

Relato de Experiência do Desenvolvimento do Sistema de Gerenciamento de Fretes para uma Transportadora de Pequeno Porte

Arthur P. Teruel¹, Gregory S. Ferreira¹, Isaque D. P. da Cruz¹, Leandro Menoni¹,
Pedro H. L. da Silva¹, Frank W. C. de Oliveira¹, Marcelo F. Terenciani¹

¹Campus Paranavaí – Instituto Federal do Paraná (IFPR)
Paranavaí – PR – Brasil

{20230006488, 20230006227, 20230006245,
20230006352, 2023000616}@estudantes.ifpr.edu.br
{frank.willian, marcelo.terenciani}@ifpr.edu.br

1. Introdução

As disciplinas de Construção de *Software* e Projeto Integrador II fomentam a formação prática de engenheiros de software por meio do desenvolvimento de sistemas para clientes reais, com metodologias e ferramentas também utilizadas no mercado de trabalho, tais como o *Scrum* [Schwaber and Sutherland 2020] e o *Kanban* [Anderson 2015]. Nessas disciplinas, os estudantes são desafiados a desenvolver um *software* para resolver um problema real durante o ano letivo, dividido por ciclos chamados de *Sprint*. *Sprint* é um termo utilizado no Scrum para designar um ciclo de desenvolvimento com duração fixa (geralmente entre 1 e 4 semanas), no qual uma equipe trabalha de forma intensiva para entregar um incremento de produto funcional. Cada *sprint* possui planejamento, execução, revisão e retrospectiva [Schwaber and Sutherland 2020].

Em cada *sprint* foram seguidas as cerimônias definidas pelo *Scrum*, onde acontecem o alinhamento com o cliente, planejamento, apresentação do produto e revisão dos resultados da equipe. Executando essas ações de forma cíclica, estimula-se a participação constante do cliente no desenvolvimento, o que permite manter o *software* alinhado às suas expectativas e novas necessidades [Schwaber and Sutherland 2020].

Nesse contexto, a aplicação do *Scrum* se mostrou particularmente adequada à realidade da empresa atendida no projeto. A empresa atendida neste projeto é a Teruel Express, uma transportadora de criação recente cuja presença digital era limitada a um contato via *WhatsApp*. Essa ausência de uma plataforma própria comprometia sua eficiência operacional, dificultava o atendimento aos clientes e reduzia sua competitividade no mercado.

2. Objetivo

Considerando as limitações identificadas, o projeto propôs o desenvolvimento de um sistema *web* para gerenciar orçamentos e entregas, automatizando processos da transportadora. Além disso, a iniciativa buscou proporcionar aos estudantes experiência prática em desenvolvimento *full-stack* e na aplicação de metodologias ágeis [Beck et al. 2001]. O resultado final foi entregar uma solução funcional, construída a partir das demandas do cliente e validada com sua participação contínua ao longo do processo de desenvolvimento.

O projeto envolveu cinco alunos do curso de Engenharia de Software, realizado no campus Paranavaí do IFPR, sob orientação de professores. A proposta era desenvolver o sistema para a transportadora Teruel Express, com prazo de um ano letivo, com duas *Sprints* por trimestre. O cliente participou ativamente, fornecendo *feedback* contínuo durante as cerimônias do *Scrum*.

Cada *Sprint* teve duração aproximada de um mês, com reuniões de planejamento, revisão e retrospectiva. Dessa forma foi possível fazer entregas incrementais e ajustes constantes com base no *feedback* do cliente. Em cada *Sprint* o foco estava em entregar funcionalidades específicas (*Sprint Backlog*), como o módulo de orçamentos e o painel de administração.

3. Desenvolvimento

Os papéis dentro da equipe foram organizados conforme o *Scrum*, com a composição de um *Product Owner*, responsável por transcrever as necessidades do cliente por meio de histórias de usuário e defender os interesses do cliente junto à equipe, três desenvolvedores que implementam as tarefas especificadas, e um *Scrum Master* com a função de remover impedimentos e fazer a gestão do projeto. Além disso, a equipe fez uso de um quadro *Kanban* no *GitHub* para gerenciar tarefas.

As funcionalidades foram levantadas e especificadas pelo *Product Owner*, que conduziu entrevistas com o cliente para entender suas necessidades. Protótipos e demonstrações do sistema foram apresentados em cada *Sprint* para validação. Esse processo iterativo garantiu que o sistema atendesse às expectativas e incorporasse melhorias sugeridas.

