



MONITORAMENTO AMBIENTAL

Smart Cities

MongoDB

Resumo

Um novo paradigma com NOT ONLY SQL - Utilizando NOSQL
Implementado em web.api.asp.NET

Wagner Machado – RM99307
Alianne Reis dos Santos – RM99937
Nathalia de Campos Nasz – RM98962
Adriano Reis Gama Monteiro - RM98133

Lista de Figuras

Figura 1	9
Figura 2	9
Figura 3	10
Figura 4	10
Figura 5	10
Figura 6	11
Figura 7	11
Figura 8	11
Figura 9	12
Figura 10	12
Figura 11	12
Figura 12	12
Figura 13	13
Figura 14	13
Figura 15	13
Figura 16	14
Figura 17	14
Figura 18	14
Figura 19	15
Figura 20	15
Figura 21	15
Figura 22	15
Figura 23	16
Figura 24	16
Figura 25	17
Figura 26	17
Figura 27	17
Figura 28	18
Figura 29	18
Figura 30	19
Figura 31	19
Figura 32	20
Figura 33	20
Figura 34	21
Figura 35	21
Figura 36	22
Figura 37	22
Figura 38	23
Figura 39	24
Figura 40	25
Figura 41	26
Figura 42	27
Figura 43	28
Figura 44	29
Figura 45	30
Figura 46	31
Figura 47	32
Figura 48	33
Figura 49	34
Figura 50	35

Figura 51	36
Figura 52	37
Figura 53	38
Figura 54	39
Figura 55	40
Figura 56	41
Figura 57	42
Figura 58	43
Figura 59	44
Figura 60	45
Figura 61	46
Figura 62	47

Sumário

Decisão de Migrar	6
Alta disponibilidade:.....	6
Escalabilidade.....	6
Flexibilidade de esquema	6
Justificativa para a Escolha de Cada Collection e seu Papel no Projeto	6
Collection: Sensores	7
Justificativa.....	7
Papel no Projeto.....	7
Collection: Leituras	7
Justificativa.....	7
Papel no Projeto.....	7
Collection: Alertas	7
Justificativa.....	7
Papel no Projeto.....	8
Collection: PrevisoesChuva	8
Justificativa.....	8
Papel no Projeto.....	8
Collection: ControleIrrigacoes	8
Justificativa.....	8
Papel no Projeto.....	8
Comandos para criação das collections e todas as operações CRUD	9
Criação das Collections	9
Primeiro Conectar ao MongoDB.....	9
Segundo Criar as Collections	9
CRUD – Documentos na Collection Alertas	10
InsertMany	10
FindOne.....	10
UpdateOne.....	10
DeleteOne	10
CRUD – Documentos na Collection Sensores	10
InsertMany	10
Find	11
FindOne.....	11
UpdateOne	11
DeleteOne	11
CRUD – Documentos na Collection Leituras	12
InsertMany	12

Find	12
FindOne	12
UpdateOne	13
DeleteOne	13
CRUD – Documentos na Collection Previsões de Chuva	13
InsertMany	13
Find	14
FindOne	14
UpdateOne	14
DeleteOne	14
CRUD – Inserindo Documentos na Collection Controle Irrigações	15
InsertMany	15
Find	15
FindOne	16
UpdateOne	16
DeleteOne	16
Evidências das Collections criadas	17
DataBase e Collections criados no MongoDB	17
DataBase e Collections criados pela API Web Asp.NET	17
Models / Collections	19
Evidencias da Manipulação dos Dados Via Swagger UI	22
Collection Sensores	22
Collection Alertas	27
Collection Leituras	32
Collection Previsões Chuva	37
Collection Controle de Irrigações	42

Um novo paradigma com NOT ONLY SQL - Utilizando NOSQL

Monitoramento Ambiental – Smart Cities

Decisão de Migrar

A escolha de migrar o projeto para MongoDB é justificada pelas características do MongoDB que beneficiam uma aplicação para cidades inteligentes:

Alta disponibilidade:

MongoDB oferece suporte a replicação, garantindo que os dados estejam sempre disponíveis.

Escalabilidade

A capacidade de escalar horizontalmente permite gerenciar grandes volumes de dados.

Flexibilidade de esquema

Permite a adaptação rápida às mudanças nos requisitos de dados.

Justificativa para a Escolha de Cada Collection e seu Papel no Projeto

A seguir, são descritas as 5 collections escolhidas e seu papel no projeto:

Collection: Sensores

Justificativa

Armazena informações detalhadas sobre os sensores instalados nos parques da cidade. Sensores são elementos críticos em um sistema de monitoramento ambiental, pois são responsáveis por coletar dados necessários para a análise de condições ambientais.

Papel no Projeto

Fornece dados essenciais sobre a localização, tipo e código dos sensores, permitindo a identificação e o gerenciamento eficiente dos dispositivos de monitoramento.

Collection: Leituras

Justificativa

Armazena os dados coletados pelos sensores. Estes dados são cruciais para entender as condições ambientais em tempo real e para a análise histórica.

Papel no Projeto

Permite o armazenamento e recuperação das leituras dos sensores, associando cada leitura a um sensor específico, facilitando a análise dos dados coletados.

Collection: Alertas

Justificativa

Armazena alertas gerados com base nas leituras dos sensores. Alertas são importantes para notificar os usuários sobre condições ambientais anormais ou potencialmente perigosas.

Papel no Projeto

Permite o armazenamento e recuperação das leituras dos sensores, associando cada leitura a um sensor específico, facilitando a análise dos dados coletados.

Collection: PrevisoesChuva

Justificativa

Armazena previsões de chuva obtidas de fontes externas. A previsão de chuva é fundamental para o planejamento e gestão de atividades relacionadas à água.

Papel no Projeto

Auxilia na tomada de decisões informadas sobre o controle de irrigação e outras atividades que dependem das condições meteorológicas.

Collection: ControleIrigacoes

Justificativa

Armazena informações sobre o controle de sistemas de irrigação automatizada. O controle eficiente da irrigação é essencial para a conservação de água e a manutenção da saúde das plantas em áreas verdes urbanas.

Papel no Projeto

Permite o controle automatizado e otimizado da irrigação com base em previsões de chuva e outras condições ambientais, garantindo o uso eficiente dos recursos hídricos.

Comandos para criação das collections e todas as operações CRUD

Criação das Collections

Primeiro Conectar ao MongoDB

```
use monitoramento_ambiental;
```

```
>_MONGOSH
> use monitoramento_ambiental;
< switched to db monitoramento_ambiental
monitoramento_ambiental> db.createCollection("alertas");
                        db.createCollection("sensores");
                        db.createCollection("leituras");
                        db.createCollection("controle_irrigacoes");
                        db.createCollection("previsoes_chuva");
```

Figura 1

Segundo Criar as Collections

My Queries

Performance

Databases

Search

admin

config

local

monitoramento_ambiental


■ alertas

■ controle_irrigacoes

■ leituras

■ previsoes_chuva

■ sensores



No saved queries yet.

Start saving your aggregations and find queries, you'll see them here.

Not sure where to start? [Visit our Docs](#) →

```
>_MONGOSH
> use monitoramento_ambiental;
< switched to db monitoramento_ambiental
> db.createCollection("alertas");
db.createCollection("sensores");
db.createCollection("leituras");
db.createCollection("controle_irrigacoes");
db.createCollection("previsoes_chuva");
< { ok: 1 }
monitoramento_ambiental>
```

Figura 2

CRUD – Documentos na Collection Alertas

InsertMany

```
> db.alertas.insertMany([
  {
    tipo_alerta: "Alerta de Qualidade do Ar",
    descricao: "Poluição do ar acima do limite recomendado",
    localizacao: "Parque Ibirapuera",
    data_hora: new ISODate("2024-08-10T16:54:37.146Z")
  },
  {
    tipo_alerta: "Alerta de Umidade do Solo",
    descricao: "Umidade do solo abaixo do limite",
    localizacao: "Parque Villa-Lobos",
    data_hora: new ISODate("2024-08-10T17:00:00.000Z")
  }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66b7dd43f1c061b2b27b66f2'),
    '1': ObjectId('66b7dd43f1c061b2b27b66f3')
  }
}
monitoramento_ambiental>
```

Figura 3

Find

```
> db.alertas.find({}).pretty()
< {
  _id: ObjectId('66b7dd43f1c061b2b27b66f2'),
  tipo_alerta: 'Alerta de Qualidade do Ar',
  descricao: 'Poluição do ar acima do limite recomendado',
  localizacao: 'Parque Ibirapuera',
  data_hora: 2024-08-10T16:54:37.146Z
}
{
  _id: ObjectId('66b7dd43f1c061b2b27b66f3'),
  tipo_alerta: 'Alerta de Umidade do Solo',
  descricao: 'Umidade do solo abaixo do limite',
  localizacao: 'Parque Villa-Lobos',
  data_hora: 2024-08-10T17:00:00.000Z
}
monitoramento_ambiental>
```

Figura 4

FindOne

```
> db.alertas.findOne({ _id: ObjectId("66b7dd43f1c061b2b27b66f2") })
< {
  _id: ObjectId('66b7dd43f1c061b2b27b66f2'),
  tipo_alerta: 'Alerta de Qualidade do Ar',
  descricao: 'Poluição do ar acima do limite recomendado',
  localizacao: 'Parque Ibirapuera',
  data_hora: 2024-08-10T16:54:37.146Z
}
monitoramento_ambiental>
```

Figura 5

UpdateOne

```
> db.alertas.updateOne(
  { _id: ObjectId("66b7dd43f1c061b2b27b66f2") },
  {
    $set: {
      tipo_alerta: "Alerta de Temperatura",
      descricao: "Temperatura elevada",
      localizacao: "Parque Ibirapuera",
      data_hora: new ISODate("2024-08-11T13:39:47Z")
    }
  }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
monitoramento_ambiental>
```

