

Desenvolvimento de um sistema auxiliar de gestão de cerimonial de casamentos

Hermes Henrique Caretta Guilherme
Graduado em Ciências da Computação – Uni-FACEF
hermescaretta@gmail.com

Fausto Gonçalves Cintra
Docente do Uni-FACEF
faustocintra@facef.br

Resumo

Planejar um casamento é uma tarefa complexa e desafiadora, levando muitos casais a buscarem a assistência de cerimonialistas para garantir que tudo transcorra conforme o planejado. Considerando que os profissionais nesse ramo precisam lidar com múltiplos aspectos da organização da cerimônia, surgiu a ideia de criar uma plataforma que centralizasse diversas ferramentas, com o propósito de facilitar o planejamento e a execução do evento. Para concretizar este projeto, procedeu-se a um levantamento de requisitos em colaboração com uma cerimonialista experiente. Posteriormente, foram elaboradas a documentação e a prototipação, e um Produto Mínimo Viável foi desenvolvido. Por fim, apresentou-se o produto resultante a uma profissional da área, para que ela destacasse o que funcionou bem no sistema e sugerisse eventuais melhorias para o futuro.

Palavras-chave: *Back-end. Cerimonial de casamentos. Front-end. Gestão de eventos. Organização de eventos.*

Abstract

Planning a wedding is a complex and challenging task, often leading many couples to seek the assistance of wedding planners to ensure that everything proceeds as planned. Considering that professionals in this field need to manage multiple aspects of ceremony organization, the idea of creating a platform that centralizes various tools to facilitate event planning and execution emerged. To bring this project to fruition, a requirements gathering process was carried out in collaboration with an experienced wedding planner. Subsequently, documentation and prototyping were developed, and a Minimum Viable Product (MVP) was created. Finally, the resulting product was presented to a professional in the field for her to highlight what worked well in the system and suggest potential improvements for the future.

Keywords: *Back-end. Event management. Event planning. Front-end. Wedding ceremonial.*

1 Introdução

Organizar um casamento é um desafio emocionante. Porém, de acordo com Finnel (2018, p. 26), "junto com a alegria de casar e ter um casamento, vêm o

inevitável estresse de planejar tudo e atender às suas próprias expectativas". Trata-se de uma ocasião crucial na vida de um casal, por isso é fundamental que tudo esteja perfeito. Para que tudo ocorra conforme o esperado, é comum que os casais contratem cerimonialistas, profissionais que auxiliarão em todo o planejamento do casamento.

Embora cada cerimonialista possua um método de trabalho distinto e possa oferecer diversos pacotes de serviços, conduziu-se uma entrevista com uma profissional da área, na qual foram obtidas informações relacionadas aos processos de trabalho e às ferramentas mais utilizadas no serviço.

Após a coleta desses dados, foi possível identificar quais funcionalidades necessitavam ser implementadas na plataforma e quais requisitos deveriam ser atendidos, sendo eles: a inclusão da possibilidade de administrar os convidados, fornecedores, lista de padrinhos, roteiros de cerimônia e festa e uma lista com as tarefas a serem feitas. Com a elaboração do esboço do projeto, foi estabelecido um *backlog* de atividades que deveriam ser cumpridas, incluindo prazos para execução. Esses itens abrangiam desde a documentação, a prototipação das telas até o desenvolvimento da plataforma e a apresentação do *software* à cerimonialista que contribuiu para o levantamento de requisitos.

2 Desafios na organização de um casamento

É importante entendermos o porquê de haver esse crescimento na demanda de cerimonialistas no planejamento de casamentos. Existem múltiplos motivos desse questionamento, mas vamos começar pelo motivo de não ter uma cerimonialista em um casamento. Um dos mais marcantes é o custo; no entanto, sendo retirada essa despesa, ganha-se no lugar a frustração de ter que pesquisar e ir atrás de todos os fornecedores, sem ter a garantia de que aquele trabalho é o melhor para o orçamento que se tem disponível.

Isso posto, contratar um profissional na área de cerimonial é quase indispensável no planejamento de um casamento. Muitas vezes, a profissão é desvalorizada pelas pessoas que acreditam ser um trabalho fácil de ser feito – uma percepção que, felizmente, vem diminuindo. Não se pode construir uma casa sem contratar um pedreiro; só ele sabe o material necessário, o que vai funcionar e o que não vai dar certo naquele terreno e como fazer para que aquela construção não caia, pois ele tem prática e se profissionalizou naquilo. O mesmo vale para o cerimonial. Como toda profissão, o cerimonialista preparou-se aquilo e vai montar o casamento dos sonhos de acordo com as possibilidades dos noivos, contratando os melhores e mais confiáveis fornecedores que caibam no seu orçamento dado.

Existem muitos aspectos a serem considerados na organização de um casamento, desde a escolha do local até a lapela dos padrinhos. É comum que alguns detalhes não sejam do conhecimento dos noivos antes que estes tenham contrato com um profissional de cerimonial. Assim, conforme Huang *et al.* (2017, p. 157), “sob a assistência de pessoal profissional, os casais podem concluir um casamento com facilidade e sucesso”.

Problemas recorrentes que são encontrados durante o planejamento de um casamento geralmente incluem: lidar com o orçamento, organizar lista de convidados,

saber quais fornecedores contratar, escolher o local e a data, coordenar os detalhes da decoração e gerenciar o tempo. Tudo isso pode ser delegado a um cerimonialista.

3 Proposta de solução

Considerando os problemas previamente mencionados, o propósito subjacente ao desenvolvimento do sistema é facilitar a vida cotidiana das cerimonialistas durante o planejamento de casamentos, ao mesmo tempo que consolida todas as ferramentas e funcionalidades em uma única plataforma. Como destacado por Finnel (2018, p. 26), "Diversos perfis de cerimonialistas existem, oferecendo serviços de assessoria completa ou voltados exclusivamente para o dia do evento, e cada um deles proporciona serviços diferenciados".

Após a realização de uma entrevista com uma cerimonialista que acumula mais de oito anos de experiência no setor, foi possível esmiuçar o processo de trabalho da profissional e coletar informações relevantes para a elaboração do sistema. Como resultado, foi desenvolvido uma plataforma web que permite a criação de tarefas para a organização, com *cards* individuais para cada tarefa e a possibilidade de adicionar *checklists* dentro dos mesmos. Além disso, o sistema permite uma organização mais precisa da lista de convidados, visando um controle mais eficiente da quantidade de adultos e crianças presentes. Algumas outras funcionalidades abrangem a gestão completa das finanças e dos fornecedores, com a possibilidade de inserir detalhes sobre os serviços contratados, bem como informações sobre data, hora e local de entrega. Ademais, a plataforma possibilita a criação e edição dos roteiros tanto da cerimônia quanto da festa, além da administração dos padrinhos do casamento. Por fim, todos os noivos dos casamentos possuem cadastro na plataforma, podendo entrar e administrar seus respectivos casamentos.

