



GLC Team 9



Meet the Team!



Drustan Yeo
NUS Y3 CS + Math
(Backend)



Ong Jing Xuan
NTU Y4 REP CS
(Frontend)



Asher Laiu
SMU Y3 Econs + IS
(Backend)



Heng Junxiang
NUS Y3 Comp Eng
(Frontend)



Ong Jung Yi
SUTD Y3 CS
(Backend)

Table of contents

1

Tech Stack

React, Spring Boot, MySQL

2

Live Demo

Pray for us :)))

3

User Stories

In-depth analysis

4

Architecture & Design

Microservices

1

Tech Stack

Tech Stack



Next.js

React.js
TailwindCSS
Material UI



Spring Boot

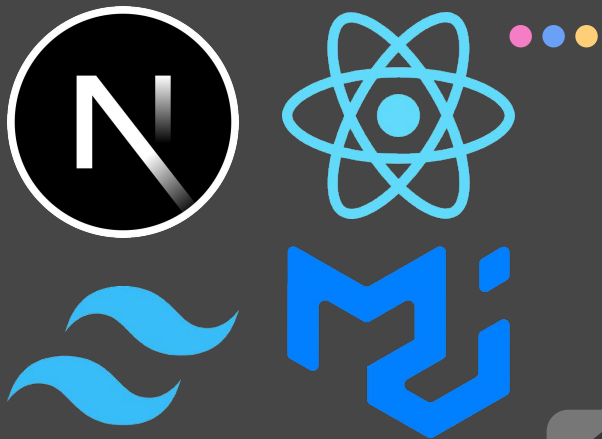
Spring Security
Spring Cloud Gateway
Server Sent Event
Lombok



MySQL

Spring Data JPA
Adminer
Docker

Next.js



1

Declarative UI

2

Component-based

3

Clean Styling

4

Consistency

Spring Boot



1

Spring Ecosystem

2

Embedded Servers

3

**Microservices
Support**

2

Live Demo

3


User Stories

User Story 1 & 2

Analysis: Trader searches for verified instruments and sends request for new instruments that are not present

- System returns exact matches within 3s
- Department validation through authentication
 - Request Form

UI Component



Home

Market Analysis

Execution

Instrument Group

Department

Exchange

Settlement Currency

Instrument

Risk Country

Trade Currency

SEARCH

UI Component



Home



Market Analysis



Execution

Instrument Group

Department

Exchange

Settlement Currency

Instrument

Risk Country

Trade Currency

Submit for Approval

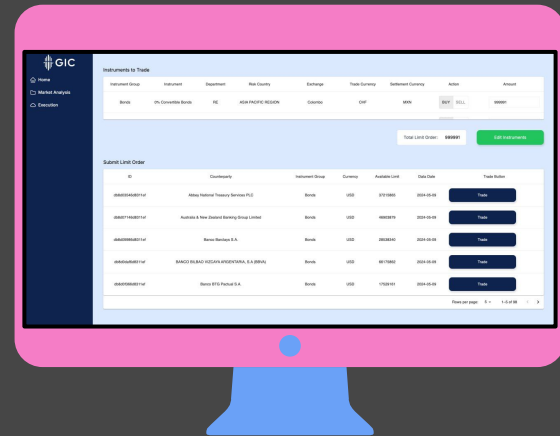


Efficient Querying of DB


- Use only one API call for multiple search fields
 - Reduces API calls to DB

User Story 3 & 4

Execution: Trader uses a reliable
Real-time Limits Dashboard for Executing
Trades Concurrently



UI Component



[Home](#)[Market Analysis](#)[Execution](#)

Instruments to Trade

Instrument Group	Instrument	Department	Risk Country	Exchange	Trade Currency	Settlement Currency	Action	Amount
Bonds	0% Convertible Bonds	RE	ASIA PACIFIC REGION	Colombo	CHF	MXN	<div>BUYSELL</div>	<div>999991</div>

Total Limit Order: 999991

Edit instruments

Submit Limit Order

ID	Counterparty	Instrument Group	Currency	Available Limit	Data Date	Trade Button
db8d03546d8311ef	Abbey National Treasury Services PLC	Bonds	USD	37215865	2024-05-09	Trade
db8d07146d8311ef	Australia & New Zealand Banking Group Limited	Bonds	USD	46903679	2024-05-09	Trade
db8d09986d8311ef	Banco Barclays S.A.	Bonds	USD	28538340	2024-05-09	Trade
db8d0da1fd8311ef	BANCO BILBAO VIZCAYA ARGENTARIA, S.A (BBVA)	Bonds	USD	66175862	2024-05-09	Trade
db8d0f366d8311ef	Banco BTG Pactual S.A.	Bonds	USD	17529161	2024-05-09	Trade

