

Gestion Pédagogique à la Sonatel Academy

L'ODC fait appel à vous pour la réalisation d'une application Web/Mobile de planification des cours et la gestion des absences

1. Module Planification des Cours

Chaque début d'année le Responsable Pédagogique(RP) gère les ressources pédagogiques de l'année :

1. Créer l'année scolaire
2. Créer les nouvelles classes (libelle,filière,niveau)
3. Planifier les classes qui seront ouvertes
4. Ajouter les nouveaux professeurs(nom,prénom,spécialité,grade).
5. Créer les salles(nom,numéro et nombre de places)
6. Créer les Semestres
7. Créer les Modules(libellé)
8. Planifier un cours (nombre heure global,semestre)
9. Planifier les sessions de cours (date,heure début,heure fin, nbre Heure,)
10. Inscrire les étudiants

NB : Les Classes,Professeurs et Salles sont créées une et une seule fois dans le système mais on peut les utiliser ou pas dans une année Scolaire.

Règles de Gestion

1. Un cours est enseigné par un professeur devant une ou plusieurs classes.
2. Dans un cours on enseigne qu'un seul module.
3. Un cours est organisé en plusieurs sessions de cours.
4. Une session de cours peut se faire en ligne ou en présentiel.Lorsque la session se fait en présentiel on devra préciser la salle.
5. Gérer la disponibilité des ressources(professeurs,Salle) lors de la planification des sessions de cours.
6. Chaque cours a un quota horaire globale a ne pas dépasser lors de la planification des ses sessions
7. L'inscription des étudiants est faite par une importation d'un fichier excel. Le fichier contient les informations de l'étudiant ,les

informations inscriptions. Les inscriptions se font toujours dans l'année scolaire en cours.

Un RP a la possibilité de lister les cours planifiés et de filtrer par état(Terminé ou En cours) et éventuellement visualiser les sessions de cours qui ont été faites, voir les étudiants qui suivent le même cours et peut filtrer par classe. Il peut aussi annuler une session de cours .

Un professeur peut lister ses cours et éventuellement visualiser les sessions de cours qui lui sont programmés . Il peut filtrer ses cours par période(jour, semaine). Il peut faire une demande d'annulation en cas d'empêchement.

Le professeur peut lister le nombre d'heures de cours qu'il a effectuées durant le mois et peut éventuellement filtrer par module.

A la fin de chaque session de chaque cours l'attaché devra valider la session de cours pour qu'on puisse le comptabiliser.

L'attaché peut lister les sessions de cours effectuées par un professeur durant le mois en cours et peut éventuellement filtrer par module en ayant des vues sur le nombre d'heures globales , déroulées et restants . Il peut aussi valider ou invalider une session de cours.

Un étudiant peut lister ses cours et éventuellement visualiser les sessions de cours qui lui sont programmés . Il peut filtrer ses cours par période(jour, semaine) et par modules.

2. Module Gestion des absences Cours

Durant une session de cours un étudiant peut marquer sa présence sur liste d'émargement. Cette présence sera validée par l'attache à la fin de la session .

L'étudiant peut visualiser ses absences , le nombre d'heures d'absence cumulées durant un semestre.

Il peut aussi justifier(date, motif) son absence qui devra être traitée (acceptée ou refusée) par l'attaché

Les absences acceptées ne sont pas comptabilisées.

Une absence est caractérisée par sa date , l'étudiant et la session de cours.

Règles de Gestion

1. Liste d'émargement n'est disponible que 30 minutes après le début du cours.
2. Une absence ne peut être justifiée qu'une seule fois.
3. Lorsqu' un étudiant atteint 10 H d'absences , il doit recevoir une notification d'avertissement, Après 20H une lettre de convocation

1.Modélisation

- Diagramme Use Case
- Diagramme de Classe

2.Contraintes

- a. Les Fonctionnalités de **l'attache** du **Professeur** et du **RP** seront réalisés avec **Angular**
- b. Les Fonctionnalités de **l'étudiant** seront réalisés avec **Flutter**

3.Livrables

- a. Définir les US du Projet.Un US est formé de fonctionnalités Front et Back.
- b. Planifier au moins 2 Sprints
- c. Livrer au moins 2 US par jour
- d. Créer le GitHub du projet et faire des push a chaque fois qu'un USA est terminé.

4.Délai

- a. Date fin du Projet Samedi **21 octobre 2023**