

UNITAU - Universidade de Taubaté

Fernanda Bueno de Oliveira

Leonardo Oliveira de Gouvêa

Protótipo de Aplicação Web para Gerenciamento de Projetos: SCOPE

Taubaté

2020

Fernanda Bueno de Oliveira
Leonardo Oliveira de Gouvêa

Protótipo de Aplicação Web para Gerenciamento de Projetos: SCOPE

Trabalho apresentado para obtenção
do Certificado de Graduação pelo
curso de Engenharia da Computação
do Departamento de Informática da
Universidade de Taubaté,
Área de Concentração: Gestão e Acompanhamento
de Projetos
Orientador: José Alberto Fernandes
Ferreira

Taubaté – SP

2020

**Grupo Especial de Tratamento da Informação - GETI
Sistema Integrado de Bibliotecas – SIBi
Universidade de Taubaté - Unitau**

O48p Oliveira, Fernanda Bueno de
Protótipo de aplicação Web para gerenciamento de projetos : Scope /
Fernanda Bueno de Oliveira, Leonardo Oliveira de Gouvêa. -- 2020.
37 f. : il.

Monografia (graduação) – Universidade de Taubaté, Departamento de
Informática, 2020.

Orientação: Prof. Dr. José Alberto Fernandes Ferreira, Departamento de
Informática.

1. Aplicações Web. 2. Administração de projetos. 3. Escopo de projeto.
I. Gouvêa, Leonardo Oliveira de II. Universidade de Taubaté. Departamento de
Informática. Graduação em Engenharia de Computação. III. Título.

CDD – 658.404

FERNANDA BUENO DE OLIVEIRA
LEONARDO OLIVEIRA DE GOUVÊA
PROTÓTIPO DE APLICAÇÃO WEB PARA GERENCIAMENTO
DE PROJETOS: SCOPE

Trabalho apresentado para obtenção
do Certificado de Graduação pelo
curso de Engenharia da Computação
do Departamento de Informática da
Universidade de Taubaté,
Área de Concentração: Gestão e Acompanhamento
de Projetos.

Data: _____

Resultado: _____

BANCA EXAMINADORA

Prof. Dr. José Alberto Fernandes Ferreira

Assinatura _____

Prof. Mestre Dawilmar Guimarães de Araujo

Assinatura _____

Prof. Especialista Edgar Israel

Assinatura _____

Dedicamos às nossas famílias que nos deram base, apoio e incentivo para chegarmos até aqui.

AGRADECIMENTOS

Ao Prof. Dr. José Alberto Fernandes Ferreira, que nos orientou no desenvolvimento desse projeto.

À banca examinadora do trabalho de graduação, composta pelos Excelentíssimos Professores: Prof. Dr. José Alberto Fernandes Ferreira, Prof. Mestre Dawilmar Guimarães de Araujo e Prof. Especialista Edgar Israel.

Ao Wantuir Felipe da Silva Junior, por compartilhar seu conhecimento e experiência, através de workshops de projetos e artigos escritos por ele mesmo.

Ao Gustavo Marques Martins, por liberar sua propriedade intelectual do HRPC para ser usada no desenvolvimento do projeto.

Ao Arthur Medeiros Bustamante, por criar o design do nome e do logo da aplicação.

A todo corpo docente da Universidade de Taubaté, que nos proporcionou os meios de conhecimento para completarmos nossa graduação.

“Planejamento de longo prazo não lida com decisões futuras, mas com o futuro das decisões presentes”

Peter Drucker

Resumo

Sabemos que ao começar um projeto, as pessoas se voltam muito para controlar prazos, custos, recursos, riscos, sem que o escopo tenha sido claramente entendido pelas equipes que irão efetivamente trabalhar no projeto. Um escopo bem montado é vital para o sucesso do projeto, pois é através dele que conseguimos determinar o orçamento, prazos, recursos, aquisições, fatores de riscos, etc. Assim é possível realizar um projeto sem ter muitos desentendimentos, deixando claro para o cliente o que está sendo feito, porque muitas vezes o cliente tem uma percepção diferente do trabalho que precisa ser feito.

Pensando nisso, este projeto deseja criar um protótipo de aplicação, para gerenciamento de projetos na web, para facilitar ainda mais a comunicação entre o time e a organização das ideias. Tudo isso de forma online, sem ter a necessidade de instalar um aplicativo no seu computador.

Palavras chave: Aplicação web, gerenciamento de projetos, escopo de projeto.

Abstract

Web application prototype for project management

We know that when starting a project, people turn a lot to control deadlines, costs, resources, risks, without the scope having been clearly understood by the teams that will effectively work on the project. A well-assembled scope is vital to the success of the project because it is through it that we can determine the budget, terms, resources, acquisitions, risk factors, etc. So it is possible to carry out a project without having many disagreements, making it clear to the client what is being done, because often the customer has a different perception of the work that needs to be done.

Thinking about the problem described, this project wants to create an application prototype, for project management on the web, to further facilitate communication between the team and the organization of ideas. All of this online, without having to install an application on your computer.

Key word: Web application, project management, project scope.

Lista de Ilustrações

Figura 1 - Tópicos a serem preenchidos na criação do escopo	16
Figura 2 - Ferramentas usadas para o desenvolvimento do Front-end	18
Figura 3 - Ferramentas usadas para desenvolvimento do Back-end.....	19
Figura 4 - Modelo Single-Thread	22
Figura 5 - Exemplo de um Hash feito pelo Bcrypt.....	23
Figura 6 - Exemplo de um token feito pelo JWT encode	24
Figura 7 - Exemplo de schema feito com o Joi	24
Figura 8 - Exemplo de código de validação lido pelo Joi	25
Figura 9 - Exemplo da chamada da API utilizando HRPC	25
Figura 10 - Fluxo da aplicação	27
Figura 11 - Modelagem do projeto	28
Figura 12 - Tela de Login.....	29
Figura 13 - Tela de cadastro	29
Figura 14 - Tela dos projetos.....	30
Figura 15 - Tela de detalhes do projeto.....	30
Figura 16 - Tela criar novo projetoTela criar novo projeto.....	31
Figura 17 - Tela de usuários de um projeto.....	32

Lista de abreviaturas e siglas

API - Application Programming Interface

JWT - Json Web Token

HTTP - Hypertext Transfer Protocol

HTTPS - Hypertext Transfer Protocol Secure

TCC - Trabalho de Conclusão de Curso

JS - JavaScript

TS - TypeScript

XML - Extensible Markup Language

JSX - JavaScript XML

ES5 - Ecma Script 5

ES6 - Ecma Script 6

REST - Representational State Transfer

Sumário

1. Objetivo, justificativa e motivação.	14
2. Introdução	15
3. O que é um Escopo?	16
3.1 Escopo do projeto	16
3.2 Escopo do produto	16
3.3 Objetivo do projeto	17
3.4 Premissas	17
3.5 Restrições	17
3.6 Exclusões	17
3.7 Requisitos	17
3.8 Dúvidas	17
4 Metodologia	18
5 Desenvolvimento	20
5.1 JavaScript	20
5.2 TypeScript	20
5.3 Single Page Application	20
5.4 React JS	21
5.5 Redux	21
5.6 Babel	21
5.7 Node JS.	22
5.8 Material UI	22
5.9 Koa	22
5.10 Bcrypt	23
5.11 JWT	23
5.12 Joi	24
5.13 HRPC	25
5.14 MongoDB	26
5.15 MongoJS	26
5.16 Robo3T	26
5.17 Postman	26
5.18 Fluxo de aplicação	26
6 Modelagem do Projeto	28
7 Funcionamento da Aplicação	29
8 Conclusão	33
9 Propriedade Intelectual	34

1. Objetivo, justificativa e motivação.

O objetivo desse projeto é criar uma aplicação web, que ajude as equipes à organizar o escopo do projeto, acompanhar e documentar todos os passos e mudanças que possam ocorrer durante a sprint, ajudando assim a prever e evitar imprevistos que podem prejudicar o projeto e ajudar na tomada de decisão.

