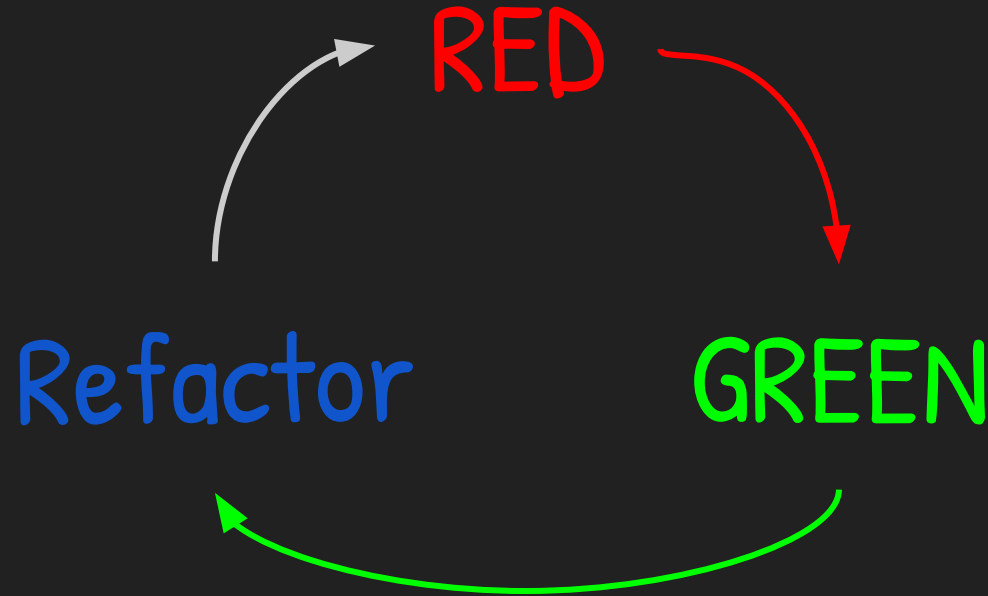


Test Driven Development



Projeto: Kanban Board

Kanban Board

← → ↺

http://localhost:8080

TODO

Card title lorem ipsum

Card title lorem ipsum

Card title lorem ipsum

Card title ...

Add Card

DOING

Card title lorem ipsum

Card title ...

Add Card

List title...

Add List

Por onde começar ?

1. Entendendo do Problema

2. Arquitetura e Design

Projeto: Kanban Board

BUCKET APIS

Create Bucket

POST /v1/buckets

```
{
  "id": "c86e2bc7-b777-4e72-a955-b62983018a85",
  "position": 1,
  "name": "TODO"
}
```

Success: 201

Error: 400 - fields validation.

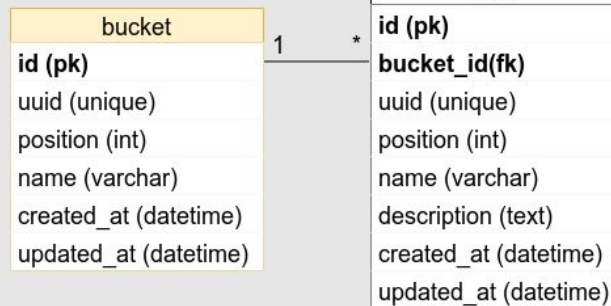
Update Bucket

PUT /v1/buckets/c86e2bc7-b777-4e72-a955-b62983018a85

```
{
  "position": 1,
  "name": "DOING"
}
```

Success: 201

Error: 400 - fields validation.



Kanban Board

http://localhost:8080

TODO

Card title lorem ipsum

Card title lorem ipsum

Card title lorem ipsum

Card title ...

Add Card

DOING

Card title lorem ipsum

Card title ...

Add Card

List title...

Add List

Projeto: Kanban Board

CARDS APIS

Create Card

```
POST /v1/cards
{
  "id": "c86e2bc7-b777-4e72-a955-b62983018a85",
  "bucketId": "f1e109b4-8e01-4f7a-aeee-4afe72d0d072",
  "position": 1,
  "name": "Fix technical debt"
}
```

Success: 201

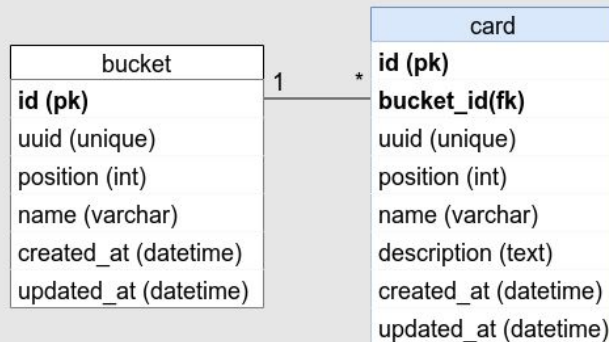
Error: 400 - fields validation.

