



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 6 marcou o encerramento da Release 03 do projeto Eng2Marketplace, consolidando as entregas iniciadas na Sprint 5 e finalizando a implementação de funcionalidades críticas para o marketplace. Esta sprint teve como foco principal a implementação do sistema de pontuação do comprador, que incentiva a interação contínua com a plataforma através de benefícios como frete grátis.

O escopo técnico envolveu desde a modelagem da entidade Pontuacao, 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 reorganização interna da equipe, com ajustes nas responsabilidades técnicas e no papel de liderança.

2. Reorganização da Equipe

Durante a planeamento da Sprint 6, ocorreu uma redistribuição de papéis dentro da equipe:

- Alan Medeiros : Passou a ocupar o papel de Gerente de Projeto, assumindo maior responsabilidade pela gestão das tarefas e alinhamento estratégico.
- Pedro Andrade : Continuou atuando como desenvolvedor backend, com foco em implementações complexas e revisão de código.
- Zayon Tavares : Permaneceu como desenvolvedor frontend, além de contribuir com a lógica de negócios e integrações.

Essa reorganização foi essencial para garantir um fluxo mais eficiente de trabalho, especialmente considerando os desafios técnicos introduzidos pelo novo requisito de pontuação.

3. Planejamento da Sprint

A Sprint 6 foi planejada com base nos critérios de aceite definidos no backlog do projeto, com destaque para o Requisito 08 (Req_08): "O comprador possui uma pontuação que lhe assegura vantagens, tipo frete grátis, por exemplo."

3.1 Objetivos da Sprint

- Implementar o sistema de pontuação do comprador.
- Integrar a pontuação ao processo de compra e avaliação.
- Atualizar testes unitários existentes.
- Documentar alterações arquiteturais e funcionais.

3.2 Itens do Backlog Selecionados

ID	Nome	Tipo	Estimativa (horas)
PBI_19	Pontuação Usuário	Funcional	40

PBI_20	Sistema Bonificação/Pontuação	de	Funcional	30
PBI_18	Documentação		Tarefa Técnica	10

Detalhes da PBI_19 - Pontuação Usuário

- Descrição : Permitir que o comprador acumule pontos com cada compra e avaliação realizada, recebendo benefícios conforme sua pontuação.
- Integrações :
 - Relacionamento com Compra.
 - Atualização automática da pontuação após cada compra ou avaliação.
- Critério de Aceite :
 - O comprador deve receber +10 pontos por compra e +5 pontos por avaliação.
 - Benefícios devem ser aplicados automaticamente na hora da compra.
 - A pontuação deve ser persistida e exibida na interface do usuário.

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 Pontuacao

```
public class Pontuacao
{
    public int Id { get; set; }
    public int CompradorId { get; set; }
    public int Pontos { get; set; }
    public DateTime DataAtualizacao { get; set; }

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

Validações foram aplicadas:

- Garantia de que a pontuação seja sempre positiva.
- Persistência automática de histórico de pontuação.

4.2 Repository - PontuacaoRepository

Implementado com métodos CRUD básicos e métodos específicos:

- AdicionarPontos(int compradorId, int pontos)
- ObterPontuacaoPorComprador(int compradorId)
- AplicarBeneficio(int compradorId)

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

4.3 Service - PontuacaoService

Camada intermediária que encapsula a lógica de negócios:

- Verifica se o comprador está elegível para receber pontos.
- Invoca o repositório para salvar a pontuação.
- Aplica benefícios automaticamente na hora da compra.

```
public void RegistrarCompra(int compradorId)
{
    var pontos = 10;
    _repository.AdicionarPontos(compradorId, pontos);
    _repository.AplicarBeneficio(compradorId);
}
```

4.4 Controller - PontuacaoController

Interface HTTP RESTful para gerenciamento de pontuação:

- Endpoint: GET /pontuacoes/{compradorId}
- Endpoint: POST /pontuacoes/adicionar

4.5 View - Interface Console

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

- Menu: "Minha Pontuação"
- Exibe a pontuação atual e benefícios disponíveis.

5. Integração com Outras Entidades

5.1 Compra

Cada compra gera +10 pontos para o comprador:

- Relacionamento direto com a entidade Compra.
- Validação: apenas compras finalizadas geram pontos.

5.2 Benefício

Os benefícios são aplicados automaticamente na hora da compra:

- Faixa de pontuação:
 - Até 50 pontos: Sem benefício.
 - De 51 a 100 pontos: Frete reduzido (-30%).
 - Acima de 100 pontos: Frete grátis.

6. Testes Unitários

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

- PontuacaoServiceTest.cs: Testa atribuição de pontos, validação de benefícios e integração com outras entidades.
- CompraServiceTest.cs: Verifica a geração automática de pontos após uma compra.

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

7. Documentação

Como parte da PBI_18, foram atualizados os seguintes documentos:

- Arquitetura do Sistema : Incluída a descrição da entidade Pontuacao e sua relação com outras entidades.
- Diagrama ER : Atualizado com a inclusão da tabela Pontuacoes.
- Relatório de Funcionalidades : Descrito o fluxo completo de pontuação e benefícios.
- 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.
- xUnit : 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, graças ao bom planejamento e comunicação clara.

10. Resultados Obtidos

✓ Entrega completa das funcionalidades previstas:

- Implementação do sistema de pontuação.
- Integração com o processo de compra e avaliação.
- Aplicação automática de benefícios na hora da compra.
- Interface de usuário funcional.
- Cobertura de testes unitários.
- Atualização da documentação técnica.

11. Considerações Finais

A Sprint 6 foi um sucesso tanto do ponto de vista técnico quanto organizacional. Mesmo com mudanças nas responsabilidades da equipe, conseguimos manter o foco nos objetivos, garantindo a entrega das funcionalidades previstas e consolidando uma base sólida para futuras evoluções do sistema.

Com a implementação do sistema de pontuação, o Eng2Marketplace avança significativamente em direção a uma experiência mais engajadora e fidelizada, incentivando os compradores a interagir continuamente com a plataforma. A documentação detalhada, os testes unitários robustos e o uso consistente de padrões arquiteturais garantem sustentabilidade e escalabilidade para futuras releases.

Agradecemos a todos os membros da equipe pelo comprometimento, dedicação e colaboração durante todo o projeto. Esse trabalho reflete não apenas o aprendizado técnico adquirido na disciplina de Engenharia de Software II, mas também habilidades valiosas de gestão, comunicação e resolução de problemas.

Este relatório marca o encerramento oficial da Release 03 do Eng2Marketplace, concluindo o ciclo completo de compra → avaliação → pontuação → benefícios. Com isso, damos por concluído o projeto, orgulhosos do que construímos e preparados para novos desafios!