

# Black Rainbow



## Postmortem iteração 3

Nesta iteração nossas histórias de usuário abrangiam a criação de um sistema de coleções e produtos, além de um sistema para escolher a empresa ativa entre suas empresas cadastradas, o que irá definir todas as suas ações dentro do sistema. Além de construir as duas histórias de usuário, tivemos tempo para implementar funcionalidades adicionais que são de suma importância. Nas imagens abaixo temos as histórias do usuário e suas respectivas implementações:

### Sistema de Empresa Ativa

#### Historia do usuario:

**Empresa Ativa**

Como representante

Para que eu tenha agilidade na hora de cadastrar clientes, pedidos e produtos

desejo um sistema aonde a empresa ativa será definida

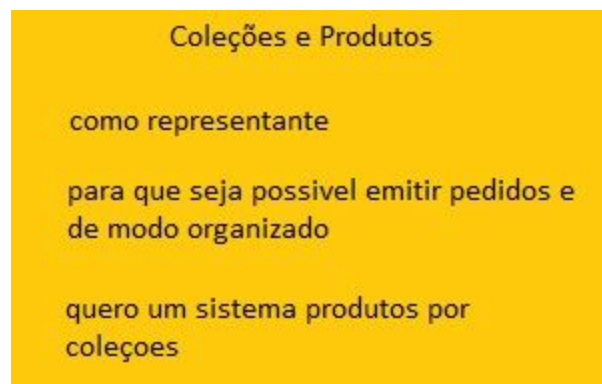
#### Implementação:

Primeiramente foi implementado o design do botão metade dropdown e metade botão padrão para a partir desse ponto ser possível suportar a ideia, com o botão posto em seu devido lugar iniciei a implementação pondo dentro da table de users uma nova coluna chamada “active\_empresa”, que pelo nome já diz que irá guardar a empresa ativa atual, logo após fiz as funções de suporte, uma de colocar uma empresa ativa e outra que retorna a empresa ativa atual, assim, com as funções e banco prontos fui colocar o botão de ativar empresa para funcionar, fiz com que o nome que aparece no botão seja o nome da empresa atual, e caso não

tenha uma empresa ativa no momento(novos usuários) ira aparecer apenas “NOBALK”, quando você aperta no dropdown, ira aparecer uma lista com as suas empresas cadastradas, e quando você clica em uma empresa ela ira ser setada como empresa ativa, mudando a coluna no table do user e atualizando a pagina, mostrando no botão agora o nome da empresa que você ativou, isso serve para que quando você for adicionar ou mostrar um cliente, pedido, produto ou coleção ele já vá automaticamente referenciado para a empresa ativa. Caso não tenha nenhuma empresa ativa e você tente acessar clientes ou produtos, você ira ser redirecionado par a pagina inicial da sua conta e ira aparecer uma mensagem explicando o motivo.

## Sistema de Coleções e Produtos

### História do usuário:



### Implementação:

Foi implementado inicialmente dois CRUDS, de Produtos e Coleções separadamente utilizando o scaffold, logo apos foi feito o mapeamento de coleções para pertencer a uma empresa utilizando reference(foreign key) e o model para has\_many e belongs\_to, assim quando uma coleção for criada ela sera setada para a empresa ativa, com as coleções mapeadas corretamente iniciei o mapeamento dos produtos, do mesmo modo, porem agora a entidade pertence a coleção, utilizei uma variável de classe(@@) para passar no momento de criação de um produto entre diversos views a coleção a qual ela é referenciada, pois o chamado para criação de produtos é feito nas views da coleção. Com a criação de produtos funcionando foi iniciado a implementação das listas de produtos das coleções dentro do “show” da coleção, como um produto obrigatoriamente pertence a uma coleção, quando você clica em “show” dessa coleção ira aparecer a lista de produtos que ela

pertence. Quando você clica na aba produtos do menu principal ira aparecer todos os produtos, independente da coleção da empresa ativa.

## Funcionalidades extras

Algumas funcionalidades extras foram adicionadas por uma sobra de tempo. Foram adicionadas as seguintes funcionalidades:

- Usuário pode deletar sua conta
- Email de ativação para cadastro de conta
- Recuperação de senha através do email

## TDD e BDD

TDD:

```
imlbf:~/workspace (master) $ rake test TEST=test/models/collection_test.rb
Started

 5/5: [=====] 100% Time: 00:00:00, Time: 00:00:00

Finished in 0.41810s
5 tests, 5 assertions, 0 failures, 0 errors, 0 skips

imlbf:~/workspace (master) $
```