4 Ferramentas similares

Existem, no mercado, algumas ferramentas que podem ser usadas para auxiliar na organização de um casamento, separados em vários tópicos que se complementam:

- *Sites* de casamento: são considerados uma forma prática e elegante de compartilhar informações importantes com os convidados, como a data e local da cerimônia, informações sobre a festa, fotos do casal e muito mais. Alguns sites também permitem enviar convites virtuais e receber confirmações de presença dos convidados. Existem diversos *sites* de casamento disponíveis, alguns gratuitos e outros pagos, porém os mais populares são o Zankyou, o Casar.com e o iCasei;
- Planilhas de orçamento: sendo um dos principais desafios na organização de um casamento, é importante ter uma planilha de orçamento que permita controlar todos os gastos relacionados, dos menores aos maiores. Essa planilha deve incluir informações como os fornecedores, valores, datas de pagamento e qualquer outra informação relevante. Existem diversas planilhas de orçamento disponíveis na internet, mas é possível criar uma de forma personalizada, utilizando ferramentas como o Excel ou o Google Sheets;
- Aplicativos organizadores: esses aplicativos permitem criar listas de tarefas, lembretes, agendas e outras informações importantes para o planejamento do casamento. Entre os mais populares estão o Trello, o Evernote e o Google

Keep. Cada um deles tem suas próprias características, mas todos oferecem recursos para ajudar a organizar as tarefas e informações relacionadas ao casamento;

- Lista de presentes *online*: é uma forma prática de receber presentes dos convidados sem preocupação com a troca de presentes indesejados. Essas listas geralmente incluem uma seleção de produtos de diversas lojas e os convidados podem escolher o que desejam presentear. Entre os sites que oferecem esse serviço estão o Casare, o Zanky e o próprio Magazine Luiza que possui uma funcionalidade de Lista de Casamento.

Pode-se notar que, para que o usuário possua a melhor experiência para obter auxílio durante os preparativos de um casamento, se faz necessário o uso de múltiplas ferramentas, diferentes entre si. Como mencionado anteriormente, para resolver tal problemática, a ferramenta desenvolvida visa centralizar todas as necessidades da cerimonialista em uma única plataforma.

5 Ferramentas de Desenvolvimento

Nessa seção serão comentadas as ferramentas e procedimentos utilizados durante o desenvolvimento do projeto.

5.1 Git e Github

Git é um projeto *open source* desenvolvido por Linus Torvald para auxiliar no versionamento de arquivos em repositórios locais e remotos. De acordo com Paulo Silveira, é “[...] uma forma fácil de gerenciar o código-fonte da aplicação, do sistema e do produto.”.

Já o Github é uma plataforma de integração com o Git que permite a hospedagem em nuvem dos repositórios criados localmente, sendo utilizado “[...] para gerenciar seu código e criar um ambiente de colaboração entre devs.” (Silveira, 2023, *online*).

5.2 Visual Studio Code

Amplamente utilizado por programadores, o Visual Studio Code é um “[...] editor de código aberto desenvolvido pela Microsoft” (HANASHIRO, 2021, *online*), sendo utilizado para auxiliar na escrita, criação e manutenção de código.

5.3 Javascript

Javascript é uma linguagem de programação de alto nível utilizada para implementar itens complexos à *web*, podendo mostrar conteúdos que se atualizam em um certo intervalo de tempo, manuseando ações realizadas pelos usuários, renderizando itens 3D e entre outros. Criado por Brendan Eich em setembro de 1995 e mantido até hoje com novas atualizações pela Associação Européia dos Fabricantes de Computadores (da sigla em inglês ECMA).

5.4 NodeJS e ReactJS

Criado em 2009 com a finalidade de ser uma alternativa para rodar códigos *Javascript* em modo *server side*, o Node.js é um *framework* “que, na prática, se reflete

na possibilidade de criar aplicações *standalone* (autossuficientes) em uma máquina servidora, sem a necessidade do navegador.” (Bessa, 2023, *online*). É amplamente utilizado no mercado nos dias atuais devido à sua natureza de *open source* e grande comunidade de usuários que contribuem com o desenvolvimento e manutenção da tecnologia.

Desenvolvido pelo time de tecnologia do Meta (antigo Facebook), o ReactJS é uma biblioteca baseada em Javascript que permite a criação de aplicações *web SPA* (*Single Page Applications*), na qual todo o fluxo de execução do *site* é realizado em apenas uma sessão, sem a necessidade de recarregar as páginas. É uma das bibliotecas mais utilizadas no mercado por conta do seu alto poder de customização e criação de componentes, o que “[...] torna o React uma tecnologia muito flexível para a solução de problemas e para a construção de interfaces reutilizáveis, uma vez que cada um destes componentes pode ser manipulado de maneira distinta” (Roveda, 2023, *online*).

5.4 MySQL

O MySQL é um sistema de gerenciamento de banco de dados de código aberto que utiliza a linguagem SQL (*Structured Query Language* – Linguagem de Consulta Estruturada) “[...] que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados” (Pisa, 2012, *online*). Foi desenvolvido e publicado em 1995, pela empresa sueca MySQL AB, posteriormente adquirida pela Sun Microsystems, que, por seu turno, tornou-se parte da Oracle Corporation em outubro de 2010.

6 Documentação

Antes de dar início ao desenvolvimento do sistema, foi realizada a criação de alguns documentos a fim de estruturar quais seriam os requisitos e funcionalidades da plataforma. Esses documentos incluem a elicitação de requisitos, diagrama BPMN (*Business Process Model and Notation*) e o modelo relacional para a estrutura do banco de dados.

