



MICROSERVICES 101

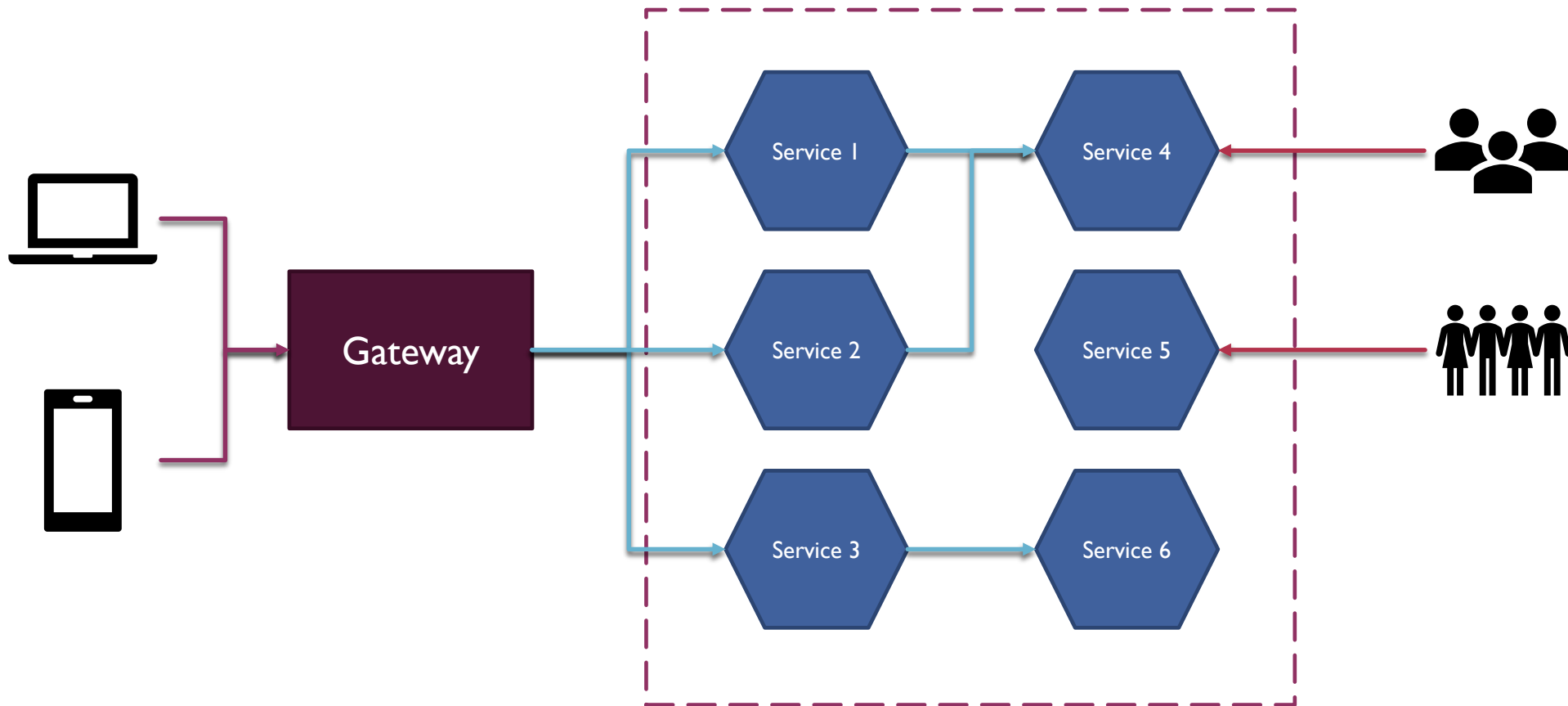
ENG. ISMAIL ANJRINI

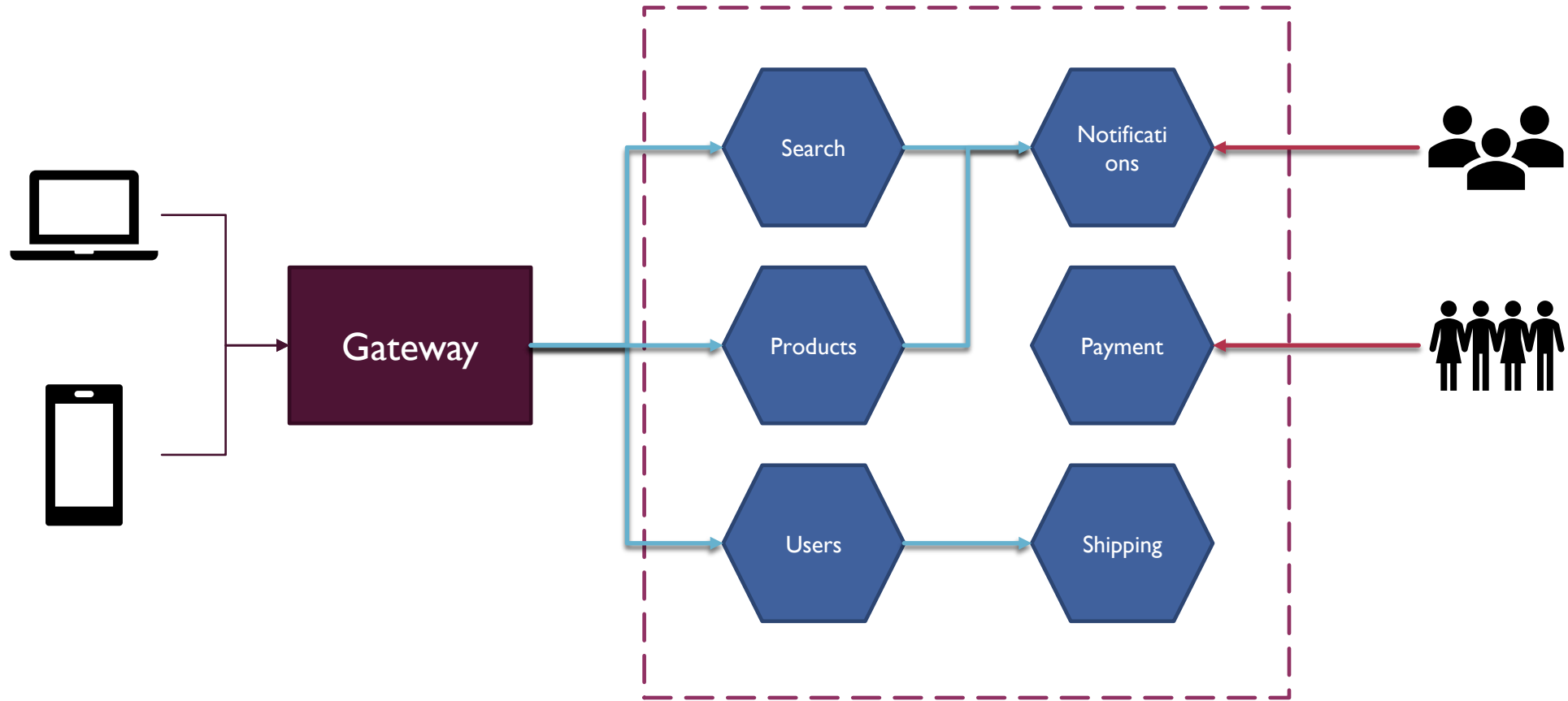


MICROSERVICE ARCHITECTURE STYLE

- Microservices are small, independent, and loosely coupled
- A single small team of developers can write and maintain a service
- Each service is a separate codebase, which can be managed by a small development team
- A team can update an existing service without rebuilding and redeploying the entire application
- Services are responsible for persisting their own data or external state
- Services communicate with each other by using well-defined APIs
- Supports polyglot programming
- Organized around Business Capabilities

