



Programação Java

## ➤ Requisito 6

// Especificações de Requisitos

Requisitos US: ml-creat-crud-for-buyer-01

<b>User Story</b>	2
<b>Representação JSON</b>	2
<b>Contratos referentes a User Story</b>	3

**Importante:**

As histórias de usuários são contadas do ponto de vista do representante, vendedor e comprador warehouse com base em suas necessidades. Os serviços são expostos a partir da base de dados referente aos compradores que são consumidos pelos representantes, comprador e vendedor solicitante. Os contratos referem-se à História do Usuário.



## Cadastro do Comprador

### User Story

User Story Code: ml-creat-crud-for-buyer-06 User Story Name: Cadastro completo do comprador	Horas estimadas
<b>COMO</b> vendedor, comprador e representante, <b>QUERO</b> poder realizar buscas, incluir, deletar e alterar um comprador <b>PARA</b> manter a base de compradores sempre atualizada ou consultar seus próprios dados (Comprador).	<b>20h</b>
<b>CENÁRIO 1:</b> Um comprador é registrado. <b>DADO QUE</b> o comprador não exista na base atual. <b>E</b> que possua um usuário para inseri-lo. <b>QUANDO</b> o usuário inclui um novo comprador na base <b>ENTÃO</b> , ele poderá ser consultado <b>E</b> alterado atualizando o atual <b>E</b> excluído	
<b>VALIDAÇÃO</b> <ul style="list-style-type: none"><li>▪ Autentique como vendedor, comprador ou representante e acesse os terminais para cadastrar um comprador.</li><li>▪ Para realizar a consulta de todos os compradores é necessário ter um perfil ADM, se não o usuário terá acesso somente a suas informações (usuário logado).</li><li>▪ Para realizar a alteração de dados de um comprador é necessário ter um perfil ADM, se não o usuário terá acesso somente a suas informações (usuário logado).</li><li>▪ Para exclusão de um comprador é necessário ter um perfil ADM, se não o usuário terá acesso somente a suas informações (usuário logado).</li><li>▪ Validação de parâmetros obrigatórios.</li></ul>	



## Representação JSON

### Request

Post/Put insert/alter Buyer

```
{
  "username": "String",
  "name": "String",
  "email": "String",
  "pass": "String",
  "address": "String"
}
```

### Response

```
[
  {
    "id": 1,
    "username": "DhMaury",
    "name": "Maury",
    "email": "DHMaury@gmail.com",
    "address": "Endereco Sul"
  },
  {
    "id": 2,
    "username": "DhKenyo",
    "name": "Kenyo",
    "email": "DHKenyo@gmail.com",
    "address": "Endereco Desc"
  },
  {
    "id": 3,
    "username": "DhMichele",
    "name": "Michele",
    "email": "DHMichele@gmail.com",
    "address": "Endereco Sao Paulo"
  }
]
```



## Contratos referentes a User Story

HTTP	Plantilla URI	Descripción	US-code
GET	api/v1/fresh-products/buyer	Listar todos compradores da base de dados com o retorno código de status "200 ok".	ml-creat-crud-for-buye-06
GET	api/v1/fresh-products/buyer/{{id}}	Listar dados de um comprador com base no id informado e com o retorno código de status "200 ok".	ml-creat-crud-for-buye-06
POST	api/v1/fresh-products/buyer	Cadastrar um comprador e devolver o código de status "201 - CREATED" e no body as informações do comprador adicionado.	ml-creat-crud-for-buye-06
PUT	api/v1/fresh-products/buyer/{{id}}	Alterar um comprador pelo número de id e devolver o código de status "200 - OK" com os dados alterados no body.	ml-creat-crud-for-buye-06
DELETE	api/v1/fresh-products/buyer/{{id}}	Deletar um comprador e devolver o código de retorno de status "200 - OK" e no body uma mensagem informando o id excluído.	ml-creat-crud-for-buye-06

Observação:

Contemple outros tipos de erros.

Use o script de carregamento.

Trabalhe com o token de acesso de um usuário autenticado.