Após realizarmos várias pesquisas na internet, percebemos que os aplicativos de gerenciamento de projetos são voltados para o gerenciamento de tarefas, onde monitora o que precisa ser feito, o que está sendo feito e o que já foi concluído, como: Trello, Monday, Asana entre outros.

Para que um projeto seja bem executado, é preciso ter um escopo bem definido e notando essa ausência de aplicações que auxiliem na hora de criação desse escopo, resolvemos criar esse protótipo de aplicação de gestão de escopo, para ajudar a melhorar a visibilidade do projeto, melhorar a organização e facilitar o entendimento do projeto como um todo.

Muitas vezes durante um projeto, os integrantes do grupo entendem de formas diferentes o objetivo do projeto e acabam assumindo que tal processo deve ser feito de tal jeito e essa falta de comunicação pode prejudicar o desenvolvimento do projeto e o resultado final. Se estiver tudo mapeado, organizado e com o histórico de responsáveis por mudanças, facilita encontrar e questionar o porquê aquela decisão foi tomada e o impacto que vai ter no projeto, conseguindo assim, fazer mudanças necessárias para garantir um bom resultado ao final do projeto.

A aplicação contará com uma área de brainstorming, contendo: objetivo, premissas, restrições, exclusões, requisitos, dúvidas, dúvidas respondidas, onde qualquer pessoa do projeto poderá interagir, terá também uma área para documentar as decisões tomadas em cada reunião e o dono do projeto, poderá gerar relatórios em pdf.

2. Introdução

No mundo atual, as organizações encontram um alto nível de competitividade, que fazem com que o desenvolvimento de um projeto seja cada vez mais complexo, pois além da dificuldade do projeto, temos uma rotatividade de pessoas cada vez maior em um projeto, o que leva à perda de informação.

Com o tempo, para facilitar o andamento do projeto, foram criados vários métodos e abordagens diferentes para o projeto. Um projeto consiste de um esforço temporário empreendido com objetivo pré-estabelecido, definido e claro, seja criar um novo produto, serviço, processo. Tem início, meio e fim determinados, duração e recursos limitados, em uma sequência de atividades relacionadas. Para conseguir organizar todas essas frentes, foram criadas algumas soluções na área de projetos.

3. O que é um Escopo?

O Escopo tem a função de determinar e documentar os objetivos específicos de cada projeto, como suas entregas, tarefas, custos e prazos. Ao longo de um projeto, essa documentação ajuda seu time a permanecer focado, fornece aos líderes as diretrizes necessárias para tomada de decisões e norteia sua execução.

Nesse protótipo de aplicação iremos abordar os seguintes escopos:

3.1 Escopo do projeto

O trabalho que deve ser realizado para entregar um produto, serviço ou resultado com as características e funções especificadas.

3.2 Escopo do produto

As características e funções que descrevem um produto, serviço ou resultado

Para iniciar um projeto, é essencial coletar algumas informações iniciais para elaborar o escopo, sendo elas:

Para conseguir estruturar um escopo de projeto, existe alguns tópicos que devem ser preenchidos, visando esclarecer e guiar o projeto.

Esses tópicos serão mostrados na figura abaixo (Figura 1) e esclarecidos nos textos a seguir.

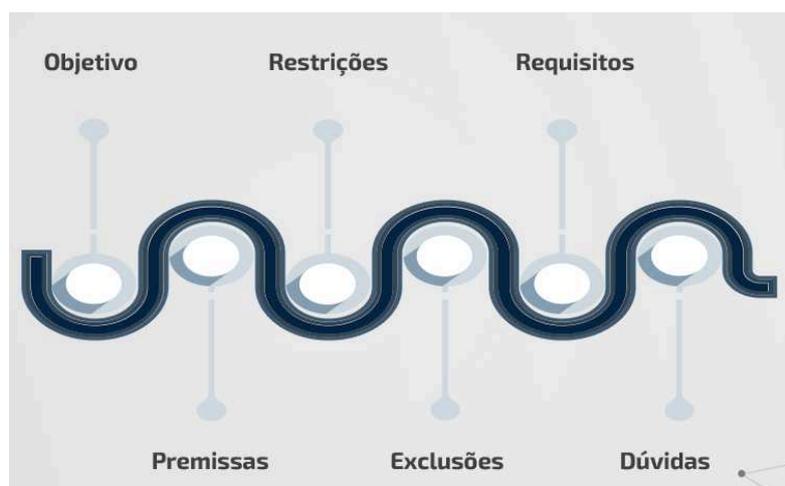


Figura 1 - Tópicos a serem preenchidos na criação do escopo

Fonte: Slides de apresentação do TCC (Própria 2020)

3.3 Objetivo do projeto

É importante ter uma direção no projeto, como por exemplo: Qual é o trabalho a ser desenvolvido e/ou executado? Quem é o cliente e/ou usuário da solução? Para quem será destinada esta solução?

3.4 Premissas

Os fatores que, para fins de planejamento, são considerados verdadeiros, reais ou certos sem prova ou demonstração. Lembrando que premissas são possíveis direcionadores de riscos, cabendo análise e investigação.

3.5 Restrições

Imposição “contratual” ou limitação de custo, prazo ou recurso, lembrando que restrições precisam de responsáveis para respeitá-las ou tratá-las. Elas precisam ser consideradas e, implicitamente, fazer parte do escopo do projeto.

3.6 Exclusões

Os objetos ou ações que não fazem parte do escopo do projeto. Seu levantamento e divulgação evita suposições equivocadas e/ou trabalho desnecessário.

3.7 Requisitos

Definir e documentar as necessidades das partes interessadas para atingir os objetivos dos projetos, que devem ser obtidos, analisados e registrados em detalhes suficientes para serem medidos durante a execução do projeto.

3.8 Dúvidas

Levantamento benéfico de dúvidas que depois de esclarecidas terão impacto positivo no escopo técnico e escopo de atividades.

4 Metodologia

Ao pensarmos nas funcionalidades da aplicação, nos vimos na necessidade de desenvolver o front-end do projeto utilizando a tecnologia de Single Page Application. Com isso em mente, optamos por usar React, que é um framework da linguagem JavaScript, pois ele utiliza tecnologias de estados e consumo de rotas do backend via HTTPS já integrados.

Seguindo a mesma linha de raciocínio do front-end, optamos também por utilizar no back-end a linguagem JavaScript, para não haver uma divergência de linguagem entre o front-end e o back-end. A framework usada no back-end, é o NodeJS e Koa, onde o NodeJS irá ser o esqueleto do back-end e o Koa criará as rotas HTTPS, para serem consumidas pelo front-end.

Para armazenar os dados da aplicação, optamos por um banco de dados não-relacional, chamado MongoDB, pois os bancos de dados não-relacionais utilizam uma estrutura baseada em documento JSON. Esse banco de dados será integrado à back-end, utilizando MongoJS, que é um core builder para MongoDB, .

Para o desenvolvimento do front-end foram utilizadas as ferramentas mostradas na figura 2, dentre elas haverá uma explicação mais detalhada de cada uma delas na próxima seção “Desenvolvimento”.

