



Universidade Federal de Itajubá

TRABALHO 1

COM241 – Administração e Gerência de Redes de Computadores

Gabriel Jose Moualem Rodrigues

2017017731

Israel Ferreira Morais

2018003534

Matheus Silva Souza

2017016064

Ygor Salles Aniceto Carvalho

2017014382

Professor: Bruno Guazelli Batista

Novembro

2020

1 - Introdução do sistema desenvolvido

Dia a dia, temos a necessidade de sair do conforto de nossos lares para que possamos nos sustentar, seja para comprar mantimentos, ir a um atendimento médico, visitar amigos e família, comparecer ao trabalho ou escola, e claro lazeres em geral.

Porém cada vez mais se torna perigoso a nossa transição nas ruas, não apenas por conta da pandemia que está presente nesse ano de 2020, mas também por questões de segurança, como assaltos, sequestros ou acidentes no geral, principalmente para crianças ou pessoas mais velhas.

Com base nisso, a nossa ideia trabalhada foi de um aplicativo de celular que permite o monitoramento de demais membros da família. O aplicativo permite que o usuário cadastre ele mesmo e seus familiares a sua escolha, dessa forma o mesmo poderá saber facilmente a localização de cada um deles através de um mapa extremamente intuitivo.

Após cada membro da família instalar em seu celular, o mesmo faz o seu cadastro, e assim pode adicionar outros membros (previamente cadastrados) a sua lista (família), cada uma dessas pessoas terá sua localização mostrada em um mapa, do qual se atualiza em tempo real.

2 - Tecnologias utilizadas

2.1 - Angular:

O Angular é um framework front-end com intuito de facilitar o desenvolvimento e dar maior suporte ao desenvolvedor. É um framework de código aberto baseado em typescript no qual o desenvolvedor pode organizar seu código em componentes, blocos, diretivas, metadados entre outros.

2.2 - Ionic:

A nossa aplicação será mobile ou seja por um aplicativo de celular, e devido a isso foi utilizado junto ao Angular a tecnologia do Ionic, uma ferramenta também de código aberto utilizado para desenvolvimento de aplicativos mobile, sólidos e híbridos no front-end.

2.3 - Express:

O express é um framework web de código aberto utilizando a tecnologia do NodeJS. Esse framework back-end foi feito para otimizar a construção de aplicações Web e API's sendo um dos mais populares e utilizados para servidores em NodeJS.

2.4 - MongoDB:

O MongoDB é um software de banco de dados orientado a documentos, de código aberto e multiplataforma. Classificado como banco de dados NoSQL ou banco de dados não relacional, este usa documentos semelhantes a JSON chamados de BSON diferentemente do SQL que utiliza tabelas para armazenamento dos dados vindo do backend.

Foi utilizado o Mongo, pois ele permite com que as aplicações modelem as informações de forma muito mais natural e mantém ainda um grande desempenho.

2.5 - Mongo Atlas:

MongoDB Atlas é um DBaaS (Banco de Dados como Serviço) disposto em nuvem global para aplicativos modernos. Ele permite que se implante o MongoDB totalmente gerenciado em AWS, Google Cloud e Azure com uma melhor automação e práticas comprovadas que garantem disponibilidade, escalabilidade e conformidade com os padrões de privacidade e segurança de dados mais exigentes. Além disso, garante o pleno monitoramento de dados e operações em curso no banco.

2.6 - Imnsonia Core

O Imnsonia Core no geral facilita no desenvolvimento e manutenção de APIs através de workflows e ferramentas extremamente úteis, como um simples design organizado e fácil de mexer usando “drag and drop”, importação e exportação de dados, criação de requisições e autenticações, assim como a visualização de respostas e de sua linha do tempo e funcionamento no geral.

Seu uso tornou muito mais fácil manter as variáveis e comandos do backend mais organizados e simples de entender, mapeando rotas e facilitando na gestão de requisições. A separação em pastas dos métodos também facilitou a sua utilização e entendimento.

