

Projet 1: Gestion des inscriptions

L'Ecole 221 fait appel à vous pour la réalisation d'une application Web de gestion des inscriptions de l'école.

Chaque début d'année le Responsable Pédagogique(RP) peut créer,lister des classes (libelle,filiere,niveau).Il a la possibilité aussi d'ajouter un professeur ainsi que ses modules,lui affecter des classes ,lister les professeurs(nom complet,grade) et lister les classes d'un professeurs.

Un professeur peut enseigner plusieurs modules et on voudrait lister les modules enseignés par un professeur ainsi les professeurs qui enseignent un module.

Les Attachés de classe font les inscriptions et les réinscriptions des étudiants durant la période d' inscription.Un étudiant peut s'inscrire plusieurs fois mais une seule fois dans une année.

Un Attaché a la possibilité de lister les étudiants (matricule,nom complet,adresse) inscrits dans une classe dans une année scolaire. sr

Un étudiant à la possibilité d'annuler ou de suspendre son inscription dans ce cas il devra en formuler la demande(motif,date) dans l'application.Cette demande est reçue et traitée(acceptée ou refusée) par le RP .Un étudiant peut lister ses demandes et de les filtrer par état alors que l'attaché peut lister les demandes de tous les étudiants inscrits à partir d'une recherche sur le matricule.

On voudrais avoir les statiques suivantes:

- l'effectif de l'école par année,
- le nombre de fille ou de garçon par année
- l'effectif par classe,
- le nombre de fille ou de garçon par classe
- le nombre d'étudiants qui ont suspendu ou annulé leur inscription par année.

NB:Toutes les connexions sont accessibles après connexion

TAF:

- 1) Définir les Acteurs**
- 2) Diagramme de Contexte**
- 3) Fonctionnalités par acteur**
- 4) Diagramme de Package**
- 5) Diagramme de Use Case**

Projet 2: Gestion immobilière

L'agence immobilière TECH 221 qui propose la location de tous types d'hébergement (maison ,appartement,studio et immeuble).

Le Gestionnaire de l' agence souhaite ajouter,lister et archiver ses logements(type,référence,adresse,surface, des images). Un logement se trouve dans une zone et appartient à un propriétaire (nom,prénom,adresse,adresse).

Un propriétaire signe un contrat de gestion (date, montant,durée) avec l'agence pour la gestion d'un logement. Le prix du loyer est égal au montant à verser au propriétaire + un forfait qui dépend de la zone + caution fixe .

Le gestionnaire a la possibilité de lister les contrats de gestion encours des propriétaires .Il peut filtrer cette liste par propriétaire et par état car un propriétaire peut annuler un contrat de gestion.

Un propriétaire peut lister ses contrats de gestion ,peut voir le cumul des montants qu'on doit lui verser chaque fin de mois et ses logements qu'il pourra filtrer par état(disponible ou pas).

Un visiteur peut voir le catalogue de logement disponible et à la possibilité de faire une demande de location lorsqu'il est connecté.

Le client (visiteur qui a un compte) peut lister ses demandes de location ,ses contrats de location en cours ou terminés. La demande de location est validée par le Responsable des Locations (RL) .Dès que la demande de location est validée il pourra se déplacer à l'agence pour finaliser et signer le contrat .

Lors de l'établissement du contrat de location on renseignera la date de début, le montant du loyer,la durée ainsi que le logement , le client ,le contrat signé et les pièces justificatives(Bulletin salaire,casier judiciaire).

Un logement peut faire l'objet de plusieurs demandes mais d'un seul contrat en cours.

Chaque client est caractérisé par son nom, prénom, date de naissance et numéro de téléphone.

Chaque client peut signer plusieurs contrats de location.

A la fin de chaque mois les clients doivent payer leur loyer au niveau de l'agence avant le 05 auprès de la responsable financière(RF).

Ce dernier enregistre le loyer de ce mois , à la possibilité d'afficher les retardataires et même filtrer par mois ou par client.

NB:Toutes les fonctionnalités du Gestionnaire,Client , RF et RL sont accessibles après connexion.

TAF:

- 1) Définir les Acteurs**
- 2) Diagramme de Contexte**
- 3) Fonctionnalités par acteur**
- 4) Diagramme de Package**
- 5) Diagramme de Use Case**

Projet 3 : Gestion Bibliothèque

La bibliothèque de l'Ecole E221 fait appel à vous pour la réalisation d'une application web de gestion des prêts.

La Bibliothèque dispose d'un certain nombre d'ouvrages, classés par rayon (Littérature, Histoire, Géographie, etc.).

Chaque ouvrage(code ,titre,date édition) est l'œuvre d'un ou plusieurs auteurs(nom,prénom,profession), et doit également être référencé selon un certain nombre de mots-clés.

Dans un ouvrage on peut en avoir plusieurs exemplaires .

Un exemplaire est caractérisé par son code , sa date d'enregistrement.

Le Responsable de la Bibliothèque(RB) à la possibilité d'ajouter,lister et les ouvrages,les rayons et les auteurs.Il peut aussi ajouter des exemplaires à un ouvrage ,ainsi qu' archiver les exemplaires perdus ou détériorés.Les exemplaires sont rangés dans des rayons.

Un visiteur peut consulter les catalogue d'ouvrages disponibles et faire une demande de prêt d'exemplaire lorsqu'il aura créé un

compte.L'adhérent (visiteur qui a un compte) peut lister ses demandes de prêts,ses prêts en cours ou retournés. La demande de prêt est

validée par le Responsable des Prêts(RP) et dès que la demande est validée il peut passer à la bibliothèque pour récupérer l'exemplaire

d'ouvrages .Lors de la récupération le RP enregistrera le prêt.

Un prêt est caractérisé par sa date,sa date de retour prévu et sa date de retour réel,l'exemplaire d'ouvrage et l'adhérent.

Chaque adhérent peut emprunter un seul livre en même temps.La durée d'un prêt est de 15 jours.

Le RB doit pouvoir lister les œuvres disponibles ainsi que leurs exemplaires et les rayons où ils se trouvent.

Il veut savoir les exemplaires indisponibles(en pret ou détériorés).

Le RP veut savoir les prêts en cours dans la bibliothèque ainsi que les prêts d'un adhérent.Il doit afficher les prêts adhérents retardataires.

NB:Toutes les fonctionnalités

TAF:

1) Définir les Acteurs

- 2) Diagramme de Contexte**
- 3) Fonctionnalités par acteur**
- 4) Diagramme de Package**
- 5) Diagramme de Use Case**