

Documentação REST

Projeto Kertana

Recurso Farmer(/api/farmers)

Obter Agricultor

Devolve a informação do agricultor.

/api/farmers/:farmerID

Parâmetros:

farmerID: ID do agricultor

Sucesso (200):

```
[
  {
    "Agricultor_ID": 1,
    "Agricultor_PrimeiroNome": "Nary",
    "Agricultor_UltimoNome": "Chaby",
    "Agricultor_Email": "nary.email.com",
    "Agricultor_Password": "12345678",
    "Agricultor_Photo": "https://thumbs.dreamstime.com/z/happy-farmer-19384661.jpg"
  }
]
```

Erro:

500: Erro de Servidor

Exemplo:

```
farmer = await $.ajax({
  url: "/api/farmers/"+farmerID,
  method: "get",
  dataType: "json"
});
```

Obter Terrenos Devolve os terrenos pertencentes ao agricultor.
/api/farmers/:farmerID/fields
Parâmetros: farmerID: ID do agricultor;
Sucesso (200): <pre>[{ "Terreno_ID": 1, "Terreno_Agricultor_ID": 1, "Terreno_Nome": "Plantação de Cenouras", "Terreno_Descricao": "A minha plantação de cenouras de Portalegre.", "Terreno_Coordenadas": "[[-8.518581663738274, 37.10718702613035], [-8.518581663738274, 37.10718702613035]]", "Terreno_AgroAPI_ID": "6013c63256bf937be6c40d50", "Terreno_FreguesiaID": 1 }]</pre>
Erro: 500: Erro de Servidor
Exemplo: <pre>fields = await \$.ajax({ url: "/api/farmers/"+farmerID+"/fields", method: "get", dataType: "json" });</pre>

Inserir um novo terreno Guarda um terreno criado pelo agricultor
/api/farmers/:farmerID/fields (post)
Parâmetros: farmerID: ID do agricultor;
Dados:{ <pre> "nome": "Plantação de Cenouras", "descricao": "A minha plantação de cenouras de Portalegre.", "coordenadas": "[[-8.518581663738274, 37.10718702613035], [-8.518581663738274, 37.10718702613035]]" , "agroId": 1, "fregID": 3 </pre> }
Sucesso (200): <pre>{ "fieldCount": 0, "affectedRows": 1, "insertId": 13, "serverStatus": 2,</pre>

```
}  
  "warningCount": 0, "message": "", "protocol41": true, "changedRows": 0
```

Erros:

500: Erro de servidor

Exemplo:

```
let name = document.getElementById('slotName').value  
let description = document.getElementById('slotDescription').value  
let agroID = await createPolygon(updateArea())  
let send = await $.ajax({  
  url: 'api/farmer/' + farmerID + '/fields',  
  method: 'post',  
  dataType: 'json',  
  data: {  
    "nome": name,  
    "descricao": description,  
    "coordenadas": JSON.stringify(updateArea()[0]),  
    "agroID": agroID,  
    "fregID": fregID  
  }  
})
```

Remover um terreno

Remove o terreno selecionado pelo agricultor

/api/farmers/:farmerID/fields/:fieldID (post)

Parâmetros:

farmerID: ID do agricultor;

fieldID: ID do terreno;

Sucesso (200):

```
{  
  "msg": "Successful database update "  
}
```

Erros:

500: Erro de servidor

Exemplo:

```
let name = document.getElementById('slotName').value  
let description = document.getElementById('slotDescription').value  
let agroID = await createPolygon(updateArea())
```

```

let send = await $.ajax({
  url: 'api/farmer/' + farmerID + '/fields',
  method: 'post',
  dataType: 'json',
  data: {
    "nome": name,
    "descricao": description,
    "coordenadas": JSON.stringify(updateArea()[0]),
    "agroID": agroID,
    "fregID": fregID
  }
})

```

Recurso Field(/api/fields)

Obter Produções

Devolve as produções num determinado terreno.

/api/fields/:fieldID/productions

Parâmetros:

fieldID: ID do terreno

Sucesso (200):

```

[
  {
    "products": [],
    "productions": [],
    "growthStates": []
  }
]

```

Erro:

500: Erro de Servidor

Exemplo:

```

let data = await $.ajax({
  url: 'api/fields/' + field.Terreno_ID + '/productions',
  method: 'get',
  dataType: 'json'
})

```

Inserir Produção

Insere uma produção criada pelo agricultor

/api/fields/:fieldID/productions (post)

Parâmetros:

fieldID: ID do terreno

Dados: {
 "date": "11/11/2020",
 "product": 1,
 "coordinates": "[[-8.518581663738274, 37.10718702613035], [-8.518581663738274, 37.10718702613035]]"
}

Sucesso (200):

```
{  
  "fieldCount": 0, "affectedRows": 1, "insertId": 3, "serverStatus": 2,  
  "warningCount": 0, "message": "", "protocol41": true, "changedRows": 0  
}
```

Erros:

500: Erro de servidor

Exemplo:

```
let production = {  
    coordinates: JSON.stringify(polygon),  
    product: productID,  
    date: date  
}
```

Remover Produção

Remove uma produção selecionada pelo agricultor

/api/fields/:fieldID/productions/:productionID (post)

Parâmetros:

fieldID: ID do terreno

productionID: ID da produção

Sucesso (200):

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

Erros:

500: Erro de servidor

Exemplo:

```
let remove = await $.ajax({  
    url: 'api/fields/'+field.Terreno_ID+'/productions/'+prodID,  
    method: 'post',  
    dataType: 'json'  
}))
```

Recurso Locations(/api/locations)

Obter Localizações

Devolve as localizações.