2.7 - Visual Studio Code

Para a escrita do código tanto no back-end quanto no front-end foi utilizado o Visual Studio Code. O Visual Studio Code é um editor de código fonte da Microsoft para Windows, Linux e MacOS. Uma das ferramentas mais utilizadas atualmente para o desenvolvimento de aplicações web e mobile, principalmente no front-end. Tem como diferencial um controle Git incorporado, realce de sintaxe, terminal próprio, complementação inteligente de código e diversos plugins dos quais podem ser escolhidos pelo usuário para ajudá-lo a desenvolver em uma linguagem ou ferramenta específica.

2.8 - Github

Github é uma plataforma online capaz de armazenar códigos de diferentes aplicações, permitindo seu controle tanto individual quanto colaborativo com um time. O mesmo utiliza uma sistema de “branches” ou galhos, do qual permite que mudanças feitas na aplicação possam ser feitas e testadas na aplicação, antes de juntá-las com a branch principal, sendo assim o Git é ótimo para manter o progresso sempre seguro e simples de controlar.

Em nosso trabalho, o Git foi utilizado como um repositório online, permitindo a segurança dos arquivos assim como seu compartilhamento, principalmente na configuração síncrona entre backend e frontend.

3 - Preparando o ambiente:

Para executar a aplicação é necessário ter na máquina instalado diversos softwares e ferramentas, logo abaixo será dado um passo a passo de como efetuar as instalações e posteriormente executar o aplicativo:

1. **Instalação do NodeJS:** O NodeJS é a base de toda a aplicação, é nele que a aplicação estará rodando. Ele pode ser encontrado no seguinte link:

<https://nodejs.org/en/download/>

The screenshot shows the Node.js Downloads page. At the top, it says 'Downloads' and 'Latest LTS Version: 14.15.0 (includes npm 6.14.8)'. Below this, it says 'Download the Node.js source code or a pre-built installer for your platform, and start developing today.' There are two main tabs: 'LTS Recommended For Most Users' and 'Current Latest Features'. Under the 'LTS' tab, there are three options: 'Windows Installer' (node-v14.15.0-x64.msi), 'macOS Installer' (node-v14.15.0.pkg), and 'Source Code' (node-v14.15.0.tar.gz). Under the 'Current' tab, there are three options: 'Windows Installer' (node-v15.0.0-x64.msi), 'macOS Installer' (node-v15.0.0.pkg), and 'Source Code' (node-v15.0.0.tar.gz). Below these, there is a table of download links for various platforms and architectures.

Platform	Architecture	Download Link
Windows	32-bit	node-v14.15.0-x86.msi
	64-bit	node-v14.15.0-x64.msi
macOS	32-bit	node-v14.15.0.pkg
	64-bit	node-v14.15.0.pkg
Linux	64-bit	node-v14.15.0.tar.gz
	64-bit	node-v14.15.0.tar.gz
Linux	ARMv7	node-v14.15.0-armv7.tar.gz
	ARMv8	node-v14.15.0-armv8.tar.gz

Escolha a opção que seja compatível com o seu sistema operacional. Efetue o download do software e instale-o na sua máquina.

2. **Instalação do Angular CLI:** Com o NodeJS já instalado temos a disposição o comando npm, abra um terminal em modo administrador e execute o comando **npm install -g @angular/cli** conforme a imagem em anexo:

```
Administrador: Prompt de Comando
Microsoft Windows [versão 10.0.19041.572]
(c) 2020 Microsoft Corporation. Todos os direitos reservados.
C:\Windows\system32>npm install -g @angular/cli
```

