



Universidade Estadual da Paraíba (UEPB)
Centro de Ciências e Tecnologia (CCT)
Curso Bacharelado em Ciências da Computação

Pedro Lucas Vaz Andrade

Otávio Ferreira Clementino

Wagner Tibúrcio da Silva Junior

Relatório de Release 02 - Funcionalidades

Campina Grande - 2025

1. Introdução	3
2. Funcionalidades Implementadas	3
2.1 Cadastro e Gerenciamento de Compradores	3
2.2 Sistema de Autenticação (Login de Comprador)	4
2.3 Carrinho de Compras	5
2.3.1 Estrutura de Dados	5
2.3.2 Funcionalidades do Carrinho	5
2.4 Integração entre Entidades	6
3. Interface com o Usuário (CLI)	6
4. Testes Realizados	7
5. Considerações Finais	7

1. Introdução

A Release 2.0.0 do sistema **Eng2Marketplace** representa uma evolução significativa em relação à primeira versão, tanto no aspecto funcional quanto arquitetural. Se na Release 1.0.0 o foco esteve no estabelecimento de uma base funcional mínima — com cadastro de produtos e lojas e persistência básica —, nesta nova versão foram adicionadas funcionalidades essenciais para viabilizar a operação de um marketplace real: cadastro e autenticação de compradores, implementação do carrinho de compras com controle de quantidade, vínculos entre entidades e validações de segurança.

O objetivo desta release foi preparar o sistema para operações transacionais entre compradores e lojas, por meio de uma abordagem organizada, modular e extensível. Com a estruturação do carrinho de compras e autenticação de usuários, damos os primeiros passos em direção a uma plataforma B2B robusta, segura e preparada para escalar em funcionalidades nas próximas versões.

2. Funcionalidades Implementadas

2.1 Cadastro e Gerenciamento de Compradores

Uma das maiores adições da Release 2.0.0 foi a introdução da entidade **Comprador**, representando usuários institucionais que realizarão compras na plataforma.

Recursos implementados:

- Interface para cadastro via terminal (CLI).
- Atributos do comprador:
 - **nome**: texto com validação mínima.
 - **cpf**: identificador único, obrigatório e validado.
 - **endereço**: local de entrega.
 - **senha**: campo obrigatório, não criptografado nesta versão.
 - **carrinho**: um **Map** para armazenar produtos e quantidades.

Validações incluídas:

- CPF duplicado é bloqueado no repositório, com exceção gerada.
- Formato mínimo de CPF é validado (sem regex ainda, planejado para a v2.1).
- A senha deve ter ao menos 3 caracteres.

Persistência:

- A entidade é salva no arquivo **compradores.json**.
- Em caso de alteração de dados, o arquivo é regravado por completo com as atualizações.

- Carrinhos de cada comprador são armazenados junto a seus dados para garantir persistência.

Essa funcionalidade cria a base para os módulos de login, carrinho e pedidos.

2.2 Sistema de Autenticação (Login de Comprador)

A partir da Release 2.0.0, compradores devem se autenticar para interagir com o sistema. Foi implementado um **mecanismo simples de login** por CPF e senha, simulando controle de sessão.

Como funciona:

- O comprador fornece seu CPF e senha.
- O sistema verifica a existência do CPF no repositório.
- Caso encontrado, valida a senha.
- Se os dados forem corretos, o comprador é “logado” e suas informações ficam disponíveis para o restante da sessão.

Benefícios:

- Garante que ações sensíveis (como adicionar ao carrinho) sejam feitas por um usuário autenticado.
- Prepara a base para controle de permissões e sessões futuras.

Desafios:

- Nesta versão, senhas ainda não são criptografadas.
- Sessão é simulada por variável em memória (futuro: JWT ou sessão real).

Pontos de melhoria para futuras versões:

- Criptografia com BCrypt.
- Controle de login simultâneo e logout.
- Restrições baseadas em perfis.

2.3 Carrinho de Compras

O **Carrinho de Compras** é uma das principais funcionalidades da Release 2.0.0, permitindo que compradores logados adicionem produtos com quantidades específicas, preparando o terreno para o futuro sistema de pedidos.

