



**UNIVERSIDADE DE SÃO PAULO  
ESCOLA DE ARTES, CIÊNCIAS E HUMANIDADES**

ACH2026 – Redes de Computadores

Gustavo Antônio Sabino Batista, Nº USP 10723732

Igor Antun da Costa Gago, Nº USP 10687509

Vinicius Hessel Pereira, Nº USP 10783006

**Relatório do Exercício de Programação 1 – Sistema de loteria online**

**INTRODUÇÃO TEÓRICA**

O tema escolhido pelo nosso grupo para o EP1 foi desenvolvimento, e a nossa proposta foi criar um sistema de loteria online, onde os usuários podem comprar e fazer apostas em diferentes sorteios programados do sistema e um websocket realtime mostra os resultados dos sorteios ao vivo e quando outros usuários fazem apostas nos sorteios.

**RESULTADOS**

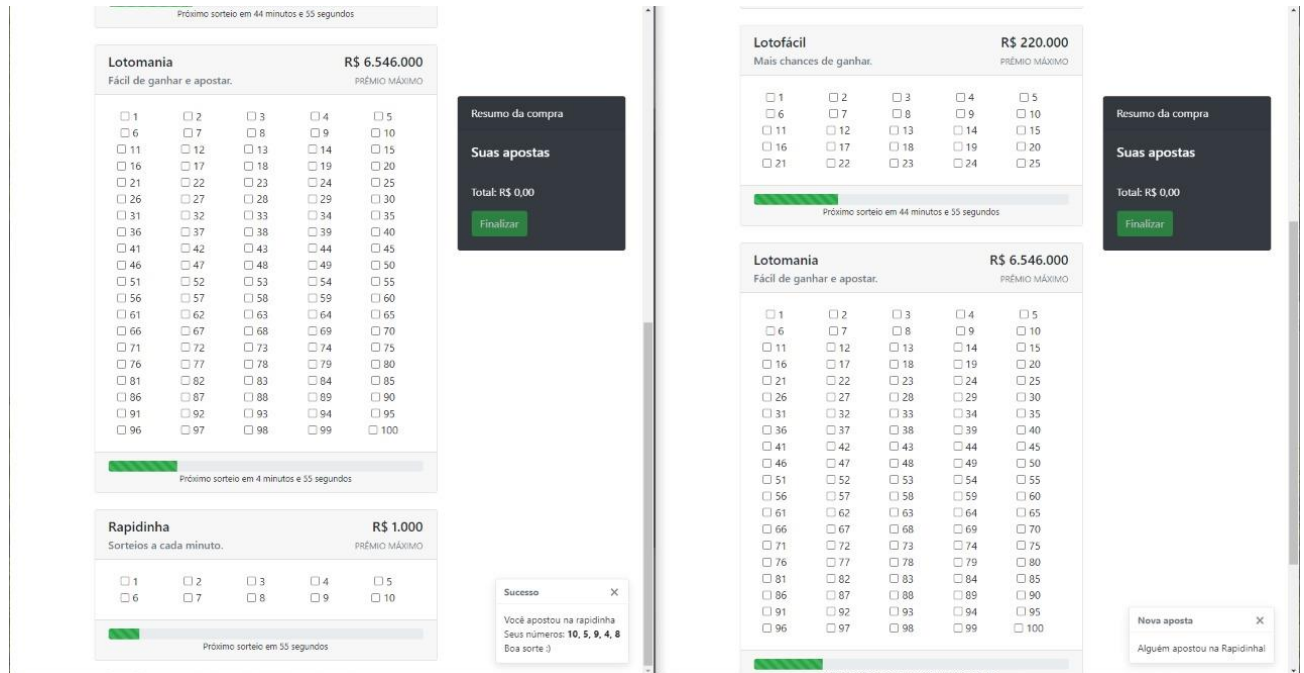
Criamos uma aplicação em Node.js, com apenas uma interface, que mostra os tipos de sorteios disponíveis, bem como uma área para o usuário selecionar quais números quer comprar dos sorteios, e um carrinho com o resumo da compra dos números. Os sorteios criados foram: “Mega-Sena” que acontece uma vez por dia e possui o maior prêmio do sistema, “Lotomania” que acontece uma vez a cada uma hora e possui o segundo maior, “Lotofácil” que acontece uma vez a cada cinco minutos e possui o segundo menor prêmio, e a “Rapidinha” que acontece uma vez a cada minuto.

Quando um usuário faz a compra de uma aposta, é gerada uma requisição HTTP do tipo POST na API do sistema, e desenvolvemos também um websocket realtime que exibe para o usuário os resultados dos sorteios, e também quando outros usuários comprem apostas para os sorteios disponíveis. Poderíamos ter feito a aposta também por websocket, mas escolhemos fazer uma request HTTP para ter um contato maior com essa tecnologia.