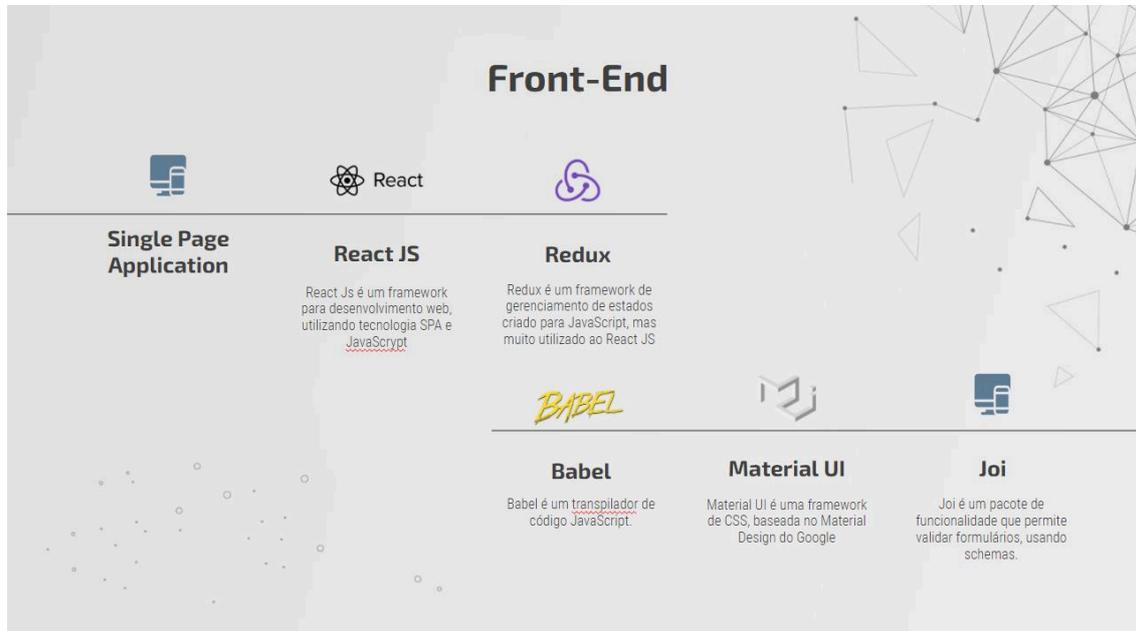


Figura 2 - Ferramentas usadas para o desenvolvimento do Front-end

Fonte: Slides de apresentação do TCC (Própria 2020)

Seguindo, temos a etapa de Back-end onde foram utilizadas as ferramentas mostradas na figura 3, uma melhor descrição destas ferramentas se encontra na próxima seção “Desenvolvimento”.

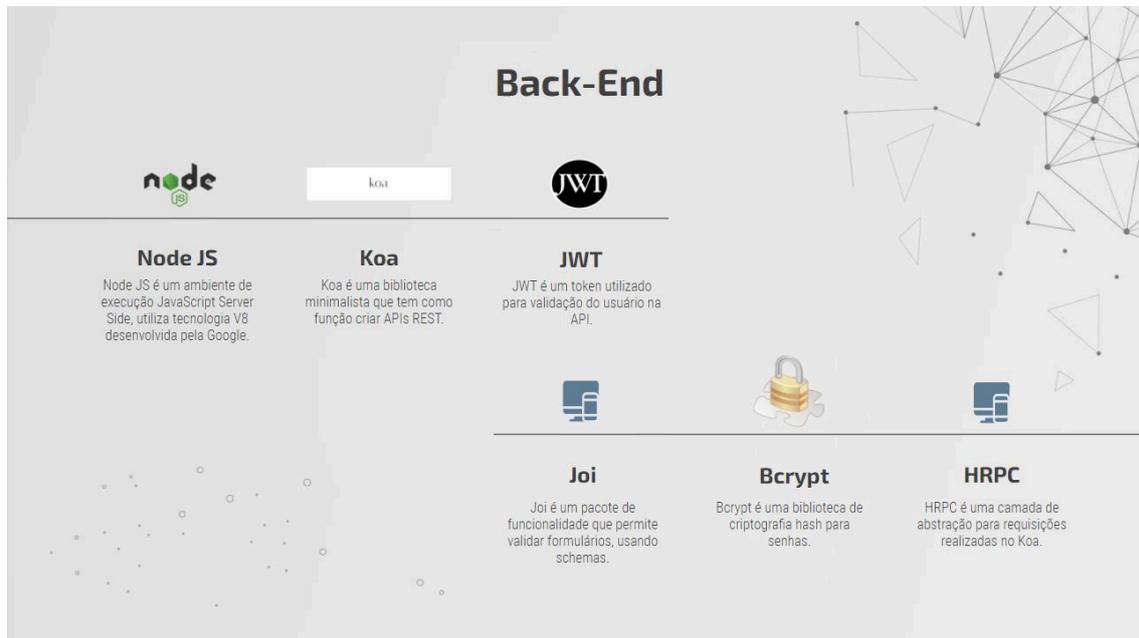


Figura 3 - Ferramentas usadas para desenvolvimento do Back-end

fonte: Slides de apresentação do TCC (Própria 2020)

5 Desenvolvimento

5.1 JavaScript

JavaScript é uma linguagem de programação que permite implementar itens complexos em páginas web. Permite criar conteúdo que se utiliza dinamicamente, controlar multimídias, imagens animadas e fazer o armazenamento de conteúdo útil em variáveis.

O JavaScript conta com a funcionalidade de API, que é um conjunto de definições e protocolos usados no desenvolvimento e na integração de software de aplicações.

A API oferece a flexibilidade necessária para simplificar o design, a administração e o uso, além de fornecer oportunidades de inovação. É possível fazer a sua solução ou serviço se comunicar com outros serviços sem precisar saber como foram implementados, gerando assim uma economia de tempo e dinheiro.

5.2 TypeScript

TypeScript é um superset para JavaScript, que adiciona ao mesmo, a definição de tipagem estática. Esses tipos são uma maneira de descrever a forma de um objeto provendo ao mesmo uma melhor documentação do código e possibilitando ainda a validação se o mesmo funcionará corretamente. E todo código desenvolvido em JavaScript é aceito pela linguagem TypeScript, porém a TypeScript não é aceita pelo JavaScript. Então qual a vantagem de utilizar TS ao invés do JS? A vantagem se dá a nível de código e técnico pois ao fim do desenvolvimento iremos utilizar uma biblioteca que converte todo o código feito em TS para JS, assim rodando em qualquer browser.

5.3 Single Page Application

O Single Page Application é uma aplicação web que interage com o browser de forma dinâmica reescrevendo a atual página web com novas informações vindas do servidor. Diferencia-se do método comum que deveria navegar para outra página para encontrar novas informações e novos componentes vindos do servidor. O objetivo principal é fazer transições mais rápidas que faça a aplicação web se parecer mais com uma aplicação nativa.

Na primeira requisição para a URL da aplicação, todos códigos HTML, JavaScript e CSS necessário é puxado pelo browser em um único carregamento, ou todo recurso apropriado para aquela página é dinamicamente puxado em resposta a ação do usuário. A página não vai sofrer recarregamento em nenhum ponto do processo, nem transfere controle para outra página da mesma aplicação. Mesmo com tudo isso, a localização atual da página, no caso URL, é guardada pelo browser de forma a montar a percepção de uma navegação de páginas separadas.

5.4 React JS

O React é uma biblioteca de JavaScript de código aberto, com o intuito de criar aplicações web Single Page Application no frontend. As empresas que desenvolvem essa ferramenta são: Facebook, Instagram e a comunidade desenvolvedora.

Segundo a própria documentação do React, sua principal função é criar UIs interativas de forma fácil, criar view simples para cada estado da sua aplicação, componentizar UI para que elas sejam utilizadas e reutilizadas em suas respectivas etapas, criando componentes encapsulados que gerenciam seu próprio estado, tornando assim possível a criação de UIs complexas.

