



Programação GO

Requisito #1

// Especificação de Requisitos

ÍNDICE

Regras de Negócios Associadas 2**Contratos referentes ao User Story** 2**Representação JSON** 2

Representação JSON do Response de GET 3

Representação JSON da Request de POST/PATCH 3

User Story 4

User Story: #1 : Criar Seller (CREATE) 4

User Story: #1 : Buscar todos os Sellers (READ) 4

User Story: #1 : Buscar Seller (READ) 5

User Story: #1 : Modificar Seller (UPDATE) 5

User Story: #1 : Eliminar Seller (DELETE) 5

Importante:

As histórias dos usuários são narradas do ponto de vista do Representante do warehouse de fulfillment com base em suas necessidades.



Regras de Negócios Associadas

ID	Descrição	Grau de Variação	Fonte
C			
RN-11	<p>É necessário que o Vendedor (Seller) seja composto por</p> <ul style="list-style-type: none">• id: Uma identificação única do vendedor• cid: Uma identificação da empresa (company_id) semelhante a cuit• company_name: o nome de uma empresa• address: um endereço• telephone: um telefone	Baixa	Política empresarial

Contratos referentes a User Story

HTTP	Modelo de URI	Descrição	US-code
GET	/api/v1/sellers	Quando a solicitação for bem-sucedida, o back-end retornará uma lista de todos os vendedores existentes.	#1
GET	/api/v1/sellers/:id	Quando a solicitação for bem-sucedida, o back-end retornará as informações solicitadas do vendedor Quando o vendedor não existir, um código 404 será devolvido	#2
POST	/api/v1/sellers	Quando a entrada de dados for bem-sucedida, um código 201 será retornado junto com o objeto inserido. Se o cid já existir, ele retornará um erro 409 Conflict. Se o objeto JSON não contiver os campos necessários, um código 422 será retornado.	#3
PATCH	/api/v1/sellers/:id	Quando a atualização dos dados for bem sucedida, o vendedor será devolvido com as informações atualizadas juntamente com um código 200 (você pode atualizar o número de campos que desejar) Se o tipo de vendedor a ser atualizado não existir, um código 404 será devolvido.	#4
DELETE	/api/v1/sellers/:id	Quando a exclusão for bem-sucedida, um código 204 será retornado. Quando o vendedor não existir, um código 404 será devolvido	#5



Representação JSON

Representação JSON do Response de GET

JSON

```
{
  "data": [
    {
      "id": 1,
      "cid": 1,
      "company_name": "Digital House",
      "address": "Monroe 860",
      "telephone": "47470000",
    }
  ]
}
```

Representação JSON da Request de POST/PATCH

JSON

```
{
  "cid": 1,
  "company_name": "Digital House",
  "address": "Monroe 860",
  "telephone": "47470000",
}
```



User Story

User Story: #1 : Criar Seller (CREATE)

User Story Code: #1 Criar CRUD para Seller User Story Name: criar Seller	Horas estimadas
COMO Representante do warehouse de fulfillment, QUERO criar um Vendedor PARA uso em outras partes do aplicativo	?
CENÁRIO 1: O vendedor não existe. DADO QUE o Vendedor não existe E cid do vendedor é único ENTÃO o Representante cria um Vendedor E o vendedor é criado com cid exclusivo CENÁRIO 2: Vendedor único por cid. DESDE que existe um Vendedor com cid E tentando criar um novo vendedor com cid existente. ENTÃO o representante retorna um error 409 Conflict	
VALIDAÇÃO <ul style="list-style-type: none">▪ Que todos os campos estejam completos.▪ cid tem que ser único.	

User Story: #1 : Buscar todos os Sellers (READ)

User Story Code: #1 Criar CRUD para Seller User Story Name: buscar todos os sellers	Horas estimadas
COMO Representante do warehouse de fulfillment, QUERO buscar todos os Vendedores PARA uso em outros lugares do aplicativo	?
CENÁRIO 1: Existem vendedores DADO QUE existem vendedores ENTÃO o Representante encontra todos os Vendedores E todos os vendedores são mostrados	

User Story: #1 : Buscar Seller (READ)

User Story Code: #1 Criar CRUD para Seller User Story Name: buscar um Seller por id	Horas estimadas
COMO Representante do warehouse de fulfillment, QUERO procurar um Vendedor por id PARA usá-lo em outro lugar no aplicativo	?



CENÁRIO 1: Existem vendedores.
DADO QUE existem vendedores.
ENTÃO o Representante procura um Vendedor por id
E o vendedor que corresponde ao id inserido é exibido

VALIDAÇÃO

- Prove que o vendedor existe.

User Story: #1 : Modificar Seller (UPDATE)

User Story Code: #1 Criar CRUD para Seller User Story Name: modificar Seller por id	Horas estimadas
COMO Representante do warehouse de fulfillment, QUERO modificar um Vendedor por id PARA que as alterações sejam refletidas e usadas em outras partes do aplicativo	?
CENÁRIO 1: O vendedor existe DADO QUE o vendedor existe ENTÃO o Representante modifica um Vendedor por id E os dados do Vendedor são modificados	
VALIDAÇÃO <ul style="list-style-type: none">▪ Provar que o vendedor existe e se realize a alteração	

User Story: #1 : Eliminar Seller (DELETE)

User Story Code: #1 Criar CRUD para Seller User Story Name: eliminar um seller	Horas estimadas
COMO representante do warehouse de fulfillment, QUERO excluir um vendedor por id PARA que as alterações sejam refletidas e não possam ser usadas em outros locais do aplicativo	?
CENÁRIO 1: O vendedor existe. DADO QUE o Vendedor existe. ENTÃO o Representante remove o Vendedor inserindo o id E o vendedor é removido	
VALIDAÇÃO <ul style="list-style-type: none">▪ Provar que o vendedor não existe com id	