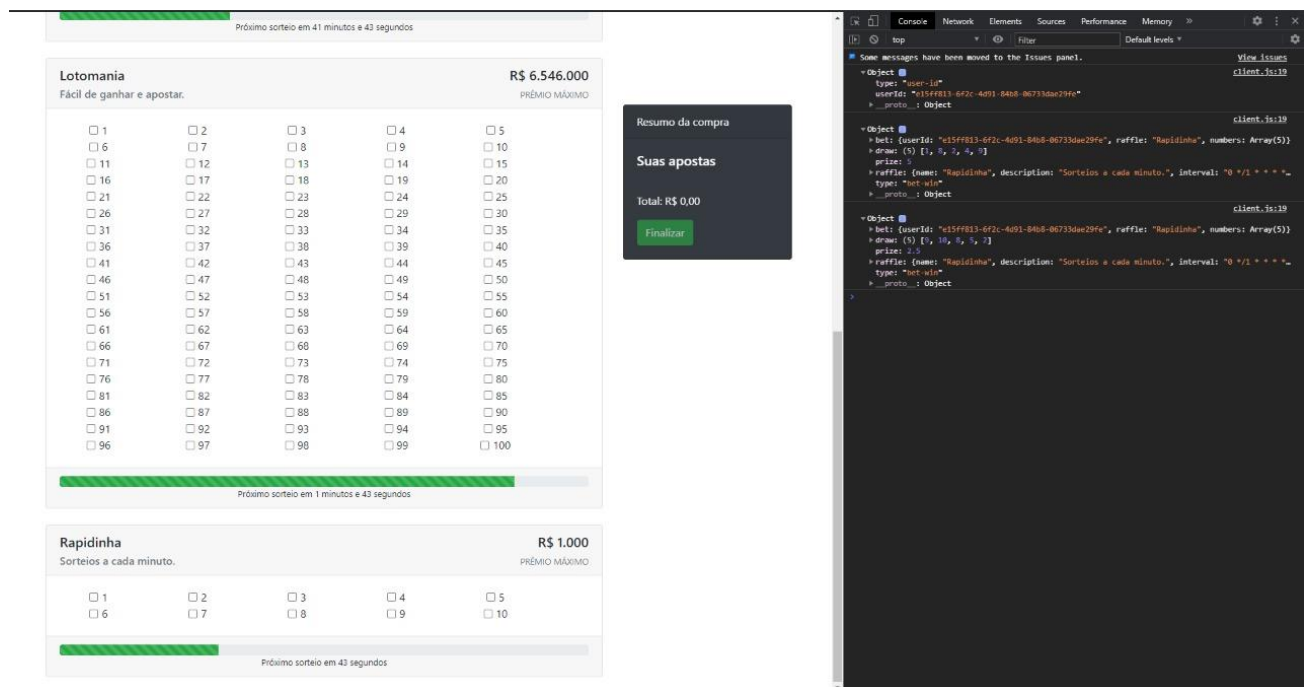
A linguagem usada foi JavaScript e os arquivos principais são, para o servidor (back-end), o `app.js` e para o navegador (front-end), o `client.js`, sendo que o servidor é aquele que serve o arquivo para o navegador. Para executar o código do EP é necessário ter instalado o Node.js, navegar até a pasta do projeto, digitar o comando `npm install`, e o comando `npm start` ou `node src/app.js` para rodar.

## TESTES

- Na esquerda, registro da aposta de um cliente, onde o mesmo recebe a confirmação da aposta pelo websocket. Na direita, outro cliente recebendo a notificação por websocket de que outro usuário teria feito uma aposta:



- Exemplo de mensagens de eventos que o servidor manda por websocket:



- Resposta de uma request HTTP do registro de uma aposta:

The image shows a screenshot of a lottery application interface on the left and a browser's developer tools on the right, illustrating the HTTP request for a bet registration.

**Lotomania Interface:**

- Próximo sorteio em 39 minutos e 19 segundos
- R\$ 6.546.000 PRÊMIO MÁXIMO
- Fácil de ganhar e apostar.
- Grid of numbers 1 to 100 (5 columns x 20 rows).
- Próximo sorteio em 4 minutos e 19 segundos

**Rapidinha Interface:**

- Sorteios a cada minuto.
- R\$ 1.000 PRÊMIO MÁXIMO
- Grid of numbers 1 to 10 (2 columns x 5 rows).
- Próximo sorteio em 19 segundos