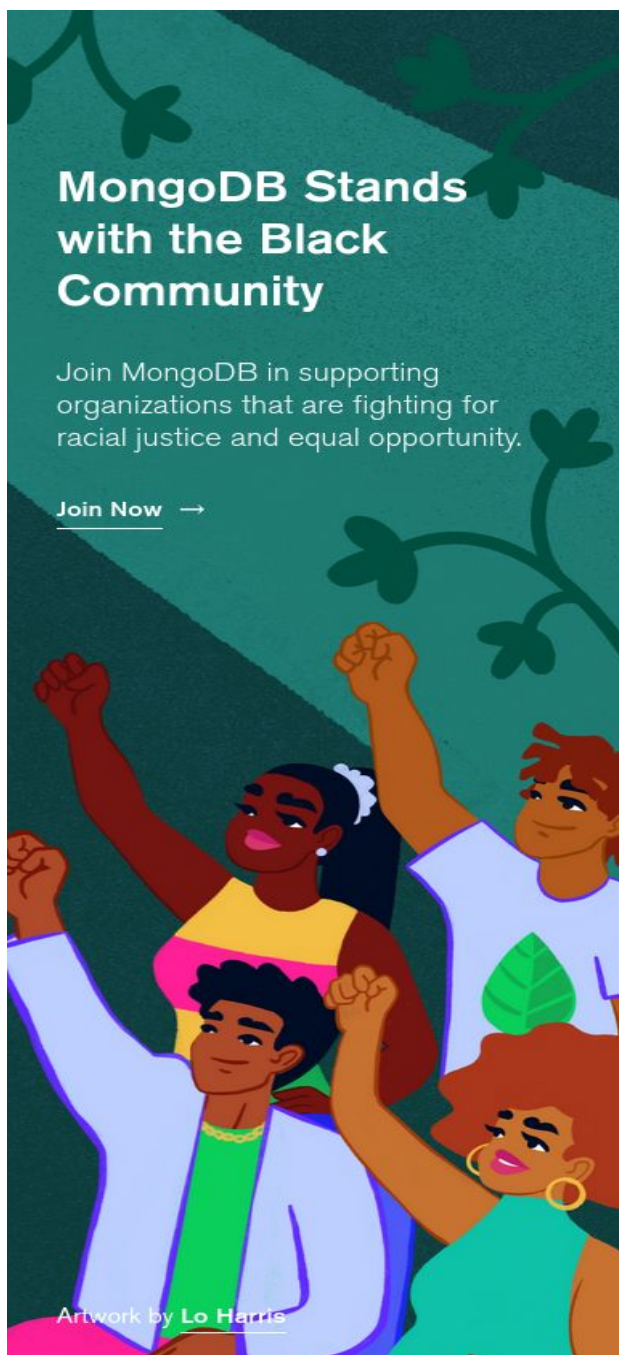
Através desse comando será instalado todas as dependências do Angular em modo global, e já estará disponível para ser utilizado na construção de uma aplicação front-end Angular.

3. **Instalação do Ionic:** Logo após a instalação do NodeJs e Angular, realizaremos a instalação do Ionic. Com o terminal em modo administrador execute o comando **npm install -g @ionic/cli** conforme imagem abaixo:


```
Administrador: Prompt de Comando
Microsoft Windows [versão 10.0.19041.572]
(c) 2020 Microsoft Corporation. Todos os direitos reservados.
C:\Windows\system32>npm install -g @ionic/cli
```

As dependências do Ionic serão instaladas de forma global e já estarão disponíveis para utilização.

4. **Criação de conta no Mongo (opcional** - o projeto já vem com uma conta no mongo vinculada, basta pular para o Tópico **3 - Executando a aplicação**):
- Não iremos trabalhar com um banco de dados disposto localmente, ao invés disso iremos utilizar uma solução mais moderna, gerenciada em nuvem. Para isso, iremos primeiro criar uma conta, acesse o link [Create Account | MongoDB](#) e preencha os campos, como segue exemplificado abaixo e clique em “Sign Up”:



Create your account

 Sign up with Google

or

Email Address

roko@ecallen.com ✓

We recommend using your work email

First Name

John ✓

Last Name

Doe ✓

Password

..... ✓

Phone Number

35999887766 ✓

Company Name

John ✓

Job Function

Student ▼

Country

Brazil ▼

☒ I accept the [Privacy Policy](#) and the [Terms of Service](#)

Sign up

Have an account? [Log in now](#)

Shared Clusters

For teams learning MongoDB or developing small applications.

