

Universidade da Madeira
Aplicações Centradas em Redes
Projeto prático 2
Aplicação de chat

Luís Carlos Gonçalves Freitas, 2029715
Manuel Joaquim Andrade Sousa Perez, 2029015

6 de Janeiro de 2019

Índice

1	Introdução	3
2	Requisitos implementados	3
2.1	Requisitos tecnológicos	3
2.1.1	Utilização de Socket.io	3
2.1.2	Utilização de EmbedJS	3
2.1.3	Utilização de Vue.js	4
2.1.4	Utilização de Express	4
2.2	Requisitos funcionais	4
2.2.1	Registo de utilizadores	4
2.2.2	Histórico de conversas	5
2.2.3	Diferentes tipos de conversas	6
3	Utilização da aplicação	6
3.1	Página Inicial	6
3.2	Login/Registo	7
3.2.1	Registo	7
3.2.2	Login com a conta do registo	8
3.2.3	Login com Facebook	8
3.3	Após ter feito login	9
4	Conclusão	10
	Referências	10

1 Introdução

Esta aplicação consistirá numa aplicação desenvolvida em Node.js utilizada para a comunicação entre diferentes pessoas em tempo real através da Internet. Como tal, terá vários tipos de utilizadores, vários tipos de conversas, e outras funcionalidades que serão abordados ao longo deste relatório.

2 Requisitos implementados

Para começar, a lista de requisitos a implementar é a seguinte [1]:

Tipo de requisito	Requisito	
Requisitos tecnológicos	Utilização de Socket.io	✓
	Utilização de EmbedJS	✓
	Utilização de Vue.js	✓
	Utilização de Express	✓
Requisitos funcionais	Registo de utilizadores	✓
	Histórico de conversas	✓
	Diferentes tipos de conversas	Públicas ✓
		Privadas ✓

Tabela 1: Lista de requisitos

2.1 Requisitos tecnológicos

2.1.1 Utilização de Socket.io

O Socket.io foi utilizado para possibilitar a comunicação em tempo real entre o(s) cliente(s) e o servidor, sendo que, neste caso, aplicaria-se à receção/envio das mensagens num dado chat (público ou privado) em tempo real.



Figura 1: Socket.io ¹

2.1.2 Utilização de EmbedJS

Da mesma forma, o EmbedJS foi usado para estruturar as views da app e os respetivos binding dos dados envolvidos, seja tanto na página principal como também nas páginas de login/registo e a página do perfil do utilizador.

¹<https://socket.io>



Figura 2: EmbedJS ²

2.1.3 Utilização de Vue.js

Do mesmo modo, a framework Vue.js foi utilizada para popular a UI desta app, sendo que trata do binding dos dados provenientes do back-end para cada componente presente nas views.



Figura 3: Vue.js ³

2.1.4 Utilização de Express

Igualmente, a framework Express foi empregue na parte da estruturação do ambiente de desenvolvimento da app em inúmeros factores como a organização dos ficheiros de código-fonte, a definição de rotas, entre outros.



Figura 4: Express ⁴

2.2 Requisitos funcionais

2.2.1 Registo de utilizadores

O registo dos utilizadores foi feito recorrendo a uma base de dados SQLite, no qual estão armazenados os detalhes acerca do utilizador - as suas credenciais de login e o seu nome.

²<https://ejs.co>

³<https://vuejs.org>

⁴<https://expressjs.com>

id	username	password	name
Filter	Filter	Filter	Filter
7	lfreitas	1234	Luís Freitas
8	ww	ww	ww
23076126259...	23076126259...		Joaquim Perez

Figura 5: Representação das mensagens na base de dados

Para além disso, tanto podem tratar-se de utilizadores com conta local, como também de utilizadores com conta ligada ao Facebook.

