

CENTRO REGIONAL UNIVERSITÁRIO DE ESP. SANTO DO PINHAL

CURSO DE ENGENHARIA DA COMPUTAÇÃO

ANGELO GABRIEL HENRIQUE DA SILVA

CONSTRUÇÃO DE UM WEBAPP AGENDA DE CONTATOS

**ESPÍRITO SANTO DO PINHAL
2023**

CENTRO REGIONAL UNIVERSITÁRIO DE ESP. SANTO DO PINHAL

CURSO DE ENGENHARIA DE COMPUTAÇÃO

ANGELO GABRIEL HENRIQUE DA SILVA

CONSTRUÇÃO DE UM WEBAPP AGENDA DE CONTATOS

Trabalho de Conclusão de Curso apresentado como exigência parcial para obtenção do título de Bacharel em Engenharia de Computação do Centro Regional Universitário de Espírito Santo do Pinhal à Banca Examinadora sob orientação do Prof. Esp. Jasiel Pereira Pinto.

**ESPÍRITO SANTO DO PINHAL
2023**

FOLHA DE APROVAÇÃO

Autor: Angelo Gabriel Henrique da Silva

Título: Construção de um webapp agenda de contatos.

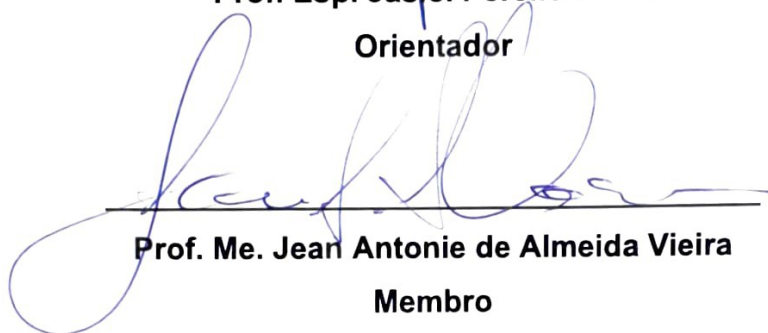
Avaliação: 8,62

Banca Examinadora



Prof. Esp. Jasiel Pereira Pinto

Orientador



Prof. Me. Jean Antonie de Almeida Vieira

Membro



Prof. Me. Gleidson Juliacci Patto

Membro

Espírito Santo do Pinhal, 21 de novembro de 2023

Construção de um webapp agenda de contatos

Da Silva, Angelo Gabriel Henrique (Unipinhal) dev.angelo.gabriel@gmail.com
Pereira Pinto, Jasiel (Unipinhal) prof.jasiel.pinto@unipinhal.edu.br

Resumo

Devido ao crescimento acelerado da área de tecnologia, estima-se que até 2024 será demandado cerca de 420 mil profissionais, fazendo com que muitas empresas busquem meios de aumentar a eficiência em sua produtividade. No Brasil, as ferramentas low-code e no-code são utilizadas principalmente em empresas de pequeno e médio porte (PME) devido à falta de mão de obra especializada em diversos setores e vem ganhando usabilidade, pois dá autonomia e liberdade a seus departamentos e profissionais. Possibilita que criem suas próprias soluções digitais, sendo uma boa opção para startups que podem chegar ao mercado mais rápido, comparado com as técnicas de desenvolvimento de software e codificação tradicional, onde o tempo de desenvolvimento pode demorar semanas, meses ou até anos, ao utilizar as ferramentas low-code e no-code possibilitam a realização em questão de horas ou dias, reduzindo significativamente os custos indiretos de desenvolvimento. Dentre as inúmeras ferramentas low-code e no-code como Appian, AppSheet, Bubble.io, Carrd, Monday.com, Mendix, UserGuiding, Webflow entre outras disponíveis; a Bubble.io foi escolhida pois vem se consolidando como uma das ferramentas no-code mais utilizadas no mercado, devido a constante atualização da plataforma nos últimos anos. O Webapp desenvolvido no projeto é uma agenda de contatos onde será cadastrado nome, telefones, e-mails, endereços; foi optado por ser desenvolvido em no-code por causa da facilitação que as ferramentas proporcionam ao criar as telas, seus designs e banco de dados, assim agilizando as etapas na manipulação dos dados, tendo como objetivo demonstrar as capacidades de desenvolvimento, que possa ser incorporado em outros projetos futuramente. O funcionamento em testes se mostrou com constância e estabilidade, porém trazendo maior dificuldade em definir as separações de usuários de forma segura, na criação e escolha dos designs, entre a organização de componentes para futuras manutenções e modificações.

