

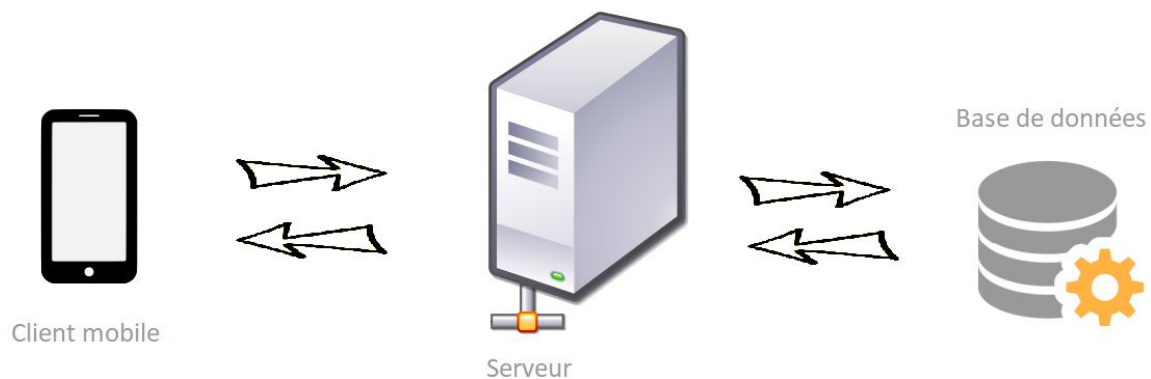
Olympics

Tracking sportif | Modélisation

Bourgeois Guillaume | Hubert Julien | Taoukilite Ahmed | Bellamy Lola

ARCHITECTURE GENERALE

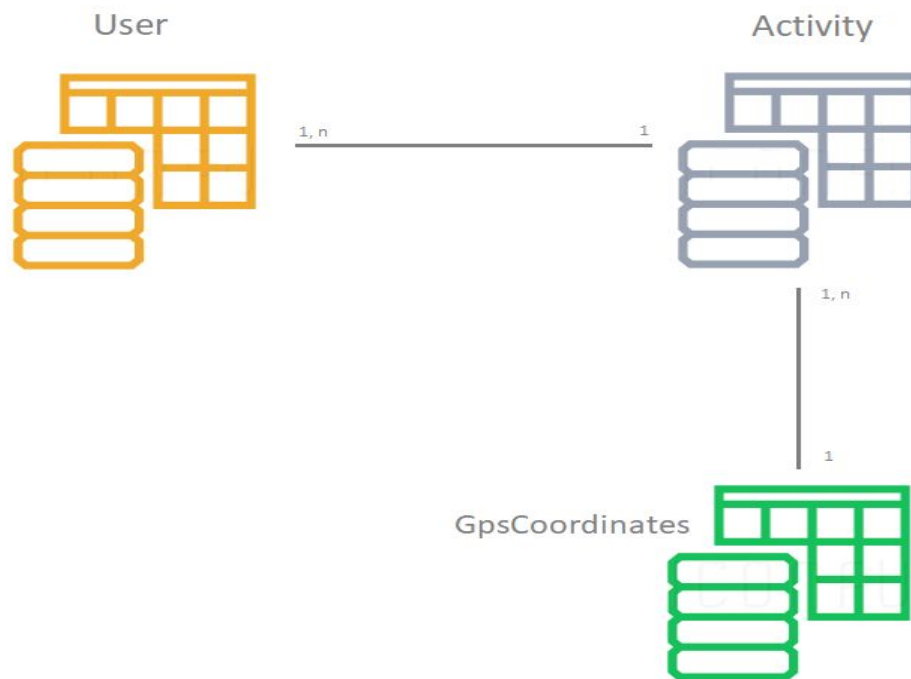
L'application Olympics est composée de trois structures distinctes communiquant entre elles de la façon suivante :



Le client mobile se charge de la collecte des données utiles pour l'application ainsi que de l'affichage de l'interface utilisateur. Ces informations sont envoyées au serveur qui les traite et les stocke dans la base de données, permettant ainsi la conservation d'un historique et la mise en place de statistiques pour l'utilisateur.