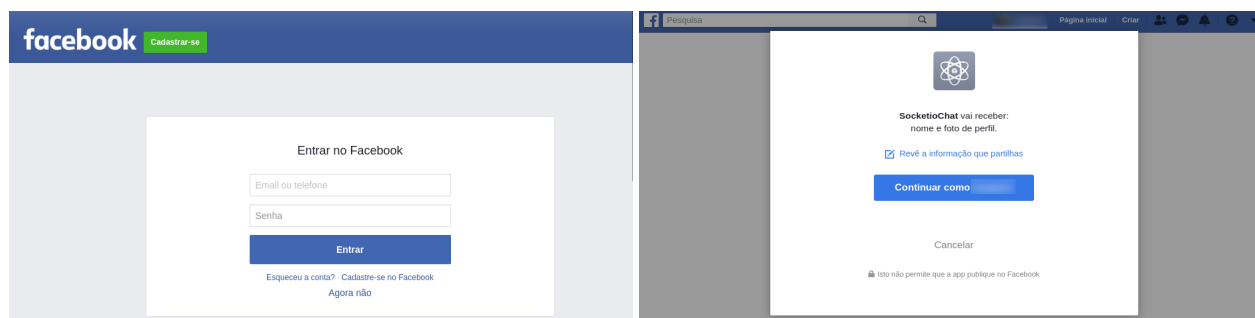


Figura 6: Exemplo de login pelo Facebook

2.2.2 Histórico de conversas

Para o histórico de conversas, tal como para o registo de utilizadores, foi utilizada uma base de dados SQLite. Neste caso, as mensagens foram armazenadas numa tabela à parte da tabela dos utilizadores. Assim sendo, cada mensagem seria composta (a nível da base de dados) por um remetente, um destinatário, o texto da mensagem e um timestamp relativo ao seu envio.

id	user_from	user_to	text	timestamp
Filter	Filter	Filter	Filter	Filter
18	2307612625936728	7	Olá	2018-11-29 02:06:56
19	7	8	só vi agora a msg	2018-11-29 02:07:13
20	7	8	desculpa	2018-11-29 02:07:17

Figura 7: Representação das mensagens na base de dados

2.2.3 Diferentes tipos de conversas

Em termos de conversas, podem haver dois tipos de conversas: **públicas** e **privadas**. Para as públicas, não é necessário estar registado nem com sessão iniciada para receber ou enviá-las, sendo que as privadas são individualizadas e estas estão associadas a um utilizador remetente e a um utilizador destinatário.

id	user_from	user_to	text	timestamp
Filter	Filter	Filter	Filter	Filter
11	2307612625936728	7	olá Luís	2018-11-29 01:59:36
12	2307612625936728		olá a todos os anónimos	2018-11-29 02:00:14
13			olá anónimo	2018-11-29 02:04:04
14			tu é que és anónimo	2018-11-29 02:04:45

Figura 8: Representação dos vários tipos de conversas

3 Utilização da aplicação

3.1 Página Inicial

Para começar, ao entrar na aplicação é mostrado ao utilizador o chat público, para a utilização deste o utilizador não precisa de ter login efetuado, ou seja, as mensagens que o utilizador enviar irá ficar como sendo de um utilizador desconhecido. Aqui o utilizador pode conversar de forma anónima com os vários utilizadores.