Figura 6

DeleteOne

```
> db.alertas.deleteOne({ _id: ObjectId("66b7dd43f1c061b2b27b66f3") })
< {
  acknowledged: true,
  deletedCount: 1
}
monitoramento_ambiental>
```

Figura 7

CRUD – Documentos na Collection Sensores

InsertMany

```
> db.sensores.insertMany([
  {
    cod_sensor: 1,
    tipo_sensor: "Sensor de Temperatura",
    localizacao: "Parque Villa-Lobos"
  },
  {
    cod_sensor: 2,
    tipo_sensor: "Sensor de Umidade do Solo",
    localizacao: "Parque Ibirapuera"
  }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66b7ddeaf1c061b2b27b66f4'),
    '1': ObjectId('66b7ddeaf1c061b2b27b66f5')
  }
}
monitoramento_ambiental>
```

Figura 8

Find

```
> db.sensores.find({}).pretty()
< {
  _id: ObjectId('66b7ddeafc061b2b27b66f4'),
  cod_sensor: 1,
  tipo_sensor: 'Sensor de Temperatura',
  localizacao: 'Parque Villa-Lobos'
}
{
  _id: ObjectId('66b7ddeafc061b2b27b66f5'),
  cod_sensor: 2,
  tipo_sensor: 'Sensor de Umidade do Solo',
  localizacao: 'Parque Ibirapuera'
}
monitoramento_ambiental>
```

Figura 9

FindOne

```
> db.sensores.findOne({ _id: ObjectId("66b7ddeafc061b2b27b66f5") })
< {
  _id: ObjectId('66b7ddeafc061b2b27b66f5'),
  cod_sensor: 2,
  tipo_sensor: 'Sensor de Umidade do Solo',
  localizacao: 'Parque Ibirapuera'
}
monitoramento_ambiental>
```

Figura 10

UpdateOne

```
> db.sensores.updateOne(
  { _id: ObjectId("66b7ddeafc061b2b27b66f5") },
  {
    $set: {
      cod_sensor: 2,
      tipo_sensor: "Sensor de Umidade do Solo",
      localizacao: "Parque Villa-Lobos"
    }
  }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
monitoramento_ambiental>
```

Figura 11

DeleteOne

```
> db.sensores.deleteOne({ _id: ObjectId("66b7ddeafc061b2b27b66f5") })
< {
  acknowledged: true,
  deletedCount: 1
}
monitoramento_ambiental>
```

Figura 12

CRUD – Documentos na Collection Leituras

InsertMany

```
> db.leituras.insertMany([
  {
    valor: 25.6,
    data_hora: new ISODate("2024-08-10T16:54:37.146Z"),
    cod_sensor: 1 // Obs: inserir um código sensor válido
  },
  {
    valor: 15.2,
    data_hora: new ISODate("2024-08-10T17:00:00.000Z"),
    cod_sensor: 2 // obs: inserir um código sensor válido
  }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66b7de63f1c061b2b27b66f6'),
    '1': ObjectId('66b7de63f1c061b2b27b66f7')
  }
}
monitoramento_ambiental>
```

Figura 13

Find

```
> db.leituras.find({}).pretty()
< {
  _id: ObjectId('66b7de63f1c061b2b27b66f6'),
  valor: 25.6,
  data_hora: 2024-08-10T16:54:37.146Z,
  cod_sensor: 1
}
{
  _id: ObjectId('66b7de63f1c061b2b27b66f7'),
  valor: 15.2,
  data_hora: 2024-08-10T17:00:00.000Z,
  cod_sensor: 2
}
monitoramento_ambiental>
```

Figura 14

FindOne

```
> db.leituras.findOne({ _id: ObjectId("66b7de63f1c061b2b27b66f6") })
< {
  _id: ObjectId('66b7de63f1c061b2b27b66f6'),
  valor: 25.6,
  data_hora: 2024-08-10T16:54:37.146Z,
  cod_sensor: 1
}
monitoramento_ambiental>
```

Figura 15

UpdateOne

```
> db.leituras.updateOne(
  { _id: ObjectId("66b7de63f1c061b2b27b66f7") },
  {
    $set: {
      valor: 25.6,
      data_hora: new ISODate("2024-08-10T17:00:00Z"),
      cod_sensor: 2
    }
  }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
monitoramento_ambiental>
```

Figura 16

DeleteOne

```
> db.leituras.deleteOne({ _id: ObjectId("66b7de63f1c061b2b27b66f6") })
< {
  acknowledged: true,
  deletedCount: 1
}
monitoramento_ambiental>
```

Figura 17

CRUD – Documentos na Collection Previsões de Chuva

InsertMany

```
>_MONGOOSH
> db.previsoes_chuva.insertMany([
  {
    data_hora: new ISODate("2024-08-10T16:54:37.146Z"),
    previsao_chuva: "Sem Chuva"
  },
  {
    data_hora: new ISODate("2024-08-10T17:00:00.000Z"),
    previsao_chuva: "Chuva"
  }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66b7dee0f1c061b2b27b66f8'),
    '1': ObjectId('66b7dee0f1c061b2b27b66f9')
  }
}
```

Figura 18

Find

```
> db.previsoes_chuva.find({}).pretty()
< {
  _id: ObjectId('66b7dee0f1c061b2b27b66f8'),
  data_hora: 2024-08-10T16:54:37.146Z,
  previsao_chuva: 'Sem Chuva'
}
{
  _id: ObjectId('66b7dee0f1c061b2b27b66f9'),
  data_hora: 2024-08-10T17:00:00.000Z,
  previsao_chuva: 'Chuva'
}
monitoramento_ambiental>
```

Figura 19

FindOne

```
> db.previsoes_chuva.findOne({ _id: ObjectId("66b7dee0f1c061b2b27b66f8") })
< {
  _id: ObjectId('66b7dee0f1c061b2b27b66f8'),
  data_hora: 2024-08-10T16:54:37.146Z,
  previsao_chuva: 'Sem Chuva'
}
monitoramento_ambiental>
```

Figura 20

UpdateOne

```
> db.previsoes_chuva.updateOne(
  { _id: ObjectId("66b7dee0f1c061b2b27b66f8") },
  {
    $set: {
      data_hora: new ISODate("2024-08-11T16:54:37Z"),
      previsao: "Sem Chuva"
    }
  }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
monitoramento_ambiental>
```

Figura 21

DeleteOne

```
> db.previsoes_chuva.deleteOne({ _id: ObjectId("66b7dee0f1c061b2b27b66f9") })
< {
  acknowledged: true,
  deletedCount: 1
}
monitoramento_ambiental>
```

Figura 22

CRUD – Inserindo Documentos na Collection Controle Irrigações

InsertMany

```
>_MONGOSH
> db.controle_irrigacoes.insertMany([
  {
    localizacao: "Parque Ibirapuera",
    estado: "Ligado",
    data_hora: new ISODate("2024-08-10T16:54:37.146Z"),
    previsao_chuva_id: "63b6f54d8d95b97e3f8b4567", // obs: inserir um ObjectId válido
    previsao_chuva: {
      data_hora: new ISODate("2024-08-10T16:54:37.146Z"),
      previsao: "Sem Chuva"
    }
  },
  {
    localizacao: "Parque Villa-Lobos",
    estado: "Desligado",
    data_hora: new ISODate("2024-08-10T17:00:00.000Z"),
    previsao_chuva_id: "63b6f54d8d95b97e3f8b4568", // obs: inserir um ObjectId válido
    previsao_chuva: {
      data_hora: new ISODate("2024-08-10T17:00:00.000Z"),
      previsao: "Chuva"
    }
  }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66b7df6cfc061b2b27b66fa'),
    '1': ObjectId('66b7df6cfc061b2b27b66fb')
  }
}
monitoramento_ambiental>|
```

Figura 23

Find

```
> db.controle_irrigacoes.find({}).pretty()
< {
  _id: ObjectId('66b7df6cfc061b2b27b66fa'),
  localizacao: 'Parque Ibirapuera',
  estado: 'Ligado',
  data_hora: 2024-08-10T16:54:37.146Z,
  previsao_chuva_id: '63b6f54d8d95b97e3f8b4567',
  previsao_chuva: {
    data_hora: 2024-08-10T16:54:37.146Z,
    previsao: 'Sem Chuva'
  }
}
{
  _id: ObjectId('66b7df6cfc061b2b27b66fb'),
  localizacao: 'Parque Villa-Lobos',
  estado: 'Desligado',
  data_hora: 2024-08-10T17:00:00.000Z,
  previsao_chuva_id: '63b6f54d8d95b97e3f8b4568',
  previsao_chuva: {
    data_hora: 2024-08-10T17:00:00.000Z,
    previsao: 'Chuva'
  }
}
monitoramento_ambiental>|
```

Figura 24

FindOne

```
> db.controle_irrigacoes.findOne({ _id: ObjectId("66b7df6cf1c061b2b27b66fa") })
< {
  _id: ObjectId('66b7df6cf1c061b2b27b66fa'),
  localizacao: 'Parque Ibirapuera',
  estado: 'Ligado',
  data_hora: 2024-08-10T16:54:37.146Z,
  previsao_chuva_id: '63b6f54d8d95b97e3f8b4567',
  previsao_chuva: {
    data_hora: 2024-08-10T16:54:37.146Z,
    previsao: 'Sem Chuva'
  }
}
```

monitoramento_ambiental>

Figura 25

UpdateOne

```
> db.controle_irrigacoes.updateOne(
  { _id: ObjectId("66b7df6cf1c061b2b27b66fa") },
  {
    $set: {
      localizacao: "Parque Ibirapuera",
      estado: "Desligado",
      data_hora: new ISODate("2024-08-10T16:54:37Z"),
      previsao_chuva_id: "63b6f54d8d95b97e3f8b4567"
    }
  }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

monitoramento_ambiental>

Figura 26

DeleteOne

