

# Recanto do Guerreiro

Gabriel Alejandro Figueiro Galindo<sup>1</sup>, João Gabriel Perez Monteiro<sup>2</sup>,  
Pedro Henrique Moreira Caixeta Ferreira<sup>3</sup>

<sup>1</sup>Instituto de Ciências Exatas e Informática  
Pontifícia Universidade de Minas Gerais (PUC Minas)  
Belo Horizonte – MG – Brasil

{gabrielfgalindo@gmail.com<sup>1</sup>, jgpgmgin31@gmail.com<sup>2</sup>}@sga.pucminas.br

{pedro.caixeta@sga.pucminas.br<sup>3</sup>}@sga.pucminas.br

**Resumo.** *O objetivo do projeto é desenvolver um software que gerencia um local que é alugado para a realização de cerimônias e outros tipos de eventos na forma de uma agenda virtual. Por meio desse aplicativo, o administrador será capaz de gerenciar o local, junto com as suas reservas e os seus custos.*

## 1. Introdução

A engenharia de *software* desempenha um papel fundamental na modernização e otimização de processos em diversos setores da sociedade. No contexto específico do Recanto Guerreiro, um local frequentemente alugado para eventos e cerimônias, a demanda crescente por seus serviços destacou a necessidade de uma gestão mais eficiente e simplificada. Atualmente, a administração depende de uma agenda em papel, cujas limitações incluem falta de controle, susceptibilidade à perda de informações e dificuldade de acesso.

Diante dessa realidade, o projeto proposto busca desenvolver um *software* que ofereça uma solução mais eficaz e conveniente do que a tradicional agenda em papel. Através dessa ferramenta digital, os usuários serão capazes de realizar todo o processo administrativo de forma mais organizada e formalizada. O aplicativo será especialmente projetado para dispositivos móveis, visando atender às necessidades do usuário a qualquer momento a partir do seu celular. Além disso, sua *interface* será intuitiva, levando em consideração a provável falta de familiaridade tecnológica de seus usuários.

As justificativas para o desenvolvimento deste trabalho são fundamentadas na importância de fornecer soluções tecnológicas acessíveis para todos os membros da sociedade, independentemente de sua idade ou conhecimento técnico. A implementação desta aplicação não apenas resolverá os desafios enfrentados pelo Recanto Guerreiro, mas também poderá ser adaptada e reaproveitada no futuro para ajudar outras pessoas que enfrentam problemas semelhantes, demonstrando assim o potencial transformador da engenharia de *software* na vida cotidiana.

## 2. Referencial Teórico

Aplicações como o *Sympla* e o *booking.com* foram selecionados como base para o desenvolvimento do projeto, devido ao seu *design* simples e intuitivo, além de também terem um foco no gerenciamento de reservas. Para adquirir mais conhecimento, ferramentas de pesquisa, como o *Google Acadêmico*, foram utilizadas para adquirir artigos de referência

para o desenvolvimento de uma *interface* simples e adequada para os usuários inexperientes.

Esses artigos focam nas dificuldades que usuários mais idosos possuem ao utilizarem *softwares*, fornecendo um *framework* para encapsular um bom *design* de *interface* para pessoas idosas[Zajicek 2004] e identificando deficiências comuns nas *interfaces* eletrônicas para usuários da mesma faixa etária[Wirtz et al. 2009]. Também foi pesquisado um artigo que investiga o uso de símbolos e palavras em *interfaces* e como elas podem ajudar pessoas mais velhas e pessoas com baixo conhecimento prévio na utilização de *softwares*[Reddy et al. 2020].

## **2.1. Extensão Universitária**

A Extensão Universitária visa ligar a universidade aos outros setores da sociedade por meio de atividades de prática extensionista com o objetivo de facilitar o compartilhamento do conhecimento produzido no ambiente acadêmico. A Extensão é fundamental para a experiência acadêmica, encorajando o protagonismo estudantil e incentivando a abordagem de desafios socioambientais importantes na atualidade[PUC-Minas 2023].

Em virtude da importância da Extensão Universitária, o projeto desenvolvido não apenas responde às necessidades operacionais da PUC, mas também visa contribuir para ao resto da sociedade e para a inclusão social de pessoas inexperientes e com falta de conhecimento tecnológico. O projeto visa reafirmar a importância da Extensão no ensino superior, por meio de sua inclusão no Proex, demonstrando sua capacidade de promover a construção de uma sociedade mais justa e inclusiva.

