

Projet de Session

# HorAlreUQO

Spécifications des exigences d'un logiciel (ADAPTE DE LA NORME IEEE 830-1993)

**Présenté au professeur :**

Karim El Guemhioui

**Réalisé par :**

**KACEM RASLEN : KACR78360306**

**LEMANI JOEL PATRICK : LEMJ22378107**

**NKONDJE NKONDJE EKENGLO ALEX : NKOA13108108**

**NGONO MESSOGUI AMELIE :**

## Table des matières

1. Introduction.....	3
2. Objectif du Document.....	4
3. Portée du Logiciel.....	4
4. Définitions.....	4
5. Description Générale.....	5
5.1 Fonctionnalités Principales.....	5
5.2 Acteurs.....	6
6. Exigences Spécifiques.....	6
6.1 Exigences comportementales.....	6
6.2 Exigences Non-comportementales.....	7
6.2.1 Exigences de Performance.....	7
6.2.2 Exigences d'Interface Utilisateur.....	7
6.2.3 Exigences de Sécurité.....	8
6.2.4 Exigences de Fiabilité.....	8
7. Liste des acteurs et des évènements Associés.....	8
8. Diagramme de cas d'utilisation.....	10
9. Cas d'utilisation prioritaire.....	11
9.1 Cas d'utilisation 1 : Inscrire au cours.....	11
9.1.1 Scénario typique (cas normal).....	12
9.1.2 Scénarios alternatifs (cas d'exception).....	12
9.2 Cas d'utilisation 2 : Créer l'horaire de cours.....	13
9.2.1 Scénario typique (cas normal).....	14
9.2.2 Scénarios alternatifs (cas d'exception).....	14
10. Conclusion.....	15
11. Document de référence.....	16

## 1. Introduction :

Le logiciel de planification des cours vise à résoudre les défis liés à l'élaboration de l'horaire des cours pour le Module de l'informatique et le module de l'ingénierie à l'Université du Québec en Outaouais (UQO). Ce défi est encore plus complexe en raison de la diversité des programmes offerts. Pour répondre à ces défis, le développement d'un logiciel de planification des cours devient impératif. Ainsi, ce document énonce de manière exhaustive les exigences comportementales et non comportementales pour HorAlreUQO, le futur logiciel de planification d'horaires de l'UQO, afin de garantir une gestion efficace et optimisée des horaires de cours.

Le présent document explore les principales caractéristiques de HorAlreUQO, son objectif, sa portée, ainsi que les définitions pertinentes. Il examine également les fonctionnalités principales, les acteurs impliqués, et les exigences spécifiques, comportementales et non-comportementales. En outre, il présente les cas d'utilisation prioritaires, notamment l'inscription au cours et la création de l'horaire de cours, tout en décrivant les scénarios typiques et alternatifs pour chaque cas.

En résumé, ce cahier des charges établit un cadre détaillé pour le développement de HorAlreUQO, visant à fournir un outil robuste, intuitif et efficace pour la gestion des horaires de cours à l'UQO, tout en répondant aux besoins variés des étudiants, des enseignants et du personnel administratif.

## 2. Objectif du Document :

Ce cahier des charges vise à définir de manière exhaustive les exigences comportementales et non comportementales que HorAlreUQO, le futur logiciel de planification d'horaires de l'UQO, devra satisfaire pour répondre aux besoins de gestion des horaires de cours complexes et variés.

## 3. Portée du Logiciel:

HorAlreUQO est conçu pour automatiser et optimiser la création des horaires de cours pour le Module d'Informatique et le Module d'Ingénierie, en tenant compte de multiples contraintes et exigences.