```
> db.controle_irrigacoes.deleteOne({ _id: ObjectId("66b7df6cf1c061b2b27b66fb") })
< {
  acknowledged: true,
  deletedCount: 1
}
```

monitoramento_ambiental>

Figura 27

Evidências das Collections criadas

DataBase e Collections criados no MongoDB

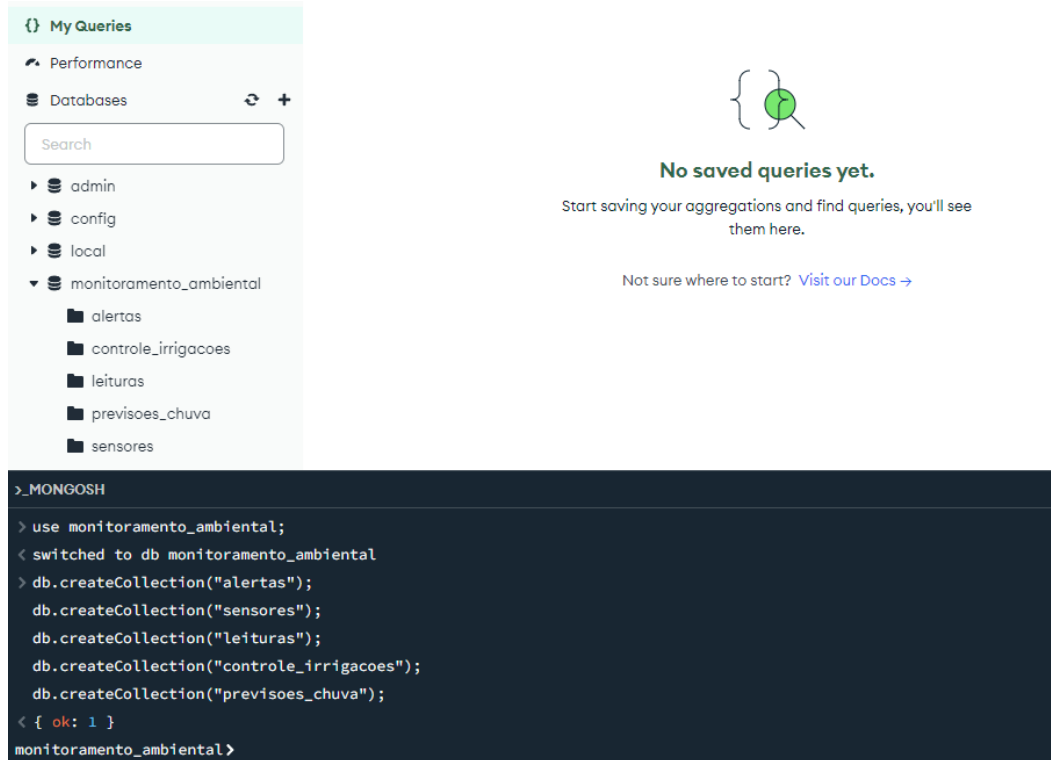


Figura 28

DataBase e Collections criados pela API Web Asp.NET

Configurações com MongoDB e Csharp e demais configurações das Services

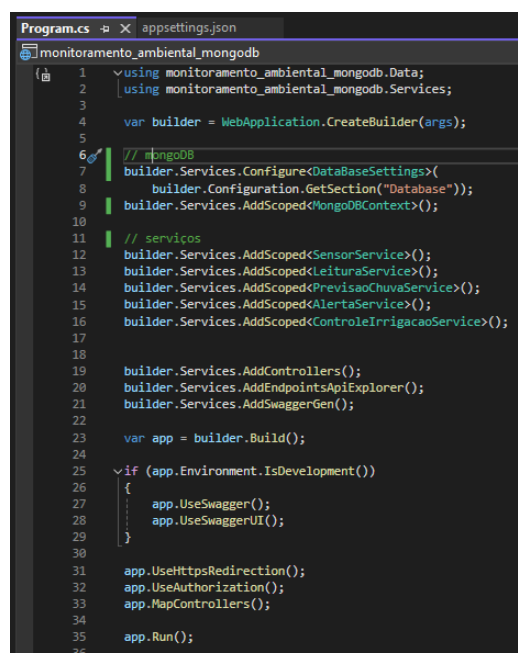
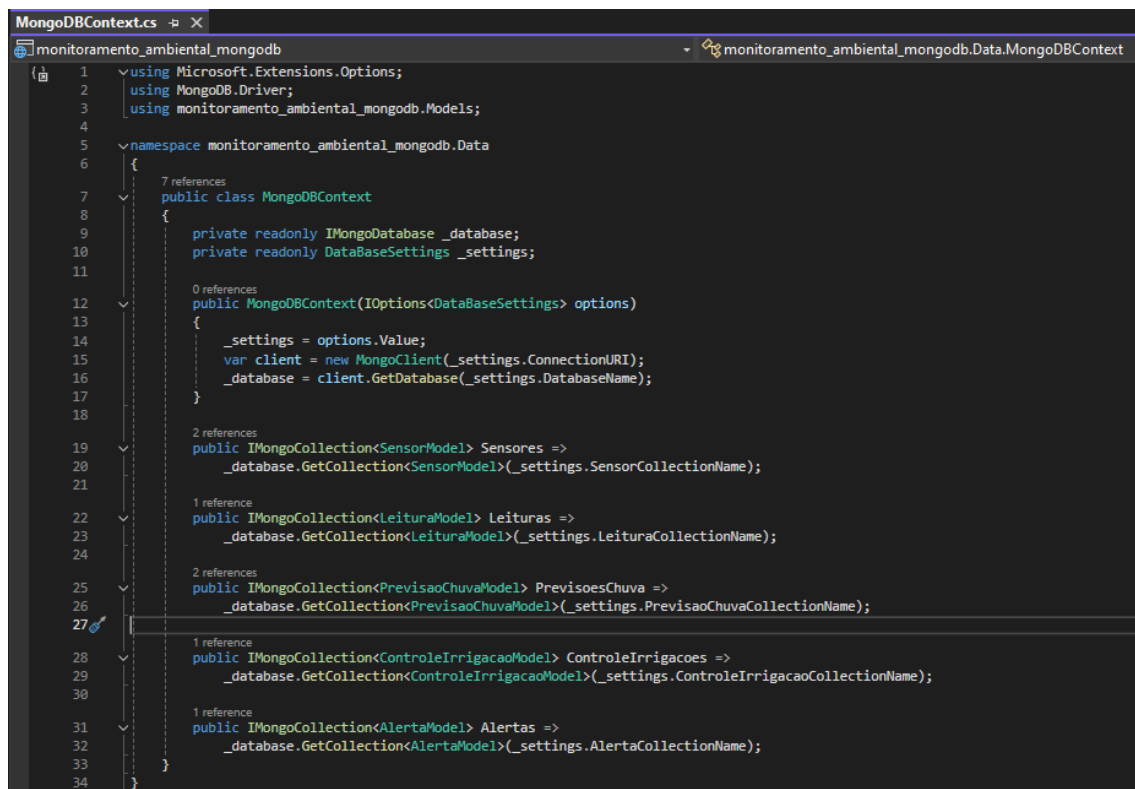


Figura 29

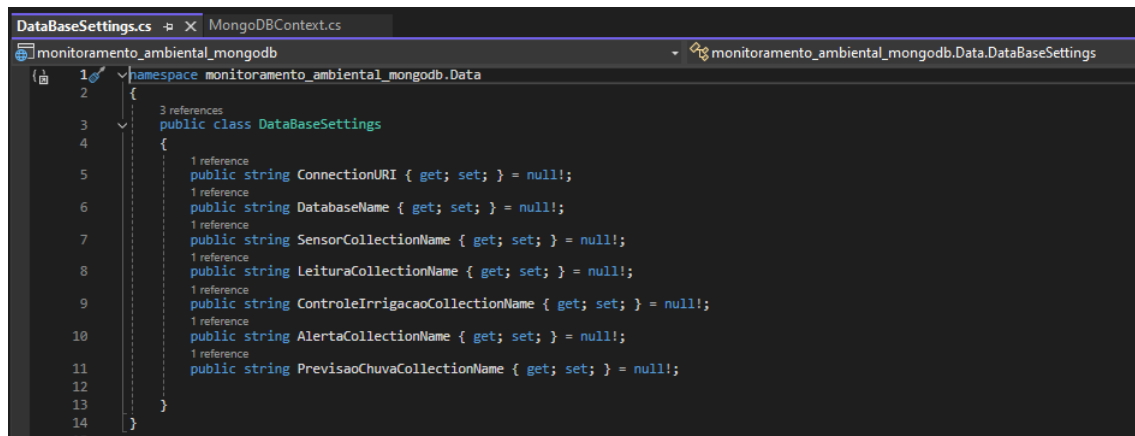
DataBase Context – Gerenciamento do DataBase e acesso as Collections