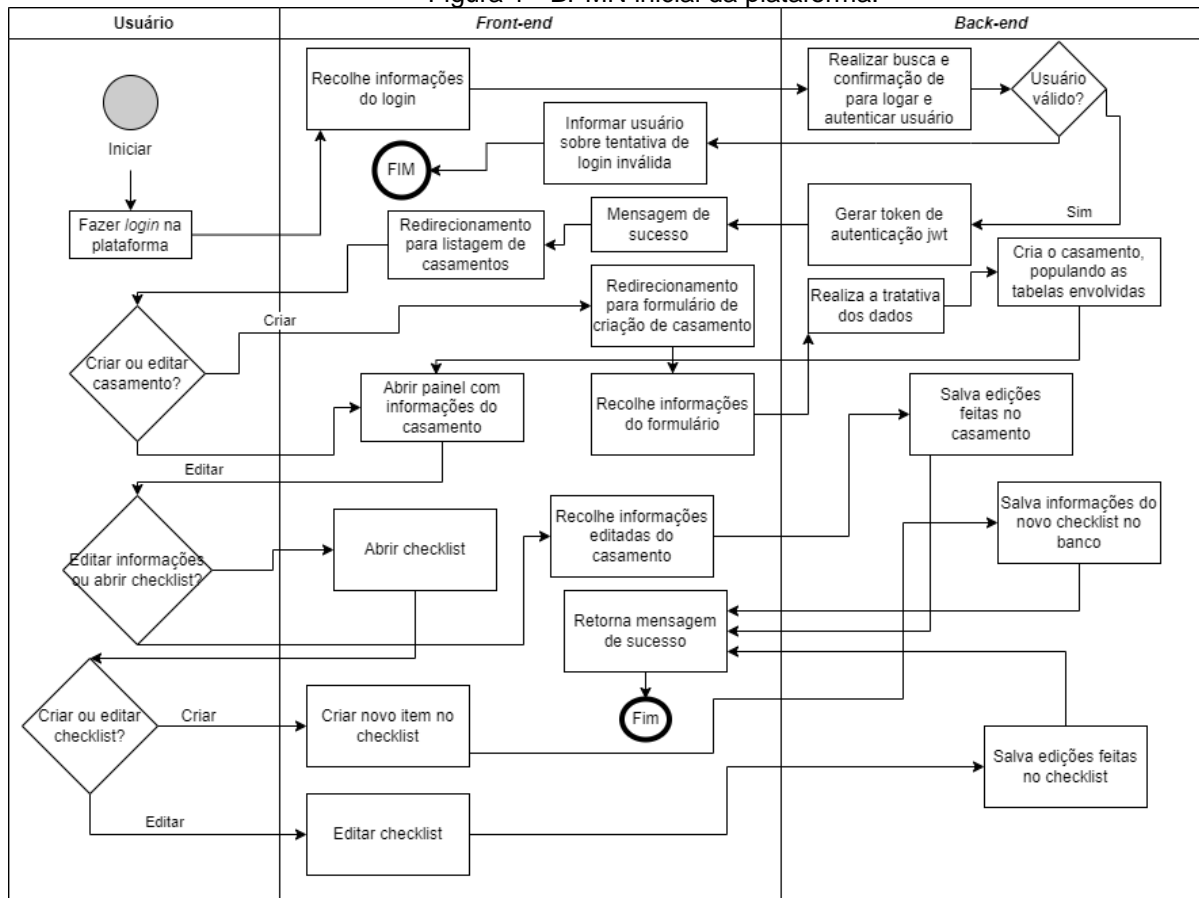
6.1 Elicitação de Requisitos

Foi realizada uma entrevista técnica com a uma cerimonialista que atua na área há mais de 8 anos, com o intuito de obter maior conhecimento acerca dos processos e ferramentas utilizadas no dia-a-dia. Com o conhecimento obtido, foi possível realizar um levantamento de funcionalidades-chaves que a plataforma deve ter, como o controle da lista de convidados com informações relevantes.

6.2 BPMN

O BPMN (em tradução livre, Notação de Modelagem de Processos de Negócios) foi desenvolvido com o intuito de auxiliar a visualização do fluxo que o usuário final terá ao utilizar o sistema. Com base na elicitação de requisitos foi realizada a modelagem do processo, contando desde o *login* do usuário até o fim da criação / edição de cada módulo da plataforma (Figura 1).

Figura 1 - BPMN inicial da plataforma.



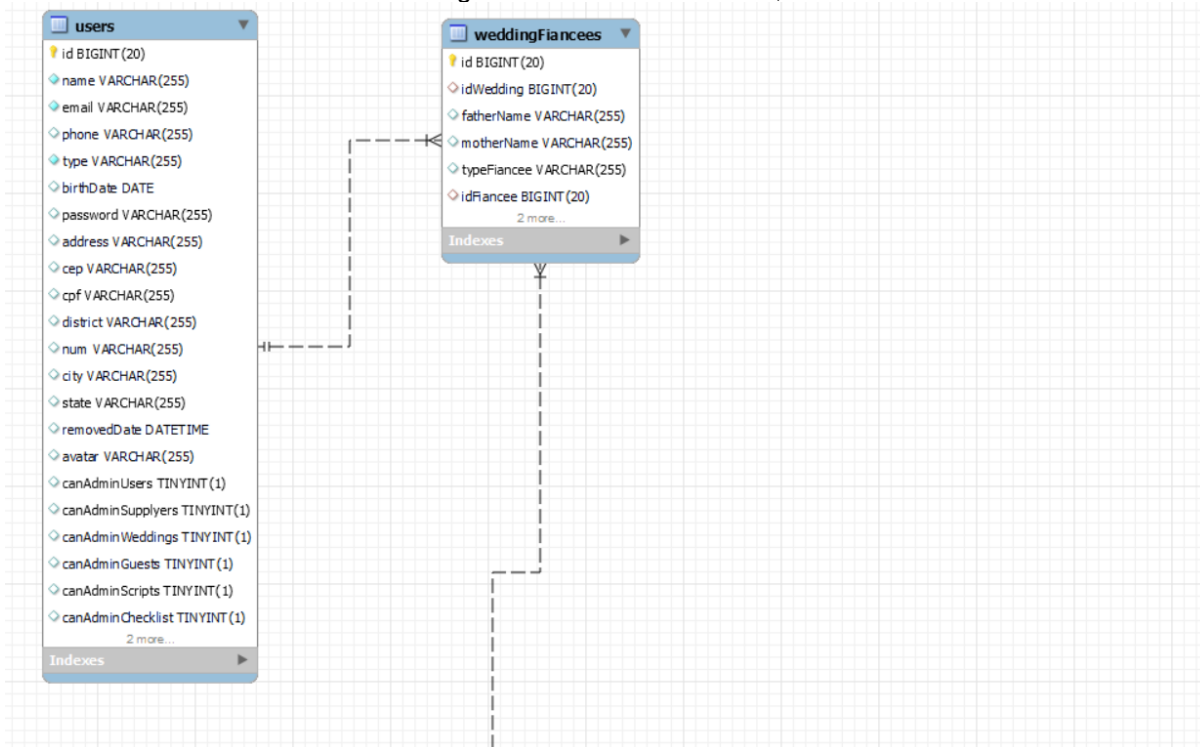
Fonte: elaborado pelos autores.

6.3 Modelo relacional do banco de dados

Após definido o fluxo de telas do sistema, foi feita a modelagem do banco de dados com base na elicitação de requisitos e no BPMN. O banco tem como principais tabelas a *users*, que é responsável pelo armazenamento e controle dos usuários e a tabela *weddings*, que é responsável pela administração do casamento e serve também como chave estrangeira para as demais tabelas referentes ao casamento.

