



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

Pedro Lucas Vaz Andrade

Alan Medeiros Alves Gonçalves

Zayon Tavares Oliveira

Relatório de Release 03 - Processo de Desenvolvimento

Campina Grande - 2025

1. Introdução

A Sprint 5 teve início, marcando o início da Release 2 (Sprints 5 e 6) do projeto Eng2Marketplace, uma plataforma de marketplace que permite a interação entre compradores e vendedores com foco em transparência, qualidade de experiência e escalabilidade. Essa sprint teve como objetivo principal a implementação da funcionalidade de avaliação de compras, bem como a adaptação da equipe ao novo modelo de trabalho após mudanças na composição dos integrantes.

O escopo técnico envolveu desde a modelagem da entidade Avaliação, passando pela integração com as camadas de persistência (Repository), lógica de negócio (Service) e interface (View), até a atualização dos testes unitários e documentação técnica. Além disso, houve uma reestruturação estratégica da equipe, com a entrada de novos membros, exigindo um processo de onboarding eficiente e alinhamento técnico imediato.

2. Reestruturação da Equipe

Durante o planejamento da Sprint 5, ocorreu uma substituição parcial na equipe:

- Saídas: Otávio e Wagner
- Entradas: Zayon Tavares e Alan Medeiros

A nova formação ficou composta por:

- Pedro Andrade – Scrum Master / Gerente de Projeto e desenvolvedor backend.
- Zayon Tavares – Desenvolvedor frontend e responsável pela interação usuário-sistema.
- Alan Medeiros – Desenvolvedor backend, especializado em persistência de dados e lógica de negócios.

2.1. Onboarding Técnico

Para garantir a continuidade das entregas e minimizar o impacto da troca de membros, Pedro conduziu um processo estruturado de onboarding, abrangendo os seguintes tópicos:

- Apresentação do domínio de negócio e contexto do sistema.
- Explicação detalhada da arquitetura em camadas (MVC): Model, View, Controller.
- Revisão do código-fonte existente, especialmente relacionado às entidades Compra, Produto, Loja.
- Demonstração prática do fluxo de trabalho no Azure DevOps (PBI's, Tasks, Boards, Sprints).
- Treinamento sobre padrões de codificação adotados (convenções, nomenclatura, comentários).
- Exemplo prático de implementação de PBI concluída anteriormente (ex: cadastro de produtos).

Esse processo foi essencial para manter o ritmo de desenvolvimento e evitar retrabalho ou desalinhamento técnico.

3. Planejamento da Sprint

A Sprint 5 foi planejada com base nos critérios de aceite definidos no backlog do projeto, com destaque para o Requisito 07 (Req_07): "O comprador deve ser capaz de avaliar sua experiência de compra, atribuindo uma nota de 1 a 5 e um comentário opcional."

3.1. Objetivos da Sprint

- Implementar a funcionalidade de avaliação de compras.
- Garantir a integração com outras entidades (Compra, Loja, Produto).
- Atualizar testes unitários existentes.
- Documentar alterações arquiteturais e funcionais.

3.2. Itens do Backlog Selecionados

ID	Nome	Tipo	Estimativa (horas)
PBI_15	Avaliação da Compra	Funcional	40
PBI_16	Melhorias em Testes	Qualidade	10
PBI_17	Atualização de Documentação	Tarefa Técnica	8

Detalhe da PBI_15 - Avaliação da Compra

- Descrição: Permitir que um comprador registre uma avaliação para uma compra específica, contendo:
 - Nota numérica (1 a 5)
 - Comentário textual (opcional)
 - Data da avaliação
- Integrações:
 - Relacionamento com Compra
 - Atualização automática da média de avaliação na Loja
- Critério de Aceite:
 - Usuário pode avaliar apenas uma vez por compra.
 - A média da loja é recalculada após cada nova avaliação.
 - O histórico de avaliações é armazenado.

4. Arquitetura e Implementação Técnica

A implementação seguiu o padrão de arquitetura em camadas adotado pelo projeto, com divisão entre Model, Repository, Service, Controller e View.

4.1. Model - Entidade Avaliacao

```
public class Avaliacao
{
    public int Id { get; set; }
    public int ComprId { get; set; }
    public int Nota { get; set; } // 1 a 5
    public string? Comentario { get; set; }
    public DateTime DataAvaliacao { get; set; }
```

```
// Navegação
public virtual Compra Compra { get; set; }
}
```

Validações foram aplicadas:

- Validação da faixa de nota (1–5)
- Impedimento de múltiplas avaliações por mesma compra

4.2. Repository - AvaliacaoRepository

Implementado com métodos CRUD básicos e métodos específicos:

- AvaliarCompra(int comprald, int nota, string comentario)
- ObterMediaPorLoja(int lojald)
- ExisteAvaliacao(int comprald) → verifica se já existe avaliação

Utilizou-se o Entity Framework Core com padrão Repository para abstrair a persistência.

