contraintes complexes et multiples
<b>Taux d'erreurs</b>	Élevé (erreurs humaines fréquentes)	Faible (contrôles automatiques intégrés)
<b>Équité de répartition</b>	Difficile à garantir (ex : surcharge d'un prof)	Optimisation équitable (heures, charges, préférences)
<b>Suivi et mise à jour</b>	Manuel et lent	Dynamique et rapide
<b>Adaptabilité aux imprévus</b>	Faible (demandant une reprise complète)	Haute (recalcul possible à tout moment)
<b>Outils nécessaires</b>	Tableurs, papier, communication manuelle	Logiciels spécialisés, ex : Google OR-Tools, OptaPlanner...

## 2.4 Analyse des besoins fonctionnels et non fonctionnels

La réussite de tout système d'information repose sur une compréhension claire et détaillée des besoins des utilisateurs finaux ainsi que des exigences techniques et organisationnelles. Cette section vise à identifier les besoins fonctionnels, c'est-à-dire les fonctionnalités attendues du système, et les besoins non fonctionnels, qui déterminent la qualité de service, la sécurité, l'ergonomie et la performance de l'application.

### 2.4.1 Identification des utilisateurs

L'application cible plusieurs profils d'utilisateurs, chacun ayant des besoins spécifiques :

- **Administrateur système** : responsable de la configuration de l'application, de la gestion des comptes, des droits d'accès et de la supervision globale.
- **Gestionnaire pédagogique** : chargé de la planification des emplois du temps, de l'affectation des salles et des enseignants, et de la résolution des conflits.
- **Enseignant** : peut consulter son emploi du temps, signaler ses indisponibilités ou proposer des créneaux de cours.
- **Étudiant / Apprenant** : consulte son planning personnalisé et peut recevoir des notifications liées aux changements.
- **Responsable technique / informaticien** (optionnel) : s'assure du bon fonctionnement technique de la plateforme (mise à jour, sécurité, sauvegardes...).

## 2.4.2 Besoins fonctionnels

Les besoins fonctionnels décrivent les principales actions que le système doit permettre. Parmi eux, on peut citer :

- **Gestion des utilisateurs** : création, authentification, gestion des rôles et des droits d'accès.
- **Gestion des ressources** : saisie et modification des salles, équipements, enseignants, filières et groupes.
- **Planification intelligente** : génération automatique d'emplois du temps en fonction des contraintes (disponibilités, capacités des salles, conflits à éviter...).
- **Réservation de ressources** : possibilité de réserver manuellement des salles ou équipements pour des événements ponctuels.
- **Visualisation des plannings** : affichage clair et dynamique des emplois du temps par utilisateur ou par ressource.
- **Historique et audit** : journalisation des actions importantes pour le suivi et la transparence.

- **Notifications** : alertes automatiques en cas de modifications de planning ou de conflit détecté.

### 2.4.3 Besoins non fonctionnels

Les besoins non fonctionnels définissent les qualités que le système doit posséder indépendamment de ses fonctionnalités.

- **Accessibilité** : l'interface doit être accessible via tout navigateur web, sur ordinateur, tablette ou smartphone.
- **Sécurité** : le système doit garantir la confidentialité des données, une authentification sécurisée, et un contrôle des accès strict.
- **Performance** : le traitement des contraintes et la génération des emplois du temps doivent s'effectuer en un temps raisonnable, même en présence de nombreux utilisateurs et ressources.
- **Scalabilité** : le système doit pouvoir évoluer pour gérer un nombre croissant d'utilisateurs et de ressources sans perte significative de performance.
- **Simplicité d'utilisation** : l'ergonomie de l'interface doit faciliter la prise en main par des utilisateurs non techniques.
- **Maintenance** : l'architecture doit permettre des mises à jour, des corrections de bugs et une évolution facilitée.
- **Interopérabilité** : possibilité d'échanger ou d'intégrer des données avec d'autres systèmes (bases de données, logiciels pédagogiques...).

### 2.5 Analyse des contraintes du système

Dans le cadre de la conception d'un système de gestion coopérative des ressources, il est essentiel d'identifier les contraintes qui influencent son fonctionnement. Ces contraintes, qu'elles soient techniques, organisationnelles ou humaines, conditionnent les choix d'architecture, de technologies, et de modélisation. Leur bonne prise en compte dès la

phase d'analyse permet de concevoir une solution réaliste, stable et adaptée aux contextes des établissements éducatifs.

### 2.5.1 Contraintes de ressources

La première famille de contraintes concerne les ressources à planifier et à gérer :

- **Contraintes de capacité** : chaque salle dispose d'une capacité d'accueil maximale qu'il faut respecter lors de l'affectation des groupes.
- **Contraintes d'équipement** : certaines activités nécessitent des équipements spécifiques (laboratoires, vidéoprojecteurs, ordinateurs...), limitant les choix de salle.
- **Contraintes humaines** : les enseignants ne peuvent intervenir que sur certains créneaux définis selon leurs disponibilités. De plus, un enseignant ne peut être affecté simultanément à plusieurs groupes.

### 2.5.2 Contraintes temporelles

Le temps est un facteur central dans la planification :

- **Horaires de fonctionnement** : les établissements ont des plages horaires définies (ex. : 8h-17h), parfois différentes selon les jours.
- **Conflits d'occupation** : une ressource (salle, enseignant) ne peut être utilisée que pour une activité à la fois à un instant donné.
- **Préférences ou interdictions de créneaux** : certaines activités pédagogiques doivent être positionnées à des moments spécifiques (ex. : TP l'après-midi, CM le matin).

### 2.5.3 Contraintes organisationnelles

Chaque établissement possède des règles internes spécifiques :

- **Règlements pédagogiques** : durée des séances (ex. : 1.30h ou 2h), nombre de séances hebdomadaires par matière, etc.
- **Structure curriculaire** : les programmes imposent des répartitions précises des disciplines entre les semestres et les groupes.

- **Contraintes d'équité** : éviter des surcharges pour certains enseignants ou des horaires désavantageux pour certains groupes.

#### 2.5.4 Contraintes techniques

Ces contraintes concernent l'implémentation et l'infrastructure du système :

- **Interopérabilité** : le système doit pouvoir échanger des données avec d'autres outils administratifs ou pédagogiques.
- **Connexion réseau** : l'application doit fonctionner correctement même en cas de débit limité ou d'interruptions temporaires.
- **Compatibilité navigateur** : l'interface doit être accessible depuis les principaux navigateurs et systèmes d'exploitation.

#### 2.5.5 Contraintes liées à l'usage collaboratif

Puisque l'approche retenue repose sur une gestion coopérative, d'autres contraintes émergent :

- **Conflits de décisions** : plusieurs utilisateurs peuvent interagir sur les mêmes ressources ; le système doit gérer ces situations.
- **Traçabilité des actions** : les modifications doivent être enregistrées et attribuables à un utilisateur pour garantir la transparence.
- **Accès simultané** : le système doit supporter des connexions concurrentes sans altérer les performances.

### 2.6 Choix méthodologiques et modélisation

Afin de concevoir un système de gestion coopérative des ressources à la fois robuste, évolutif et adapté aux établissements éducatifs, une approche méthodologique rigoureuse est nécessaire. Cette section présente les choix de modélisation retenus pour représenter les entités,

les interactions et les processus métier, ainsi que les outils mobilisés pour leur formalisation.

### **2.6.1 Approche orientée objet**

Nous avons opté pour une modélisation orientée objet, qui permet de structurer le système en entités distinctes (objets), représentant les éléments clés de l'environnement éducatif : enseignants, salles, cours, emplois du temps, etc. Cette approche facilite la réutilisabilité, l'extensibilité du code, et s'intègre naturellement à la programmation moderne. [11]

Elle est également particulièrement adaptée à l'utilisation d'outils de conception visuelle tels que les diagrammes UML.

### **2.6.2 Modèle UML**

L'Unified Modeling Language (UML) est le langage de modélisation graphique basé sur des pictogrammes conçus pour représenter, spécifier les artefacts d'un système logiciel, en outre, il est conçu pour comprendre et décrire les exigences système spécifiées et documentées, esquisser l'architecture logicielle, concevoir des solutions et communiquer des idées, car il peut être appliquée à divers systèmes, non limités au domaine informatique. UML est né de l'unification de techniques éprouvées d'analyse et de conception de grands logiciels et de systèmes complexes. [12]

#### **2.6.2.1 Points fort d'UML**

UML présente plusieurs avantages :

- Un gain de précision.
- Un gage de stabilité.
  - Construit un cadre analytique et facilite la compréhension de représentations abstraites complexes. [13]

#### **2.6.2.2 Points faible d'UML**

- La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation.

- L'intégration d'UML dans un processus n'est pas trivial, et améliorer un processus est une tâche complexe et longue. [13]

### **2.6.3 Raison du choix de Google OR-Tools**

La génération automatisée d'emplois du temps constitue une tâche complexe relevant de l'optimisation sous contraintes. Pour y répondre efficacement, nous avons sélectionné la bibliothèque Google OR-Tools, développée par Google et reconnue pour sa performance et sa flexibilité.[14]

Ses avantages incluent :

- La prise en charge de multiples contraintes combinées (ressources, horaires, préférences, etc.) ;
- Des solveurs performants (CP-SAT) adaptés aux problèmes discrets complexes, (CPSAT est une librairie open source écrite en C++, avec des APIs en Java, Python et C#). [15]
- Une documentation riche et une intégration fluide avec Python, langage de développement retenu.

### **2.6.4 Démarche de modélisation du problème de planification**

Le problème de planification a été formalisé comme un Problème de Satisfaction de Contraintes (CSP) :

- Variables : créneaux horaires, salles, enseignants, matières ;
- Domaines : valeurs possibles pour chaque variable (ex. : jours et heures disponibles) ;
- Contraintes : indisponibilités, non-conflit, capacité, types de salles, etc.

Cette modélisation a été traduite sous forme de code dans OR-Tools, permettant de générer des solutions optimisées automatiquement.

#### **2.6.4.1 Langage de modélisation formalise CSP**

Un Problème de Satisfaction de Contraintes (CSP) est défini par un ensemble de variables, chacune munie d'un domaine de valeurs, et d'un ensemble de contraintes définissant les combinaisons de valeurs autorisées. Résoudre un CSP revient à affecter à chaque variable une valeur de son domaine, de sorte que les contraintes du modèle soient vérifiées. [16]

## **2.7 Architecture technique de la solution**

Le développement d'une application web intelligente pour la gestion coopérative des ressources repose sur une architecture bien pensée, garantissant modularité, évolutivité, sécurité et accessibilité. Cette section présente l'architecture logique du système, les composants principaux, ainsi que les choix techniques réalisés pour chaque couche de l'application.

### **2.7.1 Architecture en couches**

L'architecture en couches, également connue sous le nom d'architecture multicouche, est un modèle de conception logicielle largement adopté dans l'ingénierie logicielle moderne pour faciliter l'organisation, la séparation des préoccupations et la maintenabilité des applications. Il fait référence à la manière dont les composants logiciels sont disposés et structurés de manière à créer des applications présentant une modularité, une évolutivité et une flexibilité, ainsi qu'à promouvoir la réduction du code dupliqué, l'adaptabilité aux changements et l'extensibilité pour la mise en œuvre de nouvelles fonctionnalités et exigences. [17]

Nous avons adopté une architecture en trois couches (Three-Tier Architecture), largement utilisée dans les systèmes distribués modernes. Elle permet de séparer les responsabilités, de faciliter la maintenance et de favoriser la réutilisabilité :

- Couche présentation (frontend) : interface utilisateur web.
- Couche métier (backend) : logique applicative et règles de gestion.

- Couche données : stockage et gestion des ressources dans une base de données.

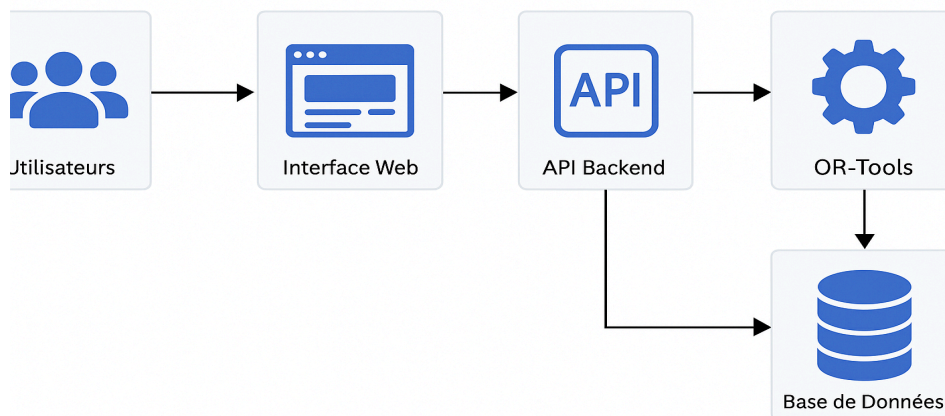
## 2.7.2 Technologies utilisées

Le tableau ci-dessous synthétise les technologies retenues pour chaque composant du système :

**Tableau 2.2 Synthèse des Technologies Utilisées par Composant du Système**

Composant	Technologie choisie	Justification
Interface utilisateur	HTML, CSS, Python, django, ajax, bootstrap	Réactivité, ergonomie, modernité
Serveur applicatif	Python	Léger, rapide, facile à intégrer avec OR-Tools
Optimisation	Google OR-Tools	Puissant moteur d'optimisation sous contraintes
Base de données	PostgreSQL	Fiabilité, support des types complexes et des transactions
Hébergement	Render	Déploiement cloud simple et sécurisé

## 2.7.3 Schéma global de l'architecture



**Figure 2.1 Schéma global de l'architecture**

Le bon fonctionnement de notre système repose sur une architecture technique bien structurée, permettant d'assurer l'interaction fluide entre les utilisateurs, l'interface web, le moteur d'optimisation et la base de données. Cette organisation modulaire vise à garantir la performance, la maintenabilité et la scalabilité de l'application.

L'architecture globale du système peut être résumée en quatre couches principales :

#### 1. **Les utilisateurs**

Il s'agit des différents acteurs du système : les administrateurs, les enseignants, les responsables de planification, et potentiellement les étudiants. Chacun interagit avec la plateforme selon ses droits et ses besoins.

#### 2. **L'interface web**

Accessible via un navigateur, cette interface constitue le point d'entrée principal du système. Elle permet la saisie des données (ressources, contraintes, demandes), la consultation des emplois du temps, et la gestion collaborative des ressources.

#### 3. **L'API backend**

Le cœur logique de l'application, développé avec un framework web (comme Flask ou Django), assure la communication entre l'interface utilisateur, la base de données et le moteur d'optimisation. Il gère les requêtes, applique les règles métiers et orchestre les traitements.

#### 4. **Le moteur d'optimisation et la base de données**

- a. **Google OR-Tools** est utilisé pour la génération automatique des emplois du temps en fonction des contraintes définies.
- b. **La base de données** (comme PostgreSQL ou MySQL) stocke toutes les informations nécessaires : profils utilisateurs, ressources, contraintes, plannings, historiques, etc.

## 2.7.4 Sécurité et accessibilité

Dans un système collaboratif destiné à gérer des ressources critiques au sein d'établissements éducatifs, la sécurité des données et l'accessibilité de la plateforme constituent des exigences fondamentales. Ces aspects influencent directement la confiance des utilisateurs, la conformité réglementaire et la qualité du service rendu.

### 2.7.4.1 Sécurité du système

La sécurité est prise en compte à plusieurs niveaux de l'architecture du système :

- **Authentification et autorisation**

L'accès à la plateforme est restreint via un système d'authentification sécurisé (par identifiants et mots de passe chiffrés). Des rôles spécifiques (administrateur, enseignant, planificateur, etc.) sont définis pour limiter l'accès aux fonctionnalités sensibles.

- **Protection des données**

Les échanges de données sont chiffrés via le protocole HTTPS. Les informations sensibles (mots de passe, données personnelles) sont stockées de manière sécurisée (hashage, chiffrement en base de données).

- **Prévention des attaques**

Des mécanismes comme (Utilisation d'ORM comme Django ORM, Échappement du contenu HTML) sont mis en place pour prévenir les attaques courantes : injections SQL, scripts intersites (XSS), falsification de requêtes intersites (CSRF), etc.

- **Sauvegarde et résilience**

Des procédures régulières de sauvegarde de la base de données permettent de prévenir la perte de données. Des mesures de redondance peuvent être ajoutées pour garantir la continuité du service.

### **2.7.4.2 Accessibilité de la plateforme**

L'accessibilité vise à garantir que tous les utilisateurs, quels que soient leur profil ou leurs équipements, puissent utiliser efficacement la plateforme.

- **Compatibilité multi-appareils**

L'interface est développée en responsive design, assurant une compatibilité avec les ordinateurs, tablettes et smartphones.

- **Accessibilité universelle**

Des bonnes pratiques d'accessibilité sont intégrées : navigation au clavier, contrastes suffisants, alternatives textuelles aux éléments visuels, conformité aux normes WCAG (Web Content Accessibility Guidelines).

- **Haute disponibilité**

L'architecture vise à minimiser les interruptions grâce à un hébergement fiable, une gestion appropriée des erreurs, et une surveillance en temps réel du service.

Ces dispositifs visent à faire de l'application un outil robuste, fiable et inclusif, adapté à un usage quotidien dans des contextes éducatifs variés.

## **2.8 Conclusion**

Ce chapitre a posé les bases analytiques et techniques de notre projet de gestion coopérative des ressources dans les établissements éducatifs. Nous avons d'abord étudié les approches existantes, des méthodes traditionnelles aux solutions basées sur l'intelligence artificielle, en soulignant leurs limites et les opportunités offertes par les technologies modernes. Cette revue critique a permis de justifier l'adoption d'une approche hybride : allier les avantages d'une interface web accessible à la puissance d'un moteur d'optimisation comme Google OR-Tools.

Par la suite, nous avons défini les besoins fonctionnels et non fonctionnels du système, en intégrant les contraintes pratiques liées à l'usage quotidien dans un cadre éducatif. L'architecture technique globale, ainsi que les technologies choisies pour le développement, ont été présentées de

manière structurée, en cohérence avec les objectifs du projet. Enfin, nous avons abordé les enjeux de sécurité et d'accessibilité, indispensables pour garantir la fiabilité et l'adoption de la solution.

Ce travail préparatoire constitue un socle solide pour aborder, dans le chapitre suivant, la conception détaillée, le développement et l'implémentation de l'application, en mettant en œuvre les modèles et technologies définis ici.

# Chapitre 3 : Conception du système

## 3.1 Introduction

La phase de conception constitue une étape charnière dans tout processus de développement logiciel. Elle permet de traduire les besoins identifiés et les choix techniques effectués lors des phases d'analyse en une modélisation concrète et cohérente du système à construire. Cette modélisation repose principalement sur des outils normalisés tels que le langage UML (Unified Modeling Language), qui offre une représentation visuelle claire de l'architecture du système, des interactions entre les acteurs, ainsi que du comportement attendu.

Dans le cadre de notre projet de gestion coopérative des ressources dans les établissements éducatifs, cette étape de conception vise à formaliser les structures, les cas d'usage, ainsi que les interactions nécessaires au bon fonctionnement de l'application. Elle prend en compte la diversité des ressources à gérer (salles, enseignants, équipements), les multiples contraintes (temps, disponibilité, types d'activités), ainsi que l'intégration d'un moteur d'optimisation basé sur l'intelligence artificielle (Google OR-Tools).

Ce chapitre présentera :

- Une définition brève du langage UML et sa pertinence dans notre contexte,
- Le diagramme de cas d'utilisation représentant les interactions entre les utilisateurs et le système,
- Le diagramme de classes définissant la structure des entités du système et leurs relations,
- Un ou plusieurs diagrammes de séquence illustrant les scénarios dynamiques de traitement,
- Et, le cas échéant, une ébauche du modèle de données.

À travers cette conception détaillée, nous établissons une base solide et cohérente pour la phase suivante de réalisation, en assurant que toutes

les fonctionnalités prévues soient correctement prises en compte dès la phase de modélisation.

## 3.2 Présentation du langage UML

Le langage UML ([Unified Modeling Language](#)) est un moyen de représenter visuellement l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes. Une application comporte des milliers de lignes de code, ce qui rend difficile la gestion des interactions et des hiérarchies. Les diagrammes UML résolvent ce problème en divisant un système logiciel en composants et sous-composants. [18]

UML propose une variété de diagrammes, répartis en deux grandes catégories :

- **Diagrammes structurels** : ils décrivent les éléments statiques du système (classes, objets, composants, etc.). Le diagramme de classes est l'un des plus utilisés dans cette catégorie. [18]
- **Diagrammes comportementaux** : ils illustrent les interactions et les processus dynamiques. Les principaux sont le diagramme de cas d'utilisation, le diagramme de séquence, et le diagramme d'activité. [18]

L'utilisation de l'UML dans notre projet permet :

- D'assurer une communication claire entre les différents membres de l'équipe de développement ;
- De structurer les besoins fonctionnels et techniques de manière formelle ;
- De faciliter la maintenance et les évolutions futures du système.

Dans la suite de ce chapitre, nous présentons trois types de diagrammes essentiels à notre conception :

- **Diagramme de cas d'utilisation** : il décrit les interactions entre les utilisateurs (administrateur, enseignant, responsable de planning...) et le système, en identifiant les principales fonctionnalités. [18]

- **Diagramme de classes** : il modélise les entités du système ainsi que leurs relations, attributs et méthodes. Ce diagramme constitue la base de la structure du code. [18]
- **Diagramme de séquence** : il détaille la chronologie des échanges entre les objets lors d'un scénario spécifique (ex. : génération d'un emploi du temps), mettant en évidence la logique de traitement.[18]

### 3.3 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation (ou *Use Case Diagram*) est un outil fondamental du langage UML permettant de représenter les interactions entre les acteurs (utilisateurs ou systèmes externes) et le système étudié. Il permet de visualiser les fonctions principales que le système doit offrir, sans entrer dans les détails techniques de l'implémentation. [12]

#### 3.3.1 Objectifs du diagramme de cas d'utilisation

Ce type de diagramme permet de :

- Identifier les différents utilisateurs du système (acteurs) et leurs rôles.
- Définir les fonctionnalités attendues par ces utilisateurs (cas d'utilisation).
- Clarifier les exigences fonctionnelles du système.
- Servir de base à la conception technique et aux tests. [12]

#### 3.3.2 Acteurs identifiés dans notre projet

Dans le cadre de notre application de gestion coopérative des ressources éducatives, nous identifions les acteurs suivants :

- **Administrateur** : gère les utilisateurs, les ressources, et les paramètres globaux du système ;
- **Enseignant** : consulte son emploi du temps, signale ses disponibilités ou ses contraintes ;
- **Responsable pédagogique** : propose des plannings, modifie des affectations, valide l'organisation générale ;

- **Étudiant** : consulte son emploi du temps ;
- **Système d'optimisation** : module interne piloté par l'IA (ex. : Google OR-Tools) chargé de générer automatiquement les emplois du temps optimisés.

### 3.3.3 Cas d'utilisation principaux

Parmi les cas d'utilisation les plus importants, on peut citer :

- Authentification des utilisateurs.
- Consultation et gestion des ressources (salles, enseignants, équipements).
- Déclaration des contraintes et préférences.
- Génération automatique d'un emploi du temps.
- Validation ou ajustement manuel d'un planning.
- Consultation des emplois du temps par les étudiants et enseignants.
- Exportation et diffusion des plannings.

Le schéma ci-dessous illustre les différentes interactions entre les utilisateurs du système et les fonctionnalités offertes par l'application :



Figure 3.1 diagramme de cas d'utilisation utilisateurs

### 3.4 Diagramme de classes

Le diagramme de classes est un élément fondamental de la modélisation orientée objet. Il permet de représenter les entités principales du système, leurs attributs, leurs méthodes, ainsi que les relations qui les unissent (association, agrégation, composition, héritage). Dans le cadre de notre application web de gestion coopérative des ressources, ce

diagramme constitue un support essentiel pour structurer les données et concevoir la base du système d'information. [12]

### 3.4.1 Objectifs du diagramme de classes

Le diagramme de classes vise à :

- Modéliser les entités métiers essentielles (utilisateurs, ressources, réservations, emplois du temps, etc.).
- Clarifier les responsabilités et interactions entre les classes.
- Servir de base à la conception de la base de données relationnelle.
- Guider l'implémentation logique côté backend. [12]

### 3.4.2 Principales classes identifiées

Voici un aperçu des principales classes de notre système :

- **Utilisateur** : classe générique représentant les acteurs (administrateur, enseignant, étudiant) avec des attributs communs comme nom, email, rôle, etc.
- **Ressource** : entité abstraite pouvant être spécialisée en Salle, équipement.
- **Planning** : regroupe les créneaux horaires et spécialité et module et les affectations de ressources.
- **Réservation salles** : enregistre les demandes et attributions de ressources pour une période donnée.
- **Module** : associé à un enseignant, une spécialité, une ressource, et une plage horaire (planning).

### 3.4.3 Présentation du diagramme

Le diagramme suivant illustre la structure statique du système à travers les classes, leurs attributs, méthodes principales, et les relations (avec cardinalités) qui les relient.



- Visualiser le flux d'exécution logique des opérations ;
- Identifier les responsabilités de chaque composant dans le traitement d'un cas d'usage ;
- Guider la conception des interfaces et des contrôleurs métier.

