

Spotted

Alunos:Luis Justino,Thiago e Anderson.

O que Spotted?

Uma nova forma de descobrir o nome e conhecer aquela pessoa admirada a distância

Tecnologia usadas

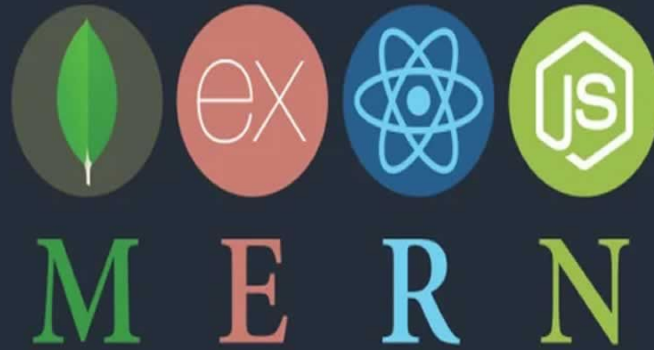
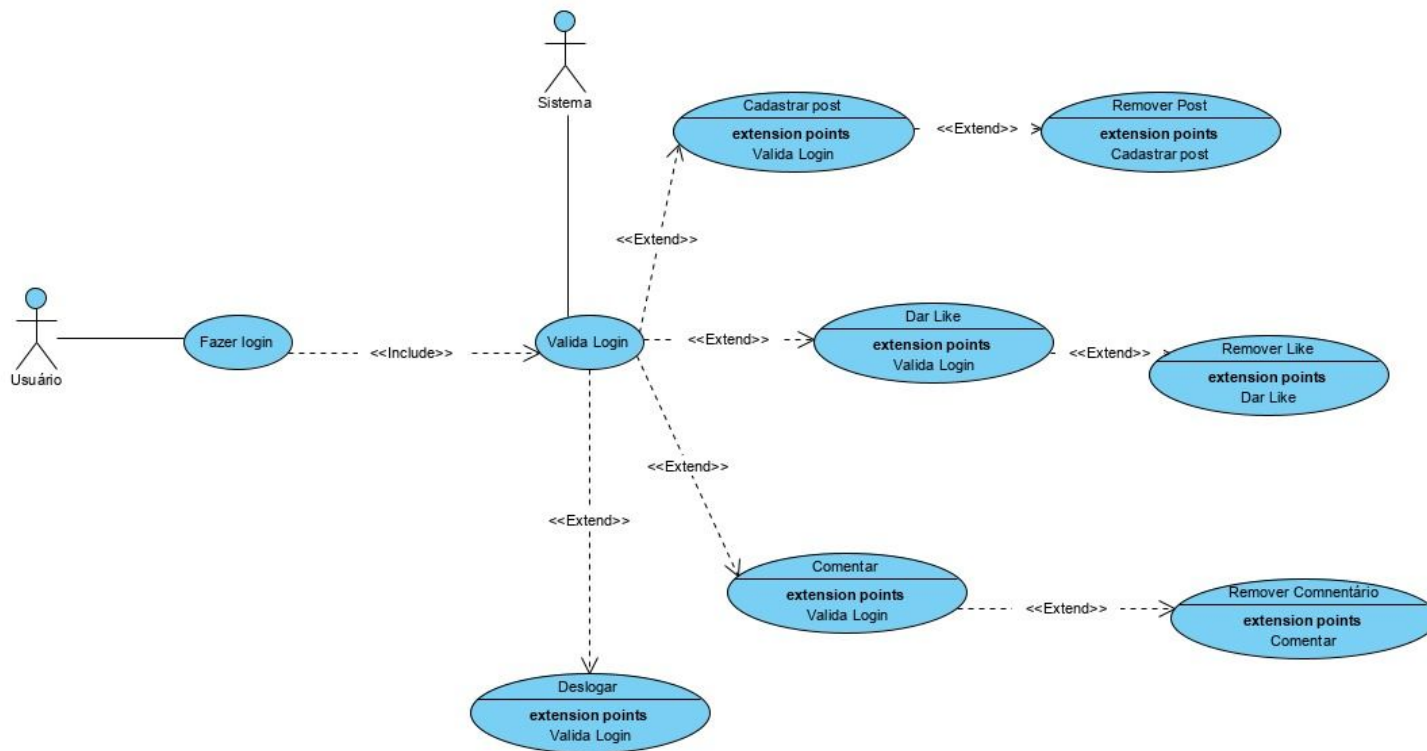


Diagrama de Uso



Fluxo de casos de uso

Descrição: Usuário usa uma conta para entrar no aplicativo, lá ele poderá ver posts dar like nos posts, comentar e postar suas próprias.