MICROSERVICE ARCHITECTURE STYLE







“microservice” is a label, not a description

Martin Fowler

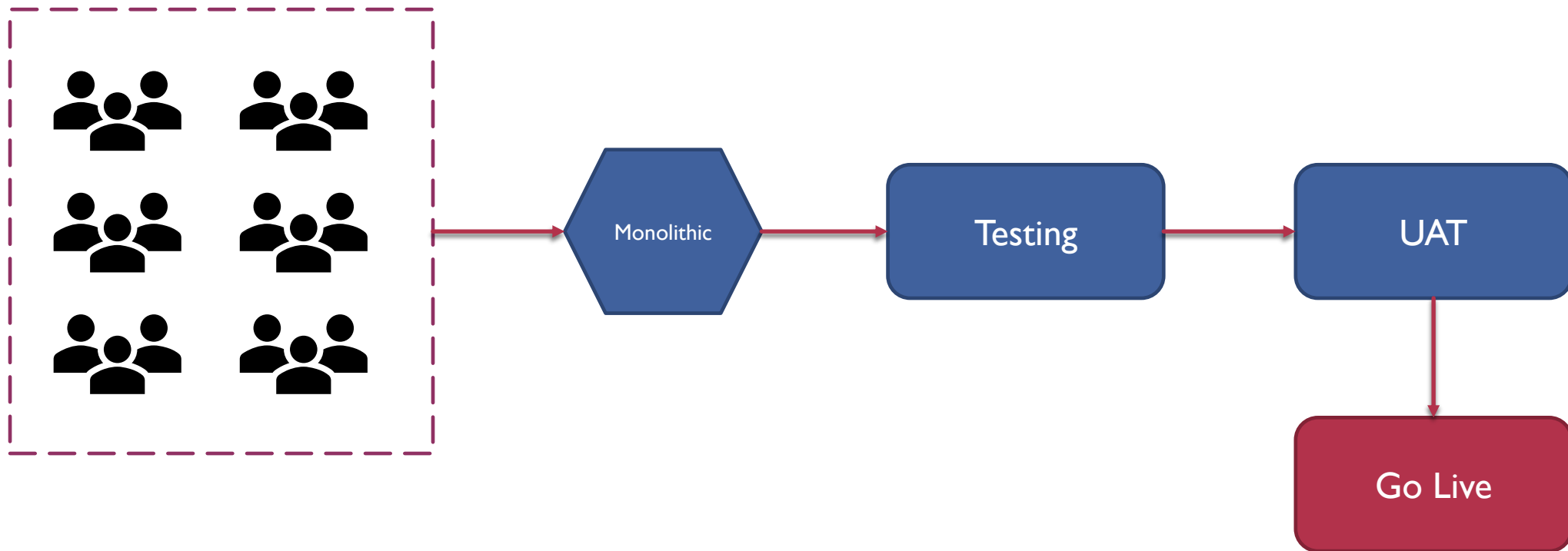


MICROSERVICES ADVANTAGES