## 4. Définitions:

- **HorAlreUQO** : Est un nom concis, facile à retenir et à prononcer, ce qui est bénéfique pour le marketing et la communication autour du logiciel. Il encapsule l'essence du logiciel un outil innovant, spécifiquement conçu pour l'UQO, utilisant l'intelligence artificielle pour optimiser la planification des horaires.
- **Les exigences comportementales** : Ce sont les critères qui peuvent être utilisés pour le fonctionnement d'un système. Elles peuvent inclure les performances, sécurité, l'interface utilisateur, comptabilité, qualité du code et maintenabilité.
- **Diagramme** : Un diagramme est une représentation graphique ou symbolique d'un ensemble d'informations, de données, de processus ou de concepts, conçu pour aider à la compréhension, à la communication ou à la visualisation de ces éléments. Les diagrammes

sont largement utilisés dans de nombreux domaines, tels que l'informatique, l'ingénierie, la gestion de projet, la science, les affaires et d'autres disciplines, pour organiser visuellement des idées complexes, des relations ou des structures.

## 5. Description Générale :

### 5.1 Fonctionnalités Principales :

- **Gestion des Contraintes** : Capable de gérer et d'appliquer des contraintes spécifiques pour la création des horaires.
- **Optimisation des Horaires** : Utilise des algorithmes d'optimisation pour proposer les horaires les plus efficaces.
- **Interface Utilisateur Intuitive** : Fournit une interface facile à utiliser pour les administrateurs et le personnel académique.
- **Rapports et Analyses** : Génère des rapports détaillés sur les horaires, les conflits, et les suggestions d'amélioration.

### 5.2 Acteurs :

- Personnel administratif de l'UQO
- Enseignant du Module d'Informatique et du Module d'Ingénierie
- Étudiant

## 6. Exigences Spécifiques :

### 6.1 Exigences comportementales :

1. Le système doit permettre aux administrateurs d'ajouter, de modifier ou de supprimer facilement des cours et leurs horaires.
2. Le système doit vérifier que les étudiants satisfont aux prérequis des cours lors de l'inscription.
3. Le système doit être capable de proposer automatiquement des horaires de cours optimisés en fonction des contraintes établies.
4. Permettre aux étudiants inscrits à temps complet de pouvoir prendre tous les cours prévus à leur programme à chaque session.
5. Permettre aux étudiants à temps complet reconnus pour un cours, de pouvoir suivre le nombre de crédits prescrits par leur statut à chaque session.
6. Permettre aux étudiants à temps complet ayant échoué un cours, de pouvoir suivre le nombre de crédits prescrits par leur statut à chaque session.
7. Permettre à un plus grand nombre d'étudiants éligibles d'accéder aux cours optionnels.
8. Empêcher les conflits horaires dans la programmation des cours en commun entre informatique et ingénierie.
9. Empêcher les étudiants suivant un cheminement régulier de ne pas avoir plus de deux activités par jour.
10. Empêcher les étudiants suivant leur cheminement régulier de ne pas avoir une activité dans la dernière tranche horaire suivie d'une au début du lendemain.
11. Empêcher les séances de TP d'avoir lieu dans la dernière tranche horaire de la journée.

12. Rendre disponible les locaux et du matériel nécessaire pour les séances de TD ou de TP programmées simultanément.
13. Empêcher la programmation de deux cours dispensés par un même enseignant dans la même journée.
14. Le système doit envoyer des notifications aux étudiants et aux enseignants en cas de changement d'horaire ou de salle.

## **6.2 Exigences Non-comportementales :**

### **6.2.1 Exigences de Performance :**

- Le système doit afficher les horaires et traiter les inscriptions en quelques secondes pour assurer une expérience utilisateur fluide.
- Le système doit être capable de gérer un grand nombre d'utilisateurs simultanés, spécialement lors des périodes d'inscription.
- 

### **6.2.2 Exigences d'Interface Utilisateur :**

- L'interface doit être accessible et facile à utiliser pour tous les utilisateurs, y compris ceux ayant des besoins spécifiques.
- Le système doit être compatible avec différents appareils et navigateurs web.

### 6.2.3 Exigences de Sécurité :

- Le système doit permettre la protection des données personnelles des étudiants et des enseignants contre les accès non autorisés.
- Le système doit permettre d'implémenter une authentification robuste pour les utilisateurs afin de sécuriser l'accès au système.

