



Estrutura Analítica do Projeto

Sistema de Gerenciamento Online

João Augusto Rocha Carloto, Natã Batista Fernandes, Rodolfo  
Dos Santos Aura

Jaguariúna, 2023

# PROJETO FINAL

## RESUMO

O PlayOnMatch é um projeto que visa criar uma plataforma inovadora para o gerenciamento de eventos esportivos. Com o objetivo de promover a prática esportiva,

a plataforma oferece funcionalidades que permitem aos usuários criar e gerenciar partidas de diferentes modalidades esportivas e tem como objetivo proporcionar uma experiência positiva,

inclusiva e emocionante para os amantes de esportes, permitindo que eles se conectem, se divirtam e desfrutem de competições saudáveis. Com uma interface amigável e recursos inovadores,

o projeto busca facilitar o gerenciamento de eventos esportivos.

## JUSTIFICATIVA

O projeto PlayOnMatch é desenvolvido para atender à necessidade crescente de uma plataforma que facilite o gerenciamento e a organização de eventos esportivos,

promovendo a participação ativa dos usuários por meio da gamificação, com foco em conectar entusiastas de esportes e oferecer uma experiência envolvente.

Com o estilo de vida moderno e a falta de tempo, muitas pessoas têm dificuldades em encontrar e participar de eventos esportivos locais.

O PlayOnMatch visa resolver esse problema, fornecendo uma plataforma centralizada onde os usuários podem descobrir facilmente as partidas que

estão acontecendo em sua região. Isso permite que os entusiastas de esportes encontrem oportunidades para praticar suas modalidades favoritas,

conhecer novas pessoas e se engajar ativamente na comunidade esportiva local.

# **OBJETIVOS**

## **GERAL**

Desenvolver uma plataforma que visa facilitar o gerenciamento de eventos esportivos e facilitar as pessoas a encontrar eventos nas regiões que se encontram, promovendo a participação ativa dos usuários.

## **ESPECÍFICO**

1. Criar um sistema de cadastro de usuários, permitindo que eles possam criar perfis e acessar a plataforma.
2. Desenvolver uma interface intuitiva e amigável, facilitando a navegação e a utilização da plataforma pelos usuários.
3. Implementar um sistema de busca de eventos esportivos, permitindo que os usuários encontrem facilmente partidas disponíveis em sua região.
4. Criar um sistema de notificações, informando os usuários sobre convites para novos eventos e solicitações de amizade.
5. Oferecer suporte técnico e realizar atualizações regulares na plataforma para garantir seu bom funcionamento e segurança dos dados dos usuários.

# INTRODUÇÃO

A prática de esportes é uma atividade que proporciona diversos benefícios para a saúde física e mental. No entanto, muitas vezes é difícil encontrar e participar de eventos esportivos locais, o que pode limitar a oportunidade de engajar-se ativamente nesse universo. Pensando nisso, o projeto PlayOnMatch surge como uma solução inovadora, que visa facilitar o gerenciamento e a organização de eventos esportivos, ao mesmo tempo em que promove a participação ativa dos usuários por meio da gamificação. Esta plataforma busca conectar entusiastas de esportes, proporcionando um ambiente virtual onde é possível encontrar, criar e participar de partidas, ampliando as oportunidades de prática esportiva, o PlayOnMatch permite aos usuários descobrir eventos esportivos em sua região

## TAP (TERMO DE ABERTURA DO PROJETO)

**Título do Projeto ou Resumo do Objetivo:**

Desenvolvimento de uma solução para encontros em eventos esportivos.

## Patrocinadores:

Nossos professores Reenye Alexandre de Lima e Wellington Fabio de Oliveira Martins que também nos ajudarão no desenvolvimento do projeto.

DESENVOLVEDORES RH (Recursos Humanos)	Natã Batista Fernandes
	João Augusto Rocha Carloto
Gerente de Projeto	Natã Batista Fernandes
Patrocinador	Reenye Alexandre de Lima e Wellington Fabio de Oliveira Martins
Cliente	Escola Senai Jaguariúna
Prazo ou data de entrega do Projeto	13/06/2023

## CRONOGRAMA

[illegible]

# **DOCUMENTAÇÃO DE DEFINIÇÃO DO ESCOPO (EAP)**

## **CONVENÇÕES, TERMOS E ABREVIÇÕES**

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

### **IDENTIFICAÇÃO DOS REQUISITOS**

Por convenção, a referência a requisitos é feita através do nome da subseção onde eles estão descritos seguidos do identificador do requisito, de acordo com a especificação a seguir: [nome da subseção, identificador do requisito] Por exemplo, o requisito funcional [Incluir Usuário.RF016] deve estar descrito em uma subseção chamada “Incluir Usuário”, em um bloco identificado pelo número [RF016]. Já o requisito não-funcional [Confiabilidade.NF008] deve estar descrito na seção de requisitos não-funcionais de Confiabilidade, em um bloco identificado por [NF008]. Os requisitos devem ser identificados com um identificador único. A numeração inicia com o identificador [RF001] ou [NF001] e prossegue sendo incrementada à medida que forem surgindo novos requisitos. Cada requisito deve fazer referência a uma regra de negócio [RN001].

