

# WEB/APP SAO KÊ - TRA CỨU DỮ LIỆU GIAO DỊCH

Dự án phân tích và tối ưu tìm kiếm dữ liệu từ file CSV

GVHD: LÊ ĐÌNH THUẬN

Danh sách thành viên:

<u>Họ và tên</u>	<u>MSSV</u>
Đặng Nguyên Bảo	2310211
Bùi Trần Duy Khang	2311402
Nguyễn Tuấn Cường	2110882
Trần Đức Tiến	2014732
Đào Hữu Gia Huy	2014732
Nguyễn Nhu Xuân	1832063
Nguyễn Ngô Hoàng Long	2011553

# NỘI DUNG

Tổng quan đề tài

Giải pháp kỹ thuật

Yêu cầu hệ thống

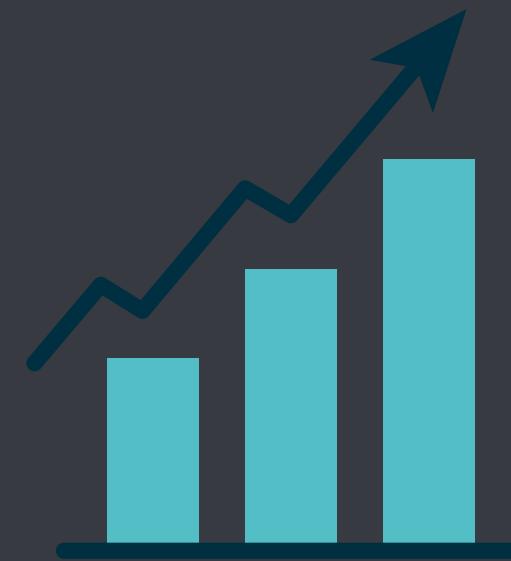
Cấu trúc source code

Kiến trúc hệ thống

Luồng hoạt động



# TỔNG QUAN ĐỀ TÀI



- Vai trò của minh bạch thông tin tài chính.
- Ứng dụng hỗ trợ tra cứu giao dịch sao kê tài chính của MTTQ VN.
- Xây dựng hệ thống dựa trên file CSV công khai.
- Mục tiêu:
  - Tra cứu nhanh, chính xác.
  - Áp dụng thuật toán và cấu trúc dữ liệu tối ưu.

# YÊU CẦU HỆ THỐNG

## Yêu cầu chức năng

- Tra cứu theo: Số tiền, tên người gửi, mã giao dịch, nội dung, thời gian
- Hiển thị nội dung tra cứu dưới dạng bảng