### 6.2.4 Exigences de Fiabilité :

- Le système doit être disponible durant les heures critiques, notamment pendant les périodes d'inscription aux cours.
- Le système doit être capable de récupérer rapidement en cas de panne pour minimiser les interruptions de service.

## 7. Liste des acteurs et des événements Associes :

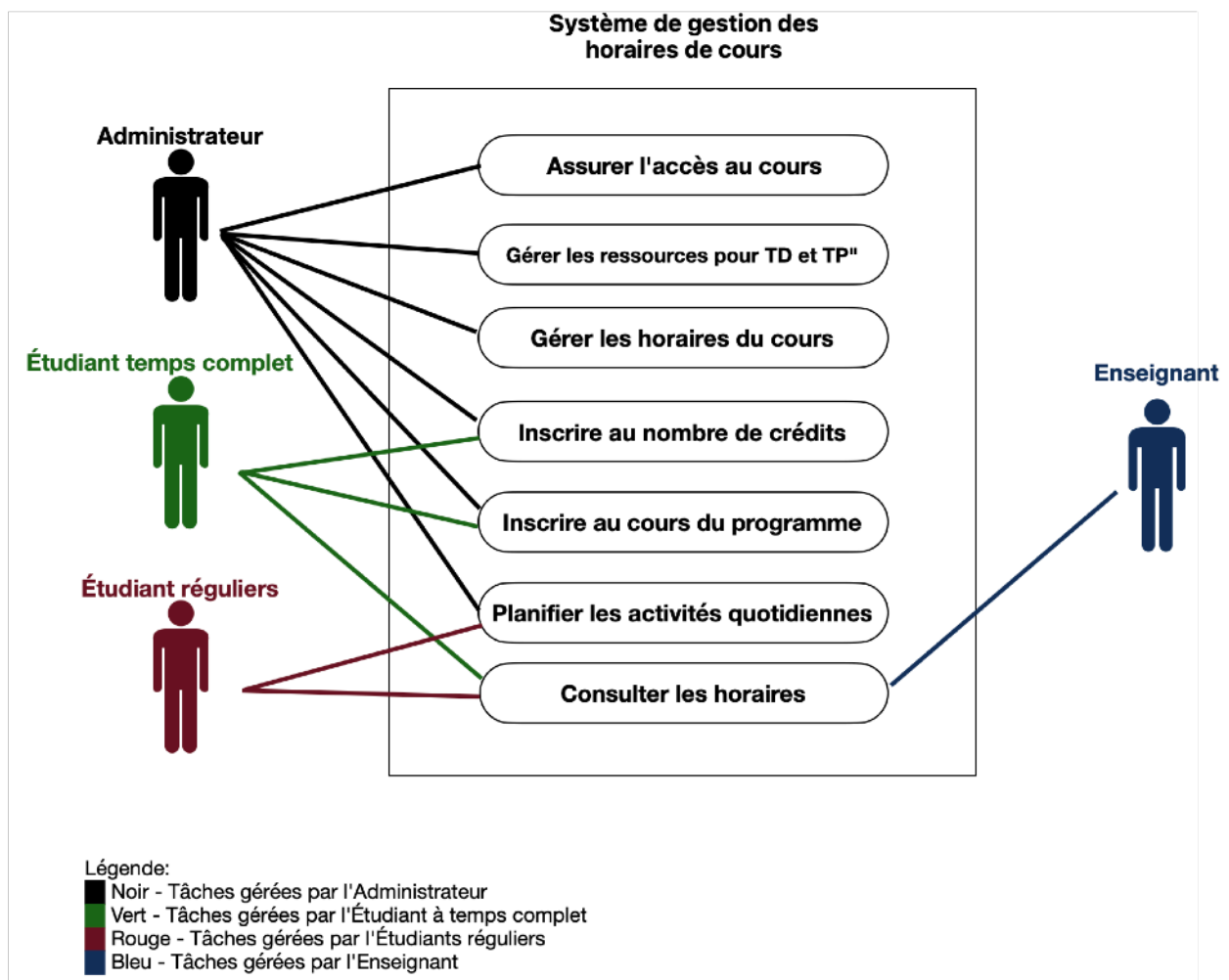
- **Système de gestion des horaires de cours** : C'est le système principal que le diagramme décrit, centré sur la gestion des horaires de cours au sein d'une institution éducative.
- **L'Administrateur** est la pierre angulaire du système, jouant un rôle critique dans :
  - **Gestion des Inscriptions** : Supervision de l'inscription des étudiants aux cours, veillant à l'alignement avec les exigences académiques.