### **PROPRIEDADES DOS REQUISITOS**

Para estabelecer a prioridade dos requisitos, foram adotadas as denominações “essencial”, “importante” e “desejável”. Essencial é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente. Importante é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim. Desejável é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

## **DESCRIÇÃO GERAL DO PROJETO:**

PlayOnMatch é um gerenciador de eventos esportivos que oferece a possibilidade de criar partidas de diferentes modalidades esportivas, bem como campeonatos.

A plataforma ainda conta com um sistema de classificação individual de jogadores e equipes, apresentados em rankings específicos do PlayOnMatch, permitindo aos usuários acompanharem seu desempenho e evolução que serão atualizados automaticamente com base no desempenho de cada jogador e equipe.

## **REGRAS DE NEGÓCIO**

Estas regras de negócio foram elencadas utilizando a técnica de levantamento orientado a ponto de vista e etnografia, observando e estudando outros aplicativos de bancos virtuais, observando suas funcionalidades básicas e objetivando aplica-las.

### **[RN01] Autenticação**

O sistema devera gerenciar o acesso dos usuários, para que eles tenham acesso a outros times, campeonatos e jogadores.

### **[RN02] Gerenciamento de usuários**

Será necessário a implementação de funcionalidades que permitam ao usuário inserir, atualizar, excluir e listar.



### **[RN03] Gerenciamento de eventos**

Será necessário a implementação de funcionalidades que permitam ao usuário criar, atualizar, excluir e listar.

### **[RN04] Sistema de notificações**

O PlayOnMatch deve garantir que essas notificações sejam relevantes e úteis, evitando o spam ou excesso de notificações que possam incomodar os usuários.

### **[RN05] Histórico de jogos de cada jogador**

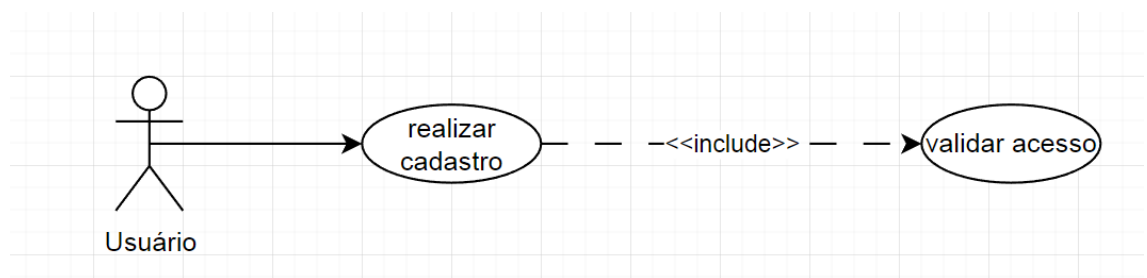
O histórico de jogos de cada jogador deve ser preciso e confiável, evitando erros ou informações incorretas, as informações devem ser acessíveis apenas pelos usuários autorizados e que sejam protegidas por medidas de segurança adequadas.

# REQUISITOS FUNCIONAIS

## [RF001] Registro de usuários

Prioridade: (x) Essencial ( ) Importante ( ) Desejável

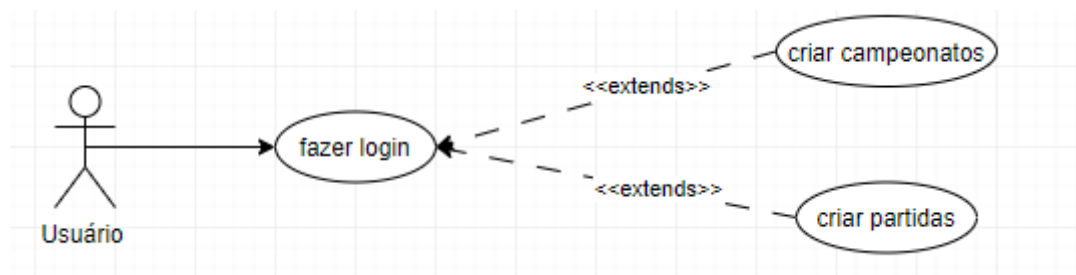
O sistema deve permitir que novos usuários se registrem fornecendo informações básicas, como nome completo, endereço de e-mail e senha.