### 3.5.2 Scénario 1 : Authentification

Ci-dessous le diagramme d'authentification d'un utilisateur qui peut être un enseignant ou étudiant.

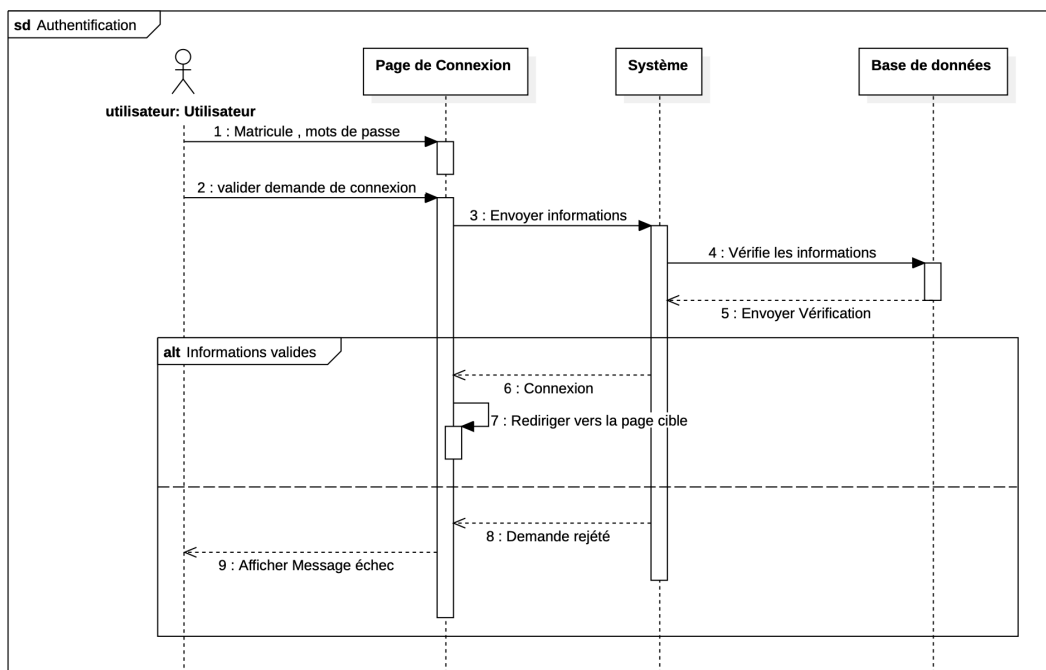


Figure 3.3 diagramme de séquence d'authentification

Ci-dessous le diagramme d'inscription d'un gestionnaire

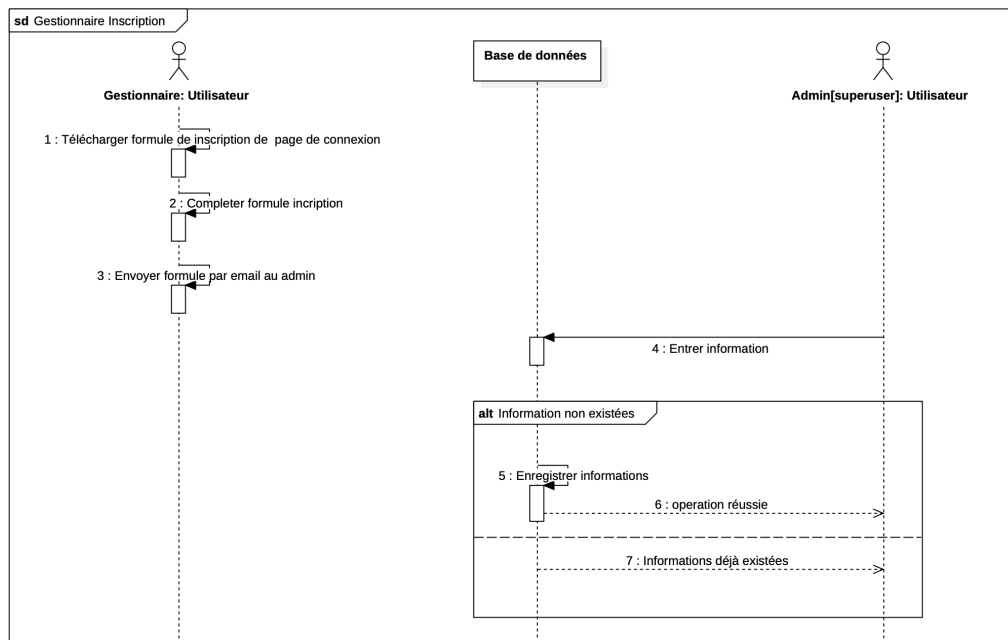


Figure 3.4 diagramme de séquence d'inscription d'un gestionnaire

### 3.5.3 Scénario 2 : Choix du groupe par un étudiant

Ce scénario illustre le processus à travers lequel un étudiant sélectionne le groupe auquel il souhaite appartenir. Ce choix est important car il conditionne les horaires qui lui seront attribués par le système lors de la génération de l'emploi du temps.

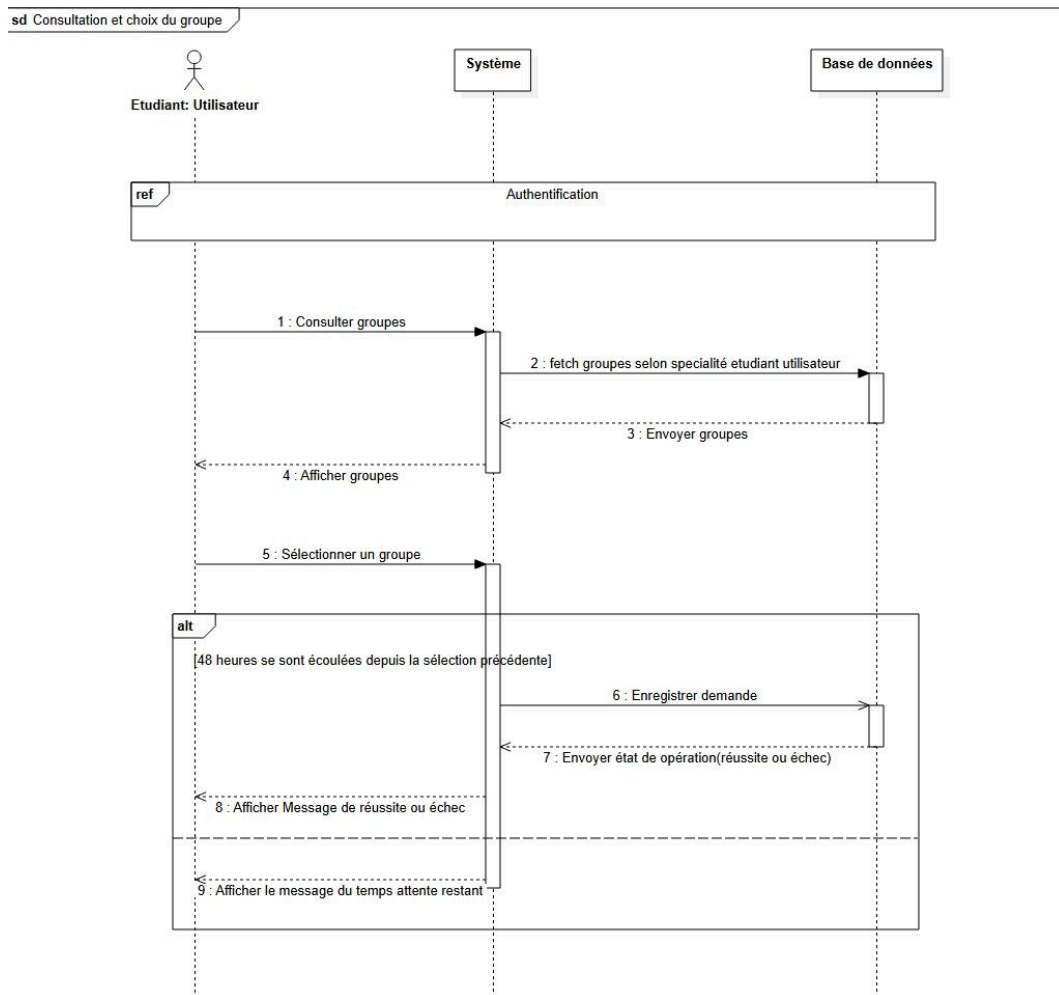


Figure 3.5 diagramme de séquence Choix du groupe par un étudiant

### 3.5.4 Scénario 3 : Déclaration des indisponibilités par un enseignant

Ce scénario décrit comment un enseignant renseigne ses plages horaires d'indisponibilité (ex. pour éviter les conflits avec d'autres activités professionnelles ou personnelles). Ces contraintes seront prises en compte par l'algorithme de planification.

#### Description du scénario :

- L'enseignant se connecte à l'interface web.

- Il accède à la section "Déclaration d'indisponibilités".
- Il sélectionne les jours et créneaux horaires durant lesquels il ne sera pas disponible.
- Il valide la soumission.
- Le système enregistre les indisponibilités dans la base de données.
- Ces données seront ensuite prises en compte par l'algorithme OR-Tools lors de la génération de l'emploi du temps.

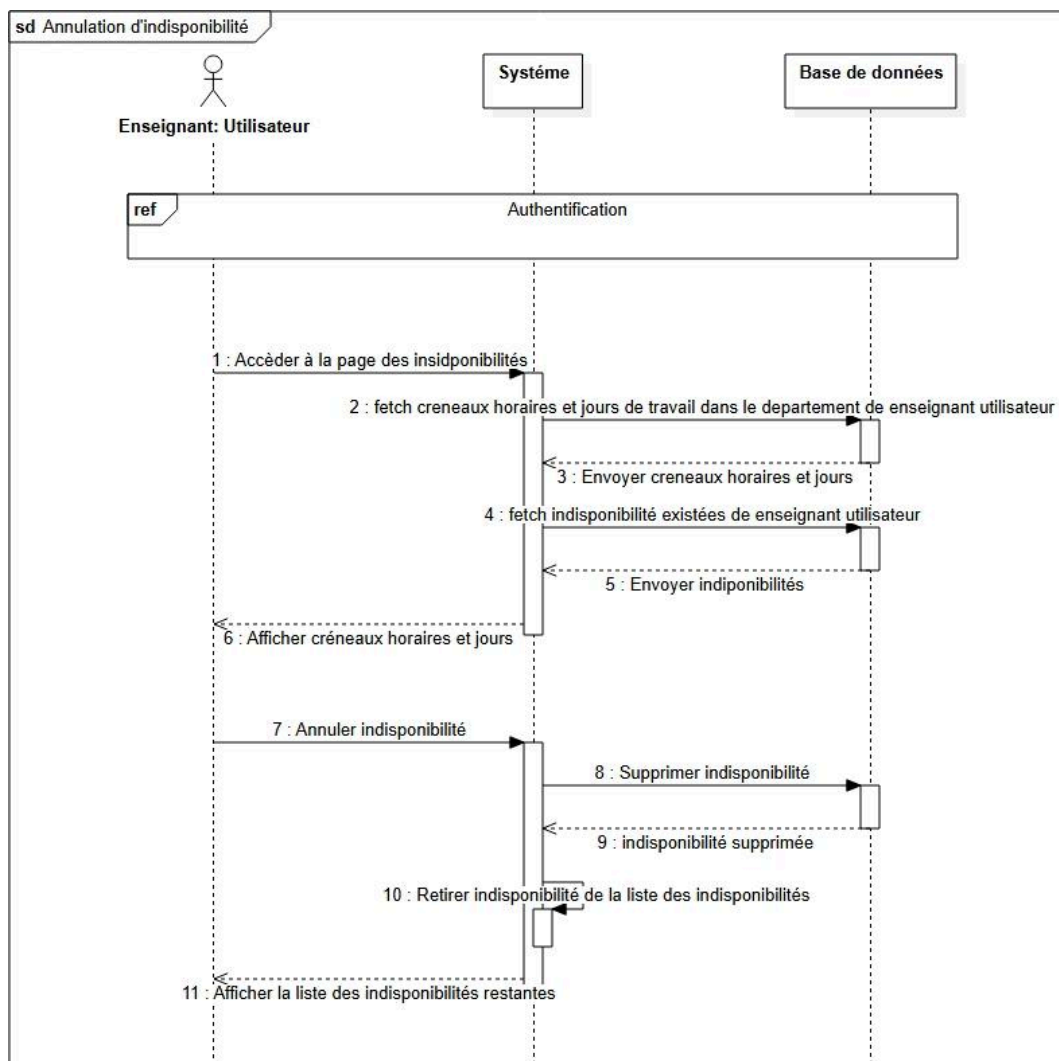


Figure 3.6 diagramme de séquence Déclaration des indisponibilités par un enseignant

### 3.5.5 Scénario 4 : génération d'un emploi du temps

Le diagramme suivant représente un scénario standard où un administrateur initie la génération automatique d'un emploi du temps, en intégrant les contraintes (disponibilités, salles, durées, etc.). Le système exploite alors le moteur d'optimisation Google OR-Tools pour produire une solution valide.

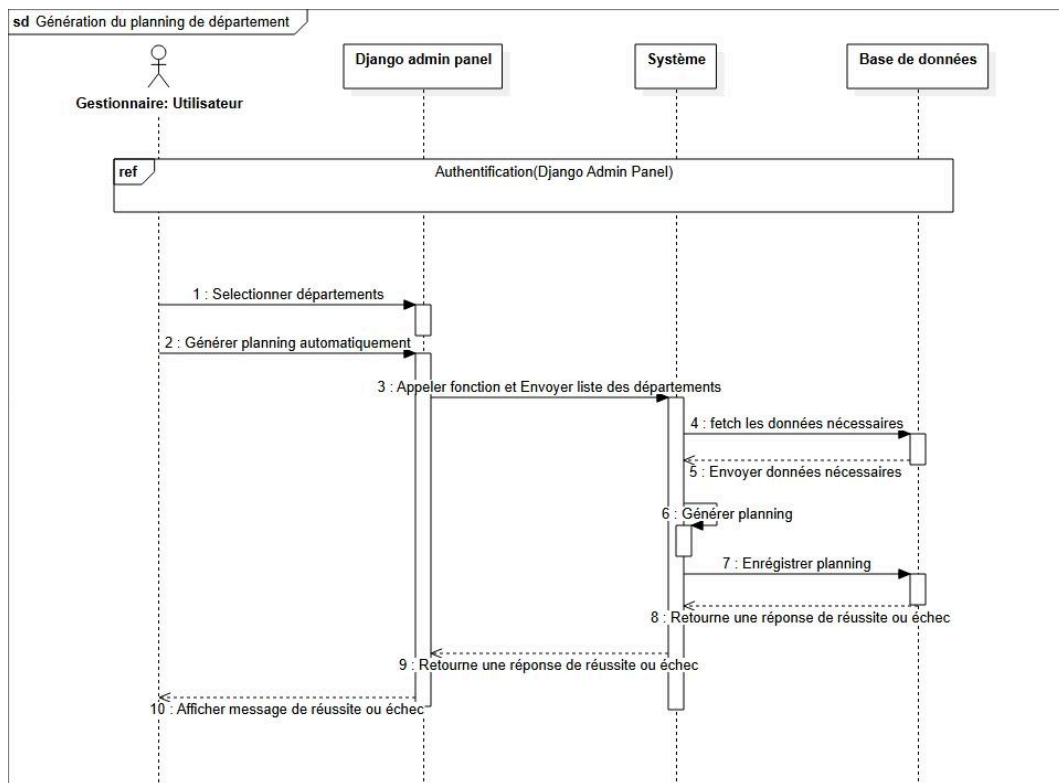


Figure 3.7 diagramme de séquence génération d'un emploi du temps

Les principales interactions représentées dans ce diagramme incluent :

1. L'administrateur accède à l'interface web et déclenche la génération.
2. L'interface appelle le backend via une API.
3. Le backend extrait les données nécessaires depuis la base de données.
4. Il construit un modèle de contraintes pour OR-Tools.

5. OR-Tools exécute l'algorithme et retourne une solution optimisée.
6. Le backend enregistre le nouvel emploi du temps.
7. Le système notifie l'administrateur et met à jour l'interface.

### **3.5.6 Scénario 5 : Réservation d'une salle par un enseignant**

Ce scénario illustre le processus par lequel un enseignant peut réserver une salle spécifique pour un besoin ponctuel (travaux pratiques, réunion, rattrapage, etc.) via l'interface de la plateforme web. Ce mécanisme vise à renforcer l'autonomie des utilisateurs tout en assurant un contrôle sur la disponibilité et l'utilisation des ressources.

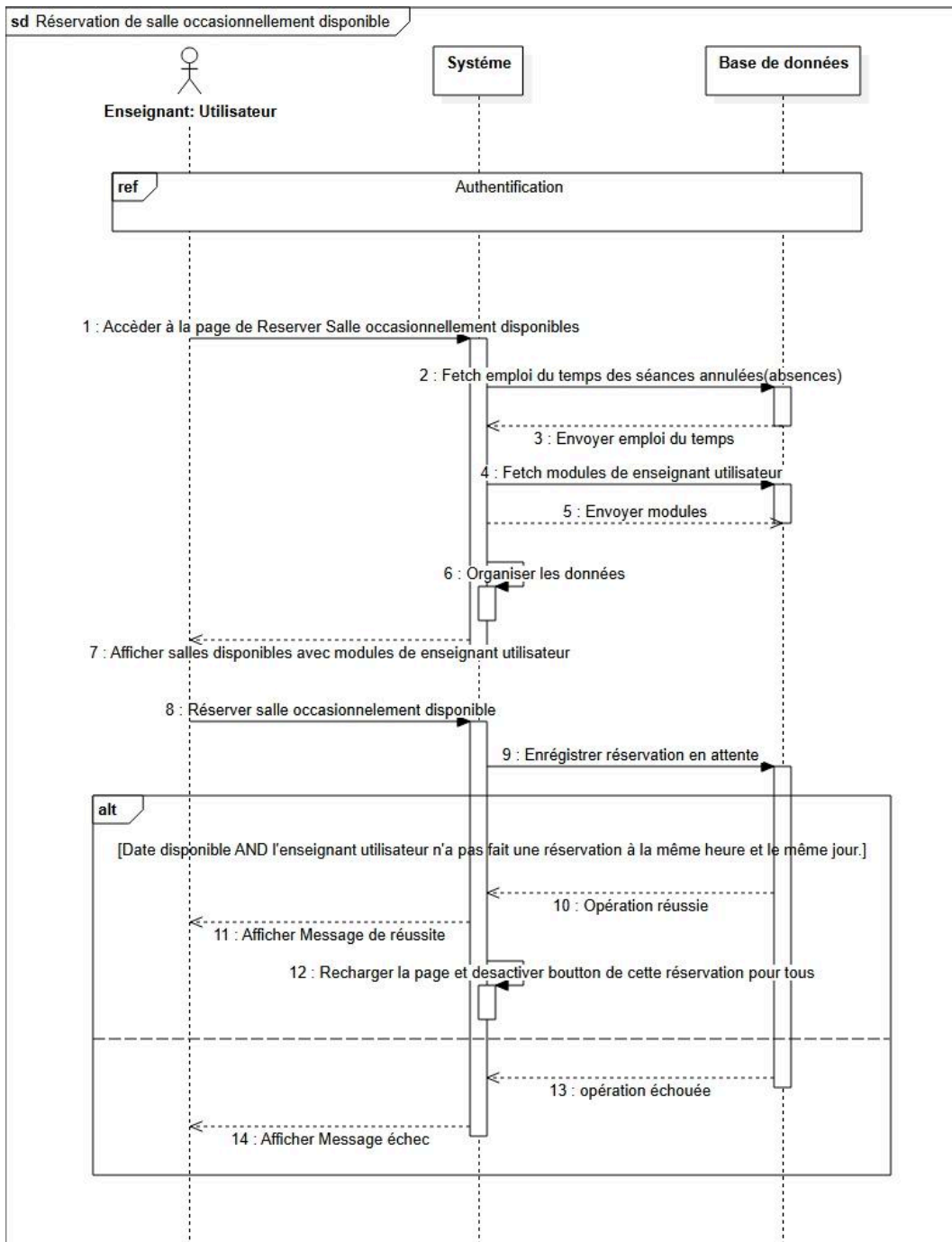


Figure 3.8 diagramme de séquence Réserve d'une salle par un enseignant

**Description du scénario :**

- L'enseignant se connecte à la plateforme via son compte utilisateur.
- Il accède à la section « Réserve de salles ».

- Le système affiche les créneaux disponibles en fonction de la date, de l'heure et du type de salle souhaitée.
- L'enseignant sélectionne un créneau libre et soumet une demande de réservation.
- Le système vérifie la disponibilité en temps réel, et envoie une demande au chef département.
- Si le créneau est disponible et la réservation est confirmée par le au chef département, un message de confirmation est affiché.

### **3.5.7 Scénario 6 : Annonce d'une absence par un enseignant**

Ce scénario décrit la procédure par laquelle un enseignant peut déclarer une absence prévue, afin de permettre à l'administration de gérer les ajustements nécessaires dans le planning (annulation, report, remplacement, etc.).

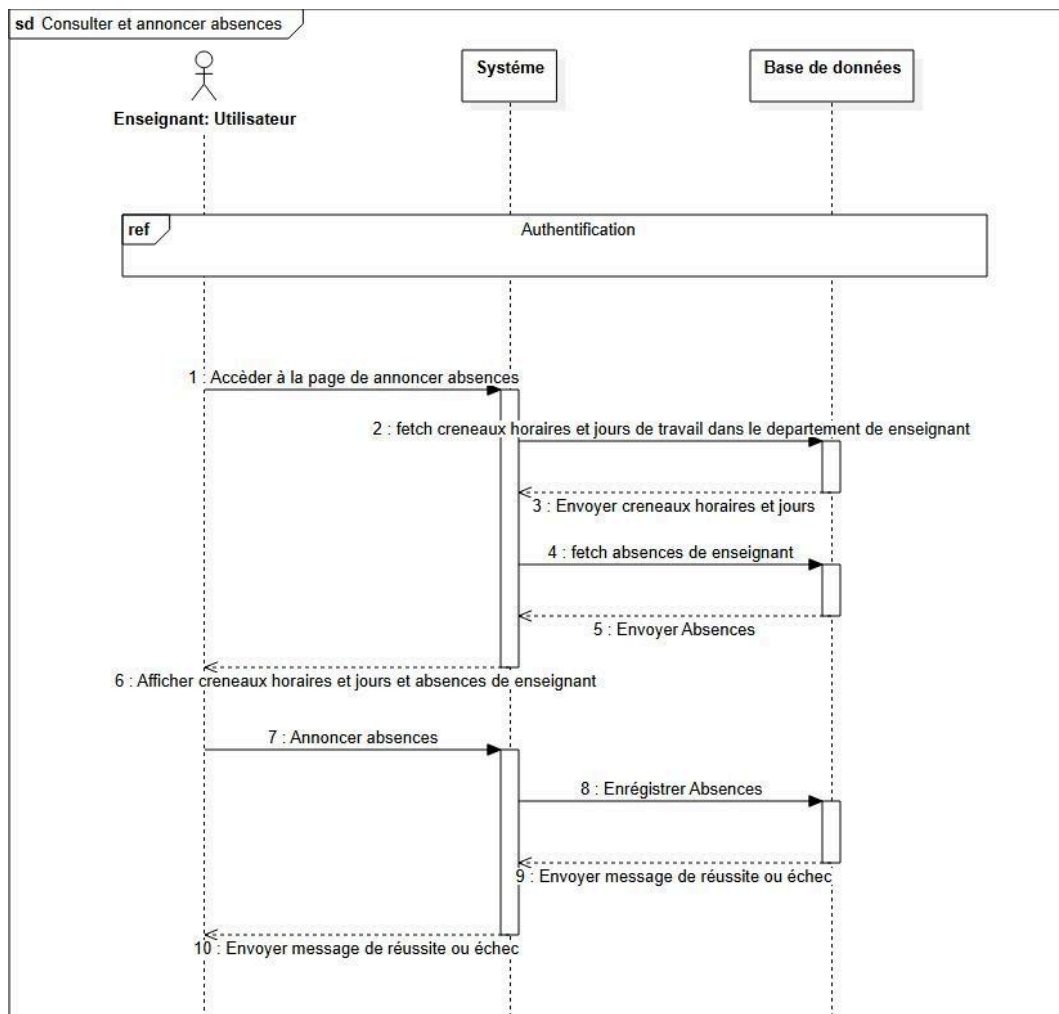


Figure 3.9 diagramme de séquence Annonce d'une absence par un enseignant

### Description du scénario :

- L'enseignant se connecte à la plateforme avec ses identifiants.
- Il accède à la section « Déclaration d'absence » dans son tableau de bord.
- Il sélectionne le jour et l'heure de la séance concernée à partir de son emploi du temps.
- Une notification est envoyée automatiquement aux étudiants concernés et aux autres parties impliquées (ex. : remplaçant).

## 3.6 Conception de la base de données

La conception de la base de données constitue une étape essentielle dans le développement d'une application web de gestion coopérative des ressources. Elle permet de structurer les informations, d'assurer la cohérence des données et de faciliter les opérations de planification, de réservation et de consultation.