Palavras-chave: low-code, no-code, desenvolvimento, software, ferramentas.

Abstract

Due to the accelerated growth in the technology area, it is estimated that around 420 thousand professionals will be required by 2024, causing many companies to look for ways to increase efficiency in their productivity. In Brazil, low-code and no-code tools are used mainly in small and medium-sized companies (SMEs) due to the lack of specialized labor in various sectors and have been gaining usability, as they give autonomy and freedom to their departments and professionals. It allows them to create their own digital solutions, being a good option for startups that can reach the market faster, compared to traditional software development and coding techniques, where development time can take weeks, months or even years, using low-code and no-code

tools makes it possible to complete this in a matter of hours or days, significantly reducing indirect development costs. Among the countless low-code and no-code tools such as Appian, AppSheet, Bubble.io, Carrd, Monday.com, Mendix, UserGuiding, Webflow among others available; Bubble.io was chosen because it has been consolidating itself as one of the most used no-code tools on the market, due to the constant updating of the platform in recent years. The Webapp developed in the project is a contact book where names, telephone numbers, emails, addresses will be registered; was chosen to be developed in no-code because of the facilitation that the tools provide when creating the screens, their designs and database, thus speeding up the steps in data manipulation, with the aim of demonstrating the development capabilities, which can be incorporated into other projects in the future. The operation in tests proved to be consistent and stable, but brought greater difficulty in defining user separations in a safe way, in creating and choosing designs, and organizing components for future maintenance and modifications.

Keywords: *low-code, no-code, development, software, tools.*

1. Introdução

Estimativas de crescimento da área da tecnologia é de crescimento acelerado, onde até 2024 serão abertas cerca de 420 mil vagas, causando uma escassez de profissionais na área (BRASCOM,2019), que criou a necessidade de criação de ferramentas que visam reduzir significativamente o tempo de desenvolvimento, assim criando inúmeras tendências entre as principais de 2023 segundo MadeinWeb (2023) está o desenvolvimento em Cloud, No-code e Low-code, pois deve-se enfatizar que enquanto as plataformas de No-code e Low-code, pode acelerar significativamente o processo do desenvolvimento, também podem limitar a customização e a flexibilidade do produto final.

Quando usadas junto aos métodos de desenvolvimento ágil, que é usado para que etapas sejam desenvolvidas em conjuntos como requisitos e implementação do sistema, assim os usos de ferramentas podem ser usados para gerar um esqueleto do programa a partir de um projeto (SOMMERVILLE, 2011).

2. Mercado Brasileiro

No Brasil essas ferramentas low-code e no-code estão sendo utilizadas principalmente em pequenas e médias empresas, devido à grande disputa ao contratar mão de obra especializada entre grandes empresas e multinacionais em diversos setores, também vem ganhando usabilidades dentro de grandes empresas

pois dá autonomia e liberdade a seus departamentos e profissionais possibilitando que eles criem suas próprias soluções digitais (NO CODE, 2022).

Essas abordagens têm um impacto significativo no mercado brasileiro, um dos seus principais benefícios é facilitar o rápido desenvolvimento de aplicações de software, isso tem um efeito significativo na capacidade das empresas de inovarem e se adaptarem rapidamente. Outras vantagens incluem a capacidade de utilizar métodos de low-code e no-code para treinar especialistas de TI em novas tecnologias e abordagens. Apesar de não eliminar completamente a necessidade de habilidades técnicas profundas, é um substituto que pode suprir parcialmente a falta de profissionais de TI no mercado brasileiro (LIMA, 2023).