O React JS utiliza uma forma de escrita parecida com o HTML denominada JSX, apesar de muito similar ao HTML, o JSX não é interpretado pelo navegador, com isso sendo necessário uma biblioteca externa para transformar o código do padrão de escrita para o padrão de leitura HTML do browser.

“Criado pela equipe de desenvolvimento do React, o JSX é uma forma de criar elementos para serem utilizados como templates de aplicação React.”

5.5 Redux

Redux é uma biblioteca para armazenamento de estados, criado por Dan Abramov. Foi criado com a ideia de simplificar o controle dos estados de uma aplicação, pois um problema fundamental no React, é que cada componente tem seus próprios estados, tornando uma aplicação React grande em um caos. Pensando nisso, o Redux traz uma ideia de compartilhar estados entre vários componentes diferentes, o que torna mais simples a manutenção de componentes React.

Para realizar essas funções, o Redux utiliza três recursos: Store, onde são guardados os dados que serão lidos pelos outros componentes, elas necessitam de uma action para ser alterados os valor que nela habitam; Actions, são funções que podem ser disparadas dentro de componentes para a alteração de dados dentro da Store. Uma vez alterado o dado dentro da Store, todos os componentes dependentes dela receberão a nova informação; Reducers, cada dado dentro da Store deve possuir seu próprio Reducer, que é uma abstração de funções da Store que especificam como o estado da aplicação irá mudar de acordo com cada Action.

5.6 Babel

Babel é um transpilador de código TypeScript (ou JavaScript ES6 para cima) em JavaScript ES5 compatível com browsers e ambientes que possuem versões mais antigas, isso é feito para ter certeza de que nossa aplicação rodará da mesma forma em todos os ambientes que rodam JavaScript.

Por que não utilizar ES5 diretamente? Usar TypeScript traz ao desenvolvedor acesso a mais novas novidades da linguagem e utilização da tipagem em suas variáveis, trazendo assim vantagens a longo prazo, como por exemplo: melhor manutenção, facilidade de leitura do código, debugar erros de valor errôneos chegando em determinadas variáveis.

5.7 Node JS.

O NodeJS é um ambiente de execução JavaScript server-side, ou seja, roda como uma aplicação standalone em uma máquina, sem precisar usar um browser. A característica principal do NodeJS é que ele usa somente uma Thread, como mostrado na figura 4, chamada de single-thread, dessa maneira menos recursos computacionais são exigidos, pois não é necessário criar uma nova thread para cada requisição recebida.

Utiliza-se do chamado Event Loop, que cria novos eventos a cada requisição recebida, desta forma não precisando esperar a finalização de uma operação para que outra seja iniciada.

Modelo single-thread

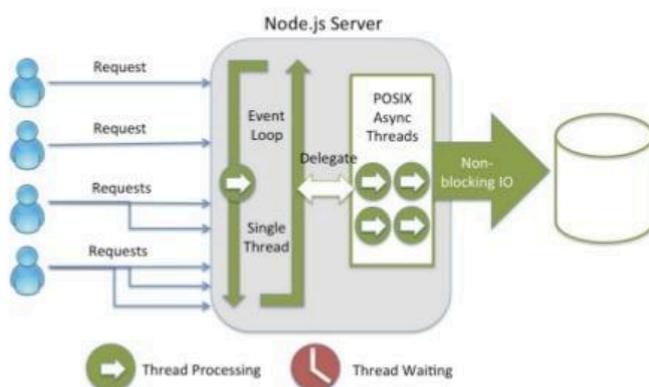


Figura 4 - Modelo Single-Thread

fonte: Performace Event Loop NodeJS (Quora 2016)

5.8 Material UI

Material UI é um framework de CSS, baseada no Material da Google. Esse framework cria componentes abstraídos do Material da Google, componentes esses escritos em React, que podem ser utilizados e modificados dentro da aplicação, facilitando assim o uso deste framework.

5.9 Koa

Koa é uma biblioteca minimalista baseada no ExpressJS, que tem como função principal ser uma base para criação de API's REST. A escolha do Koa ao invés do ExpressJS se dá por ser menor (deixando mais leve o produto final), não possui nenhum middleware pré instalado, trabalhar *out of the box* com funções assíncronas e melhor controle de erro.

5.10 Bcrypt

Bcrypt é um método de criptografia do tipo hash para senhas. Essa criptografia é utilizada quando o usuário faz o cadastro na aplicação transformando a senha do mesmo em um hash. Hash é uma criptografia de via única, uma vez gerado, o hash é praticamente impossível de reverter, sendo assim para validar a senha do usuário é criado um novo hash e comparado com o que se tem no banco, se eles forem iguais a senha condiz e o usuário pode ser logado, um exemplo de hash pode ser visto na figura 5.



Figura 5 - Exemplo de um Hash feito pelo Bcrypt

fonte: Segurança de senhas parte 3 (Engineyard 2014)

5.11 JWT

A autenticação de usuário vai se dar por forma do JSON Web Token (JWT), onde o usuário irá enviar para a rota de login as suas informações, no caso e-mail e senha, e a API devolverá um Token, figura 6, este token é criptografado e possui dentro dele os dados principais deste usuário. Quando o usuário precisar fazer qualquer requisição para a API ele irá enviar no header da requisição o seu token, desta forma a API irá criptografar e verificar a autenticidade do mesmo, validando assim a autenticidade do usuário e que a requisição dele pode ser aceita. Outra vantagem é que na requisição podemos ter acesso aos dados desse usuário sem precisar olhar no banco pois a mesma já vem na payload do JWT.

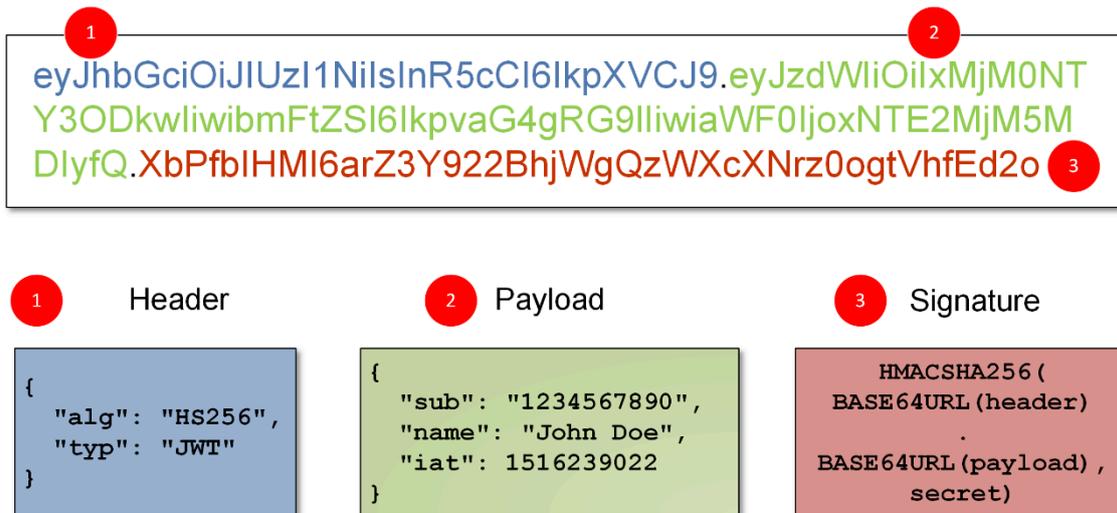


Figura 6 - Exemplo de um token feito pelo JWT encode

fonte: Segurança com Json Web Token (Securitem 2019)

