



Programação Java

Requisito 6

// Especificações de Requisitos

Requisitos US: ml-review-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.





Product's Review

User Story

User Story Code: ml-review-06 User Story Name: [nome da user story]	Horas estimadas
COMO comprador, QUERO adicionar um review do produto que comprei PARA dar um feedback ao vendedor e auxiliar outros compradores que estejam interessados nesse produto.	?
CENÁRIO 1: quero avaliar o produto que comprei.	

DADO QUE o produto foi comprado pelo comprador

E que o produto existe **E** que o comprador existe **QUANDO** produto foi finalizado **ENTÃO** o produto é avaliado

CENÁRIO 2: quero editar minha review. **DADO QUE** o produto já foi avaliado

E que o comentário existe **E** que produto existe

ENTÃO quero editar minha avaliação.

CENÁRIO 3: quero consultar minha avaliação.

DADO QUE o produto já foi avaliado

E que o comentário existe

E que produto existe

ENTÃO quero visualizar minha avaliação.

VALIDAÇÃO

- A nota do produto deve ser números inteiros entre 1 e 5.
- O comentário deve ter no máximo 300 caracteres.





Representação JSON

Request POST/PUT

```
{
    "reviewId": 5,
    "grade": 4,
    "comment": "agora sim!",
    "announcementId": 2,
    "userId": 1
}
```

Response POST

```
{
    "reviewId": 27,
    "grade": 4,
    "comment": "maravilhoso",
    "announcementId": 2,
    "userId": 1
}
```

Response PUT

```
{
    "grade": 4,
    "comment": "agora sim!"
}
```

Response GET

```
{
    "reviewId": 1,
    "grade": 4,
    "comment": "Muito bom",
    "userId": 1,
    "announcementId": 2
}
```





Product Review List

User Story

User Story Code: ml-review-list-06 User Story Name: product review list	Horas estimadas
COMO vendedor ou possível comprador, QUERO consultar as avaliações de um produto PARA estimar o quanto os compradores estão satisfeitos com as compras.	?
CENÁRIO 1: considerando que o produto foi avaliado, eu como vendedor quero comentários de um produto. DADO QUE o produto foi avaliado pelo comprador E que o produto existe, E que a review existe, QUANDO o produto comprado é avaliado, ENTÃO eu consigo consultar a lista de reviews	consultar a lista de
CENÁRIO 2: considerando que o produto foi avaliado, eu como vendedor quero comentários de um produto filtrando pela nota recebida. DADO QUE o produto foi avaliado pelo comprador E que o produto existe, E que a review existe, E que tenho a lista de reviews, QUANDO o produto comprado é avaliado, ENTÃO eu consigo consultar a lista de reviews filtrando pela nota recebida	consultar a lista de

Representação JSON

Response GET list and GET list filter by grade



```
"announcementId": 2
},
{
    "reviewId": 2,
    "grade": 4,
    "comment": "Muito bom",
    "userId": 1,
    "announcementId": 2
},
{
    "reviewId": 3,
    "grade": 4,
    "comment": "Muito bom",
    "userId": 1,
    "announcementId": 2
},
    "reviewId": 4,
    "grade": 4,
    "comment": "Muito bom",
    "userId": 1,
    "announcementId": 2
```

Contratos referentes a la User Story

HTTP	Plantilla URI	Descripción	US-code
GET	/api/ <mark>v1</mark> /fresh-product/review/?id =1	get review	ml-review-06
POST	/api/ <mark>v1</mark> /fresh-product/review		ml-review-06
		create review	
PUT	/api/ <mark>v1</mark> /fresh-product/review		ml-review-06
		update review	





GET list	/api/ <mark>v1</mark> /fresh-product/review/list ?announcementId=2	get review list by announcement id	ml-review-06
GET list by filter	/api/ <mark>v1</mark> /fresh-product/review/gr ade?announcementId=2&reviewGr ade=4	get review list by announcement id and filter by grade	ml-review-06