Rows per page: 5 1-5 of 98

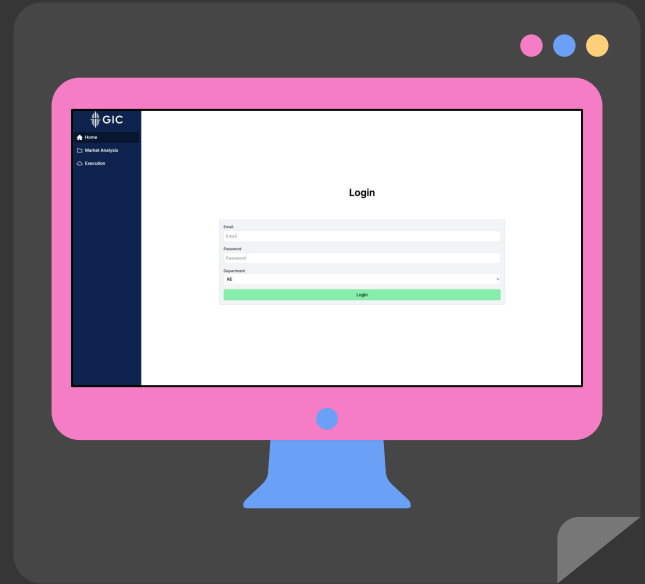


Integration With DB


- SSE To Receive Live Updates
 - Orders Are Queued
- Backend Check To Prevent Overuse