Update Card

```
POST /v1/cards/c86e2bc7-b777-4e72-a955-b62983018a85
{
  "bucketId": "f1e109b4-8e01-4f7a-aeee-4afe72d0d072",
  "position": 2,
  "name": "Fix code smells",
  "description": "Lorem ipsum dolor amina ..."
}
```

Success: 201

Error: 400 - fields validation.



The screenshot shows a web browser window titled "Kanban Board" with the URL "http://localhost:8080". The interface features two main columns: "TODO" and "DOING".

- TODO Column:** Contains three blue cards with the text "Card title lorem ipsum". Below the cards is an input field labeled "Card title ..." and an "Add Card" button.
- DOING Column:** Contains one blue card with the text "Card title lorem ipsum". Below it is an input field labeled "Card title ..." and an "Add Card" button.
- Right Panel:** Contains an input field labeled "List title..." and an "Add List" button.

Kanban

BACKEND - SPRING BOOT

http

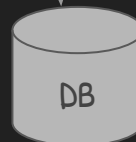
POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

jdbc

DB



Kanban

BACKEND - SPRING BOOT

http

POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

Controller

DTO

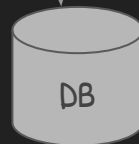
Service

Entity

Repository

jdbc

DB



Kanban

BACKEND - SPRING BOOT

http

POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

Adapter (in)

Domain

Adapter (out)

Controller

DTO

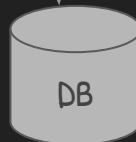
Service

Entity

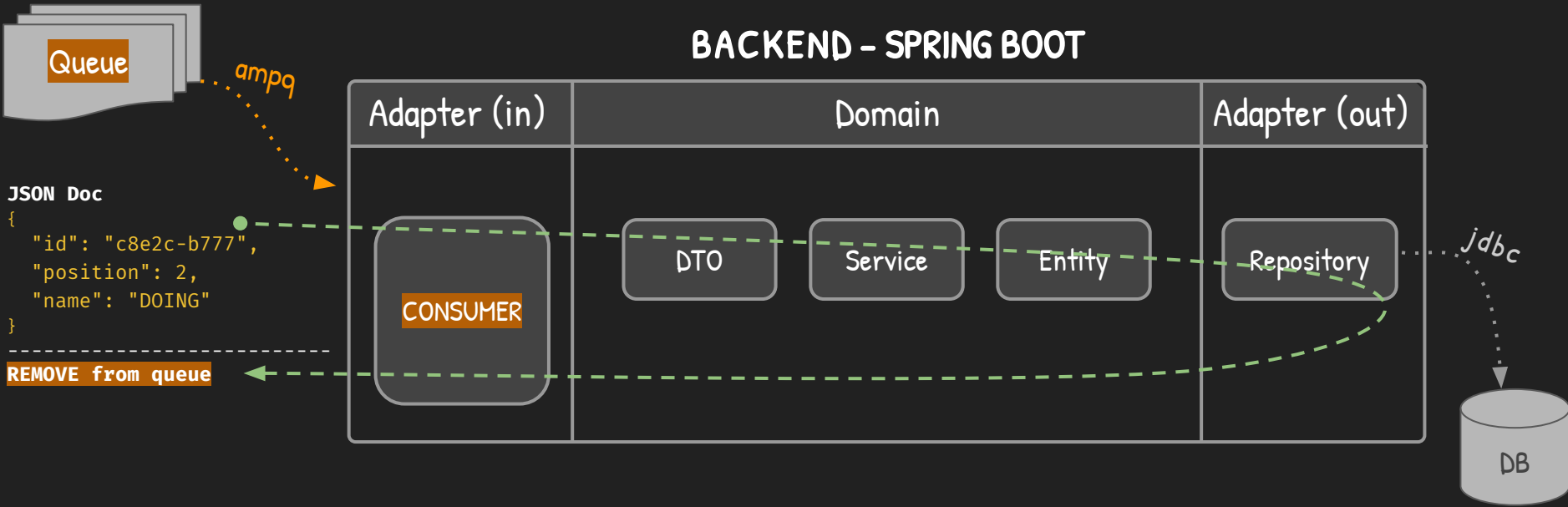
Repository

jdbc

DB



BACKEND - SPRING BOOT



Kanban

BACKEND - SPRING BOOT

POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

write/stateful

Adapter (in)

Controller

Domain

DTO

Service

Entity

Adapter (out)