5.12 Joi

Joi é uma poderosa linguagem de descrição de esquemas (schema description language) e testador de esquema (schema tester), esse descritor de esquema é usado para descrever os parâmetros (figura 7) que serão usados e uma vez recebidos os parâmetros testar (figura 8) e ver se ele condiz com o que foi descrito antes. Ele também gera erros se os parâmetros não forem iguais ao esquema lançará um erro na resposta da requisição

```
const schema = Joi.object().keys({
  nome: Joi.string().required(),
  dataDeNascimento: Joi.date().required(),
  email: Joi.string().email().required()
});
```

Figura 7 - Exemplo de schema feito com o Joi

fonte: Exemplo schema escrito em JOI (Própria 2020)

```
const { error } = schema.validate({
  nome: "Nome do usuario",
  dataDeNascimento: "10/07/1995",
  email: "usuario@email.com"
});
```

Figura 8 - Exemplo de código de validação lido pelo Joi

fonte: Exemplo validação de schema escrito em JOI (Própria 2020)

5.13 HRPC

O HRPC é uma biblioteca desenvolvida por Gustavo Marques Martins, que emprestou a Inteligência Intelectual para usarmos neste projeto. Esta lib é uma camada de abstração das requisições realizadas no Koa, onde ele estende a requisição de rotas do Koa e insere uma camada que transforma todas as rotas em POST e recebe como parâmetros o nome do método e os parâmetros que o método irá receber em um modelo de JSON (figura 9). Dentro dessa camada de abstração ele irá ler o JSON identificar o método verificar a tipagem escrita para o método em TypeScript e validar todos os parâmetros usando Joi, feito tudo isso ocorre assim o encaminhamento e execução da função.

```
{
  "method": "nome do metodo que vc quer acessar",
  "params": {"objeto que será enviado para o método"}
}
```

Figura 9 - Exemplo da chamada da API utilizando HRPC

fonte: Exemplo de Requisição HRPC (Própria 2020)

5.14 MongoDB

MongoDB é um Banco de dados não relacional (no-SQL) orientado a documento JSON, deixando assim a modelagem de dados muito mais natural, podendo ser aninhados de forma bem complexas e ainda sim sendo indexável para facilitar a busca do mesmo. A escolha desse tipo de banco se deve a compatibilidade que ele possui com a linguagem JavaScript.

5.15 MongoJS

O MongoJS é uma biblioteca JavaScript utilizada para abstrair funções do banco MongoDB, é uma biblioteca simples possuindo somente a função de conexão ao banco, criação de ObjectId (id usando no banco no-SQL) e query builder para buscar/insserir/deletar/editar dados no banco pelo back-end.

Apesar de existirem concorrentes ao MongoJS, utilizaremos nesse projeto o MongoJS pois o mesmo é um query builder e não um ORM onde o mesmo possui uma complexidade muito maior exigindo criar modelos e migrations para utilização.

5.16 Robo3T

Robo3T é uma ferramenta de visualização de banco de dados Mongo DB. Conecta-se ao banco de dados e mostra as collections que existem dentro dele, permite também criar Querys dentro do banco, proporcionando assim inserção, edição, remoção e filtragem de dados.

5.17 Postman

O Postman é uma ferramenta que dá suporte a documentação de requisições feitas pela API. Também possui funções como documentação de rotas da API, execução e testes da API e requisições gerais.

Neste projeto iremos utilizar o Postman, pois o mesmo facilita os testes das rotas criadas pela API, não necessitando criar um código ou interface visual para interagir com essa rota.

5.18 Fluxo de aplicação

O fluxo da aplicação é a requisição para a API enviada do front a partir do site desenvolvido em React. Essa requisição será tratada pelo servidor NodeJS onde será realizada toda a lógica por trás da ação. Se necessário ele também contata o banco de dados fazendo a requisição necessária (insserir, deletar, editar, procurar), feito isso ele devolve uma resposta a aplicação React com o resultado da operação, sendo ela um erro ou sucesso. Tal fluxo é mostrado na figura 10.

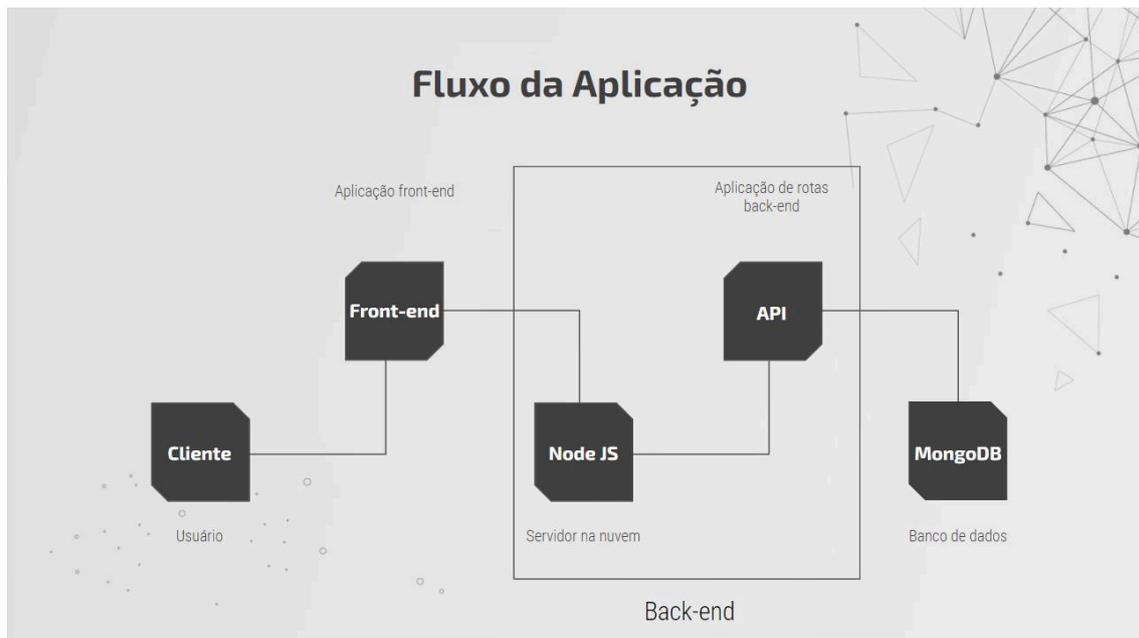


Figura 10 - Fluxo da aplicação

fonte: Slides TCC (Própria 2020)

6 Modelagem do Projeto

A modelagem inicial do projeto pode ser descrita a seguir na figura 11, o projeto se divide em usuário, com as funcionalidades de login e cadastro.

Meus Projetos, representa a criação e visualização de projetos novos e em andamento

Projeto, área onde o projeto pode ser visualizado em detalhes.