- **Gestion des Ressources** : Allocation des espaces et équipements pédagogiques, garantissant une utilisation optimale des infrastructures.
  - **Gestion des Horaires** : Établissement des horaires des cours, maximisant la disponibilité des enseignants et l'efficacité des emplois du temps.
  - **Résolution de Conflits** : Résolution des chevauchements d'horaires pour maintenir un fonctionnement harmonieux de l'emploi du temps académique.
- **L'Enseignant** interagit avec le système pour :
  - **Déclaration des Disponibilités** : Communication des périodes disponibles pour enseigner, essentielle à la planification des cours.
  - **Soumission des Besoins** : Expression des besoins spécifiques en termes de ressources pour les cours programmés.
- **L'Étudiant Régulier** utilise le système pour :
  - **Consultation des Horaires** : Visualisation des emplois du temps pour organiser son calendrier académique.
  - **Planification des Activités** : Coordination des activités journalières dans le respect des contraintes d'emploi du temps personnel.
- **L'Étudiant à Temps Complet** bénéficie des fonctionnalités suivantes :

- **Inscription aux Cours** : S'inscrire aux cours requis, avec un système d'ajustement dynamique basé sur les crédits académiques.
- **Gestion des Crédits** : Administration des crédits en fonction des résultats académiques et des prérequis des cours.

## 8. Diagramme de cas d'utilisation :



## 9. Cas d'utilisation prioritaire :

On va examiner deux cas à savoir :

### 9.1 Cas d'utilisation 1 : Inscrire au cours

<b>Titre</b>	<b>Inscrire au cours</b>
<b>Acteurs</b>	( primaire) Étudiant  (Secondaire) Aucun
<b>But</b>	Décrire le cas d'utilisation pour s'inscrire au cours
<b>Description sommaire (vue d'ensemble</b>	L'étudiant accède au système et s'inscrit aux cours en respectant son cheminement
<b>Catégorie (primaire, secondaire, optionnel)</b>	Primaire
<b>Nature (essentiel ou réel)</b>	Essentiel
<b>Cas complémentaires</b>	Gestion des listes d'attente
<b>Préconditions</b>	L'étudiant est identifié et l'horaire des cours est disponible
<b>Postconditions</b>	L'étudiant est inscrit aux cours choisis

#### 9.1.1 Scénario typique (cas normal) :

Actions des acteurs	Réponses du système
<p>1. Ce cas d'utilisation débute lorsque l'étudiant s'est déjà identifié et les et l'horaire des cours sont disponibles.</p> <p>2. Il recherche les cours requis selon son programme d'études.</p> <p>3. Il sélectionne les cours et les ajoute à son panier d'inscription.</p> <p>6. L'étudiant confirme son inscription.</p>	<p>4. Le système vérifie que les cours sélectionnés n'est pas complet, ne présentent pas de conflits d'horaire et que l'étudiant répond aux prérequis.</p> <p>7. Le système enregistre les cours choisis.</p> <p>8. Met à jour l'emploi du temps de l'étudiant.</p> <p>9. Envoie une confirmation par courriel.</p>

### 9.1.2 Scénarios alternatifs (cas d'exception) :

#### Scénario Alternatif : Cours Complet

3.a L'étudiant essaie de s'inscrire à un cours qui est déjà complet.

3.b Le système informe l'étudiant que le cours est complet et offre la possibilité de s'inscrire sur une liste d'attente.

3.c L'étudiant accepte de s'inscrire sur une liste d'attente.

3.d Le système ajoute l'étudiant sur la liste d'attente et lui envoie une modification.