4.3. Service - AvaliacaoService

Camada intermediária que encapsula a lógica de negócios:

- Verifica se a compra pertence ao usuário autenticado
- Garante que a avaliação esteja dentro dos limites permitidos
- Invoca o repositório para salvar a avaliação
- Atualiza a média na entidade Loja

```
public void RegistrarAvaliacao(int comprald, int nota, string comentario)
{
    if (!CompraEhDoUsuario(comprald))
        throw new AutorizacaoException("Você só pode avaliar suas próprias compras.");

    if (nota < 1 || nota > 5)
        throw new ValidacaoException("Nota deve estar entre 1 e 5.");

    _repository.AvaliarCompra(comprald, nota, comentario);

    var lojald = _compraRepository.ObterLojaldDaCompra(comprald);
    var media = _repository.ObterMediaPorLoja(lojald);
    _lojaService.AtualizarMediaAvaliacao(lojald, media);
}
```

4.4. Controller - AvaliacaoController

Interface HTTP RESTful para registro de avaliação:

- Endpoint: POST /avaliacoes
- Payload esperado:

```
{
  "comprald": 1,
  "nota": 5,
  "comentario": "Excelente produto!"
}
```

4.5. View - Interface Console

Na versão console do sistema, foi implementada uma navegação intuitiva:

- Menu: "Minhas Compras"
- Opção: "Avaliar esta compra"
- Inputs: Nota (1-5), Comentário (opcional)

Foi utilizada validação de entrada para garantir a correta inserção de dados.

5. Integração com Outras Entidades

5.1. Compra

Uma avaliação está sempre vinculada a uma compra:

- Relacionamento 1:1
- Validação: só pode avaliar compras finalizadas

5.2. Loja

Cada avaliação influencia diretamente a reputação da loja:

- Média geral é calculada com base nas avaliações das compras associadas à loja
- Campo adicionado: MediaAvaliacao (decimal) na entidade Loja

5.3. Produto

Embora não seja o foco imediato, futuramente a avaliação será usada para calcular a média por produto também.

6. Testes Unitários

Foram criados e/ou atualizados testes unitários utilizando o framework xUnit:

- AvaliacaoServiceTest.cs
 - Testa avaliação válida
 - Testa tentativa de avaliar duas vezes a mesma compra
 - Testa validação de nota fora do intervalo
 - Testa atualização da média da loja

Também foram revisados testes anteriores de outras funcionalidades para garantir compatibilidade e cobertura adequada.

7. Documentação

Como parte da PBI_17, foram atualizados os seguintes documentos:

- Arquitetura do Sistema : incluída a descrição da entidade Avaliacao e sua relação com outras entidades.
- Diagrama ER : atualizado com a inclusão da tabela Avaliacaoes.
- Relatório de Funcionalidades : descrito o fluxo completo de avaliação de compras.
- Relatório de Processo de Desenvolvimento : este documento.

8. Ferramentas Utilizadas

- Azure DevOps – gestão de PBIs, Tasks, sprints e board Kanban
- Git + GitHub – versionamento do código

- jUNIT - Teste Unitários

9. Execução e Acompanhamento

A execução da sprint foi monitorada diariamente através do quadro Kanban no Azure DevOps, com as seguintes colunas:

- To Do
- In Progress
- Code Review
- Done

Foram realizadas reuniões diárias de 15 minutos (stand-ups) para:

- Alinhamento de tarefas
- Identificação de bloqueios
- Ajustes de prioridades

A produtividade da equipe permaneceu alta, mesmo com a entrada de novos membros, graças ao bom planejamento e comunicação clara.

10. Resultados Obtidos

- ☒ Entrega completa da PBI_15 – Avaliação de Compras
- ☒ Integração com Compra e Loja implementada
- ☒ Atualização da média de avaliação da loja
- ☒ Interface de usuário funcional
- ☒ Cobertura de testes unitários
- ☒ Atualização da documentação técnica

11. Impactos para a Sprint 6

A implementação da avaliação de compras na Sprint 5 fornece uma base robusta para a próxima sprint. As principais oportunidades de expansão são:

- Geração de relatórios de avaliação por loja e categoria
- Listagem de avaliações com filtros por nota e data
- Sistema de recomendação baseado em avaliações positivas
- Notificações automáticas para lojas com baixa média

Além disso, a estabilidade da equipe e o conhecimento acumulado permitem que a Sprint 6 tenha um maior foco em métricas, visualizações e otimizações de performance.

12. Conclusão

A Sprint 5 foi um sucesso tanto do ponto de vista técnico quanto organizacional. Mesmo enfrentando mudanças na equipe, o grupo manteve o foco nos objetivos, garantiu a entrega das funcionalidades previstas e consolidou uma base sólida para evolução do sistema. Com a implementação da funcionalidade de avaliação de compras, o Eng2Marketplace avança significativamente em direção a uma experiência realista e interativa, agregando valor tanto para compradores quanto para vendedores. A documentação detalhada, os testes unitários e o uso consistente de padrões arquiteturais garantem sustentabilidade e escalabilidade para futuras releases.

Gráfico Burndown:

