

Documentation API Projet Final B2

Cette api vous permettra d'interagir avec notre site bibliothèque via différentes requêtes tel que la création de votre compte personnel, la modification de vos collections ou encore la recherche de livre ou d'auteur .

Cette api utilise l'authentification avec un bearer token que vous pouvez récupérer après avoir créé votre compte. Vous trouverez devant chaque url la méthode à utiliser pour la requête. Vous trouverez aussi le mot 'AUTH' qui indique que vous devrez utiliser votre token d'identification pour faire cette requête. Certaines requêtes ou certains paramètres sont réservés aux administrateurs du site, ils auront alors l'inscription « (Admin) » à côté d'eux. Tous les paramètres que vous renseignerez devront être de type string. Certains paramètres auront la mention « (optio) », pour optionnel, ce qui signifie qu'ils ne sont obligatoires pour que la requête fonctionne. Le format des réponses est en JSON.

Nous vous conseillons fortement de tester vos requêtes via des applications avant de les implémenter dans votre code.

Si « localhost » ne fonctionne pas, vous pouvez le remplacer par l'adresse ip du server : 172.16.238.11

Page d'accueil :

<http://localhost:8080/>

Aucun paramètre n'est demandé pour cette requête.

Vous recevrez une chaîne de caractère en guise de réponse avec des informations sur le projet.

Création de votre compte :

POST <http://localhost:8080/user/create>

Paramètres :

Champs	Description
email	Votre email
mdp	Votre mot de passe
photo	Votre photo de profil (Ce champs peut rester vide)
nom	Votre nom d'utilisateur

Vous recevrez une chaîne de caractère indiquant que la création de votre compte a bien été effectuée.

Récupération de votre Token d'authentification :

GET http://localhost:8080/get_token

Paramètres :

Champs	Description
email	Votre email
mdp	Votre mot de passe

Vous recevrez une chaîne de caractère formater. Vous pourrez la transformer en JSON.

Récupération des utilisateurs :

GET <http://localhost:8080/user>

Paramètres :

Vous pouvez utiliser cette route sans utiliser de paramètre. Dans ce cas vous récupérerez la liste de tous les utilisateurs. Sinon, vous pouvez spécifier votre recherche en utilisant les paramètres ci-dessous.

ATTENTION : un seul paramètre à la fois est autorisé.

Champs	Description
user_name (optio)	Le nom ou une partie du nom de l'utilisateur désiré.
user_id (optio)	L'identifiant de l'utilisateur désiré.
aleatoire (optio)	Récupération aléatoire d'utilisateur. Vous devez spécifier le nombre d'utilisateur que vous voulez. Ex : aleatoire=30 pour récupérer 30 utilisateurs.

Vous recevrez une chaîne de caractère formater. Vous pourrez la transformer en JSON.

Récupération des auteurs :

GET <http://localhost:8080/auteur>

Paramètres :

Vous pouvez utiliser cette route sans utiliser de paramètre. Dans ce cas vous récupérerez la liste de tous les auteurs. Sinon, vous pouvez spécifier votre recherche en utilisant les paramètres ci-dessous.

ATTENTION : un seul paramètre à la fois est autorisé.

Champs	Description
auteur_name (optio)	Le nom ou une partie du nom de l'auteur désiré.
auteur_id (optio)	L'identifiant de l'auteur désiré.
aleatoire (optio)	Récupération aléatoire d'auteur. Vous devez spécifier le nombre d'auteur que vous voulez. Ex : aleatoire=50 pour récupérer 50 auteur.

Vous recevrez une chaîne de caractères formater. Vous pourrez la transformer en JSON.

Récupération des catégories :

GET <http://localhost:8080/categorie>

Paramètres :

Vous pouvez utiliser cette route sans utiliser de paramètre. Dans ce cas vous récupérerez la liste de toutes les catégories. Sinon, vous pouvez spécifier votre recherche en utilisant les paramètres ci-dessous.

ATTENTION : un seul paramètre à la fois est autorisé.

Champs	Description
categorie_name (optio)	Le nom ou une partie du nom de la catégorie désiré.
categorie_id (optio)	L'identifiant de la catégorie désiré.

Vous recevrez une chaîne de caractères formater. Vous pourrez la transformer en JSON.

Récupération des collections :

GET AUTH <http://localhost:8080/collection>

Paramètres :

Vous pouvez utiliser cette route sans utiliser de paramètre. Dans ce cas vous récupérerez la liste de toutes les collections. Sinon, vous pouvez spécifier votre recherche en utilisant les paramètres ci-dessous.