2.3.1 Estrutura de Dados

O carrinho é representado como:

```
private Map<String, Integer> carrinho;
```

Onde:

- A chave é o **ID** do produto.
- O valor é a **quantidade** desejada daquele produto.

Essa estrutura oferece excelente desempenho e permite controle fácil de duplicatas e quantidades.

2.3.2 Funcionalidades do Carrinho

- **Adicionar produto ao carrinho:**
 - Verifica se o produto existe no sistema.
 - Valida se há quantidade suficiente em estoque.
 - Se o item já estiver no carrinho, acumula a quantidade.
 - Atualiza o carrinho do comprador logado.
- **Visualizar carrinho atual:**
 - Mostra lista de produtos com nome, preço unitário, quantidade adicionada e total.
 - Soma total de itens e preço acumulado.
 - Agrupa por loja de origem, se necessário.
- **Persistência do carrinho:**
 - Quando o comprador é salvo, o carrinho vai junto em `compradores.json`.
 - Ao logar novamente, o carrinho do comprador é carregado.
- **Planejamentos futuros:**

- Remover item do carrinho.
- Atualizar quantidade diretamente.
- Finalizar compra e gerar pedido.

2.4 Integração entre Entidades

Nesta versão, a comunicação entre entidades foi significativamente aprimorada:

- Produtos são vinculados diretamente à loja por CPF/CNPJ.
- Carrinho está vinculado ao CPF do comprador.
- Controllers e Repositórios fazem validações cruzadas (ex: se loja existe ao criar produto).
- A arquitetura favorece a consistência dos dados e reutilização de lógica.

3. Interface com o Usuário (CLI)

A interface do sistema permanece em linha de comando, mas recebeu melhorias na usabilidade:

- Novo menu de **Login de Comprador**.
- Novo menu "**Área do Comprador**", com:
 - Visualizar produtos por loja.
 - Adicionar produto ao carrinho.
 - Ver carrinho atual com totais.

As ações são condicionadas à autenticação: apenas usuários logados podem acessar o carrinho e adicionar produtos.

4. Testes Realizados

Durante o ciclo de desenvolvimento da v2.0.0, foi realizada uma **bateria de testes manuais** para garantir o correto funcionamento das novas funcionalidades:

Testes executados:

- Cadastro de comprador com CPF válido e inválido.
- Login com senha correta e incorreta.
- Acesso negado para ações sem login.
- Adição ao carrinho com produto inexistente.
- Adição de produtos com estoque insuficiente.
- Validação da persistência após reiniciar o sistema.
- Carrinho refletindo corretamente quantidades acumuladas.

Os testes cobriram os seguintes componentes:

- **View:** menus e entrada de dados.
- **Controller:** regras de negócio para login, cadastro e carrinho.
- **Model:** estrutura e lógica de entidades.
- **Repository:** leitura, escrita e verificação em arquivos JSON.
- **Facade:** integração entre camadas.

Cobertura geral estimada: ~85% da base funcional foi testada manualmente. Testes automatizados com JUnit estão previstos para a v2.1.0.

5. Considerações Finais

A Release 2.0.0 entrega uma versão do sistema muito mais próxima de um ambiente funcional de marketplace. A introdução de compradores, login e carrinho transforma o Eng2Marketplace em uma plataforma interativa e personalizada, capaz de armazenar o contexto do usuário e adaptar o sistema conforme suas ações.

A arquitetura em camadas e uso do padrão **Facade** continuam proporcionando modularidade e coesão, facilitando tanto o desenvolvimento incremental quanto a manutenção e testes.

A próxima etapa do projeto será focada na **finalização da compra**. Isso inclui a criação de um módulo de **pedidos** que permitirá ao comprador converter os itens do carrinho em uma compra formal, com o respectivo **decremento do estoque** dos produtos adquiridos. Essa funcionalidade marca o primeiro ciclo completo de operação comercial dentro da plataforma.

Além disso, será feito um esforço específico para **lapidar e corrigir os problemas identificados na primeira sprint**, como validações incompletas, possíveis inconsistências em arquivos JSON e melhorias na navegação da interface via terminal. O objetivo será consolidar a estabilidade do sistema e preparar o terreno para novas funcionalidades que dependerão de um núcleo sólido e funcional.