3. Objetivo

O presente projeto visou a criação de um *webapp* com a função de ser uma agenda de contatos, para agilizar e atualizar a forma como as informações de contato pessoais e comerciais são gerenciadas. Com a ajuda deste projeto, os utilizadores poderão gerir de forma eficiente e intuitiva as suas redes de contatos, atualizar e encontrar informações importantes com mais facilidade. Além disso, demonstrar as facilidades que a tecnologia *no-code* proporciona através da simplificação no desenvolvimento e seu ganho de eficiência, e como afeta na criação do *webapp* e deixá-lo de forma organizada e simplificada para que possa ser usado em outros projetos no futuro.

4. Revisão Bibliográfica

4.1 Desenvolvimento tradicional

O desenvolvimento tradicional ao longo do tempo, tem como principais características suas metodologias. As metodologias tradicionais de desenvolvimento de software fornecem abordagens e diretrizes estruturadas para o desenvolvimento de software, resultando em melhor qualidade, eficiência e satisfação do cliente. Essas metodologias incluem o Modelo Cascata, Modelo V, RUP e Modelo Espiral, cada um com características específicas que se adequam a diferentes tipos de projetos. Ao adotar a abordagem apropriada, os profissionais de desenvolvimento de software

podem alcançar resultados eficazes e satisfazer as necessidades dos clientes. As metodologias tradicionais oferecem roteiros claros e estruturados, permitindo uma abordagem disciplinada que resulta em produtos de alta qualidade (SCHMIDT, 2023).

As metodologias tradicionais também têm como fortes características suas etapas bem definidas, um exemplo seria seguindo um modelo sequencial, onde a sequência na maioria dos projetos são: planejamento de software, atividades necessárias, definição de prazos e custos, se está dentro do orçamento, execução, e entrega, onde somente fará sentido quando estiver finalizado, assim sendo, contendo uma característica negativa que não contém muita flexibilidade para mudanças ou adaptações ao longo dos projetos, causando grande preocupação aos gerentes de projetos (GUEDES, 2019).

4.2 Ferramentas no-code e low-code

As ferramentas *no-code* e *low-code* estão revolucionando a indústria de desenvolvimento de software, com empresas como Webflow e Bubble.io. Essas plataformas permitem que não desenvolvedores criem sites e aplicativos poderosos sem contratar especialistas em software, reduzindo as despesas gerais de TI e acelerando o tempo de lançamento no mercado. No entanto, eles também têm limitações, incluindo falta de controle e recursos limitados de personalização. Embora nenhum desenvolvedor esteja atualmente em perigo, o futuro modelo de desenvolvimento de software provavelmente será híbrido, com soluções de suporte que permitirão uma implantação mais rápida e eficiente de aplicativos, juntamente com especialistas de software qualificados (KRAJEWSKI, 2021).

O desenvolvimento *low-code* permite personalizar o código caso seja necessário, ou seja, necessita do básico na linguagem da ferramenta escolhida, entre as ferramentas *low-code* podem ser citadas o Appian, AppSheet, Mendix, entre outras (CLARK, 2020).

Já o desenvolvimento *no-code* não permite personalização por códigos, em compensação não necessita de conhecimento básico na linguagem da ferramenta escolhida, atualmente existem diversas ferramentas *no-code* como o Bubble.io, Caard, Monday.com, UserGuiding Webflow, entre outras (MANOEL, 2022).

5. Materiais e Métodos

Antes de escolher uma das ferramentas *no-code* apresentadas, foi preciso realizar pesquisas. Para o projeto foi escolhida a ferramenta *no-code* Bubble.io (BUBBLE, 2023), por apresentar uma boa documentação e apesar de estar totalmente em inglês, possui uma boa organização, o que facilita compreender melhor suas características, segurança e limites.

