

Table des matières

Endpoint : adr-exist-get	4
Endpoint : adresse-put	6
Endpoint : agent-put	8
Endpoint : auth-OTP.....	10
Endpoint : auth-signin.....	12
Endpoint : calculate-lap-time.....	14
Endpoint : capteur-all-get	16
Endpoint : capteur-delete	18
Endpoint : capteur-ev-get	19
Endpoint : capteur-log	21
Endpoint : capteur-ping	23
Endpoint : capteur-put.....	25
Endpoint : capteur-valeurs-recent-get.....	27
Endpoint : capteurs-valeurs-put.....	29
Endpoint : circuit-adr-circuit-put.....	30
Endpoint : circuit-adr-delete	32
Endpoint : circuit-adr-get	33
Endpoint : circuit-all	34
Endpoint : circuit-coord-circuit-put.....	35
Endpoint : circuit-coord-delete	37
Endpoint : circuit-coord-get-simple.....	38
Endpoint : circuit-coord-get	40
Endpoint : circuit-delete	42
Endpoint : circuit-get	43
Endpoint : circuit-put.....	44
Endpoint : coord-exist-get.....	46
Endpoint : count-unique-epc-get.....	48
Endpoint : ev-cptr-delete	49
Endpoint : ev-cptr-put.....	50
Endpoint : ev-org-get	52
Endpoint : evt-all	54
Endpoint : evt-delete	56
Endpoint : evt-get	57

Endpoint : evt-put	58
Endpoint : grp-all	60
Endpoint : grp-delete	61
Endpoint : grp-ev-get	62
Endpoint : grp-get	64
Endpoint : grp-pilote-get	65
Endpoint : grp-pilote-grp-p-contactinfo-update	66
Endpoint : grp-put.....	67
Endpoint : grp-session-put.....	69
Endpoint : health	70
Endpoint : orga-adr-delete	71
Endpoint : orga-adr-get	72
Endpoint : orga-adr-orga-put	73
Endpoint : orga-all	74
Endpoint : orga-coord-delete	75
Endpoint : orga-coord-get-simple.....	76
Endpoint : orga-coord-get	77
Endpoint : orga-coord-orga-put	78
Endpoint : orga-delete	79
Endpoint : orga-get	80
Endpoint : orga-pilote-get	81
Endpoint : orga-put	82
Endpoint : pilote-actif-update	84
Endpoint : pilote-adr-delete	85
Endpoint : pilote-adr-get	86
Endpoint : pilote-adr-pilote-put.....	87
Endpoint : pilote-adr-put.....	88
Endpoint : pilote-all-get	90
Endpoint : pilote-check-exists-get.....	91
Endpoint : pilote-coord-delete	92
Endpoint : pilote-coord-get-simple.....	93
Endpoint : pilote-coord-get	94
Endpoint : pilote-coord-pilote-put.....	95
Endpoint : pilote-coord-put.....	96
Endpoint : pilote-delete	97
Endpoint : pilote-detail-get	98

Endpoint : pilote-get	99
Endpoint : pilote-group-delete	100
Endpoint : pilote-grp-get	101
Endpoint : pilote-grp-put.....	103
Endpoint : pilote-grp-upsert-put.....	105
Endpoint : pilote-isAdmin	107
Endpoint : pilote-nom-prenom-update	108
Endpoint : pilote-one-adr-delete	109
Endpoint : pilote-one-coord-delete	110
Endpoint : pilote-organisateur-delete.....	111
Endpoint : pilote-organisateur-put	112
Endpoint : pilote-push	113
Endpoint : pilote-put.....	114
Endpoint : pilote-search	116
Endpoint : pilote-update-put	117
Endpoint : pilote-validate-get.....	118
Endpoint : pilote-all-role	119
Endpoint : pilote-all	120
Endpoint : session-ev-delete.....	121
Endpoint : session-get	122
Endpoint : session-nogrp-delete	123
Endpoint : session-put.....	124
Endpoint : session-same-get.....	125
Endpoint : sessions-put	126
Endpoint : ticket-all	127
Endpoint : ticket-delete	128
Endpoint : ticket-get	129
Endpoint : ticket-put.....	130
Endpoint : tours-ev-get	132
Endpoint : tours-get	133
Endpoint : tours-meilleur-moyen-tps-get.....	134
Endpoint : version.....	135

Endpoint : adr-exist-get

Méthode et URL

POST /api/adr-exist

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de vérifier l'existence d'une adresse dans la base de données à partir de ses informations principales (nom, code postal, commune, pays).

Il retourne un indicateur `found` et, si l'adresse existe, les données associées

Requête

Corps de la requête (Body)

```
{
  "adr_nom": "Nom adresse",
  "adr_code_postal": "75001",
  "adr_commune": "Paris",
  "adr_pays": "France"
}
```

Champ	Type	Requis	Nullable	Description
adr_nom	string	Oui	Non	Nom de l'adresse
adr_code_postal	string	Oui	Non	Code postal de l'adresse
adr_commune	string	Oui	Non	Commune de l'adresse
adr_pays	string	Oui	Non	Pays de l'adresse

Réponse

200 OK – Recherche effectuée

Si l'adresse existe :

```
{
  "found": true,
  "data": {
    "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "adr_nom": "Nom adresse",
  }
}
```

```
    "adr_libelle": "Libellé complet",  
    "adr_code_postal": "75001",  
    "adr_commune": "Paris",  
    "adr_pays": "France",  
  }  
}
```

Si l'adresse n'existe pas :

```
{  
  "found": false,  
  "data": ""  
}
```

Endpoint : adresse-put

Méthode et URL

PUT /api/adresse/{id?}

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)
- **Paramètre de route facultatif :** id (GUID)

Description

Cet endpoint permet de créer ou mettre à jour une adresse existante dans la base de données. Si le paramètre de route `id` n'est pas fourni, un nouvel identifiant sera généré automatiquement

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID de l'adresse à créer ou mettre à jour

Corps de la requête (Body)

```
{
  "adr_nom": "Nom adresse",
  "adr_libelle": "Libellé complet",
  "adr_code_postal": "75001",
  "adr_commune": "Paris",
  "adr_pays": "France"
}
```

Champ	Type	Requis	Nullable	Description
adr_nom	string	Oui	Oui	Nom de l'adresse
adr_libelle	string	Oui	Oui	Libellé complet de l'adresse
adr_code_postal	string	Oui	Oui	Code postal de l'adresse
adr_commune	string	Oui	Oui	Commune de l'adresse
adr_pays	string	Oui	Oui	Pays de l'adresse

Réponse

200 OK – Adresse inséré ou mis à jour avec succès

```
{
  "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["adr_nom"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : agent-put

Méthode et URL

PUT /api/agent

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de **créer un pilote (utilisateur)** dans la base de données.
Équivalent à `pilote-put`.

Requête

Corps de la requête (Body)

```
{
  "r_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_prenom": "prenom",
  "p_nom": "url",
  "p_actif": 1
}
```

Champ	Type	Requis	Nullable	Description
r_id	string (uuid)	Non	Non	ID du rôle du pilote
p_id	String (uuid)	Oui	Non	ID du pilote
p_prenom	string	Oui	Oui	Prénom du pilote
p_nom	string	Oui	Oui	Nom du pilote
p_actif	boolean	Non	Oui	État d'activité du pilote

Réponse

200 OK – Pilote insérée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status` : 200.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
```



```
    "code": "invalid_type",
    "expected": "string",
    "received": "number",
    "path": ["p_nom"],
    "message": "Expected string, received number"
  }
]
}
```

Endpoint : auth-OTP

Méthode et URL

POST /api/auth/otp

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'authentifier un pilote via un code OTP (One Time Password) à 6 chiffres.

Si l'OTP et l'identifiant sont valides, un token d'authentification est généré et retourné.

En mode debug (DEBUG_AUTH=true), le token retourné est non-encodé et contient des informations supplémentaires pour le développement.

Requête

Corps de la requête (Body)

```
{
  "id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "otp": 123456
}
```

Champ	Type	Requis	Nullable	Description
id	string (uuid)	Oui	Non	ID du pilote
otp	Number (6)	Oui	Non	Code OTP à 6 chiffres

Réponse

200 OK – Authentification réussie

```
{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..." // JWT ou objet debug
}
```

En mode debug le token est un objet non-encodé :

```
{
  "token": {
    "iss": "grapi",
    "jti": "debug-jti",
  }
}
```

```
    "userId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "roleDroit": 0,
    "iat": "2025-06-18T13:33:34.000Z",
    "debug": true
  }
}
```

400 Bad Request – Données invalides ou OTP invalide

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["id"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : auth-signin

Méthode et URL

POST /api/auth/signin

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de demander l'envoi d'un code OTP (One Time Password) à 6 chiffres pour l'authentification d'un pilote.

Le code OTP est envoyé soit par SMS (si le contacté passé dans le corps est un numéro de téléphone), soit par email (si le contacté passé dans le corps est une adresse email).

La réponse contient l'identifiant du pilote associé.