### 3.6.1 Objectifs de la base de données

La base de données doit permettre de :

- Stocker toutes les informations relatives aux ressources (salles, équipements, enseignants, étudiants, groupes, etc.) ;
- Gérer les emplois du temps et les contraintes associées ;
- Traiter les demandes de réservation et les indisponibilités ;
- Assurer la traçabilité des modifications et des validations effectuées par les utilisateurs.

## 3.7 Conception de l'interface utilisateur

La conception de l'interface utilisateur (UI) vise à offrir une expérience intuitive, ergonomique et adaptée aux différents profils d'utilisateurs (administrateurs, enseignants, étudiants). L'interface joue un rôle central dans la réussite de l'application, en facilitant l'accès aux fonctionnalités essentielles tout en assurant la clarté des informations affichées.

### 3.7.1 Principes de conception

La conception de l'interface repose sur les principes suivants :

- **Clarté** : les menus, icônes et libellés doivent être explicites ;
- **Accessibilité** : l'application doit être utilisable sur différents appareils (ordinateurs, tablettes, smartphones) ;
- **Cohérence** : les écrans doivent suivre une logique unifiée pour réduire la courbe d'apprentissage ;

- **Feedback utilisateur** : chaque action (validation, erreur, confirmation) doit être accompagnée d'un message clair.

### **3.8 Conclusion du chapitre**

Ce chapitre a permis de présenter en détail la phase de conception du système de gestion coopérative des ressources. À travers l'analyse des besoins, l'identification des acteurs, la modélisation des cas d'utilisation, des classes et des séquences, nous avons défini une architecture fonctionnelle claire, capable de répondre aux objectifs fixés. La conception de l'interface utilisateur, orientée vers la simplicité et l'accessibilité, vient renforcer l'efficacité de l'application.

Ces fondations conceptuelles solides constituent une base essentielle pour l'implémentation du système, qui sera détaillée dans le chapitre suivant. Ce dernier décrira les choix techniques finaux, l'intégration du moteur d'optimisation, ainsi que les résultats obtenus lors de la mise en œuvre.

# Chapitre 4 : Implémentation et validation

## 4.1 Introduction

Après avoir défini l'architecture fonctionnelle et les aspects conceptuels du système dans le chapitre précédent, ce quatrième chapitre est consacré à la mise en œuvre technique de l'application web de gestion coopérative des ressources. Il détaille les outils et technologies utilisés, les étapes de développement, ainsi que les mécanismes d'intégration du moteur d'optimisation des emplois du temps basé sur Google OR-Tools.

Dans un second temps, le chapitre aborde la validation de la solution, à travers des scénarios de test, L'objectif est de démontrer que le système développé répond bien aux besoins exprimés en termes d'efficacité, de fiabilité, d'utilisabilité et d'adaptabilité.

Ce chapitre s'articule autour des sections suivantes :

- Présentation de l'environnement de développement et des technologies employées.
- Implémentation des fonctionnalités principales du système.
- Intégration du moteur d'optimisation Google OR-Tools.
- Tests fonctionnels et techniques réalisés.
- Évaluation des performances et validation des objectifs du projet.

## 4.2 Environnement de développement

La réalisation de l'application web intelligente de gestion coopérative des ressources a nécessité un environnement de développement à la fois stable, flexible et adapté aux technologies modernes. Cette section décrit les outils, langages et bibliothèques utilisés tout au long du processus de développement.

## 4.2.1 Langage de programmation

Le langage principal utilisé est Python, un langage polyvalent, interprété et orienté objet, largement adopté dans le développement web. L'intelligence artificielle et l'automatisation.

### 4.2.1.1 Python



Python est un langage de programmation de haut niveau interprété, orienté objet et avec une sémantique dynamique. Ses structures de données intégrées de haut niveau, combinées au typage dynamique et à la liaison dynamique, le rendent très attrayant pour le développement rapide d'applications, ainsi que pour une utilisation comme langage de script ou de colle pour connecter les composants existants entre eux. La syntaxe simple et facile à apprendre de Python met l'accent

sur la lisibilité et réduit donc le coût de la maintenance du programme. Python prend en charge les modules et les packages, ce qui encourage la modularité du programme et la réutilisation du code. L'interpréteur Python et la vaste bibliothèque standard sont disponibles sous forme source ou binaire sans frais pour toutes les principales plates-formes, et peuvent être distribués gratuitement. [19]

### 4.2.1.2 HTML



HTML, qui signifie en anglais (HyperText Markup Language), est un langage informatique pour rédiger des pages web. Grâce à lui, il est possible de rédiger de l'hypertexte, de mettre en forme le contenu, de faire des formulaires de saisie, de rajouter dans la page des images, vidéos ou des graphismes ou encore de faire la sémantique de la page web. Ce langage fonctionne avec un système de balises qui vont servir à mettre en avant les différents éléments grâce à des titres, des sous-titres, etc.

C'est dans les années 1990 que ce langage a été créé et a évolué au fur et à mesure jusqu'à la 5ème version qui est utilisée aujourd'hui. [20]

### 4.2.1.3 CSS



CSS, ou Cascading Style Sheets (Feuilles de Style en Cascade), est un langage utilisé pour décrire l'apparence et la mise en forme d'un document écrit en HTML ou XML. En d'autres termes, alors que le HTML fournit la structure de la page (comme les titres, les paragraphes et les liens), le CSS détermine comment ces éléments doivent apparaître à l'écran. Cela inclut des aspects tels que la couleur, la taille de la police, l'espacement, et même des animations.

L'importance du CSS réside dans sa capacité à séparer le contenu de la présentation. Cela signifie que les développeurs et les designers peuvent modifier l'apparence d'un site web sans toucher à son contenu. Cette séparation offre également une plus grande flexibilité, permettant aux sites web de s'adapter à différents types d'appareils et tailles d'écran. [21]

### 4.2.1.4 JavaScript



JavaScript est un langage de programmation utilisé par les développeurs pour concevoir des sites web interactifs. Les fonctions JavaScript peuvent permettre d'améliorer l'expérience utilisateur d'un site web, de la mise à jour des flux de médias sociaux à l'affichage d'animations et de cartes interactives. En tant que langage de script côté client, c'est l'une des principales technologies du web. Lors de la navigation sur Internet, à tout moment vous pouvez par exemple voir un carrousel d'images, un menu déroulant « Cliquer pour afficher » ou le changement dynamique de la couleur des éléments d'une page web. Tout cela est possible grâce à JavaScript. [22]

## 4.2.2 Framework web

Un Framework est une boîte à outils pour un développeur web. Frame signifie *cadre* et work se traduit par *travail*. Un Framework contient des composants autonomes qui permettent de faciliter le développement d'un site web ou d'une application. [23]

#### 4.2.2.1 Django

C'est en 2005 que Django a été publié en open source. Le but de cet outil Python est de faciliter le développement rapide d'applications web complexes, tout en suivant les meilleures pratiques de développement.

Sa principale fonction est de débarrasser le programmeur des tâches les plus rébarbatives, pour qu'il puisse se focaliser sur l'écriture d'application en utilisant des motifs de code et de design communs dans le développement web.

En tant que framework web Python de haut niveau, Django permet le développement rapide de sites web sécurisés et faciles à maintenir.

Il permet de construire facilement des sites web efficaces en ajoutant des fonctions pour soutenir les opérations de caches et de bases de données.

La particularité de ce framework est qu'il couvre à la fois le front end et le back end. Il permet de gérer les bases de données et fonctionnalités back end en tant qu'outil Python, mais aussi d'éditer des fichiers et de créer des pages HTML comme un outil front end. [24]

#### 4.2.2.2 Bootstrap



Bootstrap fait partie de ces « [Front-End Framework](#) » contenant une collection d'outils utiles à la création et au design de sites et d'applications web en front end. Il contient des codes HTML et CSS ainsi que des extensions JavaScript en option, permettant la création de formulaires, de boutons, d'outils de navigations et d'autres éléments interactifs à intégrer dans une

application. Ce framework est pensé pour développer des designs interactifs adaptables à tout type d'écran, en particulier pour les smartphones. [25]

#### 4.2.3 intelligence artificielle

L'intelligence artificielle (IA) est un processus d'imitation de l'intelligence humaine qui repose sur la création et l'application d'algorithmes exécutés dans un environnement informatique dynamique. Son but est de permettre à des ordinateurs de penser et d'agir comme des êtres humains. [26]

L'usage de bibliothèques d'optimisation comme **Google OR-Tools** s'inscrit dans le domaine de l'intelligence artificielle.

#### 4.2.3.1 Moteur d'optimisation : Google OR-Tools

OR-Tools est une suite logicielle Open Source d'optimisation, conçue pour résoudre les problèmes les plus complexes du monde dans les domaines du calcul d'itinéraires des véhicules, des flux, de la programmation linéaire et entière, ainsi que de la programmation de contraintes. [27]

Pour la génération automatique des emplois du temps, la bibliothèque Google OR-Tools a été utilisée. Cet outil permet de modéliser les contraintes de planification et de trouver des solutions optimales via des algorithmes d'optimisation combinatoire. Elle prend en charge :

- La programmation par contraintes (CP-SAT Solver) ;
- La gestion fine de contraintes multiples (salles, disponibilités, etc.).

#### 4.2.4 Base de données

Le système de gestion de base de données retenu est SQLite pour le développement local, en raison de sa simplicité d'utilisation. Pour une version de production, un SGBD comme PostgreSQL peut être intégré pour une meilleure scalabilité.

#### 4.2.5 Outils complémentaires

##### 4.2.5.1 PyCharm



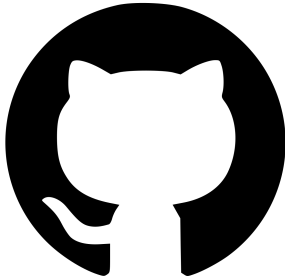
Développé par JetBrains, PyCharm est à ce jour l'IDE le plus populaire pour Python. Cette plateforme hybride est couramment utilisée pour le développement d'applications Python, et notamment par de grandes entreprises telles que Twitter, Facebook, Amazon ou Pinterest. Compatible avec Windows, Linux et macOS, PyCharm contient des modules et des packages aidant les développeurs à programmer des logiciels avec Python plus rapidement et avec moins

d'efforts. Il est aussi possible de le personnaliser pour répondre aux besoins précis d'un projet. [28]

### Qu'est-ce qu'un IDE ?

Un IDE ou environnement de développement intégré regroupe tous les outils requis pour écrire des logiciels. Il comporte notamment un éditeur et un compilateur. [28]

#### 4.2.5.2 Github



Git est un système de contrôle de version qui suit intelligemment les modifications apportées aux fichiers. Git est particulièrement utile lorsque vous et un groupe de personnes apportez tous des modifications aux mêmes fichiers en même temps.

Généralement, pour le faire dans un flux de travail basé sur Git, vous devez :

- Créez une branche à partir de la copie principale des fichiers sur lesquels vous (et vos collaborateurs) travaillez.
- Apportez des modifications aux fichiers de manière indépendante et sécurisée sur votre propre succursale personnelle.
- Laissez Git fusionner intelligemment vos modifications spécifiques dans la copie principale des fichiers, afin que vos modifications n'aient pas d'impact sur les mises à jour des autres.
- Laissez Git suivre vos modifications et celles des autres, afin que vous continuiez tous à travailler sur la version la plus récente du projet. [29]

#### 4.2.5.3 Star UML



StarUML est un logiciel de modélisation UML qui offre une interface utilisateur graphique conviviale et des fonctionnalités puissantes pour la création de différents types de diagrammes UML. Il prend en charge UML 2.x et propose une variété de fonctionnalités utiles pour les développeurs et les analystes pour concevoir et documenter leurs logiciels. L'utilisation de StarUML peut varier en

fonction des besoins spécifiques du projet et des préférences de l'utilisateur, mais il est généralement apprécié pour sa polyvalence et sa facilité d'utilisation dans le domaine de l'ingénierie logicielle. [30]

## 4.3 Implémentation des fonctionnalités clés

Le développement de l'application web intelligente s'est articulé autour de plusieurs fonctionnalités principales, Ces fonctionnalités ont été implémentées à l'aide du framework Django, qui offre un environnement robuste et structuré pour le développement d'applications web en Python.

### 4.3.1 Gestion des utilisateurs et des rôles

L'application propose une gestion fine des utilisateurs à travers différents rôles :

- **Administrateur** : gestion des ressources, des comptes utilisateurs, des emplois du temps.
- **Enseignant** : consultation et déclaration des indisponibilités, réservation de salles.
- **Étudiant** : consultation des emplois du temps selon le groupe affecté.

La gestion de l'authentification et des permissions repose sur le système intégré d'authentification de Django, complété par des décorateurs et des middleware personnalisés pour sécuriser l'accès aux différentes vues.

### 4.3.2 Déclaration des contraintes et indisponibilités

Chaque enseignant peut indiquer ses plages horaires d'indisponibilité via une interface dédiée. Ces contraintes sont automatiquement prises en compte lors de la génération des emplois du temps, évitant les conflits.

### 4.3.3 Génération automatique de l'emploi du temps

La génération automatique repose sur l'intégration de Google OR-Tools, une bibliothèque puissante d'optimisation sous contraintes. Les étapes suivies sont :

1. **Collecte des données** : matières, groupes, enseignants, salles, contraintes horaires.
2. **Modélisation du problème** : formulation des variables, contraintes (disponibilités, non-chevauchement, capacité des salles...).

3. **Lancement du solveur** : recherche d'une solution satisfaisant toutes les contraintes.
4. **Affichage et sauvegarde** : affichage des résultats dans l'interface web, avec possibilité de modifications manuelles.

#### **4.3.4 Réservation de salles et suivi**

Les enseignants peuvent également réserver des salles en dehors des créneaux d'enseignement via un module de réservation simple. Ce module vérifie automatiquement les disponibilités et empêche les conflits.

### **4.4 Sécurité, accessibilité**

La robustesse d'une application web ne se limite pas à ses fonctionnalités elle dépend aussi de sa sécurité, de son accessibilité pour tous les utilisateurs, et de ses performances en conditions réelles. Ces trois aspects ont été soigneusement pris en compte dans notre projet.

#### **4.4.1 Sécurité de l'application**

Pour garantir la confidentialité, l'intégrité et la disponibilité des données, plusieurs mécanismes de sécurité ont été intégrés :

- **Authentification et autorisation** :
  - Utilisation du système d'authentification de Django (mot de passe haché, sessions sécurisées).
  - Gestion des rôles pour restreindre l'accès aux fonctionnalités sensibles selon le profil (enseignant, étudiant, administrateur).
- **Protection contre les attaques courantes** :
  - Django protège nativement contre les attaques CSRF (Cross-Site Request Forgery).
  - Filtres contre les injections SQL via l'ORM de Django.
  - Échappement automatique des données pour prévenir les attaques XSS (Cross-Site Scripting).

## 4.4.2 Accessibilité

L'accessibilité a été prise en compte afin de rendre l'application utilisable par un public diversifié, notamment :

- **Responsive design** : l'interface s'adapte aux ordinateurs, tablettes et smartphones.
- **Navigation intuitive** : les menus sont clairs et les actions principales facilement identifiables.
- **Contrastes de couleurs et tailles de texte** : optimisés pour les utilisateurs malvoyants.
- **Utilisation d'icônes et de pictogrammes** : pour guider visuellement les utilisateurs.

## 4.5 Présentation de l'utilisation et de la manipulation du site à travers quelques interface

Nous allons faire une présentation de quelques interfaces qui sont intégrées à notre site.

### 4.5.1 Interface d'authentification

Cette interface permet aux utilisateurs de se connecter à la plateforme selon leur rôle (enseignant, étudiant, responsable département). Elle vérifie les identifiants via le système d'authentification intégré de Django.

Connexion

Matricule...

Mot de passe(Date de naissance)

Se connecter

Voulez vous être un gestionnaire d'établissement ? [Cliquez ici pour vous inscrire](#)

Gestionnaire ? [Accéder au panneau d'administration](#)

Figure 4.1 pages d'authentification enseignant, étudiant, responsable département

#### 4.5.1.1 Interface d'authentification d'administration

Cette interface permet aux administrateurs de se connecter à la plateforme d'administration et vérifie les identifiants via le système d'authentification intégré de Django.

A

Welcome

201936001363

....

Log in

Figure 4.2 pages d'authentification administrateurs

#### 4.5.1.2 Interface d'inscription d'un gestionnaire établissement

Le demandeur remplir la formule et l'envoie à l'administrateur par email et attente une reponse.

## Demande de création de compte Gestionnaire Établissement

Pour vous inscrire, veuillez :

1. Télécharger et remplir le formulaire ci-dessous
2. Envoyer le DOCX complété à l'un des administrateurs généraux :
  - o Amani Btm — amaniboutamine@gmail.com

[Télécharger le formulaire d'inscription \(DOCX\)](#)

Figure 4.3 pages d'inscription d'un gestionnaire

### 4.5.1.2.1 Interface de formulaire à remplir par gestionnaire

Le demandeur d'inscription pour devenir un gestionnaire d'établissement doit remplir toutes les informations demandées dans le formulaire ci-dessous

#### Formulaire d'Inscription

##### 1. Informations Utilisateur

Nom	.....
Prénom	.....
Date de naissance	.....
Matricule (Pièce d'identité)	.....
Email	.....
Téléphone	.....

##### 2. Informations Établissement

Nom de l'Établissement	.....
Type (Université, Lycée, CEM, Primaire, Autre)	.....
Adresse	.....

Figure 4.4 pages de formulaire d'inscription

## 4.5.2 les pages d'accueil

La page d'accueil elle offre la possibilité de visualiser les différents services des utilisateurs (**enseignant, étudiant, responsable département**) comme c'est présenté dans les figures ci-dessous.

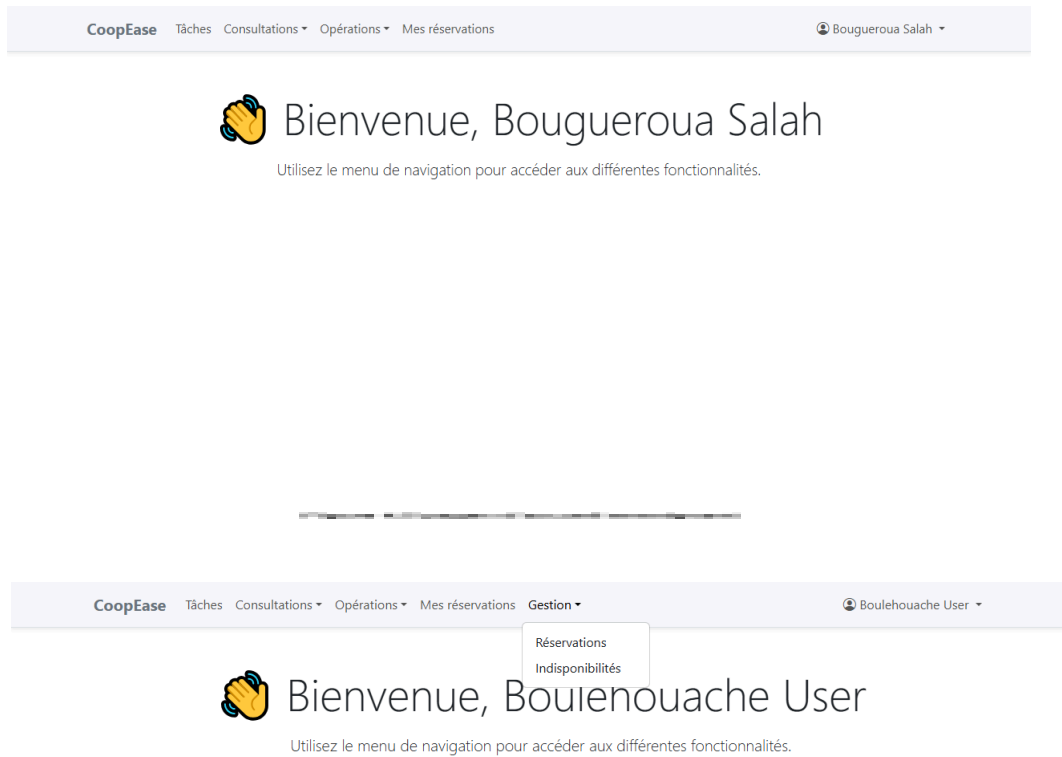


Figure 4.6 pages d'accueil responsable département



Bienvenue, Amani Boutamine

Utilisez le menu de navigation pour accéder aux différentes fonctionnalités.

Figure 4.7 pages d'accueil étudiant

### 4.5.3 Consultation de l'emploi du temps par les étudiants

Chaque étudiant peut consulter son emploi du temps selon son groupe ou sa section, L'interface est responsive et permet un accès rapide aux informations.

CoopEase Emplois du temps Affichage Modules Enseignants Groupe Amani Boutamine

Emploi du Temps

Année scolaire: 2025/2026

S1 S2

Mon Groupe Ma spécialité Département


SI - Licence3- Groupe 1

Jour/Heure	08:00	09:30	11:00	12:30	14:00	15:30
SAMEDI	TP-AppMob/ G1/ STP5/ Kara	C-SecInfo Nafir Amphi1	C-IA Hazmoune Amphi1	C-DSStruc Mansoul Amphi1	TD-RédSci/ G1/ STD15/ Nabet	C-CDStartup Mazouzi EnligneEnligne
DIMANCHE	TP-DSStruc/ G1/ STPA2/ Mansoul	C-AppMob Kara Amphi4	TD-SecInfo/ G1/ STDA10/ Nafir	TP-IA/ G1/ STP5/ Hazmoune		

Figure 4.8 l'emploi du temps d'étudiant

### 4.5.4 Consultation de l'emploi du temps par l'enseignant

L'enseignant dispose d'une interface dédiée lui permettant de consulter son emploi du temps personnel de manière claire et structurée.

 Emploi du Temps

Taches de travail de M.Bougueroua

Jour/Heure	08:00	09:30	11:00	12:30	14:00	15:30
DIMANCHE	ISIL - SecInfo TD / Salle: TD39 Section1 / G4					
LUNDI	ISIL - SecInfo TD / Salle: TD24 Section1 / G3					
MARDI	ISIL - SecInfo TD / Salle: TD17 Section1 / G1	ISIL - SecInfo Cours / Salle: Sallelec1 Section1				
MERCREDI	ISIL - SecInfo TD / Salle: TD23 Section1 / G2					

Figure 4.9 l'emploi du temps de l'enseignant

#### 4.5.5 Interface de gestion des disponibilités enseignant

Cette interface permet à un enseignant d'indiquer ses plages d'indisponibilité. Ces informations sont ensuite prises en compte lors de la génération automatique de l'emploi du temps s'ils sont approuvés par le responsable d'établissement.

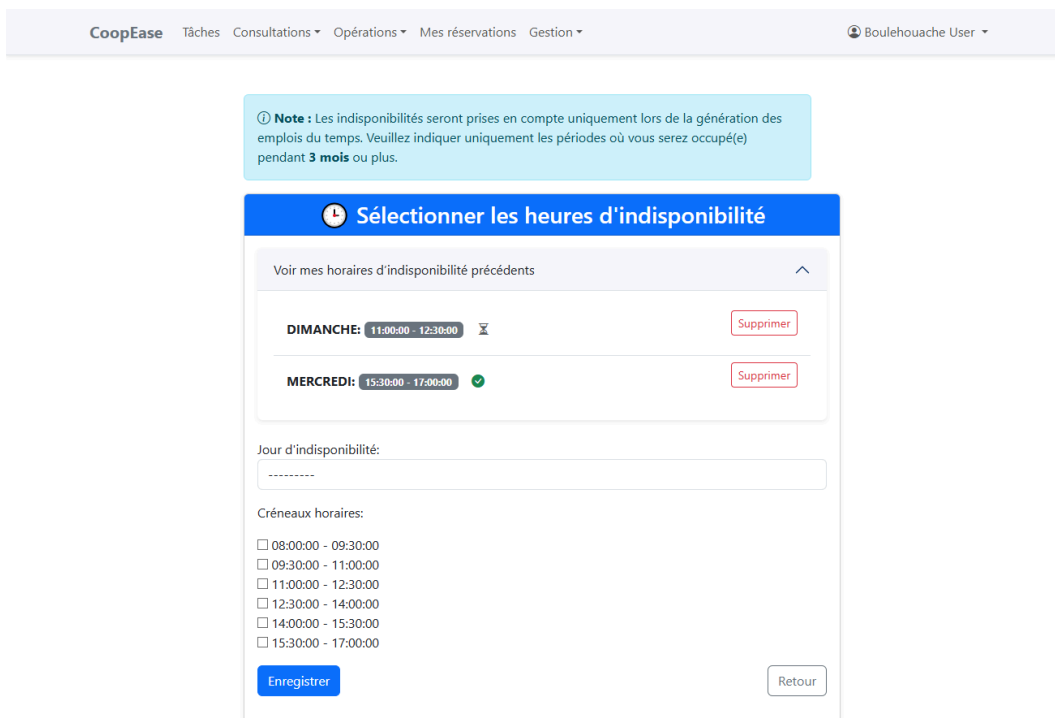


Figure 4.10 l'interface de gestion d'indisponibilités d'enseignants

#### 4.5.6 Interface des demandes des disponibilités des enseignants

Responsable de département dispose d'un tableau de bord spécifique lui permettant de consulter, approuver ou refuser chaque demande, en tenant compte des impératifs pédagogiques et organisationnels du département.