## [RF002] Criação de Partidas

Prioridade: (x) Essencial ( ) Importante ( ) Desejável

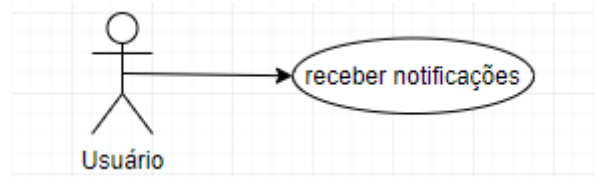
O sistema deve permitir que usuários criem partidas para diferentes esportes e determinem a data, hora e local de cada partida.



## [RF006] Sistema de notificação

Prioridade: ( ) Essencial (x) Importante ( ) Desejável

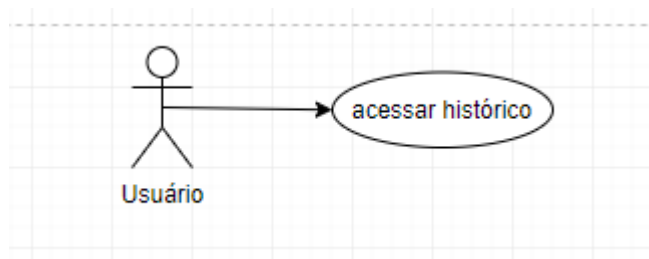
O sistema deve enviar notificações para os usuários sobre convites para participar de novos eventos esportivos, entre outros



## [RF008] Histórico de jogador

Prioridade: ( ) Essencial (x) Importante ( ) Desejável

O sistema deve mostrar um histórico de jogos para cada jogador, incluindo resultados.



# REQUISITOS NÃO FUNCIONAIS

## [NF001] LINGUAGENS DE PROGRAMAÇÃO

As linguagens de programação utilizadas no desenvolvimento deste projeto são: Back-End Javascript Framework (Node.js), Front-End HTML, CSS, JavaScript, Mobile Javascript Framework (React Native).

## [NF002] SEGURANÇA

Criptografia de senha no login/cadastro com hash code, referindo-se ao login [RF001] Os dados codificados serão armazenados no banco.

## [NF002] SERVIDORES

Os servidores utilizados neste projeto são Microsoft Azure.

