

SINOP-MT

2021

NOME DO ALUNO

Diego Parpinelli de Paula Oliveira

TEMA

Controle de portaria e presença para alunos

NOME DO ALUNO

SINOP-MT

2021

PROPOSTA DE UM SISTEMA DE INFORMAÇÃO COM ES DE **TEMA**

Controle de portaria e presença para alunos

Trabalho de Conclusão de Curso apresentado ao Centro Universitário Unifasipe, da matéria de Projeto Interdisciplinar I, lecionada pelo Prof. Esp. Willian Aparecido Hübner do Nascimento, como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

NOME DO ALUNO

Diego Parpinelli de Paula Olivera

**SUMÁRIO**

## [1. INTRODUÇÃO 4](#_Toc77270565)

## [2. OBJETIVO 4](#_Toc77270566)

## [1.1. OBJETIVO GERAL 4](#_Toc77270567)

## [1.2. OBJETIVO ESPECÍFICO 4](#_Toc77270568)

## [3. PROBLEMA 4](#_Toc77270569)

## [4. SOLUÇÕES APRESENTADAS 4](#_Toc77270570)

## [5. ETAPAS E OBJETIVOS 4](#_Toc77270571)

## [6. METODOLOGIA 4](#_Toc77270572)

## [7. CRONOGRAMA 4](#_Toc77270573)

# INTRODUÇÃO

Devido à proximidade do Concipe, sugerimos um projeto para ajudar a sanar alguma dificuldade que a equipe de organização tenha durante o evento, utilizando da análise de requisitos e reuniões com colaboradores mais experientes chegamos a conclusão que, a lista de presença manuais são um grande gargalo para toda a equipe durante o evento e após ele, pois os dados devem ser consultados de forma manual se houver alguma necessidade no futuro, tendo isso em mente desenvolveremos uma solução para esse prolema utilizando da tecnologia.

# OBJETIVO

* 1. OBJETIVO GERAL

O projeto tem como objetivo, controlar a entrada, saída e a frequência de alunos em eventos da faculdade.

* 1. OBJETIVO ESPECÍFICO

Facilitar a entrada de eventos da faculdade

Garantir que apenas pessoas inscritas no evento tenham acesso a ele.

Garantir que alunos tenham sua presença registrada no evento

# PROBLEMA

O controle de frequência é feito através de listas manuais, o que demanda muito tempo para registrar um aluno gerando longas filas e imprecisão de dados.

# SOLUÇÕES APRESENTADAS

Uma API foi desenvolvida com Node.JS para realizar a administração de eventos, inscrição de participantes, geração de qrcodes e leitura dos mesmos, essa API foi utilizada para a criação de uma aplicação web com ReactJs.

Dentro da aplicação o usuario pode criar um evento, concedendo a ele um nome e um tipo, esse deve ser escolhido entre duas opções:

Ticket: eventos do tipo ticket devem ser utilizados quando se trata de um evento único com intuito de controlar somente a entrada e saída de pessoas autorizadas.

Pass: eventos do tipo pass devem ser utilizados quando se trata de um evento composto de outros eventos, com intuito de controlar a entrada de pessoas autorizadas aos eventos menores e se necessário ao evento principal.

Após a criação do evento, podem ser feitas as inscrições de participantes, sendo necessário apenas o nome e o CPF do participante, e de forma opcional o e-mail para o recebimento do qrcode utilizado para liberar o acesso ao evento na portaria, caso não for informado o e-mail, fica a cargo da instituição definir como serão entregues os qrcodes de acesso.

# ETAPAS E OBJETIVOS

Começaremos projetando o a forma que os dados serão armazenados no bando de dados, apó isso criaremos funções para garantir que as regras de negócio sejam aplicadas.