- ✓ Highly available auto-healing cluster
- ✓ End-to-end encryption
- ✓ Role-based access control

Create a cluster

Starting at
FREE

Após isso, aparecerão três opções de cluster de servidores, escolha o gratuito no formato “Shared Clusters”, e clique em “Create a cluster”. (imagem ao lado)

Na sequência, você irá marcar as opções como Cloud Provider, o “Google Cloud” e na Region, “Sao Paulo (southamerica-east1)”. (imagem abaixo)

Welcome to MongoDB Atlas! We've recommended some of our most popular options, but feel free to customize your cluster to your needs. For more information, check our [documentation](#).

Cloud Provider & Region

GCP, Sao Paulo (southamerica-east1) ▼



★ Recommended region ⓘ

NORTH AMERICA / SOUTH AMERICA

🇺🇸 Iowa (us-central1) ★

🇧🇷 Sao Paulo (southamerica-east1) ★

ASIA PACIFIC

🇮🇳 Mumbai (asia-south1) ★

🇯🇵 Tokyo (asia-northeast1) ★

🇸🇬 Singapore (asia-southeast1) ★

🇹🇼 Taiwan (asia-east1) ★

EUROPE / MIDDLE EAST / AFRICA

🇧🇪 Belgium (europe-west1) ★

Feito isso, clique em “Create Cluster” e aguarde, demora de 1 a 3 minutos para ele ser criado.

No canto esquerdo da tela, clique no botão “Database Access”, e depois em “Add New Database User”.

Abrirá um pop-up, e nele iremos criar um usuário para nosso banco de dados.

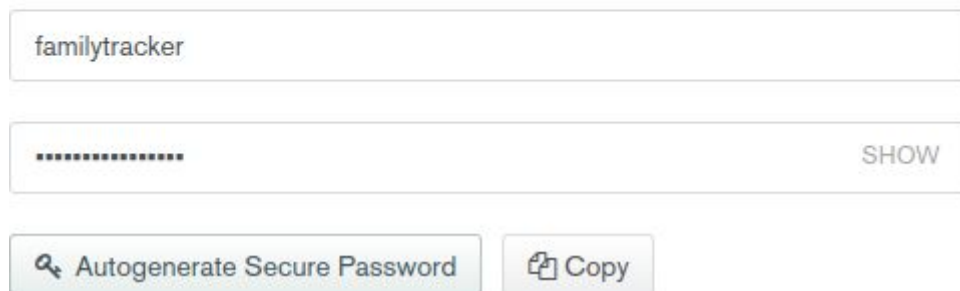
- Nome: familytracker
- Senha: definida a seu gosto

A senha, assim como o usuário, são extremamente importantes, certifique-se de guardá-los em algum local, pois irá precisar deles, quando for se conectar remotamente ao banco.

Após inserir a senha, ignore as demais opções e clique em “Add User”.

MongoDB uses **SCRAM** as its default authentication method.