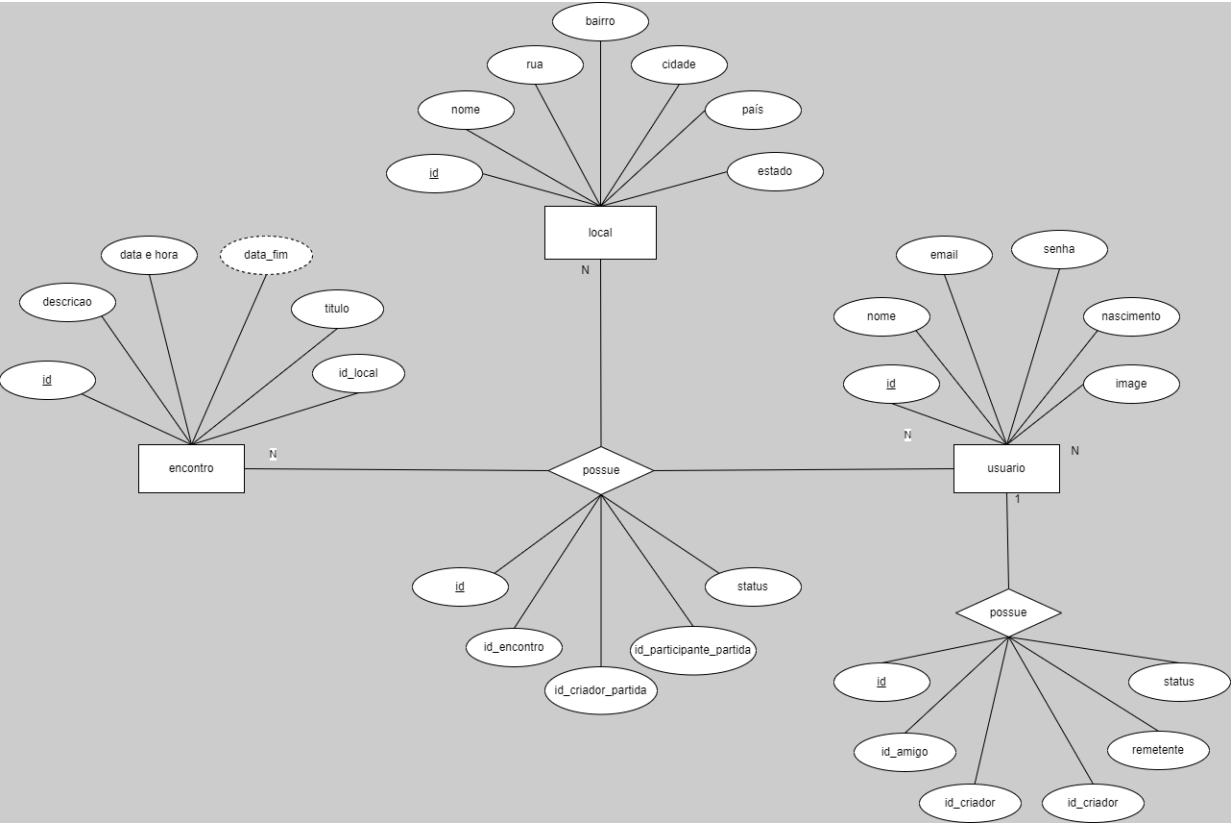
# ORÇAMENTO

SERVIÇOS	VALOR	HORA	TOTAL
DOCUMENTAÇÃO	R\$ 15,00	12	R\$ 180,00
BANCO	R\$ 20,00	8	R\$ 160,00
BACKEND	R\$ 25,00	100	R\$ 2.500,00
FRONTEND	R\$ 25,00	80	R\$ 2.000,00
MOBILE	R\$ 20,00	50	R\$ 1.000,00
CRIPTOGRAFIA	R\$ 10,00	4	R\$ 40,00
TESTE	R\$ 10,00	12	R\$ 120,00
IMPLANTAÇÃO	R\$ 5,00	4	R\$ 20,00
TOTAL		270	R\$ 12.040,00
VALOR POR PESSOA:			R\$ 6.020,00