## **2.2. A Engenharia de Software**

Atualmente, a Engenharia de *Software* tem um papel vital no desenvolvimento, implementação e manutenção de sistemas essenciais para a sociedade moderna. Dada a dependência cada vez maior em sistemas de *software*, é essencial aplicar com precisão os princípios da Engenharia de *Software* e as práticas de desenvolvimento ágil para criar sistemas robustos e intuitivos.

Segundo um estudo pesquisado, micro e pequenas empresas de *software* enfrentam desafios significativos durante o processo de desenvolvimento[Fernandes et al. 2012]. Portanto, é fundamental adotar uma metodologia ágil, devido ao tamanho reduzido da equipe e à necessidade de flexibilidade para lidar com mudanças nos requisitos. Para tratar essas questões, dois outros artigos foram buscados por oferecerem modelos e arquiteturas de referência direcionados às pequenas empresas brasileiras, com o objetivo de facilitar o desenvolvimento de projetos[Colenci and Walimir 2011][Varoto 2002].

## **2.3. Trabalhos relacionados**

Uma revisão da literatura revelou trabalhos relevantes que abordam questões semelhantes de gestão de eventos e desenvolvimento de aplicativos para esse fim. Um desses exemplos é um artigo que aborda o desenvolvimento de um sistema *web* para a gestão de eventos de um cerimonial, apresentando em sua documentação a análise de requisitos, a metodologia utilizada e seus resultados, fornecendo *insights* valiosos para o nosso projeto[Silva 2020].

Outro exemplo, inclui uma pesquisa que examinou a usabilidade de aplicativos móveis para agendamento de eventos, o que pode fornecer orientação adicional para o desenvolvimento da *interface* do usuário no projeto[Souza et al. 2018]. Esses trabalhos destacam a importância de abordar os desafios específicos enfrentados por pequenas empresas no contexto do gerenciamento de eventos e fornecem um ponto de partida sólido para o trabalho.

### 3. Metodologia

O desenvolvimento do projeto seguiu a metodologia ágil *Scrum*, que proporciona flexibilidade e adaptabilidade ao longo do processo de desenvolvimento de aplicações[Schwaber and Sutherland 2017]. A metodologia *Scrum* foi aplicada por meio da organização do trabalho em *sprints*, períodos de tempo fixos e curtos no qual o time trabalha para completar uma quantidade específica de trabalho, com a ajuda do painel de projetos do *GitHub* para a organização e acompanhamento das atividades, com a divisão de tarefas sendo realizada com base nas habilidades e preferências individuais de cada membro da equipe.

Para o desenvolvimento do *front-end*, foram selecionadas as tecnologias *web* HTML, CSS e JavaScript, enquanto o Node.js foi escolhido para o desenvolvimento do *back-end*, devido à sua alta performance e escalabilidade. O gerenciamento do banco de dados foi realizado utilizando o MySQL.

#### 3.1. Sprint 1

Na primeira *sprint*, o projeto foi iniciado com uma contextualização de seus objetivos, junto com o levantamento dos requisitos e a assinatura dos termos. Também foi realizada uma reunião entre os membros da equipe e o cliente, para adquirir uma visão mais profunda e precisa do projeto e seus objetivos.

Além disso, a documentação do *software* começou a ser desenvolvida, por meio do editor *LaTeX Overleaf*, junto com a apresentação do trabalho no *powerpoint*. O repositório *GitHub* também foi criado para armazenar todo o código e os artefatos do projeto e dar início ao planejamento da *sprint 2* por meio do *dashboard* do *GitHub*.

#### 3.2. Sprint 2

Continuando o desenvolvimento do projeto, na *sprint 2* os diagramas de casos de uso, o diagrama ER e os protótipos de tela do *software* foram realizados, assim dando uma imagem visual de como as funcionalidades do aplicativo devem ser implementadas. Correções na documentação e no diagrama ER foram realizadas posteriormente, tornando-as mais precisas e mais alinhadas com o que foi solicitado.

O *front-end* e o *back-end* do aplicativo tiveram o seu desenvolvimento iniciado, com a implementação dos seguintes requisitos: a gestão das reservas, a gestão dos clientes, o estabelecimento da conexão com o banco de dados local e o processo de verificação das datas disponíveis.

#### 3.3. Sprint 3

Na *sprint 3*, a documentação, o diagrama ER e os protótipos de tela foram atualizados para incluir os requisitos da gestão do local. O *back-end* e o *front-end* para a gestão do local

também foram implementadas, junto com correções na integração dos outros requisitos do sistema.