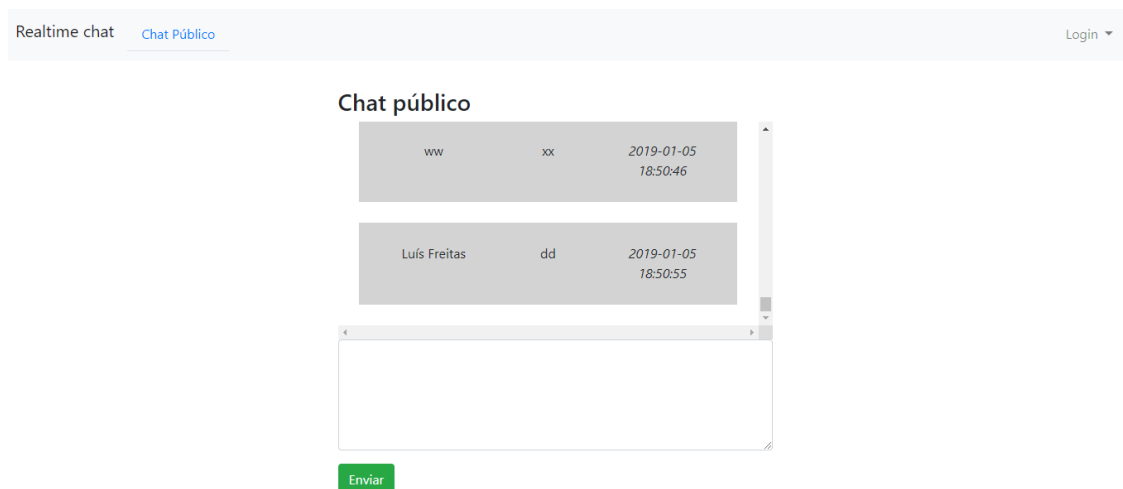


Figura 9: Página inicial/Chat Público

3.2 Login/Registo

Existe ainda a opção do utilizador criar conta, efetuar login com a conta criada e efetuar login com o Facebook.