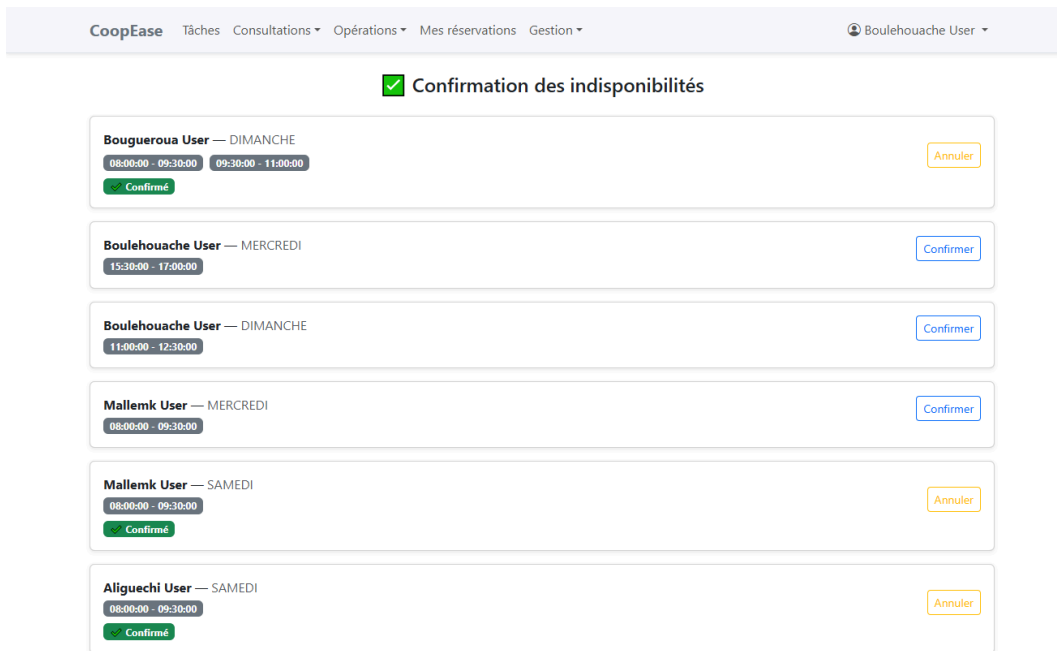


Figure 4.11 l'interface de confirmation d'indisponibilités d'enseignants

## 4.5.7 Interface d'annonce d'une absence par un enseignant

Cette interface permet à un enseignant de signaler une absence imprévue, via la plateforme.

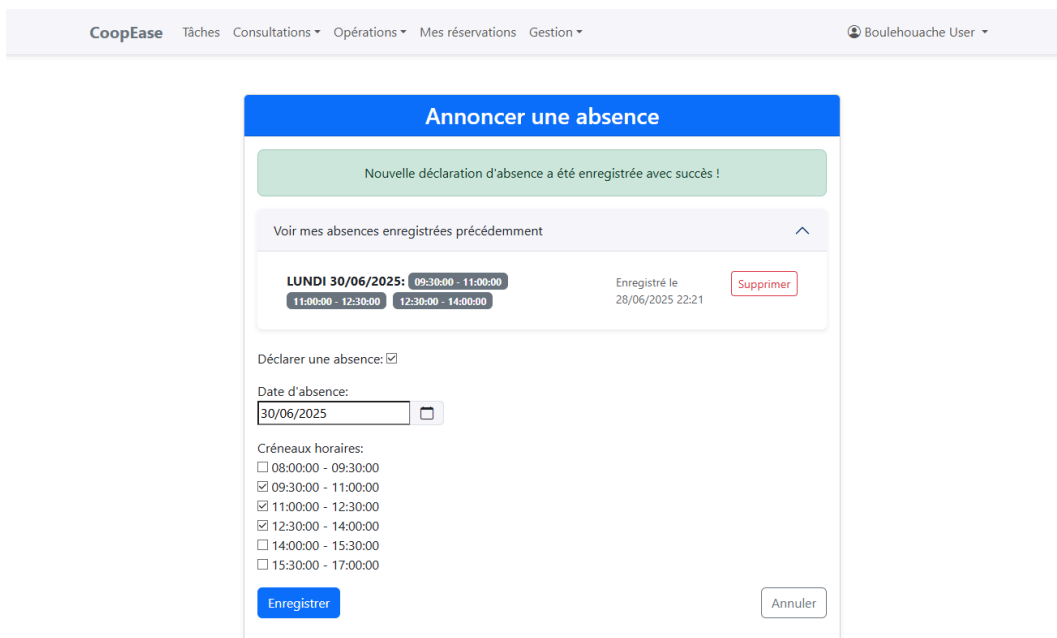




Figure 4.12 l'interface d'annoncer une absence d'un enseignant

## 4.5.8 Consultations la disponibilité des salles

Cette interface offre aux enseignants la possibilité de consulter, à tout moment, la disponibilité des salles au sein de l'établissement. Elle répond à plusieurs besoins, notamment la recherche de créneaux libres pour organiser des séances de rattrapage, des travaux dirigés supplémentaires ou remplacer un cours annulé.

 DIMANCHE

Heure	Espaces disponibles																			
08:00-09:30	1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi)																			
09:30-11:00	1 (Sallelec) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) A3 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD)																			
11:00-12:30	A10 (TD) 1 (Sallelec) 3 (Amphi) 2 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 18 (TD) 16 (TD)																			
12:30-14:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 38 (TD) 37 (TD) 23 (TD) 21 (TD) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			
14:00-15:30	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 23 (TD) 22 (TD) 83 (TP) 5 (TP) 4 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			
15:30-17:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) 3 (TP) A3 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			

 LUNDI

Heure	Espaces disponibles																			
08:00-09:30	1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi)																			
09:30-11:00	A10 (TD) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 3 (TP) A3 (TP) A2 (TP) 2 (TP) 20 (TD) 19 (TD) 18 (TD)																			
11:00-12:30	1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 37 (TD) 23 (TD) 22 (TD) 21 (TD)																			
12:30-14:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 21 (TD) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 14 (TD) 13 (TD)																			
14:00-15:30	salle (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) A3 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			
15:30-17:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) 3 (TP) A3 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			


 MARDI

Figure 4.13 l'interface pour consulte les salles disponibles

## 4.5.9 Réserveation de salle

Les enseignants peuvent soumettre une demande de réserveation ponctuelle de salle en cas de besoin (examen, session de rattrapage, etc.).

### 4.5.9.1 Réserveation des salles toujours disponible dans emploi du temps

Salle	Type	Jour	Heure	Module	Type module	Groupe	Section	Dates Réserveées	Action
13	TD	MERCREDI	08:00	-- Choisissez un module --	-- Typ	Groupe	Section	02-07-2025 (Votre réserveation)	mm/dd/yyyy Réserver
14	TD	MERCREDI	08:00	AppMob	TP	2	1		07/02/2025 Réserver
15	TD	MERCREDI	08:00	-- Choisissez un module --	-- Typ	Groupe	Section		mm/dd/yyyy Réserver
17	TD	MERCREDI	08:00	-- Choisissez un module --	-- Typ	Groupe	Section		mm/dd/yyyy Réserver

Figure 4.14 Interface de réserveation des salles toujours disponibles

### 4.5.9.2 Réserveation des salles occasionnellement disponible

Ce sont les salles ou sont libre laire ce qu'un enseignant et absente.

Salle	Type	Date	Jour	Heure	Module	Type module	Groupe	Section	Status	Action
49	TD	01-07-2025	MARDI	08:00	-- Choisissez un module --	-- Type	Groupe	Section	Réserveé	Indisponible
39	TD	01-07-2025	MARDI	09:30	-- Choisissez un module --	-- Type	Groupe	Section	Disponible	Réserver
49	TD	01-07-2025	MARDI	12:30	-- Choisissez un module --	-- Type	Groupe	Section	Disponible	Réserver
2	TP	01-07-2025	MARDI	14:00	-- Choisissez un module --	-- Type	Groupe	Section	Disponible	Réserver

Figure 4.15 Interface de réserveation des salles occasionnellement disponible

### 4.5.9.3 Interface de gestion des Réservations des salles

Les demandes de réservation des salles seront approuvées ou rejeter par le responsable de département

The screenshot shows the 'CoopEase' interface for managing room reservations. At the top, there is a navigation bar with 'CoopEase' on the left and a user profile 'Boulehouache User' on the right. Below the navigation bar, the page title is 'Réservations pas encore Approuvées'. This is followed by a table with two rows of pending reservations. Each row includes columns for 'Enseignant', 'Salle', 'Date', 'Jour', 'Heure', 'Module', 'Type', 'Groupe', 'Section', 'Date de Demande', and 'Actions'. The 'Actions' column contains 'Approuver' (green) and 'Rejeter' (red) buttons. Below this, the page title changes to 'Réservations Approuvées', followed by a table with one row of an approved reservation. The 'Actions' column for this row contains an 'Annuler' (yellow) button.

Enseignant	Salle	Date	Jour	Heure	Module	Type	Groupe	Section	Date de Demande	Actions
Boulehouache User	49 (TD)	July 1, 2025	MARDI	08:00	AppMob	Cours	Tous	Section1	28/06/2025 22:19	<button>Approuver</button> <button>Rejeter</button>
Bougueroua User	13 (TD)	June 26, 2025	JEUDI	08:00	SecInfo	Cours	1	Section1	19/06/2025 11:05	<button>Approuver</button> <button>Rejeter</button>

Enseignant	Salle	Date	Jour	Heure	Module	Type	Groupe	Section	Date de Demande	Actions
Boulehouache User	13 (TD)	July 2, 2025	MERCREDI	08:00	AutoComp	Cours		Section1	23/06/2025 14:04	<button>Annuler</button>

Figure 4.16 Interface de la gestion des réservations des salles

### 4.5.10 interface d'historique des réservations des salles

Cette interface permet aux enseignants la possibilité de consulter, à tout moment, la réservation de salle si elle est approuvée ou non.

### Mes Réservations

**Réservation De Salle** 28-06-2025 22:19

Salle réservé: 49 (TD) - Informatique  
Date: 2025-07-01 | Jour: MARDI  
Heure: 08:00:00 - 09:30:00  
Module: AppMob | Type: Cours  
Groupe: Tous | Section: Section1

**En attente**

Annuler

**Réservation De Salle** 23-06-2025 14:04

Salle réservé: 13 (TD) - Informatique  
Date: 2025-07-02 | Jour: MERCREDI  
Heure: 08:00:00 - 09:30:00  
Module: AutoComp | Type: Cours  
Groupe: | Section: Section1

**Confirmé**

Annuler

Figure 4.17 Interface de l'historique des réservations de salle

## 4.5.11 interface d'affichages pour étudiant

Cette interface est destinée aux étudiants et leur permet de rester informés en temps réel des changements affectant leur emploi du temps. Elle regroupe toutes les informations relatives aux absences des enseignants et aux séances de remplacement programmées

## Tableau d'Affichage

Séance ajoutée
<p><b>Spécialité:</b> SI - Licence3</p> <p><b>Professeur:</b> Nafir User</p> <p><b>Salle:</b> 1 (Amphi)</p> <p><b>Module:</b> SeclInfo - TD</p> <p><b>Jour:</b> MARDI</p> <p><b>Heure:</b> 08:00</p> <p><b>Groupe:</b> 1</p> <p><b>Section:</b> Section1</p> <p><b>Date:</b> 01-07-2025</p> <p>Ajouté le: 29-06-2025 17:22</p>
Absence
<p><b>Professeur:</b> Boulehouache User</p> <p><b>Date d'absence:</b> 30-06-2025</p> <p><b>Créneaux horaires:</b> 09:30, 11:00, 12:30</p> <p>Ajouté le: 28-06-2025 22:21</p>

Figure 4.18 Tableau d'affichage pour étudiant

#### 4.5.12 Interface de demande d'intégration à un groupe par un étudiant

Cette interface est conçue pour permettre aux étudiants de soumettre une demande d'intégration à un groupe spécifique. Elle facilite la gestion des affectations tout en offrant un canal de communication formel entre l'étudiant et l'administration pédagogique.

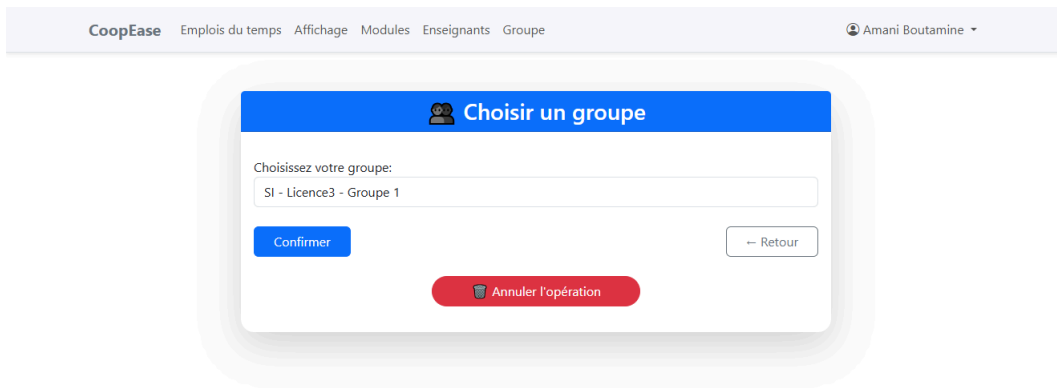


Figure 4.19 Interface de choix du groupe par un étudiant

### 4.5.13 Interface du profil

Les utilisateurs peuvent Consulter le profil et voir ces informations

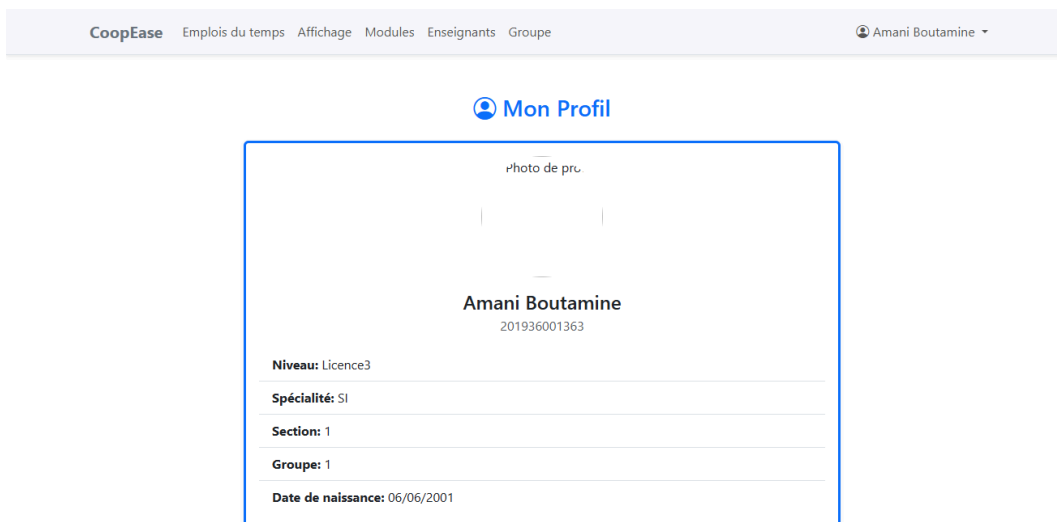


Figure 4.20 Interface du profil d'étudiant

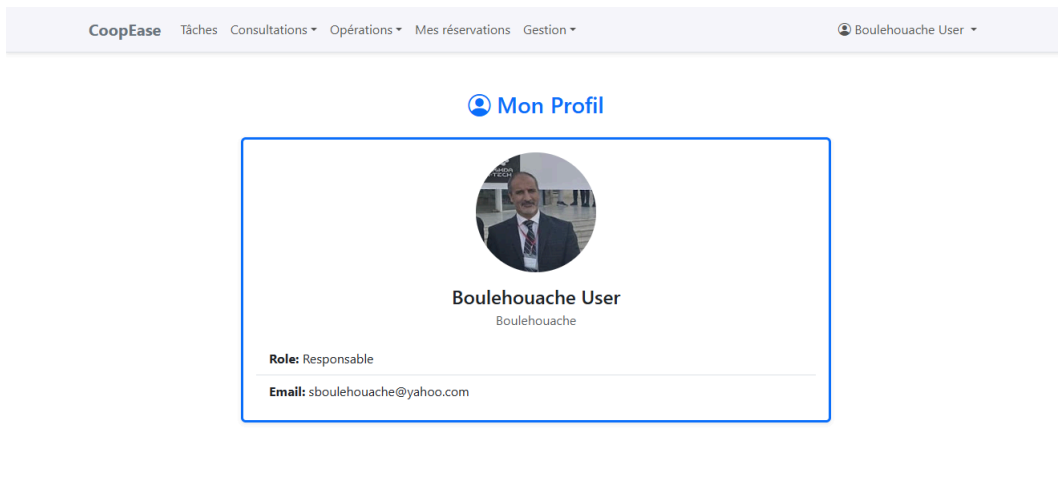


Figure 4.21 Interface du profil enseignant

#### 4.5.14 Interface des modules

Les utilisateurs peuvent Consulter les modules et voir les informations qui le concerne chaque module.

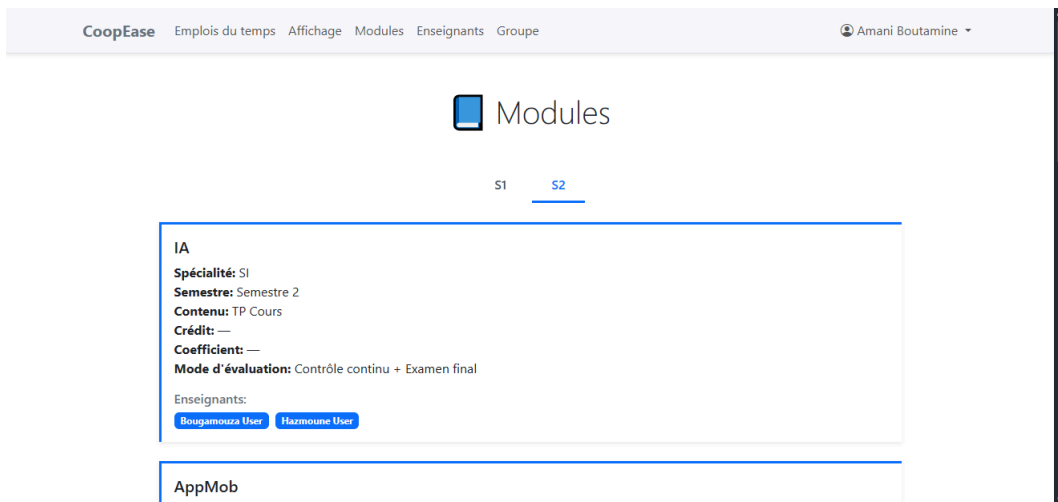


Figure 4.22 Interface modules par d'étudiant

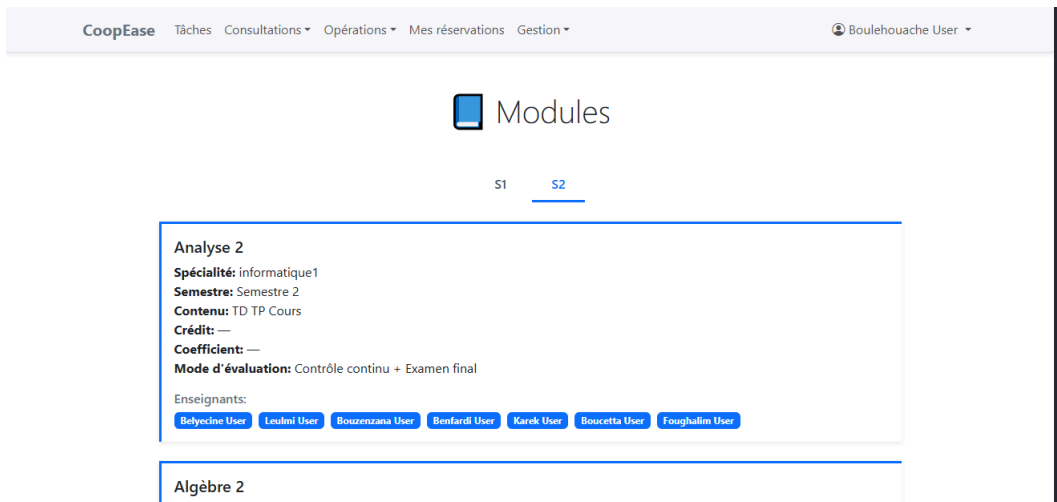


Figure 4.23 Interface modules par enseignant

#### 4.5.15 Interface liste des enseignants

Les enseignants et les étudiants peuvent voir la liste des enseignants en cas de besoin et les contactent par leurs emails ou numéros de téléphone.

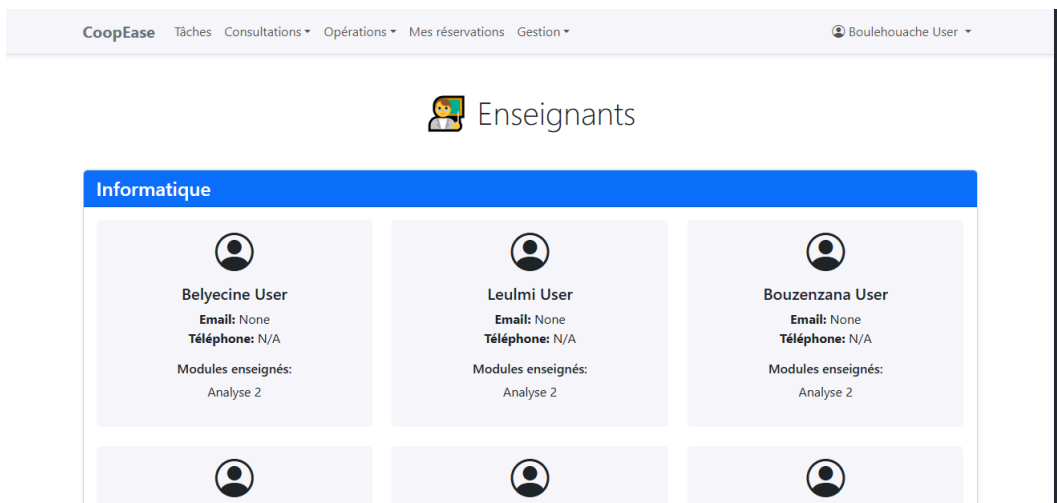


Figure 4.24 Interface liste des enseignants

## Enseignants - SI

Semestre S1 Semestre S2










 <b>Bougamouza User</b> Email: None Téléphone: N/A Modules enseignés: IA	 <b>Hazmoune User</b> Email: None Téléphone: N/A Modules enseignés: IA	 <b>Kara User</b> Email: None Téléphone: N/A Modules enseignés: AppMob
 <b>Boulehouache User</b> Email: sboulehouache@yahoo.com Téléphone: N/A Modules enseignés: AppMob	 <b>Foughali User</b> Email: None Téléphone: N/A Modules enseignés: DSStruc	 <b>Mansoul User</b> Email: None Téléphone: N/A Modules enseignés: DSStruc
 <b>Nafir User</b> Email: None Téléphone: N/A Modules enseignés: SeclInfo	 <b>Nabet User</b> Email: None Téléphone: N/A Modules enseignés: RédSci	 <b>Mazouzi User</b> Email: None Téléphone: N/A Modules enseignés: CDStartup

Figure 4.25 Interface liste des enseignants par étudiant

## 4.5.16 Interface d'Équipements disponibles

Les enseignants peuvent voir l'équipement disponible dans le département en temps réel en cas de besoin.

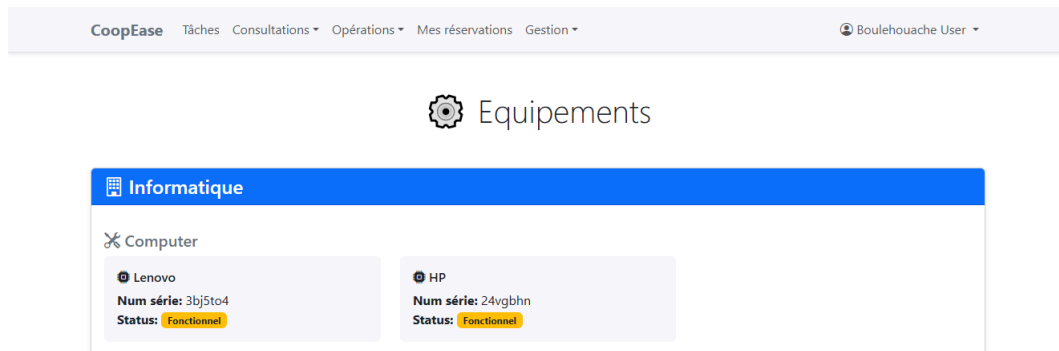


Figure 4.26 Interface liste d'équipements

## 4.5.17 Tableau de bord de l'administrateur pour la création des groupes

Ce tableau de bord fournit à l'administrateur un espace dédié pour créer, configurer et organiser les groupes pédagogiques nécessaires à la structuration des emplois du temps et à l'affectation des étudiants. Il constitue une étape préalable indispensable au bon fonctionnement des algorithmes de génération d'emploi du temps et d'affectation automatique gérés par Google OR-Tools.