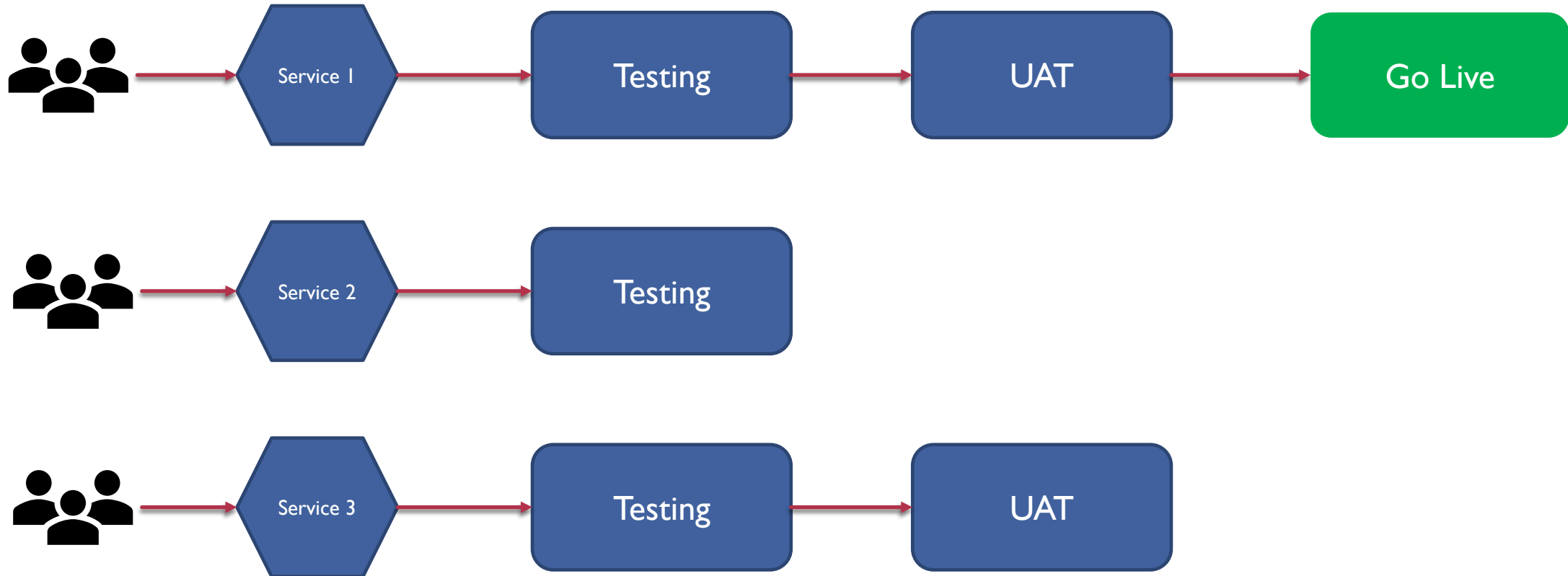
ENG. ISMAIL ANJRINI



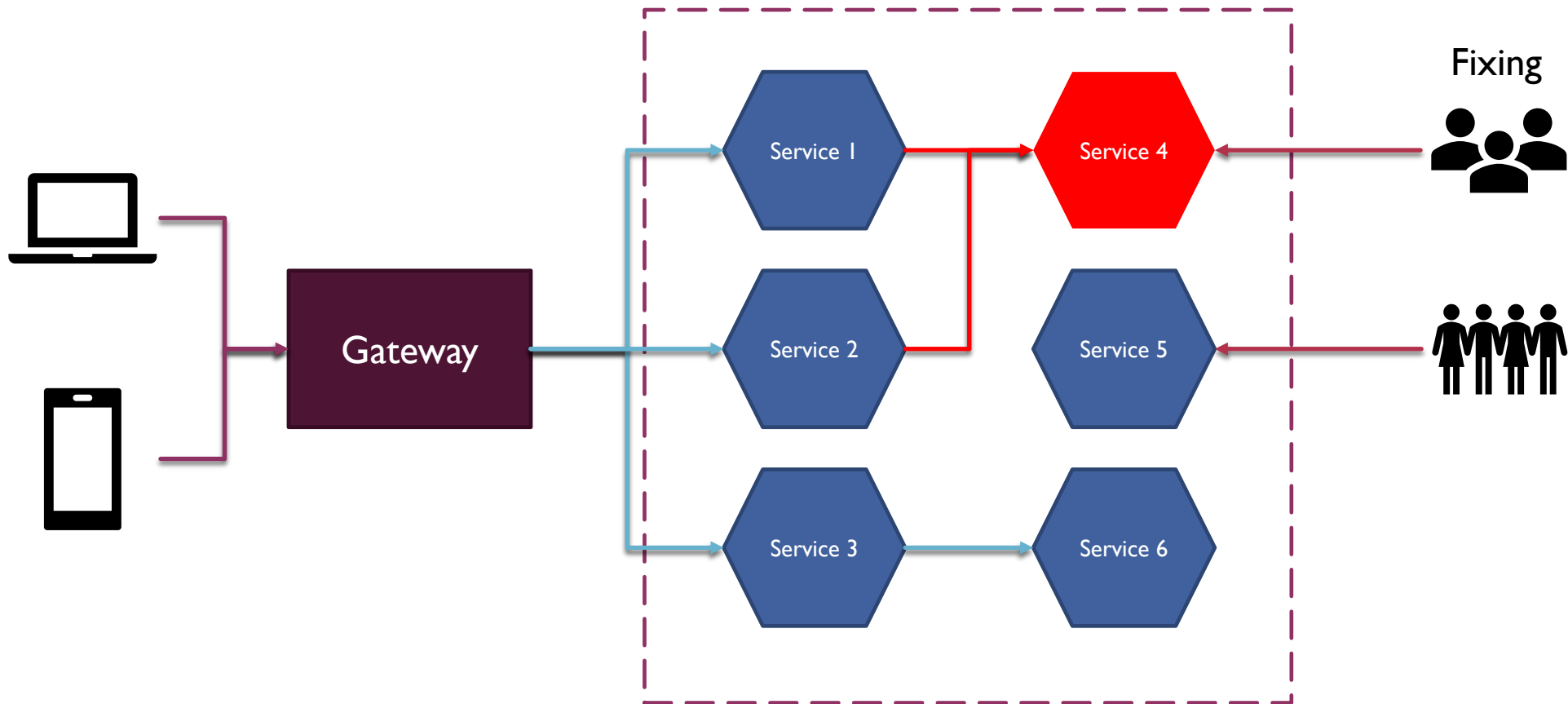
AGILITY



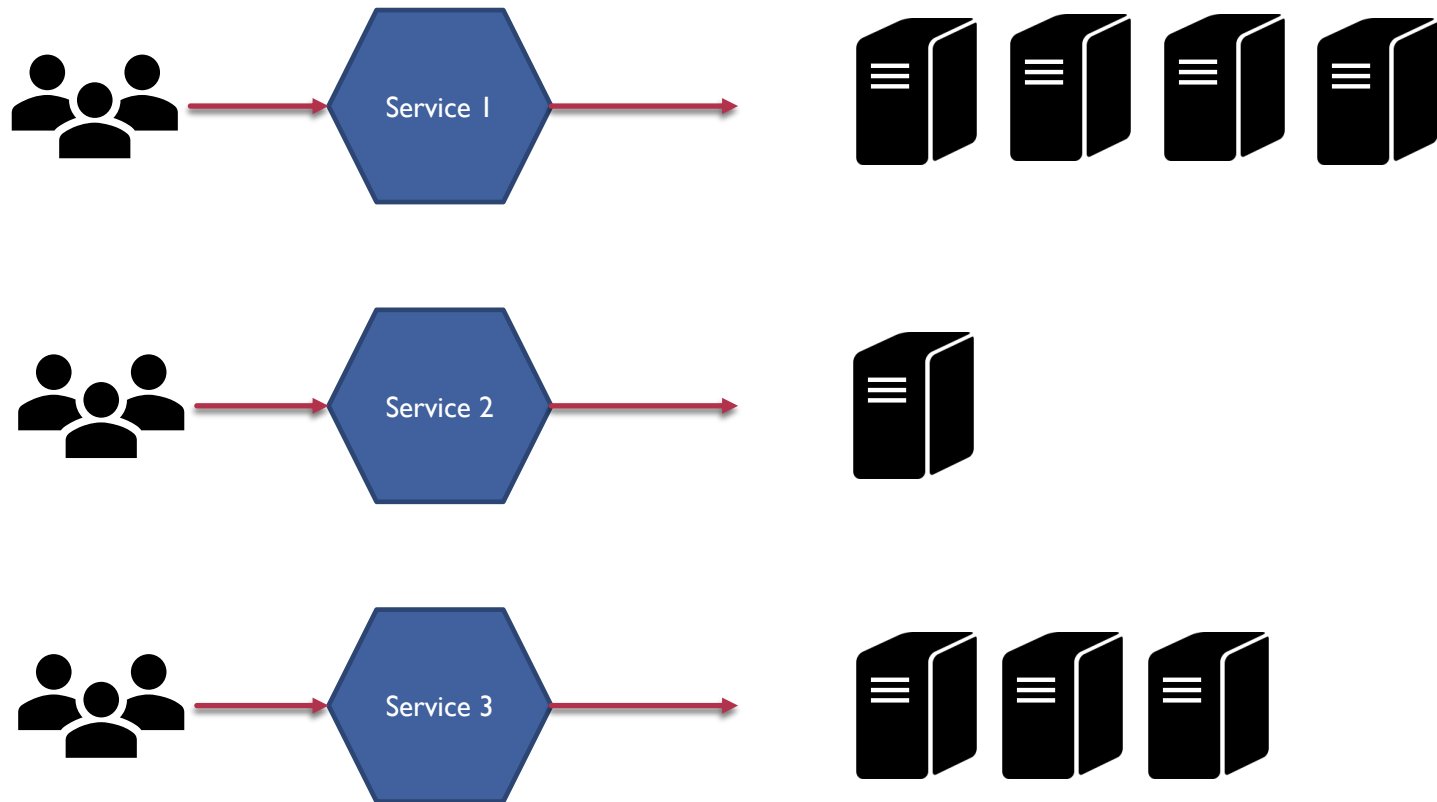
AGILITY (CONT.)