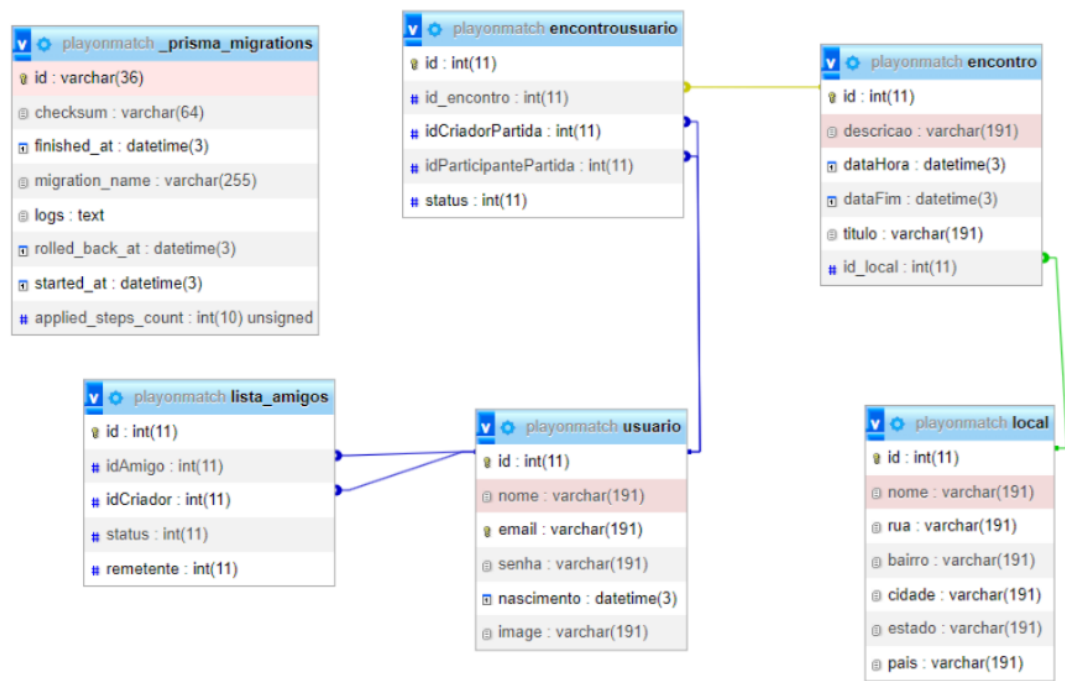
# Documentação da estrutura da solução

## Banco de Dados

### DER Conceitual



# DER Lógico



## Back-End

```
model Usuario {
  id          Int          @id @default(autoincrement())
  nome        String
  email       String       @unique
  senha       String
  nascimento  DateTime
  image       String?
  amigo       Lista_amigos[] @relation("amigo")
  criadorListaAmigo Lista_amigos[] @relation("criador")
  criadorPartida EncontroUsuario[] @relation("criadorPartida")
  participante EncontroUsuario[] @relation("participantePartida")
}

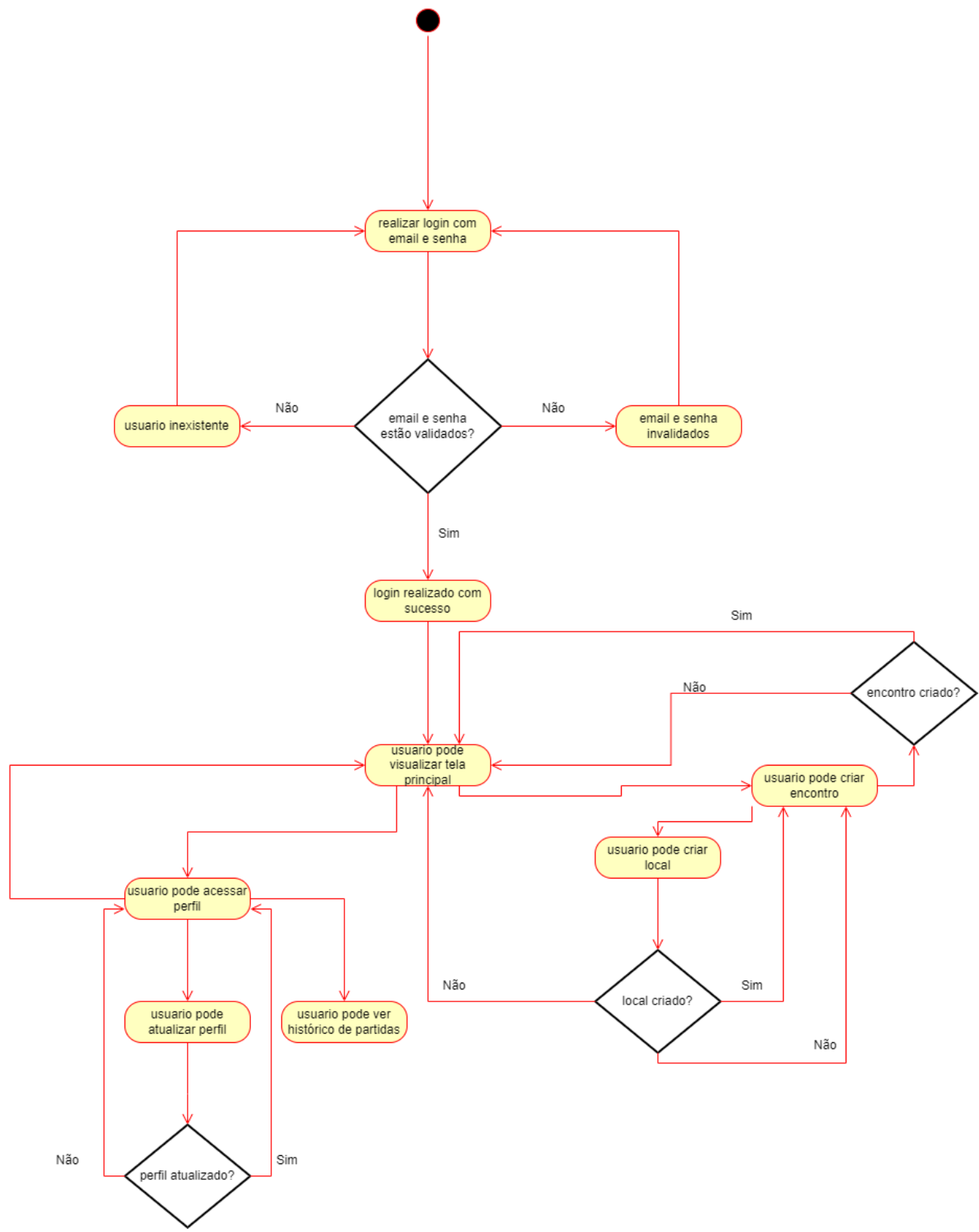
model Lista_amigos {
  id      Int      @id @default(autoincrement())
  idAmigo Int?
  idCriador Int
  status   Int      @default(0)
  remetente Int
  amigo    Usuario? @relation("amigo", fields: [idAmigo], references: [id])
  criador  Usuario  @relation("criador", fields: [idCriador], references: [id])
}

model Encontro {
  id          Int          @id @default(autoincrement())
  descricao   String
  dataHora    DateTime
  dataFim     DateTime?
  titulo      String
  id_local    Int
  local       Local        @relation(fields: [id_local], references: [id])
  EncontroUsuario EncontroUsuario[]
}

model Local {
  id      Int      @id @default(autoincrement())
  nome    String
  rua     String
  bairro  String
  cidade  String
  estado  String
  pais    String
  Encontro Encontro[]
}

model EncontroUsuario {
  id          Int      @id @default(autoincrement())
  id_encontro Int
  idCriadorPartida Int
  idParticipantePartida Int?
  status       Int      @default(0)
  idCriador    Usuario  @relation("criadorPartida", fields: [idCriadorPartida], references: [id])
  idParticipante Usuario? @relation("participantePartida", fields: [idParticipantePartida], references: [id])
  encontro     Encontro @relation(fields: [id_encontro], references: [id])
}
```