O método de cadastro no site Bubble.io é simples, inicialmente apenas e-mail e senha, ou cadastro via Google são solicitados. Pode-se notar que sua principal característica fica evidente nos controles fornecidos por suas ferramentas tanto em modo design quanto nos processos.

O editor do Bubble possui várias opções de elementos, e ainda possibilita criar elementos personalizados dentro do projeto, além de oferecer para os projetos alguns componentes pré-definidos para agilizar o desenvolvimento servindo como *template*.

6. Desenvolvimento do projeto

6.1 Criação de elementos

Ao iniciar o projeto, optou-se por utilizar o mesmo menu em todas as páginas através de um elemento reutilizável, com a finalidade de padronizar o menu em todas as páginas e facilitar manutenções futuras, utilizando todas as partes das páginas como elementos reutilizáveis dentro de um grupo para controlar a responsividade.

Junto a criação do projeto, a própria ferramenta criou três páginas que no projeto são: index, reset_pw e 404, e são páginas que não podem ser apagadas.

A Figura 1 exibe a listagem de páginas do projeto, onde é possível notar o nome das páginas.

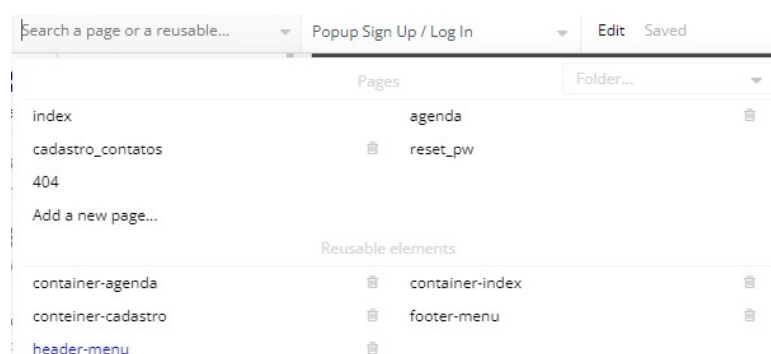


Figura 1 - Menu de navegação do editor entre elementos e páginas. Fonte: autoria própria.

O menu principal é responsável por gerenciar os elementos do projeto, permite deletar, editar e criar páginas e elementos de forma simples.

6.2 Navegação interna e restrições.

O cabeçalho foi desenvolvido para ser um menu de navegação para o usuário, onde a logo redireciona para a página index, que é padrão em vários sites, inclusive o site da documentação do Bubble. Os textos de navegação, agendas e cadastro, direcionam para as páginas, as barras verticais mostram um componente em foco, com um resumo do menu usado na versão responsiva das páginas.

Porém, houve a necessidade de ter um controle sobre o uso da ferramenta para separar os dados por usuário, então adicionou-se um sistema de login usando botões, um para realizar o login e outro para cadastrar um novo usuário. Para demonstrar que o usuário já está acessando a conta foi necessário adicionar outro conjunto de elementos, como pode ver todos os elementos exibidos na figura 2.