DIAGRAMME UML

Les tables User et Activity permettent la conservation des données nécessaires au fonctionnement de l'application. Elles sont liées de la façon suivante :



La table User dispose des champs suivants :

- idUser : Identifiant de l'utilisateur
- username : Nom d'utilisateur
- password : Mot de passe de connexion
- email : Email de connexion
- age : Age de l'utilisateur

La table Activity dispose des champs suivants :

- idActivity : Identifiant de l'activité
- type : Sport lié à l'activité
- startDate : Date de début d'activité
- endDate : Date de fin d'activité
- gpsCoords : Collection d'objet de type GpsCoordinates
- distance : Distance parcourue pendant l'activité (en mètres)

La table GpsCoordinates dispose des champs suivants :

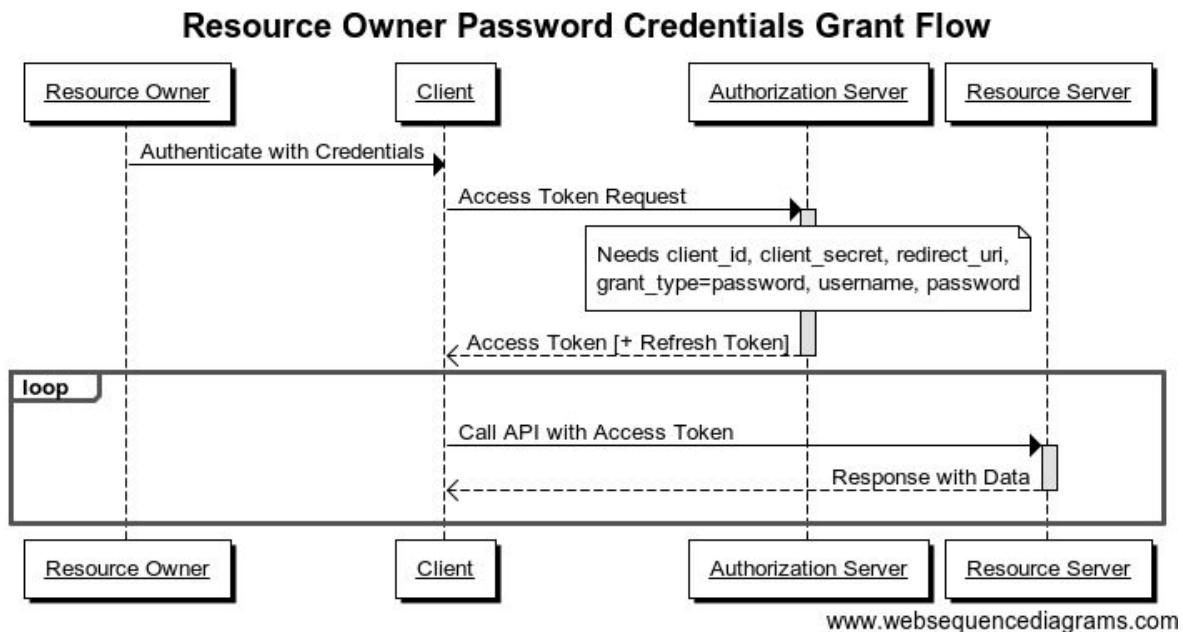
- id : Identifiant des coordonnées GPS
- activity : l'activité durant laquelle sont relevées les coordonnées GPS
- lat : la latitude
- lng : la longitude
- timestamp : date à laquelle sont relevées les coordonnées

Un utilisateur est relié à 1 jusque n activité(s). Une activité n'appartient qu'à un seul utilisateur et dispose d'un ensemble de coordonnées GPS qui n'appartiennent qu'à cette activité.

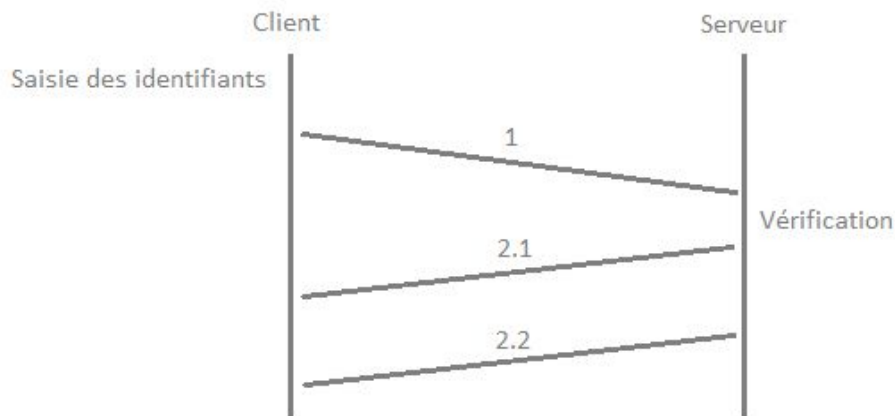
ECHANGES DE MESSAGES

Divers messages sont échangés entre le client et le serveur, pouvant être représentés à l'aide de diagrammes. La totalité des échanges se fait selon le même principe : une requête au serveur donnant lieu à une réponse adéquat ou à une erreur. Ces messages ne seront traités dans ce compte rendu que dans le premier point puis seulement si leur contenu est intéressant et pertinent.