Enquanto isso, como nas *sprint* anteriores, o planejamento da próxima *sprint* e a reorganização do *GitHub* foram realizados para que o desenvolvimento do projeto possa continuar.

### 3.4. Sprint 4

Continuando o desenvolvimento do *front-end*, mudanças no logo e no *design* da aplicação foram realizadas para deixar a *interface* mais estética. Isso também inclui mudanças na formatação do texto e na paleta de cores utilizada pelo sistema.

Ajustes também foram realizados para garantir uma experiência de usuário otimizada em dispositivos móveis, independentemente do tamanho da tela. Agora, a *interface* se adapta automaticamente às dimensões do dispositivo, proporcionando uma visualização e usabilidade de melhor qualidade para os usuários.

A respeito dos testes da aplicação, foram realizados testes de integração entre as partes do *front-end*, *back-end* e banco de dados. Os testes de integração revelaram que todas as partes estão devidamente conectadas e se comunicam sem problemas. Além dos testes de integração, também foram realizados testes para verificar se cada botão executa o comando ao qual foi designado e se o sistema exibe ao usuário corretamente as informações salvas no banco de dados.

## 4. Testes

O processo de teste do *software* foi realizado por dois indivíduos por meio da ferramenta *Postman* e utilizando o *localhost*. Todas as funcionalidades do sistema foram testadas, incluindo o processo de *login*, *logout* e o *CRUD* de todos os objetos da aplicação.

### 4.1. Login e Logout

O botão da página de *login* funciona sem nenhuma complicação, sendo testada por meio de uma conta de usuário inserida previamente no sistema. Em conjunto, o botão de *logout* do sistema foi testado e validado, funcionando corretamente em todas as páginas em que ele está disponível.

### 4.2. Create

Depois do processo de *login* e *logout*, a operação *create* foi testada para verificar se o usuário é capaz de criar novas entidades na aplicação. Caso a operação seja realizada com sucesso, uma mensagem de confirmação é exibida na tela do usuário.



#### 4.4. Update

Além de criar entidades, o usuário também deve ser capaz de editá-las. Por causa disso, páginas de edição das entidades criadas foram adicionadas ao sistema e utilizadas para testar e validar o processo de *update* dos objetos.

#### 4.5. Delete

Por último, foi testado se o usuário é capaz de deletar entidades da aplicação, removendo-as também de suas respectivas listas no sistema. Como em todos às funcionalidades anteriores, a operação *delete* foi testada e validado com sucesso.

### 5. Resultados

**Link do vídeo:**

**Link do repositório:** <https://github.com/ICEI-PUC-Minas-PPLES-TI/plf-es-2024-1-ti4-0648100-recanto-do-guerreiro>

**Link da apresentação:**

### 6. Conclusões e trabalhos futuros

### 7. Sobre as referências

#### Referências

- Colenci, A. and Walmir, E. (2011). A referencial model for small companies of development software. *IEEE Latin America Transactions*, 9(1):823–829.
- Fernandes, D. B., Tait, T. F. C., and Bruzarosco, D. C. (2012). Uma contribuição para o processo de desenvolvimento de software em micro e pequenas empresas desenvolvedoras de software. In *Anais do VIII Simpósio Brasileiro de Sistemas de Informação*, pages 145–156. SBC.
- PUC-Minas, E. (2023). Vídeo institucional 2023.
- Reddy, G. R., Blackler, A., Popovic, V., Thompson, M. H., and Mahar, D. (2020). The effects of redundancy in user-interface design on older users. *International Journal of Human-Computer Studies*, 137:102385.
- Schwaber, K. and Sutherland, J. (2017). Guia do scrum: Um guia definitivo para o scrum: As regras do jogo. *Available (in.)*, page 39.
- Silva, L. E. D. (2020). Desenvolvimento de um sistema web para a gestão de eventos de um cerimonial.
- Souza, R. P., Silva, T. R. B., Oliveira, J. M., and de Oliveira, G. A. A. (2018). Avaliação conjunta de facilidade de uso em aplicativos móveis para apoio à participação em eventos.
- Varoto, A. C. (2002). *Visões em arquitetura de software*. PhD thesis, Universidade de São Paulo.
- Wirtz, S., Jakobs, E.-M., and Ziefle, M. (2009). Age-specific usability issues of software interfaces. In *Proceedings of the IEA*, volume 17.
- Zajicek, M. (2004). Successful and available: interface design exemplars for older users. *Interacting with Computers*, 16(3):411–430.