Foram utilizadas diversas tecnologias no desenvolvimento. No *front-end*¹, foi utilizado *React* com *Vite* para criação das interfaces de usuário, previamente prototipadas no *Figma*. As interfaces foram divididas em uma página institucional com área autenticada e um painel administrativo. O *back-end*² foi desenvolvido com *Spring Boot* e banco de dados *PostgreSQL*.

Para este projeto foram implementadas duas aplicações *web*, a primeira dedicada a clientes da Teruel Express, trata-se de uma página pública³ integrada a uma área autenticada, onde os clientes podem solicitar fretes e acompanhar os seus pedidos. Já o segundo sistema⁴ tem o propósito de gerenciar estas solicitações, sendo de uso exclusivo da transportadora. Atualmente o projeto conta com o cadastro de clientes, autenticação e avaliações já implementados por completo, por outro lado, funcionalidades mais complexas como as que envolvem administração de pedidos de fretes e rotas atendidas ainda estão em fase de desenvolvimento.

4. Experiências

Desenvolver este projeto trouxe diversas vivências para a equipe, tanto positivas quanto negativas. A equipe experienciou pela primeira vez como é o processo de desenvolver

¹<https://bit.ly/42H1mWK> e <https://bit.ly/471AXnK>

²<https://bit.ly/42Csiqx>

³<https://teruelexpress.vercel.app/>

⁴<https://admin-teruelexpress.vercel.app/>

um produto para um cliente real, fazendo com que colegas de turma colaborassem como colegas de trabalho em uma empresa. A atribuição de papéis distintos a cada membro favoreceu o desenvolvimento individual em suas áreas e proporcionou uma compreensão prática do trabalho em equipe e suas nuances, evidenciando como os esforços de diferentes pessoas podem tanto se complementar quanto, em certos casos, se anular.

Um exemplo dessa dinâmica ocorreu quando a especificação inadequada de histórias pelo *Product Owner*, somada à não consideração dos pedidos do *Scrum Master* e dos desenvolvedores por maior detalhamento das tarefas, resultou em atrasos e conflitos internos, demandando intervenção dos professores. Além disso, o recorrente desempenho abaixo do esperado de um dos desenvolvedores, gerou acúmulo de dívidas técnicas em determinadas *Sprints* que precisaram ser compensadas pelos demais membros da equipe. A comunicação com o cliente também se mostrou desafiadora, pois, embora o protótipo tenha sido validado antes da implementação, alguns requisitos só foram informados após o desenvolvimento da funcionalidade, sendo revelados apenas quando o cliente a testou já implementada.

Apesar dos desafios enfrentados, a experiência prática mostrou-se extremamente enriquecedora. Após a resolução dos conflitos, a comunicação dentro da equipe melhorou significativamente, não sendo mais necessária qualquer intervenção externa. Vale ressaltar também que os desenvolvedores, juntamente com o *Scrum Master*, aprimoraram suas habilidades para o desenvolvimento colaborativo.

5. Considerações finais

Por fim, o projeto integrador incentiva a iniciativa dos alunos em aprender novos conceitos e aplicá-los, assim, a equipe aproveitou os conhecimentos adquiridos na disciplina de Interação Humano-Computador para implementar melhorias na interface, visando aprimorar a experiência do usuário final.

O projeto integrador proporcionou à equipe uma experiência prática valiosa, aplicando metodologias ágeis em um ciclo real de desenvolvimento de software para um cliente, destacando a importância da comunicação clara, colaboração e detalhamento de tarefas para entregar um produto de qualidade. Enfrentar desafios como conflitos internos, dívidas técnicas e mudanças de requisitos estimulou a resolução de problemas, adaptação e aprimoramento de habilidades, enquanto a aplicação de conceitos de Interação Humano-Computador melhorou a interface do sistema, consolidando competências essenciais em Engenharia de Software, como desenvolvimento colaborativo, planejamento ágil e integração entre teoria e prática.

Referências

- Anderson, D. J. (2015). *Kanban: sucesso evolutivo com o método ágil*. Casa do Código, São Paulo.
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001). Manifesto for agile software development. Acesso em: 20 set. 2025.
- Schwaber, K. and Sutherland, J. (2020). *O Guia do Scrum: O guia definitivo para o Scrum: As regras do jogo*. Scrum.org. Acesso em set. 2025.