Requête

Corps de la requête (Body)

```
{
  "value": "+33612345678"
}
```

Ou

```
{
  "value": "pilote@exemple.com"
}
```

Champ	Type	Requis	Nullable	Description
value	string	Oui	Non	Email ou numéro de téléphone du pilote

Réponse

200 OK – OTP envoyé avec succès

```
{
  "id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

400 Bad Request – Données invalides ou OTP invalide

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["value"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : calculate-lap-time

Méthode et URL

GET /api/calculate-lap-time

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de calculer et d'enregistrer les temps de tours à partir des données de capteurs (QR) présentes dans la table CAPTEURS_VALEURS.

Pour chaque capteur QR détecté (champ `cptr_epc_txt` contenant "QR-" et session valide), un tour est créé dans la table TOURS.

Les enregistrements traités sont marqués comme tels dans la table source.

Réponse

200 OK – Traitement effectué

```
{
  "success": true,
  "message": "Lap times calculation completed successfully",
  "data": {
    "processedCount": 12,
    "toursCreated": 8,
    "updatedRows": 8,
    "totalRecords": 20,
    "qrRecords": 10,
    "uniqueEPCs": 3,
    "createdTours": [
      {
        "tour_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF ",
        "epc": "QR-123456",
        "session_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF ",
        "timestamp": "2025-07-26T14:00:00.000Z",
        "value": 123.45
      }
      // ...jusqu'à 10 premiers tours
    ],
    "errors": [
      {
        "record_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF ",
        "error": "Invalid s_id",
        "s_id": null
      }
    ]
  }
}
```

```
        // ...si erreurs rencontrées
    ]
},
"timestamp": "2025-07-26T14:00:00.000Z"
}
```

200 OK – Aucun QR à traiter

```
{
  "success": true,
  "message": "No QR records to process",
  "data": {
    "processedCount": 0,
    "toursCreated": 0,
    "totalRecords": 20,
    "qrRecords": 0
  }
}
```

500 Internal Server Error – Erreur serveur ou SQL

```
{
  "success": false,
  "message": "Failed to connect to SQL database",
  "error": "Détail de l'erreur"
}
```

Endpoint : capteur-all-get

Méthode et URL

GET /api/capteur/all/{id?}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)
- **Paramètre de route facultatif :** id (GUID)

Description

Cet endpoint permet de récupérer la liste des capteurs enregistrés dans la base de données. Si le paramètre d'URL id est fourni, seuls les capteurs liés à l'organisation correspondante sont retournés.

Si aucun identifiant n'est fourni, tous les capteurs sont retournés

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID de à laquelle appartient les capteurs

Réponse

200 OK – Capteurs trouvés

```
{
  "data": [
    {
      "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF ",
      "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF ",
      "cptr_nom": "Nom du capteur",
      "cptr_origine": "Origine"
    },
    ...
  ],
  "count": 3,
  "filter": { "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF" }
}
```

Ou (si aucun id fourni)

```
{
```



```
"data": [  
  ...  
],  
"count": 10,  
"filter": { "all": true }  
}
```

200 OK – Aucun capteur trouvé

```
{  
  "data": [],  
  "count": 0,  
  "message": "Aucun capteur trouvé pour cette organisation"  
}
```

400 Bad Request – Données invalides

```
{  
  "error": "Identifiant d'organisation invalide",  
  "details": {  
    "id": {  
      "_errors": [  
        "L'identifiant de l'organisation doit être un UUID valide"  
      ]  
    }  
  }  
}
```

500 Internal Server Error – Erreur serveur

```
{  
  "error": "Erreur serveur",  
  "message": "Une erreur inattendue s'est produite"  
}
```

Endpoint : capteur-delete

Méthode et URL

DELETE /api/capteur/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer un capteur spécifique de la base de données en utilisant son id (cptr_id).

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID du capteur à supprimer

Réponse

204 No content – Capteur supprimée

Le status 204 est retourné dans tous les cas.

Endpoint : capteur-ev-get

Méthode et URL

POST /api/cptrs-ev

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des capteurs associés à un évènement spécifique, identifié par son `ev_id`.

Requête

Corps de la requête (Body)

```
{
  "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
ev_id	string (uuid)	Oui	Non	ID de l'évènement

Réponse

200 OK – Capteurs trouvés

```
[
  {
    "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "cptr_nom": "Capteur 1",
    "cptr_origine": "Capteur origine"
  }
  // ...
]
```

204 No Content – Aucun capteur trouvé

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["ev_id"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : capteur-log

Méthode et URL

GET /api/capteur-log/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des valeurs des capteurs pour un ou plusieurs capteurs, identifiés par leur(s) `cptr_id`.

Un filtre optionnel sur la date (`dateStart` au format `YYYY-MM-DD`) peut être appliqué via le paramètre de requête.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	string (uuid)	Oui	Un ou plusieurs identifiants de capteurs, séparés par des virgules

Paramètres de requête

Nom	Type	Requis	Description
dateStart	string	Non	Date au format <code>YYYY-MM-DD</code>

Réponse

200 OK – Valeurs trouvées

```
{
  "data": [
    {
      "cptr_valeur_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
      "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
      "cptr_epc_id": "EPC-123",
      "cptr_epc_txt": "QR-123456",
      "cptr_valeur": 123.45,
      "cptr_valeur_timestamp": "2025-07-07T10:30:00.000Z",
      "cptr_local_timestamp": "2025-07-07T10:30:00.000Z",
      "cptr_valeur_traitee": null
    }
  ]
}
```

```

    }
    // ...
  ],
  "pagination": {
    "total": 5,
    "current_page": 1,
    "total_pages": 1,
    "has_next_page": false,
    "has_previous_page": false
  },
  "filter_info": {
    "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF,6F9619FF-8B86-D011-B42D-00CF4FC96500",
    "dateStart": "2025-07-07"
  }
}

```

400 Bad Request – Données invalides

```

{
  "data": [],
  "pagination": {
    "total": 0,
    "current_page": 1,
    "total_pages": 0,
    "has_next_page": false,
    "has_previous_page": false
  },
  "filter_info": {
    "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "dateStart": "2025-07-07",
    "message": "Aucune donnée trouvée pour cette période"
  }
}

```

200 OK – Aucun résultat

```

{
  "error": "Paramètre invalide",
  "details": [
    {
      "code": "custom",
      "message": "'abc' n'est pas un UUID valide"
    }
  ]
}

```

500 Internal Server Error – Erreur serveur

```

{
  "error": "Une erreur est survenue lors de la récupération des valeurs du capteur",
  "message": "Détail de l'erreur"
}

```

Endpoint : capteur-ping

Méthode et URL

PUT /api/capteur/{id}/ping

- **Méthode** : PUT
- **Authentification requise** : Non (authLevel: anonymous)

Description

Cet endpoint permet de signaler (ping) la mise à jour d'un capteur en enregistrant la date et l'heure du ping dans la base de données.

Il met à jour le champ `cptr_update` du capteur identifié par son `id`.

Requête

Paramètres de route

Nom	Type	Requis	Description
Id	string (uuid)	Oui	ID du capteur à mettre à jour

Réponse

200 OK – Valeurs trouvés

```
{
  "success": true,
  "timestamp": "2025-07-09T12:15:00.123Z",
  "capteur_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

400 Bad Request – Données invalides

```
{
  "error": "Paramètre invalide",
  "details": [
    {
      "code": "invalid_type",
      "message": "L'identifiant du capteur doit être un UUID valide"
    }
  ]
}
```

500 Internal Server Error – Erreur serveur

```
{  
  "error": "Internal server error"  
}
```


Endpoint : capteur-put

Méthode et URL

PUT /api/capteur

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de créer ou mettre à jour un capteur dans la base de données. Si le capteur existe déjà (cptr_id présent), il est mis à jour ; sinon, il est créé.

Requête

Corps de la requête (Body)

```
{
  "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "cptr_nom": "Capteur 1",
  "cptr_origine": "Capteur origine"
}
```

Champ	Type	Requis	Nullable	Description
cptr_id	string (uuid)	Oui	Non	ID du capteur
org_id	string (uuid)	Oui	Non	ID de l'organisateur
cptr_nom	string (255 max)	Non	Oui	Nom du capteur
cptr_origine	string (255 max)	Non	Oui	Origine du capteur

Réponse

200 OK – Capteur mis à jour avec succès

```
{
  "message": "Capteur mis à jour avec succès",
  "id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "operation": "UPDATE"
}
```

201 Created – Capteur créé avec succès

```
{
  "message": "Capteur créé avec succès",
  "id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "operation": "INSERT"
}
```

400 Bad Request – Données invalides

```
{
  "error": "Validation échouée",
  "details": {
    "cptr_id": {
      "_errors": [
        "L'identifiant du capteur doit être un UUID valide"
      ]
    }
  }
}
```

500 Internal Server Error – Erreur serveur

```
{
  "error": "Erreur serveur",
  "message": "Une erreur inattendue s'est produite"
}
```

Endpoint : capteur-valeurs-recent-get

Méthode et URL

POST /api/cptr-veleurs-recent

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la valeur la plus récente d'un capteur associé à un évènement spécifique (ev_id).
La recherche se fait pour la date de l'évènement, et retourne l'heure de la dernière valeur enregistrée.

Requête

Corps de la requête (Body)

```
{
  "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
ev_id	string (uuid)	Oui	Non	ID de l'évènement

Réponse

200 OK – Valeur trouvés

```
[
  {
    "cptr_valeur_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "heure_epc": "14:29:05"
  }
  // ...
]
```

204 No Content – Aucune valeur

400 Bad Request – Données invalides

```
{
  "issues": [
```

```
{
  "code": "invalid_type",
  "expected": "string",
  "received": "number",
  "path": ["ev_id"],
  "message": "Expected string, received number"
}
]
```

Endpoint : capteurs-valeurs-put

Méthode et URL

PUT /api/epc

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'insérer une nouvelle valeur de capteur dans la table [CAPTEURS_VALEURS].