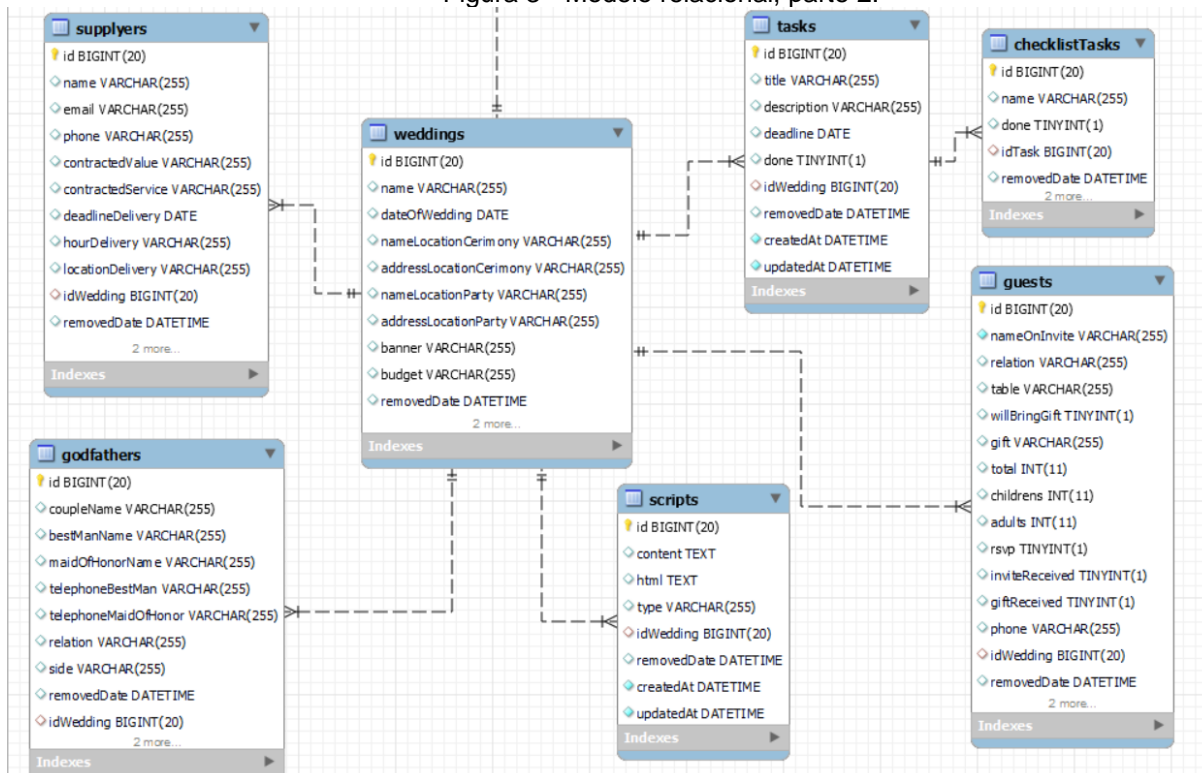
Nas Figuras 2 e 3, é possível visualizar a tabela *users* e suas colunas e a tabela *weddingFiancees* e suas colunas, essa, responsável por fazer a criação dos noivos, utilizando o *id* do usuário e o *id* casamento como chaves estrangeiras.

Figura 2 - Modelo Relacional, Parte 1.



Fonte: Os autores

Figura 3 - Modelo relacional, parte 2.



Fonte: elaborado pelos autores.

Na Figura 2, é possível visualizar a tabela *weddings* e as demais tabelas que compõem a estrutura do banco. Cada uma delas tem uma finalidade específica:

- Tabela *suppliers*: tem como finalidade armazenar e manipular as informações referentes aos fornecedores do casamento, guardando informações sobre o serviço contratado, data e local de entrega, tal qual fornecer dados para o *dashboard* financeiro;
- Tabela *godfathers*: tem como objetivo administrar as informações referentes a cada casal de padrinhos/madrinhas que farão parte do casamento;
- Tabela *scripts*: é responsável por controlar as informações dos roteiros da cerimônia e da festa;
- Tabela *guests*: tem como finalidade administrar as informações dos convidados do casamento, salvando informações como quantidade de convidados, presença confirmada e também para fornecer dados para o *dashboard*;
- Tabela *tasks*: é responsável por gerenciar as informações das tarefas criadas pela cerimonialista, guardando informações como a descrição e *deadline*;
- Tabela *checklistTasks*: é uma tabela complementar a tabela *tasks*, guardando informações referentes ao *checklist* que a/o cerimonialista pode adicionar a cada tarefa criada.

7 Prototipação

Após o levantamento dos requisitos, a criação do BPMN e a estruturação do banco de dados, foi possível realizar o desenho das telas através dos processos de rabiscoframe, *wireframe* e a prototipação de alta fidelidade, essa última feita pelo Figma.

7.1 Rabiscoframe

O rabiscoframe é um processo muito utilizado durante a etapa de prototipação das telas do projeto, pois é onde surgem as primeiras ideias de como o visual do sistema será e serve como base para a realização do *wireframe* e da prototipação de alta fidelidade.

Os desenhos foram feitos por meio de uma ferramenta *online*, semelhante ao *Paint*, na qual foi imaginada como seria o processo de *login* para o usuário, como seria a separação entre a listagem de casamentos e o acesso para administrar o casamento desejado e como estaria distribuído as informações no *dashboard* de dados gerais do casamento.

7.2 Wireframe

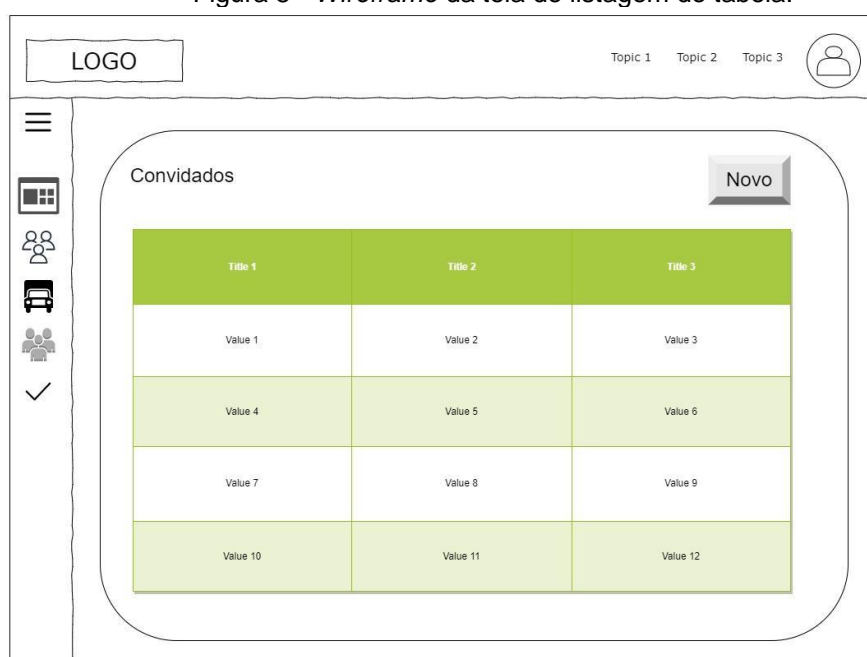
O *wireframe* (Figuras 4 e 5) é um protótipo que auxilia na diagramação das páginas, adicionando formas que representam os conteúdos que serão inseridos em cada seção da tela e como cada elemento se comporta. Foi utilizada a plataforma *draw.io* para montar os desenhos, utilizando componentes que facilitam a visualização da ideia de como a tela deve ser montada, como ícones e tabelas.

