**Wishlist**

Documentação WishList

Sumário

[1. Resumo 2](#_Toc86202810)

[2. Descrição do projeto 2](#_Toc86202811)

[Trello 3](#_Toc86202812)

[3. Back-End 3](#_Toc86202813)

[Funcionalidades 3](#_Toc86202814)

[Sistema Web 4](#_Toc86202815)

[Perfis de usuário: 4](#_Toc86202816)

[Funcionalidades: 4](#_Toc86202817)

# Resumo

Essa documentação tem como fundamento apresentar detalhadamente o desenvolvimento e organização do projeto “WishList”, destacando as etapas elaboradas do início até o término do projeto, o cronograma das atividades, o modo como foi produzido e o material com informações do sistema.

# Descrição do projeto

O projeto foi designado para que o usuário possa cadastrar e listar os desejos que ele quiser, assim como, marcar se foi realizado ou não. É uma forma de fixar aquela meta e fazer com que as pessoas tenham determinação afim de conquistar aquele desejo todos os dias que ele(a) acorda. O documento com o arquivo do projeto é encontrado no GitHub dos desenvolvedores da aplicação: livianegrini, gustrpaz, GustavoBorgesSouza e fe-lino.

### Trello

https://trello.com/b/q2pSPQfe/wishlist

# Back-End

O projeto WishList foi desenvolvido utilizando o SQL Server para a parte do banco de dados, o Microsoft Visual Studio para parte da API, em linguagem C#, Visual Studio com HTML e CSS para as telas e a utilização da biblioteca do React Js para o frontend e o Figma para o layout/design do projeto. A *Application Programming Interface (API) é uma forma de integrar sistemas* definindo as requisições e respostas *obtendo muita vantagem, por facilitar o desenvolvimento com mais agilidade precisão e principalmente organização,* seguindo protocolo HTTP.

Para a inserção e a troca de respostas é possível utilizar um formato em JSON (JavaScript Object Notation) para que o sistema possa ser acessado de várias maneiras Dispositivos diferentes sem se preocupar com a linguagem que usarão.

Para um haver a praticidade na organização a API foi feita utilizando o estilo de arquitetura REST, sigla em inglês para Representational State Transfer, ou, traduzindo, Interface de Programação de Aplicativos.

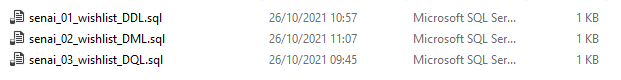
Como executar?

1º Passo – Clone o seguinte repositório do GitHub

<https://github.com/GustavoBorgesSouza/WishList>

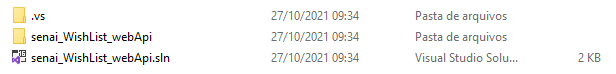


2º Passo – Execute os Scripts para a criação do banco de dados

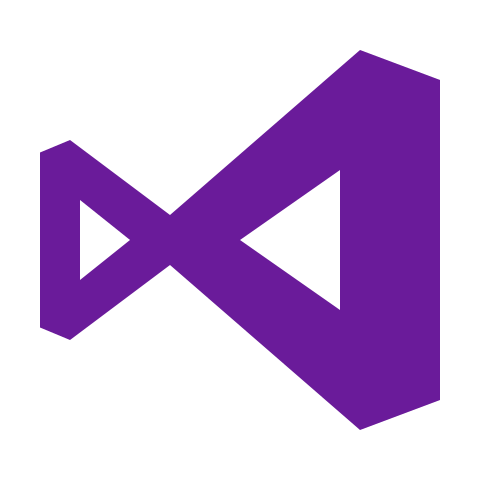


Recomendado a utilização do Microsoft SQL Server Management Studio, ou qual quer outra plataforma

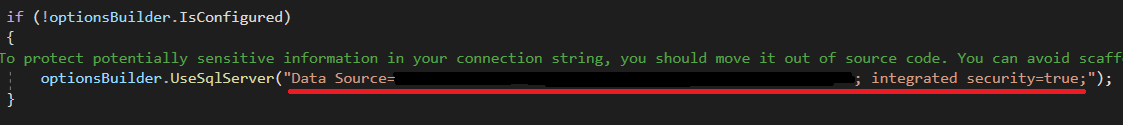
3º Passo – Fazer alteração na Context da API e rodar

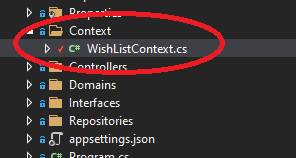
A API é encontrada na pasta BackEnd, abra o arquivo .sln 

Recomendado a utilização do Visual Studio 2019

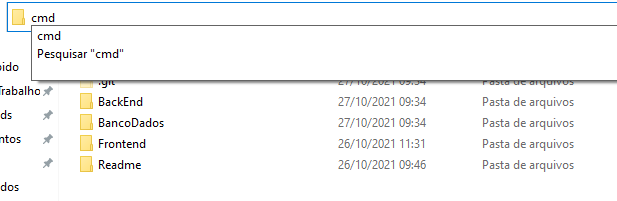


Altere o Caminho de Data Source para o nome do servidor ou da sua máquina





4º Passo – Rodar a API, apenas digitando ‘cmd’ no caminho da pasta



## Funcionalidades

* Cadastra os desejos do usuário, inserindo uma descrição
* Lista os desejos e a descrição

## Sistema Web

### Perfis de usuário:

1. **Administrador**: Área administrativa da escola;
2. **Comum**: Pode ser um espectador ou palestrante.

### Funcionalidades:

1. O **administrador** poderá cadastrar qualquer tipo de usuário;
2. O **administrador** poderá cadastrar os dados da instituição;
3. O **administrador** poderá cadastrar os tipos de eventos;
4. O **administrador** poderá cadastrar eventos livres ou restritos;
5. Qualquer usuário autenticado poderá ver todos os eventos cadastrados;
6. O usuário **comum** poderá ver os eventos que participará;
7. O usuário **comum** poderá se inscrever para assistir a um evento;
8. O **administrador** poderá aprovar a participação do espectador de um determinado evento restrito;