Interface à la base de données

L'interface à la base de données se compose de trois parties : une partie serveur et deux partie client, l'une en java pour android, l'autre en javaScript pour la page web.

La partie serveur se fait via des scripts PHP. Il existe deux scripts PHP, le premier pour écrire dans les tables de la BD est le plus souvent adressée par la page web, le second script sert à la lecture et est, pour l'heure, principale utilisé par le client android. Toutes les requêtes au script PHP se reçoivent une structure JSON en réponse.

Les partie clients Android se fait via une série de classes DbAccess. Ces classes supportent 2 type de bases de données : une locale et une distante. Le module d'accès distant envois les requêtes aux scripts PHP et décodent les réponses retournées pour encapsuler, s'il y a lieu, le résultat dans des classes facile à manipuler par l'interface usager. Le module d'accès local, gère, sous SQLite, une image qui est un sous ensemble de la base de donnés distante.

# PHP

Toutes les requêtes aux scripts PHP reçoivent une réponse en format JSON du type :

{«Status» : «Success/Fail», [data]}

Si le «Status» est «Fail», data sera un message d'erreur de la forme :

«errMsg» : «message identifiant l'erreur»

autrement data variera selon le type de requête effectuée.

Il existe 4 scripts PHP :

|  |  |  |
| --- | --- | --- |
| index.pĥp | - | Sert principalement à permettre au client de valider la présence du serveur, prévient une forme superficielle de hackage. |
| connexion.php | - | Établi la connexion avec la base de donnée.  Possède les fonctions communes à tout les accès à la base de données |
| Writer.php | - | Script permettant d'écrire dans la base de données. |
| Reader.php | - | Script de lecture de la base de données. |

Les deux dernier fichier sont ceux actuellement adressé par les diverse applications client. Le script «Writer.php» est le plus simple. Les routines ont une correspondance 1 pour 1 avec les tables de la base de données. Ses routines sont :

newLigue – Permet Permet d'inscrire une nouvelle ligue.

Paramètres :

- Nom\_Ligue – le nom de la ligue, type String

- ID\_Personne – identifiant du gestionnaire de la ligue, type Int.

Si «Status» est «Success», retourne le nouvel identifiant de la ligue sous la forme :

«Id» : «identifiant»

Set identifiant sera désigné ID\_Ligue

newPersonne – Permet d'inscrire une nouvelle fiche Personne.

Paramètres :

- Nom – Nom de famille de la personne, type String

- prenom – Prénom de la personne, type String

- Date\_Naissance – Nom de famille de la personne, type Date (facultatif)

- telephone – numéro de téléphone, type String. (facultatif)

- courriel – adresse courriel de la personne, type String. (facultatif)

- adresse – son adresse, type String. (facultatif)

Seul les nom et prénom doivent absolument être présent pour créer l'entrée dans la table.

Si «Status» est «Success», retourne le nouvel identifiant de la personne sous la forme :

«Id» : «identifiant»

Set identifiant sera désigné ID\_Personne

newEquipe – Enregistre une nouvelle équipe, liée à un ligue et une sous-ligue.

Paramètres :

- Nom\_Equipe – Nom de l'équipe , type String

- ID\_Ligue – identifiant de la ligue à laquelle appartient l'équipe, type Int

- ID\_SousLigue – identifiant de la sous-ligue à laquelle appartient l'équipe, type Int (facultatif)

Si «Status» est «Success», retourne le nouvel identifiant de l'équipe sous la forme :

«Id» : «identifiant»

Set identifiant sera désigné ID\_Equipe

newAlignement – Inscrit une nouvelle entrée dans la table des Alignements

Paramètres :

- Nom – Nom de famille de la personne, type String

- prenom – Prénom de la personne, type String

- Date\_Naissance – Nom de famille de la personne, type Date. (facultatif)

- telephone – numéro de téléphone, type String. (facultatif)

- courriel – adresse courriel de la personne, type String. (facultatif)

- adresse – son adresse, type String. (facultatif)

Seul les nom et prénom doivent absolument être présent pour créer l'entrée dans la table.

Si «Status» est «Success», retourne le nouvel identifiant de la ligue sous la forme :

«Id» : «identifiant»

Set identifiant sera désigné ID\_Alignement

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | |
|  |  |
|  |  |  | |
|  |  |
| newEquipe |  | Enregistre une nouvelle équipe, liée à un ligue et une sous-ligue | |
|  | NomEquipe – nom de l'équipe  idLigue – id de la ligue auquel l'équipe appartient  idSousLigue – id de la sous ligue auquel l'équipe appartient. (facultatif) |
| newAlignement |  | Inscrit une nouvelle entrée dans la table des Alignements | |
|  | idJoueur  idEquipe  idSaison  position  numeroChandail  temporaire |
| newSaison |  | Enregistre une nouvelle saison | |
|  | idLigue  dateDebut  dateFin  numeroSaison |
| newPartie |  | Inscrit une nouvelle Partie dans la table. | |
|  | lieu  duree  date  idEquipe1  idEquipe2  pointage1 (facultatif)  pointage2 (facultatif) |

Le script «Reader.php» plus complexe car il doit combiner les information de plusieurs tables. Ces routines sont :

validateLogin – Valide le nom d'usager et le mot de passe.

Paramètres :

userName – nom d'usager

passsWord – mot de passe e l'usager

retourne : un objet contenant

- le status (Success/Fail)

- un objet contenant l'id, nom et prénom de l'usager

- Un objet contenant une liste de Compétences

listeEquipe – fait la liste des équipes d'une ligue

Paramètres :

idLigue – Id de la ligue

retourne : un objet contenant

- le status (Success/Fail)

- Un objet contenant une liste d'équipes

listeJoueur – Fait la liste des joueurs d'une équipe

Paramètres :

idEquipe – Id de l'équipe

retourne : un objet contenant

- le status (Success/Fail)

- Un objet contenant une liste de joueur

listeJoueurLigue – Fait la liste des joueurs d'une ligue

Paramètres :

idLigue – Id de la ligue

retourne : un objet contenant

- le status (Success/Fail)

- Un objet contenant une liste de joueur

listeLigue – Fait la liste des ligues d'un gestionnaire

Paramètres :

idGestionnaire – Id du gestionnaire

retourne : un objet contenant

- le status (Success/Fail)

- Un objet contenant une liste de ligues

listeLigueParMarqueur – Fait la liste des ligues pour lesquels un marqueur est autorisé

Paramètres :

idMarqueur – Id du marqueur

retourne : un objet contenant

- le status (Success/Fail)

- Un objet contenant une liste de ligues

listeGestionnaires

Paramètres :

aucun

retourne : un objet contenant

- le status (Success/Fail)

- Un objet contenant une liste de gestionnaires

# Android