```
1 using Microsoft.Extensions.Options;
2 using MongoDB.Driver;
3 using monitoramento_ambiental_mongodb.Models;
4
5 namespace monitoramento_ambiental_mongodb.Data
6 {
7     7 references
8     public class MongoDBContext
9     {
10         private readonly IMongoDatabase _database;
11         private readonly DataBaseSettings _settings;
12
13         0 references
14         public MongoDBContext(IOption<DataBaseSettings> options)
15         {
16             _settings = options.Value;
17             var client = new MongoClient(_settings.ConnectionURI);
18             _database = client.GetDatabase(_settings.DatabaseName);
19
20         2 references
21         public IMongoCollection<SensorModel> Sensores =>
22             _database.GetCollection<SensorModel>(_settings.SensorCollectionName);
23
24         1 reference
25         public IMongoCollection<LeituraModel> leituras =>
26             _database.GetCollection<LeituraModel>(_settings.LeituraCollectionName);
27
28         2 references
29         public IMongoCollection<PrevisaoChuvaModel> PrevisoesChuva =>
30             _database.GetCollection<PrevisaoChuvaModel>(_settings.PrevisaoChuvaCollectionName);
31
32         1 reference
33         public IMongoCollection<ControleIrrigacaoModel> ControleIrigacoes =>
34             _database.GetCollection<ControleIrrigacaoModel>(_settings.ControleIrigacaoCollectionName);
35
36         1 reference
37         public IMongoCollection<AlertaModel> Alertas =>
38             _database.GetCollection<AlertaModel>(_settings.AlertCollectionName);
39     }
40 }
```

Figura 30

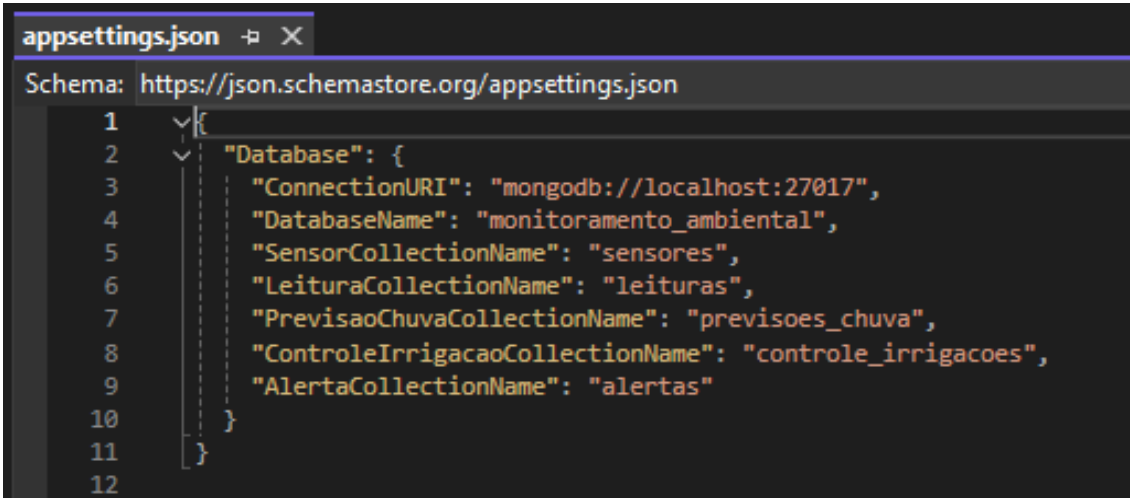
DataBaseSettings



```
1 namespace monitoramento_ambiental_mongodb.Data
2 {
3     3 references
4     public class DataBaseSettings
5     {
6         1 reference
7         public string ConnectionURI { get; set; } = null;
8         1 reference
9         public string DatabaseName { get; set; } = null;
10        1 reference
11        public string SensorCollectionName { get; set; } = null;
12        1 reference
13        public string LeituraCollectionName { get; set; } = null;
14        1 reference
15        public string ControleIrigacaoCollectionName { get; set; } = null;
16        1 reference
17        public string AlertaCollectionName { get; set; } = null;
18        1 reference
19        public string PrevisaoChuvaCollectionName { get; set; } = null;
20    }
21 }
```

Figura 31

Nome do DataBase e das Collections

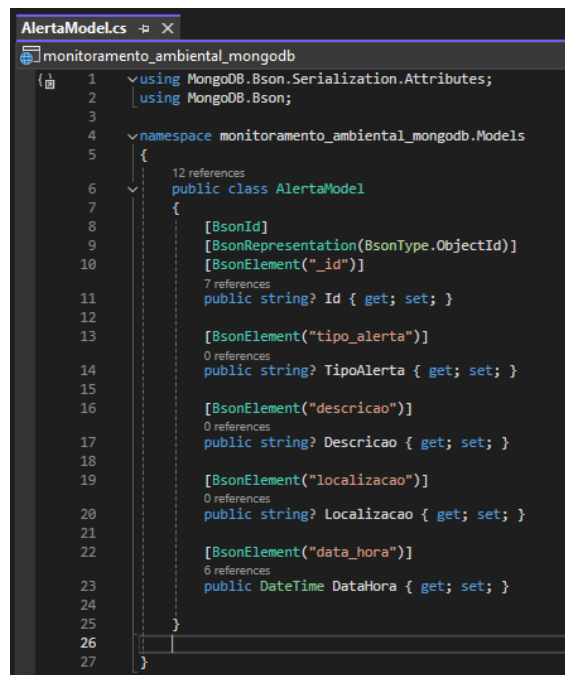


```
appsettings.json ✕
Schema: https://json.schemastore.org/appsettings.json
1  {
2    "Database": {
3      "ConnectionURI": "mongodb://localhost:27017",
4      "DatabaseName": "monitoramento_ambiental",
5      "SensorCollectionName": "sensores",
6      "LeituraCollectionName": "leituras",
7      "PrevisaoChuvaCollectionName": "previsoes_chuva",
8      "ControleIrrigacaoCollectionName": "controle_irrigacoes",
9      "AlertaCollectionName": "alertas"
10   }
11 }
12
```

Figura 32

Models / Collections

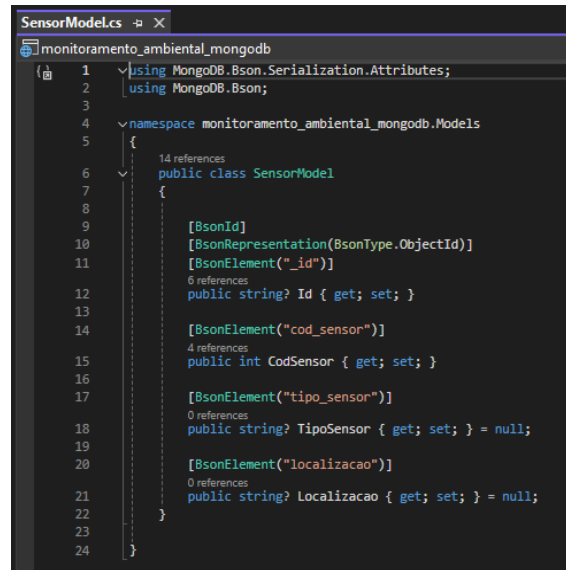
Alerta



```
AlertaModel.cs ✕
monitoramento_ambiental_mongodb
1  using MongoDB.Bson.Serialization.Attributes;
2  using MongoDB.Bson;
3
4  namespace monitoramento_ambiental_mongodb.Models
5  {
6      12 references
7      public class AlertaModel
8      {
9          [BsonId]
10         [BsonRepresentation(BsonType.ObjectId)]
11         [BsonElement("_id")]
12         public string? Id { get; set; }
13
14         [BsonElement("tipo_alerta")]
15         0 references
16         public string? TipoAlerta { get; set; }
17
18         [BsonElement("descricao")]
19         0 references
20         public string? Descricao { get; set; }
21
22         [BsonElement("localizacao")]
23         0 references
24         public string? Localizacao { get; set; }
25
26         [BsonElement("data_hora")]
27         6 references
28         public DateTime DataHora { get; set; }
29     }
30 }
```

Figura 33

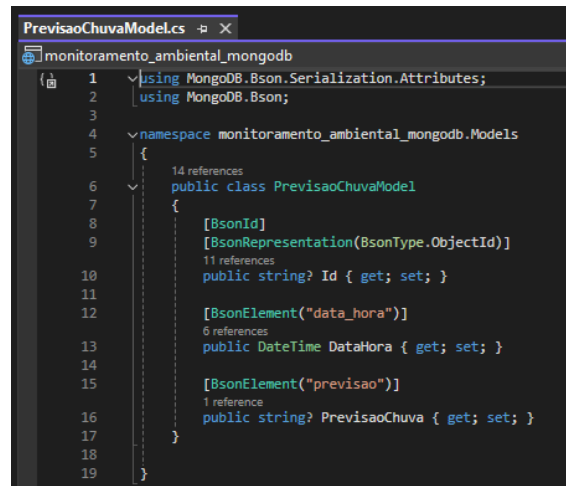
Sensor



```
1 using MongoDB.Bson.Serialization.Attributes;
2 using MongoDB.Bson;
3
4 namespace monitoramento_ambiental_mongodb.Models
5 {
6     public class SensorModel
7     {
8         [BsonId]
9         [BsonRepresentation(BsonType.ObjectId)]
10        [BsonElement("_id")]
11        public string? Id { get; set; }
12
13        [BsonElement("cod_sensor")]
14        public int CodSensor { get; set; }
15
16        [BsonElement("tipo_sensor")]
17        public string? TipoSensor { get; set; } = null;
18
19        [BsonElement("localizacao")]
20        public string? Localizacao { get; set; } = null;
21    }
22 }
```

Figura 34

Previsão de Chuva