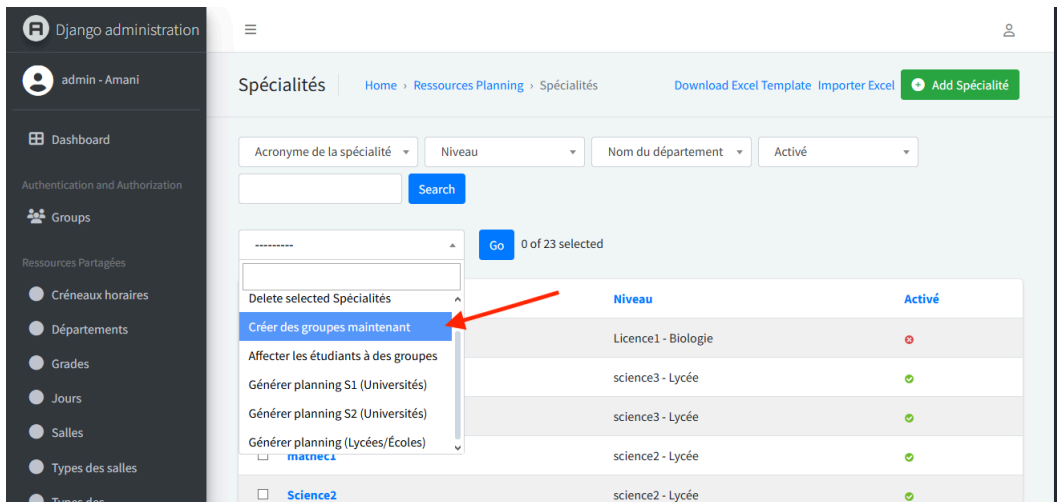


Figure 4.27 Interface de tableau de bord de l'administrateur et déclenchement de création des groupes

#### 4.5.18 Tableau de bord de l'administrateur génération des emplois du temps

Le tableau de bord offre une vue centralisée sur la gestion des ressources (salles, enseignants, groupes, horaires, modules, spécialités). Il permet aussi de déclencher la génération automatique des emplois du temps via l'outil OR-Tools.

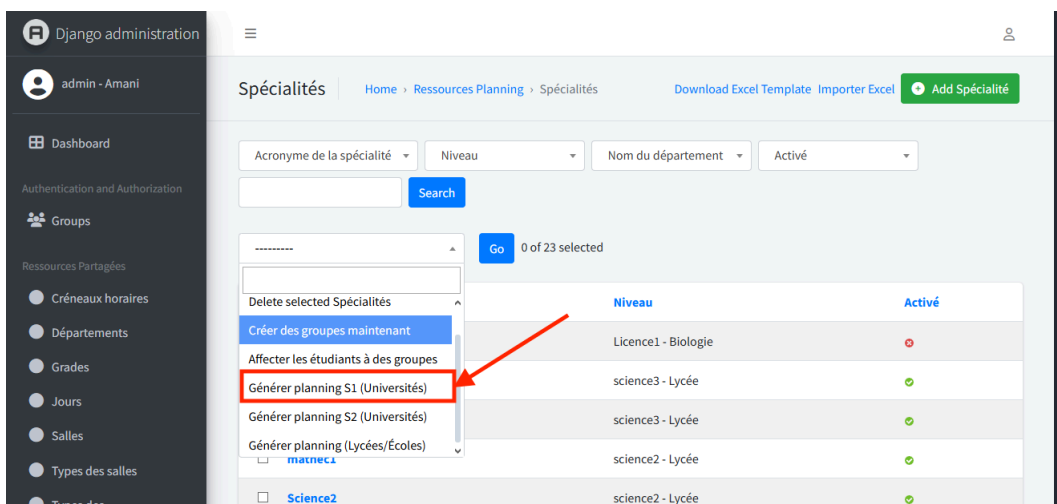


Figure 4.28 Interface de tableau de bord de l'administrateur et déclenchement de la génération automatique des emplois du temps

## 4.5.19 Tableau de bord de l'administrateur pour l'affectation des étudiants aux groupes

Cette interface permet à l'administrateur de superviser l'affectation automatique des étudiants aux groupes pédagogiques, réalisée à l'aide du moteur d'optimisation **Google OR-Tools**. L'affectation tient compte de plusieurs paramètres : disponibilités, préférences, équilibre des effectifs, contraintes pédagogiques et capacités des groupes.

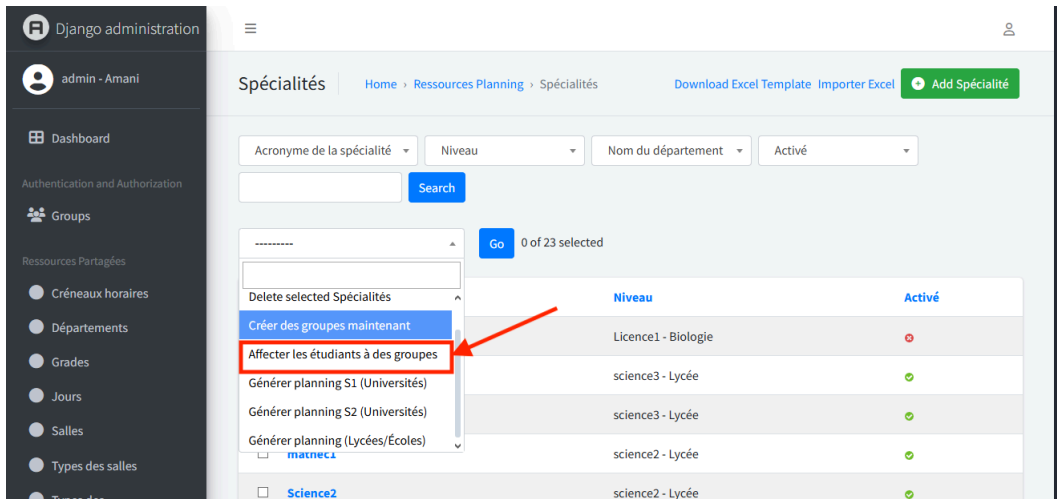


Figure 4.29 Interface de l'administrateur pour faire affectation des étudiants à des groupes

## 4.5.20 Tableau de bord de l'administrateur

Le tableau de bord de l'administrateur constitue une interface centrale permettant aux responsables (gestionnaires ou administrateurs système) d'accéder aux différentes fonctionnalités de gestion, de suivi, et de modification des données du système. Cette interface joue un rôle stratégique dans le maintien de la cohérence, la supervision des responsables dès l'établissement des opérations et l'adaptation rapide aux imprévus (changements d'enseignant, changements d'affectation, conflits d'emploi du temps...).

## 4.6 Conclusion

Ce chapitre a présenté en détail la phase d'implémentation technique de notre application web intelligente dédiée à la gestion coopérative des ressources pédagogiques. En s'appuyant sur des technologies modernes

comme Python, Django, Bootstrap et Google OR-Tools, nous avons pu concevoir un système à la fois robuste, interactif et évolutif.

L'intégration du moteur d'optimisation OR-Tools a permis d'automatiser efficacement la génération des emplois du temps, tout en prenant en compte les contraintes réelles du milieu éducatif. Les fonctionnalités développées telles que la gestion des utilisateurs, la réservation de salles, le suivi des disponibilités et des absences, ainsi que l'affichage dynamique des plannings ont été pensées pour répondre aux besoins spécifiques des différents acteurs : administrateurs, enseignants, étudiants et responsables pédagogiques.

La validation du système, à travers une série de tests fonctionnels et scénarios réalistes, a permis de confirmer sa fiabilité, sa facilité d'utilisation et son potentiel d'adaptation à des environnements diversifiés. L'interface utilisateur a été conçue pour être accessible, intuitive et compatible avec différents supports, garantissant ainsi une expérience fluide pour tous les profils d'utilisateurs.

En somme, cette étape d'implémentation confirme la faisabilité technique de notre solution et constitue une base solide pour un déploiement opérationnel. Le prochain chapitre abordera les conclusions générales du projet, les limites rencontrées et les perspectives d'évolution envisageables.

## Conclusion générale

Dans un contexte où la gestion des ressources partagées devient un enjeu stratégique pour les établissements éducatifs, ce mémoire a proposé une solution innovante alliant intelligence artificielle, technologies web modernes et approche coopérative. Partant d'un constat réel de dysfonctionnements dans la planification manuelle ou semi-automatisée des emplois du temps, notre travail s'est inscrit dans une démarche d'ingénierie logicielle rigoureuse visant à concevoir, développer et valider une application web intelligente de gestion coopérative des ressources.

Après une étude approfondie des besoins, contraintes et approches existantes, nous avons fait le choix d'intégrer la bibliothèque Google OR-Tools comme moteur d'optimisation sous contraintes, au sein d'un système web développé avec Python et Django. Cette combinaison nous a permis d'automatiser la génération des emplois du temps, de réduire les conflits d'allocation, et de garantir une meilleure équité dans la répartition des ressources.

Le système développé prend en charge différents profils d'utilisateurs – administrateurs, enseignants, étudiants et propose une interface conviviale, accessible sur divers appareils, facilitant la déclaration des contraintes, la consultation des plannings, la réservation de salles, et la gestion des imprévus. Les tests fonctionnels ont montré que la solution répond de manière satisfaisante aux exigences identifiées en matière d'efficacité, de fiabilité et d'utilisabilité.

L'évaluation de la performance du moteur d'optimisation mériterait d'être approfondie. De plus, certaines fonctionnalités comme la gestion dynamique des absences avec remplacement automatique pourraient faire l'objet d'un développement futur.

En perspective, plusieurs pistes peuvent être envisagées pour améliorer et étendre la solution proposée : intégration de l'apprentissage automatique pour prédire les contraintes récurrentes, extension du système à d'autres types de structures (hôpitaux, centres de formation...), ou encore internationalisation de l'interface.

En somme, ce mémoire apporte une contribution concrète à la transformation numérique de la gestion pédagogique, en démontrant que l'intelligence artificielle et les technologies collaboratives peuvent

efficacement soutenir les processus organisationnels des établissements éducatifs.

## Bibliographie

- [1] <https://vitrinelinguistique.oqlf.gouv.qc.ca/fiche-gdt/fiche/8370901/ressource-partagee>
- [2] La gestion coopérative : théorie et pratique: De la spécificité à l'avantage concurrentiel - Émergence d'un nouveau paradigme De Daniel Côté  
[https://www.google.dz/books/edition/La\\_gestion\\_coop%C3%A9rative\\_th%C3%A9orie\\_et\\_prat/4N9eEAAAQBAJ?hl=fr&gbpv=1&printsec=frontcover](https://www.google.dz/books/edition/La_gestion_coop%C3%A9rative_th%C3%A9orie_et_prat/4N9eEAAAQBAJ?hl=fr&gbpv=1&printsec=frontcover)
- [3]  
<https://fr.linkedin.com/advice/0/how-do-you-manage-conflicts-arising-from-resource-29t7e?lang=fr>
- [4][https://depinfoskikda.com/depinfoskikdalaravel/public/student\\_emploi\\_du\\_temps](https://depinfoskikda.com/depinfoskikdalaravel/public/student_emploi_du_temps)
- [5][https://www.researchgate.net/publication/235417909\\_Conception\\_d'un\\_systeme\\_pour\\_l'emploi\\_du\\_temps\\_universitaire](https://www.researchgate.net/publication/235417909_Conception_d'un_systeme_pour_l'emploi_du_temps_universitaire)
- [6] <https://www.agence.coop/media/579/download?attachment>
- [7] [http://observatoire.tic.free.fr/pdf/dossiersThematiques/5\\_Travail\\_collaboratif.pdf](http://observatoire.tic.free.fr/pdf/dossiersThematiques/5_Travail_collaboratif.pdf)
- [8]<https://dspace.ummo.dz/server/api/core/bitstreams/ecc471f4-a1cd-47ca-98dd-ee0b97c98157/content>
- [9] <https://www.univ-montp3.fr/fr/node/186800/pdf>
- [10] <https://examino.fr/blog/ia/outils-ia-planification-cours>
- [11] <https://blog.hubspot.fr/website/programmation-orientee-objet>
- [12] <http://info.arqendra.net>
- [13] <http://www.iro.umontreal.ca/~dift6825/UML.htm>
- [14] <https://developers.google.com/optimization?hl=fr>
- [15] <https://www.cosling.com/fr/cpsat>
- [16] <https://www.cosling.com/fr/choco-solver#constraint-programming>
- [17] <https://appmaster.io/fr/glossary/architecture-en-couches>

- [18] <https://www.lucidchart.com/blog/fr/types-de-diagrammes-UML>
- [19] <https://www.python.org/doc/essays/blurb/>
- [20] <https://habefast.ch/glossaire/html/>
- [21] <https://datascientest.com/css3-tout-savoir>
- [22] <https://datascientest.com/css3-tout-savoir>
- [23] <https://www.pure-illusion.com/lexique/definition-de-framework#:~:text=Un%20Framework%20est%20une%20boite.web%20ou%20d'une%20application.>
- [24] <https://datascientest.com/django-tout-savoir>
- [25] <https://datascientest.com/bootstrap-creez-votre-page-web>
- [26] <https://www.netapp.com/fr/artificial-intelligence/what-is-artificial-intelligence/>
- [27] <https://developers.google.com/optimization?hl=fr>
- [28] <https://datascientest.com/pycharm-tout-savoir>
- [29] <https://docs.github.com/en/get-started/start-your-journey/about-github-and-git>
- [30] <https://www.tresfacile.net/le-logiciel-de-modelisation-startuml/>

# Annexe



الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث العلمي



Ministère de L'enseignement Supérieur et de la Recherche Scientifique

Université 20 Aout 1955 Skikda  
Faculté Des Sciences  
Département D'informatique

جامعة 20 أوت 1955 سكيكدة  
كلية العلوم  
قسم الإعلام الآلي

Projet de fin : d'étude dans le cadre de l'article ministériel 1275 en vue de l'obtention du Diplôme Master informatique

Spécialité : Réseaux et Systèmes Distribués

## THÈME

# **CoopEase : Une Application Web Intelligente de Gestion Coopérative des Ressources Pédagogiques**

Réalisé par :

- Laifa Hocine Chiheb
- Boutamine Amani

Encadré par :

- Dr. Boulehouache Soufiane

Co-encadré par :

- Dr. Bougueroua Salah

Année Universitaire 2024 – 2025

# Introduction du Projet

## Carte d'Information

### Équipe de Supervision

<b>Nom :</b>	<b>Boulehouch Soufiane</b>	<b>Bougueroua Salah</b>
<b>Rôle :</b>	Superviseur principal	Superviseur secondaire
<b>Spécialisation :</b>	Intelligence Artificielle	Systèmes Informatiques

**Table 1 :** Équipe de Supervision

### Équipe de Travail

<b>Nom de l'étudiant :</b>	<b>Laifa Hocine Chiheb</b>	<b>Boutamine Amani</b>
<b>Spécialisation :</b>	Réseaux et systèmes distribués	Réseaux et systèmes distribués
<b>Faculté :</b>	Informatique	Informatique

**Table 2 :** Équipe de Travail

# Table des matières

Liste des figures

Liste des tableaux

<u>Table des matières</u>	<u>3</u>
<u>Introduction Générale</u>	<u>8</u>
<u>1. L'idée du projet</u>	<u>10</u>
<u>2. Les valeurs suggérées</u>	<u>10</u>
<u>2.1 Pour les étudiants</u>	<u>11</u>
<u>2.2 Pour les enseignants</u>	<u>11</u>
<u>2.3 Pour l'administration</u>	<u>11</u>
<u>3. L'équipe</u>	<u>12</u>
<u>4. Les objectifs du projet</u>	<u>12</u>
<u>5. Le planning de réalisation du projet</u>	<u>13</u>
<u>Axe 2 : Aspects innovants</u>	<u>14</u>
<u>1. La nature des innovations</u>	<u>14</u>
<u>1.1 innovation technique</u>	<u>14</u>
<u>1.2 Innovation fonctionnelle</u>	<u>14</u>
<u>1.3 Innovation d'automatisation</u>	<u>14</u>
<u>1.4 Innovation d'usage</u>	<u>14</u>
<u>2. Les domaines d'innovation</u>	<u>14</u>
<u>2.1 Transformation numérique de l'éducation</u>	<u>14</u>
<u>2.2 Innovation pédagogique</u>	<u>15</u>
<u>2.3 Innovation organisationnelle</u>	<u>15</u>
<u>2.4 Innovation inclusive</u>	<u>15</u>
<u>3. Comparaison entre la Création Manuelle et Automatique d'un Emploi du Temps</u>	<u>15</u>
<u>Axe 3 : Analyse stratégique du marché</u>	<u>16</u>
<u>1. Le segment du marché</u>	<u>16</u>
<u>2. Analyse SWOT</u>	<u>16</u>
<u>3. Intensité de la concurrence</u>	<u>17</u>
<u>4. La stratégie marketing</u>	<u>18</u>
<u>4.1 Approche B2B ciblée</u>	<u>18</u>

4.2 Valorisation des bénéfices concrets	18
4.3 Stratégie de notoriété progressive	18
4.4 Modèle économique simple et accessible	18
4.5 Partenariats stratégiques	19
<b>Axe 4 : Plan de production et organisation</b>	<b>20</b>
1. Le processus de development	20
1.1 Analyse des besoins	20
1.2 Conception technique	20
1.3 Développement des modules	20
1.4 Phase de tests	21
1.5 Mise en oeuvre	21
2. L'approvisionnement	21
2.1 Ressources logicielles	21
2.2 Ressources techniques	21
2.3 Ressources humaines	22
2.4 Services externes (à prévoir)	22
3. La main d'œuvre	22
3.1 Phase de développement initial	22
3.2 Phase de déploiement et d'évolution	23
3.3 Vision collaborative	23
4. Les principaux partenaires	23
4.1 Partenaires académiques	23
4.2 Partenaires institutionnels	24
4.3 Partenaires techniques	24
4.4 Partenaires utilisateurs	24
<b>Axe 5 : Plan financier</b>	<b>25</b>
1. Les coûts et les charges	25
1.1 Coûts de développement	25
1.2 Coûts d'exploitation	25
1.3 Coûts de communication	25
1.4 Coûts juridiques et administratifs	25
2. Le chiffre d'affaires	26
2.1 Modèle B2B (Business to Business)	26
2.2 Modèle évolutif	26

3. Les comptes de résultats escomptés (prévisionnels)	26
4. Le plan de trésorerie	26
Axe 6 : Prototype Expérimental	28
1. Structure et Principe de Fonctionnement du Système	28
2. Les interfaces	28
2.1 les pages d'accueils	28
2.2 Interface d'authentification	30
2.3 Interface d'authentification d'administration	31
2.4 Interface d'inscription d'un gestionnaire établissement	31
2.4.1 Interface de formulaire à remplir par gestionnaire	32
2.5 Consultation de l'emploi du temps	32
2.6 Interface de gestion des disponibilités enseignant	33
2.7 Interface des demandes des disponibilités des enseignants	34
2.8 Interface d'annonce d'une absence par un enseignant	35
2.9 Consultations la disponibilité des salles	36
2.10 Réserveation de salle	37
2.10.1 Réserveation des salles toujours disponible dans emploi du temps	38
2.10.2 Réserveation des salles occasionnellement disponible	38
2.11 Interface de gestion des Réserveation des salles	38
2.12 interfaces d'historique des réservations des salles	39
2.13 interfaces d'affichages pour étudiant	39
2.14 Interface de demande d'intégration à un groupe par un étudiant	40
2.15 Interface du profil	41
2.16 Interface des modules	42
2.17 Interface liste des enseignants	43
2.18 Interface d'Equipements disponible	45
2.19 Tableau de bord de l'administrateur pour la création des groupes	45
2.20 Tableau de bord de l'administrateur pour génération de l'emploi du temps	46
2.21 Tableau de bord de l'administrateur pour l'affectation des étudiants aux groupes	46
2.22 Tableau de bord de l'administrateur	47
BMC	48
Conclusion Générale	49

## Liste des figures

[Figure 6.1 pages d'accueil enseignant](#)

[Figure 6.2 pages d'accueil responsable département](#)

[Figure 6.3 pages d'accueil étudiant](#)

[Figure 6.4 pages d'authentification enseignant, étudiant, responsable département](#)

[Figure 6.5 pages d'authentification administrateurs](#)

[Figure 6.6 pages d'inscription d'un gestionnaire](#)

[Figure 6.7 pages de formulaire d'inscription](#)

[Figure 6.8 l'emploi du temps d'étudiant](#)

[Figure 6.9 l'emploi du temps de l'enseignant](#)

[Figure 6.10 l'interface de gestion d'indisponibilités d'enseignants](#)

[Figure 6.11 l'interface de confirmation d'indisponibilités d'enseignants](#)

[Figure 6.12 l'interface d'annoncer une absence d'un enseignant](#)

[Figure 6.13 l'interface pour consulte les salles disponibles](#)

[Figure 6.14 Interface de réservation des salles toujours disponibles](#)

[Figure 6.15 Interface de réservation des salles occasionnellement disponible](#)

[Figure 6.16 Interface de la gestion des réservations des salles](#)

[Figure 6.17 Interface de l'historique des réservations de salle](#)

[Figure 6.18 Tableau d'affichage pour étudiant](#)

[Figure 6.19 Interface de choix du groupe par un étudiant](#)

[Figure 6.20 Interface du profil d'étudiant](#)

[Figure 6.21 Interface du profil enseignant](#)

[Figure 6.22 Interface modules par d'étudiant](#)

[Figure 6.23 Interface modules par enseignant](#)

[Figure 6.24 Interface liste des enseignants](#)

[Figure 6.25 Interface liste des enseignants par étudiant](#)

[Figure 6.26 Interface liste d'équipements](#)

[Figure 6.27 Interface de tableau de bord de l'administrateur et déclenchement de création des groupes](#)

[Figure 6.28 Interface de tableau de bord de l'administrateur et déclenchement de la génération automatique des emplois du temps](#)

[Figure 6.29 Interface de l'administrateur pour faire affectaion des etudiants a des groupes](#)

[Figure 6.30 Model BMC](#)

## Liste des tableaux

[Tableau 1.1 planning de réalisation du projet](#)

[Tableau 2.1 Comparaison entre la méthode Manuelle et Automatique](#)

[Tableau 3.1 Analyse SWOT](#)

[Tableau 5.1 planning de réalisation du projet](#)

# Introduction Générale

Dans un monde de plus en plus numérisé, la gestion efficace des ressources partagées constitue un enjeu majeur, notamment dans le secteur de l'éducation. Les établissements scolaires et universitaires sont confrontés à des défis complexes : emplois du temps surchargés, salles et équipements limités, absences non anticipées, et une coordination souvent difficile entre les différents acteurs (administration, enseignants, étudiants). Ces problématiques, encore largement traitées de manière manuelle ou via des outils peu adaptés, génèrent des pertes de temps, des conflits de réservation et un manque de visibilité sur l'utilisation des ressources.

C'est dans ce contexte que s'inscrit notre projet : une application web intelligente conçue pour automatiser, optimiser et coordonner la gestion des ressources pédagogiques au sein des établissements éducatifs, publics comme privés. Le cœur de l'innovation repose sur l'intégration de l'intelligence artificielle (Google OR-Tools) pour la planification intelligente des emplois du temps, et sur une interface collaborative permettant aux différents utilisateurs (gestionnaires, enseignants, étudiants) de communiquer efficacement et en temps réel.

L'ambition ne se limite pas à une simple solution académique. Il s'agit d'un véritable projet entrepreneurial, pensé pour être scalable et pérenne, avec une vision de déploiement à grande échelle dans l'écosystème éducatif algérien, voire à l'international. À travers ce mémoire, nous adoptons donc une approche hybride : à la fois technique (conception et développement de la solution) et stratégique (analyse de marché, modèle économique, potentiel de croissance, plan de production et de financement).

Le présent document est structuré autour de cinq axes principaux :

1. Présentation du projet, incluant l'idée, les valeurs apportées aux différentes parties prenantes, les objectifs, l'équipe et le planning de réalisation.
2. Les aspects innovants, à la fois sur le plan technologique (optimisation, IA, automatisation) et organisationnel (collaboration en ligne, flexibilité).
3. L'analyse stratégique du marché, pour évaluer les besoins, la concurrence actuelle, et définir une stratégie de positionnement.
4. Le plan de production et d'organisation, détaillant les processus de développement, les ressources nécessaires, les partenaires et la structure organisationnelle.

5. Le plan financier, qui estime les coûts, chiffre d'affaires potentiel, seuil de rentabilité et projections de trésorerie.

À travers ce mémoire, notre objectif est de démontrer que cette solution peut non seulement répondre à une problématique réelle du terrain, mais aussi représenter une opportunité d'innovation et de transformation numérique à fort impact dans le secteur éducatif.

# Axe 1 : Présentation du projet

## 1. L'idée du projet

Les établissements éducatifs tels que les universités, les écoles supérieures ou les instituts de formation, la gestion des ressources partagées constitue un défi majeur. Ces ressources incluent notamment les salles de cours, amphithéâtres, laboratoires, équipements informatiques, ainsi que les ressources humaines telles que les groupes d'étudiants et les enseignants. Elles sont généralement en nombre limité et font l'objet de sollicitations multiples, souvent simultanées, de la part de plusieurs départements ou filières. Cette situation impose une coordination fine et rigoureuse dans le temps et dans l'espace afin d'éviter les conflits d'utilisation et d'assurer une distribution équitable et efficiente.