Password Authentication

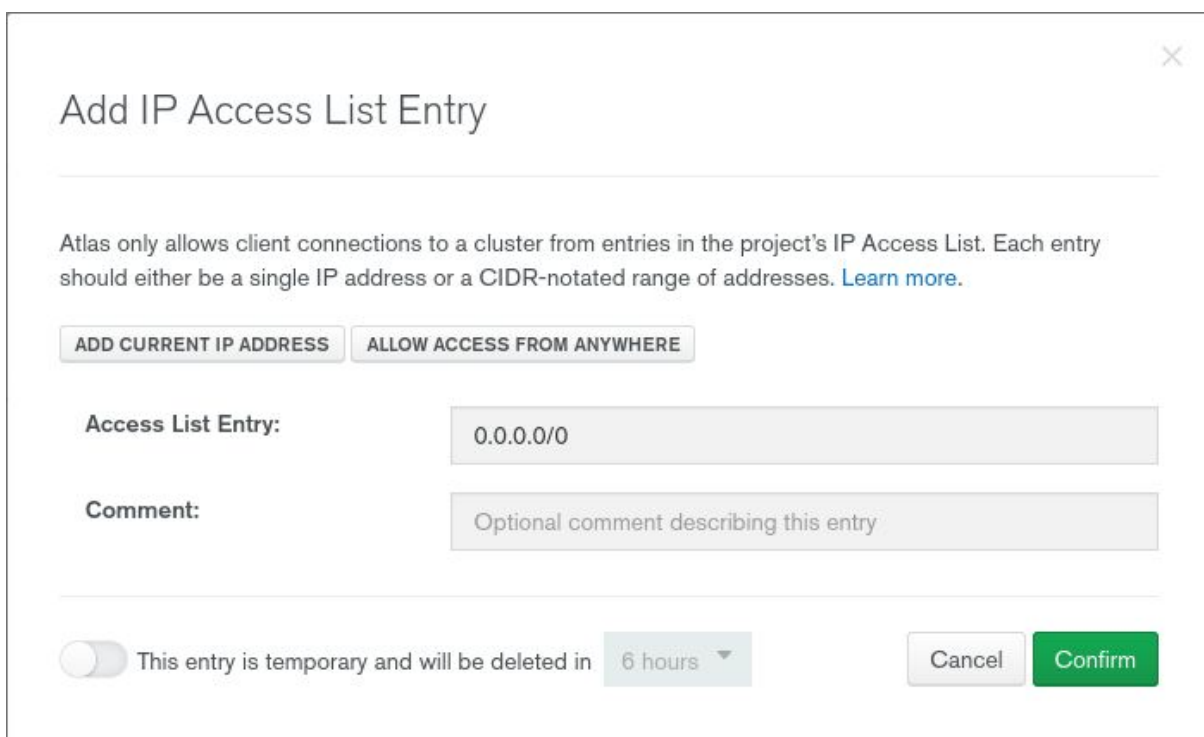


familytracker

..... SHOW

Autogenerate Secure Password Copy

No canto esquerdo da tela, clique na opção “Network Access” e depois em “Add IP Address”. No pop-up que abrir clique no botão “Allow access from anywhere” e depois em “Confirm”. Aguarde cerca de um minuto para que seja realizado o deploy automático das alterações.



Add IP Access List Entry

Atlas only allows client connections to a cluster from entries in the project's IP Access List. Each entry should either be a single IP address or a CIDR-notated range of addresses. [Learn more.](#)

ADD CURRENT IP ADDRESS ALLOW ACCESS FROM ANYWHERE

Access List Entry: 0.0.0.0/0

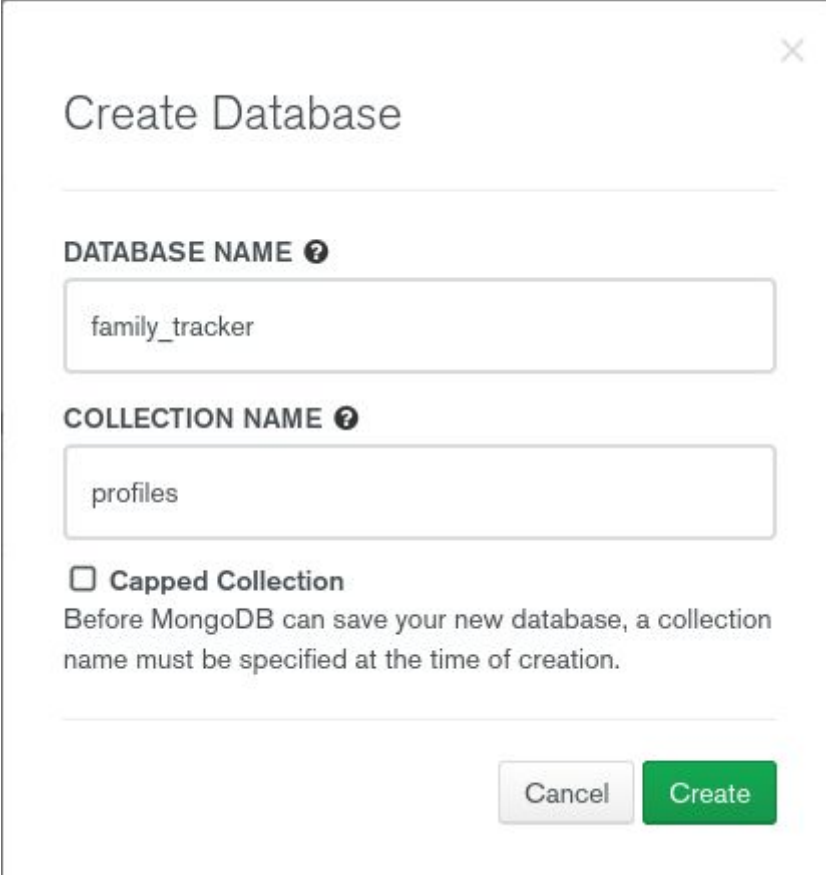
Comment: Optional comment describing this entry