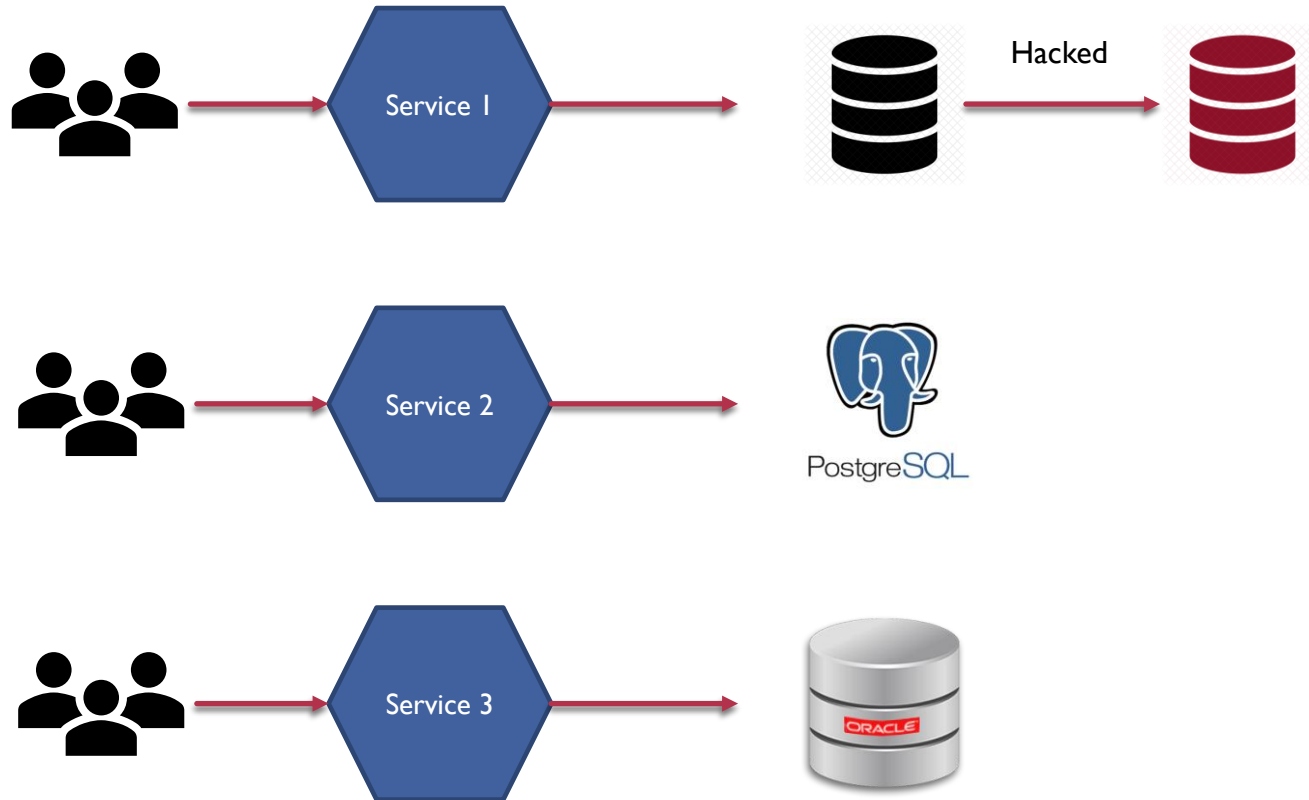
FAULT ISOLATION



SCALABILITY



DATA ISOLATION



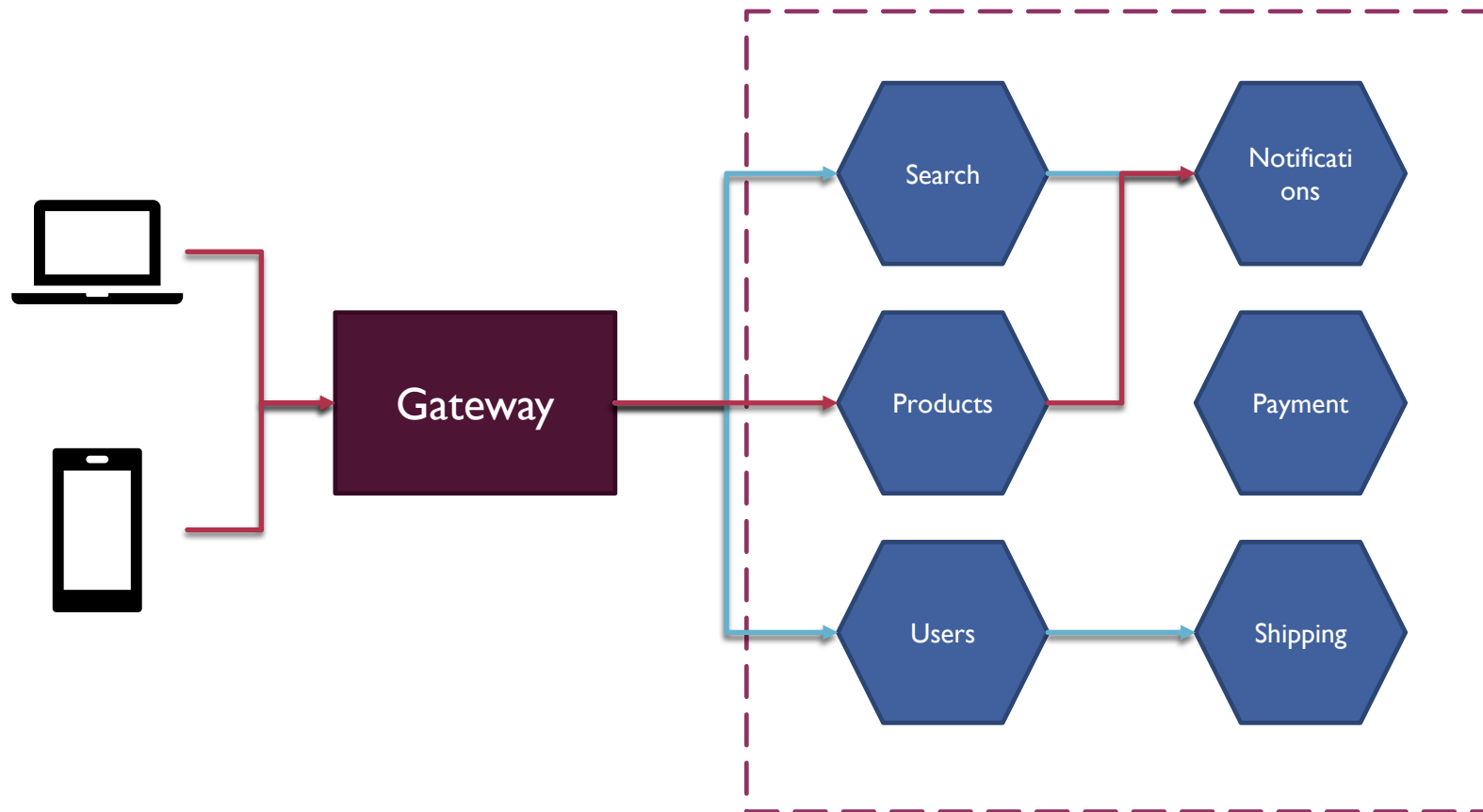


MICROSERVICES CHALLENGES

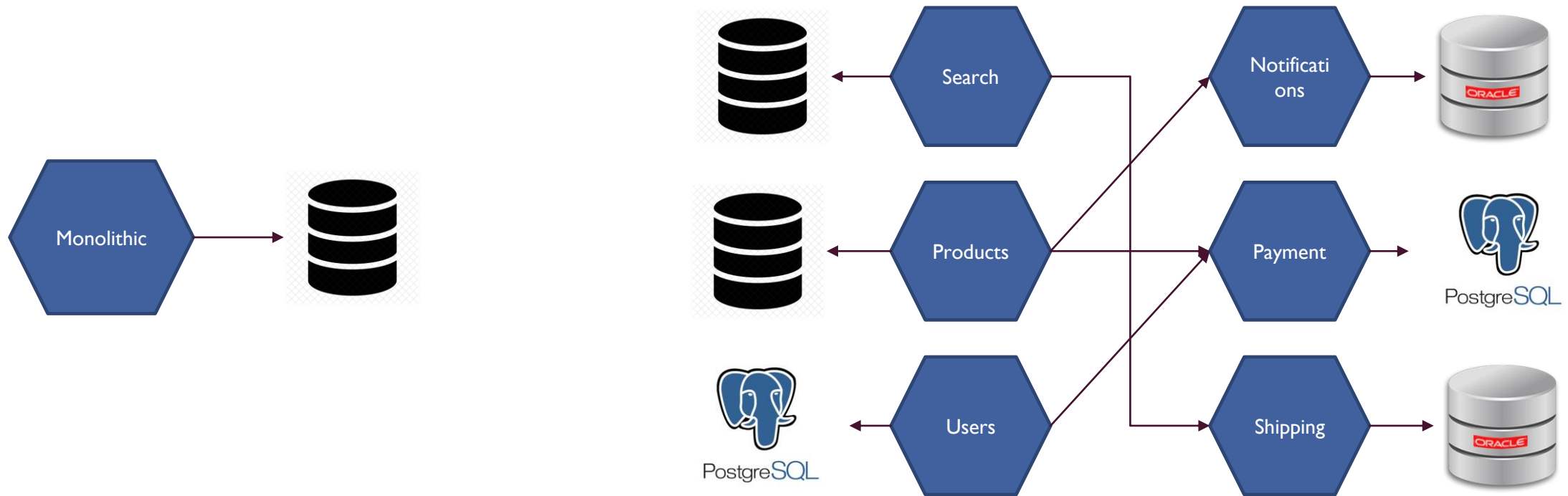
ENG. ISMAIL ANJRINI