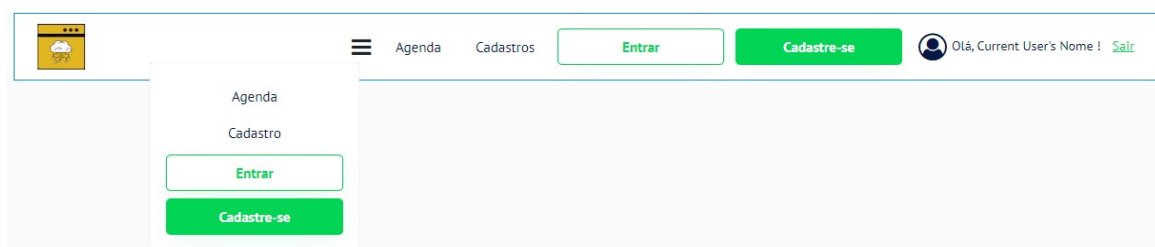


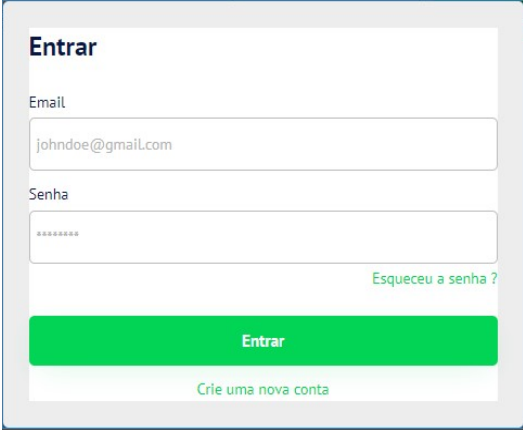
Figura 2 – Menu principal. Fonte: autoria própria

Para criar o controle das restrições e separação em usuários foi necessário consultar a documentação para o entendimento do sistema de banco de dados, onde são armazenados em servidores Amazon's Relational Database Service (RDS) que fazem parte do Amazon Web Services (AWS), também usando como tecnologia para relação com o banco a PostgreSQL, onde por décadas já vem sendo testada e auditada para sua estabilidade e segurança, junto a ser baseada no padrão SQL que é um formato muito utilizado, permitindo compatibilidade com outros bancos de dados.

Ao criar o projeto, a ferramenta também criou um banco de dados especial chamado *User*, esse banco de dados possui alguns campos ocultos ao criador do projeto, que são o *Password* e o *Email confirmed*, o campo password segundo a documentação é devido a seu meio de *Hash* padronizado para todos os usuários da

ferramenta e ambos tem a restrição que só pode ser alterado com ações nas quais devem ser configuradas e colocadas no projeto conforme a necessidade, configurou-se apenas o redefinir senha, incluindo o campo nome no banco de dados.

Por se tratar de um processo simples e com poucos dados a serem coletados optou-se por criar um único *popup* para realizar cadastro e o login, além de criar os processos nos *Workflow*, conforme pode ser visto na Figura 3.



O formulário de login, intitulado "Entrar", apresenta dois campos de entrada: "Email" com o valor "johndoe@gmail.com" e "Senha" com caracteres ocultos por pontos. Um link "Esqueceu a senha?" está posicionado à direita do campo de senha. Um botão verde com o texto "Entrar" está centralizado na base do formulário. Abaixo do botão, há um link "Crie uma nova conta".

Figura 3 - Popup de cadastro e login de usuário. Fonte: autoria própria.

A área de *workflow* é dividida em eventos onde dentro deles existem etapas, bem similares a funções e seus parâmetros, onde todas as etapas são executadas conforme suas condições, mas sempre em sequência, permitindo fazer uma única busca ou modificação no início da etapa e trabalhar com esse dado dentro do evento, onde cada página tem a sua área de eventos isolada das demais, para interação entre páginas, pode usar o envio de dados entre uma e outra através da função de mudança de página e enviar os dados no campo disponível. Em páginas mais extensas pode se criar pastas/grupos de eventos dentro das áreas de *workflow* de cada página conforme a necessidade de separação e organização.

O rodapé foi desenvolvido de forma mais simples, apenas criando e colocando os ícones de cada rede social e definindo as ações de redirecionamento de página.

O *container* da página index foi criado com a finalidade de ser a tela rosto do webapp fazendo receber o usuário não cadastrado ou que não está logado, caso ele saia em outra página será direcionado à esta não alterar ou deletar algum dado, assim finalizando a parte de restrições necessárias para as separações de usuários consequentemente a segurança dos dados individuais de cada usuário.

