

Travail de session

420-345-RI PROGRAMMATION D'INTERFACE GRAPHIQUE

Table des matières

I.	Équipes et remise du travail	1
1.	Équipes.....	1
2.	Remise du travail	1
II.	Mise en contexte.....	2
III.	Utilisation du programme.....	3
1.	Partie administrateur	3
a)	Authentification	3
b)	Pages de l'administrateur	3
c)	Statistiques.....	3
d)	Exportation de données.....	3
2.	Partie publique	4
IV.	Exigences du programme.....	5
1.	Expérience utilisateur.....	5
2.	Au niveau du code.....	5
3.	Sécurité, robustesse et cohésion.....	5
4.	Travail collaboratif.....	5

I. Équipes et remise du travail

1. Équipes

Le travail est à réaliser en équipe de **2 personnes exactement**

Les deux membres de l'équipe doivent partager leur travail sur GitHub.

2. Remise du travail

La date de remise du travail est le **11 Décembre 2024 à 15h00**

Le travail doit être remis dans un fichier compressé contenant :

- Le projet WinUi
- Un fichier texte avec les noms des membres de l'équipe et le lien vers GitHub

Les travaux remis en retard ne seront pas acceptés.

II. Mise en contexte

Un centre de loisirs organise sur l'année différentes activités (escalade, raquette, cirque, tricot, rock acrobatique, ...). On s'intéresse à la gestion de la participation des adhérents aux différentes activités organisées par le centre, dans le but d'établir des statistiques sur la pratique des activités, et pouvoir ainsi mieux les planifier et mieux connaître la population qui les pratique.

Les informations répertoriées sur les **adhérents** sont : un **numéro d'identification**, leur **nom**, leur **prénom**, **adresse**, la **date de naissance** et **l'âge** calculé par un déclencheur automatiquement après l'insertion de la date de naissance.

Le numéro d'identification est composé des initiales de l'adhérent, de son année de naissance et d'un nombre aléatoire de 3 chiffres, le tout séparé par des tirets.

Exemple : une personne se nommant Alice Trudel née le 15 mai 2003 aura comme numéro d'identification **AT-2003-152**.

ATTENTION : un adhérent doit avoir **au moins 18 ans** lors de son inscription.

Pour chaque adhérent, on va enregistrer également la liste des séances d'activités auxquelles il participe.

Les **activités** proposées au catalogue du centre sont référencées par leur **nom** et **leur type**. Pour chaque activité, il y aura d'une part le **coût d'organisation par client** et le **prix de vente par client**.

Exemple : pour une activité donnée, le coût d'organisation par client est de 150\$ et le prix de vente pour la participation par client est de 225\$.

Une activité peut se donner sur différentes **séances**. Pour chaque séance d'activité, on veut pouvoir retrouver ses **dates** et **heures** d'organisation, le **nombre de places disponibles**, ainsi qu'une **note d'appréciation** attribuée individuellement par chaque client qui s'y est inscrit.

Il est à noter que le site du centre est géré par un administrateur qui s'identifie par un mot de passe pour accéder à son interface personnalisée. Il pourra ainsi ajouter, supprimer ou modifier les informations sur les participants.

C'est également l'administrateur qui crée les activités, avec le prix d'organisation et le prix de vente. Il crée également les différentes catégories d'activité (escalade, raquette, etc.).

C'est aussi l'administrateur qui crée les séances pour une activité.

Enfin, il peut afficher les statistiques sur les activités (nombre de participants total, le participant le plus actif, la moyenne des notes d'appréciation, etc.).

III. Utilisation du programme

Toute personne qui démarre le programme arrive sur la page des activités proposées par le centre. Chaque activité doit indiquer son nom et avoir la moyenne des notes d'évaluation.

1. Partie administrateur

a) Authentification

L'authentification de l'administrateur doit se faire dans une boîte de dialogue. On doit rentrer le nom d'utilisateur et le mot de passe. Si un champ n'est pas entré, on doit avoir les messages adéquats. Si l'authentification ne fonctionne pas, on doit également avoir le message indiquant que les informations entrées ne correspondent pas.

Une fois l'authentification réussie, l'administrateur peut aller vers les pages du logiciel qui lui sont dédiées.

Il doit pouvoir se déconnecter.

b) Pages de l'administrateur

Votre interface doit avoir les mécanismes nécessaires (pages et/ou boîtes de dialogue) pour l'ajout, la modification et la suppression des **adhérents**, des **activités**, des séances.

c) Statistiques

Vous devez avoir une page permettant d'afficher des statistiques. On doit avoir obligatoirement les statistiques suivantes :

- Nombre total d'adhérents
- Nombre total d'activités
- Nombre d'adhérents pour chacune des activités
- La moyenne des notes d'appréciation pour chacune des activités

Vous devez également ajouter au moins 3 autres statistiques qui incluent les activités, les séances et les participants.

d) Exportation de données

L'administrateur doit pouvoir exporter dans des fichiers CSV :

- La liste des adhérents
- La liste des activités

2. Partie publique

Un adhérent pourra commencer une session en entrant son numéro d'identification. Son nom devra apparaître sur l'interface. Il pourra se déconnecter à n'importe quel moment.

Les parties dédiées à l'administrateur ne sont pas visibles.

Pour participer à une activité, un adhérent pourra cliquer sur celle-ci, choisir la séance qui lui convient.

ATTENTION : un adhérent ne peut pas s'inscrire plus d'une fois à la même activité.

L'adhérent pourra également mettre une note d'évaluation (note sur 5) à une activité à laquelle il participe.

IV. Exigences du programme

Les exigences du travail constituent également les critères de correction. Ils feront donc partie des points pris en compte pour l'attribution de points.

Il est à noter également que le programme ne doit pas planter. Des points seront retirés dans les situations où le programme plante.

1. Expérience utilisateur

Il est de votre ressort que l'interface soit :

- Conviviale
- Facile à utiliser
- Esthétique

La navigation doit se faire facilement et vous devez utiliser les contrôles adéquats.

2. Au niveau du code

Vous devez avoir toutes les requêtes vers la base de données dans un ou plusieurs singletons. Vous pouvez avoir autant de singletons que vous le désirez.

Vous pouvez ajouter toute classe que vous jugerez utile pour le bon fonctionnement et la bonne organisation de votre projet.

Les classes et tous les autres fichiers créés devront avoir des noms significatifs.

3. Sécurité, robustesse et cohésion

Vous devez préparer vos requêtes avant leur exécution.

Les informations visibles ainsi que les opérations effectuées doivent se faire en fonction de la personne authentifiée.

On ne peut pas avoir plus d'une connexion en même temps.

Votre programme ne doit pas planter. Il est de votre responsabilité de mettre en place les mécanismes nécessaires pour ne pas que cela arrive.

Il est de votre responsabilité de faire les validations nécessaires ainsi que d'afficher les messages adéquats en cas d'erreur.

La pertinence des messages (erreur, avertissement, réussite) sera prise en compte dans la note.

4. Travail collaboratif

Vous devez avoir votre travail sur GitHub. Votre projet devra être public et on devra y voir les « commits » de chacun des membres de l'équipe.