☐ This entry is temporary and will be deleted in 6 hours

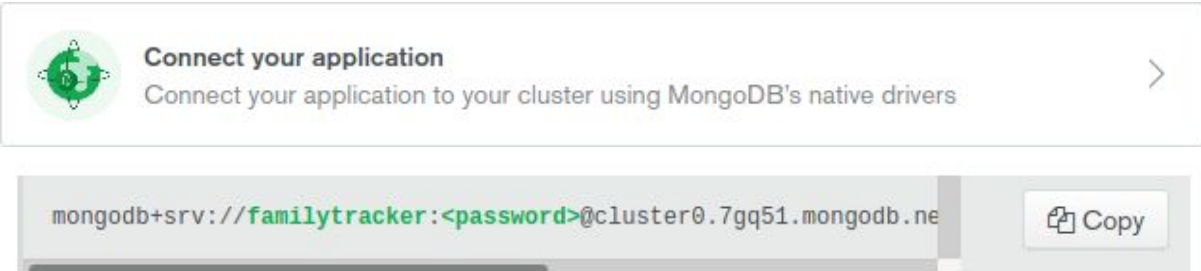
Cancel Confirm

Agora no canto esquerdo, clique em “Clusters” para retornar à tela principal, lá, clique no botão **COLLECTIONS**. E em seguida, em “Add My Own Data”.

No pop-up que abrirá, iremos criar o nosso banco. Portanto, para evitar incompatibilidade, insira exatamente as mesmas informações que estão na imagem abaixo, e clique em “Create”.

A screenshot of a 'Create Database' dialog box. It has a title bar with a close button (X). The main title is 'Create Database'. Below it, there are two input fields. The first is labeled 'DATABASE NAME ?' and contains the text 'family_tracker'. The second is labeled 'COLLECTION NAME ?' and contains the text 'profiles'. Below these fields, there is a checkbox labeled 'Capped Collection' which is currently unchecked. Underneath the checkbox, there is a note: 'Before MongoDB can save your new database, a collection name must be specified at the time of creation.' At the bottom right of the dialog, there are two buttons: 'Cancel' and 'Create'.

Pronto, o banco está criado, agora clique em “Clusters”, do lado esquerdo. E após voltar para a tela principal, clique em **CONNECT**. Escolha a seguinte opção “Connect your application”, e abrirá uma tela com o conteúdo da segunda imagem abaixo:

A screenshot of a 'Connect your application' dialog box. It features a green circular icon with a MongoDB logo on the left. The text reads 'Connect your application' followed by 'Connect your application to your cluster using MongoDB's native drivers'. There is a right-pointing arrow on the right side. Below this, there is a text box containing the MongoDB connection string: 'mongodb+srv://familytracker:<password>@cluster0.7gq51.mongodb.net'. To the right of the text box is a 'Copy' button with a clipboard icon.