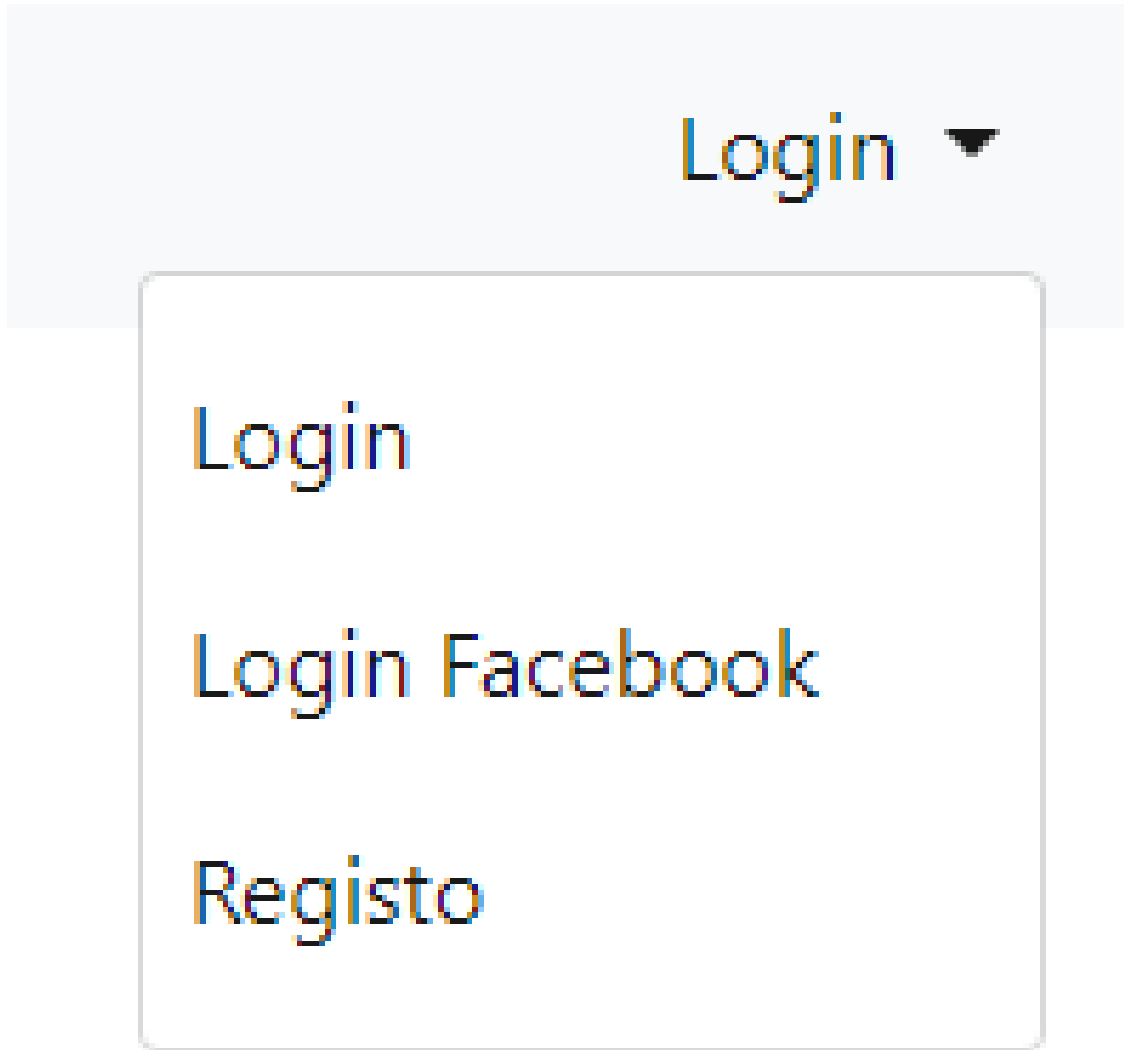


Figura 10: Login/Registo

3.2.1 Registo

É pedido ao utilizador para inserir os seguintes dados para que a conta possa ser criada.

Registo

Nome

Nome de Utilizador

Palavra-Passe

[Registrar](#)

Figura 11: Registo

3.2.2 Login com a conta do registo

É pedido os seguintes dados, referentes ao registo que foi efetuado anteriormente.

Login

Nome de Utilizador

Palavra-Passe

[Entrar](#)

Figura 12: Login com a conta do registo

3.2.3 Login com Facebook

O utilizador pode optar por não criar conta na aplicação, fazendo login com o Facebook, tal como existe na maioria das aplicações atualmente.

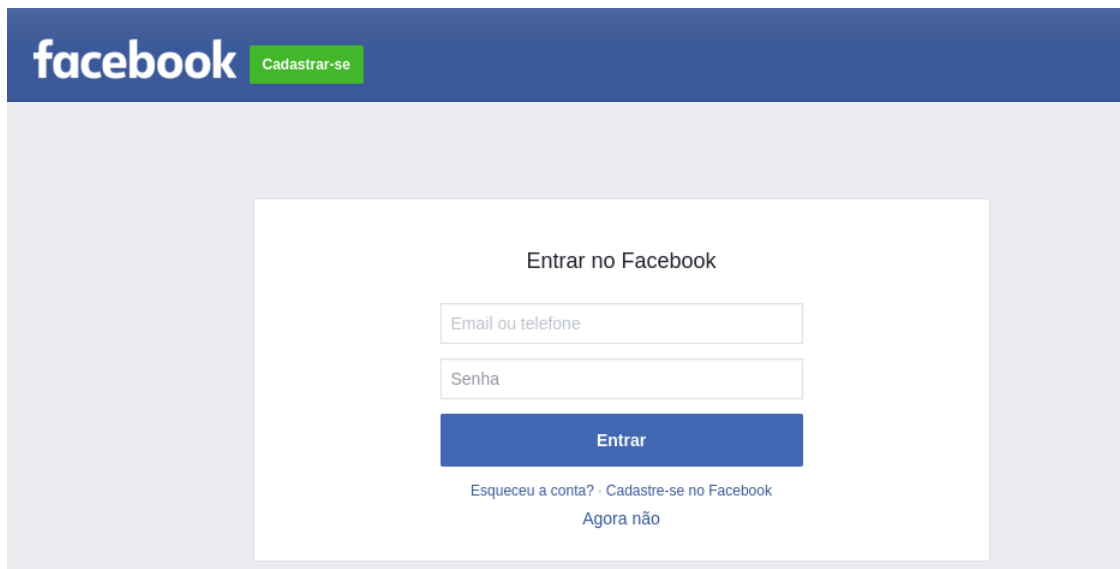


Figura 13: Login com Facebook

3.3 Após ter feito login

Após o utilizador ter feito login na aplicação, seja com conta local, ou com o Facebook, o utilizador passa a ser identificado na aplicação.