Figura 4 - Wireframe da tela de dashboard.



Fonte: elaborado pelos autores.

Figura 5 - Wireframe da tela de listagem de tabela.



Fonte: elaborado pelos autores.

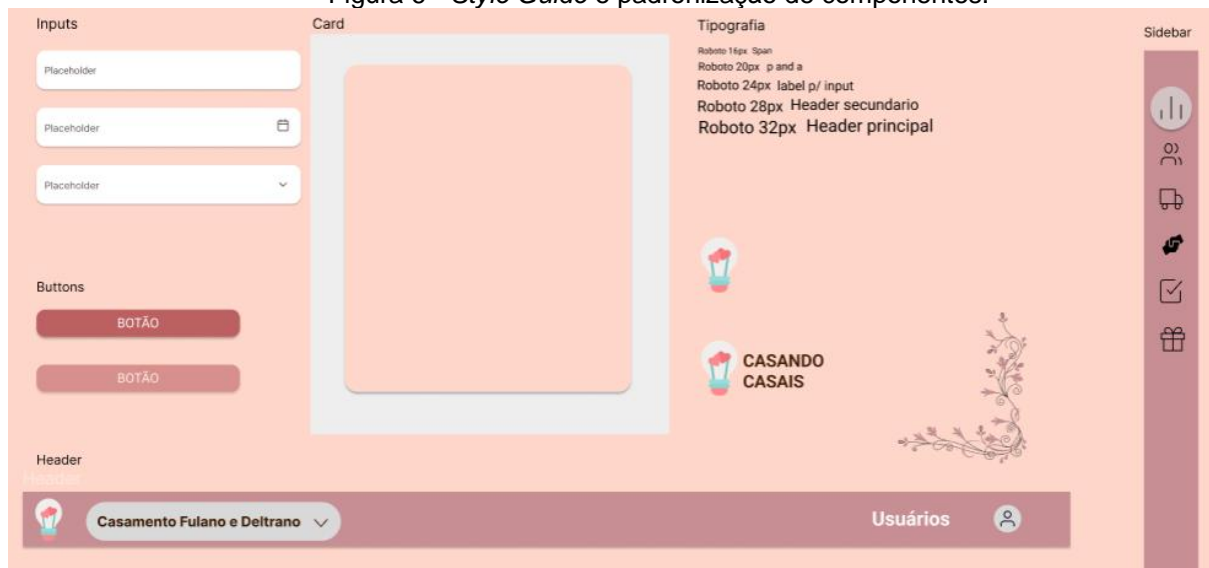
7.3 Style Guide

Uma parte vital na criação do visual do *site* é ter um guia de estilos para cada elemento (Figura 6), com padronização de botões, *inputs*, menus, etc. Para compor a

estilização dos componentes, foram utilizadas cores ligadas à tons rosas escuros, com uma cor de fundo branca como padrão e rosa mais claro para o os *cards*.

Nessa etapa também foram criados o nome da plataforma, Casando Casais e o logo da plataforma. Ambos foram pensados de uma maneira que harmonizasse com o estilo pensado para a plataforma e que representasse a missão do sistema, que é auxiliar cerimonialistas a, justamente, casar casais.

Figura 6 - *Style Guide* e padronização de componentes.

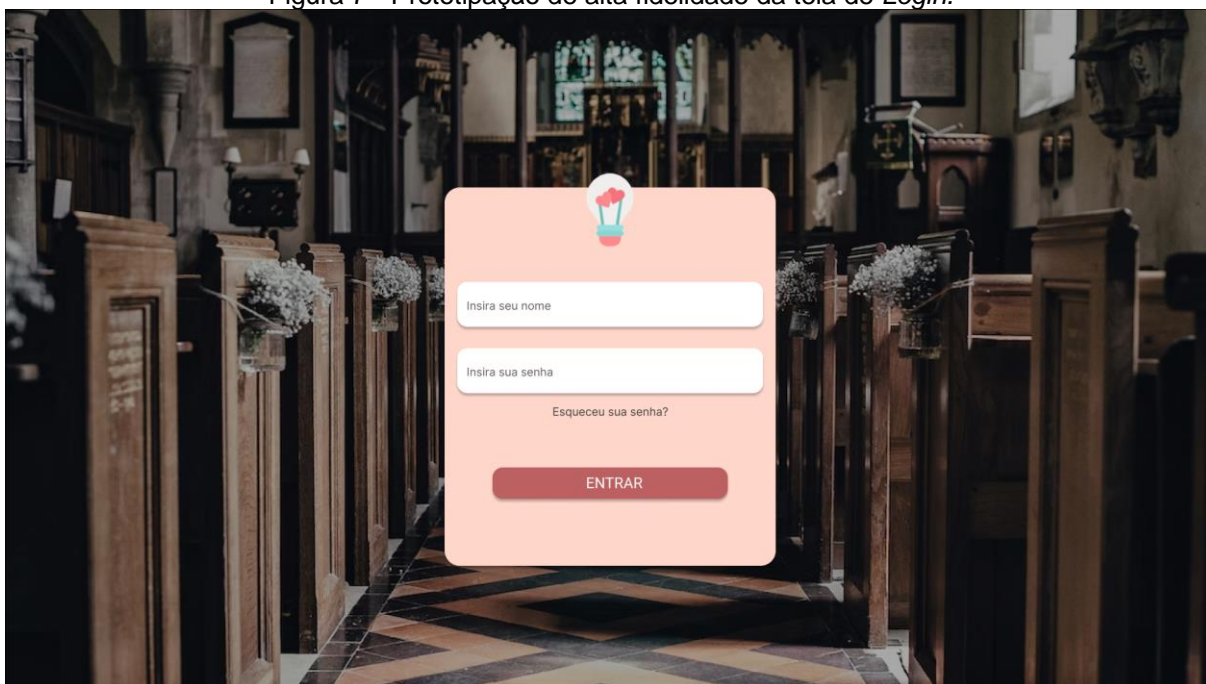


Fonte: elaborado pelos autores.

