**API Specification Doc**

**(*456enBinaire*)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 1.0 | 14 mars 2024 | Les aigles | Initial pour activités |
|  |  |  |  |

Index

[Méthodes](#_Toc162121393)

[Création d’une activité](#_Toc162121394)

[Modification d’une activité](#_Toc162121395)

[Suppression d’une activité](#_Toc162121396)

[Sélection d’une activité](#_Toc162121397)

[Sélection de toutes les activités](#_Toc162121398)

[Sélection de toutes les activités d’un utilisateur](#_Toc162121399)

[Sélection de toutes les activités situées entre deux (2) dates inclusivement](#_Toc162121400)

[Sélection de toutes les activités d’un utilisateur situées entre deux (2) dates inclusivement](#_Toc162121401)

[Glossaire](#_Toc162121402)

[HTTP Status Codes](#_Toc162121403)

[Conventions](#_Toc162121404)

Méthodes

Création d’une activité

Ajoute une activité à la base de données.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /activity/ |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| BODY | activité | Activity |

**activité**

Une instance de la classe *Activity*.

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | La réponse correspond à l’objet *Activity* créé dans la base de données, dont le *id\_activity* créé par la base de données.  Exemple JSON :  {  "id\_activity": 425,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  } |

Modification d’une activité

Modifier une activité dans la base de données.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **PUT** | /activity/ |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| BODY | activité | Activity |

**activité**

Une instance de la classe *Activity* **modifiée**.

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | La réponse correspond à l’objet *Activity* modifié dans la base de données.  **Si l’objet en paramètre n’existe pas dans la base de données, un nouvel objet est créé avec un identifiant automatique.**  Exemple JSON :  {  "id\_activity": 425,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires modifiés"  } |

Suppression d’une activité

Supprimer une activité de la base de données.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **DELETE** | /activity/{id\_activity} |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| URL\_PARAM | <id\_activity> | number |

**id\_activity**

L’identifiant de l’activité à supprimer.

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | L’objet *Activity* sélectionné est supprimé de la base de données. |
| 404 | L’objet *Activity* sélectionné n’existe pas. |

Sélection d’une activité

Sélectionne une activité spécifique.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** | /activity/{id\_activity} |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| URL\_PARAM | <id\_activity> | number |

**id\_activity**

L’identifiant de l’activité que l’on veut sélectionner.

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | La réponse correspond à l’objet *Activity* sélectionné, tel qu’il apparait dans la base de données.  Exemple JSON :  {  "id\_activity": 425,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  } |
| 200 | Si l’objet *Activity* sélectionné n’existe pas, alors la réponse est :  null. |

Sélection de toutes les activités

Sélectionne une activité spécifique.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** | /activity |

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | La réponse correspond à une liste d’objets *Activity*.  Exemple JSON :  [  {  "id\_activity": 425,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  },  {  "id\_activity": 426,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  }  ] |
| 200 | S’il n’y a aucun objet *Activity* dans la base de données, alors la réponse est :  [ ]. |

Sélection de toutes les activités d’un utilisateur

Sélectionne toutes les activités associées à un utilisateur.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** | /activity/user/{id\_user} |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| URL\_PARAM | <id\_user> | number |

**id\_user**

L’identifiant de l’utilisateur dont on veut sélectionner les activités.

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | La réponse correspond à une liste d’objets *Activity* associés à un utilisateur spécifique.  Exemple JSON :  [  {  "id\_activity": 425,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  },  {  "id\_activity": 426,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  }  ] |
| 200 | S’il n’y a aucun objet *Activity* dans la base de données, alors la réponse est :  [ ]. |

Sélection de toutes les activités situées entre deux (2) dates inclusivement

Sélectionne toutes les activités situées dans un intervalle de dates donné.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** | /activity/{start\_date}/{end\_date} |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| URL\_PARAM | <start\_date> | Text |
| URL\_PARAM | <end\_date> | Text |

**start\_date**

La date de début de l’intervalle.

**end\_date**

La date de fin de l’intervalle.

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | La réponse correspond à une liste d’objets *Activity* situés dans l’intervalle de dates.  Exemple JSON :  [  {  "id\_activity": 425,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  },  {  "id\_activity": 426,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  }  ] |
| 200 | S’il n’y a aucun objet *Activity* correspondant à la requête dans la base de données, alors la réponse est :  [ ]. |

Sélection de toutes les activités d’un utilisateur situées entre deux (2) dates inclusivement

Sélectionne toutes les activités associées à un utilisateur dans un intervalle de dates donné.

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** | /activity/{id\_user}/{start\_date}/{end\_date} |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| URL\_PARAM | <id\_user> | Number |
| URL\_PARAM | <start\_date> | Text |
| URL\_PARAM | <end\_date> | Text |

**id\_user**

L’identifiant de l’utilisateur dont on veut sélectionner les activités.

**start\_date**

La date de début de l’intervalle.

**end\_date**

La date de fin de l’intervalle.

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | La réponse correspond à une liste d’objets *Activity* associés à un utilisateur spécifique dans l’intervalle de dates.  Exemple JSON :  [  {  "id\_activity": 425,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  },  {  "id\_activity": 426,  "id\_client": 22,  "type\_activity": "course",  "description\_activity": "course au parc Maisonneuve",  "date": "2024-03-14",  "duration\_min": 60,  "distance\_km": 20,  "heart\_beat\_min": 125.25,  "temperature\_c": 22.5,  "comment": "commentaires"  }  ] |
| 200 | S’il n’y a aucun objet *Activity* correspondant à la requête dans la base de données, alors la réponse est :  [ ]. |

Glossaire

HTTP Status Codes

2XX - Succès

4XX – Erreur du client HTTP

5XX – Erreur du serveur

| **Status Code** | **Description** |
| --- | --- |
| 200 | OK |
| 201 | Created |
| 202 | Accepted (Request accepted, and queued for execution) |
| 400 | Bad request |
| 401 | Authentication failure |
| 403 | Forbidden |
| 404 | Resource not found |
| 405 | Method Not Allowed |
| 409 | Conflict |
| 412 | Precondition Failed |
| 413 | Request Entity Too Large |
| 500 | Internal Server Error |
| 501 | Not Implemented |
| 503 | Service Unavailable |

Conventions

* Client - Application cliente.
* Status - Code de réponse (*HTTP Status Code*).
* Toutes les réponses sont au format JSON.
* Tous les paramètres de *request* sont obligatoires à moins qu’ils ne soient explicitement marqués comme [optionnel].
* Le type de valeurs acceptées pour un paramètre de *request* s’affiche dans la colonne valeurs comme ceci [**10**|<any number>]. Le symbole | signifie OU. Si le paramètre est [optionnel], la valeur par défaut est affichée en gras bleu, comme **10** est écrit dans [**10**|<any number>].