Clique em “Copy” e após isso, substitua **<password>** pela senha que você havia definido para o usuário do banco de dados e **<dbname>** pelo nome do seu banco de dados, que é “family_tracker”.

Conforme a imagem abaixo, cole essa string, como o valor da chave “url”, no arquivo “database.config.js” em “familyTrackerBackend/config”:


```
You, 9 days ago | 1 author (You)
1 module.exports = {
2   url: 'mongodb+srv://familytracker:cstDRCSLpkReSUUV@cluster.yh5ln.mongodb.net/family_tracker?retryWrites=true&w=majority'
3 }
4
```

4 - Executando a aplicação:

4.1 - Backend

Acesse pelo terminal, o diretório raiz do projeto “familyTrackerBackend” e execute o seguinte comando, para instalar as dependências do projeto e para iniciar o servidor/API:

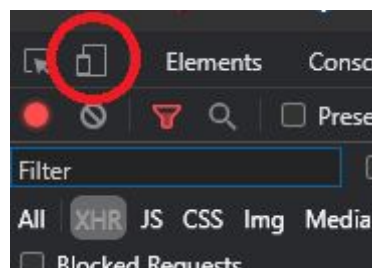
npm install && npm start

Pronto, a API está conectada a um banco remoto e aguardando as requisições.

4.2 Frontend

Acesse pelo terminal, o diretório raiz do projeto “familyTracker” e execute o seguinte comando, para instalar as dependências do projeto e para iniciar o front-end: **npm install && npm start**

Pronto, o front-end poderá ser acessado pelo navegador em <http://localhost:4200>, como a aplicação é voltada a plataforma mobile recomenda-se que o navegador esteja em modo mobile, você pode realizar este processo clicando sobre a página com o botão direito em “inspecionar” e clicar no ícone do mobile mostrado na imagem abaixo:



5 - Resultados Esperados

É esperado que o aplicativo atenda a todos os recursos prometidos de maneira eficaz. Que a localização de familiares seja em tempo real e o mais fiel possível do mapa com a realidade, tendo como objetivo proporcionar ao usuário uma segurança de saber onde seus entes queridos se encontram caso alguém se perca, principalmente para os mais vulneráveis como crianças e idosos.

Não apenas isso mas também propor uma experiência interessante de poder encontrar pessoalmente seus parentes mais próximos, além de caso o dispositivo móvel seja furtado também será possível saber a localização exata em tempo real onde o mesmo se encontra.

6 - Procedimento de Desenvolvimento

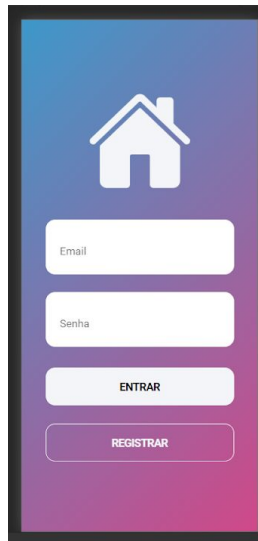
O grupo se comunicou tanto por mensagens rápidas quanto em reuniões online pelo uso do Google Meets, foi feito o brainstorming sobre qual ideia seria trabalhada, assim como as melhores maneiras de abranger essa mesma ideia de forma rápida e coesa.

Para agilização do processo, o grupo foi dividido em trabalho nas áreas de backend e frontend, trabalhando ambas em paralelo, porém ainda mantendo contato para garantir um funcionamento satisfatório.

Para a eliminação de bugs e aquisição de feedback em geral, os membros do grupo simularam situações reais sobre como o aplicativo funcionaria, utilizando seus aparelhos Androids, simulando uma família conectada.

