

# Documentação REST do projeto TCP

## Tactical Combat Planner

### 1. Recurso Teams ( /api/teams)

#### Listar todos as teams

Devolve a lista de todos as teams e com dados do ID, nome e descrição(o necessario para mostrar nas tabelas

**url:** /api/teams/  
**(get)**

Parâmetros:  
**none**

Sucesso (200):

```
[
  {
    "id": 45,
    "name": "TSMA",
    "description": "We play solo in mid but airsoft"
  },
  {
    "id": 46,
    "name": "FAZE",
    "description": "Funny Airsoft ZZZZZ Ecologists"
  }, ...
]
```

Erro:  
500: Erro de Servidor

#### Selecionar uma equipa especifica

buscar os dados da equipa ao fornecer o seu ID

**url:** /api/teams/:teamID  
**(get)**

Parâmetros:  
**teamID(obrigatorio):** id da team pretendida obter os dados

Sucesso (200):

```
{
  "privacy": 1,
  "id": 44,
```

```
"name": "SKT TAirsoft",  
"description": "Seriously competitive team"  
}
```

Erros:

500: Erro do servidor

400: a team não foi encontrada

```
{ "msg": "team id not found" }
```

### Inserir novo membro

Inserir um novo membro a uma team existente

url: `"/api/teams/:teamID/newmember"`

(post)

Parâmetros:

**teamID (obrigatorio):** id da team pretendida adicionar este membro

Dados:

```
{  
  "player": "5",  
  "ranking": "2",  
  "role": "1"  
}
```

Sucesso (200):

```
{  
  "fieldCount": 0,  
  "affectedRows": 1,  
  "insertId": 0,  
  "serverStatus": 2,  
  "warningCount": 0,  
  "message": "",  
  "protocol41": true,  
  "changedRows": 0  
}
```

Erros:

500: Erro de servidor

400: a team não foi encontrada

```
{ "msg": "team id not found" }
```

## Inserir nova equipa

Inserir uma nova equipa

**url:** `/api/teams/`

**(post)**

Parâmetros:

**none**

Dados:

```
{
  "name": "SKT",
  "desc": "SKT team description",
  "privacy": "1",
  "player": "1"
}
```

Sucesso (200):

```
{
  "status": 200,
  "data": {
    "fieldCount": 0,
    "affectedRows": 1,
    "insertId": 47,
    "serverStatus": 2,
    "warningCount": 0,
    "message": "",
    "protocol41": true,
    "changedRows": 0
  }
}
```

Erros:

500: Erro de servidor

Receber todos os membros de uma equipa especificada

**url:** `/api/teams/:teamID/members`

**(get)**

Parâmetros:

**teamID (obrigatorio):** id da team pretendida obter os dados

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "Player1",
    "Role": "Rifleman",
    "Ranking": "Leader"
  },
  {
    "id": 6,
    "name": "Player6",
    "Role": "Rifleman",
    "Ranking": "Player"
  }
]
```

Erros:

500: Erro do servidor

400: a team não foi encontrada

```
{ "msg": "team id not found" }
```

Receber todas as táticas de uma equipa especificada

**url:** `/api/teams/:teamID/tactics`

**(get)**

Parâmetros:

**teamID (obrigatorio):** id da team pretendida obter os dados

Sucesso (200):

```
[
  {
    "id": 9,
    "name": "Basic attack",
    "fieldID": 19,
    "field": "Forest hills",
    "image": "data:image/png;base64, ..."
  },
  { ... } , ...
]
```

Erros:

500: Erro do servidor

400: a team não foi encontrada

```
{ "msg": "team id not found" }
```

## Alterar a role de um jogador dentro de uma equipa

`/api/teams/:teamID/player/:playerid/role/`  
(put)

Parâmetros:

`teamID` = id da equipa onde se encontra o player

`playerid` = id do player que vamos trocar a role

Body:

`Obj.role` = id da role

Dados:

```
{
  "role": 1
}
```

Sucesso (200):

```
{
  "fieldCount": 0,
  "affectedRows": 1,
  "insertId": 0,
  "serverStatus": 2,
  "warningCount": 0,
  "message": "(Rows matched: 1 Changed: 1 Warnings: 0",
  "protocol41": true,
  "changedRows": 1
}
```

Erros:

500: Erro de servidor

## remover um jogador de uma equipa

`/api/teams/:teamid/player/:playerid`  
(delete)

Parâmetros:

`teamid` = id da equipa onde se encontra o player

`playerid` = id do player que vamos trocar a role

Sucesso (200):