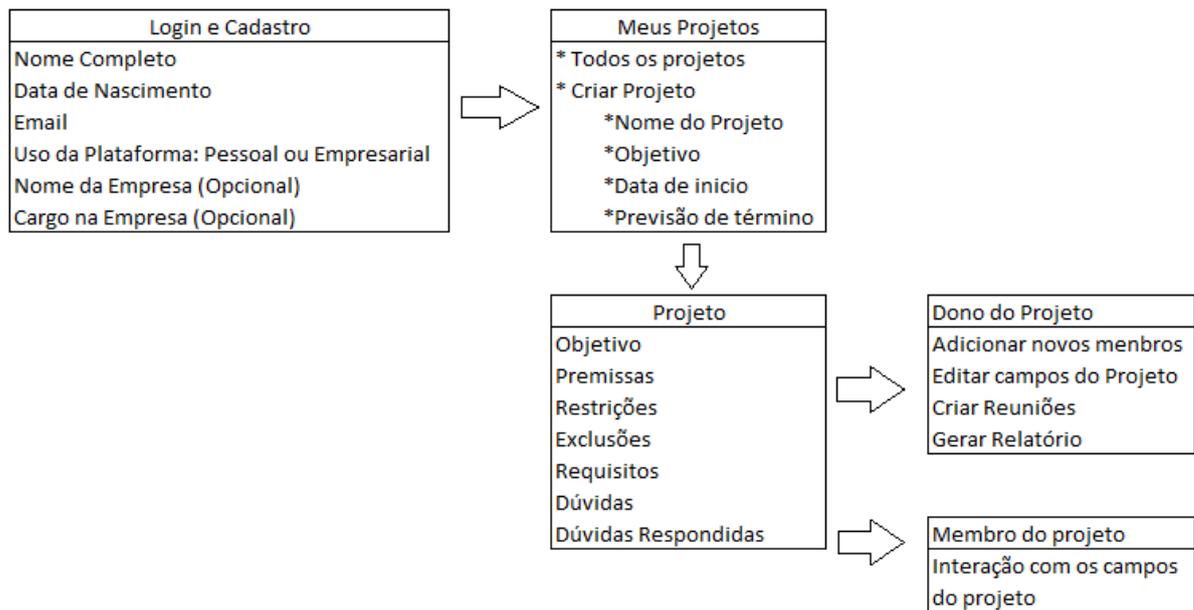


Figura 11 - Modelagem do projeto

fonte: Modelagem do Projeto (Própria 2020)

7 Funcionamento da Aplicação

Tela de Login e Cadastro (figura 12 e 13): Será a tela onde o usuário poderá entrar no aplicativo ou criar uma nova conta utilizando alguns dados, como:

- Nome Completo
- Data de Nascimento
- E-mail
- Senha

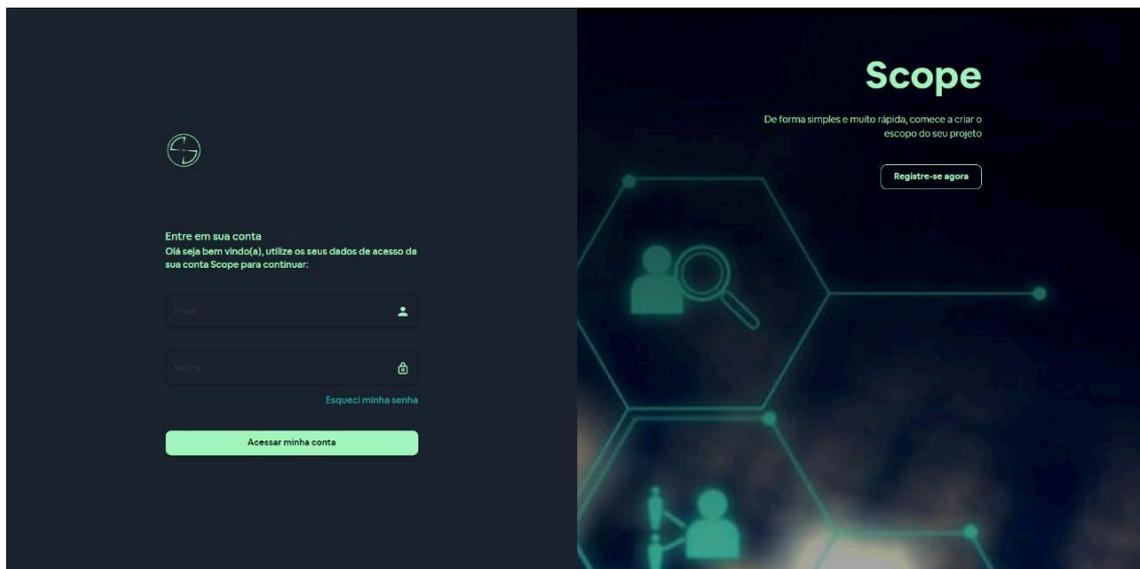


Figura 12 - Tela de Login

fonte: Tela de Login da aplicação (Própria 2020)

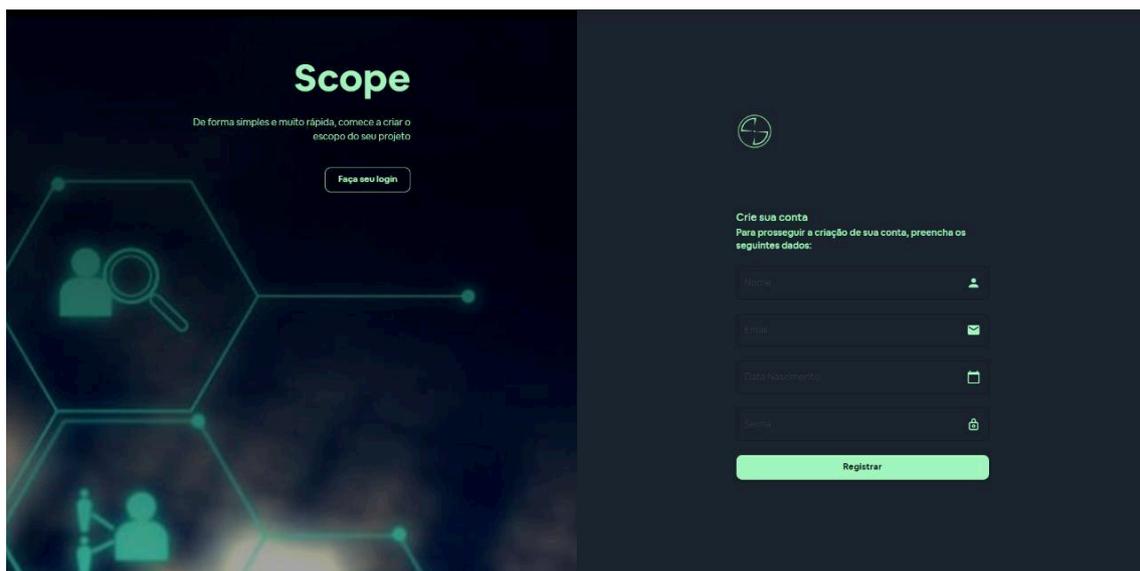


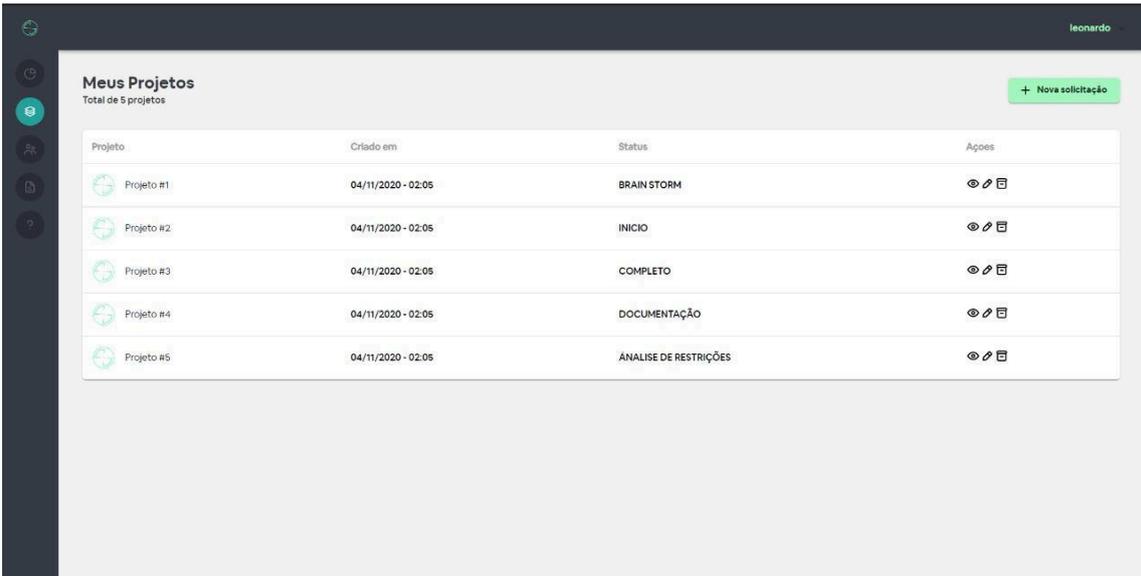
Figura 13 - Tela de cadastro

fonte: Tela de cadastro da aplicação (Própria 2020)

Tela inicial (figura 14):

Meus Projetos: Nesta tela, irá aparecer os projetos que o usuário atualmente participa e estão em andamento.