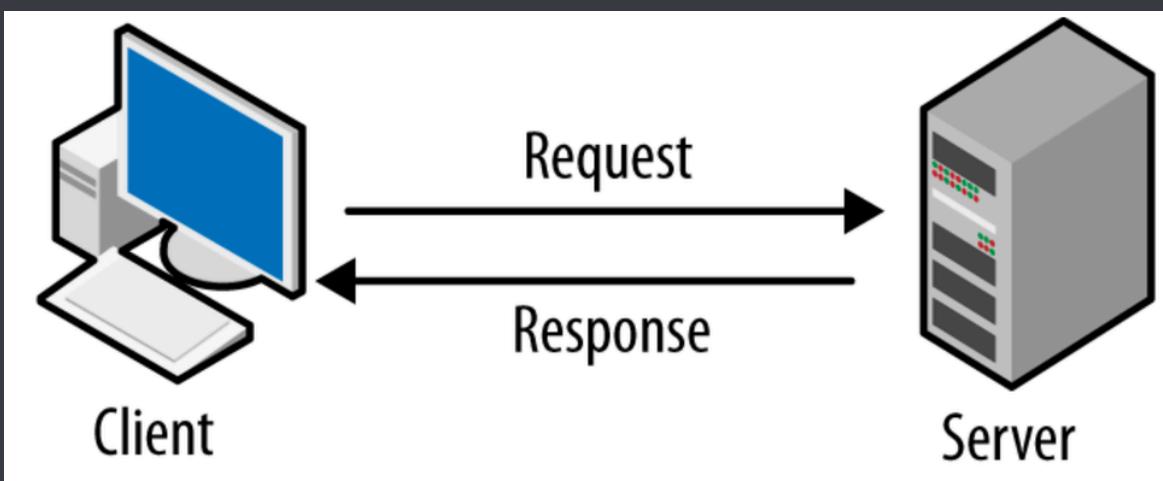


## Yêu cầu phi chức năng

- Hiệu năng cao
- Giao diện thân thiện
- Dễ mở rộng và bảo trì

# KIẾN TRÚC HỆ THÔNG ➔

Sử dụng sơ đồ client-server

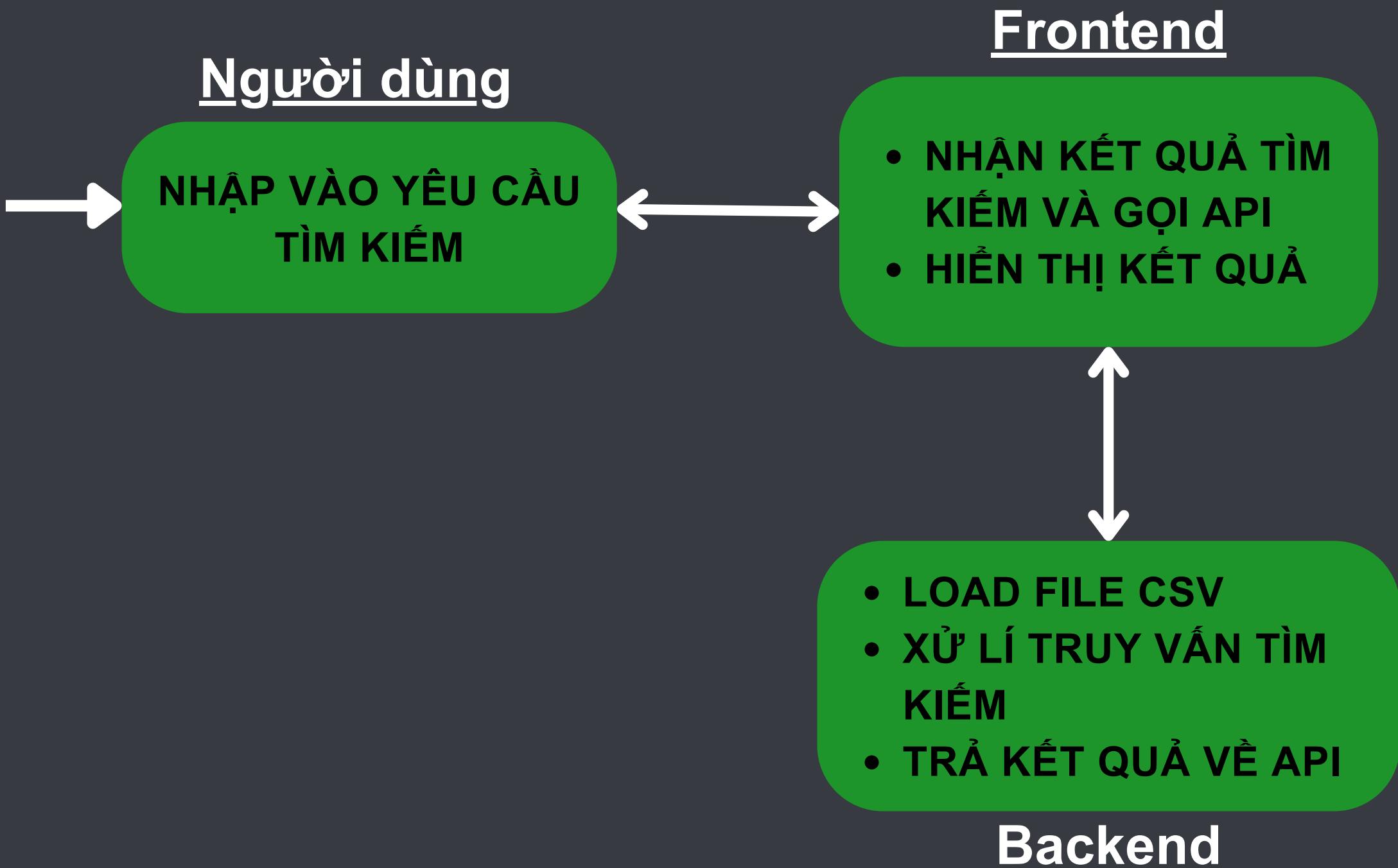


Backend: Xử lý file CSV, cung cấp API hỗ trợ tìm kiếm

Frontend: Hỗ trợ tìm kiếm và hiển thị kết quả

Công cụ: NodeJS, Express, Cors, parse-csv, React, TailwindCSS

# LUỒNG HOẠT ĐỘNG



# BINARY SEARCH

```
● ● ●  
1 let lo = 0,  
2   hi = this.keys[type].length - 1;  
3 let lower_bound = lo,  
4   upper_bound = hi; // the result will contains items in this range  
5 while (lo <= hi) {  
6   let mid = Math.floor((lo + hi) / 2);  
7   // a <= b => a - b <= 0  
8   if (comparator(from, this.keys[type][mid]) <= 0) {  
9     hi = mid - 1;  
10    lower_bound = mid;  
11  } else {  
12    lo = mid + 1;  
13  }  
14}  
15 (lo = 0), (hi = this.keys[type].length - 1);  
16 while (lo <= hi) {  
17   let mid = Math.floor((lo + hi) / 2);  
18   // a >= b => a - b >= 0  
19   if (comparator(to, this.keys[type][mid]) >= 0) {  
20     lo = mid + 1;  
21     upper_bound = mid;  
22   } else {  
23     hi = mid - 1;  
24   }  
25}  
26 let results = [];  
27 for (let i = lower_bound; i <= upper_bound; i++) {  