6.3 Banco de dados

O banco de dados de contatos é composto pelos campos, nome, sobrenome, telefone e e-mail, porém para dar fins mais amplos e poder servir em sistemas de entrega ou registro completo de empresas foi implementado os campos: CEP, cidade, estado (UF), bairro, logradouro, número. Após a criação para divisão de dados por usuários utilizou-se o campo *Creator* do tipo *User*, que é um dos campos criados automaticamente ao gerar um novo banco de dados e que não pode ser apagado, com sua função principal de vincular os dados ao usuário é preenchido automaticamente ao ser criado um registro, os outros campos criados foram *Modified Date*, *Create Date* e *Slug*, com mesmas propriedades, porém com tipos diferentes. A configuração do banco de dados pode ser vista na Figura 4.

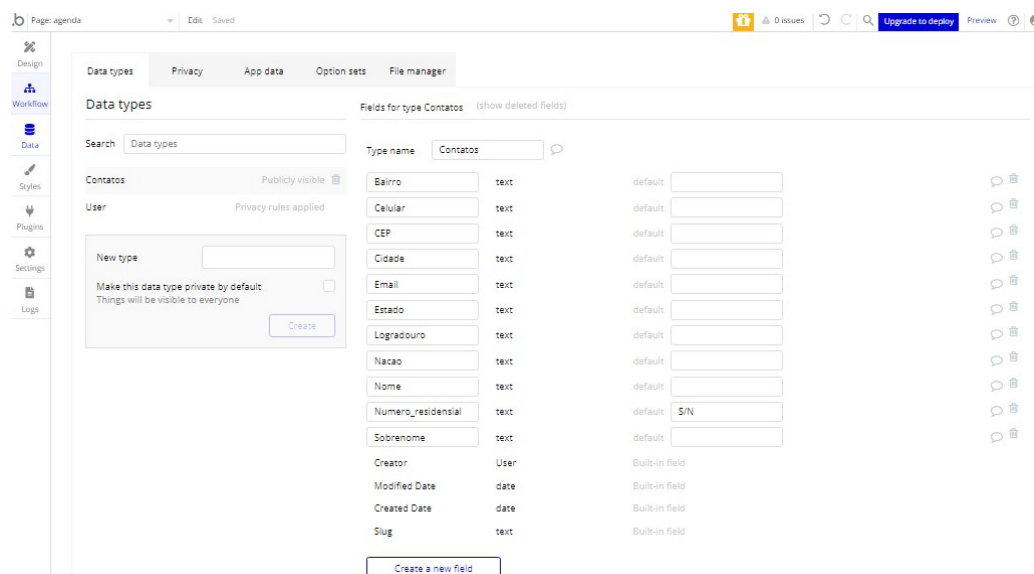


Figura 4 – Banco de dados contatos com campos e tipos definidos. Fonte: autoria própria.

6.4 Visualização e manipulação do banco de dados

A criação de dados foi iniciada com desenvolvimento pela tela de cadastro, que consiste em três botões principais, salvar contato, voltar e editar. O botão voltar tem a simples função de limpar os campos de entrada, limpar os dados da página e redirecionar para a página agendas, onde se exibe uma visualização geral dos dados; o botão editar só irá aparecer quando existirem dados na página, que devem ser passados por uma ação na página agendas transferido como path, ele conterà a função de destravar os campos de entrada e mudar a função dos botões salvar contato

e voltar para salvar edição e cancelar; assim o botão salvar é responsável pelo funcionamento total na manipulação de dados desta página, que consiste em: verificar se contém dados para a edição ou não, se tiver dados para a edição ele modifica os dados de forma local e realiza os mesmos procedimentos essenciais da criação de dados que são automatizados pela ferramenta, logo após faz a limpeza dos dados da página para que possa realizar novos cadastros em sequência, o procedimento de cadastro do botão salvar contato é a criação dos dados de forma local em seguida realiza as funções conectar, registrar e desconectar em sequência automaticamente, evitando possíveis falhas devido a esquecimento de qualquer etapa essencial. A Figura 5 exibe a tela de cadastro de novo contato.