Na tela de projetos, irá conter um botão para gerar relatórios do projeto em pdf. O relatório irá conter dados do projeto e dados do andamento do projeto.

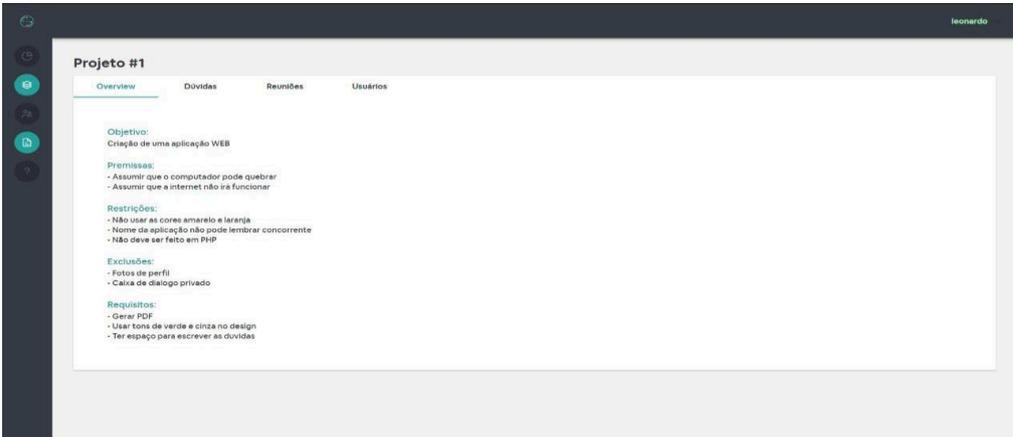


Projeto	Criado em	Status	Ações
Projeto #1	04/11/2020 - 02:05	BRAIN STORM	👁️ ✎️ 🗑️
Projeto #2	04/11/2020 - 02:05	INICIO	👁️ ✎️ 🗑️
Projeto #3	04/11/2020 - 02:05	COMPLETO	👁️ ✎️ 🗑️
Projeto #4	04/11/2020 - 02:05	DOCUMENTAÇÃO	👁️ ✎️ 🗑️
Projeto #5	04/11/2020 - 02:05	ANALISE DE RESTRICÇÕES	👁️ ✎️ 🗑️

Figura 14 - Tela dos projetos

fonte: Tela de “Meus Projetos” da aplicação (Própria 2020)

- Ao clicar em um projeto, você será redirecionado à tela de detalhes desse projeto, figura 15.



Projeto #1
<p>Objetivo: Criação de uma aplicação WEB</p> <p>Premissas: - Assumir que o computador pode quebrar - Assumir que a internet não irá funcionar</p> <p>Restrições: - Não usar as cores amarelo e laranja - Nome da aplicação não pode lembrar concorrente - Não deve ser feito em PHP</p> <p>Exclusões: - Fotos de perfil - Caixa de diálogo privado</p> <p>Requisitos: - Gere PDF - Usar tons de verde e cinza no design - Ter espaço para escrever as dúvidas</p>

Figura 15 - Tela de detalhes do projeto

fonte: Tela de “Detalhes” da aplicação (Própria 2020)

- Tela criar novo projeto (figura 16): Ao clicar em novo projeto, abrirá um novo modal para a criação do projeto, onde pedirá as seguintes informações:

- Nome do Projeto

-Objetivo: Qual é o trabalho a ser desenvolvido e/ou executado? Quem é o cliente e/ou usuário da solução?

- Budget do projeto: Qual o investimento que o projeto irá ter para ser realizado

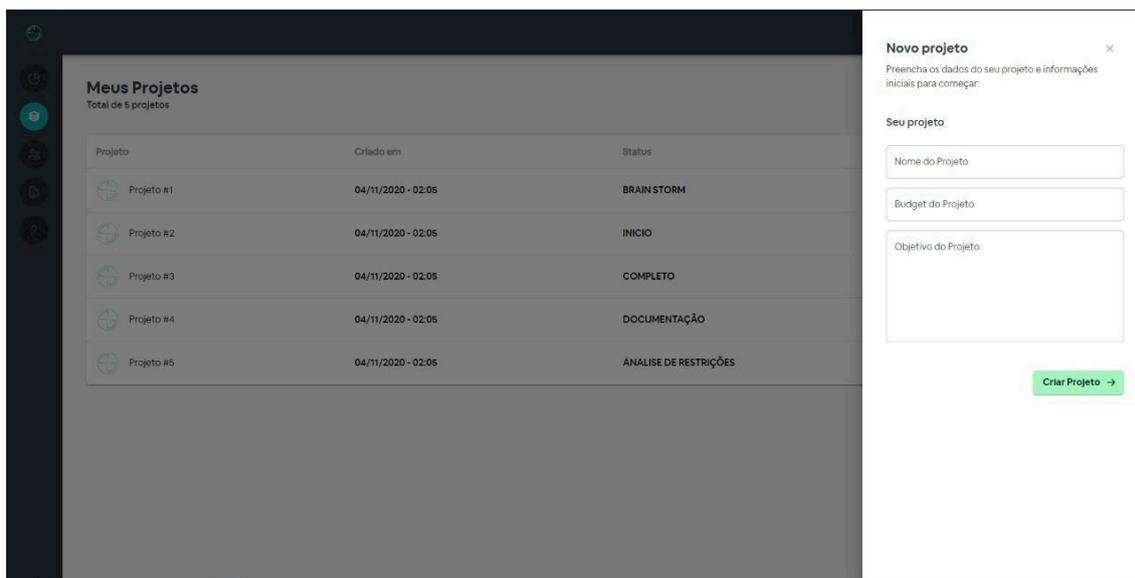


Figura 16 - Tela criar novo projetoTela criar novo projeto

fonte: Modal de criação de novo Projeto (Própria 2020)

Tela de projeto: Essa tela, será dividida em dono do projeto, membros da equipe e uma tela de reunião.

A tela do dono do projeto, irá gerenciar o momento atual do projeto, podendo por exemplo, adicionar novos usuários no projeto, editar os campos e dar acesso para outro membro editar, pode criar reuniões que irão definir o andamento do projeto, com escopos e históricos.

A tela do membro da equipe, será um espaço onde ele poderá interagir com os membros da equipe, comentando e dando opiniões nos campos do projeto Objetivo, premissas, restrições, exclusões, requisitos, dúvidas e dúvidas respondidas).