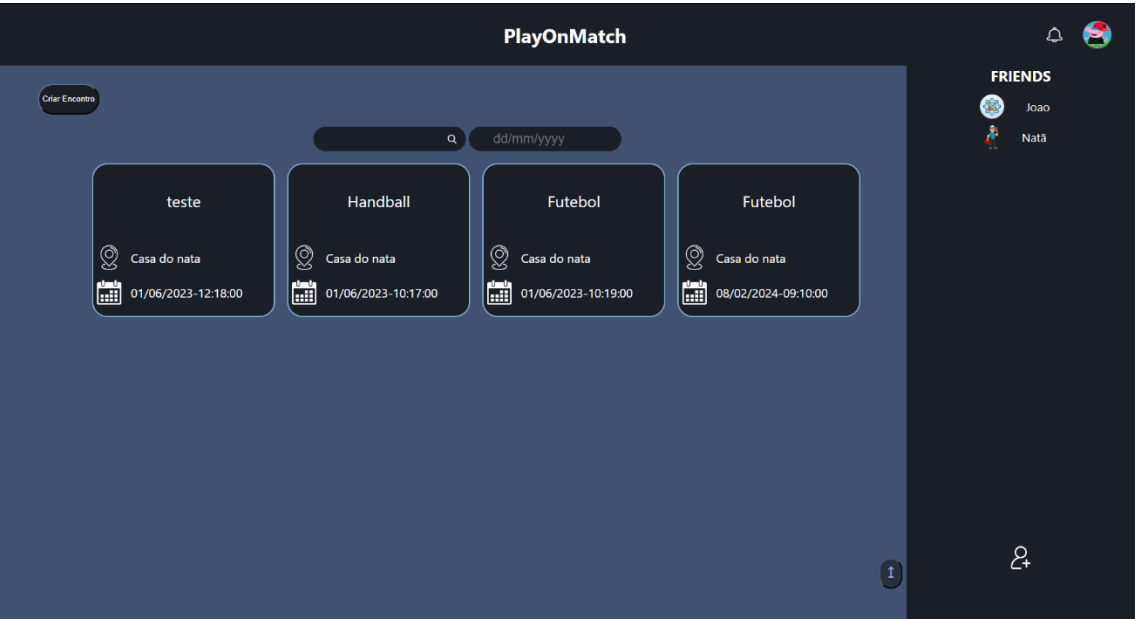
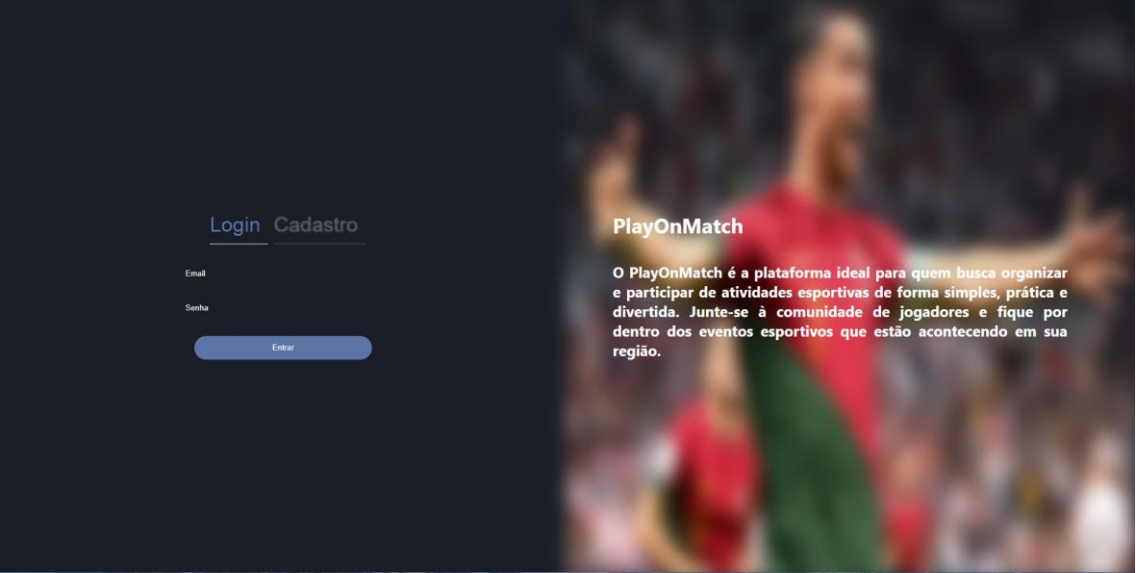
# Interface Web Mobile