User Story 5

Integrated Platform: All-in-one webapp
for Traders



UI Component



Home

Market Analysis

Execution

Login

Email

Password

Department

RE

Login

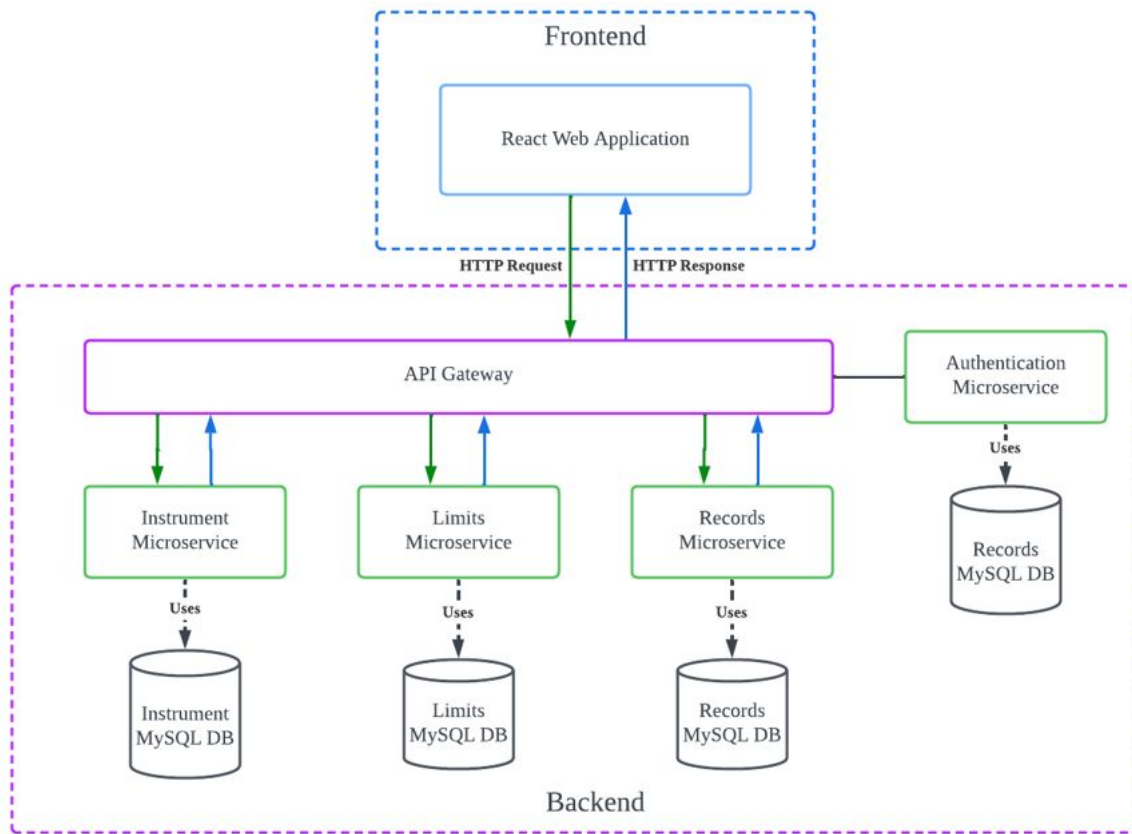
4

Architecture & Design

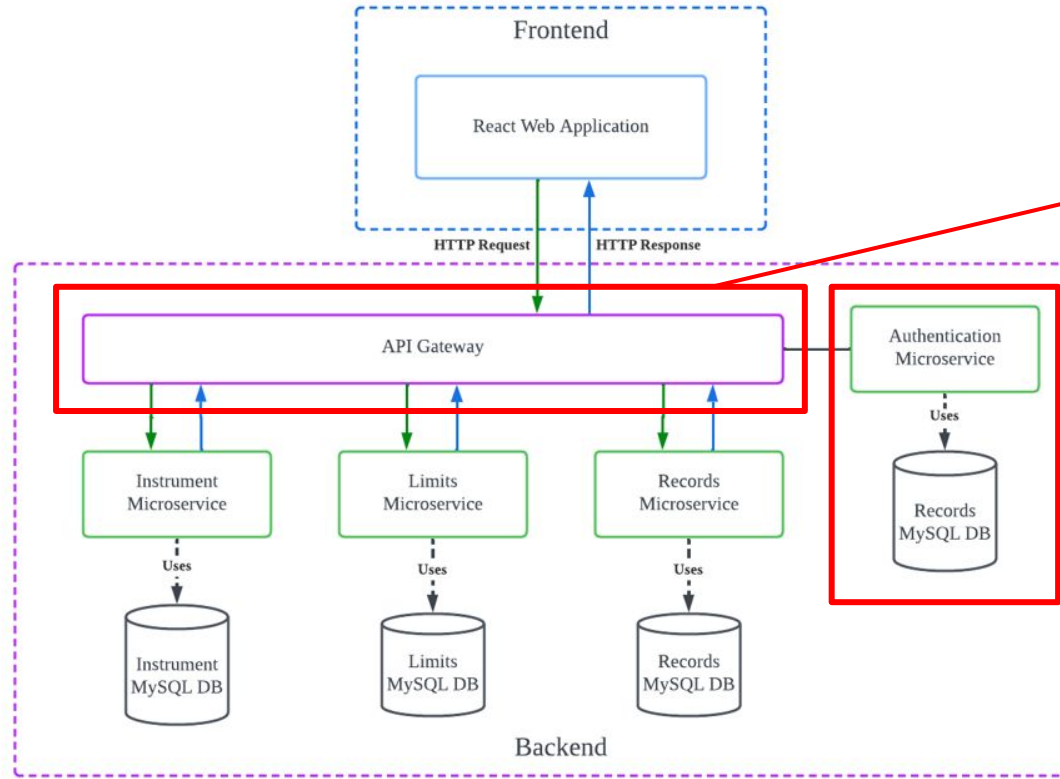
What is Microservices Architecture?

- Microservices architecture breaks down an application into **independent, modular services** that each handle a specific functionality.
- Each service is **developed, deployed, and scaled independently**, while communicating with others through APIs.

System Architecture Design



System Architecture Design



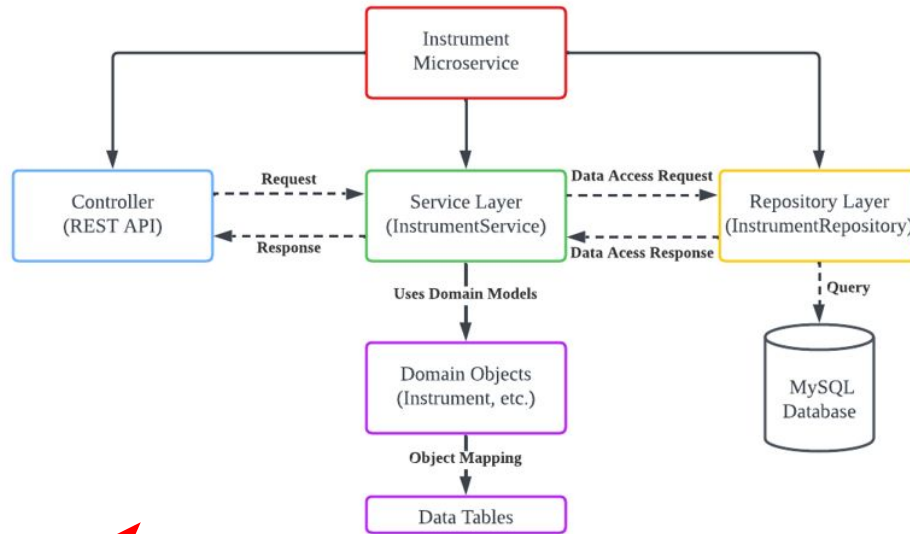
API Gateway:

Centralized routing point, handling authentication and traffic distribution to individual services.