Fluxo Básico:

- Usuário cria uma conta no aplicativo
- Usuário insere o e-mail e senha para cadastro
- Após o cadastro o usuário entra no aplicativo com o e-mail e a senha cadastrada
- Usuário faz um post no aplicativo
- Usuário faz um comentário em um outro post
- Usuário dá um gostei em um post
- Usuário deleta o post que ele criou
- Usuário sai do aplicativo

Fluxo de casos de uso

Fluxo Alternativo 1:

Usuário cria uma conta no aplicativo

 Usuário insere o e-mail e senha para cadastro

 Após o cadastro o usuário entra no aplicativo com o e-mail e a senha cadastrada

 Usuário faz um post no aplicativo

 Usuário sai do aplicativo

Fluxo de casos de uso

Fluxo Alternativo 2:

Usuário cria uma conta no aplicativo

Usuário insere o e-mail e senha para cadastro

Após o cadastro o usuário entra no aplicativo com o e-mail e a senha cadastrada

Usuário faz um post no aplicativo

Usuário faz um post no aplicativo

Usuário faz um comentário em um outro post

Usuário sai do aplicativo

Diagrama de sequência

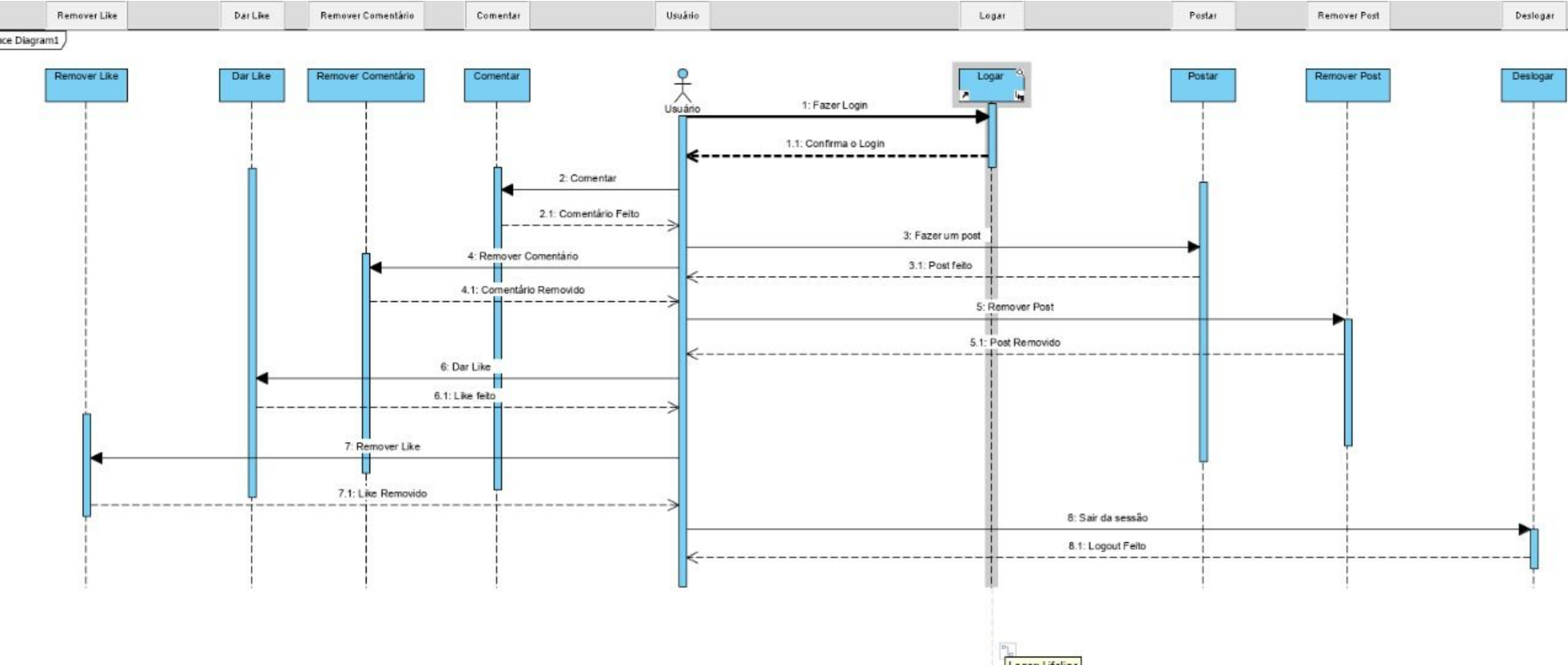


Diagrama de atividade

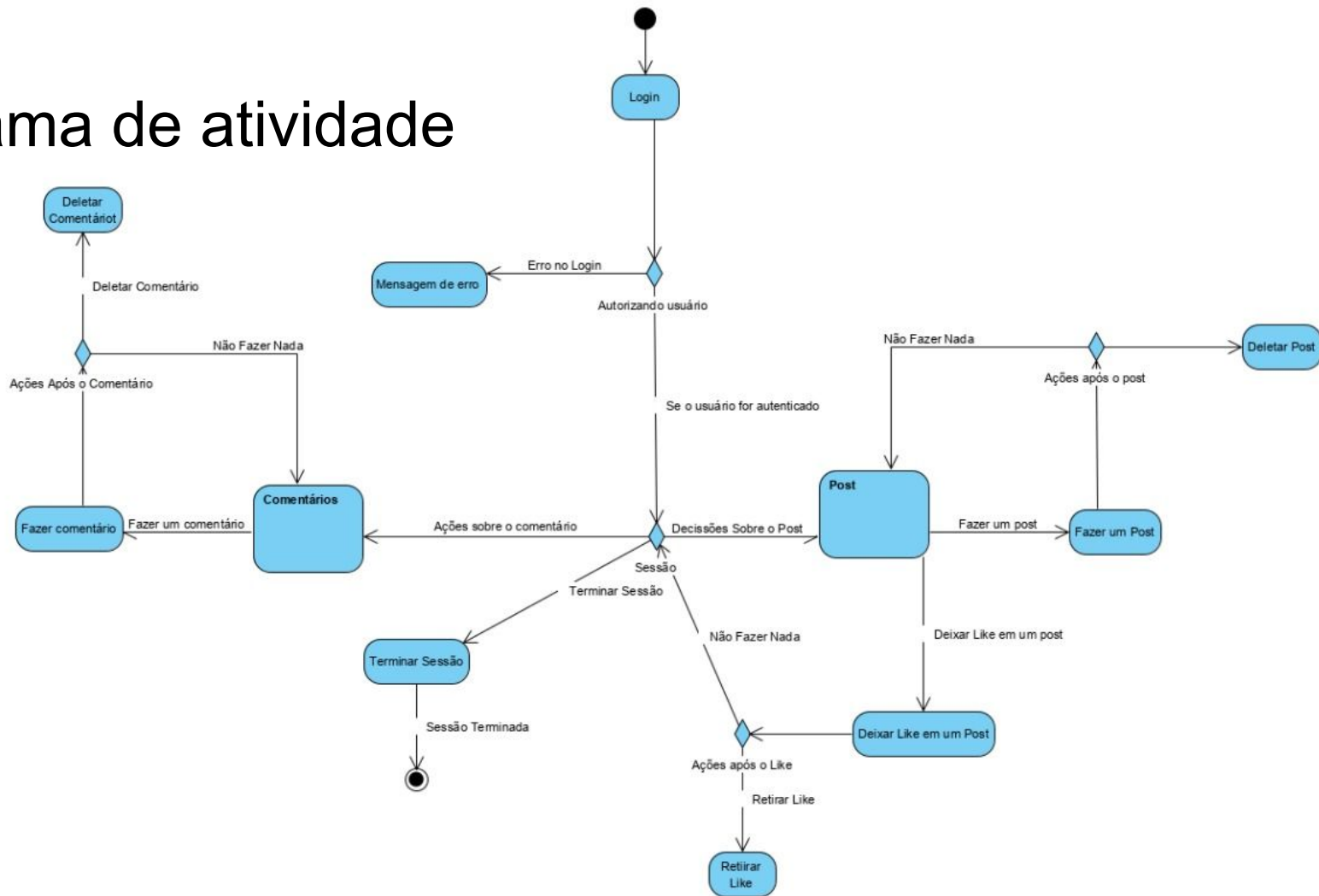


Diagrama de estado

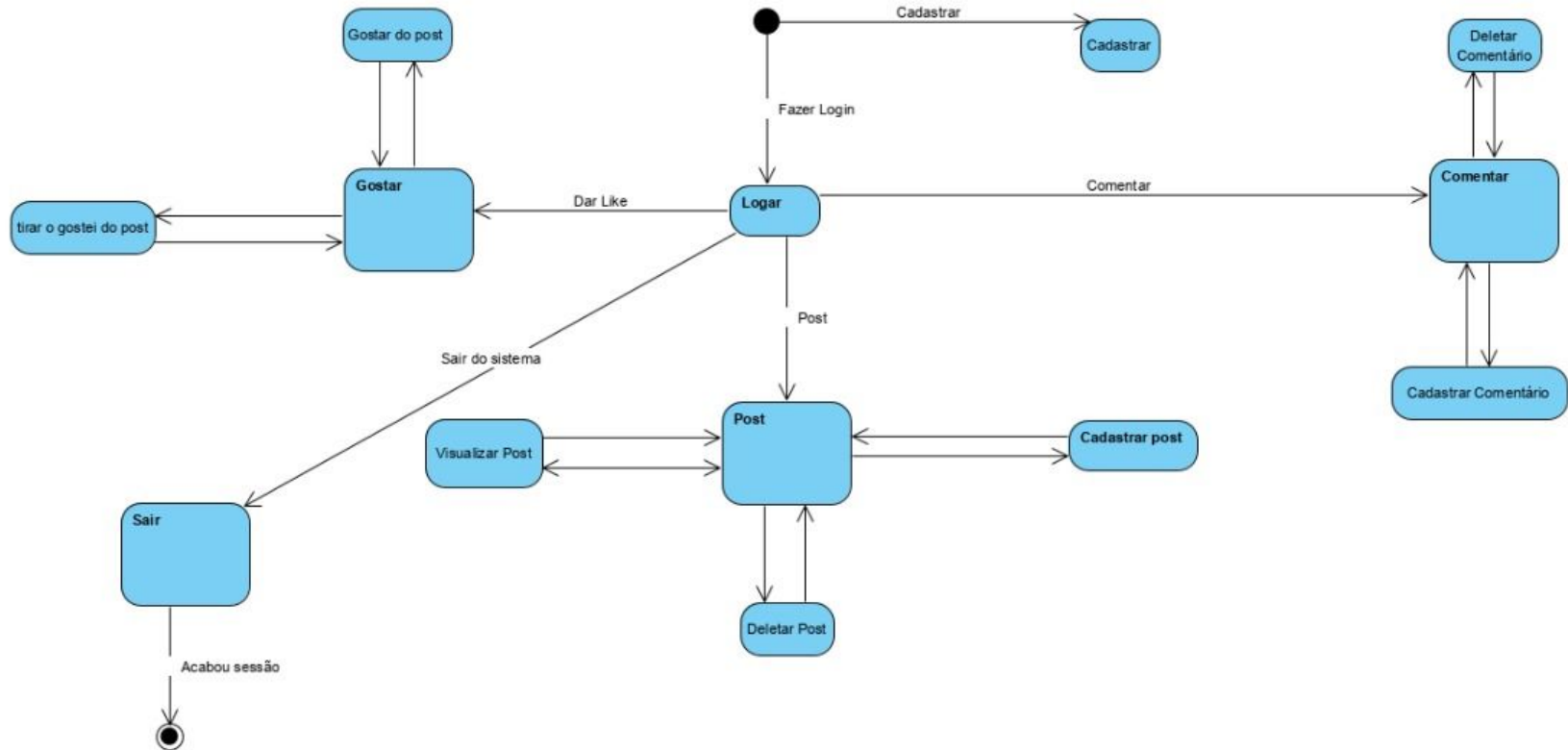
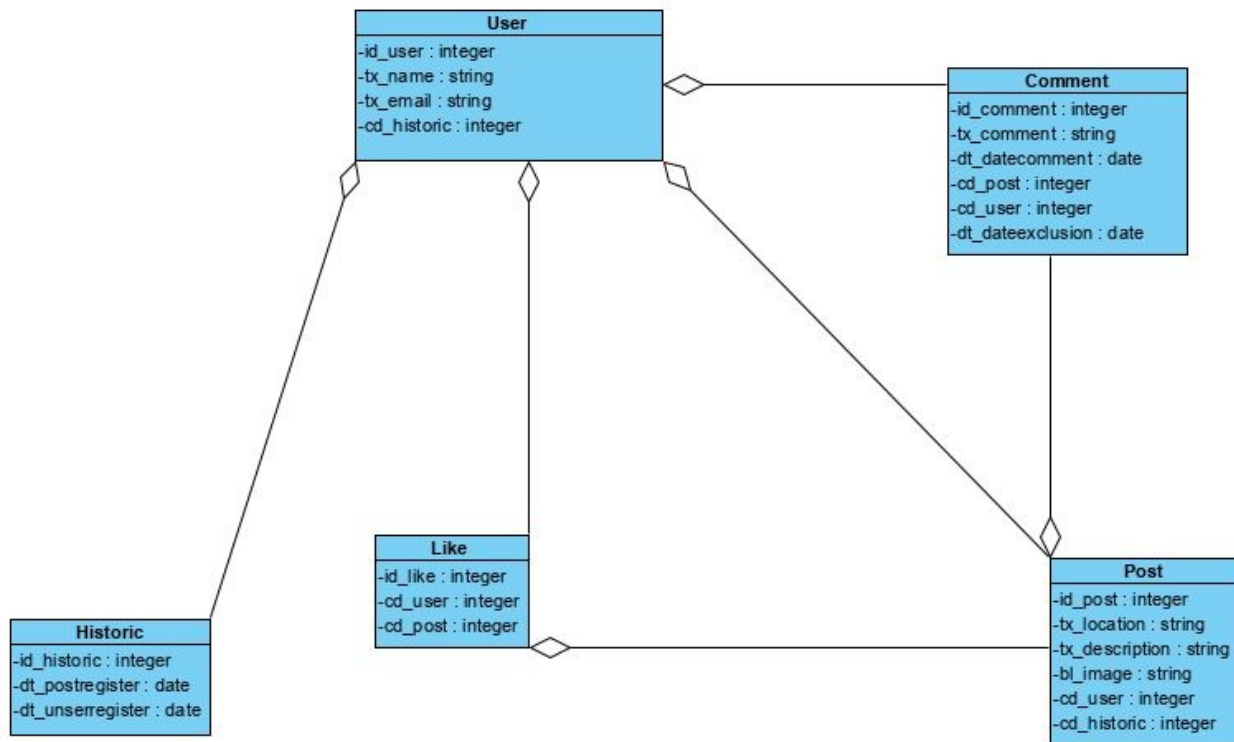