7.3 Prototipação de alta fidelidade

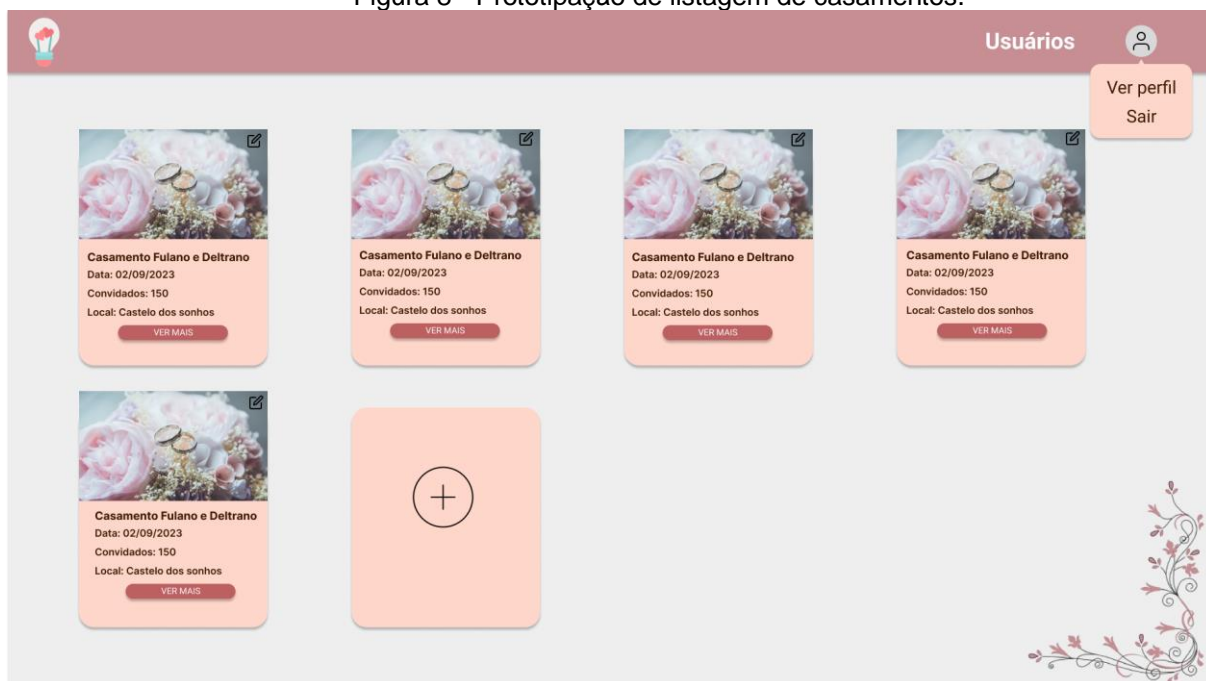
A última etapa de prototipação foi a de alta fidelidade (Figuras 7 a 13), para a qual foi utilizada a base dos *wireframes* e o *style guide* na concepção final das telas da plataforma, que posteriormente foram desenvolvidas no *front-end*.

Figura 7 - Prototipação de alta fidelidade da tela de *Login*.



Fonte: elaborado pelos autores.

Figura 8 - Prototipação de listagem de casamentos.



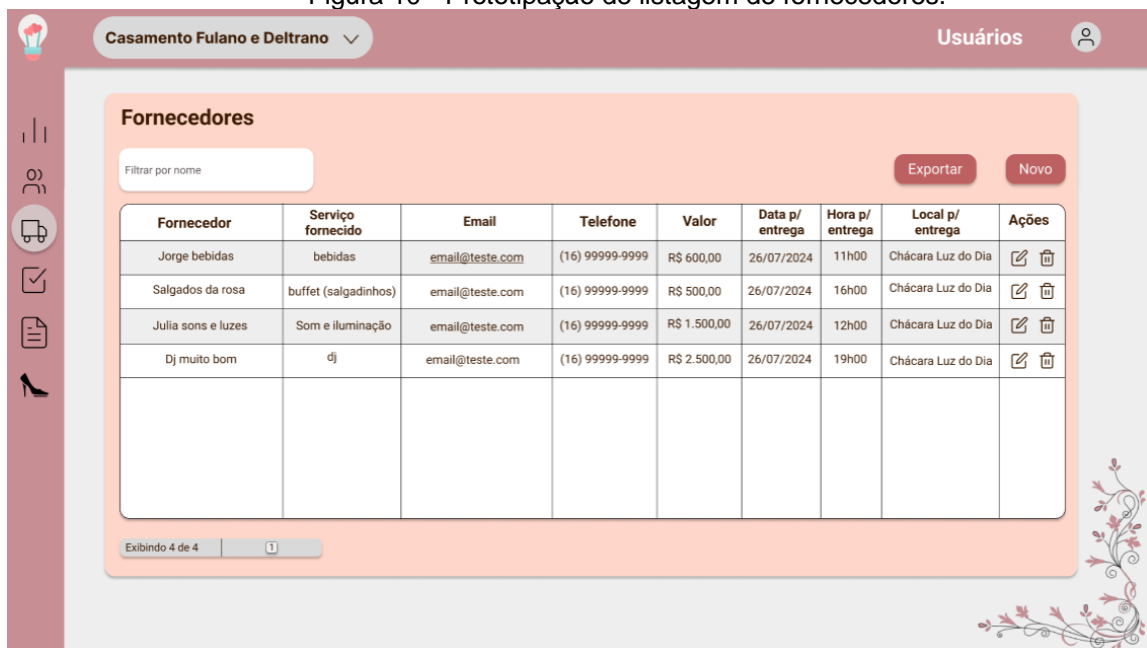
Fonte: elaborado pelos autores.

Figura 9 - Prototipação da tela de *dashboard*.



Fonte: elaborado pelos autores.

Figura 10 - Prototipação de listagem de fornecedores.



Fonte: elaborado pelos autores.

Figura 11 - Prototipação de listagem de fornecedores.

Casamento Fulano e Deltrano

Usuários

← Cadastrar fornecedor

Insira o nome do fornecedor:
Ex: João do Balão

Insira o serviço contratado:
Ex: Balões

Insira o email do fornecedor:
teste@teste.com

Insira o telefone do fornecedor:
Ex: (XX) X XXXX-XXXX

Insira o valor do serviço contratado:
R\$ X.XXX,XX

Insira a data para entrega do serviço contratado:
DD/MM/AAAA

Insira a hora para entrega do serviço contratado:
HH:mm

Insira o local para entrega do serviço contratado:
Ex: Local da festa - Chácara Bom Dia

Cancelar Salvar

Fonte: elaborado pelos autores.

Figura 12 - Prototipação de criação de roteiros.

Casamento Rachel e Finn

Usuários

Roteiro Cerimônia Festa

SALVAR

Local: Igreja Ohio - **Horário previsto:** 17h00 às 18h00

1º Entrada - NOIVA (Música: "Can't help falling in love, Elvis Presley")
1. Noiva Rachel com a mãe Shelby

2º Entrada - ALIANÇAS (Música: PENDRIVE)
1. Dama Tina (sobrinha) e Brittany (sobrinha) com as alianças e balão gás hélio

3º VOTOS dos noivos

4º AGRADECIMENTO dos noivos para os convidados

5º ASSINATURA LIVRO para todos os convidados (música: PENDRIVE)

Observações Gerais

- Acompanhamento de montagem as 9h00 até as 11h30 - Santana;
- Cartório às 10h30 - Noivos;
- Não temos entrada dos padrinhos/ Não temos entrada do noivo Finn/ Não temos lapelas;
- Noiva/mãe e sogra no salão de beleza às 12h00 até às 15h00;
- Buquê com flores secas (já está com a noiva);
- Recolher todas as cadeiras da cerimônia e organizar no salão de festa enquanto acontece as assinaturas dos convidados no livro.