Toutefois, l'évolution des structures pédagogiques, la diversification des programmes, et la complexité croissante des contraintes logistiques rendent les méthodes traditionnelles de gestion de plus en plus inadaptées. Le recours à des processus manuels ou semi-automatisés devient non seulement fastidieux, mais également source d'erreurs et d'inefficacités. Ce constat souligne la nécessité de solutions plus intelligentes, automatisées et collaboratives pour répondre aux enjeux actuels. Notre projet propose une application web intelligente de gestion coopérative des ressources.

Cette solution permet d'automatiser la planification des emplois du temps en prenant en compte les contraintes de disponibilité, les absences, les demandes de réservation, et les remplacements de séances. Grâce à un moteur d'optimisation basé sur Google OR-Tools, l'application garantit une planification efficace, équitable et réactive.

Elle offre une interface intuitive adaptée aux besoins des gestionnaires, enseignants et étudiants, avec des fonctionnalités clés comme la déclaration d'indisponibilités, la consultation en temps réel des plannings, ou encore l'affectation dynamique des groupes.

## 2. Les valeurs suggérées

Le projet repose sur des valeurs fortes :

- **Innovation** : intégration de l'intelligence artificielle pour résoudre un problème réel de terrain.
- **Simplicité** : interfaces conviviales et accessibles à tous les niveaux d'utilisateurs.

- **Transparence** : meilleure circulation de l'information entre les parties prenantes.
- **Collaboration** : gestion coopérative impliquant enseignants, étudiants et responsables pédagogiques.
- **Optimisation durable** : meilleure utilisation des ressources existantes sans en créer de nouvelles.

Notre solution vise à créer un impact positif pour l'ensemble des parties prenantes. Elle propose des valeurs concrètes pour trois catégories principales d'utilisateurs :

## 2.1 Pour les étudiants

- Accès facile et rapide à leur emploi du temps depuis tout appareil connecté.
- Transparence en temps réel sur les absences d'enseignants et les séances de remplacement.
- Choix autonome du groupe d'étude selon les préférences ou contraintes personnelles.
- Notifications automatiques pour rester informés des modifications ou annulations.

## 2.2 Pour les enseignants

- Déclaration simple de leurs indisponibilités ou absences via une interface dédiée.
- Consultation des salles disponibles pour organiser des séances de rattrapage ou des réunions.
- Réduction des conflits d'emploi du temps grâce à l'optimisation automatique.
- Gain de temps administratif, en se concentrant sur leur cœur de métier : l'enseignement.

## 2.3 Pour l'administration

- Optimisation automatique des emplois du temps (grâce à Google OR-Tools).
- Réduction des erreurs humaines, des conflits de réservation et des tâches répétitives.
- Gestion centralisée des groupes, salles, enseignants et absences via des tableaux de bord puissants.

- Importation rapide des listes d'étudiants via fichiers Excel, avec création automatique des comptes.

### 3. L'équipe

Le projet a été réalisé dans le cadre d'un Master 2 en Informatique. Il a été conduit par une équipe de deux étudiants, sous la supervision d'un encadrement académique et d'un Co-encadrant.

- **Boutamine Amani** : Développeuse principale, responsable de l'architecture générale, du développement back-end avec Django, du développement front-end, de l'intégration du moteur d'optimisation Google OR-Tools, ainsi que de l'intégration complète entre les différentes couches de l'application.
- **Laifa Hocine Chiheb** : Concepteur théorique et rédacteur principal du mémoire, Responsable de la structuration du contenu conceptuelles et méthodologiques pour guider les choix techniques et fonctionnels du projet, ainsi que la conception et la réalisation de la présentation PowerPoint du mémoire.
- **Encadrant** : **Dr Boulehouache Soufiane**, enseignant-chercheur, ayant accompagné le projet sur les aspects méthodologiques et scientifiques.
- **Co-encadrant** : **Dr Bougueroua Salah**, ayant contribué à l'analyse des besoins fonctionnels et à la validation des choix techniques.

Afin d'envisager le passage à une phase de déploiement à plus grande échelle, l'équipe pourrait s'élargir à court terme avec les profils suivants :

- Un(e) designer UX/UI, pour améliorer l'expérience utilisateur.
- Un responsable produit, pour piloter les évolutions fonctionnelles et techniques.
- Un chargé de relations établissements/clients, pour assurer le déploiement et le support auprès des utilisateurs.

### 4. Les objectifs du projet

- Automatiser la création d'emplois du temps selon les contraintes réelles.
- Réduire les erreurs humaines et les conflits de planification.
- Gagner du temps pour les équipes administratives et pédagogiques.
- Fluidifier la communication entre les acteurs d'un établissement.

- Offrir un outil évolutif intégrable dans des structures variées (lycées, universités, centres de formation).

## 5. Le planning de réalisation du projet

Tableau 1.1 planning de réalisation du projet

Étape	Période	Détail
<b>Étude du besoin</b>	Mois 1	Enquêtes, analyse de terrain
<b>Spécifications techniques</b>	Mois 2	Modélisation UML, cas d'usage
<b>Développement de l'application</b>	Mois 3 à 4	Backend (Django), frontend, API
<b>Intégration du moteur OR-Tools</b>	Mois 5	Algorithme d'optimisation
<b>Tests et corrections</b>	Mois 6	Cas réels simulés
<b>Présentation et validation</b>	Mois 7	Soutenance et démo
<b>Version mobile (perspective)</b>	Après validation de la web app	Extension multiplateforme

## Axe 2 : Aspects innovants

### 1. La nature des innovations

Notre solution se distingue par son approche hybride, combinant des technologies web modernes, l'intelligence artificielle et une expérience utilisateur optimisée. Elle introduit plusieurs niveaux d'innovation :

#### 1.1 innovation technique

Intégration de l'outil d'optimisation Google OR-Tools pour générer automatiquement les emplois du temps en tenant compte des contraintes complexes (indisponibilités, nombre de groupes, disponibilités des salles, etc.).

#### 1.2 Innovation fonctionnelle

Multiplis interfaces personnalisées pour chaque type d'utilisateur (enseignant, étudiant, gestionnaire), offrant une interaction simple, efficace et adaptée aux besoins spécifiques de chacun.

#### 1.3 Innovation d'automatisation

Importation automatique des listes d'étudiants via fichiers Excel, avec génération des comptes d'accès (matricule + mot de passe), ce qui réduit fortement le travail manuel des gestionnaires.

#### 1.4 Innovation d'usage

Système de notification intégré (email ou application mobile à venir) qui alerte automatiquement les utilisateurs en cas de changement (absence d'un enseignant, nouvelle séance, etc.).

### 2. Les domaines d'innovation

Ce projet s'inscrit dans plusieurs domaines d'innovation qui renforcent sa pertinence et sa valeur ajoutée :

#### 2.1 Transformation numérique de l'éducation

En proposant un outil centralisé, intelligent et accessible, l'application contribue à moderniser la gestion des établissements scolaires et universitaires.

## 2.2 Innovation pédagogique

L'adaptation des emplois du temps en temps réel, la prise en compte des besoins individuels des étudiants et des enseignants, ainsi que la réduction des interruptions non communiquées, favorisent un environnement d'apprentissage plus stable et réactif.

## 2.3 Innovation organisationnelle

L'automatisation de tâches chronophages et sujettes à erreurs (comme la planification ou la gestion des absences) permet aux établissements de recentrer les ressources humaines sur des activités à plus forte valeur ajoutée.

## 2.4 Innovation inclusive

L'application est conçue pour être utilisable sur le web (et prochainement en version mobile), afin de garantir un accès équitable à tous les utilisateurs, où qu'ils soient, avec une interface intuitive adaptée à tous les niveaux de compétences numériques.

## 3. Comparaison entre la Création Manuelle et Automatique d'un Emploi du Temps

Tableau 2.1 Comparaison entre la méthode Manuelle et Automatique

Critère	Création Manuelle	Création Automatique (par algorithme/logiciel)
<b>Temps nécessaire</b>	Long et fastidieux (plusieurs jours)	Très rapide (quelques minutes)
<b>Ressources humaines</b>	Nécessite l'intervention de plusieurs personnes	Intervention humaine minimale après paramétrage
<b>Gestion des conflits</b>	Difficile à détecter (ex : chevauchement de salle)	Gérée automatiquement par l'algorithme
<b>Flexibilité</b>	Faible : chaque modification demande une reprise	Haute : les modifications sont propagées automatiquement
<b>Complexité supportée</b>	Limité à des structures simples	Capable de gérer des contraintes complexes et multiples
<b>Taux d'erreurs</b>	Élevé (erreurs humaines fréquentes)	Faible (contrôles automatiques intégrés)
<b>Équité de répartition</b>	Difficile à garantir (ex : surcharge d'un prof)	Optimisation équitable (heures, charges, préférences)
<b>Suivi et mise à jour</b>	Manuel et lent	Dynamique et rapide
<b>Adaptabilité aux imprévus</b>	Faible (demandant une reprise complète)	Haute (recalcul possible à tout moment)
<b>Outils nécessaires</b>	Tableurs, papier, communication manuelle	Logiciels spécialisés, ex : Google OR-Tools, OptaPlanner...

# Axe 3 : Analyse stratégique du marché

## 1. Le segment du marché

EN Algérie notre cible principale concerne les établissements éducatifs publics et privés (collèges, lycées, universités, école privée) confrontés à des problématiques de planification, de coordination des ressources et de digitalisation des processus.

Le marché secondaire inclut les organismes de formation et les centres de formation professionnelle, qui gèrent également des emplois du temps complexes.

## 2. Analyse SWOT

Une stratégie marketing est directement liée aux objectifs de l'entreprise et elle doit traduire ses ambitions à l'échelle d'un marché. Son établissement s'appuie en priorité sur les opportunités qui se présentent, et pour cela on a le modèle de l'analyse SWOT.

C'est un outil qui permet à une entreprise d'identifier ses avantages concurrentiels, de comprendre ses faiblesses, d'exploiter les opportunités du marché et de se prémunir contre les menaces potentielles. Elle aide à orienter la stratégie et la prise de décision en mettant en évidence les domaines dans lesquels l'entreprise peut se développer et les aspects qu'elle doit améliorer. Notre modèle d'analyse SWOT a pour but de mettre en évidence les forces et les faiblesses internes de notre projet, ainsi que les opportunités et les menaces externes qui peuvent impacter sa réussite. Voici les résultats pour CoopEase :

Tableau 3.1 Analyse SWOT

<p style="text-align: center;"><b>Force</b></p> <ul style="list-style-type: none"> <li>● Planification automatisée grâce à l'IA, réduisant erreurs humaines et conflits d'emploi du temps.</li> <li>● Application flexible, accessible via navigateur, sans installation, pour tous les utilisateurs (étudiants, enseignants, administration).</li> <li>● Notre service répond à un besoin réel.</li> <li>● Offre une solutionne-logistique éducative Innovante.</li> <li>● Conçue pour être facile à prendre en main, avec tableaux de bord dédiés pour chaque type d'utilisateur.</li> <li>● Intégration rapide des étudiants via fichiers Excel, avec génération automatique de comptes utilisateurs.</li> </ul>	<p style="text-align: center;"><b>Faiblesse</b></p> <ul style="list-style-type: none"> <li>● Besoin d'un hébergement sécurisé et stable, avec des ressources suffisantes.</li> <li>● L'adoption peut nécessiter une formation initiale pour les utilisateurs (enseignants, gestionnaires).</li> <li>● Une première expérience dans la création d'une entreprise</li> <li>● Nécessite une expertise en développement Python/Django et en modélisation avec OR-Tools.</li> <li>● Nous pourrions faire face à des problèmes D'adoption par les clients en raison de la méfiance ou du manque de confiance envers les nouvelles technologies.</li> </ul>
<p style="text-align: center;"><b>Opportunité</b></p> <ul style="list-style-type: none"> <li>● Extension possible via une application mobile pour une accessibilité accrue.</li> <li>● Intérêt croissant des écoles privées, universités et instituts pour des solutions de gestion optimisées.</li> <li>● En proposant des solutions innovantes et efficace.</li> <li>● Possibilité de collaboration avec les ministères ou les plateformes éducatives nationales.</li> </ul>	<p style="text-align: center;"><b>Menace</b></p> <ul style="list-style-type: none"> <li>● Réticence de certains acteurs à abandonner les méthodes traditionnelles.</li> <li>● Difficultés potentielles de financement dans certains établissements publics.</li> <li>● Présence de solutions commerciales étrangères bien établies.</li> <li>● À l'avenir il peut avoir des concurrents qui propose des solutions ou offre similaire</li> </ul>

### 3. Intensité de la concurrence

En Algérie, le marché des outils numériques de gestion des ressources pédagogiques reste très peu structuré et faiblement concurrentiel. La majorité des établissements scolaires et universitaires continuent de recourir à des méthodes traditionnelles ou semi-automatisées, souvent basées sur :

- Des tableaux Excel conçus manuellement pour les emplois du temps.
- Des échanges informels par téléphone ou email en cas d'absence d'un enseignant.

- Une centralisation excessive des décisions, ralentissant la réactivité face aux imprévus.
- Une absence d'interfaces unifiées accessibles aux enseignants et étudiants.

Peu ou pas d'outils intelligents sont utilisés actuellement pour gérer dynamiquement les contraintes de ressources (salles, enseignants, groupes) dans les établissements.

Notre solution arrive donc sur un marché vierge, où l'offre reste limitée, et où la demande est en croissance sous l'impulsion de la transformation numérique des institutions éducatives, notamment avec les réformes engagées par le ministère de l'Enseignement Supérieur et de la Recherche Scientifique.

Cette faible intensité concurrentielle représente une opportunité stratégique majeure : en se positionnant tôt avec une solution adaptée au contexte local, notre projet peut rapidement devenir un acteur de référence.

## 4. La stratégie marketing

Afin d'assurer une adoption progressive et durable de notre solution par les établissements éducatifs, notre stratégie marketing repose sur plusieurs leviers adaptés au contexte local :

### 4.1 Approche B2B ciblée

Nous visons directement les décideurs institutionnels (directeurs d'établissements, responsables informatiques, ministères, les école privée) à travers :

- Des présentations personnalisées (pitches, démonstrations en établissement).
- La participation à forums, salons et séminaires éducatifs.
- Un référencement via des canaux institutionnels (ex : site du ministère, plateformes éducatives).

### 4.2 Valorisation des bénéfices concrets

Nous mettons en avant des résultats tangibles :

- Gain de temps administratif.
- Meilleure gestion des imprévus (absences, remplacements).
- Communication fluide entre étudiants, enseignants et administration.
- Optimisation de l'usage des ressources existantes.

### 4.3 Stratégie de notoriété progressive

- Mise en œuvre dans un établissement pilote pour démontrer l'efficacité de la solution.
- Création d'un site vitrine et de contenus pédagogiques (vidéos explicatives, tutoriels).
- Promotion sur les réseaux sociaux professionnels (LinkedIn, Facebook Éducation, etc.).

### 4.4 Modèle économique simple et accessible

- Version gratuite ou subventionnée pour les premiers établissements partenaires.
- Abonnement annuel pour les établissements avec accompagnement technique.
- Possibilité d'une offre freemium, avec des fonctionnalités avancées payantes.

### 4.5 Partenariats stratégiques

- Coopération avec des universités techniques pour test et amélioration continue.
- Collaboration avec des incubateurs ou structures d'innovation éducative.
- Appui des ministères concernés pour favoriser l'intégration nationale.

# Axe 4 : Plan de production et organisation

## 1. Le processus de développement

Le développement de l'application repose sur une architecture web robuste et évolutive, construite autour du framework Django (Python). Ce choix technologique garantit à la fois la sécurité, la maintenabilité et la rapidité de développement. Le système intègre un moteur d'optimisation basé sur Google OR-Tools, permettant une planification intelligente des ressources (emplois du temps, salles, enseignants), en tenant compte de multiples contraintes.

Le processus de production s'est articulé autour de plusieurs étapes clés :

### 1.1 Analyse des besoins

Recueil des besoins réels des établissements éducatifs (écoles, lycées, universités), à travers des échanges avec des gestionnaires, enseignants et étudiants. Cette phase a permis de définir des cas d'usage concrets et des priorités fonctionnelles.

### 1.2 Conception technique

Modélisation des entités métier (utilisateurs, ressources, sessions, absences), des flux de données et des interfaces utilisateurs. Des diagrammes UML ont été utilisés pour clarifier l'architecture du système.

### 1.3 Développement des modules

Réalisation progressive de chaque fonctionnalité du système :

- Authentification sécurisée des utilisateurs
- Gestion des disponibilités et absences
- Génération automatique de l'emploi du temps
- Consultation et notification des changements
- Tableaux de bord pour les administrateurs
- Interfaces spécifiques pour chaque acteur (étudiant, enseignant, gestionnaire)

## 1.4 Phase de tests

Vérification fonctionnelle, tests unitaires et validation avec un panel d'utilisateurs pilotes. Ces tests ont permis d'ajuster certaines interfaces et de corriger les anomalies.

## 1.5 Mise en oeuvre

Déploiement de l'application sur un serveur cloud sécurisé (ou sur un serveur local selon le contexte de l'établissement). Ce déploiement garantit une accessibilité continue via navigateur, sans installation préalable.

Une des fonctionnalités particulièrement innovantes du processus est l'automatisation de l'intégration des utilisateurs. En effet, l'application permet :

- L'importation directe de la liste des étudiants à partir d'un fichier Excel standardisé ;
- La création automatique des comptes utilisateurs, avec un matricule comme identifiant et un mot de passe généré automatiquement.

Cette fonctionnalité réduit significativement la charge administrative en début d'année et facilite l'activation rapide du système par les gestionnaires d'établissement.

## 2. L'approvisionnement

Dans le cadre d'un projet logiciel comme cette application web intelligente de gestion des ressources, l'approvisionnement ne concerne pas des matières premières physiques, mais plutôt des ressources logicielles, techniques et humaines nécessaires au développement, au déploiement et à l'exploitation de la solution.

### 2.1 Ressources logicielles

- **Frameworks et bibliothèques open source** : Django, Google OR-Tools, Bootstrap, Pandas, etc. Ces outils sont gratuits et largement utilisés dans l'industrie.
- **Systèmes de gestion de base de données** : PostgreSQL ou MySQL.
- **Outils de versioning et collaboration** : Git et GitHub ou GitLab, pour assurer un suivi rigoureux du code source.

### 2.2 Ressources techniques

- **Serveur d'hébergement** : Un serveur cloud (comme AWS, DigitalOcean, ou un hébergement universitaire) est requis pour déployer l'application. À court

terme, un simple VPS (Virtual Private Server) suffit ; à long terme, une infrastructure évolutive peut être envisagée.

- **Matériel informatique de développement** : Ordinateurs portables des développeurs avec environnement de développement local configuré (Python, VS Code, etc.).

## 2.3 Ressources humaines

- **Développeurs** : Initialement l'équipe projet (toi et ton binôme), encadrée par des enseignants.
- **Conseillers pédagogiques** : Pour affiner les besoins fonctionnels du système.
- **Futurs collaborateurs** : À mesure que le projet se développe, il pourra nécessiter un administrateur système, un responsable client, un designer UX/UI et potentiellement un data scientist pour améliorer l'optimisation.

## 2.4 Services externes (à prévoir)

- **Nom de domaine et certificat SSL** : Pour sécuriser l'accès web.
- **Service de messagerie** : SMTP pour les notifications automatiques (absences, changements d'emplois du temps).
- **Sauvegardes et monitoring** : Outils de surveillance et de sauvegarde régulière des données.

## 3. La main d'œuvre

Le développement et la gestion de l'application nécessitent une main d'œuvre qualifiée, composée à la fois de profils techniques, fonctionnels et relationnels. L'organisation de la main d'œuvre s'appuie sur deux phases : la phase initiale de réalisation du projet (académique) et la phase de mise en production et de développement futur (startup).

### 3.1 Phase de développement initial

- **Développeurs** : Deux étudiants de Master 2 en Informatique (**Laifa Hocine Chiheb** et **Boutamine Amani**), responsables du développement complet de l'application (backend, frontend, intégration de l'optimisation).
- **Encadrants académiques** : Un enseignant-chercheur encadrant principal (**bouhouache sofiane**), et un co-encadrant (**bougueroua salah**), apportant expertise, relecture et validation des choix techniques et fonctionnels.

- **Contributeurs pédagogiques** : Des responsables d'établissement et enseignants ayant contribué à l'expression des besoins et aux tests utilisateurs.

## 3.2 Phase de déploiement et d'évolution

À l'avenir, une montée en charge nécessitera la constitution d'une équipe élargie :

- **Développeur full stack** : Pour assurer les évolutions techniques.
- **UX/UI Designer** : Pour améliorer l'expérience utilisateur sur le web et sur l'application mobile prévue.
- **Responsable produit** : Pour piloter les fonctionnalités, gérer les priorités et recueillir les retours utilisateurs.
- **Administrateur système** : Pour gérer le déploiement, la sécurité, les sauvegardes et les performances de l'application.
- **Chargé de relations établissements** : Pour assurer l'accompagnement des utilisateurs, la formation et le support.

## 3.3 Vision collaborative

L'application repose sur une logique de co-construction. Les utilisateurs (enseignants, gestionnaires, étudiants) seront régulièrement impliqués dans les retours d'expérience pour faire évoluer le produit au plus près des besoins réels.

## 4. Les principaux partenaires

Le succès du projet repose sur un écosystème de partenaires qui contribuent à différentes phases de la conception, du déploiement et de la valorisation de l'application. Ces partenaires apportent un soutien à la fois technique, institutionnel, académique et stratégique.

### 4.1 Partenaires académiques

- **Université (Établissement d'enseignement supérieur)**

L'université constitue le cadre initial du projet. Elle a fourni l'environnement pédagogique et technique pour sa réalisation, notamment à travers le suivi des encadrants et l'accès aux plateformes de test.

- **Encadrants (boulehouache sofiane et bougueroua salah)**

Ils ont assuré le pilotage scientifique et technique du projet, garantissant sa pertinence par rapport aux besoins réels des établissements éducatifs.

## 4.2 Partenaires institutionnels

- **Ministère de l'Enseignement Supérieur / Éducation Nationale**

Le projet vise à répondre aux problématiques concrètes rencontrées dans les établissements publics. Ces institutions peuvent jouer un rôle-clé dans :

- La diffusion de l'application via leurs plateformes officielles.
- L'implémentation à l'échelle régionale ou nationale.
- Le financement ou le soutien logistique lors de la phase de déploiement.

## 4.3 Partenaires techniques

- **Fournisseurs d'hébergement cloud**

Pour la mise en production de la plateforme (par exemple : AWS, OVH, Scaleway).

- **Outils open-source**

Le projet repose sur plusieurs technologies libres et performantes :

- Django (framework web Python)
- Google OR-Tools (moteur d'optimisation)
- PostgreSQL / MySQL (bases de données)

## 4.4 Partenaires utilisateurs

- **Établissements éducatifs volontaires (publics et privés)**

Des établissements pilotes (universités, lycée, collège, écoles privées, instituts de formation) seront sollicités pour tester la solution dans un cadre réel, fournir des retours, et contribuer à l'amélioration continue de l'outil.

- **Écoles privées**

Intéressées par des solutions innovantes pour se différencier et améliorer leur gestion interne, elles représentent un segment stratégique et réactif pour l'adoption rapide de l'application.

# Axe 5 : Plan financier

## 1. Les coûts et les charges

Le développement et le déploiement de l'application impliquent plusieurs catégories de dépenses :

### 1.1 Coûts de développement

- Temps de travail des développeurs (valorisé en équivalent-journées).
- Matériel informatique (ordinateur, licences éventuelles).
- Frais liés aux outils techniques : hébergement, nom de domaine, services cloud.

### 1.2 Coûts d'exploitation

- Hébergement web (serveur cloud sécurisé).
- Maintenance de la plateforme (mises à jour, surveillance).
- Services de messagerie (notifications, e-mails).
- Sauvegardes automatiques et sécurité des données.

### 1.3 Coûts de communication

- Création d'une identité visuelle (logo, charte graphique).
- Création d'un site vitrine de présentation.
- Publicité ciblée (réseaux sociaux, plateformes éducatives).

### 1.4 Coûts juridiques et administratifs

- Enregistrement de la startup.
- Dépôt de la marque / protection de la propriété intellectuelle.
- Rédaction des CGU / politique de confidentialité.

## 2. Le chiffre d'affaires

Le modèle économique repose sur un abonnement annuel ou semestriel proposé aux établissements éducatifs.

## 2.1 Modèle B2B (Business to Business)

- Établissements universitaires publics ou privés, instituts, écoles supérieures.
- Tarification adaptée à la taille (nombre d'utilisateurs) :
  - o < 500 utilisateurs : 150 000 DA / an
  - o 500–1000 utilisateurs : 250 000 DA / an
  - o 1000 utilisateurs : 350 000 DA / an

## 2.2 Modèle évolutif

- Ajout possible de services premium (support dédié, module mobile, statistiques avancées).
- Application mobile proposée comme option complémentaire (licence ou abonnement).