Authentication

Microservice: Secures the system using JWT, applied once for all services through the API gateway.

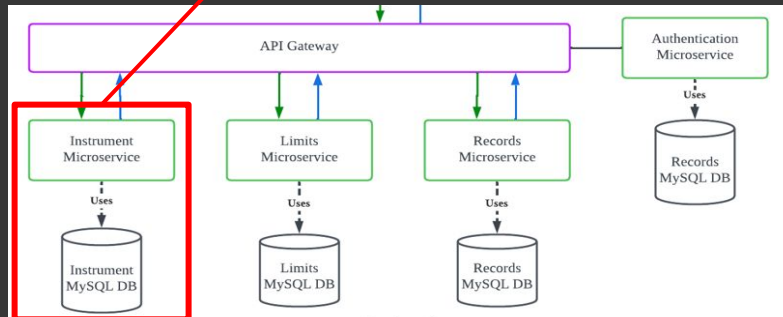
Instrument Microservices Design

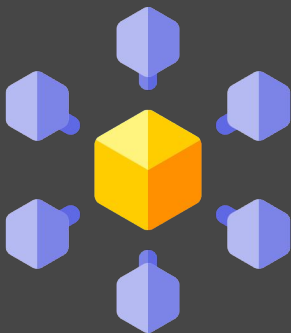


Instrument Microservice: Handles instrument search and queries.

Limits Microservice: Manages limit tracking and trade execution.

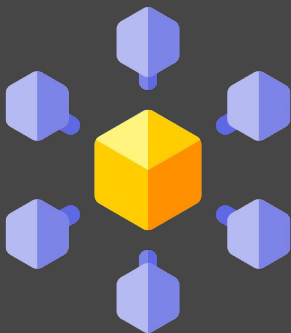
Records Microservice: Logs trade approvals and user actions.





Pros

- **Scalability:** Each service can be scaled independently based on usage demand.
- **Flexibility:** We can deploy and update services without affecting the entire system.
- **Resilience:** If one service fails, others continue to function, reducing downtime.



Cons

- **Complexity:** Managing multiple services introduces complexity in terms of deployment, monitoring, and communication between services.
- **Overhead:** Requires additional infrastructure and careful management of service-to-service communication.

Frontend Code Design

Instrument Group

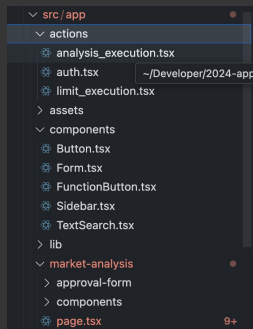
🔍 Enter search here

```
{Object.entries(filters).map((filter : {string, string} , idx : number ) => {  
  const label: string = filter[0]  
  const input: string = filter[1]  
  
  return (  
    <li key={idx}>  
      <TextSearch  
        label={label}  
        text={input}  
        handleChange={(label === "Instrument") ? handleInstrumentChange  
      />  
    </li>  
  )  
})}
```

Component-Based Architecture

Reusable, self-contained UI components that manage their own logic
System is modular and maintainable

Frontend Code Design



Separation of Concerns

- Divides UI components into two types:
 - Container: Handles data fetching, logic, and state management.
 - Presentation: Receives data via props and focuses solely on UI display.
- Keeps code clean and consistent

```
12
13 export default function RootLayout({
14   children,
15 }: Readonly<
16   children: React.ReactNode;
17 >): {
18   return (
19     <html lang="en">
20       <body className={`flex flex-row flex-start ${inter.className}`}>
21         <div className="w-64 min-h-screen">
22           <Sidebar/>
23         </div>
24         {children}
25       </body>
26     </html>
27   );
28 }
```

Code Reuse and Consistent Styling

- Reusable components
- Nextjs layout tools
- Tailwindcss configuration
- Material design guidelines

THANK YOU!