Fonte: elaborado pelos autores.

Figura 13 - Prototipação da tela *Landing Page*



Fonte: elaborado pelos autores.

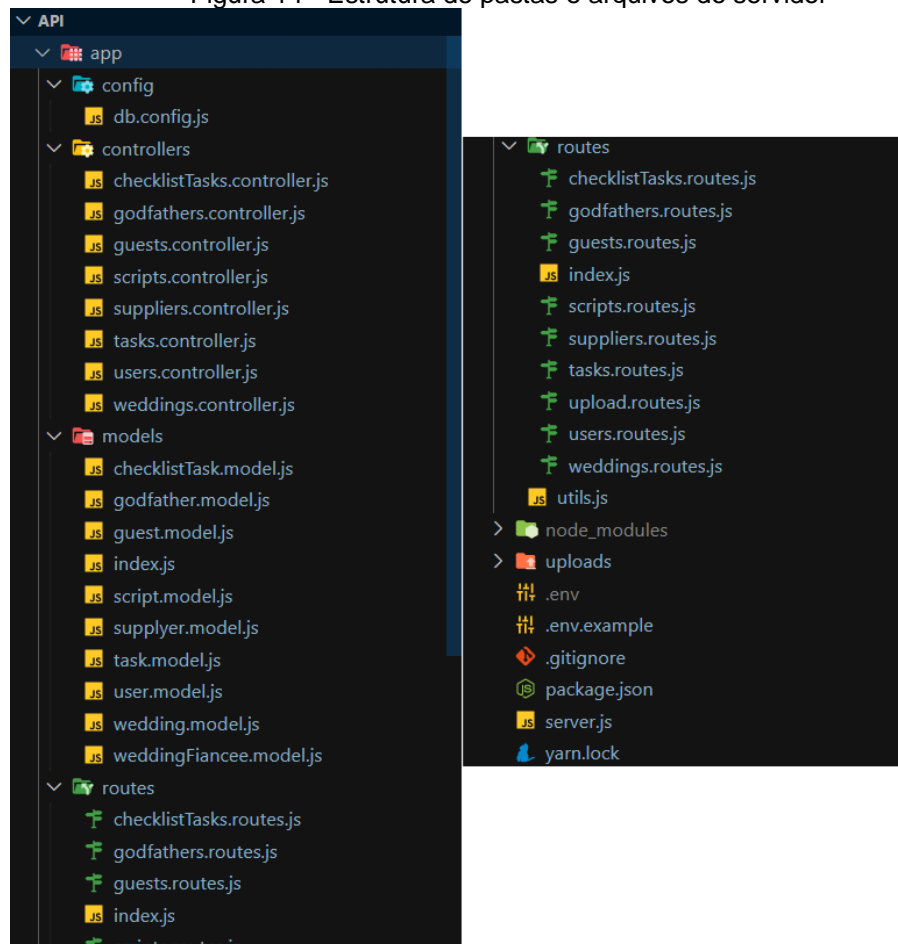
8 Desenvolvimento

O desenvolvimento da plataforma foi separado em duas frentes, o *back-end* (ou servidor, como será chamado), que é responsável por receber e enviar dados ao *front-end* e realizar a conexão com o banco de dados, e o *front-end* (ou cliente, como será referenciado) que gerencia toda a parte visual e de interação do usuário, tal qual receber e enviar as comunicações de dados do servidor.

8.1 Servidor

O servidor funciona como uma *API RESTful*¹ e foi desenvolvido utilizando Node.js, Express (um *framework* em *JavaScript* que possibilita a criação de rotas do padrão *HTTP Requests*) e Sequelize, um *type ORM*, que é “[...] uma biblioteca que ajuda a trabalhar com bancos de dados relacionais em seus projetos *Node.js*” (Kriger, 2023, *online*). A estrutura do projeto é ilustrada pela Figura 14.

Figura 14 - Estrutura de pastas e arquivos do servidor



Fonte: elaborado pelos autores.

A execução inicial se dá pelo arquivo *server.js* na qual é iniciada uma instância do *Express* e são feitas as primeiras configurações da API, sincronização do banco

¹ *API RESTful* é um método no qual dois sistemas utilizam para realizar a troca de informações de forma segura através de requisições *HTTP*.

de dados por meio do arquivo *index.js* disponível na pasta *models* e inicialização das rotas disponíveis para acesso, que vem do arquivo *index.js* da pasta *routes*.

As *models* servem para realizar conexão com o banco de dados e realizar a sincronização das tabelas, que são declaradas, cada uma, em seu respectivo arquivo que segue o padrão *tableName.models.js*, possuindo a declaração de cada coluna e suas relações (chaves estrangeiras).

Dentro da pasta *routes* existe um arquivo referente a cada tabela, seguindo o padrão de nomenclatura *tableName.routes.js*, com a finalidade de configurar as rotas que servirão como pontos de comunicação do cliente com o servidor mediante requisições HTTP utilizando os métodos GET (obter informações), POST (salvar informações), PUT (editar informações) e DELETE (no sistema não é realizado remoção total das linhas, mas sim adicionado uma data de remoção, para que seja possível reativar o dado a qualquer momento). Cada método dentro do arquivo de configuração de rota utiliza uma função, que está declarada dentro do arquivo *tableName.controller.js* na pasta *controllers*, função essa que irá manipular os dados de acordo com a sua finalidade, realizando buscas, inserções e atualizações.

Todas as rotas, com exceção do *login*, utilizam um *middleware* de autenticação que funciona com base em JWT (*JSON Web Token*). Trata-se de um *token* único gerado no momento que o usuário acessa a plataforma e é posteriormente utilizado para verificar se o usuário está autorizado a enviar requisições para as rotas protegidas do servidor.

8.2 Cliente

O cliente da plataforma foi desenvolvido com ReactJS e Styled Components, a qual é uma biblioteca que implementa *Css in JS*², possibilitando a criação de componentes que seguem um padrão de estilização e que podem ser reaproveitados em diversas páginas. A estrutura do projeto é demonstrada na Figura 15.

² *Css in JS* é uma solução que permite a criação de códigos de estilização CSS utilizando *Javascript*, oferecendo maior modularização e capacidade de reutilização de estilos no projeto.

Figura 15 - Estrutura de pastas e arquivos do projeto *Front-end*



Fonte: elaborado pelos autores.

A plataforma usa a biblioteca *react-router-dom* como base para montar a estrutura de rotas para cada página, configurada no arquivo *router.js* dentro da pasta *router*. Cada rota possui um parâmetro *element* que recebe uma instância do componente *ProtectedRoute*, responsável por verificar se o usuário está autenticado na plataforma, e um componente da página a ser configurada, conforme exemplo da Figura 16, na qual é passada a página *Weddings* que foi configurada no caminho *pages/weddings/index.js*.