Si l'identifiant de la valeur (cptr_valeur_id) n'est pas fourni, il est généré automatiquement.

Le timestamp est converti au format SQL datetime ou utilise l'heure du serveur si absent.

Requête

Corps de la requête (Body)

```
{
  "cptr_epc_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "cptr_epc_txt": "Capteur A",
  "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "cptr_valeur": 42.5,
  "cptr_valeur_timestamp": 1724150400000
}
```

Champ	Type	Requis	Nullable	Description
cptr_valeur_id	string (uuid)	Non	Non	ID de la valeur
cptr_epc_id	string	Oui	Non	ID EPC du capteur
cptr_epc_txt	string	Oui	Oui	Texte du capteur
cptr_id	string (uuid)	Oui	Non	ID du capteur
cptr_valeur	number	Oui	Oui	Valeur mesurée
cptr_valeur_timestamp	number	Oui	Oui	Timestamp Unix (ms)
Cptr_local_timestamp	time	Non	Non	Timestamp local (non utilisé dans l'insert)

Réponse

--à faire--

Endpoint : circuit-adr-circuit-put

Méthode et URL

PUT /api/circuit-adr-circuit

- **Méthode :** PUT
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer une adresse à un circuit dans la base de données. Il crée ou met à jour la relation entre un circuit (c_id) et une adresse (adr_id).

Requête

Corps de la requête (Body)

```
{
  "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
c_id	string (uuid)	Oui	Non	ID du circuit
adr_id	string (uuid)	Oui	Non	ID de l'adresse

Réponse

200 OK – Association créée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un status: 200.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["c_id"],
      "message": "Expected string, received number"
    }
  ]
}
```


Endpoint : circuit-adr-delete

Méthode et URL

POST /api/circuit-adrs

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les adresses associées à un circuit spécifique dans la base de données, en utilisant son identifiant (c_id).

Si le paramètre c_id est absent ou vide, aucune suppression n'est effectuée.

Requête

Corps de la requête (Body)

```
{
  "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
}
```

Champ	Type	Requis	Nullable	Description
c_id	string (uuid)	Non	Oui	ID du circuit

Réponse

204 No Content – Adresses supprimées

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un status: 200.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["c_id"],
      "message": "Format UUID invalide"
    }
  ]
}
```


Endpoint : circuit-adr-get

Méthode et URL

GET /api/circuit-adr/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des adresses associées à un circuit spécifique, identifié par son `id`.

Les informations retournées incluent l'identifiant, le nom, le libellé, le code postal, la commune, le pays, la longitude et la latitude de chaque adresse.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	string (uuid)	Oui	ID du circuit

Réponse

200 OK – Adresses trouvées

```
[
  {
    "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "adr_nom": "Nom adresse",
    "adr_libelle": "Libellé complet",
    "adr_code_postal": "75001",
    "adr_commune": "Paris",
    "adr_pays": "France",
    "adr_longitude": "2.3522",
    "adr_latitude": "48.8566"
  }
  // ...
]
```

200 OK – Aucun résultat

Endpoint : circuit-all

Méthode et URL

GET /api/circuits

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de récupérer tous les circuits depuis la base de données sous forme de tableau.

Réponse

200 OK – Liste des circuits retournée

```
[
  {
    "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "c_nom": "Nom du circuit",
    // autres champs...
  },
  {
    "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500",
    "c_nom": "Nom du circuit 2",
    // autres champs...
  }
  // ...
]
```

Endpoint : circuit-coord-circuit-put

Méthode et URL

PUT /api/circuit-coord-circuit

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer une coordonnée à un circuit dans la base de données. Il crée ou met à jour la relation entre un circuit (`c_id`) et une adresse (`coord_id`).

Requête

Corps de la requête (Body)

```
{
  "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "coord_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
c_id	string (uuid)	Oui	Non	ID du circuit
coord_id	string (uuid)	Oui	Non	ID de la coordonnée

Réponse

200 OK – Association créée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status` : 200.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["c_id"],
      "message": "Expected string, received number"
    }
  ]
}
```


Endpoint : circuit-coord-delete

Méthode et URL

POST /api/circuit-coords

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les coordonnées associées à un circuit spécifique dans la base de données, en utilisant son identifiant (c_id).

Si le paramètre c_id est absent ou vide, aucune suppression n'est effectuée.

Requête

Corps de la requête (Body)

```
{
  "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
}
```

Champ	Type	Requis	Nullable	Description
c_id	string (uuid)	Non	Oui	ID du circuit

Réponse

204 No Content – Coordonnées supprimées

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un status: 200.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["c_id"],
      "message": "Format UUID invalide"
    }
  ]
}
```

Endpoint : circuit-coord-get-simple

Méthode et URL

POST /api/circuit-coord-simple

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des coordonnées associées à un circuit spécifique, identifié par son `c_id`.

Les informations retournées incluent l'identifiant, la valeur et le type de chaque coordonnée.

Requête

Corps de la requête (Body)

```
{
  "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
<code>c_id</code>	string (uuid)	Non	Oui	ID du circuit

Réponse

200 OK – Coordonnées trouvées

```
[
  {
    "coord_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "coord_valeur": "48.8566,2.3522",
    "coord_type": "GPS"
  }
  // ...
]
```

204 No Content – Aucun résultat

400 Bad Request – Données invalides

```
{
  "issues": [
```

```
{
  "code": "invalid_type",
  "expected": "string",
  "received": "number",
  "path": ["c_id"],
  "message": "Format UUID invalide"
}
]
```

Endpoint : circuit-coord-get

Méthode et URL

POST /api/circuit-coord

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des emails et numéros de téléphone associées à un circuit spécifique, identifié par son ID.

La réponse sépare les valeurs en deux tableaux : emails et phones

Requête

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
org_id	string (uuid)	Non	Oui	ID du circuit

Réponse

200 OK – Coordonnées trouvées

```
{
  "emails": [
    "contact@exemple.com",
    "autre@exemple.com"
  ],
  "phones": [
    "+33123456789",
    "+33987654321"
  ]
}
```

204 No Content – Aucun résultat

400 Bad Request – Données invalides


```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["org_id"],
      "message": "Format UUID invalide"
    }
  ]
}
```

Endpoint : circuit-delete

Méthode et URL

DELETE /api/circuits/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer un circuit spécifique de la base de données en utilisant son id (c_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du circuit à supprimer

Réponse

204 No content – Circuit supprimée

Le status 204 est retourné dans tous les cas.

Endpoint : circuit-get

Méthode et URL

GET /api/circuits/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer un circuit spécifique depuis la base de données en utilisant son id (c_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du circuit recherché

Réponse

200 OK – Circuit trouvé

```
{
  "c_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "c_nom": "nom",
  "c_description": "description",
  "c_illustration": "url",
  "c_longueur_totale": 1,
  "c_largeur_piste": 1,
  "c_ligne_droite": 1,
  "c_virage_gauche": 1,
  "c_virage_droite": 1,
  "c_max_decibels": 98
}
```

404 Not Found – Aucun circuit trouvé pour l’ID donné

Endpoint : circuit-put

Méthode et URL

PUT /api/circuits/{id?}

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)
- **Paramètre de route facultatif :** id (GUID)

Description

Cet endpoint permet de **créer ou mettre à jour un circuit** dans la base de données.

- Si un id est fourni dans l'URL, le circuit qu'il désigne est mis à jour.
- Si aucun id n'est fourni, un nouveau circuit est créé et son id est généré automatiquement.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID du circuit à ajouter ou mettre à jour

Corps de la requête (Body)

```
{
  "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "c_nom": "nom",
  "c_description": "description",
  "c_illustration": "url",
  "c_longueur_totale": 1,
  "c_largeur_piste": 1,
  "c_ligne_droite": 1,
  "c_virage_droite": 1,
  "c_virage_gauche": 1,
  "c_decibels_max": 98
}
```

Champ	Type	Requis	Nullable	Description
adr_id	string (uuid)	Non	Oui	ID de l'adresse du circuit
c_nom	string	Non	Oui	Nom du circuit
c_description	string	Non	Oui	Description du circuit
c_illustration	string	Non	Oui	URL de l'illustration du circuit

Champ	Type	Requis	Nullable	Description
c_longueur_totale	number (int positif)	Non	Oui	Longueur totale du circuit
c_largeur_piste	number (int positif)	Non	Oui	Largeur de la piste du circuit
c_virage_droite	number (int positif)	Non	Oui	Nombre de virages à droite dans le circuit
c_virage_gauche	number (int positif)	Non	Oui	Nombre de virages à gauche dans le circuit
max_decibels_max	number (positif)	Non	Oui	

Réponse

200 OK – Circuit insérée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status: 200`.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["c_nom"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : coord-exist-get

Méthode et URL

POST /api/coord-exist

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de vérifier l'existence d'une coordonnée dans la base de données à partir de sa valeur (coord_valeur).

Il retourne un indicateur found et, si la coordonnée existe, les données associées

Requête

Corps de la requête (Body)