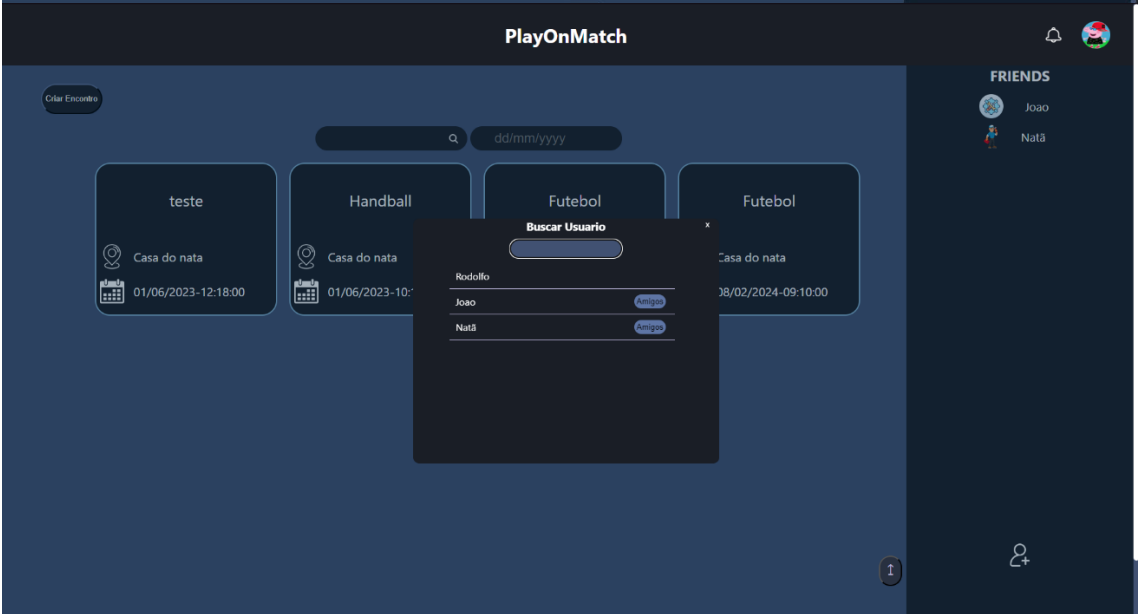
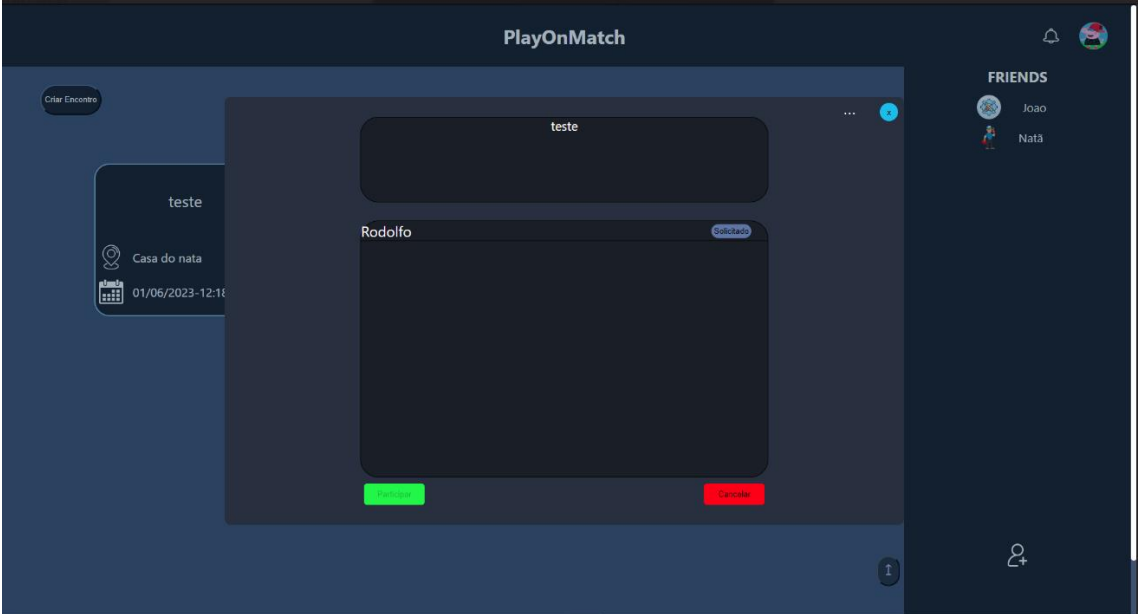
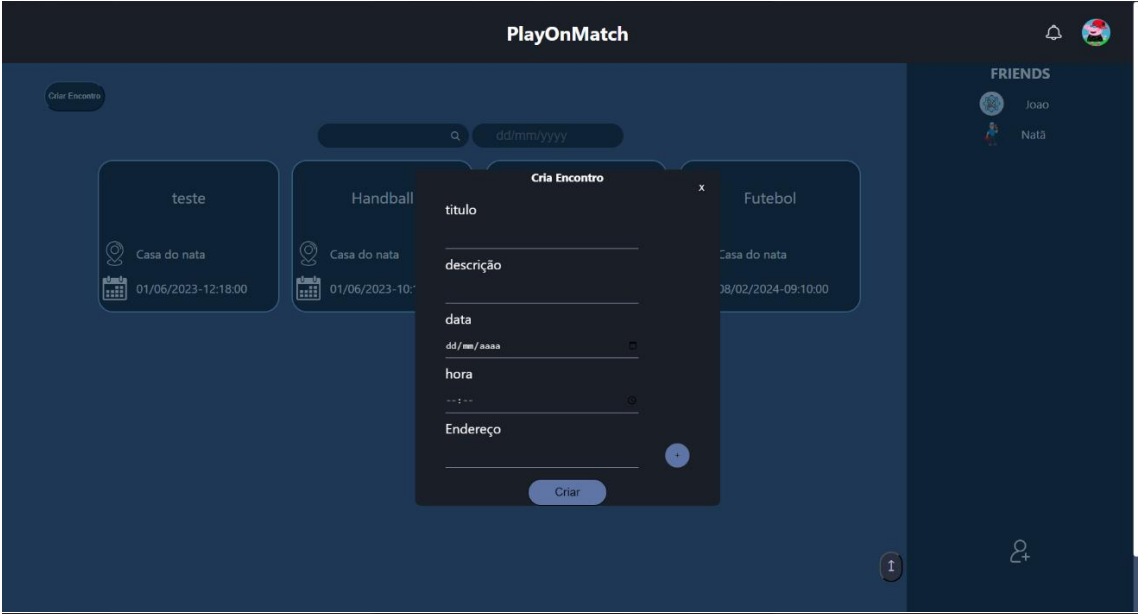





