

Nom du Groupe
 Groupe : 4 étudiants Maximum (Pas de considération), Date remise: A définir.....
 Pourcentage:40%.....Soutenance :25%..... Date soutenance : A définir

Développer une application de pari sportif en PYTHON, qui permettra aux amateurs de Football de parier sur leurs équipes afin de gagner de l'argent. Il s'agit d'une application Desktop utilisant l'API graphique PYQT (Version 6) et connectée à un SGBDR (Sqlite ou Mysql). La première version de cette application comprendra les 4 modules suivants :

- a) Gestion des comptes
- b) Gestion des Matches
- c) Pariage
- d) Paiement.

Description des Modules :

- a) **Gestion des comptes** : Ce module permet de gérer les comptes des parieurs :
 - Avant de pouvoir choisir des matches à parier, ils doivent d'abord se connecter à leur compte.
 - Un parieur peut voir seulement les informations sur son compte, sur l'historique des matches qu'il a choisi et sur ses paiements.
 - A chaque match parié, le solde du compte du parieur doit diminuer du prix fixé pour le match choisi.

Champs et opérations pour ce module :

Code (unique, à générer),
Nom du parieur (Nom du parieur),
Prénom (Prénom du parieur),
Sexe (Masculin, Féminin),
Date de naissance (Date de naissance),
Téléphone (Téléphone),
NIF/CIN (Nif/Cin),
Nom d'utilisateur (Nom d'utilisateur du parieur),
Mot de passe (Mot de passe du parieur),
Solde (Solde du compte du parieur à zéro et non modifiable lors de la création)
ETAT (A : Actif, S : Supprimer, F: fermé).
 La liste des champs n'est pas exhaustive, vous pouvez en ajouter si nécessaire.

Opérations :

Parieur : Créer compte, visualiser informations, modifier info compte
Administrateur : Créer compte, afficher liste des comptes, recharger compte.



b) **Matches** : Ce module permet de gérer les matches. Il contient les champs et les opérations suivantes :

ID (Code unique),

Type de match (Championnat, coupe du monde, élimatoire, amical)

Pays (Pays où joue le match)

Date match (Date du match)

Heure match (Heure du match)

Equipe receveuse (Equipe receveuse)

Equipe visiteuse (Equipe visiteuse) ;

Cote (Cote)

Score final (0 :0 au montant de la saisie et état="N") ;

Etat (N : Non Encore Joué, E: En cours, T : Terminé, A: Annulé, S :Supprimé),

..... La liste des champs n'est pas exhaustive, vous pouvez en ajouter si nécessaire.

Opération : Enregistrer, Afficher, Modifier (Choisir T si le match est terminé puis ajoute le score final), Imprimer.

c) **Pariage** : Ce module permet aux parieurs de choisir leurs matches.

Il contient les champs et les opérations suivantes :

ID (Numéro unique auto incrémenté),

ID Compte (Numéro du compte),

ID Match (Numéro du match),

Date pariage (Date pariage),

Montant parié (montant parié),

Score prévu (Score prévu par le parieur),

..... La liste des champs n'est pas exhaustive, vous pouvez en ajouter si nécessaire.

Opérations : Enregistrer, Afficher, liste des paris gagnés, imprimer.

NB : Un parieur peut parier seulement sur un match dont son ETAT est égal à N

d) **Paielements** : Ce module permet de payer les gagnants des matches pariés.

Il contient les champs et les opérations suivantes :

ID (Numéro unique auto incrémenté),

ID Pariage (Numéro du pariage)

Date paiement (Date du paiement),

..... La liste des champs n'est pas exhaustive, vous pouvez en ajouter si nécessaire.

Opérations : Afficher liste des paiements, imprimer une fiche de paiement, imprimer la liste des paiements.

Travail Conceptuel :

Conditions minimales à satisfaire :

- 1) Avant de pouvoir accéder au Système, chaque utilisateur doit s'authentifier.
- 2) Chaque Module du Projet doit correspondre à une ou plusieurs tables de la base de données et doit utiliser une ou plusieurs classes python.
- 3) Le montant total des paris d'un utilisateur pour une journée ne doit pas dépasser 75000 gourdes.



Travail à remettre :

- 1) Le code source complet du programme.
- 2) Le Script complet de la base de données.
- 3) Un fichier texte (.txt) contenant les différents membres du groupe et les paramètres d'accès au serveur de base de données et de l'application.

Projet complet à déposer au secrétariat de l'INUKA sur une seule clé USB pour la salle sur le format : nomgroupe_vacation.zip à la date <<à définir>>.

Bon travail, Que Dieu vous bénisse !

