



PROJETO EM GRUPO

Módulo 5 - CRUD API

Vamos ver o vídeo?



Todos os direitos reservados
©2022 Resilia Educação

RESILIA |  **Senac**

CONTEXTO



Você e sua equipe foram escalados para desenvolverem **propostas de APIs** que serão o produto mínimo viável de um aplicativo. Vocês devem escolher o tema do aplicativo para identificar as entidades.



O QUE É PARA FAZER?

Nesse projeto você e sua *squad* serão responsáveis por definir quais são as entidades que o projeto precisa contemplar e cada um de vocês será responsável por implementar uma dessas entidades.

⇒ **Entrega:** deve ser feita em um único repositório (ou seja, o grupo vai trabalhar no mesmo repositório criando todas as rotas para as entidades mapeadas).

OBJETIVOS



O projeto deverá obrigatoriamente:

- ⇒ Utilizar o padrão MVC;
- ⇒ Utilizar os verbos HTTP seguindo o padrão REST;
- ⇒ Implementar todas as operações de CRUD;
- ⇒ Utilizar o padrão de projeto (design pattern) DAO para abstração de transações no banco, com Promises;
- ⇒ Utilizar o README.md do repositório para documentação, contendo informações como:
 - Como instalar as dependências do projeto;
 - Como executar o projeto;
 - Quais são as rotas possíveis;
 - Quaisquer outros pontos que você achar necessários;
- ⇒ Utilização de async/await para operações no banco (DAO)
- ⇒ Ter o código fonte hospedado em um repositório no Github.

OBJETIVOS



As rotas a serem desenvolvidas para as entidades são:

- ⇒ **Listagem de todos os registros;**
- ⇒ **Listagem de um único registro com base em um parâmetro de rota;**
- ⇒ **Inserção de um registro;**
- ⇒ **Modificação de um registro;**
- ⇒ **Deleção de um registro.**

APRESENTAÇÃO



- ⇒ Todos os integrantes do grupo deverão apresentar;
- ⇒ Utilize slides de apoio para sua apresentação;

EXTRAS



⇒ Algumas funcionalidades e técnicas podem ser desenvolvidas como extras no projeto São elas:

- Testes automatizados utilizando frameworks;
- Hospedagem da API;
- Validação e verificação de dados (tipo e características) utilizando models;
- Utilização de algum ORM ou ODM acompanhado de banco de dados de escolha.

Obs: Caso opte por utilizar ORM/ODM, não será necessária a implementação do DAO. Neste caso, as Models devem ser responsáveis pelas transações no banco.



F.A.Q.

Precisamos criar um README para o repositório?

Sim, esse é um requisito obrigatório do projeto. Fiquem atentos com todos os requisitos obrigatórios, pois a checagem do projeto será realizada com base nesses itens.

Preciso entregar os requisitos extras?

Não, eles estão ali como um desafio extra para os grupos que conseguirem avançar rapidamente pelo projeto e para praticar alguns pontos que foram abordados de forma mais introdutória na jornada.

Posso usar uma outra hospedagem sem ser o Heroku?

Sim, sem problema. Como esse é um requisito extra vocês podem entregar ou não essa etapa (uma alternativa é o Cyclic).

Todos devem trabalhar no código do projeto?

Sim, vamos checar no Github os envios e quem participou escrevendo código para o projeto, se organizem para dividirem as tarefas e garantir que todos estão atuando em algum ponto do código.

MÃO NA MASSA

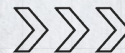


Momento 1 - Início

Comece criando um plano de ação!
Planejamento é a parte mais importante de um projeto.

Utilize o tempo para descobrir **o que você já sabe** e o que **ainda falta aprender** (ou pesquisar).

Façam uma **análise dos dados** escolhidos e comecem a etapa de estruturação dos dados.

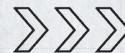


MÃO NA MASSA



Momento 2 - Andamento

Comecem a **escrever o código** que vai realizar a funcionalidade proposta, lembrem-se de consultar a documentação e outros exemplos na internet em caso de dúvidas.



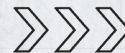
MÃO NA MASSA



Momento 3 - Refinamento

Comecem a **realizar os testes** da solução que está sendo desenvolvida.

Aproveitem o momento para **revisar o código e pensar em otimizações** para o projeto.





Momento 4 - Finalização e Entrega

Façam os **últimos ajustes** e preparem a apresentação com base nas informações coletadas, lembrem-se que a apresentação também deve ser entregue no repositório com os demais itens.

Não esqueça de subir tudo para o **Github** e adicionar as informações no *README* detalhando o projeto, as perguntas e as respostas.

RUBRICA



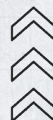
Conteúdo	Habilidades
NodeJS/ Express	<ol style="list-style-type: none">1. O Projeto está funcionando e não apresenta erros.2. O código segue as práticas sugeridas no framework utilizado.3. Implementa todas as operações de CRUD.4. Foi utilizado algum ORM ou ODM no projeto.5. A API está publicada e o link para testes está no README do repositório
Programação e operações no código	<ol style="list-style-type: none">1. O programa foi estruturado corretamente no padrão de Model, View e Controller, e todas as classes estão corretamente criadas;2. Foram aplicados corretamente os conceitos de POO e de reutilização/reaproveitamento de código.3. Utiliza o padrão de projeto DAO para abstração de transações no banco, com Promises.4. Utiliza async/await para operações no banco (DAO).5. Nenhum problema na sintaxe do projeto.
Funcionalidades da API	<ol style="list-style-type: none">1. Utilizar os verbos HTTP seguindo o padrão REST.2. Todas as funcionalidades/rotas da API foram implementadas.3. Todas as funcionalidades funcionam de forma adequada.4. Validação e verificação de dados (tipo e características) utilizando models.5. Foram implementadas funcionalidades extras na API (por exemplo, validação e verificação dos parâmetros



RUBRICA



Conteúdo	Habilidades
Documentação	<ol style="list-style-type: none">1. Foi desenvolvido um Readme com as informações do projeto.2. O Readme está bem completo apresentando as etapas para configuração do projeto, execução e testes de todas as suas funcionalidades;3. O Readme conta com contextualização do projeto, informando o problema a ser resolvido.4. O Readme contém informações extras, formatação do conteúdo para facilitar a leitura, imagens, links externos, etc.5. Os slides foram entregues junto do projeto.
Git/GitHub	<ol style="list-style-type: none">1. Entregou o link do repositório no Github.2. O código está completo/funcionando no Github.3. O repositório conta com readme, requerimentos e documentação.4. Todos os integrantes contribuíram com o repositório.5. As descrições dos commits/PRs estão bem redigidas e apresentam bem as mudanças realizadas.





**Até a próxima e
#confianoprocesso**



Todos os direitos reservados
©2022 Resilia Educação