## 3. Les comptes de résultats escomptés (prévisionnels)

Voici une projection financière simplifiée sur 3 ans :

Tableau 5.1 planning de réalisation du projet

Année	Nombre d'établissements clients	Chiffre d'affaires estimé	Charges annuelles	Résultat net estimé
2025	5	1 000 000 DA	400 000 DA	600 000 DA
2026	15	3 000 000 DA	1 000 000 DA	2 000 000 DA
2027	30	6 500 000 DA	2 000 000 DA	4 500 000 DA

Ces résultats dépendent du rythme de déploiement, de la rétention client, et du soutien institutionnel.

## 4. Le plan de trésorerie

Le plan de trésorerie vise à assurer la stabilité financière du projet sur ses premiers mois de lancement :

- **Dépenses prioritaires à couvrir** dès le lancement :
  - o Développement final (tests, sécurité).

- o Hébergement et infrastructure.
- o Communication initiale.
- **Sources de financement possibles :**
  - o Apports personnels des fondateurs.
  - o Concours étudiants ou fonds d'innovation.
  - o Subventions publiques (ministère, incubateurs).
  - o Accords avec établissements pilotes.

# Axe 6 : Prototype Expérimental

## 1. Structure et Principe de Fonctionnement du Système

Ce prototype constitue une preuve de concept opérationnelle permettant de démontrer la faisabilité de la solution proposée.

## 2. Les interfaces

Dans ce qui suit, nous allons présenter quelques captures d'écran des interfaces graphiques développées par notre plateforme.

### 2.1 les pages d'accueils

La page d'accueil offre la possibilité de visualiser les différents services des utilisateurs (**enseignant, étudiant, responsable département**) comme c'est présenté dans les figures ci-dessous.



Figure 6.1 pages d'accueil enseignant

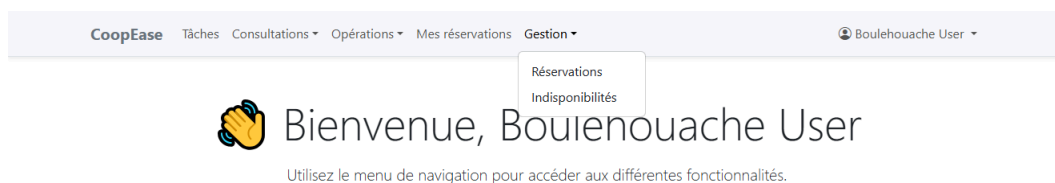


Figure 6.2 pages d'accueil responsable département

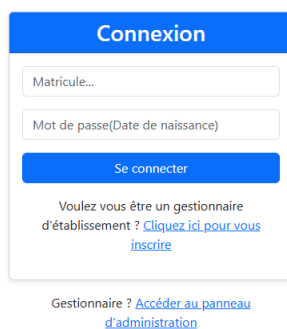
 Bienvenue, Amani Boutamine

Utilisez le menu de navigation pour accéder aux différentes fonctionnalités.

**Figure 6.3 pages d'accueil étudiant**

## 2.2 Interface d'authentification

Cette interface permet aux utilisateurs de se connecter à la plateforme selon leur rôle (enseignant, étudiant, responsable département). Elle vérifie les identifiants via le système d'authentification intégré de Django.



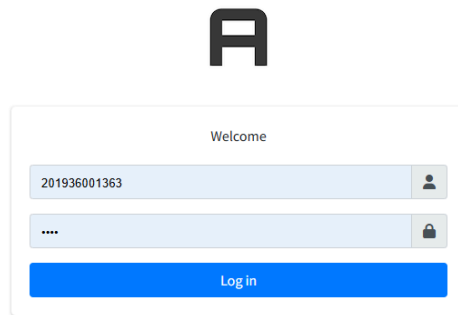
The screenshot shows a login form with a blue header labeled 'Connexion'. It contains two input fields: 'Matricule...' and 'Mot de passe(Date de naissance)'. Below the fields is a blue button labeled 'Se connecter'. Underneath the button, there is a text prompt: 'Voulez vous être un gestionnaire d'établissement ? [Cliquez ici pour vous inscrire](#)'. At the bottom of the form, there is a link: 'Gestionnaire ? [Accéder au panneau d'administration](#)'.

**Figure 6.4 pages d'authentification enseignant, étudiant, responsable département**

## 2.3 Interface d'authentification d'administration

Cette interface permet aux administrateurs de se connecter à la plateforme d'administration et vérifie les identifiants via le système d'authentification intégré de

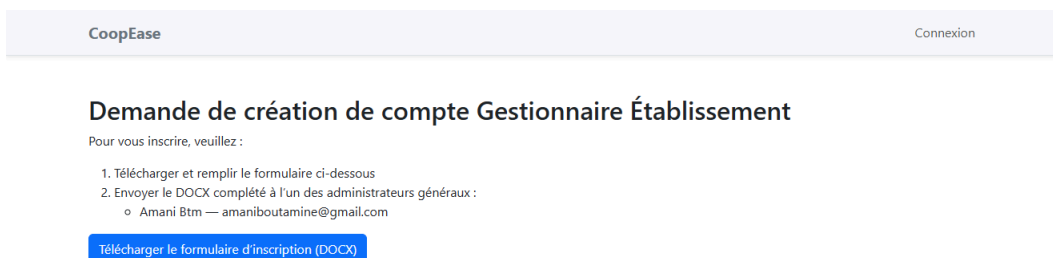
Django.



**Figure 6.5** pages d'authentification administrateurs

## 2.4 Interface d'inscription d'un gestionnaire établissement

Le demandeur remplir la formule et l'envoie à l'administrateur par email et attende une repense.



**Figure 6.6** pages d'inscription d'un gestionnaire

### 2.4.1 Interface de formulaire à remplir par gestionnaire

Le demandeur d'inscription pour devenir un gestionnaire d'établissent doit remplir tous les informations demande dans le formulaire ci-dessous

## Formulaire d'Inscription

### 1. Informations Utilisateur

Nom	.....
Prénom	.....
Date de naissance	.....
Matricule (Pièce d'identité)	.....
Email	.....
Téléphone	.....

### 2. Informations Établissement

Nom de l'Établissement	.....
Type (Université, Lycée, CEM, Primaire, Autre)	.....
Adresse	.....

Figure 6.7 pages de formulaire d'inscription

## 2.5 Consultation de l'emploi du temps

Chaque un des utilisateurs (étudiant ou enseignants) peut consulter son emploi.

L'enseignant dispose d'une interface dédiée lui permettant de consulter son emploi du temps personnel de manière claire et structurée.

Étudiant peut consulter son emploi du temps selon son groupe ou sa section, L'interface est responsive et permet un accès rapide aux informations.

# Emploi du Temps

Année scolaire: 2025/2026

S1 S2

Mon Groupe Ma spécialité Département

SI - Licence3- Groupe 1						
Jour/Heure	08:00	09:30	11:00	12:30	14:00	15:30
SAMEDI	TP-AppMob/ G1/ STP5/ Kara	C-SecInfo Nafir Amphi1	C-IA Hazmoune Amphi1	C-DSStruc Mansoul Amphi1	TD-RédSci/ G1/ STD15/ Nabet	C-CDStartup Mazouzi EnligneEnligne
DIMANCHE	TP-DSStruc/ G1/ STPA2/ Mansoul	C-AppMob Kara Amphi4	TD-SecInfo/ G1/ STDA10/ Nafir	TP-IA/ G1/ STP5/ Hazmoune		

Figure 6.8 l'emploi du temps d'étudiant

CoopEase Tâches Consultations Opérations Mes réservations						
Bougueroua Salah						
Taches de travail de M.Bougueroua						
Jour/Heure	08:00	09:30	11:00	12:30	14:00	15:30
DIMANCHE	ISIL - SecInfo TD / Salle: TD39 Section1 / G4					
LUNDI	ISIL - SecInfo TD / Salle: TD24 Section1 / G3					
MARDI	ISIL - SecInfo TD / Salle: TD17 Section1 / G1	ISIL - SecInfo Cours / Salle: Sallelec1 Section1				
MERCREDI	ISIL - SecInfo TD / Salle: TD23 Section1 / G2					

Figure 6.9 l'emploi du temps de l'enseignant

## 2.6 Interface de gestion des disponibilités enseignant

Cette interface permet à un enseignant d'indiquer ses plages d'indisponibilité. Ces informations sont ensuite prises en compte lors de la génération automatique de l'emploi du temps s'ils sont approuvés par le responsable d'établissement.

**Note :** Les indisponibilités seront prises en compte uniquement lors de la génération des emplois du temps. Veuillez indiquer uniquement les périodes où vous serez occupé(e) pendant **3 mois** ou plus.

🕒 **Sélectionner les heures d'indisponibilité**

Voir mes horaires d'indisponibilité précédents ^

**DIMANCHE:** 11:00:00 - 12:30:00 ✕ Supprimer

**MERCREDI:** 15:30:00 - 17:00:00 ✔ Supprimer

Jour d'indisponibilité:

Créneaux horaires:

- 08:00:00 - 09:30:00
- 09:30:00 - 11:00:00
- 11:00:00 - 12:30:00
- 12:30:00 - 14:00:00
- 14:00:00 - 15:30:00
- 15:30:00 - 17:00:00

Enregistrer
Retour

**Figure 6.10** l'interface de gestion d'indisponibilités d'enseignants

### 2.7 Interface des demandes des disponibilités des enseignants

Responsable de département dispose d'un tableau de bord spécifique lui permettant de consulter, approuver ou refuser chaque demande, en tenant compte des impératifs pédagogiques et organisationnels du département.

CoopEase Tâches Consultations Opérations Mes réservations Gestion
Boulehouache User

✔ **Confirmation des indisponibilités**

**Bougueroua User** — DIMANCHE

08:00:00 - 09:30:00 09:30:00 - 11:00:00 Annuler

✔ Confirmé

**Boulehouache User** — MERCREDI

15:30:00 - 17:00:00 Confirmer

**Boulehouache User** — DIMANCHE

11:00:00 - 12:30:00 Confirmer

**Mallemk User** — MERCREDI

08:00:00 - 09:30:00 Confirmer

**Mallemk User** — SAMEDI

08:00:00 - 09:30:00 Annuler

✔ Confirmé

**Aliguechi User** — SAMEDI

08:00:00 - 09:30:00 Annuler

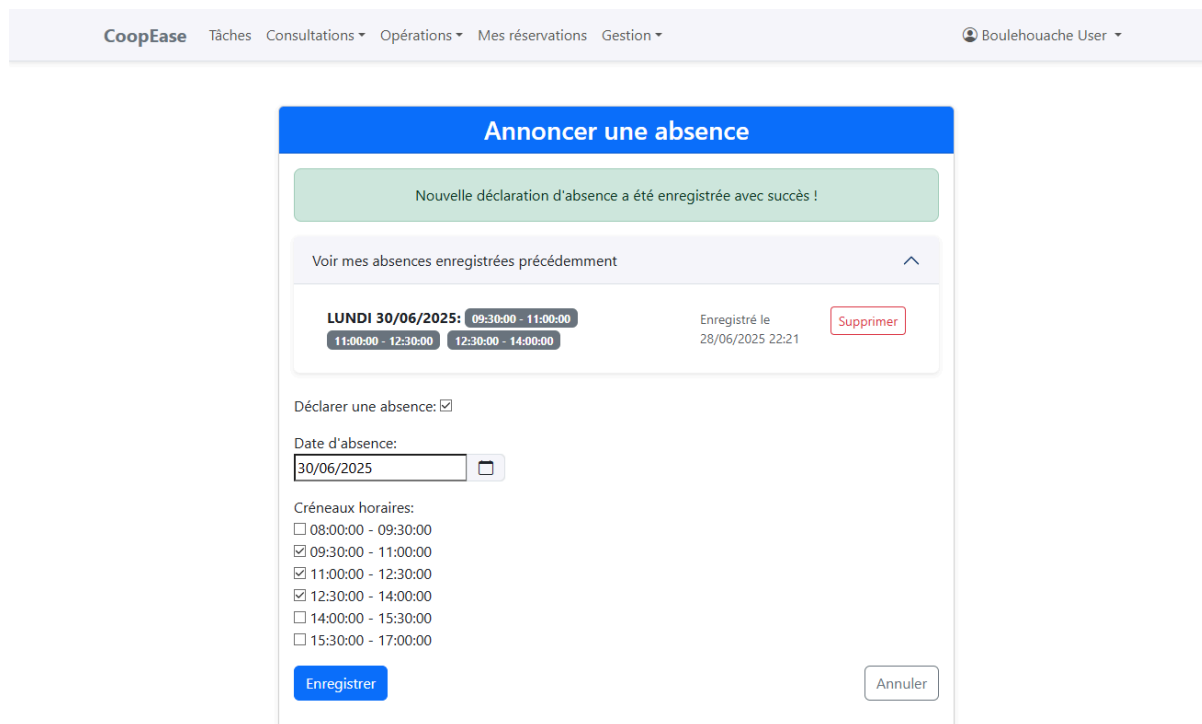
✔ Confirmé

**Figure 6.11** l'interface de confirmation d'indisponibilités d'enseignants

32

## 2.8 Interface d'annonce d'une absence par un enseignant

Cette interface permet à un enseignant de signaler une absence imprévue, via la plateforme.



The screenshot shows the 'Annoncer une absence' (Announce an absence) interface. At the top, there is a navigation bar with 'CoopEase' and menu items: 'Tâches', 'Consultations', 'Opérations', 'Mes réservations', and 'Gestion'. The user is identified as 'Boulehouache User'. The main content area has a blue header 'Annoncer une absence' and a green success message: 'Nouvelle déclaration d'absence a été enregistrée avec succès !'. Below this is a section 'Voir mes absences enregistrées précédemment' with an upward arrow. It displays a list of absences for 'LUNDI 30/06/2025' with time slots: 09:30:00 - 11:00:00, 11:00:00 - 12:30:00, and 12:30:00 - 14:00:00. The first slot is highlighted. To the right, it says 'Enregistré le 28/06/2025 22:21' and has a 'Supprimer' button. Below the list, there is a checkbox 'Déclarer une absence: [checked]'. The 'Date d'absence:' is set to '30/06/2025' with a calendar icon. Under 'Créneaux horaires:', there are checkboxes for various time slots: 08:00:00 - 09:30:00, 09:30:00 - 11:00:00 (checked), 11:00:00 - 12:30:00 (checked), 12:30:00 - 14:00:00 (checked), 14:00:00 - 15:30:00, and 15:30:00 - 17:00:00. At the bottom, there are 'Enregistrer' and 'Annuler' buttons.

Figure 6.12 l'interface d'annoncer une absence d'un enseignant

## 2.9 Consultations la disponibilité des salles

Cette interface offre aux enseignants la possibilité de consulter, à tout moment, la disponibilité des salles au sein de l'établissement. Elle répond à plusieurs besoins, notamment la recherche de créneaux libres pour organiser des séances de rattrapage, des travaux dirigés supplémentaires ou remplacer un cours annulé.

## 📌 DIMANCHE

Heure	🏠 Espaces disponibles																			
08:00-09:30	1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi)																			
09:30-11:00	1 (Sallelec) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) A3 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD)																			
11:00-12:30	A10 (TD) 1 (Sallelec) 3 (Amphi) 2 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 18 (TD) 16 (TD)																			
12:30-14:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 38 (TD) 37 (TD) 23 (TD) 21 (TD) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			
14:00-15:30	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 23 (TD) 22 (TD) 83 (TP) 5 (TP) 4 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			
15:30-17:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) 3 (TP) A3 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			

## 📌 LUNDI

Heure	🏠 Espaces disponibles																			
08:00-09:30	1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi)																			
09:30-11:00	A10 (TD) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 3 (TP) A3 (TP) A2 (TP) 2 (TP) 20 (TD) 19 (TD) 18 (TD)																			
11:00-12:30	1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 37 (TD) 23 (TD) 22 (TD) 21 (TD)																			
12:30-14:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 21 (TD) 20 (TD) 17 (TD) 16 (TD) 14 (TD) 13 (TD)																			
14:00-15:30	salle (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) A3 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			
15:30-17:00	salle (TD) A10 (TD) 1 (Sallelec) 4 (Amphi) 3 (Amphi) 2 (Amphi) 1 (Amphi) 49 (TD) 39 (TD) 38 (TD) 37 (TD) 24 (TD) 23 (TD) 22 (TD) 21 (TD) 83 (TP) 5 (TP) 4 (TP) 3 (TP) A3 (TP) A2 (TP) 2 (TP) 1 (TP) 20 (TD) 19 (TD) 18 (TD) 17 (TD) 16 (TD) 15 (TD) 14 (TD) 13 (TD)																			

## 📌 MARDI

Figure 6.13 l'interface pour consulte les salles disponibles

## 2.10 Réservation de salle

Les enseignants peuvent soumettre une demande de réservation ponctuelle de salle en cas de besoin (examen, session de rattrapage, etc.).

Il Ya deux type de salle :

## 2.10.1 Réserveation des salles toujours disponible dans emploi du temps

CoopEase Tâches Consultations Opérations Mes réservations Gestion Boulehouache User

### Salles & Plages horaires disponibles

Jour: MERCREDI Heure: Tous Type de salle: Tous Numéro de salle: Tous

Salle	Type	Jour	Heure	Module	Type module	Groupe	Section	Dates Réservées	Action
13	TD	MERCREDI	08:00	-- Choisissez un module --	-- Typ	Groupe	Section	02-07-2025 (Votre réservation)	mm/dd/yyyy Réserver
14	TD	MERCREDI	08:00	AppMob	TP	2	1		07/02/2025 Réserver
15	TD	MERCREDI	08:00	-- Choisissez un module --	-- Typ	Groupe	Section		mm/dd/yyyy Réserver
17	TD	MERCREDI	08:00	-- Choisissez un module --	-- Typ	Groupe	Section		mm/dd/yyyy Réserver

Figure 6.14 Interface de réserveation des salles toujours disponibles

## 2.10.2 Réserveation des salles occasionnellement disponible

Ce sont les salles où sont libre lauré ce qu'un enseignant est absent.

CoopEase Tâches Consultations Opérations Mes réservations Gestion Boulehouache User

### Salles Occasionnellement Disponibles (En cas d'absence)

Salle	Type	Date	Jour	Heure	Module	Type module	Groupe	Section	Status	Action
49	TD	01-07-2025	MARDI	08:00	-- Choisissez un module --	-- Type	Groupe	Section	Réservé	Indisponible
39	TD	01-07-2025	MARDI	09:30	-- Choisissez un module --	-- Type	Groupe	Section	Disponible	Réserver
49	TD	01-07-2025	MARDI	12:30	-- Choisissez un module --	-- Type	Groupe	Section	Disponible	Réserver
2	TP	01-07-2025	MARDI	14:00	-- Choisissez un module --	-- Type	Groupe	Section	Disponible	Réserver

Figure 6.15 Interface de réserveation des salles occasionnellement disponible

## 2.11 Interface de gestion des Réservations des salles

Les demandes de réserveation des salles seront approuvées ou rejeter par le responsable de département

## Réservations pas encore Approuvées

Enseignant	Salle	Date	Jour	Heure	Module	Type	Groupe	Section	Date de Demande	Actions
Boulehouache User	49 (TD)	July 1, 2025	MARDI	08:00	AppMob	Cours	Tous	Section1	28/06/2025 22:19	<span style="background-color: #28a745; color: white; padding: 2px 5px;">Approuver</span> <span style="background-color: #dc3545; color: white; padding: 2px 5px; margin-left: 5px;">Rejeter</span>
Bougueroua User	13 (TD)	June 26, 2025	JEUDI	08:00	SecInfo	Cours	1	Section1	19/06/2025 11:05	<span style="background-color: #28a745; color: white; padding: 2px 5px;">Approuver</span> <span style="background-color: #dc3545; color: white; padding: 2px 5px; margin-left: 5px;">Rejeter</span>

## Réservations Approuvées

Enseignant	Salle	Date	Jour	Heure	Module	Type	Groupe	Section	Date de Demande	Actions
Boulehouache User	13 (TD)	July 2, 2025	MERCREDI	08:00	AutoComp	Cours		Section1	23/06/2025 14:04	<span style="background-color: #ffc107; padding: 2px 5px;">Annuler</span>

**Figure 6.16 Interface de la gestion des réservations des salles**

## 2.12 interfaces d'historique des réservations des salles

Cette interface permet aux enseignants la possibilité de consulter, à tout moment, la réservation de salle et s'ils sont approuvés ou non.

### 🕒 Mes Réservations

**Réservation De Salle** 28-06-2025 22:19

Salle réservé: 49 (TD) - Informatique  
 Date: 2025-07-01 | Jour: MARDI  
 Heure: 08:00:00 - 09:30:00  
 Module: AppMob | Type: Cours  
 Groupe: Tous | Section: Section1

🕒 En attente

🗑️ Annuler

**Réservation De Salle** 23-06-2025 14:04

Salle réservé: 13 (TD) - Informatique  
 Date: 2025-07-02 | Jour: MERCREDI  
 Heure: 08:00:00 - 09:30:00  
 Module: AutoComp | Type: Cours  
 Groupe: | Section: Section1

✅ Confirmé

🗑️ Annuler

**Figure 6.17 Interface de l'historique des réservations de salle**

## 2.13 interfaces d'affichages pour étudiant

Cette interface est destinée aux étudiants et leur permet de rester informés en temps réel des changements affectant leur emploi du temps. Elle regroupe toutes les informations relatives aux absences des enseignants et aux séances de remplacement programmées

## Tableau d'Affichage

Séance ajoutée
<p><b>Spécialité:</b> SI - Licence3</p> <p><b>Professeur:</b> Nafir User</p> <p><b>Salle:</b> 1 (Amphi)</p> <p><b>Module:</b> SecInfo - TD</p> <p><b>Jour:</b> MARDI</p> <p><b>Heure:</b> 08:00</p> <p><b>Groupe:</b> 1</p> <p><b>Section:</b> Section1</p> <p><b>Date:</b> 01-07-2025</p> <p>Ajouté le: 29-06-2025 17:22</p>
Absence
<p><b>Professeur:</b> Boulehouache User</p> <p><b>Date d'absence:</b> 30-06-2025</p> <p><b>Créneaux horaires:</b> 09:30, 11:00, 12:30</p> <p>Ajouté le: 28-06-2025 22:21</p>

Figure 6.18 Tableau d'affichage pour étudiant

## 2.14 Interface de demande d'intégration à un groupe par un étudiant

Cette interface est conçue pour permettre aux étudiants de soumettre une demande d'intégration à un groupe spécifique. Elle facilite la gestion des affectations tout en offrant un canal de communication formel entre l'étudiant et l'administration pédagogique.