The image shows a web form for registering a new contact. The form is titled "Contato" and has a "Voltar" button at the top right. The fields are: "Nome *" (test), "Sobrenome" (edição), "E-mail *" (test@test.com), "Celular *" (3598888555), "CEP" (99999999), "Cidade" (test), "UF" (UF), "Bairro" (bairro), "Logradouro" (av rua teste), and "Numero" (S/N). A red message below the fields says "Campos com * são obrigatórios!". At the bottom is a green "Salvar contato" button. The top navigation bar includes "Agenda", "Cadastros", and a user profile "Olá, Angelo Gabriel!" with a "Sair" link. A "Built on Bubble" logo is in the bottom right corner.

Figura 5 – Tela de cadastro de novo contato. Fonte: autoria própria

A página agenda é responsável por exibir uma visualização simplificada de todos os contatos, listando apenas os dados definidos como obrigatórios na hora do cadastro, foi adicionado a página um sistema de busca utilizando um campo de entrada e um plugin gratuito desenvolvido em Javascript pela equipe ZEROQCODE em 2019 e atualizado de forma recorrente, com funcionamento simples onde é definido quais campos serão comparados dentro do banco de dados indicado, sinalizando também o campo de entrada de dados e que devolve um *array*. Para exibir os dados foi definido um repetidor de elementos, onde foi atribuído o tipo do banco de dados junto a uma pesquisa filtrada pelo campo Creator, retornando apenas os dados relacionados ao usuário, criou-se um elemento que será repetido em sequência com

os dados em ordem alfabética do campo Nome, nesse elemento foram colocados três componentes com uma função, o primeiro é um *checkbox*, onde serve de seleção de dados do comando *delete* em massa, o segundo elemento é o de visualização (ícone olho), que redireciona para a página de criação, o terceiro o ícone (lixeira vermelha) é o *delete* unitário, que abre um *popup* confirmando se realmente deseja deletar o item, pedindo uma revalidação de senha para realizar a ação. Na página ainda foram adicionados os botões para delete em massa e para a inclusão de novos contatos.

7. Resultados

A funcionalidade de separação de dados por usuário demonstrou eficácia, incluindo as restrições de acesso às páginas de agenda e cadastro. Em caso de um usuário não registrado, ao clicar em "sair", é redirecionado automaticamente para a página inicial, denominada "index". O processo de cadastro de contatos é essencial para a vinculação e armazenamento dos dados do usuário, possibilitando recorrência de uso e facilitando o acesso em diferentes locais, contanto que haja conectividade à internet. Após a conclusão bem-sucedida do cadastro, um aviso em azul é apresentado na tela, informando que o procedimento foi realizado, seguido pelo nome do cadastro; em seguida, os campos são limpos. O sistema de busca, durante testes, demonstrou eficácia em seus resultados, priorizando posições e permitindo buscas por nome, sobrenome, e-mail e telefone. Além disso, há a capacidade de selecionar e deletar contatos, tanto de forma individual quanto múltipla. A visualização completa redireciona para outra página com o envio de dados correspondente.

8. Conclusão

Após os estudos realizados, conclui-se que existe uma falta de profissionais na área de tecnologia para demanda do mercado brasileiro, assim fazendo com que muitas empresas busquem novas metodologias ou ferramentas que possam auxiliar o desenvolvimento o deixando de forma mais rápida, mantendo ou até mesmo melhorando a qualidade dos produtos desenvolvidos, também aumentando a inovação, assim ampliando o conhecimento sobre metodologias do desenvolvimento tradicional que podem ser aplicadas em conjunto com as ferramentas no-code e low-code, melhorando ainda mais usabilidade e eficácia nos desenvolvimentos de novos

projetos, assim sendo uma ótima complementação que pode ser aplicada desde um microprojeto à um grande projeto já em desenvolvimento. Apesar de trazer facilidade nas criações de telas e de manipulação de dados, traz a dificuldade de organização de processos devido às limitações a comentários não visíveis diretamente nas etapas, fazendo que definição de uma legenda de cores para organização seja mais duradoura.

