

Documentation Api Starbill Mobile

➤ Connexion à l'application mobile

L'utilisateur entre son email et son mot de pass

L'url: login

Method : POST

Champ à envoyer : email et password

Si l'authentification est valide une réponse Json est renvoyé, elle comporte trois variables :

- ◆ success : qui peut prendre true ou false (true dans le cas ou c'est validé et vice vers).
- ◆ User : Un tableau contenant tous les informations de l'utilisateur connecté
- ◆ token : qui sert de clé pour l'authentification.

➤ S'enregistrer sur l'application mobile

L'utilisateur entre son nom, prénom, son numéro de téléphone, son adresse, son email et son mot de pass et le mot de pass de confirmation.

L'url: register

Method : POST

Champ à envoyer : nom, prenom, telephone, adresse, email, password

Si l'insertion a été valide une réponse Json est renvoyé, elle comporte deux variables :

- ◆ success : qui peut prendre true ou false (true dans le cas ou c'est validé et vice vers).
- ◆ User : Un tableau contenant tous les informations de l'utilisateur connecté.

➤ Se deconnecter à l'application mobile

L'utilisateur clique sur le bouton déconnecté

L'url: logout?token

Method : GET

Champ à envoyer: aucun champ à envoyer

Cet l'url doit etre suivit du token de l'utilisateur connecté

Ex : logout?token=nnksnkdnlknjkjkklasnbklbdjkjbsaklbnklasklbnls

Le token sera récupéré lors de la connexion de l'utilisateur

➤ La liste des voyages de la semaines

Cet url renvoie tous les voyages de la semaine

L'url: voyages_all

Method : GET

Champ à envoyer: aucun champ à envoyer

Si la connexion à la base de données été valide une réponse Json est renvoyé, elle comporte deux variables :

- ◆ success : qui peut prendre true ou false (true dans le cas ou c'est validé et vice vers).

- ◆ data: Un tableau contenant tous les voyages de la semaine, également toutes les informations d'une station de départ (id_station,ville_station,adresse,latitude_station,longitude_station,agence_id,ville_id) et toutes les informations d'une station de d'arrivée (id_station,ville_station,adresse,latitude_station,longitude_station,agence_id,ville_id)

➤ Liste des réservations des voyages

Cet url renvoie toutes les réservations faite par un utilisateur

L'url: reservation_all

Method : GET

Champ à envoyer: token

Si l'utilisateur est connecté et que le token valide une réponse Json est renvoyé, elle comporte deux variables :

- ◆ success :qui peut prendre true ou false (true dans le cas ou c'est validé et vice vers).
- ◆ data: Un tableau contenant toutes les réservation faites par l'utilisateur connecté. (id_reservation,etat_reservation,nbre_place_reservation,prix_unitaire,nbre_place_utilise,code,type_paiement,voyage_id,paiement_id)

➤ Les tickets

L'url: tickets_all

Method : GET

Champ à envoyer: id & token

L'utilisateur connecté.Pour avoir les les tickets générés par l'utilisateur il faut envoyé l'id de réservation et le token de l'utilisateur connecté.

Cet url renvoie une response Json qui comporte deux variables :

- ◆ Success : qui peut prendre true ou false (true dans le cas ou c'est validé et vice vers)
- ◆ data : Un tableau contenant toutes les tickets faites par cet utilisateur connecté.

➤ Rechercher les voyages

Renvoie les voyages trouvés

L'url: rechercher_voyage

Method : GET

Champ à envoyer: ville_depart, ville_arrivee, nombre_place et la date

Si la recherche est trouvé, cet url renvoie une réponse Json comportant deux variables :

- ◆ Success : qui peut prendre true ou false (true dans le cas ou c'est validé et vice vers)
- ◆ data : Un tableau contenant toutes les concernant la recherche des voyages