Diagrama de classes



Relatórios

1 - Relacione o nome e e-mail de usuários com uma postagem no mês de maio de 2019. Ordene o relatório pelo nome de forma decrescente.

```
select public.user.tx_nome, public.user.tx_email from public.user inner join  
public.post on public.post.cd_user = public.user.id_user order by tx_nome  
desc;
```

Relatórios

2- Relacione o nome do usuário, o código do post e a quantidade de likes de posts feitos por usuários cujo email termine em 'com.br'. Ordene o relatório do post com mais likes para o post com menos likes.

```
select public.user.tx_nome, public.post.id_post,public.post.nr_like from public.user  
inner join public.post on public.post.cd_user =public.user.id_user where  
public.user.tx_nome like '%.com.br' order by nr_like desc;
```

Relatórios

3- Relacione o código do post, nome do post e somente o último comentário de todos os posts feitos em 2019. Posts sem comentários devem ser relacionados também. Ordene o relatório do post mais antigo para o post mais recente

```
select public.user.tx_nome, public.post.id_post, (SELECT * FROM
public."coment" where public."coment".dt_cadastro BETWEEN '2019-01-01'
and '2019-12-31' order by dt_cadastro desc LIMIT 1) from public.user inner
join public.post on public.post.cd_user =public.user.id_user where
public.user.tx_nome like '%.com.br' order by dt_cadastro asc;
```

Relatórios

4- Relacione o código do usuário, nome do usuário e descrição do post para posts realizados em São Miguel do Oeste antes de 2020 e que possuem mais de 100 likes. Ordene o relatório pela descrição do post de forma decrescente.

```
select public.user.tx_nome, public.user.id_user, public.post.tx_descricao from public.user inner join  
public.post on public.post.cd_user =public.user.id_user inner join public.historico on  
public.post.cd_user =public.user.id_user where public.post.tx_localizacao like 'São Miguel do  
Oeste' and public.post.nr_like > 100 and public.historico.dt_cadastro < '2020-01-01 00:00:00' order  
by tx_descricao desc;
```