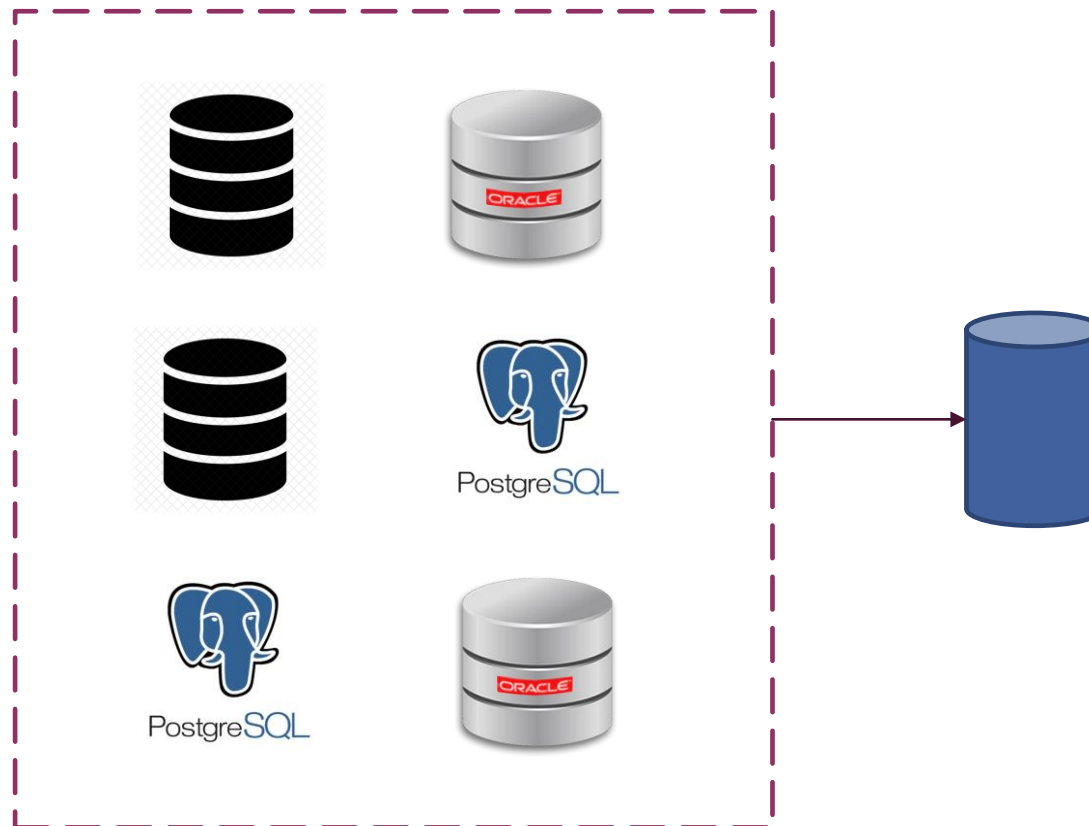
OBSERVABILITY



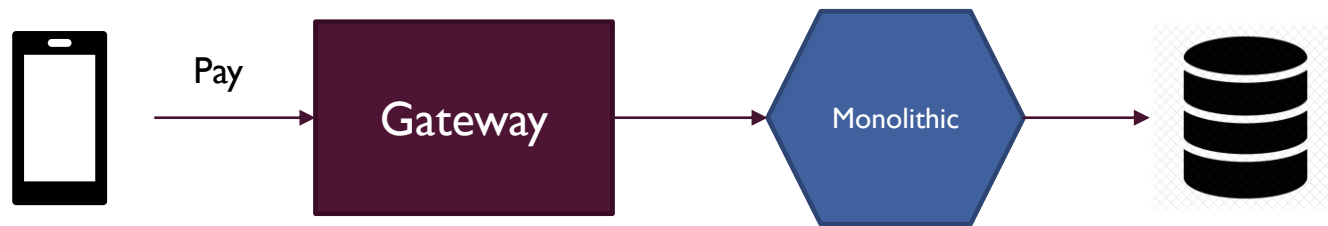
COMPLEXITY



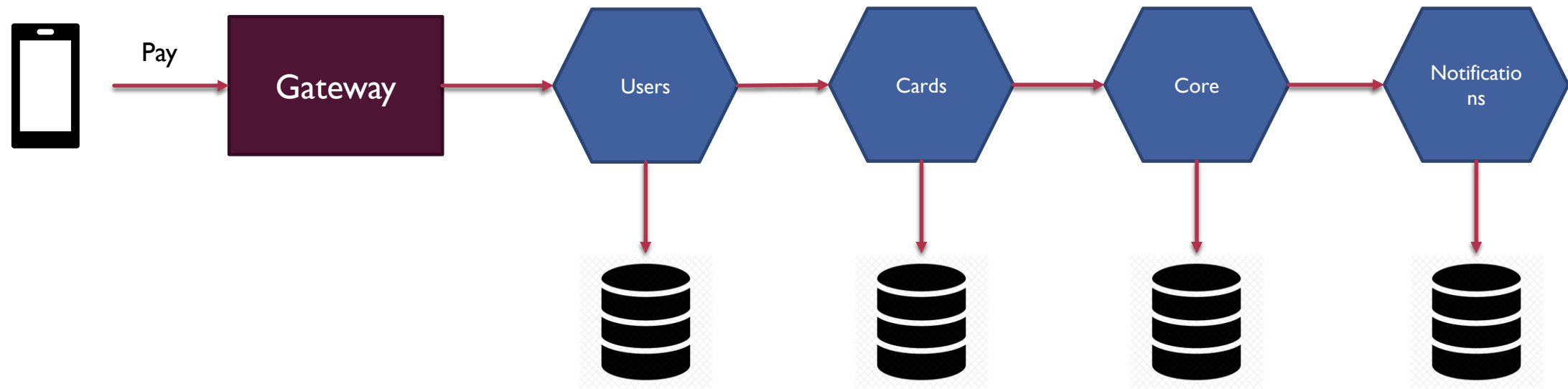
DATA INTEGRITY



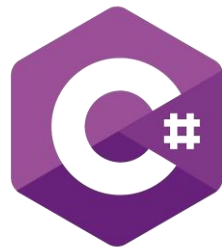
PERFORMANCE



PERFORMANCE (CONT.)