```
1 using MongoDB.Bson.Serialization.Attributes;
2 using MongoDB.Bson;
3
4 namespace monitoramento_ambiental_mongodb.Models
5 {
6     public class PrevisaoChuvaModel
7     {
8         [BsonId]
9         [BsonRepresentation(BsonType.ObjectId)]
10        public string? Id { get; set; }
11
12        [BsonElement("data_hora")]
13        public DateTime DataHora { get; set; }
14
15        [BsonElement("previsao")]
16        public string? PrevisaoChuva { get; set; }
17    }
18 }
```

Figura 35

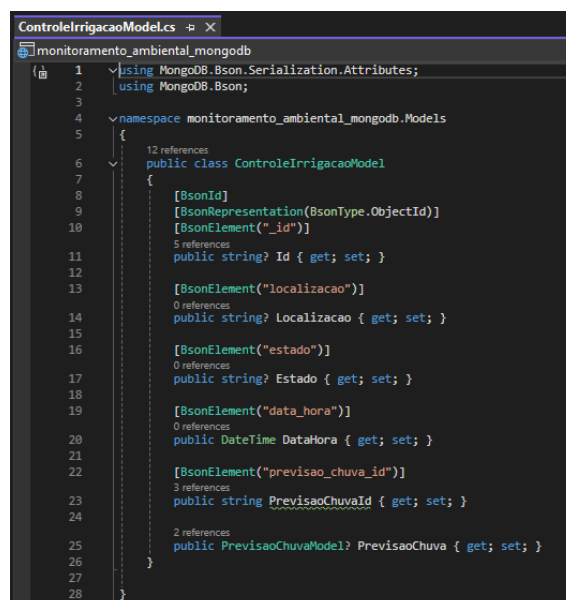
Leitura



```
1 using MongoDB.Bson.Serialization.Attributes;
2 using MongoDB.Bson;
3 using System.Text.Json.Serialization;
4 using Amazon.Auth.AccessControlPolicy;
5
6 namespace monitoramento_ambiental_mongodb.Models
7 {
8     12 references
9     public class LeituraModel
10     {
11         [BsonId]
12         [BsonRepresentation(BsonType.ObjectId)]
13         [BsonIgnoreIfDefault]
14         [BsonElement("_id")]
15         public string? Id { get; set; }
16         7 references
17         private decimal valor;
18         [BsonElement("valor")]
19         [BsonRepresentation(BsonType.Decimal128)]
20         0 references
21         public decimal Valor
22         {
23             get => valor;
24             set => valor = Math.Round(value, 2);
25         }
26         [BsonElement("data_hora")]
27         6 references
28         public DateTime DataHora { get; set; }
29         [BsonElement("cod_sensor")]
30         2 references
31         public int CodSensor { get; set; }
32         2 references
33         public SensorModel? Sensor { get; set; }
34     }
35 }
```

Figura 36

Controle de Irrigação



```
1 using MongoDB.Bson.Serialization.Attributes;
2 using MongoDB.Bson;
3
4 namespace monitoramento_ambiental_mongodb.Models
5 {
6     12 references
7     public class ControleIrrigacaoModel
8     {
9         [BsonId]
10         [BsonRepresentation(BsonType.ObjectId)]
11         [BsonElement("_id")]
12         5 references
13         public string? Id { get; set; }
14         [BsonElement("localizacao")]
15         0 references
16         public string? Localizacao { get; set; }
17         [BsonElement("estado")]
18         0 references
19         public string? Estado { get; set; }
20         [BsonElement("data_hora")]
21         0 references
22         public DateTime DataHora { get; set; }
23         [BsonElement("previsao_chuva_id")]
24         3 references
25         public string PrevisaoChuvaId { get; set; }
26         2 references
27         public PrevisaoChuvaModel? PrevisaoChuva { get; set; }
28     }
29 }
```

Figura 37

Evidencias da Manipulação dos Dados Via Swagger UI

Collection Sensores

Insert

POST /api/Sensor

Parameters

No parameters

Request body

application/json

```
{  "codSensor": 1,  "tipoSensor": "Sensor de Temperatura",  "localizacao": "Parque Villa-Lobos"}  
```

Execute

Clear

Responses

Curl

```
curl -X 'POST' \  'https://localhost:7072/api/Sensor' \  -H 'accept: text/plain' \  -H 'Content-Type: application/json' \  -d '{  "codSensor": 1,  "tipoSensor": "Sensor de Temperatura",  "localizacao": "Parque Villa-Lobos"}'  
```

Request URL

https://localhost:7072/api/Sensor

Server response

Code

Details

201

undocumented

Response body

```
{  "success": true,  "message": "Sensor cadastrado com sucesso",  "createdItem": {    "id": "66b7d86e1f6a8e72d73ad74c",    "codSensor": 1,    "tipoSensor": "Sensor de Temperatura",    "localizacao": "Parque Villa-Lobos"  }  }
```

Response headers

```
content-type: application/json; charset=utf-8  date: Sat, 10 Aug 2024 21:15:26 GMT  location: https://localhost:7072/api/Sensor/66b7d86e1f6a8e72d73ad74c  server: Kestrel
```

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{  "id": "string",  "codSensor": 1,  "tipoSensor": "string",  "localizacao": "string"}  
```

Figura 38

GetAll

GET /api/Sensor

Parameters

Cancel

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7072/api/Sensor' \
  -H 'accept: text/plain'
```

Request URL

https://localhost:7072/api/Sensor

Server response

Code

Details

200

Response body

```
{
  "id": "66b7d86c1f6a8c72d73ad74c",
  "codSensor": 1,
  "tipoSensor": "Sensor de Temperatura",
  "localizacao": "Parque Villa-Lobos"
},
{
  "id": "66b7d9261f6a8c72d73ad74d",
  "codSensor": 1,
  "tipoSensor": "Sensor de Temperatura",
  "localizacao": "Parque Thirapuera"
},
{
  "id": "66b7d9681f6a8c72d73ad74e",
  "codSensor": 1,
  "tipoSensor": "Sensor de Umidade do Solo",
  "localizacao": "Parque Thirapuera"
},
{
  "id": "66b7d97c1f6a8c72d73ad74f",
  "codSensor": 1,
  "tipoSensor": "Sensor de Umidade do Solo",
  "localizacao": "Parque Villa-Lobos"
}
}
```

Download

Response headers

```
content-type: application/json; charset=utf-8
date: Sat, 10 Aug 2024 21:15:59 GMT
server: Kestrel
```

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accepts header.

Example Value | Schema

```
{
  "id": "string",
  "codSensor": 1,
  "tipoSensor": "string",
  "localizacao": "string"
}
```

Figura 39

GetById

GET

/api/Sensor/{id}

^

Cancel

Name	Description
id <small>* required</small>	
string	66b7d9681f6a8e72d73ad74e
(path)	

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
'https://localhost:7072/api/Sensor/66b7d9681f6a8e72d73ad74e' \
-M 'Accept: text/plain'
```

Request URL

```
https://localhost:7072/api/Sensor/66b7d9681f6a8e72d73ad74e
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "sensor": { "id": "66b7d9681f6a8e72d73ad74e", "codSensor": 3, "tipSensor": "Sensor de Unidade do Solo", "localizacao": "Parque Ibirapuera" }, "message": "Sensor encontrado com sucesso." }</pre></div><div><div>Download</div></div></div> <div><div>Response headers</div><div><pre>content-type: application/json; charset=utf-8 date: Sat, 10 Aug 2024 21:20:53 GMT server: Kestrel</pre></div></div>

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{
  "id": "string",
  "codSensor": 0,
  "tipSensor": "string",
  "localizacao": "string"
}
```

Figura 40

Update

PUT

/api/Sensor/{id}

Cancel

Reset

Parameters

Name	Description
id <small>required</small>	
string	66b7d9681f6a8e72d73ad74e
(path)	

Request body

application/json

```
{  "id": "66b7d9681f6a8e72d73ad74e",  "codSensor": 3,  "tipoSensor": "Sensor Temperatura",  "localizacao": "Parque Villa-Lobos"}  
```

Execute

Clear

Responses

Curl

```
curl -X 'PUT' \  https://localhost:7072/api/Sensor/66b7d9681f6a8e72d73ad74e' \  -H 'accept: */*' \  -H 'Content-Type: application/json' \  -d '{  "id": "66b7d9681f6a8e72d73ad74e",  "codSensor": 3,  "tipoSensor": "Sensor Temperatura",  "localizacao": "Parque Villa-Lobos"}'  
```

Request URL

https://localhost:7072/api/Sensor/66b7d9681f6a8e72d73ad74e

Server response

Code	Details
201	<div><div>Response body</div><div><pre>{ "success": true, "message": "Sensor atualizado com sucesso", "createdItem": { "id": "66b7d9681f6a8e72d73ad74e", "codSensor": 3, "tipoSensor": "Sensor Temperatura", "localizacao": "Parque Villa-Lobos" } } </pre></div><div>Download</div></div>

Response headers

```
content-type: application/json; charset=utf-8  date: Sat, 10 Aug 2024 21:22:48 GMT  location: https://localhost:7072/api/Sensor/66b7d9681f6a8e72d73ad74e  server: Kestrel  
```

Responses

Code	Description	Links
200	Success	No links

Figura 41

Delete

DELETE

/api/Sensor/{id}

^

Parameters

Cancel

Name	Description
id * required string (path)	<div>66b7d9681f6a8e72d73ad74e</div>

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
'https://localhost:7072/api/Sensor/66b7d9681f6a8e72d73ad74e' \
-H 'accept: */*'
```

Request URL

```
https://localhost:7072/api/Sensor/66b7d9681f6a8e72d73ad74e
```

Server response

Code	Details
200	<div><div>Response body</div><div>Sensor removido com sucesso.</div><div><div>Download</div></div></div> <div><div>Response headers</div><div>content-type: text/plain; charset=utf-8 date: Sat, 18 Aug 2024 21:25:47 GMT server: Kestrel</div></div>

Responses

Code	Description	Links
200	Success	No links

Figura 42

Collection Alertas

Insert

POST /api/Alerta

Parameters

CancelReset

No parameters

Request body

application/json