```
{
  "coord_valeur": "48.8566,2.3522"
}
```

Champ	Type	Requis	Nullable	Description
coord_valeur	string	Oui	Non	Valeur de la coordonnée

Réponse

200 OK – Recherche effectuée

Si la coordonnée existe :

```
{
  "found": true,
  "data": {
    "coord_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "coord_valeur": "48.8566,2.3522",
    "coord_type": "GPS"
    // autres champs...
  }
}
```

Si l'adresse n'existe pas :

```
{  
  "found": false,  
  "data": ""  
}
```

Endpoint : count-unique-epc-get

Méthode et URL

POST /api/count-unique-epc

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de compter le nombre d'EPC uniques (valeurs commençant par QR-) pour une session donnée (s_id) dans la table TOURS.

Requête

Corps de la requête (Body)

```
{
  "s_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
s_id	string (uuid)	Non	Oui	ID de la session

Réponse

200 OK – Nombre d'EPC uniques

5

400 Bad Request – Données invalides

Si la coordonnée existe :

```
{
  "message": "Paramètres 's_id' requis ou invalides."
}
```


Endpoint : ev-cptr-delete

Méthode et URL

DELETE /api/ev-cptr/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les associations capteur-événement pour un événement spécifique de la base de données en utilisant son id (ev_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du circuit à supprimer

Réponse

204 No content – Association supprimée

Le status 204 est retourné dans tous les cas.

Endpoint : ev-cptr-put

Méthode et URL

PUT /api/ev-cptr

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer un capteur à un évènement dans la base de données. Il crée ou met à jour la relation entre un évènement (ev_id) et un capteur (cptr_id).

Requête

Corps de la requête (Body)

```
{
  "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "cptr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
ev_id	string (uuid)	Oui	Non	ID de l'évènement
cptr_id	string (uuid)	Oui	Non	ID du capteur

Réponse

200 OK – Association créée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un status: 200.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["ev_id"],
    }
  ]
}
```

```
    "message": "Expected string, received number"  
  }  
]  
}
```

Endpoint : ev-org-get

Méthode et URL

POST /api/ev-org

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des évènements futurs pour une organisation, à partir de l'identifiant d'un évènement (ev_id).

Requête

Corps de la requête (Body)

```
{
  "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
ev_id	string (uuid)	Non	Oui	ID du circuit

Réponse

200 OK – Liste d'évènement futurs

```
[
  {
    "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "ev_nom": "Nom de l'évènement",
    "ev_date": "2025-07-01T09:00:00+00:00"
    // autres champs...
  }
  // ...
]
```

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",

```

```
    "received": "number",  
    "path": ["ev_id"],  
    "message": "Expected string, received number"  
  }  
]  
}
```

404 Not Found – Aucun résultat

Endpoint : evt-all

Méthode et URL

POST /api/ev

- **Méthode** : POST
- **Authentification requise** : Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des évènements selon les droits (r_droits) et l'identifiant du pilote (p_id).

Requête

Corps de la requête (Body)

```
{
  "r_droits": 1,
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
r_droits	number	Oui	Non	Niveau de droits du pilote
p_id	string (uuid)	Oui	Non	ID du pilote

Réponse

200 OK – Liste des évènements

```
[
  {
    "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "ev_nom": "Nom de l'évènement",
    // autres champs...
  },
  // ...
]
```

204 No Content – Aucun résultat

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["p_id"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : evt-delete

Méthode et URL

DELETE /api/ev/{id}

- **Méthode :** DELETE
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de supprimer un évènement spécifique de la base de données en utilisant son id (ev_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'évènement à supprimer

Réponse

204 No content – Évènement supprimé

Le status 204 est retourné dans tous les cas.

Endpoint : evt-get

Méthode et URL

GET /api/ev/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer les informations d'un évènement spécifique depuis la base de données en utilisant son id (ev_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'évènement recherché

Réponse

200 OK – Évènement trouvé

```
{
  "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "ev_nom": "Nom de l'évènement",
  "ev_date": "2024-12-17T09:00:00+00:00"
  // autres champs...
}
```

404 Not Found – Aucun évènement trouvé pour l'ID donné

Endpoint : evt-put

Méthode et URL

PUT /api/ev/{id?}

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)
- **Paramètre de route facultatif :** id (GUID)

Description

Cet endpoint permet de **créer ou mettre à jour un évènement** dans la base de données.

- Si un id est fourni dans l’URL, l’évènement qu’il désigne est mis à jour.
- Si aucun id n’est fourni, un nouvel évènement est créé et son id est généré automatiquement.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID de l’évènement à ajouter ou mettre à jour

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "c_if": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "ev_nom": "debut",
  "ev_description": "description",
  "ev_date": "2024-12-19T09:00:00+00:00",
  "ev_heure_debut": "09:00:00",
  "ev_heure_fin": "18:00:00",
  "ev_pause_debut": "13:00:00",
  "ev_pause_duree": 60
}
```

Champ	Type	Requis	Nullable	Description
org_id	string (uuid)	Oui	Non	ID de l’organisateur de l’évènement
c_id	string (uuid)	Oui	Non	ID du circuit accueillant l’évènement
ev_nom	string	Oui	Non	Nom du circuit
ev_description	string	Non	Oui	Description de l’évènement

Champ	Type	Requis	Nullable	Description
ev_date	string (datetime)	Non	Oui	Date de l'évènement
ev_heure_debut	string	Non	Oui	Heure de début de l'évènement
ev_heure_fin	string	Non	Oui	Heure de fin de l'évènement
ev_pause_debut	string	Non	Oui	Heure de début de la pause
ev_pause_duree	number (int)	Non	Oui	Durée de la pause

Réponse

200 OK – Évènement insérée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status: 200`.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["ev_nom"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : grp-all

Méthode et URL

GET /api/grp

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de récupérer tous les groupes depuis la base de données sous forme de tableau.

Réponse

200 OK – Liste des groupes retournée

```
[
  {
    "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "grp_nom": "Nom du groupe",
    // autres champs...
  },
  {
    "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500",
    "grp_nom": "Nom du groupe 2",
    // autres champs...
  }
  // ...
]
```

Endpoint : grp-delete

Méthode et URL

DELETE /api/grp/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer un groupe spécifique de la base de données en utilisant son id (grp_id).

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID du groupe à supprimer

Réponse

204 No content – Groupe supprimée

Le status 204 est retourné dans tous les cas.

Endpoint : grp-ev-get

Méthode et URL

GET /api/grp-ev/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des groupes de pilotes participant à un évènement spécifique désigné grâce à son id (ev_id).

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'évènement dont on recherche les groupes

Réponse

200 OK – Groupes trouvé

```
[
  {
    "grp_nom": "nom groupe",
    "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "grp_couleur": "bleu",
    "grp_ordre": 1,
    "grp_capacite": 2
  },
  {
    "grp_nom": "groupe vide",
    "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500",
    "grp_couleur": "rouge",
    "grp_ordre": 2,
    "grp_capacite": 4
  }
]
```

204 No Content – Aucun groupe pour l'évènement recherché

400 Bad Request – Format UUID invalide

500 Internal Server Error – Erreur serveur

Une erreur s'est produite lors de l'exécution de la requête SQL.

Endpoint : grp-get

Méthode et URL

GET /api/grp/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer les informations d'un groupe spécifique depuis la base de données en utilisant son id (ev_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du groupe recherché

Réponse

200 OK – Groupe trouvé

```
{
  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "grp_nom": "Nom du groupe"
  // autres champs...
}
```

404 Not Found – Aucun groupe trouvé pour l'ID donné

Endpoint : grp-pilote-get

Méthode et URL

GET /api/grp-pilote/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des identifiants de groupes (grp_id) associés à un pilote spécifique, en utilisant son identifiant (p_id).

La réponse retourne un tableau d'identifiants de groupes.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du pilote recherché

Réponse

200 OK – Groupes trouvés

```
[  
  "6F9619FF-8B86-D011-B42D-00CF4FC964FF",  
  "6F9619FF-8B86-D011-B42D-00CF4FC96500"  
  // ...  
]
```

200 OK – Aucun groupe trouvé

```
[]
```

400 Bad Request – Format UUID invalide

```
{  
  "error": "Invalid UUID format"  
}
```

500 Internal Server Error – Erreur serveur

Endpoint : grp-pilote-grp-p-contactinfo-update

Méthode et URL

PATCH /api/grp-pilote-grp-p-contactinfo

- **Méthode :** PATCH
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de mettre à jour les informations de contact (grp_p_contactinfo) d'un pilote pour un groupe spécifique dans la base de données.