GOVERNANCE

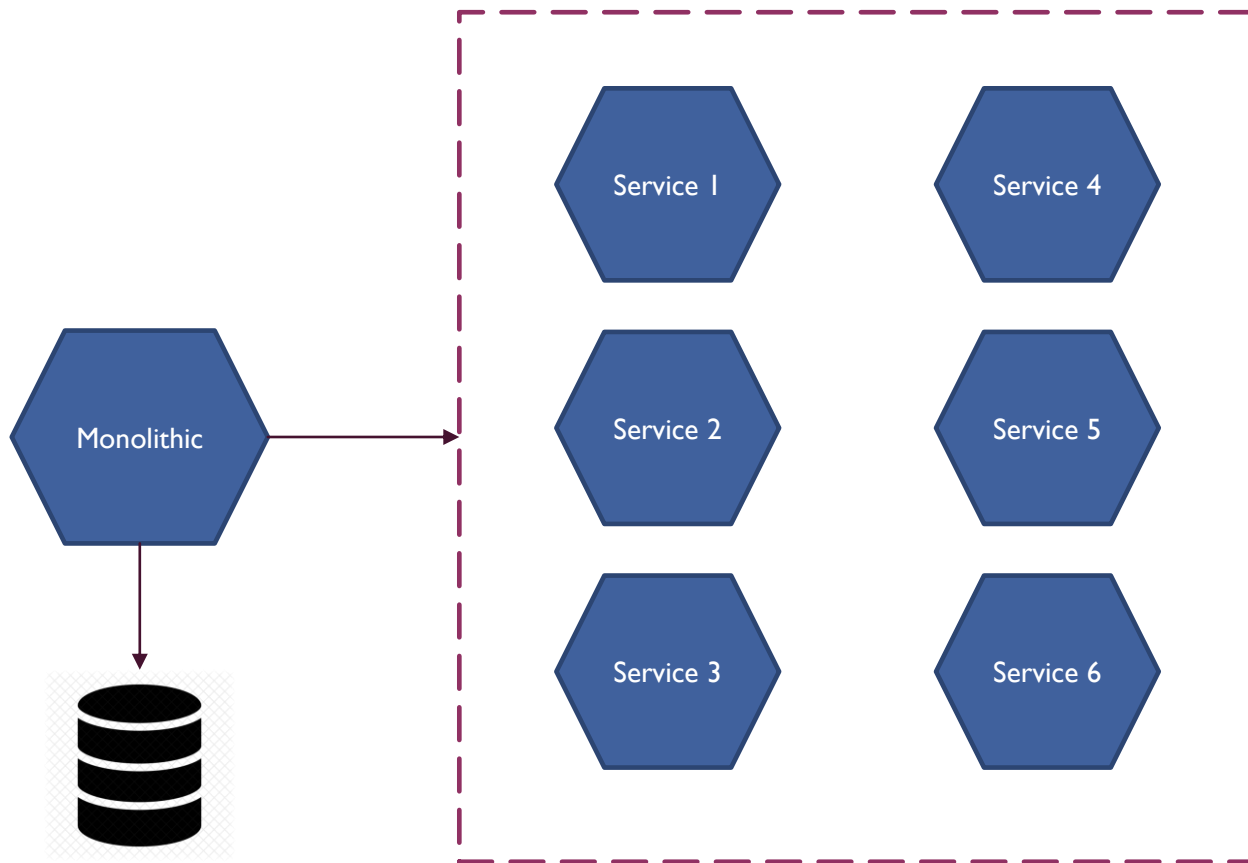




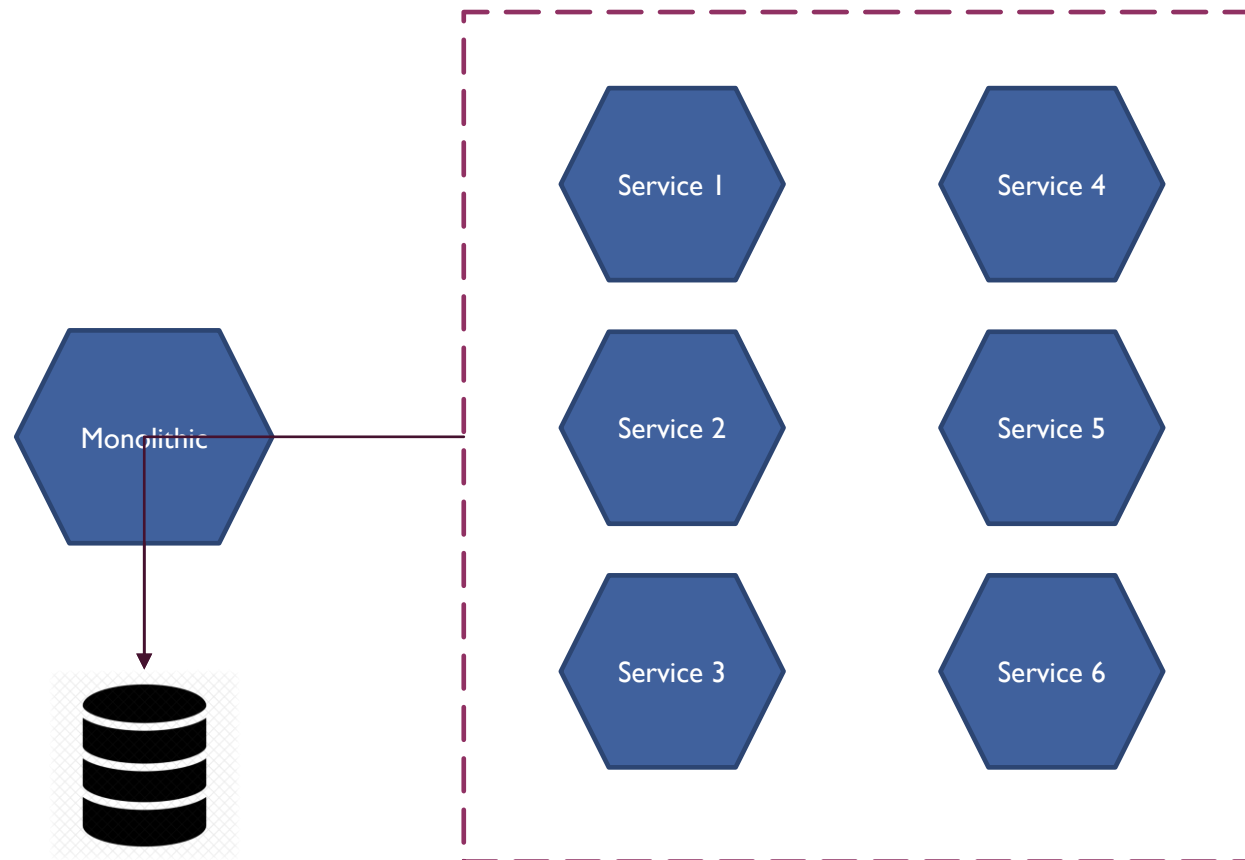
FROM MONOLITHIC TO MICROSERVICES

ENG. ISMAIL ANJRINI

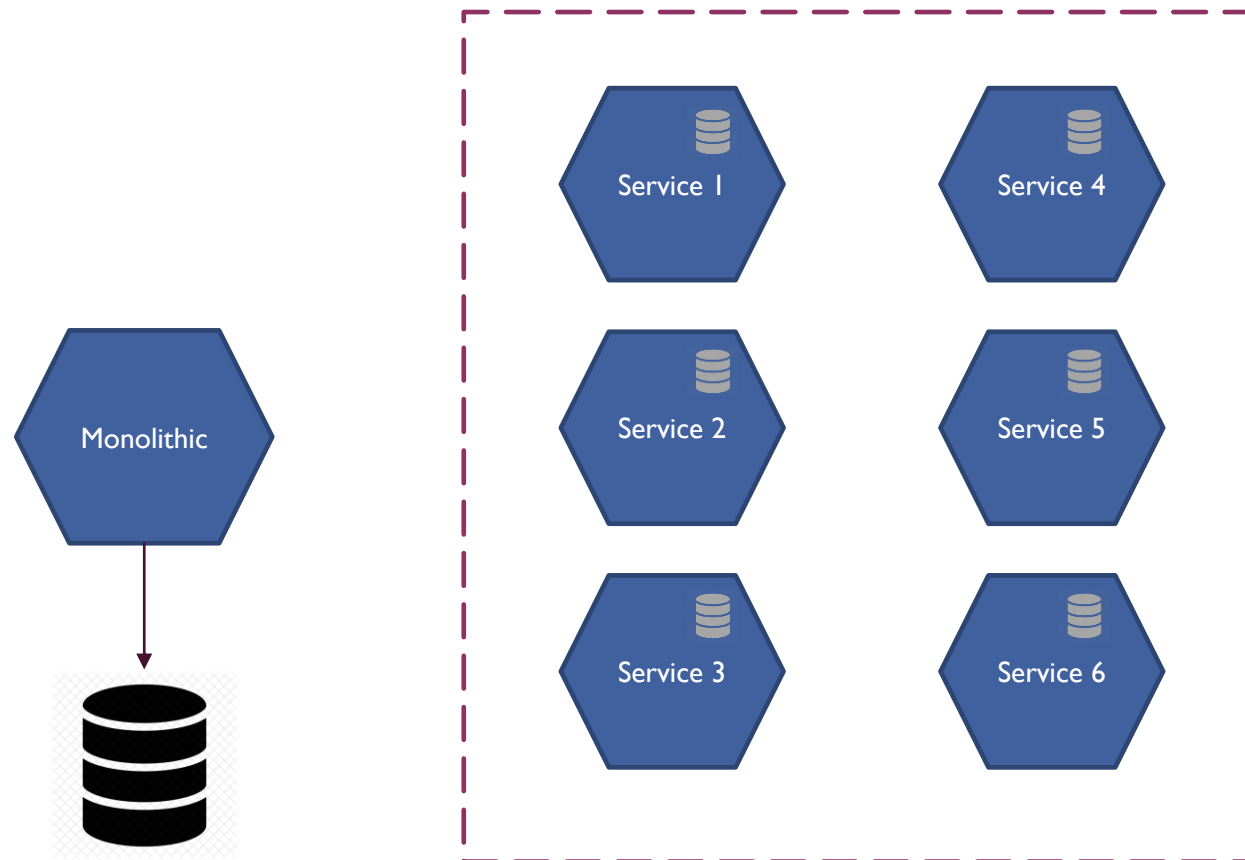




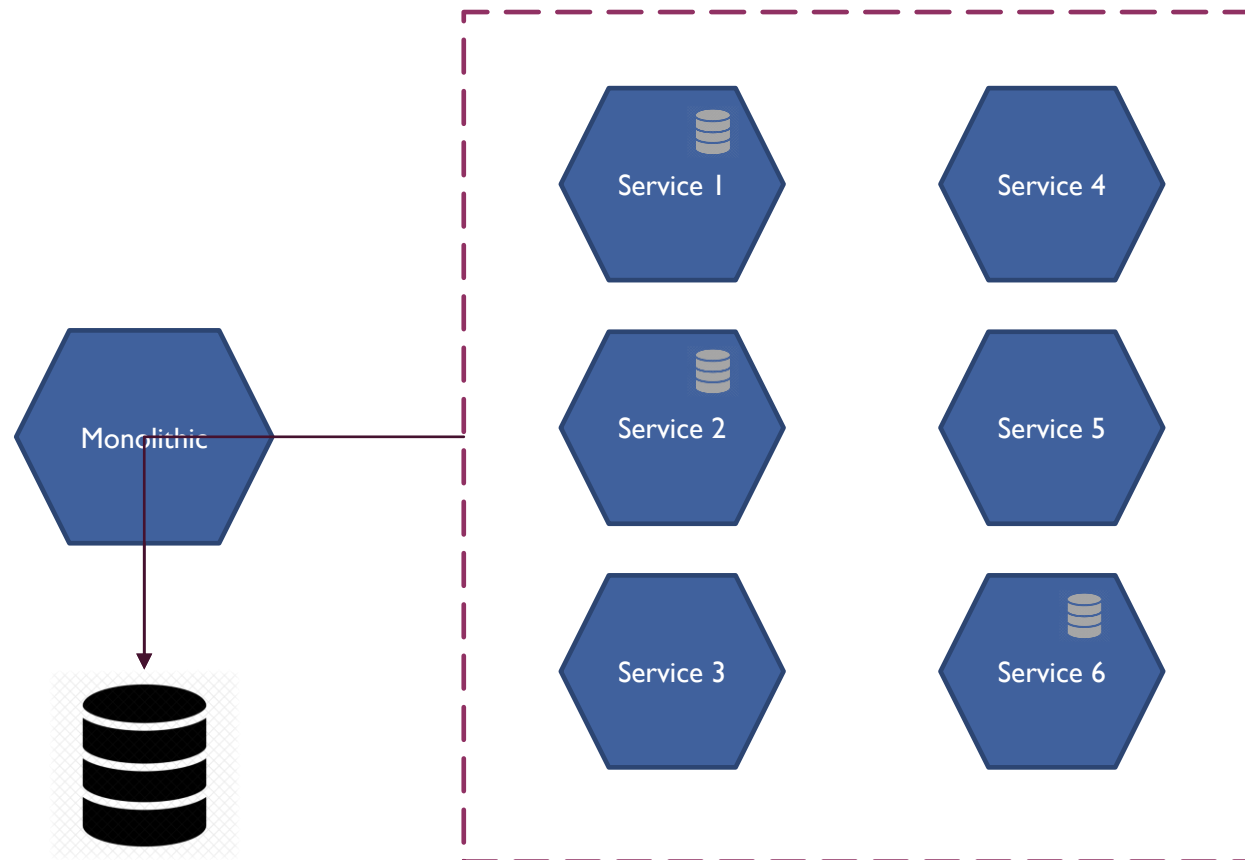
SHARED DATABASE



DEDICATED DATABASE



HYBRID DATABASE MODEL

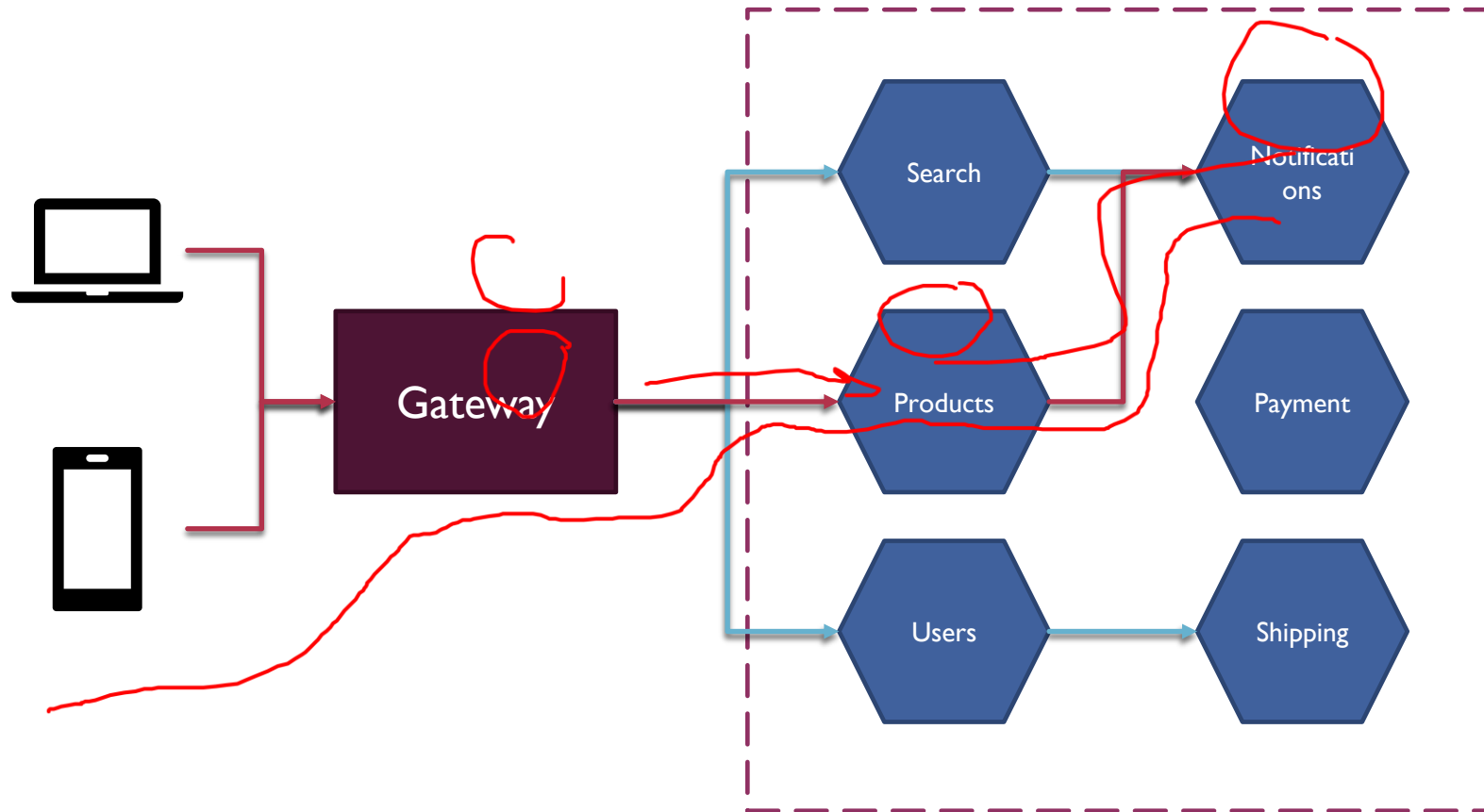




MICROSERVICES - OBSERVABILITY

ENG. ISMAIL ANJRINI





DISTRIBUTED TRACING

- A trace is a group of transactions with a common root
- When a trace travels through multiple services, it is known as a distributed trace
- Distributed tracing enables you to analyze performance throughout your microservice architecture

APPLICATION PERFORMANCE MONITORING

- Application performance monitoring (APM) is the collection of tools and processes
- Designed to help IT professionals ensure that enterprise applications meet the performance
- Application performance monitoring platform should focus on infrastructure monitoring, as well as tracking the user experience
- APM tools provide administrators with the data they need to quickly discover, isolate and solve problems





















5.16s Response time
5.16s Processing time

Execution breakdown

82 % Other
18 % CPU

🔍 Search name, url, sql, attribute,...

- ✓  /cart
_80,443
 - ✓  /cart
Hipster Shop Frontend - Production
 -  **GetSupportedCurrencies**
hipstershop.CurrencyService
 - ✓  /hipstershop.CartService/GetCart
cartservice
-
-  **GetCart**
hipstershop.CartService
-
-  **ListRecommendations**