```
{  "tipoAlerta": "Alerta de Unidade do Solo",  "descricao": "Umidade baixa",  "localizacao": "Parque Villa-Lobos",  "dataHora": "2024-08-12T12:31:31.130Z"}}
```

ExecuteClear

Responses

Curl

```
curl -X 'POST' \  'https://localhost:7072/api/Alerta' \  -H 'Accept: text/plain' \  -H 'Content-Type: application/json' \  -d '{  "tipoAlerta": "Alerta de Unidade do Solo",  "descricao": "Umidade baixa",  "localizacao": "Parque Villa-Lobos",  "dataHora": "2024-08-12T12:31:31.130Z"}'
```

Request URL

https://localhost:7072/api/Alerta

Server response

CodeDetails

201

Undo/Commented

Response body

```
{  "success": true,  "message": "Alerta cadastrado com sucesso",  "createdItem": {    "id": "66ba00dfe19c5d1838de011",    "tipoAlerta": "Alerta de Unidade do Solo",    "descricao": "Umidade baixa",    "localizacao": "Parque Villa-Lobos",    "dataHora": "2024-08-12T09:32:26.6496583"}  }}
```

Download

Response headers

```
content-type: application/json; charset=utf-8date: Mon, 12 Aug 2024 12:32:26 GMTlocation: https://localhost:7072/api/Alerta/66ba00dfe19c5d1838de011server: Kestrel
```

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

```
{  "id": "string",  "tipoAlerta": "string",  "descricao": "string",  "localizacao": "string",  "dataHora": "2024-08-12T12:32:26.700Z"}}
```

Figura 43

GET

/api/Alerta

Parameters

Cancel

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'GET' \  
  'https://localhost:7072/api/Alerta' \  
  -M 'accept: text/plain'
```

Request URL

https://localhost:7072/api/Alerta

Server response

Code

Details

200

Response body

```
{  
  {  
    "id": "66b7d443f1c861b2b27b66f2",  
    "tipoAlerta": "Alerta de Temperatura",  
    "descricao": "Temperatura elevada",  
    "localizacao": "Parque Thirapuera",  
    "dataHora": "2024-08-11T18:35:47"  
  },  
  {  
    "id": "66ba00d4fe19c5d1838de011",  
    "tipoAlerta": "Alerta de Umidade do Solo",  
    "descricao": "Umidade baixa",  
    "localizacao": "Parque Villa-Lobos",  
    "dataHora": "2024-08-12T09:32:26.649"  
  }  
}
```

Response headers

```
content-type: application/json; charset=utf-8  
date: Mon, 12 Aug 2024 12:33:22 GMT  
server: Kestrel
```

Responses

Code

Description

Links

200

Success

No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

```
{  
  {  
    "id": "string",  
    "tipoAlerta": "string",  
    "descricao": "string",  
    "localizacao": "string",  
    "dataHora": "2024-08-12T12:33:23.248Z"  
  }  
}
```

Figura 44

GetById

GET /api/Alerta/{id}

Cancel

Parameters

Name	Description
id <small>* required</small>	
string (path)	66ba00dafa19c5d1838de011

ExecuteClear

Responses

Curl

curl -X 'GET' \nhttps://localhost:7072/api/Alerta/66ba00dafa19c5d1838de011' \n-M 'accept: text/plain'

Request URL

https://localhost:7072/api/Alerta/66ba00dafa19c5d1838de011

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "alerta": { "id": "66ba00dafa19c5d1838de011", "tipoAlerta": "Alerta de Unidade do Solo", "descricao": "Unidade baixa", "localizacao": "Parque Villa-Lobos", "dataHora": "2024-08-12T09:32:26.649" }, "message": "Alerta encontrado com sucesso." }</pre></div><div>Download</div></div> <div><div>Response headers</div><div>content-type: application/json; charset=utf-8 date: Mon, 12 Aug 2024 12:34:01 GMT server: Kestrel</div></div>

Responses

Code	Description	Links
200	SUCCESS	No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

```
{  "id": "string",  "tipoAlerta": "string",  "descricao": "string",  "localizacao": "string",  "dataHora": "2024-08-12T12:34:02.403Z"}
```

Figura 45

Update

PUT

/api/Alerta/{id}

Cancel

Reset

Parameters

Name	Description
id * required	
string	66ba00dfe19c5d1838de011
(path)	

Request body

application/json

```
{  "id": "66ba00dfe19c5d1838de011",  "tipoAlerta": "Alerta de Unidade do Solo",  "descricao": "Unidade Baixa",  "localizacao": "Parque Ibirapuera",  "dataHora": "2024-08-12T12:35:31.998Z"}

```

Execute

Clear

Responses

Curl

```
curl -X 'PUT' \
  'https://localhost:7072/api/Alerta/66ba00dfe19c5d1838de011' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": "66ba00dfe19c5d1838de011",
    "tipoAlerta": "Alerta de Unidade do Solo",
    "descricao": "Unidade Baixa",
    "localizacao": "Parque Ibirapuera",
    "dataHora": "2024-08-12T12:35:31.998Z"
  }'

```

Request URL

https://localhost:7072/api/Alerta/66ba00dfe19c5d1838de011

Server response

Code	Details
204	<div>Response headers<div>date: Mon, 12 Aug 2024 12:36:18 GMTserver: Kestrel</div></div>

Responses

Code	Description	Links
200	Success	No links

Figura 46

Delete

DELETE

/api/Alerta/{id}

^

Parameters

Cancel

Name	Description
id * required string (path)	66ba00dfe19c5d1838de011

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
  "https://localhost:7072/api/Alerta/66ba00dfe19c5d1838de011" \
  -H 'Accept: */*'
```

Request URL

https://localhost:7072/api/Alerta/66ba00dfe19c5d1838de011

Server response

Code	Details
200	<div><div>Response body</div><div>Alerta removido com sucesso.</div><div><div>Download</div></div></div> <div><div>Response headers</div><div>content-type: text/plain; charset=utf-8 date: Mon, 12 Aug 2024 12:36:59 GMT server: Kestrel</div></div>

Responses

Code	Description	Links
200	Success	No links

Figura 47

Collection Leituras

Insert

POST/api/Leitura

Parameters

CancelReset

No parameters

Request body

application/json

```
{
  "valor": 25.8,
  "datahora": "2024-08-11T14:57:24.835Z",
  "codSensor": 2,
  "sensor": {
    "id": "string",
    "codSensor": 0,
    "tipoSensor": "string",
    "localizacao": "string"
  }
}
```

ExecuteClear

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7072/api/Leitura' \
  -H 'accept: text/plain' \
  -H 'Content-Type: application/json' \
  -d '{
    "valor": 25.8,
    "datahora": "2024-08-11T14:57:24.835Z",
    "codSensor": 2,
    "sensor": {
      "id": "string",
      "codSensor": 0,
      "tipoSensor": "string",
      "localizacao": "string"
    }
  }'
```

Request URL

https://localhost:7072/api/Leitura

Server response

CodeDetails

201

undocumented

Response body

```
{
  "success": true,
  "message": "Leitura criada com sucesso",
  "createdItem": {
    "id": "66b8d170df46f00a657e16df",
    "valor": 25.8,
    "datahora": "2024-08-11T11:57:52.4966515",
    "codSensor": 2,
    "sensor": {
      "id": "66b8d170df46f00a657e16df",
      "codSensor": 2,
      "tipoSensor": "Sensor Unidade do Solo",
      "localizacao": "Parque Villa-Lobos"
    }
  }
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Sun, 11 Aug 2024 14:57:51 GMT
location: https://localhost:7072/api/Leitura/66b8d170df46f00a657e16df
server: Kestrel
```

Response

CodeDescriptionLinks

200

Success

No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{
  "id": "string",
  "valor": 0,
  "datahora": "2024-08-11T14:57:52.508Z",
  "codSensor": 0,
  "sensor": {
    "id": "string",
    "codSensor": 0,
    "tipoSensor": "string",
    "localizacao": "string"
  }
}
```

Figura 48

GetAll

GET /api/Leitura

Parameters

Cancel

No parameters

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
  https://localhost:7072/api/Leitura \
  -H 'accept: text/plain'
```

Request URL

https://localhost:7072/api/Leitura

Server response

CodeDetails

200

Response body

```
{
  "id": "66bdcdd8006c130e05956e",
  "valor": 10.0,
  "datahora": "2024-08-11T11:42:53.520",
  "codSensor": 1,
  "sensor": {
    "id": "6d3d4deaf1c861b2b27b66f4",
    "codSensor": 1,
    "tipoSensor": "Sensor de Temperatura",
    "localizacao": "Parque Villa-Lobos"
  }
},
{
  "id": "66bdcdd8006c130e05956e",
  "valor": 20.0,
  "datahora": "2024-08-11T11:57:52.494",
  "codSensor": 1,
  "sensor": {
    "id": "6d3d4deaf1c861b2b27b66f4",
    "codSensor": 1,
    "tipoSensor": "Sensor de Temperatura",
    "localizacao": "Parque Villa-Lobos"
  }
},
{
  "id": "66bdcdd8006c130e05956e",
  "valor": 30.0,
  "datahora": "2024-08-11T12:07:52.494",
  "codSensor": 1,
  "sensor": {
    "id": "6d3d4deaf1c861b2b27b66f4",
    "codSensor": 1,
    "tipoSensor": "Sensor de Temperatura",
    "localizacao": "Parque Villa-Lobos"
  }
}
}
```

Download

Response headers

```
content-type: application/json; charset=utf-8
date: Sun, 11 Aug 2024 14:58:38 GMT
server: Kestrel
```