1. **Authentication** : L'utilisation du framework OAuth2 permet la mise en place d'une authentification dont le fonctionnement est décrit par le diagramme officiel suivant:



De manière simplifiée, l'échange de messages est le suivant :



Le message 1 contient les identifiants de connexion de l'utilisateur, soit son email et mot de passe. Ce dernier est encrypté avec bcrypt pour plus de sécurité.

Le message 2.1 représente la réussite de la connexion. Il s'agit d'un token qui sera utilisé pour les requêtes suivantes dans le but de vérifier la provenance des données reçues.

Le message 2.2 est un message d'erreur indiquant à l'utilisateur qu'au moins l'un de ses identifiant est faux, le couple (identifiant, mot de passe) soumis n'étant associé à aucune entrée en base.

2. **Inscription** : La création de compte se fait par le biais d'une requête createUser contenant un identifiant, un mot de passe encrypté avec bcrypt et une adresse email. Une fois le nouvel utilisateur créé, une requête d'authentification comme vue ci-dessus est automatiquement générée, permettant ainsi l'accès immédiat au contenu de l'application.
3. **Déconnexion** : La déconnexion d'un utilisateur se traduit par la révocation de ses jetons d'accès et de rafraîchissement, empêchant l'utilisateur d'effectuer d'avantages de requêtes vers le serveur.

4. **Profil** : La gestion du profil se fait par le biais de deux requêtes permettant la consultation et la modification des données enregistrées sur l'application. La consultation du profil ne nécessite que l'identifiant de l'utilisateur courant pour l'affichage de ses données dans un formulaire. Depuis ce même formulaire, l'utilisateur peut modifier les entrées et ainsi mettre à jour son profil. Le message envoyé au serveur contient alors une structure similaire à celle d'un utilisateur.
5. **Statistique** : A l'aide de son id, un utilisateur peut consulter ses statistiques sur une période souhaitée. Pour un message /user/id/stats envoyé au serveur, la réponse contient un objet en plusieurs dimensions formaté comme suit :

```
1  "general": {
2    "distance": {
3      "average": 1218,
4      "longest": 7654
5    },
6    "duration": {
7      "average": 10040,
8      "longest": 17572,
9      "shortest": 0
10   },
11   "proportions": {
12     "ski": 28.571428,
13     "run": 57.142857,
14     "cycle": 7.142857,
15     "ride": 0,
16     "walk": 7.142857
17   }
18 },
19 "monthly": {
20   0: {
21     "month": 2,
22     "year": 2018,
23     "distance": {
24       "average": 1218,
25       "longest": 7654
26     },
27     "duration": {
28       "average": 10040,
29       "longest": 17572,
30       "shortest": 0
31     },
32     "proportions": {
33       "ski": 28.571428,
34       "run": 57.142857,
35       "cycle": 7.142857,
36       "ride": 0,
37       "walk": 7.142857
38     }
39   },

```

L'objet partiel présenté ci-contre est divisé en deux parties :

- general : les données sur les distances et durées des activités ainsi que les proportions d'activités correspondant à chaque sport.
- monthly : pour chaque mois de l'échantillon demandé, les mêmes valeurs, permettant ainsi la mise en place de graphiques de suivi mensuel pour une meilleure visualisation de la progression des performances du sportif.

6. **Enregistrement d'une nouvelle activité** : Au démarrage d'un timer, une première requête crée une activité sur le serveur. Au cours de l'effort une mise à jour de la position GPS est envoyée toutes les 3 secondes. Chacune de ces mise à jour entraîne une requête au serveur créant une nouvelle entrée dans la table GPSCoordinates. Une fois sa séance terminée, le sportif arrête le timer et émet ainsi une requête permettant de clore l'activité en cours.
7. **Historique des activités** : Une requête visant à récupérer l'historique d'une personne contient l'identifiant de l'utilisateur en question ainsi que le sport dont on veut retrouver les activités. A partir de cette liste, une nouvelle requête peut être émise afin d'obtenir plus de détail sur activité spécifique. Cette requête contiendra alors l'identifiant de l'activité visée.