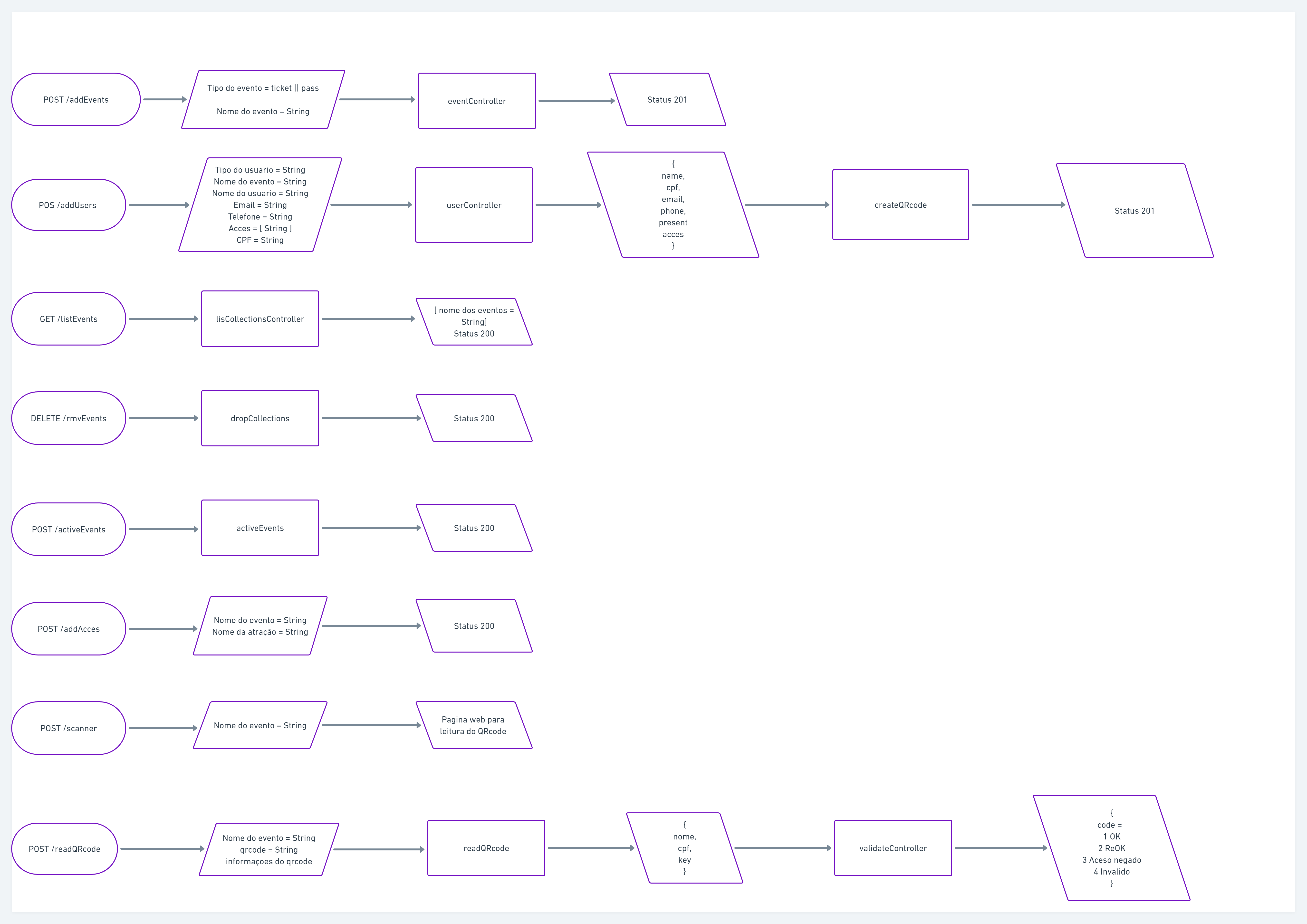
Todo qrcode deve ser único para cada evento.

O qrcode deve conter apenas dados de referência e não dados do usuario.