ATTENTION : un seul paramètre à la fois est autorisé.

Champs	Description
collection_id (optio)	L'identifiant de la collection désiré.

Vous recevrez une chaîne de caractères formater. Vous pourrez la transformer en JSON.

Récupération des commentaires:

GET <http://localhost:8080/commentaire>

Paramètres :

Vous pouvez utiliser cette route sans utiliser de paramètre. Dans ce cas vous récupérerez la liste de tous les commentaires. Sinon, vous pouvez spécifiez votre recherche en utilisant les paramètres ci-dessous.

ATTENTION : un seul paramètre à la fois est autorisé.

Champs	Description
commentaire_id (optio)	L'identifiant du commentaire désiré.
livre_id (optio)	L'identifiant du livre dont on souhaite récupérer les commentaires.

Vous recevrez une chaine de caractère formater. Vous pourrez la transformer en JSON.

Récupération des livres :

GET <http://localhost:8080/livre>

Paramètres :

Vous pouvez utiliser cette route sans utiliser de paramètre. Dans ce cas vous récupérerez la liste de tous les livres. Sinon, vous pouvez spécifiez votre recherche en utilisant les paramètres ci-dessous.

ATTENTION : un seul paramètre à la fois est autorisé.

Champs	Description
livre_name (optio)	Le nom ou une partie du nom du livre désiré.
livre_id (optio)	L'identifiant du livre désiré.
auteur_id (optio)	Identifiant de l'auteur des livres voulu.
aleatoire (optio)	Récupération aléatoire de livre. Vous devez spécifiez le nombre de livre que vous voulez. Ex : aleatoire=70 pour récupérer 70 livres.

Vous recevrez une chaine de caractère formater. Vous pourrez la transformer en JSON.

Vérification de votre email :

GET http://localhost:8080/verif_email

Cette route permet de vérifier si l'adresse mail que vous voulez utiliser est déjà prise par un autre utilisateur.

Paramètres :

:

Champs	Description
email	L'email que vous voulez vérifier

Vous recevrez une chaîne de caractère formater ou il y aura le nombre de personne avec cette adresse email (0 ou 1). Vous pourrez la transformer en JSON.

Création d'un auteur (Admin) :

POST AUTH <http://localhost:8080/auteur/create>

Paramètres :

Champs	Description
nom	Nom de l'auteur à créer.
desc	Description de l'auteur à créer.
photo	Photo de l'auteur à créer (ce paramètre peut être vide)

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Suppression d'un auteur (Admin) :

POST AUTH <http://localhost:8080/auteur/delete>

Paramètres :

Champs	Description
auteur_id	L'identifiant de l'auteur désiré.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Création d'une catégorie (Admin) :

POST AUTH <http://localhost:8080/categorie/create>

Paramètres :

Champs	Description
nom	Nom de la catégorie à créer.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Suppression d'une catégorie (Admin) :

POST AUTH <http://localhost:8080/categorie/delete>

Paramètres :

Champs	Description
categorie_id	Identifiant de la catégorie à supprimer.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Suppression d'un utilisateur :

POST AUTH <http://localhost:8080/user/delete>

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir supprimer votre compte.

Paramètres :

Champs	Description
user_id	Identifiant de l'utilisateur à supprimer.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Changer les informations d'un utilisateur :

POST AUTH http://localhost:8080/user/change_info

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir modifier les informations de votre compte. De plus cette requête doit contenir au moins un paramètre.

Paramètres :

Champs	Description
email (optio)	Nouvel email de l'utilisateur.
mdp (optio)	Nouveau mot de passe de l'utilisateur.
photo (optio)	Nouvelle photo de l'utilisateur.
nom (optio)	Nouveau nom de l'utilisateur.
is_admin (optio) (Admin)	Nouvelles permissions de l'utilisateur.
user_id (Admin)	Identifiant de l'utilisateur dont on souhaite changer les informations.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Suivre un utilisateur :

POST AUTH http://localhost:8080/user/follow_user

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir ajouter un utilisateur à suivre avec votre compte.

Paramètres :

Champs	Description
follow_user_id	Identifiant de l'utilisateur à suivre.
user_id (Admin)	Identifiant de l'utilisateur à qui ont veux rajouter un utilisateur à suivre.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Ne plus suivre un utilisateur :

POST AUTH http://localhost:8080/user/unfollow_user

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir supprimer un utilisateur à suivre avec votre compte.

