

## Notify: Desenvolvimento de um Software para Comunicados Institucionais com Adaptação do SCRUM em Sala de Aula

Carla Oliveira<sup>1</sup>, Cauê Yanagihara<sup>1</sup>, Eduardo Cavazin<sup>1</sup>, Rafael Sanches<sup>1</sup>,  
Vitor Tavares<sup>1</sup>, Marcelo F. Terenciani<sup>1</sup>, Frank Willian C. de Oliveira<sup>1</sup>

<sup>1</sup>Instituto Federal do Paraná - Campus Paranavaí (IFPR)

{carlasdeoliveiraf, caueyanagihara, eduardotorcav, rafaelsanches.s35, vtavares.eng}@gmail.com, {marcelo.terenciani, frank.willian}@ifpr.edu.br

Em uma sociedade caracterizada pela presença do conhecimento e da informação, diversas tecnologias têm promovido diferentes impactos no cotidiano, como nas formas de pensar, agir e se comunicar [da Silva Rocha et al. 2019]. Atualmente, as Tecnologias da Informação e Comunicação (TIC) estão presentes em diversos âmbitos, inclusive na área da educação, em que percorrem desde o processo de aprendizagem até a relação família-escola [Martins 2017].

Com a inserção de meios tecnológicos nas instituições de ensino, a comunicação entre a escola e os responsáveis também é afetada [Martins 2017]. Tradicionalmente, essa comunicação é realizada por meio de bilhetes impressos, reuniões presenciais ou por ligações telefônicas, métodos que, embora funcionais, apresentavam limitações, como atrasos na entrega de mensagens e dificuldades de registro e/ou acompanhamento.

Embora existam diversas ferramentas, como o WhatsApp [Meta 2024] e outras plataformas de mensagens instantâneas, elas não são totalmente adequadas para atender às especificidades do ambiente escolar. Essas ferramentas geralmente são limitadas na gestão, como o agrupamento de mensagens por turma, curso, responsáveis ou destinatário individual, além de relatórios de envio e confirmação de leitura. Além disso, o uso de aplicativos genéricos pode gerar problemas de privacidade, controle de acesso, sobrecarga de informações, dificultando a comunicação eficiente.

Neste sentido, este trabalho propõe o desenvolvimento de um sistema específico, denominado Notify, para mediar a comunicação entre família e escola, oferecendo funcionalidades alinhadas às necessidades institucionais do IFPR - Campus Paranavaí. Esse software foi desenvolvido como atividade prática das disciplinas de Projeto Integrador II e Construção de Software, do 3º ano do curso de Bacharelado em Engenharia de Software.

O Notify está organizado em duas partes: uma *web*, direcionada aos servidores da instituição de ensino, que deve ser utilizada para a escrita e envio dos comunicados; e outra *mobile*, destinada aos responsáveis dos estudantes, para que eles recebam os comunicados emitidos pela instituição. O envio pode ser realizado para todos os responsáveis, agrupados por curso e turma ou de maneira individual.

Para o desenvolvimento, conforme proposto pelos docentes responsáveis, a metodologia utilizada foi adaptada do SCRUM, um *framework*<sup>1</sup> ágil utilizado no desenvolvimento de softwares. O SCRUM promove uma abordagem colaborativa e iterativa,

<sup>1</sup>Estrutura de processos e práticas de trabalho. Neste contexto, orienta como uma equipe deve colaborar para alcançar objetivos de maneira ágil, priorizando a flexibilidade e a adaptação ao longo do projeto.

facilitando a adaptação às mudanças e o aprimoramento contínuo, sendo essas características fundamentais em ambientes de aprendizado dinâmicos e em constante evolução [Schwaber and Sutherland 2020].

O trabalho foi desenvolvido por uma equipe de cinco integrantes, dentre os quais haviam um *Product Owner* (PO), um *Scrum Master* e três desenvolvedores. Ao longo da execução das tarefas, organizadas em *sprints* com duração média de um mês, foram realizados encontros de acompanhamento, além de reuniões de planejamento, revisão e retrospectiva, inspiradas nos eventos do SCRUM.

Para auxiliar a organização, foi utilizado o GitHub Projects para gerenciamento das tarefas do projeto, viabilizando a existência do *Product Backlog*<sup>2</sup> e *Sprint Backlog*<sup>3</sup>, com suporte de um quadro Kanban digital. Também foi utilizada a plataforma Discord como o principal meio de comunicação entre a equipe, permitindo troca de mensagens e chamadas de voz. Para o versionamento do código, utilizou-se o Git aliado ao GitHub, assegurando que as alterações fossem documentadas e integradas de forma organizada.

No desenvolvimento do software em sua parte *web*, foram adotadas as tecnologias React JS e Spring Boot, que estavam alinhadas às disciplinas já cursadas pelos membros da equipe. Essa escolha não apenas garantiu a coesão entre as diferentes partes do projeto, mas também permitiram que a equipe aproveitasse o conhecimento prévio, resultando em um desenvolvimento mais eficiente e integrado.

Conforme estipulado pelos docentes responsáveis, o projeto deveria possuir um cliente e uma demanda real. Para tanto, o cliente escolhido foi o Diretor Geral do Campus Paranavaí, a partir da apresentação da necessidade de um sistema de gerenciamento de comunicados para os responsáveis dos estudantes. Essa demanda foi transformada em histórias de usuário e incluída no *Product Backlog*, garantindo que o software desenvolvido fosse relevante e atendesse às necessidades do solicitante.