28   this.data[type][this.keys[type][i]].forEach((item) => results.push(item));  
29}  
30 return { items: results };
```

*Binary search implementation*

# BOYER MOORE

```
● ● ●  
1 class BoyerMoore {  
2     buildBadMatchTable(key) {  
3         const tableObj = {};  
4         const keyLength = key.length;  
5         for (let i = 0; i < keyLength - 1; i++) {  
6             tableObj[key[i]] = keyLength - 1 - i;  
7         }  
8         tableObj[key[keyLength - 1]] = keyLength;  
9         return tableObj;  
10    }  
11  
12    implement(str, pattern) {  
13        const badMatchTable = this.buildBadMatchTable(pattern);  
14        let offset = 0;  
15        const patternLastIndex = pattern.length - 1;  
16        const maxOffset = str.length - pattern.length;  
17  
18        while (offset <= maxOffset) {  
19            let scanIndex = patternLastIndex;  
20            while (scanIndex >= 0 && pattern[scanIndex] === str[offset + scanIndex]) {  
21                scanIndex--;  
22            }  
23            if (scanIndex < 0) {  
24                return offset;  
25            }  
26            const badMatchChar = str[offset + patternLastIndex];  
27            offset += badMatchTable[badMatchChar] || pattern.length;  
28        }  
29        return -1;  
30    }  
31}
```