```
{
  "fieldCount": 0,
  "affectedRows": 1,
}
```

```
"insertId": 0,  
"serverStatus": 2,  
"warningCount": 0,  
"message": "(Rows matched: 1 Changed: 1 Warnings: 0",  
"protocol41": true,  
"changedRows": 1  
}
```

Erros:  
500: Erro de servidor

### Alterar o ranking de um jogador dentro de uma equipa

`/api/teams/:teamid/player/: oldleader /giveLead/: newleader`  
(put)

Parâmetros:

**teamid** = id da equipa onde se encontra o player

**oldleader** = id do player que vai perder o ranking de lider

**newleader** = id do player que vai ganhar o role de lider

Sucesso (200):

```
{  
  "fieldCount": 0,  
  "affectedRows": 1,  
  "insertId": 0,  
  "serverStatus": 2,  
  "warningCount": 0,  
  "message": "(Rows matched: 1 Changed: 1 Warnings: 0",  
  "protocol41": true,  
  "changedRows": 1  
}
```

Erros:  
500: Erro de servidor

`status: 400, msg: 'couldnt find one of these IDs'`

Receber um membro específico de uma equipe

```
url: /api/teams/:teamID/members/:playerID
```

(get)

Parâmetros:

**teamid** = id da equipe onde se encontra o player

**playerID** id do player que queremos ver as teams dele

Sucesso (200):

```
{
  "id": 1,
  "name": "Player1",
  "age": 25,
  "email": ppp@gmail.com,
  "description": "Lorem ipsum dolor sit amet,.",
  "photo": "Player1.png",
  "role": "Scout",
  "ranking": "Player"
}
```

Erros:

500: Erro do servidor

400: Id do player ou/e id da team não encontrado

```
msg: 'one of the ids was not found'
```

## 2. Recurso Tacticas (/api/tactics)

### Criar uma nova tática

Dependendo do objeto que vem no body, pode ser criado para team ou para group

**url:** `"/api/tactics/"`

(post)

Dados:

```
{
  "name": "ATAQUE",
  "team_or_group": 1, // pode ser id de uma equipa ou evento, dependendo do value
  escolhido
  "field": 1,
  "player": 1,
  "image_path": " data:image/png;base64,..."
}
```

```
{
  "status": 200,
  "data": {
    "fieldCount": 0,
    "affectedRows": 1,
    "insertId": 11,
    "serverStatus": 2,
    "warningCount": 0,
    "message": "",
    "protocol41": true,
    "changedRows": 0
  }
}
```

Erros:

500: Erro de servidor

Receber tática especificada

**url:** `/api/tactics/:tacticID`

(get)

Parâmetros:

**tacticID (obrigatorio):** id da tática pretendida obter os dados



Sucesso (200):

```
{
  "id": 1,
  "name_field": "Serra de Sintra",
  "name": "Rush attack!",
  "team": 1,
  "path": "data:image/png;base64,..."
}
```

Erros:

500: Erro do servidor

400: a tactic não foi encontrada

```
{ "msg": "tactic id not found" }
```

Receber tactica especificada para um grupo

**url:** /api/tactics/events/:event/groups/:groupnum

**(get)**

Parâmetros:

**event (obrigatorio):** id da evento de onde esta o group

**groupnum (obrigatorio):** id do groupnum que tem a tactica

Sucesso (200):

```
{
  "image_tactic": "data:image/png;base64,...",
  "name_tactic": "DEFEND!"
}
```

Erros:

500: Erro do servidor

400: a tactic não foi encontrada

```
msg: 'one of the ids not found'
```

### 3. Recurso Roles(cargos) (/api/roles)

Receber todas as roles na DB

url: /api/roles  
(get)

Parâmetros:

None

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "Rifleman"
  },
  {
    "id": 2,
    "name": "Designated Marksman"
  },
  {
    "id": 3,
    "name": "Grenadier"
  },
  {
    "id": 4,
    "name": "Breachier"
  },
  {
    "id": 5,
    "name": "Support Gunner"
  },
  {
    "id": 6,
    "name": "Sniper"
  },
  {
    "id": 7,
    "name": "Scout"
  }
]
```

Erros:

500: Erro do servidor

400: nenhuma role foi encontrada

```
msg: 'roles not found'
```

#### 4 Recurso Player (/api/players)

Receber todos os players

**url:** /api/players/

**(get)**

Parâmetros:

**none**

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "Player1",
    "age": 25
  },
  {
    "id": 2,
    "name": "Player2",
    "age": 20
  }, . . .
]
```

Erros:

500: Erro do servidor

400: nenhum player encontrado

**msg:** 'no players found'

Receber player específico

**url:** /api/players/:playerID

**(get)**

Parâmetros:

**playerID** id do player que queremos info

Sucesso (200):

```
{
  "id": 1,
  "name": "Player1",
  "age": 25,
```

```
    "email": "ppp@gmail.com",
    "description": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent rutrum diam quis orci tempor semper. Nullam bibendum pulvinar nulla id fringilla. Pellentesque bibendum est lectus, ultrices efficitur massa cursus vel.",
    "photo": "Player1.png"
}
```

Erros:

500: Erro do servidor

400: nenhum player encontrado com este id

```
msg: 'player id not found'
```

Receber equipas que este player está incluído

```
url: /api/players/:playerID/teams
```

(get)

Parâmetros:

```
playerID id do player que queremos ver as teams dele
```

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "FaZe",
    "description": "If nothing can faze you, you are unflappable."
  }, ... {}
]
```

Erros:

500: Erro do servidor

400: nenhum player encontrado com este id

```
msg: 'player id not found'
```

Receber equipas que este player está incluído e é líder

```
url: /api/players/:playerID/leaderships/teams
```

(get)

Parâmetros:

`playerID` id do player que queremos ver as teams dele

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "FaZe"
  }, . . .
]
```

Erros:

500: Erro do servidor

400: este player não é lider de nenhuma equipa

`Return empty object`

Receber grupos que este player está incluído e é lider e o evento deste grupo

`url: /api/players/:playerID/leaderships/groups`

**(get)**

Parâmetros:

`playerID` id do player que queremos ver as teams dele

Sucesso (200):

```
[
  {
    "ranking": 1,
    "groupNumber": 2,
    "name_event": "Serra games!",
    "event": 1
  }, . . .
]
```

Erros:

500: Erro do servidor

400: este player não é lider de nenhum grupo

`Return empty object`



## 5 Recurso notificações (/api/notifications)

Receber notificações de um jogador

**url:** /api/notifications/player/:playerID

**(get)**

Parâmetros:

**playerID** id do player que queremos ver as notificações dele

Sucesso (200):

```
[
  {
    "id_notif": 6,
    "receiver": 3,
    "sender": 1,
    "teamInv": 1,
    "eventInv": null,
    "text_notif": "You have been invited to join FaZe by Player1",
    "invite": 1,
    "status": 1
  }, . . .
]
```

Erros:

500: Erro do servidor

400: nenhuma notificação encontrado para este player

**msg:** 'no notifications for this player'

Contar quantas notifications este player tem

**url:** /api/notifications/player/:playerID/count

**(get)**

Parâmetros:

**playerID** id do player que queremos ver as notificações dele

Sucesso (200):

```
{
  "num": 1
}
```



Erros:

500: Erro do servidor

400: nenhum notificação encontrado para este player

```
msg:'no notifications for this player'
```

## Alterar estado de uma notificação e se for um convite, POST o player na equipa/evento

/api/notifications/

(post)

Body:

```
{
  "id": "3", // id da notif
  "status": "2", //estado da notification
  "target": "1", //team ou event
  "newmember": "1", //pessoa q vai juntar
  "isTeam": "false" //decide se o target é team ou evento
}
```

**OU caso nao seja uma notificação de convite**

```
{
  "id": "3", // id da notif
  "status": "3", //estado da notification
}
```

Sucesso (200):

```
{
  "fieldCount": 0,
  "affectedRows": 1,
  "insertId": 0,
  "serverStatus": 2,
  "warningCount": 0,
  "message": "(Rows matched: 1 Changed: 1 Warnings: 0",
  "protocol41": true,
  "changedRows": 1
}
```

Erros:

500: Erro de servidor

Contar quantas notifications este player tem

url: /api/notifications/:idNotif

(get)

Parâmetros:

idNotif id do notificação que queremos ver as dados

Sucesso (200):