/api/locations

Sucesso (200):

```
[
  {
    "Freguesia_ID": 1,
    "Freguesia_Nome": "Lisboa",
    "Freguesia_Coordenadas": "[[-9.085713896778003, 38.82203162944219],...,[ -9.085713896778003, 38.82203162944219]]",
    "Freguesia_PoligonoID": 0
  }
]
```

Erro:

500: Erro de Servidor

Exemplo:

```
let locations = await $.ajax({
  url: '/api/locations',
  method: 'get',
  dataType: 'json'
});
```

Obter produtos

Devolve os produtos aptos para o cultivo numa localização.

/api/locations/:locationID/products

Parâmetros:

locationID: ID da localização

Sucesso (200):

```
[
  {
    "Produto_ID": 2,
    "Produto_Nome": "Cenoura",
    "Produto_Photo": "https://www.tasteofhome.com/wp-content/uploads/2019/01/carrots-hutterstock_789443206.jpg",
    "Produto_TipoSoloDescricao": "Profundo, rico e leve. De preferência, arenoso.",
    "Produto_EpocaSemearDescricao": "entre Outubro e Novembro em túnel de proteção. De Fevereiro a Abril em terra.",
    "Produto_EpocaColheitaDescricao": "entre Maio e Agosto",
  }
]
```

<pre> "Produto_ExposicaoDescricao": "Ensolarada "</pre>
Erro: 500: Erro de Servidor
Exemplo: <pre>let suitable_products = await \$.ajax({ url: "/api/locations/"+localizacao.id+"/products", method: 'get', dataType: 'json' })</pre>

Recurso Products(/api/products)

Obter Produtos Devolve os produtos.
/api/products
Sucesso (200): <pre>[{ "Produto_ID": 1, "Produto_Nome": "Tomate", "Produto_Photo": "https://www.almanac.com/sites/default/files/styles/primary_image_in_article/public/image_nodes/tomatoes_helios4eos_gettyimages-edit.jpeg?itok=4KrW14a4", "Produto_TemperaturaMaxima": 35, "Produto_TemperaturaMinima": 16, "Produto_TempoGerminacao": 11, "Produto_TempoMaturacao": 80, "Produto_HumidadeSoloMaxima": 60, "Produto_HumidadeSoloMinima": 85, "Produto_Mes_Inicial": 3, "Produto_Mes_Final": 6, "Produto_TipoSoloDescricao": "Rico e bem drenado. Trabalhado em profundidade.", "Produto_EpocaSemearDescricao": "De Fevereiro a Maio.", "Produto_EpocaColheitaDescricao": "De Julho a Outubro.", "Produto_ExposicaoDescricao": "Ensolarada. " },]</pre>
Erro: 500: Erro de Servidor
Exemplo:

```
let products = await $.ajax({
  url: '/api/products',
  method: 'get',
  dataType: 'json'
});
```

Obter Produto

Devolve um produto específico.

/api/products/:productID

Sucesso (200):

```
[
  {
    "Produto_ID": 1,
    "Produto_Nome": "Tomate",
    "Produto_Photo": "https://www.almanac.com/sites/default/files/styles/primary_image_in_article/public/image_nodes/tomatoes_helios4eos_gettyimages-edit.jpeg?itok=4KrW14a4",
    "Produto_TemperaturaMaxima": 35,
    "Produto_TemperaturaMinima": 16,
    "Produto_TempoGerminacao": 11,
    "Produto_TempoMaturacao": 80,
    "Produto_HumidadeSoloMaxima": 60,
    "Produto_HumidadeSoloMinima": 85,
    "Produto_Mes_Inicial": 3,
    "Produto_Mes_Final": 6,
    "Produto_TipoSoloDescricao": "Rico e bem drenado. Trabalhado em profundidade.",
    "Produto_EpocaSemearDescricao": "De Fevereiro a Maio.",
    "Produto_EpocaColheitaDescricao": "De Julho a Outubro.",
    "Produto_ExposicaoDescricao": "Ensolarada. "
  },
]
```

Erro:

500: Erro de Servidor

Exemplo:

```
let locations = await $.ajax({
  url: '/api/locations',
  method: 'get',
  dataType: 'json'
});
```

Obter localizações

Devolve as localizações aptas para o cultivo um produto.

/api/products/:productID /locations

Parâmetros:

productID: ID do produto

Sucesso (200):

```
[
  {
    "Freguesia_ID": 1,
    "Freguesia_Nome": "Lisboa",
    "Freguesia_Coordenadas": "[[-9.085713896778003, 38.82203162944219],...,[ -9.085713896778003, 38.82203162944219]]",
    "Freguesia_PoligonoID": 0
  }
]
```

Erro:

500: Erro de Servidor

Exemplo:

```
let suitable_locations = await $.ajax({
  url: "/api/products/"+product.id+"/locations",
  method: 'get',
  dataType: 'json'
})
```