hipstershop.RecommendationService on server frontend-* (grpc://hipstersho ...)
 -  **GetProduct**
hipstershop.ProductCatalogService
 -  **GetProduct**
hipstershop.ProductCatalogService
 - ✓  **invoke**
user-info-service in us-east-1
 - ✓  **DynamoDB.Scan**

 <https://dynamodb.us-east-1.amazonaws.com/>
Requests to public networks
 - ✓  /
shippingholidaysservice-*
 -  /
PHP on shippingholidaysservice-*
 - ✓  <http://translationservice:8111/>
flask translationservice-* server on port 8111
 -  /
translationservice-api
 -  **GetAds**
hipstershop.AdService
 -  **GetQuote**
hipstershop.ShippingService
 -  **Convert**
hipstershop.CurrencyService

GetCart



Summary

Timing

Code level

Logs

Errors

Filter log by content or status

Timestamp	Status	Content	Details
2021-12-12, 23:59:51	NONE	{"@t":"2021-12-12T22:59:51.2152709Z","@m":"Slow GetCartAsync request detected for userId=d777fe05-6a37-4914-83d4-39598440117f","@i":"200339eb","trace_id":"b48fde53b13b9121b163b8494daa389a","span_id":"00000000C	✓
2021-12-12, 23:59:51	NONE	{"@t":"2021-12-12T22:59:51.2152275Z","@m":"GetCartAsync called with userId=d777fe05-6a37-4914-83d4-39598440117f","@i":"9c15b2c0","trace_id":"b48fde53b13b9121b163b8494daa389a","span_id":"000000000	✓

[View logs for this trace](#)[View process logs](#)

- Projects
- Issues
- Discover
- Performance
- Alerts
- Releases
- User Feedback

- Dashboards
- Activity
- Stats
- Settings

- Help
- What's new
- Collapse

Performance > Trace View

Open in Discover

Trace ID: 7u08dfghfh7dfhh

Event Breakdown ?