Futuras melhorias que podem ser implementadas, a utilização de uma API externa para o registro de endereços buscados por CEP, a exportação desses contatos para vinculação com uma conta no Google ou até mesmo exportar separadamente os registros para outros sistemas.

9. Referências Bibliográficas

BRASCOM. **SOBRAM VAGAS EM TECNOLOGIA – CRESCIMENTO DO SETOR ACELERA** - 09 maio 2019 **Brasscom**. Disponível em: <<https://brasscom.org.br/sobram-vagas-em-tecnologia-crescimento-do-setor-acelera>>. Acesso em: 09 out. 2023

BUBBLE. **Bubble Docs** -Bubble.io. Disponível em: <<https://manual.bubble.io>> Acesso em: 03 nov. 2023

CLARK, J. **As 10 melhores soluções Low-Code** – 22 set. 2020 **BACK4APP**. Disponível em: <<https://blog.back4app.com/pt/as-10-melhores-solucoes-low-code>> Acessado em: 29 out. 2023

GUEDES, M. **Metodologias ágil x tradicional: Quais as diferenças?** – abr. 2019 **TREINAWEB_**. Disponível em <<https://www.treinaweb.com.br/blog/metodologias-agil-x-tradicional-quais-as-diferencas>> Acessado em: 13 nov. 2023.

KRAJEWSKI, R. **Council Post: The Rise Of No-Code And Low-Code Solutions: Will Your CTO Become Obsolete?** - 14 out 2021 **Forbes**. Disponível em: <<https://www.forbes.com/sites/forbestechcouncil/2021/10/14/the-rise-of-no-code-and-low-code-solutions-will-your-cto-become-obsolete>>. Acesso em: 03 out. 2023

LIMA, F. **O futuro promissor do Low-code e No-code no mercado brasileiro diante da escassez de profissionais de TI** - 18 jun. 2023 **LinkedIn**. Disponível em:

<<https://pt.linkedin.com/pulse/o-futuro-promissor-do-low-code-e-no-code-mercado-brasileiro-lima>>. Acesso em 11 out. 2023

MadeinWeb. **As Top 10 Tendências de Desenvolvimento de Software a serem observadas em 2023.** Plataforma **MadeinWeb**. Disponível em: <<https://madeinweb.com.br/as-top-10-tendencias-de-desenvolvimento-de-software-a-serem-observadas-em-2023> >. Acesso em: 28 set. 2023

MANOEL, E. 7 ferramentas no-code para usar hoje - 15 ago. 2022 **LinkedIn**. Disponível em: < <https://pt.linkedin.com/pulse/7-ferramentas-no-code-para-usar-hoje-ederson-d%C3%A9-manoel->>. Acesso em: 29 out. 2023

NO CODE, **No code: você sabe o que é?** 28 jan. 2022. Plataforma StartSe. Disponível em: <<https://www.startse.com/artigos/no-code-voce-sabe-o-que-e>>. Acesso em: 02 out. 2023

SCHNIDT, J. **Metodologias de desenvolvimento de software tradicionais: Um olhar detalhado** – 04 ago. 2023 **Awari**. Disponível em: <<https://awari.com.br/metodologias-de-desenvolvimento-de-software-tradicionais-um-olhar-detalhado> > Acesso em: 11 nov. 2023

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed, p.02-495, São Paulo: Pearson Prentice Hall, 2011. Disponível em: <www.facom.ufu.br/~william/Disciplinas%202018-2/BSI-GSI030-EngenhariaSoftware/Livro/engenhariaSoftwareSommerville.pdf>. Acesso em: 05 out. 2023

ZEROQCODE. **Fuzzy search & Autocomplete**, 25 mar. 2019. **Bubble**. Disponível em: <<https://bubble.io/plugin/fuzzy-search--autocomplete-1553006094610x835866904531566600>>. Acesso em: 02 nov. 2023