Test do Model de coleções, o qual garante que as coleções serão validas de acordo com os requisitos, tendo que ter obrigatoriamente os campos de nome e empresa\_id(ja que tem a referencia), tambem que nome tem q ser menor do que 140 caracteres, e que tem que ser destruído caso o seu pai(uma empresa) seja destruído tambem. Os testes também valem para validar o sistema de empresa ativa, já que no setup é feito a verificação da empresa ativa.

```
imlbf:~/workspace (master) $ rake test TEST=test/models/item_test.rb
Started

 4/4: [=====] 100% Time: 00:00:00, Time: 00:00:00

Finished in 0.45217s
4 tests, 4 assertions, 0 failures, 0 errors, 0 skips

imlbf:~/workspace (master) $
```

Test do model de Items, garante que os items serão validos de acordo com os seguintes requisitos, o nome tem que estar obrigatoriamente presente, juntamente com o collection\_id ja que pertence a uma coleção. tambem que a referencia tem

que ser única e estar presente. Os testes também valem para validar o sistema de empresa ativa, já que no setup é feito a verificação da empresa ativa.

```
imlbf:~/workspace (master) $ rake test TEST=test/controllers/collections_controller_test.rb
Started

  7/7: [=====] 100% Time: 00:00:00, Time: 00:00:00

Finished in 0.91687s
7 tests, 13 assertions, 0 failures, 0 errors, 0 skips

imlbf:~/workspace (master) $
```

Tests do controller de coleções, que garante que as funções de index, new, create, edit, destroy, update estão funcionando devidamente. Os testes também valem para validar o sistema de empresa ativa, já que no setup é feita a verificação da empresa ativa.

```
imlbf:~/workspace (master) $ rake test TEST=test/controllers/items_controller_test.rb
Started

  6/6: [=====] 100% Time: 00:00:00, Time: 00:00:00

Finished in 0.77964s
6 tests, 10 assertions, 0 failures, 0 errors, 0 skips

imlbf:~/workspace (master) $
```

Test do controller de itens, que garante que as funções de index, new, create, edit, destroy, update estão funcionando devidamente. Os testes também valem para validar o sistema de empresa ativa, já que no setup é feita a verificação da empresa ativa.

```
imlbf:~/workspace (master) $ rake test TEST=test/mailers/user_mailer_test.rb
Started

  2/2: [=====] 100% Time: 00:00:01, Time: 00:00:01

Finished in 1.21427s
2 tests, 16 assertions, 0 failures, 0 errors, 0 skips

imlbf:~/workspace (master) $
```

Test do envio de emails para resetar senha e ativação de conta.

```
imlbf:~/workspace (master) $ rake test
Started

  89/89: [=====] 100% Time: 00:00:02, Time: 00:00:02

Finished in 2.59523s
89 tests, 209 assertions, 0 failures, 0 errors, 0 skips

imlbf:~/workspace (master) $
```

Todos os testes unitários.

BDD:

### Teste Para pagina de coleções

```
Scenario: Acess collections page # features/colecoes.feature:3
  Given I m in my admin page on nobalk # features/step_definitions/colecoes_steps.rb:1
  When I press the button collections # features/step_definitions/colecoes_steps.rb:5
  Then I should see my collections list # features/step_definitions/colecoes_steps.rb:9
```

Entrando na pagina de inicial o usuário ao clicar no botão Coleções será redirecionado para a pagina de coleções.

### Troca de empresas

```
Scenario: Switch enterprises # features/trocaempresa.feature:3
  Given I am logged on Nobalk # features/step_definitions/trocaempresa_steps.rb:1
  Then I click in the button # features/step_definitions/trocaempresa_steps.rb:5
  And I choose another enterprise # features/step_definitions/trocaempresa_steps.rb:8
  Then I should see the switch of enterprises # features/step_definitions/trocaempresa_steps.rb:11
```

Entrando na pagina inicial o usuario ao clicar no botao de empresas e escolhendo a empresa sera redirecionado para a pagina da nova empresa.

### Avaliação dos membros:

**Ian Fireman:** Realmente ativo e produtivo durante toda criação e desenvolvimento das histórias, além de testes de verificação. Também deu suporte nas tarefas de documentação.

**Danilo Matheus:** Empenhou-se nas tarefas de design, testes de validação e revisões.

**Abhner Araujo:** Atuou nas taferas de documentação, gerenciamento e planejamento da iteração.

## **Conclusões**

- **A iteração correu sem maiores problemas, os requisitos designados para esta iteração foram completos.**
- **Nos aproximamos da etapa final e foi possível desenvolver a maior parte dos requisitos.**
- **A equipe pode, finalmente, utilizar a ferramenta Git produtivamente.**
- **Maior domínio sobre o rails e especificamente o banco de dados.**