Para além do chat privado, o utilizador passa a ter mais uma funcionalidade, o chat privado, aqui o utilizador escolhe o utilizador que pretende conversar e a mensagem é enviada apenas para aquele utilizador.

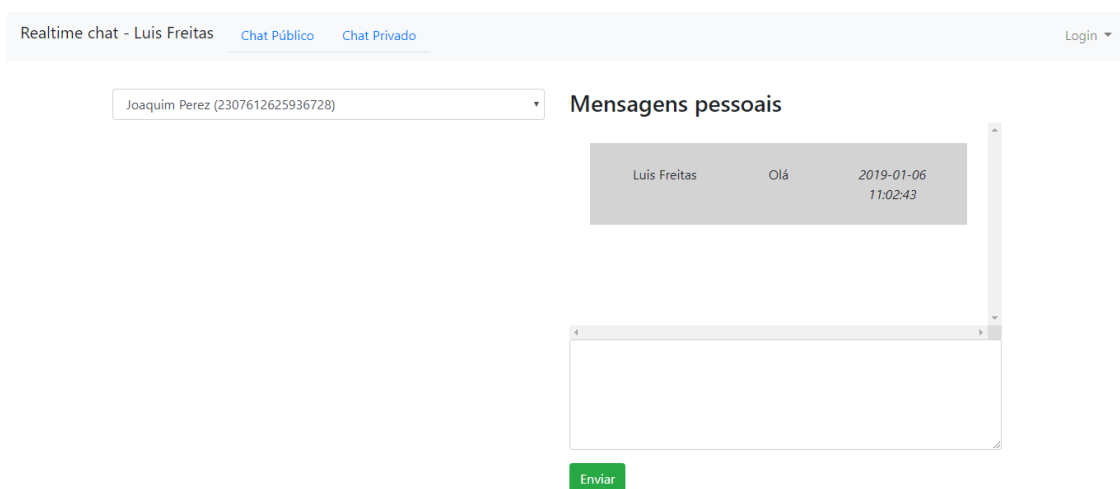


Figura 14: Chat privado

4 Conclusão

Por fim, os objetivos gerais da aplicação que se pretendia implementar foram alcançados, ou seja, ter um chat público em que todos os utilizadores comunicam com todos, um chat privado e login com uma aplicação externa, que no nosso caso foi o Facebook.

A nível mais teórico foi muito útil pois passamos a conhecer melhor algumas frameworks, tais como Node.js, EmbedJS e Vue.js, sendo que estas duas últimas foram utilizadas para facilitar o tratamento dos dados para as views, sendo frameworks de front-end.

Outro componente muito importante que aprendemos foi o Socket.io, o que facilita bastante em aplicações deste tipo (de tempo real), ou seja, em vez de estarmos constantemente a ir a base de dados verificar se para aquele utilizador existiam mensagens novas, o que iria ser muito dispendioso, ao utilizarmos o Socket.io não temos de estar a fazer nenhuma verificação, quando houverem mensagens novas, todos os utilizadores ligados aquele socket vão receber os dados sem necessidade constante de verificar quando existem dados no socket.

Referências

- [1] Enunciado do projeto prático 2, <https://moodle.cee.uma.pt/pluginfile.php/160/course/section/978/Enunciado%20do%20trabalho%20pr%C3%A1tico%20P2.pdf>.