Desenvolvimento de uma Aplicação para Gestão de Ordens de Serviço de uma Assistência Técnica de Balanças

Arthur H. V. Shiroshima, Felipe F. L. Victorino, Giovanna B. de S. Goes, Isaque C. Xavier, Maria E. F. Bruning, Murilo Rolim, Evanise A. C. Ruiz, Frank W. C. Oliveira, Marcelo F. Terenciani

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

20241pvai100300{03, 41, 20, 12, 29, 06}@estudantes.ifpr.edu.br
{evanise.ruiz, frank.willian, marcelo.terenciani}@ifpr.edu.br

No âmbito do curso de Bacharelado em Engenharia de Software, ofertado pelo Instituto Federal do Paraná (IFPR) – Campus Paranavaí, é ministrado o componente disciplinar de Projeto Integrador. Esta matéria possui caráter multidisciplinar, abrangendo disciplinas como: Arquitetura e Padrões de Sistemas; Construção de Software; Interação Humano-Computador; Programação Web e Desenvolvimento para Dispositivos Móveis, em um único projeto em equipe.

O projeto alinha os conceitos teóricos e práticos das disciplinas e aplica-os em contextos reais para a resolução de problemas práticos envolvendo pessoas e processos reais, como previsto no Projeto Pedagógico do Curso [IFPR 2023]. Para a realização do projeto, foi escolhida uma empresa que atua no ramo de avaliação, manutenção e venda de balanças comerciais, agrícolas e industriais. Atualmente, os registros das ordens de serviço desta empresa são feitos em papel, enquanto a comunicação com clientes e coordenação interna são feitas por meio do WhatsApp, descentralizando as operações e dificultando o gerenciamento da empresa.

Diante deste cenário, esse trabalho tem como objetivo apresentar os resultados parciais do processo de desenvolvimento de uma solução multiplataforma para centralização dos serviços da empresa, considerando o escopo de assistência técnica de balanças e o gerenciamento de ordens de serviços. Espera-se que a solução reduza o esforço administrativo, crie condições necessárias para crescimento, além de promover a organização da empresa.

A metodologia de desenvolvimento baseou-se na metodologia *Scrum* [Schwaber e Sutherland 2020] modificado pelos docentes conforme limitações acadêmicas e utilizando o sistema de quadro *Kanban*. A estrutura da equipe consistiu em seis pessoas: um *Scrum Master* (SM), um *Product Owner* (PO) e quatro desenvolvedores. Para o projeto e adesão aos papéis do *Scrum*, utilizou-se o *GitHub Projects* para a gestão de tarefas e Histórias de Usuário (HUs).

O PO responsabilizou-se pela elaboração de HUs, organização e planejamento das tarefas previstas e comunicação com o parceiro especialista do domínio para a extração de requisitos, utilizando principalmente o método da entrevista para entender o perfil e necessidades do usuário. O SM encarregou-se pela avaliação de código e produtividade usando a plataforma GitHub. Os desenvolvedores foram responsáveis pela criação do código-fonte, considerando necessidades técnicas do *Scrum* e o versionamento perante a ferramenta Git e Github.

Para a revisão da produtividade da equipe, foram utilizadas ferramentas como o gráfico *burndown*, que relaciona o nível de esforço numérico necessário para conclusão da *sprint* com o período de tempo de conclusão. O esforço total é mensurado pela unidade de esforço, definida pelo desenvolvimento de um formulário de dois campos, um botão, com estilização presente e sem persistência de dados. O esforço de uma tarefa é atribuído por meio do *Planning Poker*, uma ferramenta utilizada para classificar as tarefas com base no seu esforço. A Tabela 1 apresenta a síntese dos total de esforço definido durante as reuniões das *sprints* concluídas, elaborada com dados extraídos do repositório do projeto[1].

Tabela 1. Esforço total das tarefas.

Sprint	**Período**	**Esforço total**	*Commits* totais	***PR*s totais**	**Total de HUs**
1	20/03 - 05/05	15	103	21	10
2	22/05 - 03/07	24,5	125	18	14
3	31/07 - 28/08	25	55	9	9
4	11/09 - 09/10	31	191	23	20

Fonte: Dados extraídos do repositório do projeto na primeira quinzena de outubro de 2026.

As atividades de desenvolvimento se iniciaram em fevereiro de 2026, com a realização de uma *sprint* inicial, cuja entrega foi uma página *web* para o projeto[2]. Os requisitos e instruções referentes à página foram escritos previamente pelos docentes avaliadores, porém todas as HUs subsequentes foram escritas pelo PO. Após o término da *sprint* inicial, realizou-se uma reunião com o parceiro para definição do escopo e funcionalidades do sistema.