Ainda na estrutura de pastas e arquivos do repositório, dentro da pasta *pages* foi realizada a separação entre quais páginas são gerais e quais são referentes apenas aos casamentos selecionados. Estas se encontram dentro da pasta *admin*.

Figura 16 - Exemplo de configuração de rotas

```
42 <Route  
43   path="/weddings"  
44   exact  
45   element={  
46     <ProtectedRoute isLogged={isLogged}>  
47       <Weddings />  
48     </ProtectedRoute>  
49   }  
50 />
```

Fonte: elaborado pelos autores.

9 Controle de qualidade com cerimonialista

Foi realizada uma reunião, no dia 20/08/2023, com a cerimonialista-cliente, durante a qual foi apresentado um Produto Mínimo Viável (*MVP*, da sigla em inglês) com o funcionamento da plataforma envolvendo todo processo de criação, listagem, edição e remoção de cada um dos módulos.

Durante a reunião, foram levantados com a profissional da área alguns pontos positivos e outros negativos sobre o produto demonstrado, sendo eles:

- Pontos negativos: o campo de inserção de datas em formulários poderia ter um funcionamento mais intuitivo; alguns campos em certos cadastros poderiam ser opcionais (como, por exemplo, informações de contato com os padrinhos e data de nascimento dos noivos); todo o módulo de tarefas/*checklist* poderia ser montado sobre de uma lista pré definida, com a possibilidade de ser alterada de acordo as necessidades do casamento em questão; a possibilidade de automação do preenchimento da tabela de convidados por meio de uma planilha existente e algumas outras informações complementares em alguns formulários de cadastro.
- Pontos positivos: a plataforma apresenta uma boa identidade visual, contando com cores harmoniosas e elementos bem distribuídos e uma boa usabilidade do sistema, sendo de fácil compreensão os fluxos de execução de cada tela. Também foi constatado que os formulários de convidados, fornecedores, roteiros e o próprio casamento possuem campos considerados de extrema importância e que de fato facilitam na compreensão do contexto de cada tabela a ser preenchida.

10 Considerações finais

O projeto teve origem na ideia de centralizar as ferramentas utilizadas pelas(os) cerimonialistas no seu dia a dia, com o objetivo de aumentar a produtividade e a organização durante o planejamento de casamentos. Mediante a realização de um levantamento de requisitos em colaboração com uma profissional da área, foi possível planejar a execução do projeto. Isso incluiu a elaboração do *backlog* de atividades, a criação de documentações, como o *BPMN* e o modelo relacional do banco de dados, e a prototipagem das telas, utilizando métodos como *rabiscoframe*, *wireframe* e prototipação de alta fidelidade. Além disso, foi feito o desenvolvimento do código fonte

do servidor e do cliente, bem como o controle de qualidade da plataforma, em parceria com uma cerimonialista.

Após a reunião em que a plataforma foi apresentada à profissional da área, ficou evidente que, embora tenham sido feitos progressos em algumas áreas, como a interface e a usabilidade, outras necessitam de revisão, especialmente as funcionalidades relacionadas às tarefas e ao *checklist*. Portanto, serão necessários ajustes mínimos na plataforma.

Para o futuro do projeto, está prevista a implementação de algumas funcionalidades adicionais. Isso inclui a capacidade de importar uma planilha de convidados para o sistema, a possibilidade de exportar um relatório com os dados do *dashboard* e o *download* dos roteiros de cerimônia e festa. Além disso, há planos para o desenvolvimento de um aplicativo móvel que servirá como um facilitador para o grande dia. Com esse aplicativo, será possível controlar a presença dos convidados, identificar suas mesas e gerenciar a lista de tarefas para o dia do evento.

Referências

- BESSA, André. **Node.JS**: o que é, como funciona esse ambiente de execução JavaScript e um Guia para iniciar. [S. l.], 2023. Disponível em: https://www.alura.com.br/artigos/node-js?gclid=Cj0KCQjw7PCjBhDwARIsANo7CgnR_P85MfJjkKBRUFS6Boi0neOgjSIF6Gnp8C8k6N9tWq6RlnEenT8aAs31EALw_wcB. Acesso em: 4 jun. 2023.
- BESSA, André. **TYPEORM O QUE É: CONCEITO, PARA QUE SERVE E COMO INSTALAR?**, de André Bessa. [S. l.], 14 mar. 2023. Disponível em: <https://kenzie.com.br/blog/typeorm/#:~:text=O%20TypeORM%20é%20uma%20biblioteca,abstrata%20e%20orientada%20a%20objetos>. Acesso em: 21 ago. 2023.
- FINNEL, Clare. **A History and Analysis of Weddings and Wedding Planning**. EventPRO, [S. l.], p. 1-63, 18 dez. 2018. Disponível em: https://scholarsarchive.jwu.edu/cgi/viewcontent.cgi?article=1037&context=student_scholarship. Acesso em: 26 mar. 2023.
- HANASHIRO, Akira. **VS Code - O que é e por que você deve usar?**. [S. l.], maio/2021. Disponível em: <https://www.treinaweb.com.br/blog/vs-code-o-que-e-e-por-que-voce-deve-usar>. Acesso em: 6 out. 2023.
- HUANG, Han-Chen et al. ANALYSIS OF IMPORTANCE OF THE PROFESSIONAL ABILITIES REQUIRED BY PERSONNEL IN WEDDING PLANNER SERVICES. **The International Journal of Organizational Innovation**, [S. l.], p. 49, 4 abr. 2017. Disponível em: <https://www.ijoi-online.org/attachments/article/52/FINAL%20ISSUE%20VOL%209%20NUM%204%20APRIL%202017%20SECTION%20B.PDF#page=49>. Acesso em: 26 mar. 2023.
- KRIGER, Brunno. **TYPEORM O QUE É: CONCEITO, PARA QUE SERVE E COMO INSTALAR?**. [S. l.], 14 mar. 2023. Disponível em: <https://kenzie.com.br/blog/typeorm/>. Acesso em: 2 out. 2023.

PISA, Pedro. **O que é e como usar o MySQL?**. [S. l.], 17 abr. 2012. Disponível em: <https://www.techtudo.com.br/noticias/2012/04/o-que-e-e-como-usar-o-mysql.ghtml>. Acesso em: 4 jun. 2023.

ROVEDA, Ugo. **O QUE É REACT: PARA QUE SERVE, COMO FUNCIONA E CARACTERÍSTICAS**. [S. l.], 2023. Disponível em: <https://kenzie.com.br/blog/react/>. Acesso em: 4 jun. 2023.

SILVEIRA, Paulo. **O QUE É GIT E GITHUB: COMO CONFIGURAR E PRIMEIROS PASSOS**. [S. l.], 7 jun. 2023. Disponível em: <https://www.alura.com.br/artigos/o-que-e-git-github>. Acesso em: 2 out. 2023.