```
{
  "id_notif": 1,
  "receiver": 1,
  "sender": 4,
  "teamInv": null,
  "eventInv": 2,
  "text_notif": "Player 'Player4' is requesting to join your team 'Lil Hills Jogo nº1!'.",
  "invite": 0,
  "status": 2
}
```

Erros:

500: Erro do servidor

400: nenhuma notification encontrado com este id

msg: 'notification id not found'

## Criar notificação para: Player1 pede pra juntar à equipa do player2

`/api/notifications/player/:playerID/teams/:teamID/request`

(post)

Body:

```
{
  player: loggedPlayer,
  team: teamID,
  text: "Player '" + player.name + "' is requesting to join your team '" + teamOBJ.name + "'."
}
```

Sucesso (200):

```
{
  "fieldCount": 0,
  "affectedRows": 1,
  "insertId": 0,
  "serverStatus": 2,
  "warningCount": 0,
  "message": "(Rows matched: 1  Changed: 1  Warnings: 0",
  "protocol41": true,
  "changedRows": 1
}
```

Erros:

500: Erro de servidor

## Criar notificação para: Player1 pede pra juntar ao evento do player2

/api/notifications/player/:playerID/events/:eventID/request

(post)

Body:

```
{
  player: loggedPlayer,
  event: eventID,
  text: "Player '" + player.name + "' is requesting to join your event '" + eventOBJ.name + "'."
}
```

Sucesso (200):

```
{
  "fieldCount": 0,
  "affectedRows": 1,
  "insertId": 0,
  "serverStatus": 2,
  "warningCount": 0,
  "message": "(Rows matched: 1  Changed: 1  Warnings: 0",
  "protocol41": true,
  "changedRows": 1
}
```

Erros:

500: Erro de servidor

## 6 Recurso Field (/api/fields)

Get todos os fields com as suas coordenadas

**url:** /api/fields/

**(get)**

Parâmetros:

**none**

Sucesso (200):

{ // cada coordenada é um objeto dentro de um array, cada posição do array é um field diferente

```
[
  {
    "id": 1,
    "lat": "38.77818255664792",
    "lng": "-9.4238791092491",
    "name": "Serra de Sintra"
  },
  {
    "id": 1,
    "lat": "38.77584893587013",
    "lng": "-9.424056495784498",
    "name": "Serra de Sintra"
  }, ...
]
```

Erros:

500: Erro do servidor

400: nenhum field encontrado

Get todos os fields sem coordenadas

**url:** `/api/fields/distinct`

**(get)**

Parâmetros:

**none**

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "Serra de Sintra"
  },
  {
    "id": 2,
    "name": "Lil Hills"
  },
  {
    "id": 3,
    "name": "Parque verde"
  }, ...
]
```

Erros:

500: Erro do servidor

400: nenhum field encontrado

Get coordenadas de um field específico

**url:** `/api/fields/:fieldID/coordinates`

**(get)**

Parâmetros:

**fieldID :** ID do field que queremos ver as coordenadas

Sucesso (200):

```
[
  {
    "lat": "38.77818255664792",
```

```
    "lng": "-9.4238791092491",  
    "name": "Serra de Sintra"  
  },  
  {  
    "lat": "38.77584893587013",  
    "lng": "-9.424056495784498",  
    "name": "Serra de Sintra"  
  },  
  {  
    "lat": "38.77497596921387",  
    "lng": "-9.423003263230328",  
    "name": "Serra de Sintra"  
  }, . . .  
]
```

Erros:

500: Erro do servidor

400: nenhum field encontrado com esse ID

## 7 Recurso Eventos (/api/events)



Get todos os events

**url:** /api/events

**(get)**

**Parâmetros:**

**none**

**Sucesso (200):**

```
[
  {
    "id": 1,
    "name": "Serra games!",
    "field": "Serra de Sintra",
    "date": "2021-03-24"
  },
  {
    "id": 2,
    "name": "Lil Hills Jogo nº1!",
    "field": "Lil Hills",
    "date": "2021-04-30"
  },
  . . .
]
```

**Erros:**

500: Erro do servidor

400: nenhum event encontrado

Get evento específico

**url:** /api/events/:eventID

(get)

Parâmetros:

**eventID:** ID do evento que queremos

Sucesso (200):

```
{
  "id": 1,
  "name": "Serra games!",
  "field": 1,
  "date": "2021-03-24",
  "duration": "21600",
  "team_size": 10,
  "group_num": 3,
  "privacy": 1
}
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse ID

Get eventos que este player esta associado

**url:** /api/events/player/:playerID

(get)

Parâmetros:

**playerID:** ID do evento que queremos

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "Serra games!",
    "field": "Serra de Sintra",
    "date": "2021-03-24"
  },
  {
    "id": 2,
    "name": "Lil Hills Jogo nº1!",

```

```
    "field": "Lil Hills",  
    "date": "2021-04-30"  
  }, . . .  
]
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse player ID

Get players de um evento especifico

**url:** /api/events/:eventID/players  
**(get)**

Parâmetros:

**eventID:** ID do evento que queremos

Sucesso (200):

```
[  
  {  
    "name": "Player2",  
    "ranking": 2,  
    "id": 2,  
    "event": 1,  
    "team": null  
  },  
  {  
    "name": "Player4",  
    "ranking": 2,  
    "id": 4,  
    "event": 1,  
    "team": null  
  }  
]
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse ID

Get player específico de um evento específico

**url:** `/api/events/:eventID/players/:playerid`  
**(get)**

Parâmetros:

**eventID:** ID do evento que queremos  
**:playerid:** id do player

Sucesso (200):

```
{
  "name": "Player2",
  "ranking": 2,
  "id": 2,
  "event": 1,
  "team": null
}
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse ID

Get players de um grupo do evento especificado

**url:** `/api/events/:eventID/groups/:num/members`  
**(get)**

Parâmetros:

**eventID:** ID do evento que queremos  
**:num:** numero do grupo dentro do event

Sucesso (200):

```
[
  {
    "id_player": 1,
    "ranking": 1,
    "name_player": "Player1",
    "team": null
  }, ...
]
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse ID



Get numero de grupos de um event

**url:** `/api/events/:eventID/numofgroups`  
**(get)**

Parâmetros:

**eventID:** ID do evento que queremos

Sucesso (200):

```
{
  "group_num": 3
}
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse ID

Get todos os eventos que estao associados a um field especificado

**url:** `/api/events/fields/:fieldID`

**(get)**

Parâmetros:

**fieldID:** ID do field que queremos

Sucesso (200):

```
[
  {
    "id": 1,
    "name": "Serra games!",
    "date": "2021-03-24",
    "duration": "21600",
    "numOfTeams": 10,
    "group_num": 3
  }, . . .
]
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse ID



Get settings de um evento e info extra de field e do dono do evento

**url:** `/api/events/:eventID/settings`

**(get)**

Parâmetros:

**eventID:** ID do event que queremos

Sucesso (200):

```
{
  "id": 1,
  "name": "Serra games!",
  "field": "Serra de Sintra",
  "date": "2021-03-24",
  "duration": "21600",
  "team_size_event": 10,
  "group_num": 3,
  "privacy": 1,
  "name_player": "Player1",
  "email_player": "ppp@gmail.com"
}
```

Erros:

500: Erro do servidor

400: nenhum event encontrado com esse ID

**Inserir novo evento**

**url:** `/api/events/`

**(post)**

Parâmetros:

**none**

```
{
  "name": "New event!",
  "field": 1,
  "date": "2021-5-5",
}
```

```
"duration": 12600,  
"groupNum": 2,  
"teamsSize": 5,  
"privacy": 1,  
"player": "1"  
}
```

Sucesso (200):

```
{  
  "status": 200,  
  "data": {  
    "fieldCount": 0,  
    "affectedRows": 1,  
    "insertId": 5,  
    "serverStatus": 2,  
    "warningCount": 0,  
    "message": "",  
    "protocol41": true,  
    "changedRows": 0  
  }  
}
```

Erros:

500: Erro de servidor



## Inserir novo evento

`/api/events/`  
(post)

Parâmetros:  
none

```
{
  "name": "New event!",
  "field": 1,
  "date": "2021-5-5",
  "duration": 12600,
  "groupNum": 2,
  "teamsSize": 5,
  "privacy": 1,
  "player": "1"
}
```

Sucesso (200):

```
{
  "status": 200,
  "data": {
    "fieldCount": 0,
    "affectedRows": 1,
    "insertId": 5,
    "serverStatus": 2,
    "warningCount": 0,
    "message": "",
    "protocol41": true,
    "changedRows": 0
  }
}
```

Erros:  
500: Erro de servidor