Requête

Corps de la requête (Body)

```
{  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",  "grp_p_contactinfo": "contact@exemple.com"}
```

Nom	Type	Requis	Nullable	Description
grp_id	string (uuid)	Oui	Non	ID du groupe
p_id	string (uuid)	Oui	Non	ID du pilote
grp_p_contactinfo	string	Oui	Non	Informations de contacte à mettre à jour

Réponse

--à faire--

Endpoint : grp-put

Méthode et URL

PUT /api/grp/{id?}

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)
- **Paramètre de route facultatif :** id (GUID)

Description

Cet endpoint permet de créer ou mettre à jour un groupe existant dans la base de données. Si le paramètre de route id est fourni il est prioritaire sur celui du corps de la requête

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID du groupe à créer ou mettre à jour (prioritaire sur celui donnée dans le corps de la requête)

Corps de la requête (Body)

```
{
  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "grp_nom": "Nom du groupe",
  "grp_description": "Description du groupe",
  "grp_couleur": "bleu",
  "grp_capacite": "4",
  "grp_date_creation": "2024-6-24T10:31:58Z",
  "grp_ordre": 1
}
```

Champ	Type	Requis	Nullable	Description
grp_id	string (uuid)	Oui	Non	ID du groupe (L’URL est prioritaire)
ev_id	string (uuid)	Oui	Non	ID de l’évènement que compose le groupe
grp_nom	string	Oui	Non	Nom du groupe
grp_description	string	Non	Oui	Description du groupe
grp_couleur	string	Non	Oui	Couleur associée au groupe
grp_capacite	string	Non	Oui	Nombre de pilotes pouvant composer le groupe

Champ	Type	Requis	Nullable	Description
grp_date_creation	string	Non	Oui	Date de création du groupe
grp_ordre	number	Non	Oui	Numéro du groupe au sein de l'évènement qu'il compose

Réponse

200 OK – Groupe inséré ou mis à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status: 200`.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["grp_id"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : grp-session-put

Méthode et URL

PUT /api/grp-session

- **Méthode :** PUT
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer un groupe à une session dans la base de données. Il crée ou met à jour la relation entre une session (s_id) et un groupe (grp_id).

Requête

Corps de la requête (Body)

```
{
  "s_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
s_id	string (uuid)	Oui	Non	ID de la session
grp_id	string (uuid)	Oui	Non	ID du groupe

Réponse

--à faire--

Endpoint : health

Méthode et URL

GET /api/health

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint sert à tester la connectivité à la base de données SQL configurée dans l'environnement (connectionStringSetting: 'main').

Réponse

204 No Content – Succès

La connexion à la base de données est fonctionnelle

500 Internal Server Error – Échec

Une erreur est survenue lors de la tentative de connexion à la base.

Endpoint : orga-adr-delete

Méthode et URL

POST /api/orga-adrs

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les adresses associées à une organisation spécifique dans la base de données, en utilisant son identifiant (org_id).

Si le paramètre org_id est absent ou vide, aucune suppression n'est effectuée.

Requête

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
}
```

Champ	Type	Requis	Nullable	Description
org_id	string (uuid)	Non	Oui	ID de l'organisation

Réponse

--à faire--

Endpoint : orga-adr-get

Méthode et URL

GET /api/orga-adr/{id}

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de récupérer la liste des adresses associées à un organisateur spécifique, identifié par son id (org_id).

Les informations retournées incluent l'identifiant, le nom, le libellé, le code postal, la commune et le pays de chaque adresse.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	string (uuid)	Oui	ID de l'organisateur

Réponse

--à faire--

Endpoint : orga-adr-orga-put

Méthode et URL

PUT /api/orga-adr-orga

- **Méthode :** PUT
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer une adresse à un organisateur dans la base de données. Il crée ou met à jour la relation entre un organisateur (`org_id`) et une adresse (`adr_id`).

Requête

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
org_id	string (uuid)	Oui	Non	ID de l'organisateur
adr_id	string (uuid)	Oui	Non	ID de l'adresse

Réponse

--à faire--

Endpoint : orga-all

Méthode et URL

POST /api/orgas

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des organisateurs selon les droits (r_droits) et l'identifiant du pilote (p_id) sous forme de tableau.

Requête

Corps de la requête (Body)

```
{
  "r_droits": 1,
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
r_droits	number	Oui	Non	Niveau de droit du pilote
p_id	string (uuid)	Oui	Non	ID du pilote

Réponse

--à faire--

Endpoint : orga-coord-delete

Méthode et URL

POST /api/orga-coords

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les coordonnées associées à une organisation spécifique dans la base de données, en utilisant son identifiant (org_id).
Si le paramètre org_id est absent ou vide, aucune suppression n'est effectuée.

Requête

Corps de la requête (Body)

```
{  
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"  
}
```

Champ	Type	Requis	Nullable	Description
org_id	string (uuid)	Non	Oui	ID de l'organisation

Réponse

--à faire--

Endpoint : orga-coord-get-simple

Méthode et URL

POST /api/orga-coord-simple

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des coordonnées associées à un organisateur spécifique, identifié par son `org_id`.

Les informations retournées incluent l'identifiant, la valeur et le type de chaque coordonnée.

Requête

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
org_id	string (uuid)	Non	Oui	ID de l'organisateur

Réponse

--à faire--

Endpoint : orga-coord-get

Méthode et URL

POST /api/orga-coord

- **Méthode :** POST
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des emails et numéros de téléphone associées à un organisateur spécifique, identifié par son ID (`org_id`).

La réponse sépare les valeurs en deux tableaux : `emails` et `phones`.

Requête

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
org_id	string (uuid)	Non	Oui	ID de l'organisateur

Réponse

--à faire--

Endpoint : orga-coord-orga-put

Méthode et URL

PUT /api/orga-coord-orga

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer une coordonnée à un organisateur dans la base de données. Il crée ou met à jour la relation entre un organisateur (org_id) et une coordonnée (coord_id).

Requête

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "coord_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
org_id	string (uuid)	Oui	Non	ID de l'organisateur
coord_id	string (uuid)	Oui	Non	ID de la coordonnée

Réponse

--à faire--

Endpoint : orga-delete

Méthode et URL

DELETE /api/orgas/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer un organisateur spécifique de la base de données en utilisant son identifiant (org_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'organisateur à supprimer

Réponse

--à faire--

Endpoint : orga-get

Méthode et URL

GET /api/orgas/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer une organisation spécifique depuis la base de données en utilisant son id (c_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'organisation recherché

Réponse

--à faire--

Endpoint : orga-pilote-get

Méthode et URL

GET /api/orga-pilote/{id}

- **Méthode :** GET
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des identifiants d'organismes (org_id) associés à un pilote spécifique, en utilisant son identifiant (p_id).

La réponse retourne un tableau d'identifiants d'organismes.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du pilote recherché

Réponse

--à faire--

Endpoint : orga-put

Méthode et URL

PUT /api/orgas/{id?}

- **Méthode :** PUT
 - **Authentification requise :** Non (authLevel: anonymous)
 - **Paramètre de route facultatif :** id (GUID)
-

Description

Cet endpoint permet de **créer ou mettre à jour un organisateur** dans la base de données.

- Si un `id` est fourni dans l'URL, l'organisateur qu'il désigne est mis à jour.
 - Si aucun `id` n'est fourni, un nouvel organisateur est créé et son `id` est généré automatiquement.
-

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID de l'organisateur à ajouter ou mettre à jour

Corps de la requête (Body)

```
{
  "org_nom": "nom"
}
```

Champ	Type	Requis	Nullable	Description
org_nom	string	Oui	Oui	Nom de l'organisateur

Réponse

200 OK – Organisateur insérée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status` : 200.

400 Bad Request – Données invalides

```
{
```

```
"issues": [  
  {  
    "code": "invalid_type",  
    "expected": "string",  
    "received": "number",  
    "path": ["org_nom"],  
    "message": "Expected string, received number"  
  }  
]  
}
```

Endpoint : pilote-actif-update

Méthode et URL

PATCH /api/pilote-actif

- **Méthode :** PATCH
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de mettre à jour le statut actif (p_actif) d'un pilote dans la base de données.

Il active le pilote correspondant à l'identifiant fourni (p_id).

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
p_id	string (uuid)	Oui	Non	ID du pilote

Réponse

--à faire--

Endpoint : pilote-adr-delete

Méthode et URL

POST /api/pilote-adrs

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les adresses associées à un pilote spécifique dans la base de données, en utilisant son identifiant (p_id).

Si le paramètre p_id est absent ou vide, aucune suppression n'est effectuée.

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
}
```

Champ	Type	Requis	Nullable	Description
p_id	string (uuid)	Non	Oui	ID du pilote

Réponse

--à faire--

Endpoint : pilote-adr-get

Méthode et URL

GET /api/pilote-adr/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des adresses associées à un pilote spécifique, identifié par son id (p_id).

Les informations retournées incluent l'identifiant, le nom, le libellé, le code postal, la commune et le pays de chaque adresse.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	string (uuid)	Oui	ID du pilote

Réponse

--à faire--

Endpoint : pilote-adr-pilote-put

Méthode et URL

PUT /api/pilote-adr-pilote

- **Méthode :** PUT
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer une adresse à un pilote dans la base de données. Il crée ou met à jour la relation entre un pilote (p_id) et une adresse (adr_id).