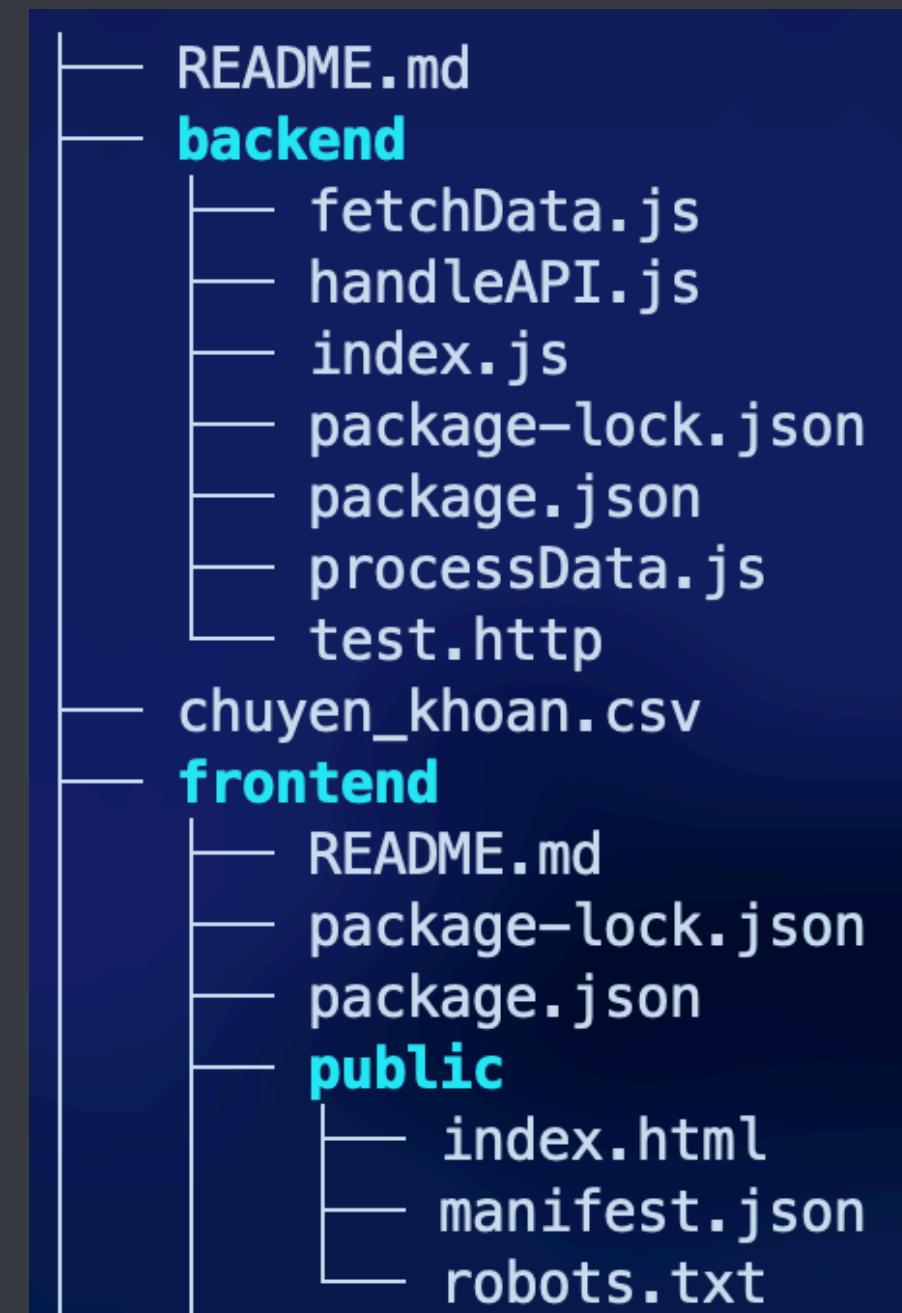
*Class boyer moore*

# CẤU TRÚC FILE JSON

```
4  ▼ {  
5  ▼   "items": [  
6  ▼     {  
7      "data_time": "02/09/2024_5078.73943",  
8      "trans_no": 5,  
9      "credit": 50000,  
10     "debit": 0,  
11     "detail": "MBV р.6925071164.mung ngay quo  
12   },  
13   {  
14     "data_time": "02/09/2024_5216.8149",  
15     "trans_no": 6,  
16     "credit": 2000,  
17     "debit": 0,  
18     "detail": "524322.020924.175952.2091945"  
19   }  
20 ]  
21 }
```

*Query trans\_no từ 5 đến 6*

# CẤU TRÚC SOURCE CODE



# THANK YOU