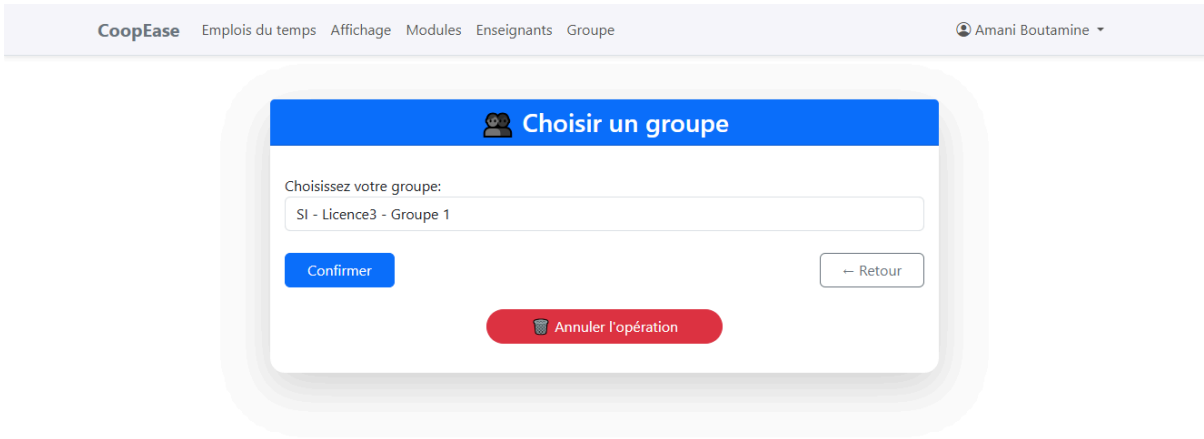


Figure 6.19 Interface de choix du groupe par un étudiant

## 2.15 Interface du profil

Les utilisateurs peuvent Consulter le profil et voir ces informations

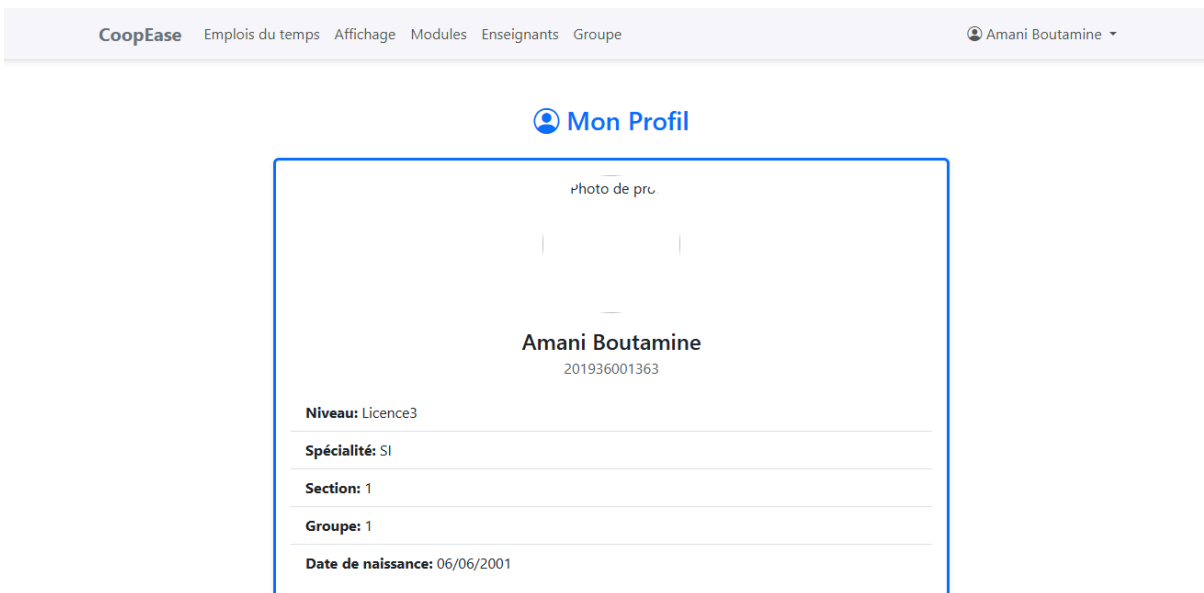
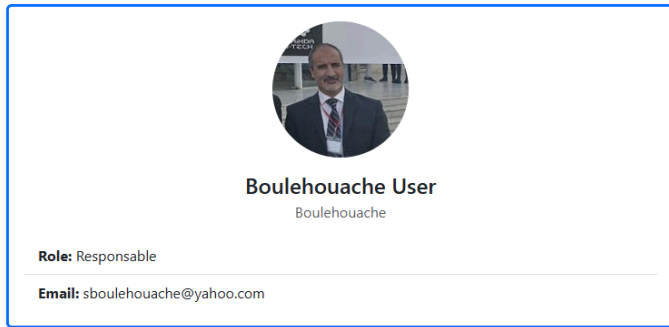


Figure 6.20 Interface du profil d'étudiant

### Mon Profil

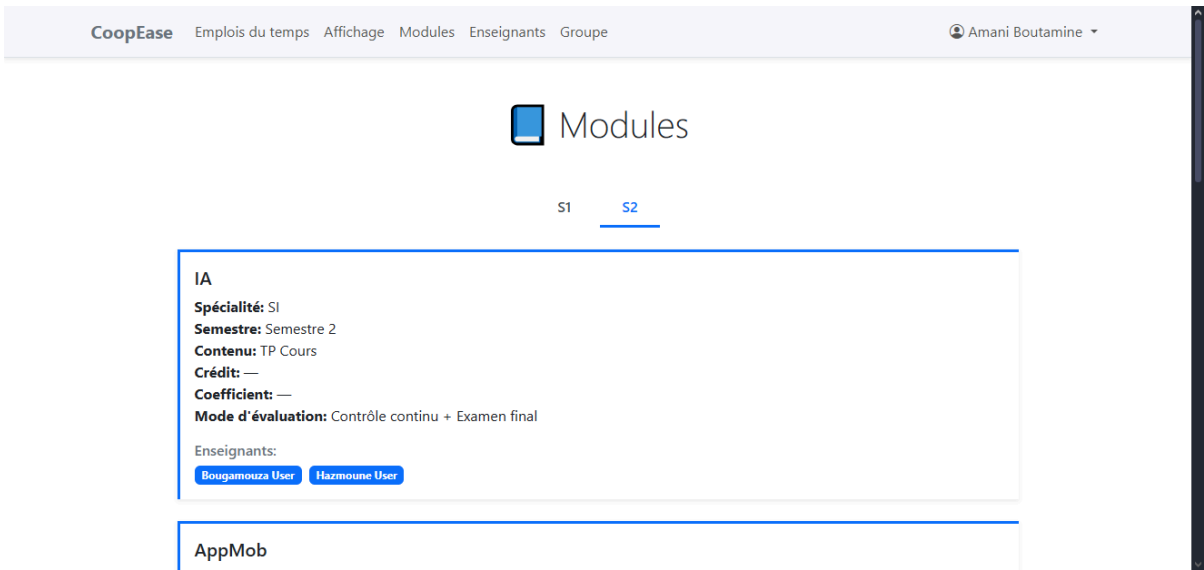


A circular profile picture of a man in a suit. Below it, the text reads: **Boulehouache User**, Boulehouache. **Role:** Responsable. **Email:** sboulehouache@yahoo.com

Figure 6.21 Interface du profil enseignant

## 2.16 Interface des modules

Les utilisateurs peuvent consulter les modules et voir les informations qui concerne chaque module.



CoopEase Emplois du temps Affichage Modules Enseignants Groupe Amani Boutamine

### Modules

S1 S2

**IA**  
**Spécialité:** SI  
**Semestre:** Semestre 2  
**Contenu:** TP Cours  
**Crédit:** —  
**Coefficient:** —  
**Mode d'évaluation:** Contrôle continu + Examen final

Enseignants:  
Bougamouza User Hazmoune User

AppMob

Figure 6.22 Interface modules par d'étudiant

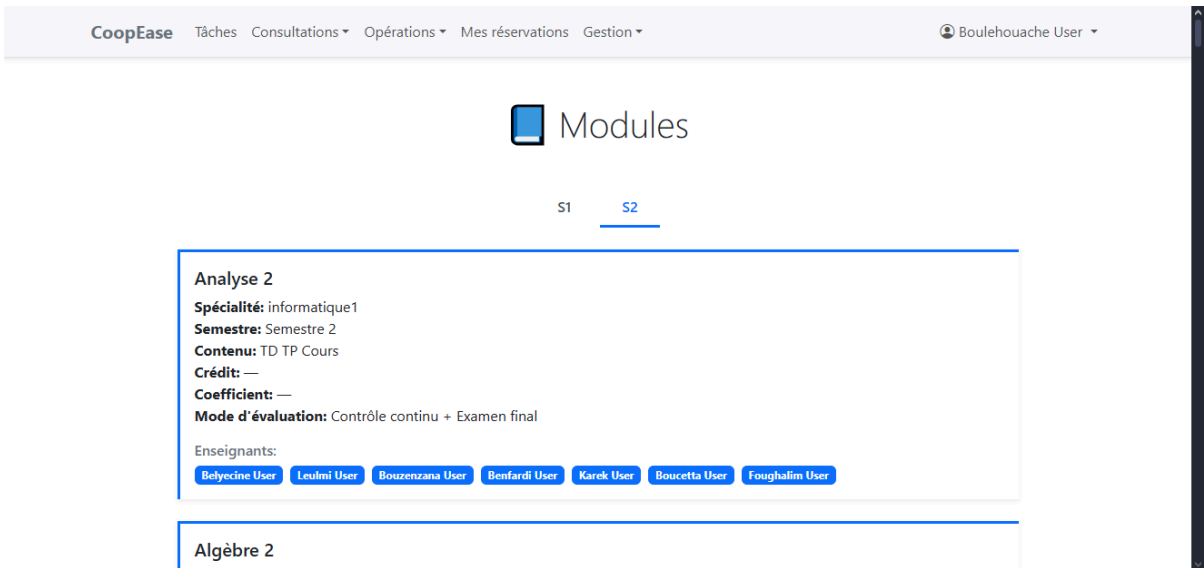


Figure 6.23 Interface modules par enseignant

## 2.17 Interface liste des enseignants

Les enseignants et les étudiants peuvent voir la liste des enseignants en cas de besoin et les contactent par leurs emails ou numéros de téléphone.

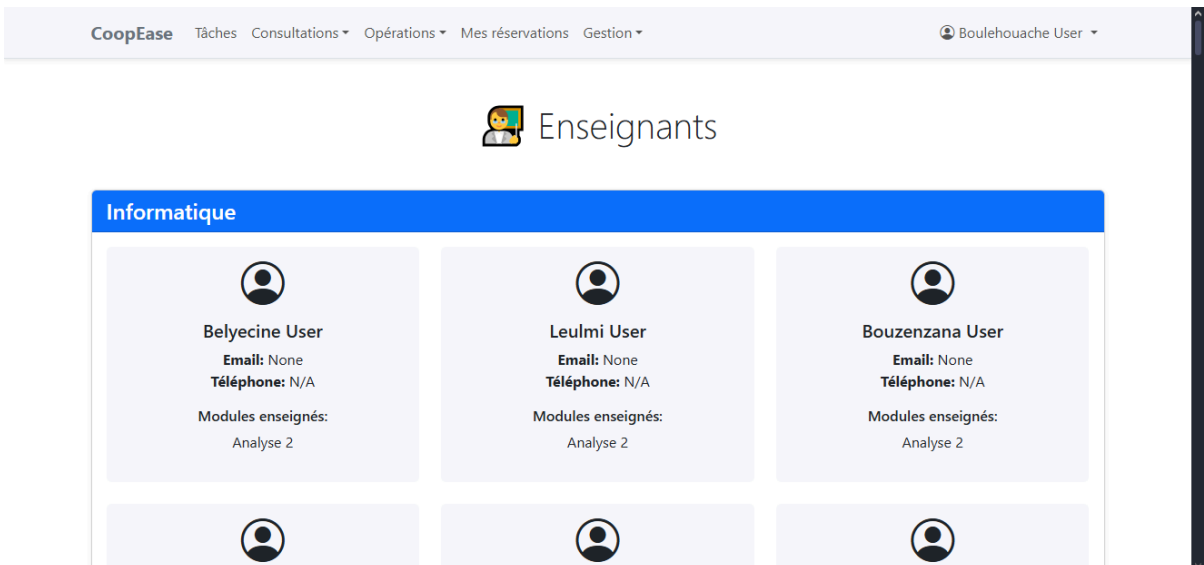


Figure 6.24 Interface liste des enseignants

## Enseignants - SI

Semestre S1 Semestre S2










 <b>Bougamouza User</b> Email: None Téléphone: N/A Modules enseignés: IA	 <b>Hazmoune User</b> Email: None Téléphone: N/A Modules enseignés: IA	 <b>Kara User</b> Email: None Téléphone: N/A Modules enseignés: AppMob
 <b>Boulehouache User</b> Email: sboulehouache@yahoo.com Téléphone: N/A Modules enseignés: AppMob	 <b>Foughali User</b> Email: None Téléphone: N/A Modules enseignés: DSStruc	 <b>Mansoul User</b> Email: None Téléphone: N/A Modules enseignés: DSStruc
 <b>Nafir User</b> Email: None Téléphone: N/A Modules enseignés: SecInfo	 <b>Nabet User</b> Email: None Téléphone: N/A Modules enseignés: RédSci	 <b>Mazouzi User</b> Email: None Téléphone: N/A Modules enseignés: CDStartup

Figure 6.25 Interface liste des enseignants par étudiant

## 2.18 Interface d'Équipements disponibles

Les enseignants peuvent voir l'équipement disponible dans le département en temps réel en cas de besoin.

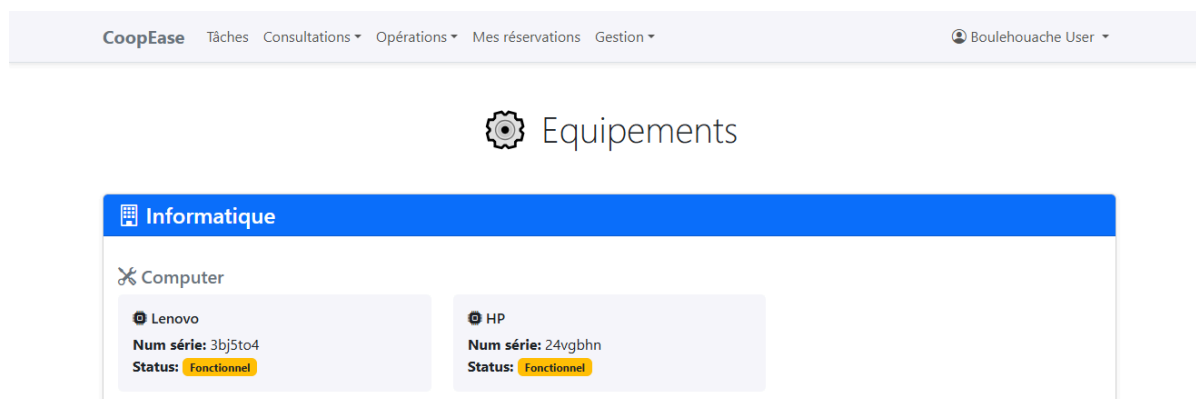


Figure 6.26 Interface liste d'équipements

## 2.19 Tableau de bord de l'administrateur pour la création des groupes

Ce tableau de bord fournit à l'administrateur un espace dédié pour créer, configurer et organiser les groupes pédagogiques nécessaires à la structuration des emplois du temps et à l'affectation des étudiants. Il constitue une étape préalable indispensable au bon fonctionnement des algorithmes de génération d'emploi du temps et d'affectation automatique gérés par Google OR-Tools.

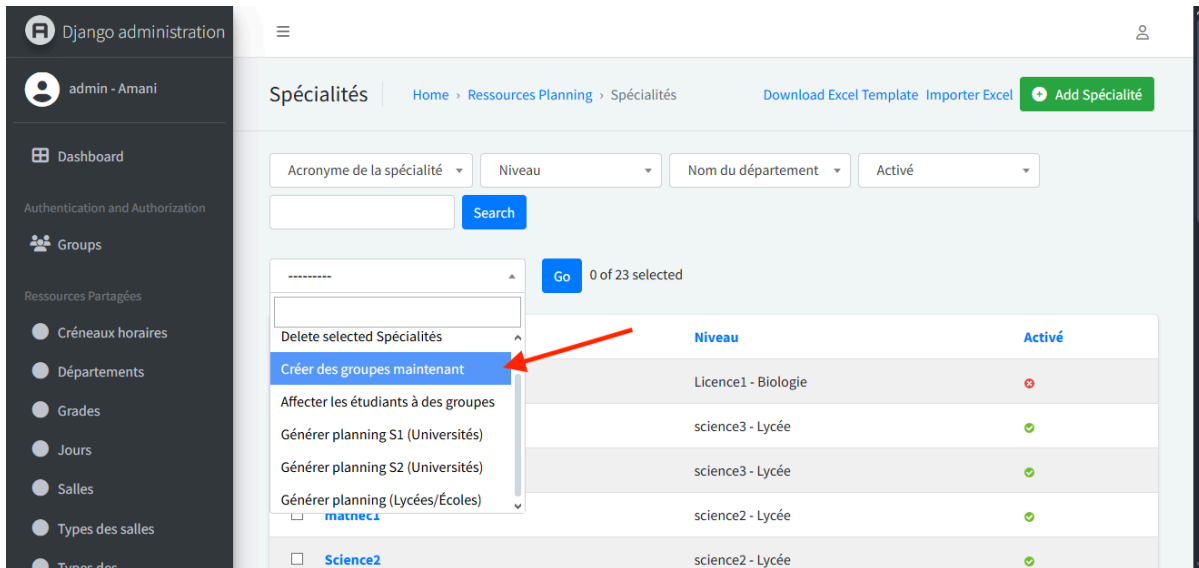


Figure 6.27 Interface de tableau de bord de l'administrateur et déclenchement de création des groupes

## 2.20 Tableau de bord de l'administrateur pour génération de l'emploi du temps

Le tableau de bord offre une vue centralisée sur la gestion des ressources (salles, enseignants, groupes, horaires, modules, spécialités). Il permet aussi de déclencher la génération automatique des emplois du temps via l'outil OR-Tools.

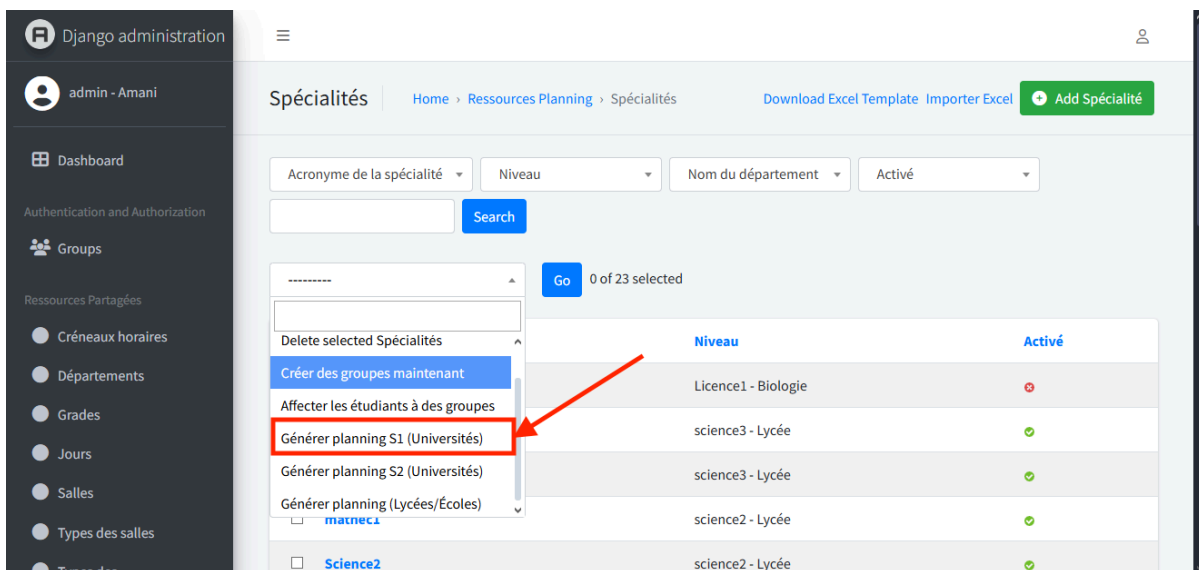


Figure 6.28 Interface de tableau de bord de l'administrateur et déclenchement de la génération automatique des emplois du temps

## 2.21 Tableau de bord de l'administrateur pour l'affectation des étudiants aux groupes

Cette interface permet à l'administrateur de superviser l'affectation automatique des étudiants aux groupes pédagogiques, réalisée à l'aide du moteur d'optimisation Google OR-Tools. L'affectation tient compte de plusieurs paramètres : disponibilités,

préférences, équilibre des effectifs, contraintes pédagogiques et capacités des groupes.

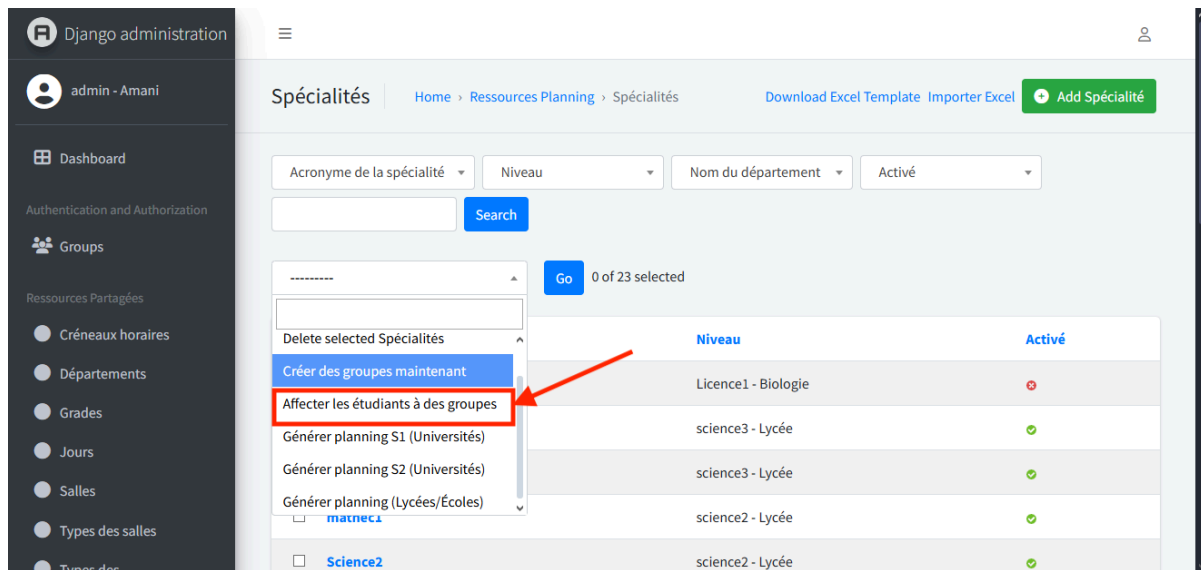


Figure 6.29 Interface de l'administrateur pour faire affectation des étudiants à des groupes

## 2.22 Tableau de bord de l'administrateur

Le tableau de bord de l'administrateur constitue une interface centrale permettant aux responsables (gestionnaires ou administrateurs système) d'accéder aux différentes fonctionnalités de gestion, de suivi, et de modification des données du système. Cette interface joue un rôle stratégique dans le maintien de la cohérence, la supervision des responsables dès l'établissement des opérations et l'adaptation rapide aux imprévus (changements d'enseignants, changements d'affectation, conflits d'emploi du temps...).

# Business Model Canvas BMC

Nom du projet : CoopEase  
Date : 01/06/2025










Partenaires clés 	Activités clés 	Propositions de valeur 	Relations clients 	Segments de marché 
<ul style="list-style-type: none"> <li>- Ministère de l'Éducation et de l'Enseignement Supérieur</li> <li>- Écoles et universités pilotes</li> <li>- Incubateurs / pépinières de startups</li> <li>- Fournisseurs de solutions cloud (ex : AWS, Azure, OVH)</li> </ul>	<ul style="list-style-type: none"> <li>- Développement continu de la plateforme</li> <li>- Tests avec établissements pilotes</li> <li>- Suivi client et maintenance</li> <li>- Communication et sensibilisation dans le secteur éducatif</li> </ul>	<ul style="list-style-type: none"> <li>- Automatisation intelligente de la planification</li> <li>- Optimisation de l'utilisation des ressources (salles, enseignants, groupes)</li> <li>- Réduction des conflits d'emploi du temps</li> <li>- Plateforme collaborative, intuitive et accessible</li> <li>- Importation automatique des listes d'étudiants via Excel</li> <li>- Notifications en temps réel (absences, remplacements)</li> </ul>	<ul style="list-style-type: none"> <li>- Accompagnement à l'intégration (support technique et pédagogique)</li> <li>- Organisation de sessions de formation pour les utilisateurs, réductions sur les abonnements.</li> <li>- Support client (chat, email)</li> <li>- Formations pour les administrateurs locaux</li> </ul>	<ul style="list-style-type: none"> <li>- Établissements éducatifs publics (universités, lycées, collèges)</li> <li>- Écoles privées</li> </ul>
	<b>Ressources clés</b>  <ul style="list-style-type: none"> <li>- Équipe de développement (full-stack, back-end, UX/UI)</li> <li>- Infrastructure serveur (cloud ou local)</li> <li>- Accès aux OR-Tools (optimisation IA)</li> <li>- Réseaux internet et ordinateurs et téléphone</li> </ul>		<b>Canaux</b>  <ul style="list-style-type: none"> <li>- Plateforme web en ligne (web app)</li> <li>Réseaux sociaux : Page Facebook, Chaîne YouTube, LinkedIn, Instagram, X (ex-Twitter)</li> <li>- Référencement via plateformes institutionnelles (ex : ministère, rectorats)</li> <li>- Partenariats avec universités ou d'établissements privées</li> </ul>	
		<b>Structure des coûts</b>  <ul style="list-style-type: none"> <li>- Développement et maintenance de l'application</li> <li>- Hébergement et infrastructure serveur</li> <li>- Marketing et prospection client</li> <li>- Support et formation des utilisateurs</li> </ul>	<b>Flux de revenus</b>  <ul style="list-style-type: none"> <li>- Abonnement annuel pour établissements (tarification selon la taille)</li> <li>- Installation en local (licence + frais d'intégration)</li> <li>- Version premium avec modules supplémentaires (statistiques, IA prédictive)</li> <li>- Frais d'abonnement + frais de licence.</li> </ul>	

Figure 6.30 Model BMC

## Conclusion Générale

Ce projet d'application web intelligente de gestion coopérative des ressources est né d'un constat réel : la complexité croissante de la planification des emplois du temps et de la coordination des ressources humaines et matérielles dans les établissements éducatifs, en particulier en Algérie. Face aux méthodes traditionnelles encore largement répandues, cette solution propose une rupture technologique, fonctionnelle et organisationnelle.

Conçu initialement dans le cadre d'un Master 2 en informatique, ce projet s'est transformé en une véritable initiative entrepreneuriale, avec une architecture technique robuste, une approche orientée utilisateur, et un potentiel de déploiement à grande échelle. En s'appuyant sur les technologies modernes telles que Django, Google OR-Tools et une interface intuitive, l'application permet de réduire considérablement les erreurs humaines, d'optimiser les ressources et d'améliorer la qualité de vie des étudiants, enseignants et gestionnaires.

Grâce à sa modularité, sa flexibilité et son modèle économique adapté, la plateforme est en mesure de répondre aux besoins des établissements publics comme privés, et d'accompagner durablement leur transition numérique.

L'ambition de ce projet ne s'arrête pas à sa première version : une application mobile, une interface encore plus intuitive, des analyses prédictives et une intégration inter-établissements sont autant d'évolutions prévues pour les prochaines étapes.

Nous croyons fermement que cette solution peut devenir un outil central pour moderniser la gestion des établissements éducatifs, et contribuer activement à une école algérienne plus efficace, plus équitable et plus connectée.