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{
  "id": "string",
  "valor": 0,
  "datahora": "2024-08-11T14:58:39.299Z",
  "codSensor": 0,
  "sensor": {
    "id": "string",
    "codSensor": 0,
    "tipoSensor": "string",
    "localizacao": "string"
  }
}
```

Figura 49

GET /api/Leitura/{id}

Parameters

Cancel

Name	Description
id <small>required</small> string (path)	66b8d170df46f00a657e16df

ExecuteClear

Responses

Curl

curl -X 'GET' \
'https://localhost:7072/api/Leitura/66b8d170df46f00a657e16df' \
'-H 'accept: text/plain'

Request URL

https://localhost:7072/api/Leitura/66b8d170df46f00a657e16df

Server response

Code	Details
200	<div>Response body</div> <div><pre>{ "Leitura": { "id": "66b8d170df46f00a657e16df", "valor": 25.0, "dataHora": "2024-08-11T11:57:52.494", "codSensor": 2, "sensor": { "id": "66b8cf0dd46f00a657e16dd", "codSensor": 2, "tipoSensor": "Sensor Unidade do Solo", "localizacao": "Parque Villa-Lobos" } }, "message": "Leitura encontrada com sucesso." }</pre></div> <div>Download</div>

Response headers

```
content-type: application/json; charset=utf-8  
date: Sun, 11 Aug 2024 14:59:27 GMT  
server: Kestrel
```

Responses

Code	Description	Links
200	Success	No links

Media type
text/plain

Controls Accept header.

Example Value | Schema

```
{  
  "id": "string",  
  "valor": 0,  
  "dataHora": "2024-08-11T14:59:38.378Z",  
  "codSensor": 0,  
  "sensor": {  
    "id": "string",  
    "codSensor": 0,  
    "tipoSensor": "string",  
    "localizacao": "string"  
  }  
}
```

Figura 50

Update

PUT

/api/Leitura/{id}

Cancel

Reset

Parameters

Name	Description
id <small>* required</small>	
string	66b8cded8006e139e059569c
(path)	

Request body

application/json

```
{
  "id": "66b8cded8006e139e059569c",
  "valor": 18.9,
  "dataHora": "2024-08-11T15:00:21.637Z",
  "codSensor": 1,
  "sensor": {
    "id": "string",
    "codSensor": 0,
    "tipoSensor": "string",
    "localizacao": "string"
  }
}
```

Execute

Clear

Responses

Curl

```
curl -X 'PUT' \
  'https://localhost:7072/api/Leitura/66b8cded8006e139e059569c' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": "66b8cded8006e139e059569c",
    "valor": 18.9,
    "dataHora": "2024-08-11T15:00:21.637Z",
    "codSensor": 1,
    "sensor": {
      "id": "string",
      "codSensor": 0,
      "tipoSensor": "string",
      "localizacao": "string"
    }
  }'
```

Request URL

https://localhost:7072/api/Leitura/66b8cded8006e139e059569c

Server response

Code

Details

201

undocumented

Response body

```
{
  "success": true,
  "message": "Leitura atualizada com sucesso.",
  "createdItem": {
    "id": "66b8cded8006e139e059569c",
    "valor": 18.9,
    "dataHora": "2024-08-11T12:00:56.8897404",
    "codSensor": 1,
    "sensor": {
      "id": "66b7ddea1c061b2b27b6644",
      "codSensor": 1,
      "tipoSensor": "Sensor de Temperatura",
      "localizacao": "Parque Villa-Lobos"
    }
  }
}
```

Download

Response headers

```
content-type: application/json; charset=utf-8
date: Sun, 11 Aug 2024 15:00:56 GMT
location: https://localhost:7072/api/Leitura/66b8cded8006e139e059569c
server: Kestrel
```

Responses

Code	Description	Links
200	Success	No links

Figura 51

Delete

DELETE

/api/leitura/{id}

⌵

Parameters

Cancel

Name	Description
id <small>* required</small>	
string	66b8cded8006e139e059569c
<small>(path)</small>	

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
  https://localhost:7072/api/leitura/66b8cded8006e139e059569c' \
  -H 'accept: */*'
```

Request URL

https://localhost:7072/api/leitura/66b8cded8006e139e059569c

Server response

Code	Details
200	<div><div>Response body</div><div>leitura removida com sucesso.</div><div><div>Download</div></div></div> <div><div>Response headers</div><div>content-type: text/plain; charset=utf-8 date: Sun, 11 Aug 2024 15:01:50 GMT server: Kestrel</div></div>

Responses

Code	Description	Links
200	Success	No links

Figura 52

Collection Previsões Chuva

Insert

POST /api/PrevisaoChuva

Parameters

No parameters

Request body

application/json

```
{
  "dataHora": "2024-08-11T15:09:43.473Z",
  "previsaoChuva": "Com Chuva"
}
```

Execute

Clear

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7072/api/PrevisaoChuva' \
  -H 'accept: text/plain' \
  -H 'Content-Type: application/json' \
  -d '{
    "dataHora": "2024-08-11T15:09:43.473Z",
    "previsaoChuva": "Com Chuva"
  }'
```

Request URL

https://localhost:7072/api/PrevisaoChuva

Server response

Code

Details

201

Undocumented

Response body

```
{
  "success": true,
  "message": "Previsão de Chuva cadastrada com sucesso",
  "createdItem": {
    "id": "66b8d445df46f0ba57e16e0",
    "dataHora": "2024-08-11T15:09:57.313926",
    "previsaoChuva": "Com Chuva"
  }
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Sun, 11 Aug 2024 15:09:56 GMT
location: https://localhost:7072/api/PrevisaoChuva/66b8d445df46f0ba57e16e0
server: Kestrel
```

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

Example Value

Schema

```
{
  "id": "string",
  "dataHora": "2024-08-11T15:09:57.338Z",
  "previsaoChuva": "string"
}
```

Figura 53

GetAll

GET/api/PrevisaoChuva

Parameters

Cancel

No parameters

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
'https://localhost:7072/api/PrevisaoChuva' \
-H 'accept: text/plain'
```

Request URL

```
https://localhost:7072/api/PrevisaoChuva
```

Server response

CodeDetails

200

Response body

```
{
  "id": "66b8d445df46f90a657e16e0",
  "dataHora": "2024-08-11T12:09:57.313",
  "previsaoChuva": "Com Chuva"
},
{
  "id": "66b8d49adf46f90a657e16e1",
  "dataHora": "2024-08-11T12:11:22.235",
  "previsaoChuva": "Sem Chuva"
},
{
  "id": "66b8d4a6df46f90a657e16e2",
  "dataHora": "2024-08-11T12:11:34.805",
  "previsaoChuva": "Com Chuva"
}
]
```

Response headers

```
content-type: application/json; charset=utf-8
date: Sun, 11 Aug 2024 15:11:38 GMT
server: Kestrel
```

Responses

CodeDescriptionLinks

200

Success

No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{
  "id": "string",
  "dataHora": "2024-08-11T15:11:38.914Z",
  "previsaoChuva": "string"
}
```

Figura 54

GET

/api/PrevisaoChuva/{id}

⌵

Parameters

Cancel

Name	Description
id required string (path)	<div>66b8d49adf46f00a657e16e1</div>

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7072/api/PrevisaoChuva/66b8d49adf46f00a657e16e1' \
  -H 'accept: text/plain'
```

Request URL

```
https://localhost:7072/api/PrevisaoChuva/66b8d49adf46f00a657e16e1
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "previsaoChuva": { "id": "66b8d49adf46f00a657e16e1", "dataHora": "2024-08-11T12:11:22.235", "previsaoChuva": "Sem Chuva" }, "message": "Previs\u00e3o de Chuva encontrada com sucesso." }</pre></div><div><div>Download</div></div></div> <div><div>Response headers</div><div><pre>content-type: application/json; charset=utf-8 date: Sun, 11 Aug 2024 15:12:42 GMT server: Kestrel</pre></div></div>

Responses

Code	Description	Links
200	SUCCESS	No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

```
{
  "id": "string",
  "dataHora": "2024-08-11T15:12:43.686Z",
  "previsaoChuva": "string"
}
```

Figura 55

Update

PUT

/api/PrevisaoChuva/{id}

Parameters

Cancel

Reset

Name	Description
id <small>* required</small>	
string (path)	66b8d49adf46f00a657e16e1

Request body

application/json

```
{
  "id": "66b8d49adf46f00a657e16e1",
  "dataHora": "2024-08-11T15:13:22.565Z",
  "previsaoChuva": "Com Chuva"
}
```

Execute

Clear

Responses

Curl

```
curl -X 'PUT' \
  'https://localhost:7072/api/PrevisaoChuva/66b8d49adf46f00a657e16e1' \
  -H 'Accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": "66b8d49adf46f00a657e16e1",
    "dataHora": "2024-08-11T15:13:22.565Z",
    "previsaoChuva": "Com Chuva"
  }'
```

Request URL

https://localhost:7072/api/PrevisaoChuva/66b8d49adf46f00a657e16e1

Server response

Code	Details
201	<div><div>Response body</div><div><pre>{ "success": true, "message": "Previs\u00e3o de Chuva atualizada com sucesso", "createdItem": { "id": "66b8d49adf46f00a657e16e1", "dataHora": "2024-08-11T12:13:35.6421889", "previsaoChuva": "Com Chuva" } }</pre></div><div><div>Response headers</div><div><pre>content-type: application/json; charset=utf-8 date: Sun, 11 Aug 2024 15:13:35 GMT location: https://localhost:7072/api/PrevisaoChuva/66b8d49adf46f00a657e16e1 server: Kestrel</pre></div></div><div><div>Download</div></div></div>

Responses

Code	Description	Links
200	Success	No links

Figura 56

Delete

DELETE

/api/PrevisaoChuva/{id}

^

Parameters

Cancel

Name	Description
id * required string (path)	<div>66b8d49adf46f00a657e16e1</div>

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
'https://localhost:7072/api/PrevisaoChuva/66b8d49adf46f00a657e16e1' \
-H 'accept: */*'
```