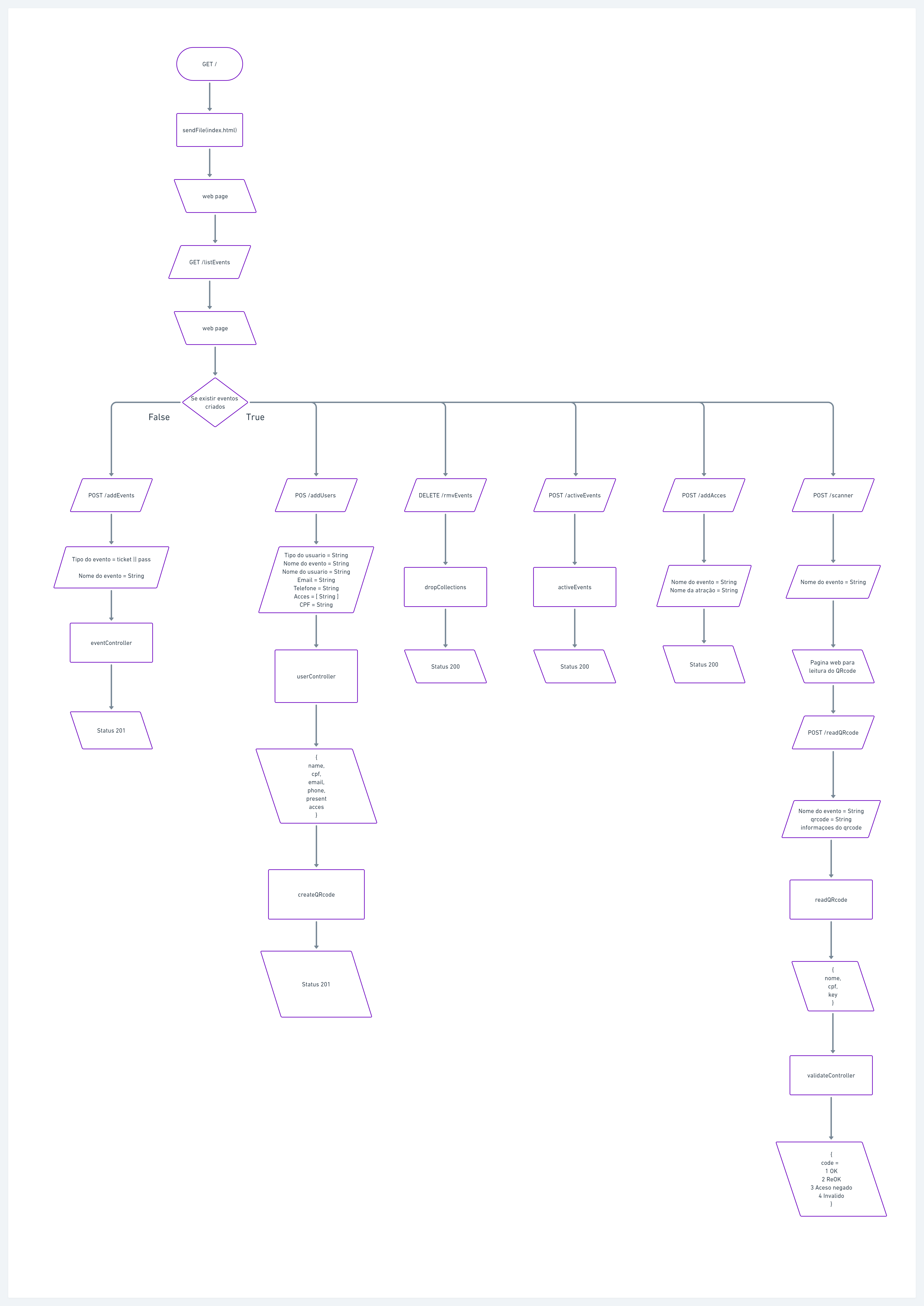
Toda vez que um usuário valido for verificado, deve ser informado se é a primeira vez que está sendo verificado ou se já foi verificado antes

O sistema deve ser portável e deve funcionar em ambiente local.

|  |  |  |
| --- | --- | --- |
| **Etapa** | **Período** | **Descrição** |
| **1** | 01/09/2022 | Coleta de informações com o cliente. |
| **2** | 20/09/2022 | Protótipo inicial desenvolvido. |
| **3** | 01/02/2021 | Teste inicial e aprovação do projeto pelo cliente. |



Rotas disponíveis na API



Fluxograma geral

# METODOLOGIA

**6.1 Node.js**

Como um ambiente de execução JavaScript assíncrono orientado a eventos, o Node.js é projetado para desenvolvimento de aplicações escaláveis de rede.

O node.js foi utilizado para o desenvolvimento da API devido a sua grade variedade de módulos para essa função.

Módulos utilizados:

Express, utilizado para realizar o roteamento da API.

Fille System, utilizado para manipulação de arquivos no sistema operacional.

Mongodb: drive de conexão ao banco de dados noSQL Mongodb

Cryptojs: utilizado para criptografar e descriptografar informações contidas no qrcode.

Qrcode, para geração de qrcodes.

Instascan: para leitura de qrcodes atraves do navegador

Nodemailer: envio de e-mails

**6.2 React**

React é uma biblioteca JavaScript para construção de interfaces de usuário.

React compõe toda parte de front-end da aplicação web, ela foi escolhida devido sua alta performance e simplicidade de utilização, através dela é executado todas requisições HTTP para API, também é responsável por gerenciar formulários necessários para API e a renderização de novos componentes na tela.

**6.3 Mongodb**

Mongodb oferece um modelo de dados de documento flexível junto ao suporte para consultas ad-hoc, indexação secundária e agregações em tempo real para fornecer maneiras poderosas de acessar e analisar seus dados.

# CONCLUSÃO

Após finalizado o desenvolvimento e implementação do projeto, demonstre aqui os resultados obtidos, mostre telas do software, tabelas de funções, gráficos demonstrativos.

# REFERENCIAL TEÓRICO

Mongodb: [MongoDB Community Download | MongoDB](https://www.mongodb.com/try/download/community2)

Reactjs: [React – Uma biblioteca JavaScript para criar interfaces de usuário (reactjs.org)](https://pt-br.reactjs.org/)

Node.js: [Sobre | Node.js (nodejs.org)](https://nodejs.org/pt-br/about/)