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
p_id	string (uuid)	Oui	Non	ID du pilote
adr_id	string (uuid)	Oui	Non	ID de l'adresse

Réponse

--à faire--

Endpoint : pilote-adr-put

Méthode et URL

PUT /api/pilote-adr

- **Méthode** : PUT
- **Authentification requise** : Non (authLevel: anonymous)
- **Paramètre de route facultatif** : id (GUID)

Description

Cet endpoint permet de créer ou mettre à jour une adresse dans la base de données.
Les champs sont normalisés :

- adr_nom : accents supprimés
- adr_commune : accents supprimés, minuscule puis première lettre en majuscule
- adr_pays : accents supprimés, tout en majuscules

Requête

Corps de la requête (Body)

```
{
  "adr_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "adr_nom": "Nom adresse",
  "adr_libelle": "Libellé complet",
  "adr_code_postal": "75001",
  "adr_commune": "paris",
  "adr_pays": "france"
}
```

Champ	Type	Requis	Nullable	Description
adr_id	string (uuid)	Oui	Non	ID de l'adresse
adr_nom	string	Non	Oui	Nom de l'adresse
adr_libelle	string	Non	Oui	Libellé complet de l'adresse
adr_code_postal	string	Non	Oui	Code postal de l'adresse
adr_commune	string	Non	Oui	Commune de l'adresse
adr_pays	string	Non	Oui	Pays de l'adresse

Réponse

--à faire--

Endpoint : pilote-all-get

Méthode et URL

GET /api/pilote-all/{id_ev}

- **Méthode :** GET
- **Authentication requisite :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste de tous les pilotes associés à un événement spécifique (id_ev).
Les pilotes dont le nom ou le prénom contient "inconnu" sont exclus.
Les informations retournées incluent les données du pilote, du groupe pilote et du groupe.

Requête

Paramètres de route

Nom	Type	Requis	Description
Id_ev	string (uuid)	Oui	ID de l'évènement

Réponse

--à faire--

Endpoint : pilote-check-exists-get

Méthode et URL

POST /api/pilote-check-exists

- **Méthode** : POST
- **Authentification requise** : Non (authLevel: anonymous)

Description

Cet endpoint permet de vérifier l'existence d'un pilote dans la base de données à partir de son nom, prénom et contact (email ou téléphone).

La recherche est insensible à la casse et sensible ou insensible aux accents selon le champ.

Il retourne un indicateur `found` et, si le pilote existe, son identifiant (`p_id`).

Requête

Corps de la requête (Body)

```
{
  "nom": "Dupont",
  "prenom": "Jean",
  "contact": "jean.dupont@email.com"
}
```

Champ	Type	Requis	Nullable	Description
nom	string	Oui	Non	Nom du pilote
prenom	string	Oui	Non	Prénom du pilote
contact	string	Oui	Non	Email ou téléphone du pilote

Réponse

--à faire--

Endpoint : pilote-coord-delete

Méthode et URL

POST /api/pilote-coords

- **Méthode :** POST
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les coordonnées associées à un pilote spécifique dans la base de données, en utilisant son identifiant (p_id).

Si le paramètre p_id est absent ou vide, aucune suppression n'est effectuée.

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
p_id	string (uuid)	Non	Oui	ID du pilote

Réponse

--à faire--

Endpoint : pilote-coord-get-simple

Méthode et URL

POST /api/opilote-coord-simple

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des coordonnées associées à un pilote spécifique, identifié par son p_id.

Les informations retournées incluent l'identifiant, la valeur et le type de chaque coordonnée.

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
p_id	string (uuid)	Non	Oui	ID du pilote

Réponse

--à faire--

Endpoint : pilote-coord-get

Méthode et URL

POST /api/pilote-coord

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des emails et numéros de téléphone associées à un pilote spécifique, identifié par son ID (p_id).

La réponse sépare les valeurs en deux tableaux : emails et phones.

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
p_id	string (uuid)	Non	Oui	ID du pilote

Réponse

--à faire--

Endpoint : pilote-coord-pilote-put

Méthode et URL

PUT /api/pilote-coord-pilote

- **Méthode :** PUT
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'associer une coordonnée à un pilote dans la base de données. Il crée ou met à jour la relation entre un pilote (p_id) et une coordonnée (coord_id).

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "coord_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
p_id	string (uuid)	Oui	Non	ID du pilote
coord_id	string (uuid)	Oui	Non	ID de la coordonnée

Réponse

--à faire--

Endpoint : pilote-coord-put

Méthode et URL

PUT /api/pilote-coords

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de créer ou mettre à jour une coordonnée (email ou téléphone) dans la base de données.

Si le type est `phone`, le numéro est normalisé au format français (10 chiffres commençant par 0).

Les valeurs sont enregistrées en minuscules et sans accents.

Requête

Corps de la requête (Body)

```
{
  "coord_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "coord_type": "phone",
  "coord_valeur": "+33 6 01 02 03 04"
}
```

Champ	Type	Requis	Nullable	Description
coord_id	string (uuid)	Oui	Non	ID de la coordonnée
coord_type	string	Non	Oui	Type de la coordonnée ("phone" ou "email")
coord_valeur	string	Non	Oui	Valeur de la coordonnée

Réponse

--à faire--

Endpoint : pilote-delete

Méthode et URL

DELETE /api/pilote/{id}

- **Méthode :** DELETE
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer un pilote spécifique de la base de données en utilisant son identifiant (p_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du pilote à supprimer

Réponse

--à faire--

Endpoint : pilote-detail-get

Méthode et URL

GET /api/pilote-details/{id}

- **Méthode :** GET
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer les détails d'un pilote spécifique, identifié par son p_id.
La réponse inclut :

- Les informations du pilote (p_id)
- La liste des événements auxquels il est associé, avec :
 - Les informations de chaque événement (id, nom, date, heure de début/fin)
 - Le groupe associé (grp_nom)
 - Les sessions du groupe (id, heure de début/fin), triées par heure de début

Requête

Paramètres de route

Nom	Type	Requis	Description
id	string (uuid)	Oui	ID du pilote

Réponse

--à faire--

Endpoint : pilote-get

Méthode et URL

GET /api/pilote/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer un pilote spécifique depuis la base de données en utilisant son id (p_id).

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID du pilote recherché

Réponse

200 OK – Pilote trouvé

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "r_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_prenom": "prenom",
  "p_nom": "nom",
  "p_actif": "1",
  ...
}
```

404 Not Found – Aucun pilote trouvé pour l’ID donné

Endpoint : pilote-group-delete

Méthode et URL

DELETE /api/pilote-grp/{id}/{id_grp}

- **Méthode :** DELETE
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer l'association entre un pilote et un groupe dans la base de données, en utilisant l'identifiant du pilote (id) et l'identifiant du groupe (id_grp).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du pilote
id_grp	GUID	Oui	ID du groupe

Réponse

--à faire--

Endpoint : pilote-grp-get

Méthode et URL

GET /api/pilote-grp/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des groupes de pilotes participant à un évènement spécifique désigné grâce à son id (ev_id) ainsi que les informations des pilotes les composant.

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'évènement dont on recherche les groupes

Réponse

200 OK – Groupes trouvé

```
[
  {
    "grp_nom": "nom groupe",
    "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "grp_couleur": "bleu",
    "grp_ordre": 1,
    "grp_capacite": 2,
    "pilotes": [
      {
        "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
        "p_nom": "nom pilote 1",
        "p_prenom": "prenom pilote 1",
        "p_email": "pilote1@exemple.com",
        ...
      },
      {
        "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500",
        "p_nom": "nom pilote 2",
        "p_prenom": "prenom pilote 2",
        "p_email": "pilote2@exemple.com",
        ...
      }
    ]
  }
]
```

```
]
},
{
  "grp_nom": "groupe vide",
  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500",
  "grp_couleur": "rouge",
  "grp_ordre": 2,
  "grp_capacite": 4,
  "pilotes": []
}
]
```

204 No Content – Aucun groupe pour l'évènement recherché

Endpoint : pilote-grp-put

Méthode et URL

PUT /api/pilote-grp/{id?}

- **Méthode** : PUT
- **Authentification requise** : Non (authLevel: anonymous)
- **Paramètre de route facultatif** : id (GUID)

Description

Cet endpoint permet d'insérer un pilote dans un groupe déjà existant. . Si le paramètre de route id est fourni il est prioritaire sur celui du corps de la requête

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID du groupe auquel ajouter le pilote (prioritaire sur celui donnée dans le corps de la requête)

Corps de la requête (Body)