Request URL

```
https://localhost:7072/api/PrevisaoChuva/66b8d49adf46f00a657e16e1
```

Server response

Code	Details
200	<div><div>Response body</div><div>Previsão de Chuva removida com sucesso.</div><div><div>Download</div></div></div> <div><div>Response headers</div><div>content-type: text/plain; charset=utf-8 date: Sun, 11 Aug 2024 15:14:34 GMT server: Kestrel</div></div>

Responses

Code	Description	Links
200	Success	No links

Figura 57

Collection Controle de Irrigações

Insert

POST/api/controleIrrigacao

Parameters

CancelReset

No parameters

Request body

application/json

```
{  "localizacao": "Parque Ibirapuera",  "estado": "ligado",  "datahora": "2024-08-11T15:15:34.227Z",  "previsaoChuvaId": "66b8d4ad646f0ba57e16e2",  "previsaoChuva": {    "id": "string",    "datahora": "2024-08-11T15:15:34.227Z",    "previsaoChuva": "string"  }}
```

ExecuteClear

Responses

Curl

```
curl -X 'POST' \  https://localhost:7072/api/ControleIrrigacao \  -H 'accept: text/plain' \  -H 'Content-Type: application/json' \  -d '{  "localizacao": "Parque Ibirapuera",  "estado": "ligado",  "datahora": "2024-08-11T15:15:34.227Z",  "previsaoChuvaId": "66b8d4ad646f0ba57e16e2",  "previsaoChuva": {    "id": "string",    "datahora": "2024-08-11T15:15:34.227Z",    "previsaoChuva": "string"  }  }'
```

Request URL

https://localhost:7072/api/ControleIrrigacao

Server response

CodeDetails

201

Undocumented

Response body

```
{  "success": true,  "message": "Controle de irrigação criado com sucesso",  "createdItem": {    "id": "66b8d5c1df46f0ba57e16e3",    "localizacao": "Parque Ibirapuera",    "estado": "ligado",    "datahora": "2024-08-11T15:15:34.227Z",    "previsaoChuvaId": "66b8d4ad646f0ba57e16e2",    "previsaoChuva": {      "id": "66b8d4ad646f0ba57e16e3",      "datahora": "2024-08-11T15:11:34.895Z",      "previsaoChuva": "Com Chuva"    }  }  }
```

Response headers

content-type: application/json; charset=utf-8date: Sun, 11 Aug 2024 15:16:16 GMTlocation: https://localhost:7072/api/ControleIrrigacao/66b8d5c1df46f0ba57e16e3server: Kestrel

Response

CodeDescriptionLinks

200

Success

No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{  "id": "string",  "localizacao": "string",  "estado": "string",  "datahora": "2024-08-11T15:16:17.313Z",  "previsaoChuvaId": "string",  "previsaoChuva": {    "id": "string",    "datahora": "2024-08-11T15:16:17.313Z",    "previsaoChuva": "string"  }  }
```

Figura 58

GetAll

GET /api/ControleIrrigacao

Parameters

Cancel

No parameters

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7072/api/ControleIrrigacao' \
  -H 'accept: text/plain'
```

Request URL

```
https://localhost:7072/api/ControleIrrigacao
```

Server response

CodeDetails

200

Response body

```
{
  "id": "66b8d5c1d46f00a657c16c3",
  "localizacao": "Parque Ibirapuera",
  "estado": "ligado",
  "dataHora": "2024-08-11T15:15:34.227Z",
  "previsaoChuvaId": "66b8d4a6d46f00a657c16c2",
  "previsaoChuva": {
    "id": "66b8d4a6d46f00a657c16c2",
    "dataHora": "2024-08-11T15:11:34.805Z",
    "previsaoChuva": "Com Chuva"
  }
},
  "id": "66b8d510d46f00a657c16c4",
  "localizacao": "Parque Villa-Lobos",
  "estado": "ligado",
  "dataHora": "2024-08-11T15:15:34.227Z",
  "previsaoChuvaId": "66b8d4a6d46f00a657c16c2",
  "previsaoChuva": {
    "id": "66b8d4a6d46f00a657c16c2",
    "dataHora": "2024-08-11T15:11:34.805Z",
    "previsaoChuva": "Com Chuva"
  }
}

```

Response headers

```
content-type: application/json; charset=utf-8
date: Sun, 11 Aug 2024 15:18:18 GMT
server: Kestrel
```

Responses

CodeDescriptionLinks

200

Success

No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

```
{
  "id": "string",
  "localizacao": "string",
  "estado": "string",
  "dataHora": "2024-08-11T15:18:18.738Z",
  "previsaoChuvaId": "string",
  "previsaoChuva": {
    "id": "string",
    "dataHora": "2024-08-11T15:18:18.738Z",
    "previsaoChuva": "string"
  }
}

```

Figura 59

GetByld

GET

/api/ControleIrrigacao/{id}

Parameters

Cancel

Name	Description
id <small>* required</small>	
string	66b8d610df46f00a657e16e4
(path)	

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7072/api/ControleIrrigacao/66b8d610df46f00a657e16e4' \
  -H 'accept: text/plain'
```

Request URL

https://localhost:7072/api/ControleIrrigacao/66b8d610df46f00a657e16e4

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "controleIrrigacao": { "id": "66b8d610df46f00a657e16e4", "localizacao": "Parque Villa-Lobos", "estado": "ligado", "datahora": "2024-08-11T15:15:34.227Z", "previsaoChuvaId": "66b8d610df46f00a657e16e2", "previsaoChuva": { "id": "66b8d610df46f00a657e16e2", "datahora": "2024-08-11T15:11:34.985Z", "previsaoChuva": "Com Chuva" } }, "message": "Controle de irrigação encontrado com sucesso." }</pre></div><div>Download</div></div> <div><div>Response headers</div><div><pre>content-type: application/json; charset=utf-8 date: Sun, 11 Aug 2024 15:19:03 GMT server: Kestrel</pre></div></div>

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{
  "id": "string",
  "localizacao": "string",
  "estado": "string",
  "datahora": "2024-08-11T15:19:04.211Z",
  "previsaoChuvaId": "string",
  "previsaoChuva": {
    "id": "string",
    "datahora": "2024-08-11T15:19:04.211Z",
    "previsaoChuva": "string"
  }
}
```

Figura 60

Update

PUT

/api/ControleIrrigacao/{id}

Cancel

Reset

Parameters

Name	Description
id <small>* required</small>	
string (path)	66b8d610df46f00a657e16e4

Request body

application/json

```
{  "id": "66b8d610df46f00a657e16e4",  "localizacao": "Parque Ibirapuera",  "estado": "Ligado",  "dataHora": "2024-08-11T15:19:28.278Z",  "previsaoChuvaId": "66b8d4a6df46f00a657e16e2",  "previsaoChuva": {    "id": "string",    "dataHora": "2024-08-11T15:19:28.278Z",    "previsaoChuva": "string"  }}
```

Execute

Clear

Responses

Curl

```
curl -X 'PUT' \  'https://localhost:7072/api/ControleIrrigacao/66b8d610df46f00a657e16e4' \  -H 'accept: */*' \  -H 'Content-Type: application/json' \  -d '{  "id": "66b8d610df46f00a657e16e4",  "localizacao": "Parque Ibirapuera",  "estado": "Ligado",  "dataHora": "2024-08-11T15:19:28.278Z",  "previsaoChuvaId": "66b8d4a6df46f00a657e16e2",  "previsaoChuva": {    "id": "string",    "dataHora": "2024-08-11T15:19:28.278Z",    "previsaoChuva": "string"  }  },}'
```

Request URL

https://localhost:7072/api/ControleIrrigacao/66b8d610df46f00a657e16e4

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "controleIrrigacao": { "id": "66b8d610df46f00a657e16e4", "localizacao": "Parque Ibirapuera", "estado": "Ligado", "dataHora": "2024-08-11T15:19:28.278Z", "previsaoChuvaId": "66b8d4a6df46f00a657e16e2", "previsaoChuva": { "id": "66b8d4a6df46f00a657e16e2", "dataHora": "2024-08-11T15:11:34.805Z", "previsaoChuva": "Com Chuva" } }, "message": "Controle de Irrigação atualizado com sucesso"}</pre></div><div>Download</div></div> <div><div>Response headers</div><div><pre>content-type: application/json; charset=utf-8date: Sun, 11 Aug 2024 15:20:16 GMTserver: Kestrel</pre></div></div>

Responses

Code	Description	Links
200	Success	No links

Figura 61

Delete

DELETE

/api/ControleIrrigacao/{id}

Parameters

Cancel

Name	Description
id <small>* required</small>	
string	66b8d610df46f00a657e16e4
(path)	

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
'https://localhost:7072/api/ControleIrrigacao/66b8d610df46f00a657e16e4' \
-H 'accept: */*'
```

Request URL

```
https://localhost:7072/api/ControleIrrigacao/66b8d610df46f00a657e16e4
```

Server response

Code	Details
200	<div><div>Response body</div><div>Controle de Irrigação removido com sucesso.</div><div>Download</div></div> <div><div>Response headers</div><div>content-type: text/plain; charset=utf-8 date: Sun, 11 Aug 2024 15:21:01 GMT server: Kestrel</div></div>

Response

Code	Description	Links
200	Success	No links

Figura 62