## Scénario Alternatif : Conflit d'Horaire

3.a L'étudiant tente de s'inscrire à deux cours qui se chevauchent.

3.b Le système détecte le conflit d'horaire et alerte l'étudiant.

3.c L'étudiant doit choisir entre les cours en conflit avant de pouvoir finaliser son inscription.

### 9.2 Cas d'utilisation 2 : Créer l'horaire de cours :

<b>Titre</b>	<b>Créer l'horaire de cours</b>
<b>Acteurs</b>	(primaire) Administration (Secondaire) Aucun
<b>But</b>	Décrire le cas d'utilisation pour la création d'horaire de cours
<b>Description sommaire (vue d'ensemble)</b>	L'administrateur accède au système et crée l'horaire de des cours pour chaque session
<b>Catégorie (primaire, secondaire, optionnel)</b>	Primaire
<b>Nature (essentiel ou réel)</b>	Essentiel
<b>Cas complémentaires</b>	Gestion des conflits d'horaire
<b>Préconditions</b>	L'administrateur est identifié et a accès au système
<b>Postconditions</b>	L'horaire est mis à jour dans le système

### 9.2.1 Scénario typique (cas normal) :

Actions des acteurs	Réponses du système
1. Ce cas d'utilisation débute lorsque l'administrateur s'est déjà identifié et est connecté au système.	
2. L'administrateur sélectionne l'option de créer d'horaire.	
	3. Le système présente une interface pour créer les horaires
4. L'administrateur saisit les informations des cours, incluant les contraintes spécifiques.	
	5. Le système vérifie les contraintes et valide les informations.
6. L'administrateur confirme la création.	7. Le système enregistre l'horaire créé avec succès et affiche un message de confirmation de la création de l'horaire.

### 9.2.2 Scénarios alternatifs (cas d'exception) :

5.a Le système détecte une contrainte spécifique non respectée (par exemple, un chevauchement entre deux .

5.b Le système affiche un message d'erreur indiquant la contrainte.

5.c Le système invite l'administrateur à corriger les informations saisies et retour à 4.

5.d l'administrateur apporte les corrections nécessaires.

5.e Retour à 5. Si succès passe à 6. Sinon retourne à 5a .

## 10. Conclusion :

En conclusion, le développement du logiciel de planification des cours, HorAlreUQO, représente une étape cruciale pour l'Université du Québec en Outaouais (UQO) afin de relever les défis complexes liés à la gestion des horaires de cours. Ce cahier des charges a permis de définir de manière exhaustive les exigences comportementales et non comportementales pour HorAlreUQO, garantissant ainsi un cadre solide pour son développement et sa mise en œuvre.

En répondant aux besoins variés des acteurs impliqués, tels que les étudiants, les enseignants et le personnel administratif, HorAlreUQO vise à automatiser et à optimiser la création des horaires de cours pour les modules d'informatique et d'ingénierie. Les fonctionnalités principales, telles que la gestion des contraintes, l'optimisation des horaires, et les rapports détaillés, ainsi que les exigences de performance, d'interface utilisateur, de sécurité et de fiabilité, ont été rigoureusement définies pour assurer une expérience utilisateur fluide et sécurisée.

De plus, les cas d'utilisation prioritaires, comme l'inscription au cours et la création de l'horaire de cours, ont été examinés en détail, offrant des scénarios typiques et alternatifs pour guider le développement et tester la fonctionnalité du logiciel.

En mettant en œuvre HorAlreUQO, l'UQO peut s'attendre à une optimisation significative de ses processus de planification des cours, contribuant ainsi à améliorer l'efficacité opérationnelle et à garantir une expérience d'apprentissage harmonieuse pour ses étudiants. En somme,

HorAlreUQO représente un outil essentiel pour l'UQO dans sa quête d'excellence académique et organisationnelle.

## 11. Document de référence :

- Notes des Cours et TD GEN1423-01 Génie logiciel
- [Software Engineering](#)  
10ème ed.  
*Ian Sommerville*  
Addison-Wesley, 2016
- Exemple de cahier de charge fait par les étudiants de L'UQAM