```
{
  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "veh_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "grp_p_attribut_perso": "attribut perso",
  "grp_p_cpctr1": 1,
  "grp_p_cpctr2": 2,
  "grp_p_contactinfo": "pilote@example.com",
  "grp_p_tps_median": 123456,
  "grp_p_tps_meilleur": 123455,
  "grp_p_locked": true,
  "grp_p_actif": true
}
```

Champ	Type	Requis	Nullable	Description
grp_id	string (uuid)	Oui	Non	ID du groupe (L'URL est prioritaire)
p_id	string (uuid)	Oui	Non	ID du pilote
veh_id	string (uuid)	Non	Oui	ID du véhicule
grp_p_attribut_perso	string (32 max) / number	Non	Oui	Attribut du pilote

Champ	Type	Requis	Nullable	Description
grp_p_cpctr1	string (32 max) / number	Non	Oui	
grp_p_cpctr2	string (32 max) / number	Non	Oui	
grp_p_contactinfo	string (255 max)	Non	Oui	Email/numéro de téléphone du pilote
grp_p_tps_median	number	Non	Oui	Temps médian du pilote parmi les tours qu'il a fait dans ce groupe
grp_p_tps_meilleur	number	Non	Oui	Meilleur temps du pilote parmi les tours qu'il a fait dans ce groupe
grp_p_locked	boolean	Non	Oui	
grp_p_actif	boolean	Non	Oui	Activité du pilote (true si actif, false si inactif)

Réponse

200 OK – Pilote insérée avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status: 200`.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["grp_id"],
      "message": "Expected string, received number"
    }
  ]
}
```


Endpoint : pilote-grp-upsert-put

Méthode et URL

PUT /api/pilote-grp-up

- **Méthode** : PUT
- **Authentification requise** : Non (authLevel: anonymous)
- **Paramètre de route facultatif** : id (GUID)

Description

Cet endpoint permet d’insérer ou mettre à jour un pilote dans un groupe existant.
Les temps (grp_p_tps_median, grp_p_tps_meilleur) sont transmis en millisecondes.

Requête

Corps de la requête (Body)

```
{
  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "veh_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "grp_p_attribut_perso": "attribut perso",
  "grp_p_cptrl": 1,
  "grp_p_cptrl2": 2,
  "grp_p_contactinfo": "pilote@example.com",
  "grp_p_tps_median": 123456,
  "grp_p_tps_meilleur": 123455,
  "grp_p_locked": true,
  "grp_p_actif": true
}
```

Champ	Type	Requis	Nullable	Description
grp_id	string (uuid)	Oui	Non	ID du groupe
p_id	string (uuid)	Oui	Non	ID du pilote
veh_id	string (uuid)	Non	Oui	ID du véhicule
grp_p_attribut_perso	string (32 max) / number	Non	Oui	Attribut du pilote
grp_p_cptrl	string (32 max) / number	Non	Oui	Compteur 1
grp_p_cptrl2	string (32 max) / number	Non	Oui	Compteur 2
grp_p_contactinfo	string (255 max)	Non	Oui	Email/numéro de téléphone du pilote

Champ	Type	Requis	Nullable	Description
grp_p_tps_median	number	Non	Oui	Temps médian (ms)
grp_p_tps_meilleur	number	Non	Oui	Meilleur temps (ms)
grp_p_locked	boolean	Non	Oui	Verrouillage
grp_p_actif	boolean	Non	Oui	Activité du pilote

Réponse

--à faire--

Endpoint : pilote-isAdmin

Méthode et URL

GET /api/pilote-isadmin/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de vérifier si un rôle (`r_id`) correspond à un administrateur dans la base de données.

Il retourne `true` si le champ `r_droits` du rôle vaut 999999, sinon `false`.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	string (uuid)	Oui	ID du rôle à vérifier

Réponse

--à faire--

Endpoint : pilote-nom-prenom-update

Méthode et URL

PATCH /api/pilote-nom-prenom

- **Méthode :** PATCH
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de mettre à jour le nom, le prénom et le statut actif (p_actif) d'un pilote dans la base de données.

Le nom et le prénom sont remplacés par une valeur aléatoire du type inconnuXXXXX (où XXXXX est un nombre aléatoire à 5 chiffres).

Le champ r_id est mis à null et p_actif à false.

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
p_id	string (uuid)	Oui	Non	ID du pilote

Réponse

--à faire--

Endpoint : pilote-one-adr-delete

Méthode et URL

DELETE /api/pilote-one-adr/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer une adresse spécifique de la base de données, en utilisant son identifiant (adr_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'adresse

Réponse

--à faire--

Endpoint : pilote-one-coord-delete

Méthode et URL

DELETE /api/pilote-one-coord/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer une coord spécifique de la base de données, en utilisant son identifiant (coord_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de la coordonnée

Réponse

--à faire--

Endpoint : pilote-organisateur-delete

Méthode et URL

DELETE /api/pilote-organisateur/{org_id}/{p_id}

- **Méthode :** DELETE
- **Authentication requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer l'association entre un pilote et un organisateur dans la base de données, en utilisant l'identifiant du pilote (p_id) et l'identifiant du groupe (org_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
p_id	GUID	Oui	ID du pilote
org_id	GUID	Oui	ID de l'organisateur

Réponse

--à faire--

Endpoint : pilote-organisateur-put

Méthode et URL

PUT /api/pilote-organisateur

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'insérer ou de mettre à jour une association entre un organisateur, un rôle et un pilote dans la base de données.

Il crée ou met à jour la relation entre un organisateur (org_id), un pilote (p_id) et un rôle (r_id).

Requête

Corps de la requête (Body)

```
{
  "org_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "r_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Champ	Type	Requis	Nullable	Description
org_id	string (uuid)	Oui	Non	ID de l'organisateur
p_id	string (uuid)	Oui	Non	ID du pilote
R_id	string (uuid)	Oui	Non	ID du rôle

Réponse

--à faire--

Endpoint : pilote-push

Méthode et URL

PUT /api/pilotes

- **Méthode** : PUT
- **Authentication requise** : Non (authLevel: anonymous)

Description

Cet endpoint permet de **créer ou de mettre à jour un pilote** dans la base de données.

Requête

Corps de la requête (Body)

```
{
  "r_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_prenom": "prenom",
  "p_nom": "url",
  "p_actif": 1
}
```

Champ	Type	Requis	Nullable	Description
r_id	string (uuid)	Non	Non	ID du rôle du pilote
p_id	String (uuid)	Oui	Non	ID du pilote
p_prenom	string	Oui	Oui	Prénom du pilote
p_nom	string	Oui	Oui	Nom du pilote
p_actif	boolean	Non	Oui	État d'activité du pilote

Réponse

--à faire--

Endpoint : pilote-put

Méthode et URL

PUT /api/pilote

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de **créer ou de mettre à jour un pilote (utilisateur)** dans la base de données.

Requête

Corps de la requête (Body)

```
{
  "r_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_prenom": "prenom",
  "p_nom": "url",
  "p_actif": 1
}
```

Champ	Type	Requis	Nullable	Description
r_id	string (uuid)	Non	Non	ID du rôle du pilote
p_id	String (uuid)	Oui	Non	ID du pilote
p_prenom	string	Oui	Oui	Prénom du pilote
p_nom	string	Oui	Oui	Nom du pilote
p_actif	boolean	Non	Oui	État d'activité du pilote

Réponse

200 OK – Pilote insérée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un status: 200.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
```

```
    "code": "invalid_type",
    "expected": "string",
    "received": "number",
    "path": ["p_nom"],
    "message": "Expected string, received number"
  }
]
```

Endpoint : pilote-search

Méthode et URL

GET /api/pilotes/search?q={terme}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de rechercher des résultats par nom ou prénom

Requête

Paramètres de requête

Nom	Type	Requis	Description
q	string	Oui	Terme à rechercher (minimum 3 caractères)

Réponse

--à faire--

Endpoint : pilote-update-put

Méthode et URL

PUT /api/pilotes/{id}

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de **créer ou de mettre à jour un pilote** dans la base de données.

Requête

Paramètres de route

Nom	Type	Requis	Description
p_id	GUID	Oui	ID du pilote (écraser par le body)

Corps de la requête (Body)

```
{
  "r_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_prenom": "prenom",
  "p_nom": "url",
  "p_actif": 1
}
```

Champ	Type	Requis	Nullable	Description
r_id	string (uuid)	Non	Non	ID du rôle du pilote
p_id	String (uuid)	Oui	Non	ID du pilote
p_prenom	string	Oui	Oui	Prénom du pilote
p_nom	string	Oui	Oui	Nom du pilote
p_actif	boolean	Non	Oui	État d'activité du pilote

Réponse

--à faire--

Endpoint : pilote-validate-get

Méthode et URL

PUT /api/pilote-validate

- **Méthode** : PUT
- **Authentification requise** : Non (authLevel: anonymous)

Description

Cet endpoint permet de valider les informations d'un pilote (nom, prénom, email, téléphone). Il retourne les informations du pilote, ainsi que les emails et téléphones associés.