Repository

jdbc

DB

org.seariver.kanbanboard

write

adapter

> in

> out

domain

> application

> core

> exception

Kanban

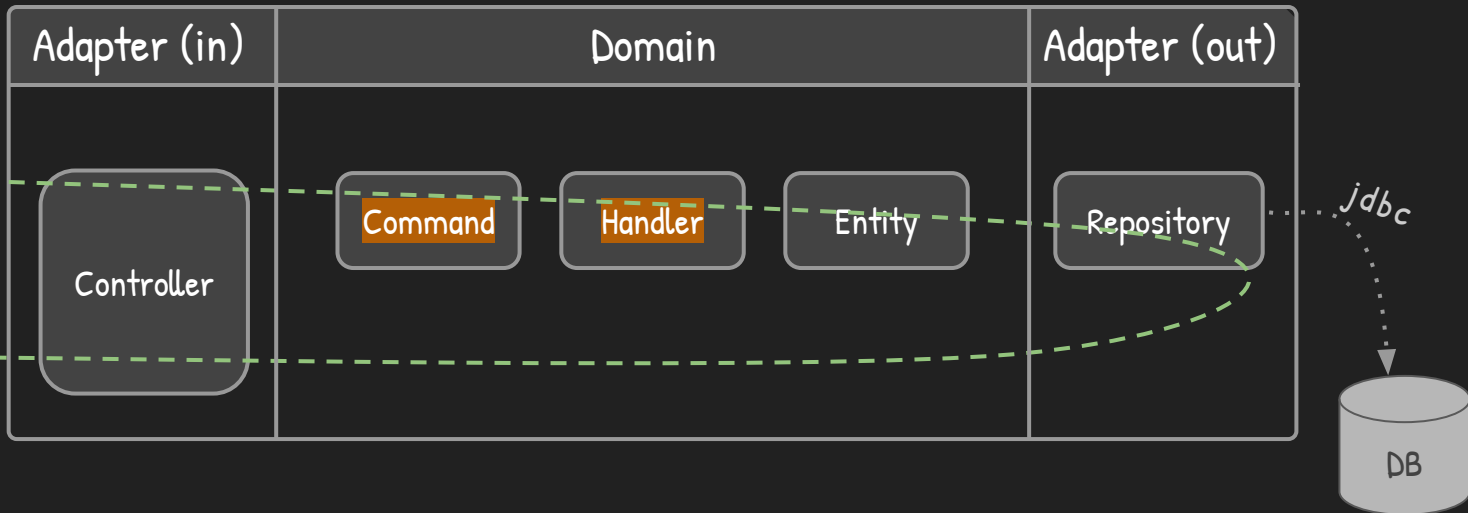
BACKEND - SPRING BOOT

POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

write/stateful



C.Q.R.S

Command Query Responsibility Separation

Kanban

BACKEND - SPRING BOOT

http

POST /v1/buckets

```
{
  "id": "c8e2c-b777",
  "position": 2,
  "name": "DOING"
}
```

RESPONSE: 201

write/stateful

read/stateless

GET /v1/buckets

RESPONSE: 200

```
[
  {
    "id": "c8e2c-b777",
    "position": 1,
    "name": "TODO"
  },
  {
    "id": "c8e2c-b777",
    "position": 2,
    "name": "DOING"
  }
]
```

Adapter (in)

Domain

Adapter (out)

Controller

Command

Handler

Entity

Repository

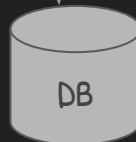
jdbc

DB

Controller

DTO

Repository



Kanban

BACKEND - SPRING BOOT

http

POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

write/stateful

read/stateless

GET /v1/buckets?params...

RESPONSE: 200

```
[  
  {  
    "id": "c8e2c-b777",  
    "position": 1,  
    "name": "TODO"  
  },  
  {  
    "id": "c8e2c-b777",  
    "position": 2,  
    "name": "DOING"  
  }  
]
```

Adapter (in)

Domain

Adapter (out)

Controller

Command

Handler

Entity

Repository

jdbc

DB

Controller

Query

Handler

Repository

DTO

Kanban

BACKEND - SPRING BOOT

http

POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

write/stateful

read/stateless

GET /v1/buckets

RESPONSE: 200

```
[  
  {  
    "id": "c8e2c-b777",  
    "position": 1,  
    "name": "TODO"  
  },  
  {  
    "id": "c8e2c-b777",  
    "position": 2,  
    "name": "DOING"  
  }  
]
```

Adapter (in)

Domain

Adapter (out)