Paramètres :

Champs	Description
unfollow_user_id	Identifiant de l'utilisateur à suivre.
user_id (Admin)	Identifiant de l'utilisateur à qui ont veux rajouter un utilisateur à suivre.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Suivre un auteur :

POST AUTH http://localhost:8080/user/follow_auteur

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir ajouter un auteur à suivre avec votre compte.

Paramètres :

Champs	Description
follow_auteur_id	Identifiant de l'auteur à suivre.
user_id (Admin)	Identifiant de l'utilisateur à qui ont veux rajouter un auteur à suivre.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Ne plus suivre un auteur :

POST AUTH http://localhost:8080/user/unfollow_auteur

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir supprimer un auteur à suivre avec votre compte.

Paramètres :

Champs	Description
unfollow_auteur_id	Identifiant de l'auteur à suivre.
user_id (Admin)	Identifiant de l'utilisateur à qui ont veux rajouter un auteur à suivre.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Créer une collection :

POST AUTH <http://localhost:8080/collection/create>

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir créer une collection avec votre compte.

Paramètres :

Champs	Description
nom	Nom de la collection
is_private	Est-ce que vous voulez que cette collection soit privée ou non (Boolean)
user_id (Admin)	Identifiant de l'utilisateur à qui ont veux rajouter une collection.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Supprimer une collection :

POST AUTH <http://localhost:8080/collection/delete>

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir supprimer une collection avec votre compte.

Paramètres :

Champs	Description
collection_id	Identifiant de la collection que l'on veut supprimer.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.

Ajouter un livre à une collection :

POST AUTH http://localhost:8080/collection/add_livre

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir ajouter un livre à une de vos collections avec votre compte.

Paramètres :

Champs	Description
collection_id	Identifiant de la collection ou l'on veut ajouter un livre
livre_id	Identifiant du livre que l'on veut rajouter à la collection.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Supprimer un livre d'une collection :

POST AUTH http://localhost:8080/collection/delete_livre

ATTENTION : Vous et les administrateurs êtes les seuls à pouvoir supprimer un livre à une de vos collections avec votre compte.

Paramètres :

Champs	Description
collection_id	Identifiant de la collection ou l'on veut supprimer un livre
livre_id	Identifiant du livre que l'on veut supprimer de la collection.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Ajouter un commentaire :

POST AUTH <http://localhost:8080/commentaire/add>

Paramètres :

Champs	Description
livre_id	Identifiant du livre que l'on veut commenter.
text_commentaire	Contenu du commentaire.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Supprimer un commentaire (Admin) :

POST AUTH <http://localhost:8080/commentaire/delete>

Paramètres :

Champs	Description
commentaire_id	Identifiant du commentaire que l'on veut supprimer.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Ajouter un livre (Admin) :

POST AUTH <http://localhost:8080/livre/add>

Paramètres :

Champs	Description
nom	Nom du livre que l'on veut ajouter.
description	Description du livre que l'on veut ajouter.
photo	Photo du livre que l'on veut ajouter.
isbn	ISBN du livre que l'on veut ajouter.
editeur	Editeur du livre que l'on veut ajouter.
prix	Prix du livre que l'on veut ajouter.
auteur_id	Identifiant de l'auteur du livre que l'on veut ajouter.
categorie_id	Identifiant de la catégorie à laquelle ce livre appartient.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Supprimer un livre (Admin) :

POST AUTH <http://localhost:8080/livre/delete>

Paramètres :

Champs	Description
livre_id	Identifiant du livre que l'on veut supprimer.

Vous recevrez une chaîne de caractère confirmant la réussite de votre requête.

Changer les informations d'un livre (Admin) :

POST AUTH http://localhost:8080/livre/change_info

Paramètres :

Champs	Description
livre_id	Identifiant du livre dont on veut changer les informations.
nom (optio)	Nouveau nom du livre.
description (optio)	Nouvelle description du livre.
photo (optio)	Nouvelle photo du livre.
isbn (optio)	Nouveau ISBN du livre.
editeur (optio)	Nouvel éditeur du livre.
prix (optio)	Nouveau prix du livre.
auteur_id (optio)	Nouvel identifiant de l'auteur du livre.
categorie_id (optio)	Identifiant de la nouvelle catégorie à laquelle ce livre appartient.

Vous recevrez une chaine de caractère confirmant la réussite de votre requête.