# EXECUÇÃO

## Web





PlayOnMatch




Rodolfo  
4 Partidas 2 Amigos

Partidas

teste	Casa do nata	01/06/2023	12:18:00	Não Acabou
-------	--------------	------------	----------	------------

PlayOnMatch



AMIGOS X

#2 Joao

#3 Natã

#1 Rodolfo

#1 Rodolfo

Partidas

teste	Casa do nata	01/06/2023	12:18:00	Não Acabou
-------	--------------	------------	----------	------------

Partidas

teste	Casa do nata	01/06/2023	12:18:00	Não Acabou
Handball	Casa do nata	01/06/2023	10:17:00	Não Acabou
Futebol	Casa do nata	01/06/2023	10:19:00	Não Acabou
Futebol	Casa do nata	08/02/2024	09:10:00	Não Acabou

Rodolfo

4 Partidas 2 Amigos

teste

participantes

#1Rodolfo

#2Joao

X

VS

tests

Campinas-Brasil

Casa do nata

01/06/2023

teste

Handball

Futebol

Futebol

Casa do nata

Casa do nata

01/06/2023

08/02/2024

10:19:00

09:10:00

Não Acabou

Não Acabou

Não Acabou

Não Acabou

## LoginCadastro

LOGIN

CADASTRO

teste3

teste3@gmail.com

...

Data de nascimento:

Selecione o dia

1



Selecione o mês

1



Selecione o ano

1901



Criar

# LoginCadastro

LOGIN

CADASTRO

## LOGIN

Digite o seu e-mail

teste3@gmail.com

Digite a sua senha

...

Entrar



teste

Descricao: teste

Local: Serra Dourada

Endereço:

Data: 2023-04-12T00:00:00.000Z

Encerrar Partida

ertreter

Descricao: teertert

Local: Serra Dourada

Endereço:

Data: 2024-04-04T00:00:00.000Z

Participar

Teste 77

Titulo do encontro

Teste 45

Descrição do encontro

Teste 45

Selecione o Local

Nova



Selecione o dia

31



Selecione o mês

5



Selecione o ano

2024



Cadastrar



Digite o nome do local

Digite o nome da rua

Digite o nome do bairro

Digite o nome da cidade

Digite o nome do estado

Digite o nome do País

senai  
senai@gmail.com  
07/07/2007  
[Atualizar perfil](#)

### Histórico

Local: Serra Dourada  
Cidade: Jaguariúna - Brasil  
Data e hora: 09/03/2024 - 00:00

Local: Serra Dourada  
Cidade: Jaguariúna - Brasil  
Data e hora: 18/05/2023 - 00:00

teste3

teste3@gmail.com

Data de nascimento:

Selecione o dia

1



Selecione o mês

1



Selecione o ano

1901



Atualizar

## CONCLUSÃO OU RESULTADOS

### Link Portfólio

- <https://batista29.github.io>
- RodolfoAura.github.io
- <https://Joao-Augusto0.github.io>

### Link Github

- <https://github.com/batista29>
- <https://github.com/RodolfoAura>
- <https://github.com/Joao-Augusto0>

## BIBLIOGRAFIAS

- <https://stackoverflow.com/>
- <https://chat.openai.com/>
- <https://reactnative.dev/>
- <https://expo.dev/>
- <https://www.npmjs.com/>
- <https://reactnavigation.org/>
- <https://github.com/react-native-picker/picker>