Inicialmente, outra empresa que atua no ramo de desenvolvimento de sistemas de planejamento de recursos empresariais (ERP) havia sido escolhida como parceira para a elaboração do projeto, porém, por questões internas, esta empresa desistiu do projeto. Este fato aconteceu no primeiro dia da *sprint* 1, onde já existia um planejamento definido e HUs escritas, o que estagnou o progresso até o estabelecimento de uma nova parceria. Isso obrigou a equipe a replanejar o projeto, reutilizar o que já havia sido criado e restringir o desenvolvimento à telas relacionadas ao fluxo de autenticação.

Na sequência, com a empresa parceira redefinida, uma nova série de requisitos foram formulados e classificados. Entre as necessidades identificadas destacam-se a disponibilidade do sistema para dispositivos móveis e a sincronização entre os dados de múltiplos dispositivos, o que justificou a escolha do *framework* Flutter para a construção da interface gráfica. Observa-se que a escolha do Flutter e a linguagem de programação Dart também foi tomada a fim de atender a ementa da disciplina de Desenvolvimento para Dispositivos Móveis [IFPR 2023].

As duas *sprints* subsequentes abrangeram a implementação das interfaces gráficas do sistema, tendo em foco os cadastros de clientes, endereços e peças no sistema. O formulário em papel utilizado pela empresa foi usado como base para a formação do fluxo de cadastro. Durante a *sprint* 3, houve o início da criação do fluxo do cadastro de ordens de serviço e na *sprint* seguinte, a finalização da interface e a implementação parcial da persistência, resultando em uma dívida de tarefa para outra *sprint*.

[1]https://github.com/CS-PI-2026-Argonautas/front-end/

[2]https://cs-pi-2026-argonautas.github.io/landing-page/index.html

Durante o decorrer da terceira *sprint*, também foram discutidos detalhes sobre a arquitetura de persistência no servidor, levando em conta a regra de negócio relativa ao uso do aplicativo sem conexão com a internet e sincronização de dados, a discussão continuou até o início da próxima *sprint*. Foram consideradas várias possibilidades de implementação, como a implementação manual da sincronização de dados e o uso de serviços de terceiros. Durante a quarta *sprint*, optou-se o uso do Firebase Firestore como serviço de armazenamento de dados externo, que já oferece o armazenamento local e sincronização.

Como mostra a Tabela 1, houve um aumento gradual da quantidade de tarefas totais, salvo a *sprint* 3, e do esforço tomado pela equipe ao longo das *sprints*. Este fato indica a evolução da equipe e o aumento da confiança dos desenvolvedores quanto às tarefas propostas. A quantidade menor de *commits* e *pull requests* na *sprint* 3 é observada devido à densidade das tarefas, visto que as tarefas que abrangiam esta *sprint* consistiam na implementação da interface e regras de negócio referentes ao fluxo de ordem de serviço e a então configuração do banco de de dados antes da definição do Firestore.

Foram encontradas limitações durante o desenvolvimento do projeto, como a impossibilidade de imersão total no projeto por conta de demandas de outras disciplinas e pessoais dos integrantes. Além disso, a falta de experiência acerca das ferramentas e processos utilizados teve um grande impacto no início das atividades, visto que o desenvolvimento ocorreu ao mesmo tempo que os integrantes da equipe aprendiam sobre o conteúdo necessário para a produção no trabalho.

Contudo, com a evolução das *sprints*, evidenciou-se o progresso dos integrantes da equipe quanto a aderência de padrões do desenvolvimento, ambos mensurados com a análise do *Kanban*, a produtividade com o uso da linguagem Dart e o Flutter e o gráfico *burndown*. A proposta da disciplina está sendo atendida visto a aplicação da multidisciplinaridade para o aprendizado de conceitos como: gestão de tarefas e priorização de tarefas, liderança de equipe, trabalho em equipe, o uso do *Scrum* na prática, aplicação do versionamento de código em um projeto grande pela primeira vez e aprendizado da linguagem Dart e Flutter em um contexto prático.

Atualmente, o aplicativo encontra-se em desenvolvimento e não está em estado funcional para o uso pelo parceiro. Porém, é planejado até o final da *sprint* 5 a finalização do cadastro de serviços, o conserto de falhas referentes à usabilidade e identidade visual, a persistência de dados e a implementação de segurança para validação em ambiente real. Para trabalhos futuros, visa-se o teste do projeto em produção, analisando e validando aspectos funcionais do sistema, a correção de possíveis erros de implementação e possivelmente a disponibilização do serviço para outros parceiros.

Referências

IFPR (2023). Projeto Pedagógico do Curso de Bacharelado em Engenharia de Software. Disponível em: https://ifpr.edu.br/paranavai/wp-content/uploads/sites/21/2025/01/PPC-de-Engenharia-de-Software-%E2%80%93-2025.pdf Acesso em: 19 ago. 2026. Instituto Federal do Paraná (IFPR), Campus Paranavaí.

Schwaber, K. e Sutherland, J. (2020). The scrum guide: The definitive guide to scrum - the rules of the game. Disponível em: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf. Acesso em: 19 ago. 2026.