7 - Instruções para uso

O objetivo do aplicativo é ser simples e intuitivo. Ao abrir o app o usuário deverá registrar-se, clicando no botão Registrar conforme imagem abaixo:



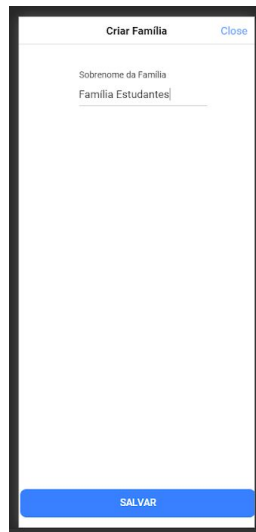
Então será redirecionado para a tela de registro onde deve inserir seu e-mail, senha e idade, conforme abaixo:

The screenshot shows a mobile app interface for registration. At the top, there is a blue header bar with a back arrow and the word "Back" on the left, and the title "Registrar" on the right. Below the header, there are four input fields: "Email" with the value "gabriel.mouallem@gmail.com", "Senha (pelo menos 8 caracteres)" with a masked password "*****", "Nome" with the value "Gabriel Mouallem", and "Idade" with the value "22". Below these fields is a blue button labeled "REGISTRAR".

Após registrado basta logar com o usuário criado. Logo cairá na tela onde o usuário terá a opção de cadastrar uma nova família ou participar de uma já previamente cadastrada:

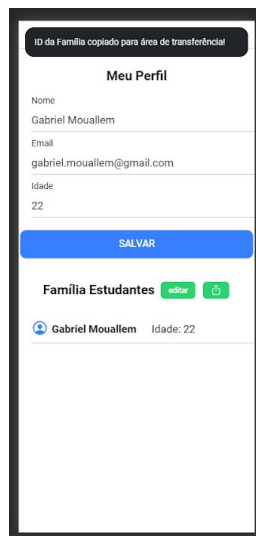
The screenshot shows a mobile app interface for login. At the top, there is a dark grey header bar with the text "Bem vindo ao Family Tracker". Below the header, there is a section titled "Meu Perfil" which contains four input fields: "Nome" with the value "Gabriel Mouallem", "Email" with the value "gabriel.mouallem@gmail.com", "Idade" with the value "22", and a blue button labeled "SALVAR". Below this section, there is a section titled "Minha Família" which contains two red buttons: "CADASTRAR MINHA FAMÍLIA" and "PARTICIPAR".

Caso queira criar uma nova família, o mesmo deve clicar em Cadastrar minha família. Após isso exibirá uma tela simples onde o usuário deve inserir o nome da sua família que deseja cadastrar. No momento do cadastro será gerado automaticamente um ID da família referente ao cadastrado:



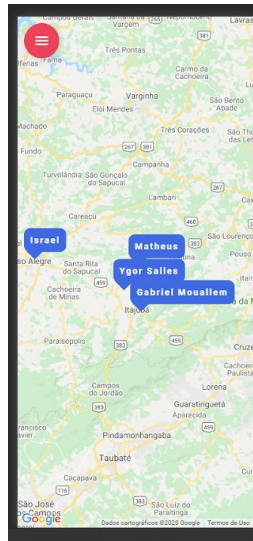
Este mesmo cadastro poderá ser editado posteriormente assim como o cadastro de cada membro do perfil. Já salvo a família, pode fechar a tela em CLOSE, e na tela de perfil do usuário já exibirá a família no qual ele é membro.

Como não foi adicionado ninguém, o usuário poderá clicar no botão com ícone ao lado do botão “editar” assim será copiado o ID da família para área de transferência do dispositivo.



O usuário poderá então enviar o ID para algum membro da família via Whatsapp ou qualquer outro meio de comunicação que ele desejar.

Considerando que o parente desse usuário recebeu o ID e esse parente já tenha um perfil no aplicativo, o parente pode então, na página de perfil, colar o ID no campo “ID da Família” e clicar no botão “Participar”. Logo ambos já farão parte da mesma família podendo localizar uns aos outros no mapa em tempo real:



Como pode ser representado no exemplo acima. Foi cadastrado quatro membros de uma família e todos podem ter acesso a localização de cada membro em tempo real.