



## Programação Java

# Requisito 6

// Especificações de Requisitos

## Requisitos US: ml-develop-fidelity-06

User Story	1
Representação JSON	2
Contratos referentes a User Story	3

#### Importante:

As histórias de usuários são contadas do ponto de vista [do ator americano] de acordo com suas necessidades. Os serviços são expostos a partir do depósito de distribuição para serem consumidos pelo solicitante. Os contratos referem-se à História do Usuário.

Este requisito é livre e é realizado individualmente, partindo do pressuposto de que deve agregar valor à aplicação, utilizando dados existentes ou novos que se considerem necessários. Dito isso, o participante fica livre para fazer um aplicativo separado que se comunica com o primeiro já criado, se desejar. Esclarecimento: você precisa criar um novo aplicativo.





## [Incrementar pontos bónus de fidelidade]

# **User Story**

User Story Code: ml-develop-fidelity-06 User Story Name: Incrementar pontos bónus de fidelidade	Horas estimadas
COMO Representante do armazém de distribuição, QUERO Inserir, modificar, deletar e buscar prêmios e benefícios que os usuários podem comprar com seus pontos de fidelidade PARA melhorar a experiência do usuário	18
CENÁRIO 1: Um prêmio é registrado, atualizado, deletado ou buscado	
DADO QUE exista no caso de atualizar, deletar e buscar.	
QUANDO a ação é disparada	
ENTÃO o registro do prêmio no banco de dados sofre a ação	

CENÁRIO 2: Um usuário troca seus pontos de fidelidade por um prêmio

DADO QUE o usuário tenha pontos o suficiente

QUANDO a ação de troca é disparada

**ENTÃO** é validado se o prêmio existe

E o prêmio tem um identificador

E o usuario existe

E se o usuário tem uma quantia de pontos maior que suficiente para trocar pelo prêmio E após isso, se os pontos do usuário são debitados

#### VALIDAÇÃO

- Autentique como ...... e acesse os terminais.
- Validação de parâmetros obrigatórios.





# Representação JSON

## Request

```
{
  "name": "String",
  "cost": "Long"
}
```

## Response

```
"id": "Long",
  "name": "String",
  "cost": "Long"
}
```

# Contratos referentes a la User Story

HTTP	Plantilla URI	Descripción	US-code
GET	/api/ <mark>v1</mark> /fidelity	Obtenha uma lista de todos os prêmios/benefícios que estão disponibilizados pros usuários comprarem	ml-develop-fidelit y-06





POST	/api/ <mark>v1</mark> /fidelity	Cadastre um prêmio ou benefício que pode ser comprado pelos usuários posteriormente, retorne o prêmio criado e um status "201 CREATED"	ml-develop-fidelit y-06
POST	/api/v1/fidelity/{customer}/{bonus}	Um usuário com id = customer, realiza a troca de seus pontos por um prêmio de id = bônus, existente no banco de dados, e tem seus pontos subtraídos	ml-develop-fidelit y-06
PUT	/api/ <mark>v1</mark> /fidelity/{id}	Sobrescreva um prêmio / benefício no banco de dados com novas informações. Retorne o prêmio atualizado e um status "201 CREATED"	ml-develop-fidelit y-06
DELETE	/api/ <mark>v1</mark> /fidelity/{id}	Apague um prêmio / benefício não permitindo mais ele ser resgatado. Devolva uma mensagem de string caso bem sucedido e o código "201 CREATED"	ml-develop-fidelit y-06
GET	/api/v1/fidelity/{id}	Obtenha os dados de um prêmio/benefício específico a partir do seu ID de identificação. Retorne o prêmio e um status "201 CREATED"	ml-develop-fidelit y-06

Observação: Contemple outros tipos de erros. Use o script de carregamento