**Resumo da compra (Overlay):**

- Suas apostas
- Total: R\$ 0,00
- Finalizar

**HTTP Request (Developer Tools):**

- Name: localhost
- Request URL: http://localhost:3000/bet
- Request Method: POST
- Status Code: 200 OK
- Remote Address: [::1]:3000
- Referer Policy: strict-origin when cross-origin
- Response Headers:
  - Connection: keep-alive
  - Content-Length: 7
  - Date: Mon, 26 Oct 2020 15:21:33 GMT
- Request Headers:
  - Accept: \*/\*
  - Accept-Encoding: gzip, deflate, br
  - Accept-Language: pt-BR,pt;q=0.9,en-US;q=0.8,en;q=0.7
  - Cache-Control: no-cache
  - Connection: keep-alive
  - Content-Length: 192
  - Content-Type: text/plain;charset=UTF-8
  - Host: localhost:3000
  - Origin: http://localhost:3000
  - Pragma: no-cache
  - Referer: http://localhost:3000/
  - sec-ch-ua: "Chromium";v="86", "Not(A)Brand";v="99", "Google Chrome";v="86"
  - sec-ch-ua-mobile: ?0
  - Sec-Fetch-Dest: empty
  - Sec-Fetch-Mode: cors
  - Sec-Fetch-Site: same-origin
  - User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.111 Safari/537.36
- Request Payload:
 

```
{
  "userId": "68a279a8-af68-461c-9465-d23a5e9676f1",
  "raffle": "Rapidinha",
  "numbers": [4, 9, 8, 3, 7]
}
```

- Exemplo de notificação por websocket do resultado de um sorteio para o usuário:

The image shows a screenshot of the lottery application interface, similar to the previous one, but with a winning notification overlay for the Rapidinha game.

**Lotomania Interface:**

- Próximo sorteio em 43 minutos e 1 segundos
- R\$ 6.546.000 PRÊMIO MÁXIMO
- Fácil de ganhar e apostar.
- Grid of numbers 1 to 100 (5 columns x 20 rows).
- Próximo sorteio em 2 minutos e 1 segundos

**Rapidinha Interface:**

- Sorteios a cada minuto.
- R\$ 1.000 PRÊMIO MÁXIMO
- Grid of numbers 1 to 10 (2 columns x 5 rows).
- Próximo sorteio em 1 segundos

**Resumo da compra (Overlay):**

- Suas apostas
- Total: R\$ 0,00
- Finalizar

**Notification Overlay:**

- Você ganhou!
- Você acertou os números sorteados pela Rapidinha!
- Seu prêmio: **R\$ 2,50**
- Sua aposta: 4, 9, 8, 3, 7
- Números sorteados: 9, 10, 8, 5, 2

- Exemplo de logs do servidor:

```
npm
x + v
usp
cd
~ usp
~/usp cd rc/ep-redes
~/usp/rc/ep-redes main npm start

> ep-redes@1.0.0 start /home/hessel/usp/rc/ep-redes
> node src/app.js

Server ready at :3000
2020-10-26T15:43:22.976Z - GET request received at /
2020-10-26T15:43:23.044Z - GET request received at /client.js
2020-10-26T15:43:23.049Z - GET request received at /timers.js
2020-10-26T15:43:24.321Z - ::1 connected
2020-10-26T15:43:30.360Z - POST request received at /bet
2020-10-26T15:43:30.362Z - New bet made in lottery "Rapidinha". Numbers: 10,5,9,3,8
2020-10-26T15:43:52.061Z - POST request received at /bet
2020-10-26T15:43:52.062Z - New bet made in lottery "Mega-Sena". Numbers: 15,20,25,30,35,40
2020-10-26T15:44:00.010Z - Rapidinha: (1 bets) - 6,1,3,10,5
2020-10-26T15:44:00.010Z - 170befb1-7e97-4142-99f3-4fa07c6f3d11 - won 5 (3/5)- Rapidinha
```

- Exemplo simulando a queda do servidor e a mudança do status do websocket para desconectado:

Loteria online

Desconectado

Mega-Sena

A loteria que paga milhões.

☐ 1

☐ 6

☐ 11

☐ 16

☐ 21

☐ 26

☐ 31

☐ 36

☐ 41

☐ 46

☐ 51

☐ 56

Lotofácil

R\$ 45.000.000

PRÊMIO MÁXIMO

Resumo da compra

~p/rc/ep-redes

usp

cd

~ usp

~/usp cd rc/ep-redes

~/usp/rc/ep-redes main npm start

> ep-redes@1.0.0 start /home/hessel/usp/rc/ep-redes

> node src/app.js

Server ready at :3000

2020-10-26T15:43:22.976Z - GET request received at /

2020-10-26T15:43:23.044Z - GET request received at /client.js

2020-10-26T15:43:23.049Z - GET request received at /timers.js

2020-10-26T15:43:24.321Z - ::1 connected

2020-10-26T15:43:30.360Z - POST request received at /bet

2020-10-26T15:43:30.362Z - New bet made in lottery "Rapidinha". Numbers: 10,5,9,3,8

2020-10-26T15:43:52.061Z - POST request received at /bet

2020-10-26T15:43:52.062Z - New bet made in lottery "Mega-Sena". Numbers: 15,20,25,30,35,40

2020-10-26T15:44:00.010Z - Rapidinha: (1 bets) - 6,1,3,10,5

2020-10-26T15:44:00.010Z - 170befb1-7e97-4142-99f3-4fa07c6f3d11 - won 5 (3/5)- Rapidinha

^C