

Banco de Dados

Importar o backup do banco de dados no SQL Server caso você não tenha.

Iremos utilizar o mesmo para manhã/tarde.

SENAI_SVIGUFO_MANHA_BACKEND.bak

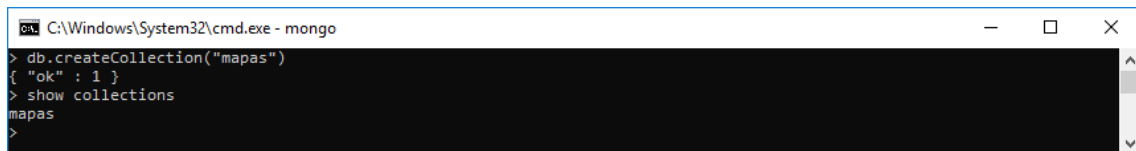
MongoDB

No mongodb, vamos criar uma database e criar uma nova collection para guardar os dados de localização dos usuários.

use svigufo

```
db.createCollection("mapas")
```

```
show collections
```



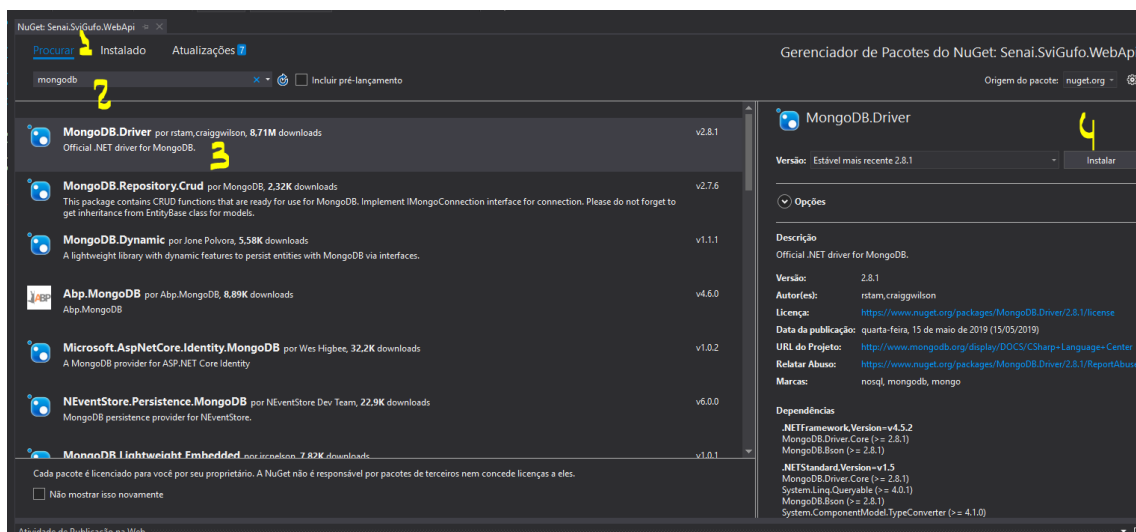
```
C:\Windows\System32\cmd.exe - mongo
> db.createCollection("mapas")
{ "ok" : 1 }
> show collections
mapas
>
```

Projeto BackEnd

Abrir projeto do backend. Alterar a string de conexão do banco de dados caso seja necessário.

Gerenciar pacotes do nuget.

Pesquisar por mongodb. Instalar *MongoDB.Driver*.



Adicionar um novo Domain chamado LocalizacaoDomain.cs.

Criação do domínio

```
public class LocalizacaoDomain
{
    [BsonId]
    [BsonRepresentation(BsonType.ObjectId)]
    public string Id { get; set; }

    [BsonElement("latitude")]
    [BsonRequired]
    public string Latitude { get; set; }

    //[BsonElement("longitude")]
    [BsonRequired]
    public string Longitude { get; set; }

    [BsonElement("usuario")]
    public int UsuarioId { get; set; }
}
```

Não incluir o UsuarioId

Criação da Interface

```
public interface ILocalizacaoRepository
{
    void Cadastrar(LocalizacaoDomain localizacao);

    List<LocalizacaoDomain> ListarTodos();

    List<LocalizacaoDomain> BuscarPorIdUsuario(int id);
}
```

Não incluir o BuscarPorIdUsuario

Criação do repositório

```

public class LocalizacaoRepository : ILocalizacaoRepository
{
    private readonly IMongoCollection<LocalizacaoDomain> _localizacoes;

    public LocalizacaoRepository()
    {
        var client = new MongoClient("mongodb://localhost:27017");
        var database = client.GetDatabase("svigufo");
        _localizacoes = database.GetCollection<LocalizacaoDomain>("mapas");
    }

    public List<LocalizacaoDomain> BuscarPorIdUsuario(int id)
    {
        return _localizacoes.Find(localizacao => localizacao.UsuarioId == id).ToList();
    }

    public void Cadastrar(LocalizacaoDomain localizacao)
    {
        _localizacoes.InsertOne(localizacao);
    }

    public List<LocalizacaoDomain> ListarTodos()
    {
        return _localizacoes.Find(localizacao => true).ToList();
    }
}

```

Não incluir o BuscarPorIdUsuario

Criação do Controller

```
[Produces("application/json")]
[Route("api/[controller]")]
[ApiController]
public class LocalizacoesController : ControllerBase
{
    private ILocalizacaoRepository LocalizacaoRepository { get; set; }

    public LocalizacoesController()
    {
        LocalizacaoRepository = new LocalizacaoRepository();
    }

    [HttpGet]
    public IActionResult ListarTodos()
    {
        try
        {
            return Ok(LocalizacaoRepository.ListarTodos());
        }
        catch (System.Exception ex)
        {
            return BadRequest();
        }
    }

    [HttpPost]
    public IActionResult Cadastrar(LocalizacaoDomain localizacao)
    {
        try
        {
            LocalizacaoRepository.Cadastrar(localizacao);
            return Ok();
        }
        catch (System.Exception ex)
        {
            return BadRequest();
        }
    }
}
```