Durante a primeira *sprint*, foi estipulada a criação de uma *landing page* para a divulgação do projeto, utilizando HTML, CSS e JavaScript, sem o auxílio de *frameworks*<sup>4</sup>. As histórias de usuário foram inicialmente definidas pelos professores, a fim de oferecer um direcionamento aos alunos. Mas, ao longo do processo, o PO e o Scrum Master perceberam que era viável realizar mudanças e adicionar novas histórias, que foram projetadas e implementadas pelos membros da equipe de desenvolvimento, permitindo maior flexibilidade e adaptação às necessidades do projeto.

A decisão de focar na criação do site durante a primeira *sprint* objetivou evitar a ociosidade da equipe de desenvolvimento, visto que o PO havia sido designado recentemente e ainda precisava definir as histórias em concordância com o cliente. Nesse momento, foram traçados rumos de desenvolvimento que permitiram iniciar uma primeira impressão e um entendimento do fluxo de trabalho, essa abordagem garantiu que a equipe pudesse avançar, mesmo em meio à definição das diretrizes do projeto.

<sup>2</sup>Lista priorizada de funcionalidades desejadas, correções e outras demandas necessárias para o produto. Pode ser atualizado conforme o surgimento de novas necessidades e mudança de prioridades.

<sup>3</sup>Lista de tarefas derivada do Product Backlog, composto pelos itens definidos pela equipe de desenvolvimento compromete-se a entregar ao final da sprint.

<sup>4</sup>Modelos de desenvolvimento voltados para a construção de aplicações. Neste contexto, refere-se a um conjunto de bibliotecas e ferramentas de código que aceleram e auxiliam no desenvolvimento.

A partir da segunda *sprint*, o foco foi a implementação da interface gráfica (*front-end*) do Notify. As histórias de usuário foram refinadas e priorizadas. Para definir as tarefas a serem realizadas, foi realizada uma reunião de planejamento utilizando o método de *Planning Poker*. Essa técnica é uma forma eficaz de estimar o esforço necessário para cada história, permitindo que a equipe tenha uma visão clara do escopo do projeto e das metas a serem alcançadas em cada *sprint* [Mahnič and Hovelja 2012].

Ao final da segunda *sprint*, todo o *sprint backlog* foi implementado e foi realizada uma *Sprint Review*, permitindo que o cliente visualizasse o progresso do projeto e fornecesse *feedback* sobre o trabalho realizado. Essa interação foi essencial para garantir o atendimento das expectativas do cliente e alinhamento às suas necessidades, além de viabilizar a identificação de erros ou divergências e a realização de correções.

Antes do início da terceira *sprint*, foi realizada uma reunião de retrospectiva, na qual a equipe refletiu sobre o trabalho realizado até o momento, identificando pontos fortes e áreas de melhoria. Essa prática permitiu que a equipe ajustasse suas estratégias e processos, visando aprimorar o desempenho e a eficiência do projeto. A partir dessas reflexões, foram estabelecidas metas e ações para a próxima *sprint*, garantindo que o projeto continuasse a evoluir de maneira consistente e alinhada às expectativas do cliente.

Durante a terceira *sprint*, o foco foi direcionado para a implementação da estrutura interna das interfaces gráficas criadas, além da inclusão de novas funcionalidades. As histórias de usuário foram refinadas e priorizadas com base nas necessidades do cliente. Para definir as tarefas a serem executadas, a equipe conduziu uma reunião de planejamento, novamente utilizando o método de *Planning Poker*.

Ao final da terceira *sprint*, todas as histórias de usuário selecionadas para desenvolvimento foram implementadas ou parcialmente concluídas. Contudo, a equipe não conseguiu implementar as funcionalidades relacionadas à autenticação de usuários do sistema, devido a atrasos em uma disciplina correlata. As tarefas pendentes foram adicionadas ao *product backlog*, com a intenção de serem abordadas nas próximas *sprints*. Dessa forma, O Notify está parcialmente funcional. Pretende-se realizar a implementação das funcionalidades de autenticação de usuários e a finalização do *front-end* e *back-end* do sistema, bem como desenvolver a sua versão *mobile*.

## Referências

- da Silva Rocha, P. C., Jucá, S. C. S., and da Silva, S. A. (2019). A evolução das tecnologias da informação e comunicação na perspectiva de touraine, bell e castells. *Research, Society And Development*, 8(5):01–12.
- Mahnič, V. and Hovelja, T. (2012). On using planning poker for estimating user stories. *Journal of Systems and Software*, 85(9):2086–2095.
- Martins, F. (2017). Novas tecnologias e comunicação escola-pais: implicações para uma escola democrática e interface com o currículo oculto. *Revista Tempos e Espaços em Educação*, 10(23):87–97.
- Meta (2024). Whatsapp official website. <https://www.whatsapp.com/>. Último acesso: 01-11-2024.
- Schwaber, K. and Sutherland, J. (2020). *O Guia do Scrum – O guia definitivo para o Scrum: As regras do jogo*. ScrumGuides.org. Último acesso: 01-11-2024.