Controller

Comm

OK

and

FAIL

Entity

Repository

jdbc

DB

Controller

Query

OK

FAIL

Repository

1

0

10

10K

Um

Nenhum

Alguns

Muitos

Kanban

BACKEND - SPRING BOOT

http

POST /v1/buckets

```
{  
  "id": "c8e2c-b777",  
  "position": 2,  
  "name": "DOING"  
}
```

RESPONSE: 201

write/stateful

read/stateless

GET /v1/buckets

RESPONSE: 200

```
[  
  {  
    "id": "c8e2c-b777",  
    "position": 1,  
    "name": "TODO"  
  },  
  {  
    "id": "c8e2c-b777",  
    "position": 2,  
    "name": "DOING"  
  }  
]
```

Adapter (in)

Domain

Adapter (out)

Controller

Command

Handler

Entity

Repository

jdbc

DB

BUS

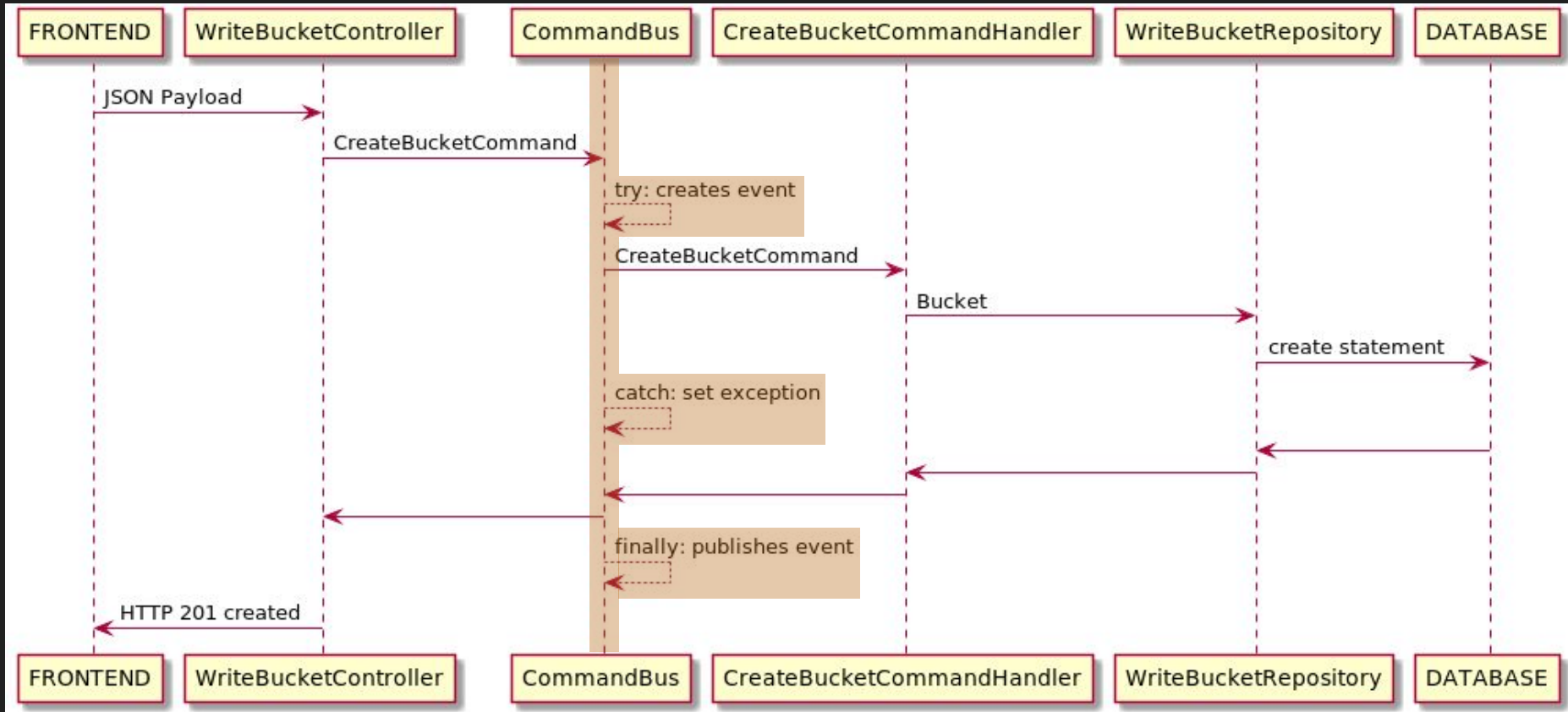
Query

Handler

Repository

DTO

Diagrama de Sequencia: Create Bucket





twitch.tv/mariores



youtube.com/user/mariores



github.com/mariores

**JAVA
DAS
RUAS**