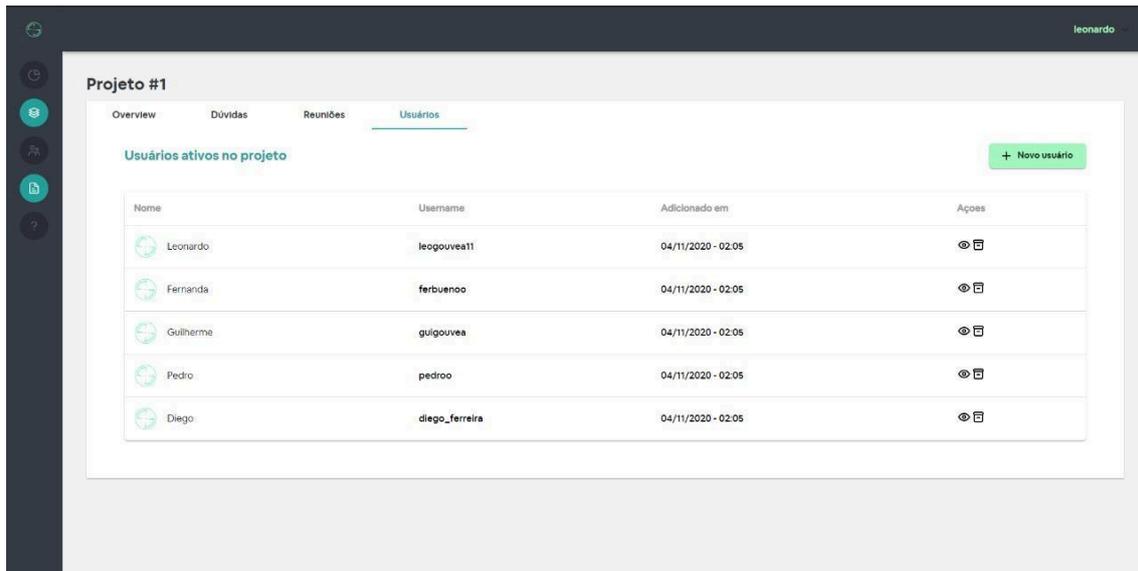


Figura 17 - Tela de usuários de um projeto

fonte: Tela de “Usuários” da aplicação (Própria 2020)

A tela de reunião, será um espaço onde será documentado tudo o que foi decidido nessa reunião e salvo no histórico, como: ata da reunião, participantes da reunião, data da reunião, ações que saíram dessa reunião.

8 Conclusão

Conclui-se que com a aplicação web de gerenciamento de escopo, Scope, é possível ganhar eficiência na área de gestão de projetos, onde há uma deficiência na organização do escopo de um projeto. Pensando nisso, a aplicação surgiu como uma ideia de solução para este problema, organizando as ideias do projeto, de forma clara e em uma única aplicação.

Dentro da aplicação, a pessoa responsável irá ter uma visão mais ampla do projeto em si e com isso ajudar no alinhamento do time para melhorar o entendimento do projeto pelas partes interessadas, esclarecendo as funcionalidades, requisitos, premissas e restrições, proporcionando assim uma análise mais clara dos riscos do projeto, levantando potenciais problemas e direcionando as ações para os responsáveis que irão conseguir achar uma solução.

O escopo é o primeiro passo para a criação de um projeto, tendo um escopo bem organizado e alinhado, os próximos passos para dar continuidade ao projeto, serão mais fáceis de acompanhar os resultados, irá melhorar a comunicação entre os times, colocando as pessoas certas para o desenvolvimento, reduzindo assim o homem-hora de determinada etapa, diminuindo os atrasos nas entregas e distribuindo melhor o budget do projeto.

A aplicação atualmente se encontra como um protótipo, onde ainda há muita margem de criação antes de seu lançamento para testes, porém pesquisando com profissionais da área de projetos, entende-se que eles veem potencial na ideia de ter uma ferramenta focada em gerência de escopo do projeto, para assim dar continuidade de forma clara e objetiva ao mesmo.

Para futuras melhorias, sugere-se a implementação de novas funcionalidades que levem o software a ser menos passivo, havendo uma maior comunicação entre os membro da equipe e não só do líder do projeto.

9 Propriedade Intelectual

HRPC - Gustavo Marques Martins

10 Bibliografia

A RELEVÂNCIA DO ESCOPO DO PROJETO - Wantuir Felipe da Silva Junior e Fábio de Toledo Fontana – SETI – Seminário Embraer de Tecnologia e Inovação.

Alexandre Rocha Marcondes. IMasters, 2020, Single Page Applications e outras maravilhas da web moderna

Disponível em <<https://imasters.com.br/desenvolvimento/single-page-applications-e-outras-maravilhas-da-web-moderna/>>

Acesso em: 18 de mai de 2020

Andrei L. Hostinger, 2020, O que é react e Javascript

Disponível em <<https://www.hostinger.com.br/tutoriais/o-que-e-react-javascript/>>

Acesso em: 2 de jun de 2020

Andreia Silva Justu, Euax, 2018. Gerenciamento de Escopo do Projeto

Disponível em: <<https://www.euax.com.br/2018/12/gerenciamento-escopo-do-projeto/>>

Acesso em: 15 de dez de 2019

Ana Paula de Andrade. TreinaWeb, 2020, O que é JSX

Disponível em: <<https://www.treinaweb.com.br/blog/o-que-e-jsx/>>

Acesso em: 20 de set de 2020

Auth0. JWT, 2020, JWT Introduction

Disponível em <<https://jwt.io/introduction/>>

Acesso em: 12 de abr de 2020

Facebook Open Source Inc. ReactJs, 2020, Introduction to reactjs

Disponível em <<https://pt-br.reactjs.org/>>

Acesso em: 19 de mai de 2020

Facebook Open Source. ReactJS, 2020, Getting Started React JS

Disponível em <<https://pt-br.reactjs.org/docs/getting-started.html/>>

Acesso em: 15 de jun de 2020

Koa Inc. KoaJS, 2020, koa Documentation

Disponível em <<https://koajs.com/>>

Acesso em: 23 de jul de 2020

Lenon. Opus Software, 2020, Node.js – O que é, como funciona e quais as vantagens

Disponível em <<https://www.opus-software.com.br/node-js//>>

Acesso em: 2 de jun de 2020

Marylene Guedes. TreinaWeb, 2020, O que é Redux

Disponível em: <<https://www.treinaweb.com.br/blog/o-que-e-redux/>>

Acesso em: 20 de set de 2020

MaterialUI. MaterialUI, 2020, MaterialUI Documentation

Disponível em <<https://material-ui.com/pt/>>

Acesso em: 12 de abr de 2020

Microsoft. TypeScript, 2020, TypeScript handbook

Disponível em <<https://www.typescriptlang.org/docs/handbook/intro.html/>>

Acesso em: 25 de jul de 2020

MongoDB inc. MongoDB, 2020, MongoDB Documentation

Disponível em <<https://docs.mongodb.com/>>

Acesso em: 12 de abr de 2020

Open JS Foundation. NodeJs, 2020, Documentation NodeJS

Disponível em <<https://nodejs.org/en/docs/>>

Acesso em: 1 de jun de 2020

Postman Inc. Postman, 2020, Getting Started Postman Documentation

Disponível em <<https://learning.postman.com/docs/getting-started/introduction/>>

Acesso em: 15 de jun de 2020

Public. BCrypt, 2020, Bcrypt lib

Disponível em <<https://www.npmjs.com/package/bcrypt/>>

Acesso em: 25 de jul de 2020

Sumus. Sumus, 2018. Aplicativos de Gestão

Disponível em: <<https://www.sumus.com.br/publicacoes/aplicativos-de-gestao/>>

Acesso em: 23 de mar de 2020

Thiago França. IMasters, 2020, Importância escopo de projetos

Disponível em <<https://imasters.com.br/devsecops/importancia-escopo-projeto/>>

Acesso em: 18 de mai de 2020