Requête

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "nom": "Dupont",
  "prenom": "Jean",
  "email": "jean.dupont@example.com",
  "phone": "0601020304"
}
```

Champ	Type	Requis	Nullable	Description
p_id	string (uuid)	Non	Oui	ID du pilte
nom	string	Oui	Non	Nom du pilote
prenom	string	Oui	Non	Prénom du pilote
email	string (email)	Non	Oui	Email du pilote
phone	string	Non	Oui	Téléphone du pilote

Réponse

--à faire--

Endpoint : pilote-all-role

Méthode et URL

GET /api/pilotes-role

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de récupérer la liste de tous les pilotes ayant un rôle associé (directement ou via un organisateur).

Pour chaque pilote, la réponse inclut ses informations et la liste de ses relations avec les organisateurs (id, nom, rôle).

Réponse

--à faire--

Endpoint : pilote-all

Méthode et URL

GET /api/pilotes

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de récupérer la liste de tous les pilotes dont le nom et le prénom **ne contiennent pas** "inconnu".

Les résultats sont triés par nom (p_nom).

Réponse

--à faire--

Endpoint : session-ev-delete

Méthode et URL

DELETE /api/session/{id}

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer une session spécifique de la base de données en utilisant son id (s_id).

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID de la session à supprimer

Réponse

204 No content – Session supprimée

Le status 204 est retourné dans tous les cas.

Endpoint : session-get

Méthode et URL

GET /api/session/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer une session spécifique depuis la base de données en utilisant son id (s_id).

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID de la session recherché

Réponse

200 OK – Session trouvé

```
{
  "s_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "s_index": 1,
  "s_group": [
    {
      "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
      "grp_nom": "nom",
      "grp_couleur": "couleur"
    }
  ],
  "s_starttime": "09:00:00",
  "s_endtime": "09:20:00",
  "s_duree": 20,
  "s_ispause": false
}
```

404 Not Found – Aucune session trouvé pour l’ID donné

Endpoint : session-nogrp-delete

Méthode et URL

DELETE /api/session/del

- **Méthode :** DELETE
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de supprimer toutes les sessions de la table [SESSIONS] qui **ne sont associées à aucun groupe** (c'est-à-dire dont l'identifiant s_id n'apparaît pas dans [GROUPES_SESSIONS]).

Réponse

--à faire--

Endpoint : session-put

Méthode et URL

PUT /api/session

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de créer ou mettre à jour une session dans la table [SESSIONS].
Les champs acceptés sont : identifiant de session (s_id), index, heure de début/fin, statut pause, durée, identifiant d'événement (ev_id).

Requête

Corps de la requête (Body)

```
{
  "s_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "s_index": 1,
  "s_starttime": "2025-06-27T08:00:00.000Z",
  "s_endtime": "2025-06-27T09:00:00.000Z",
  "s_ispause": false,
  "s_duree": 3600,
  "ev_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500"
}
```

Champ	Type	Requis	Nullable	Description
s_id	string (uuid)	Oui	Non	ID de la session
s_index	number	Non	Oui	Index de la session
s_starttime	string	Non	Oui	Heure de début de la session
s_endtime	string	Non	Oui	Heure de fin de la session
s_ispause	boolean	Non	Oui	Défini si une session est une pause
s_duree	number	Non	Oui	Durée de la session en secondes
ev_id	string (uuid)	Non	Oui	ID de l'évènement dont fait partie la session

Réponse

--à faire--

Endpoint : session-same-get

Méthode et URL

GET /api/same-session/{id}

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de récupérer les informations détaillées d'une session spécifique (s_id), ainsi que les données du groupe associé.

Requête

URL

Nom	Type	Requis	Description
id	GUID	Oui	ID de la session

Réponse

--à faire--

Endpoint : sessions-put

Méthode et URL

PUT /api/sessions

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet d'insérer ou de mettre à jour une session dans la base de données via une procédure stockée.

Les champs acceptés sont : identifiant de session (s_id), identifiant de groupe (grp_id), index, heure de début/fin, statut pause, durée.

Requête

Corps de la requête (Body)

```
{
  "s_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "grp_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500",
  "s_index": 1,
  "s_starttime": "2025-06-27T08:00:00.000Z",
  "s_endtime": "2025-06-27T09:00:00.000Z",
  "s_ispause": false,
  "s_duree": 3600
}
```

Champ	Type	Requis	Nullable	Description
s_id	string (uuid)	Oui	Non	ID de la session
grp_id	string (uuid)	Oui	Non	ID du groupe
s_index	number	Non	Oui	Index de la session
s_starttime	string (date)	Non	Oui	Heure de début de la session
s_endtime	string (date)	Non	Oui	Heure de fin de la session
s_ispause	boolean	Non	Oui	Défini si la session est une pause ou non
s_duree	number	Non	Oui	Durée en seconde

Réponse

--à faire--

Endpoint : ticket-all

Méthode et URL

GET /api/tickets

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de récupérer tous les ticket depuis la base de données sous forme de tableau.

Réponse

200 OK – Liste de ticket retournée

```
[
  {
    "tck_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "ticket_nbr": 1,
    "tck_note": "note",
    "tck_date": "2024-12-19T08:00:00+00:00",
    "tck_etat": 1
  },
  {
    "tck_id": "6F9619FF-8B86-D011-B42D-00CF4FC96500",
    "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
    "ticket_nbr": 2,
    "tck_note": "note",
    "tck_date": "2024-12-19T09:00:00+00:00",
    "tck_etat": 1
  }
  ...
]
```

Endpoint : ticket-delete

Méthode et URL

DELETE /api/tickets/{id}

- **Méthode :** DELETE
 - **Authentication requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint permet de supprimer un ticket spécifique de la base de données, en utilisant son identifiant (coord_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du ticket

Réponse

--à faire--

Endpoint : ticket-get

Méthode et URL

GET /api/tickets/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer un ticket spécifique depuis la base de données en utilisant son id (tck_id).

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID du ticket recherché

Réponse

200 OK – Ticket trouvé

```
{
  "tck_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "ticket_nbr": 1,
  "tck_note": "note",
  "tck_date": "2024-12-19T08:00:00+00:00",
  "tck_etat": 1
}
```

404 Not Found – Aucun ticket trouvé pour l’ID donné

Endpoint : ticket-put

Méthode et URL

PUT /api/ticket/{id?}

- **Méthode :** PUT
- **Authentification requise :** Non (authLevel: anonymous)
- **Paramètre de route facultatif :** id (GUID)

Description

Cet endpoint permet de **créer ou mettre à jour un ticket** dans la base de données.

- Si un id est fourni dans l’URL, le ticket qu’il désigne est mis à jour.
- Si aucun id n’est fourni, un nouveau ticket est créé et son id est généré automatiquement.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Non	ID du ticket à ajouter

Corps de la requête (Body)

```
{
  "p_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "ticket_nbr": 1,
  "tck_note": "note",
  "tck_date": "2024-12-19T08:00:00+00:00",
  "tck_etat": 1
}
```

Champ	Type	Requis	Nullable	Description
p_id	string (uuid)	Oui	Oui	ID de l'utilisateur à l'origine du ticket
ticket_nbr	number (int)	Oui	Oui	
tck_note	string	Oui	Oui	Contenu du ticket
tck_date	string (datetime)	Oui	Oui	Date et heure auxquelles le ticket à été émis
adr_etat	number (int)	Oui	Oui	

Réponse

200 OK – Ticket insérée ou mise à jour avec succès

Aucun corps spécifique de réponse n'est retourné à ce stade, seulement un `status: 200`.

400 Bad Request – Données invalides

```
{
  "issues": [
    {
      "code": "invalid_type",
      "expected": "string",
      "received": "number",
      "path": ["tck_note"],
      "message": "Expected string, received number"
    }
  ]
}
```

Endpoint : tours-ev-get

Méthode et URL

GET /api/tours-ev/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des tours associés à un événement spécifique (ev_id).

La réponse inclut les informations du tour, du pilote, de la session et du groupe.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'évènement

Réponse

--à faire--

Endpoint : tours-get

Méthode et URL

GET /api/tours/{id}

- **Méthode :** GET
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer la liste des tours associés à un événement spécifique (ev_id).

Les données retournées incluent les informations du tour, du pilote, du groupe, de la session, et des attributs personnalisés.

Les résultats sont triés par date de tour (sp_timestamp) décroissante.

Requête

Paramètres de route

Nom	Type	Requis	Description
id	GUID	Oui	ID de l'évènement

Réponse

--à faire--

Endpoint : tours-meilleur-moyen-tps-get

Méthode et URL

POST /api/tours-meilleur-moyen-tps

- **Méthode :** POST
- **Authentification requise :** Non (authLevel: anonymous)

Description

Cet endpoint permet de récupérer le meilleur temps (tps_meilleur) et le temps moyen (tps_moyen) pour une session donnée (s_id).

Il utilise la procédure stockée [sp_best_mean_time_session].

Requête

Corps de la requête (Body)

```
{
  "s_id": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

Nom	Type	Requis	Nullable	Description
s_id	string (uuid)	Oui	Oui	ID de la session

Réponse

--à faire--

Endpoint : version

Méthode et URL

GET /api/version

- **Méthode :** GET
 - **Authentification requise :** Non (authLevel: anonymous)
-

Description

Cet endpoint retourne la version de l'application définie dans `package.json`. Si la version n'est pas trouvée alors la valeur retournée sera `null`

Réponse

200 OK – Version retournée avec succès

```
{
  "version": "1.3.5"
}
```

200 OK – Version absente

```
{
  "version": null
}
```