18 Transactions | 2 Errors

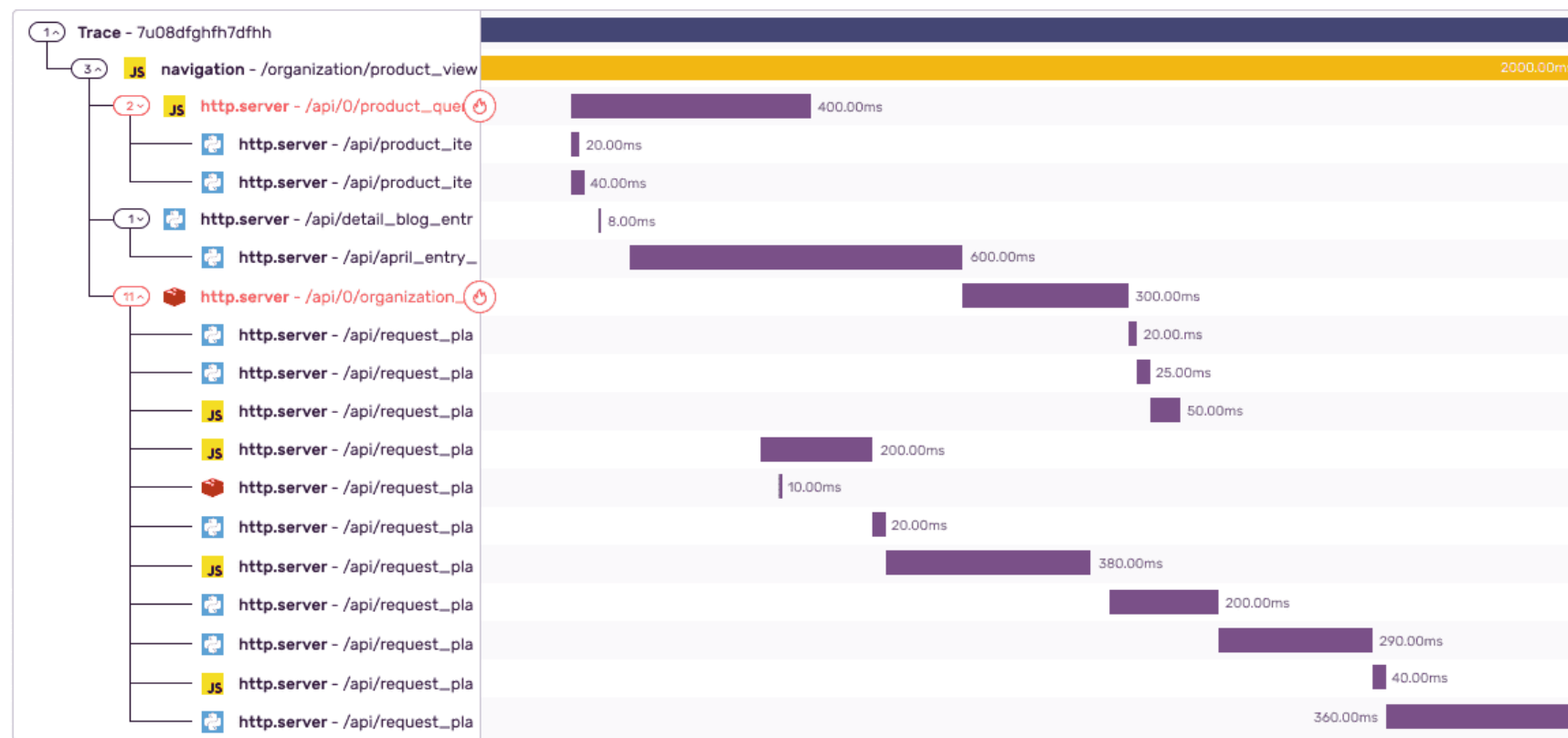
Across 3 projects

Total Duration ?

2000ms

14 hours ago

Search transactions



Trace sample < 2 of 2 >

Actions ▾

View full trace

7 minutes ago | 999 ms (100% of trace)

http://opbeans-frontend:3000/dashboard

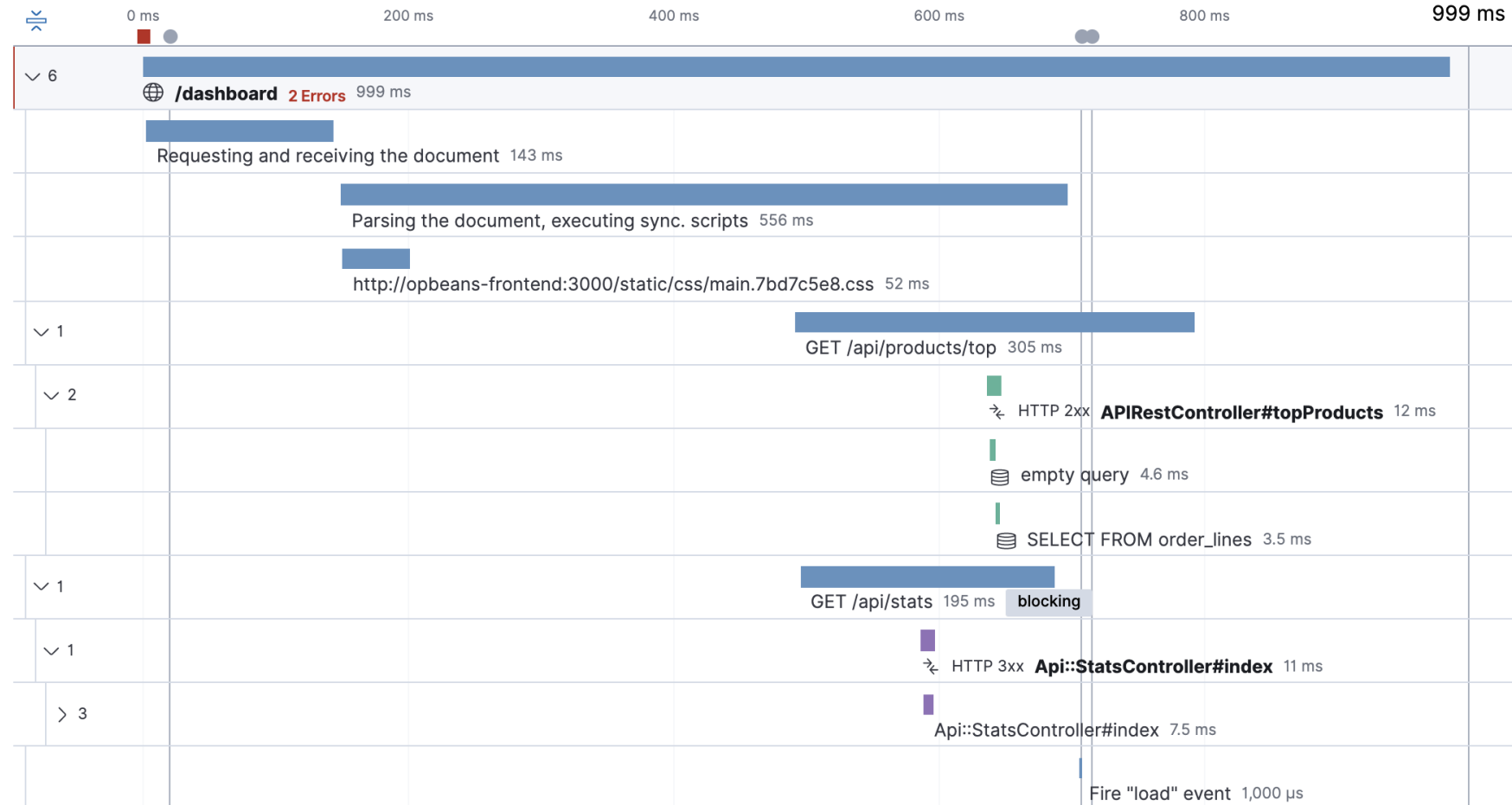
2 Errors

Electron (2.0.18)

Timeline

Metadata

Services ● opbeans-rum ● opbeans-java ● opbeans-ruby

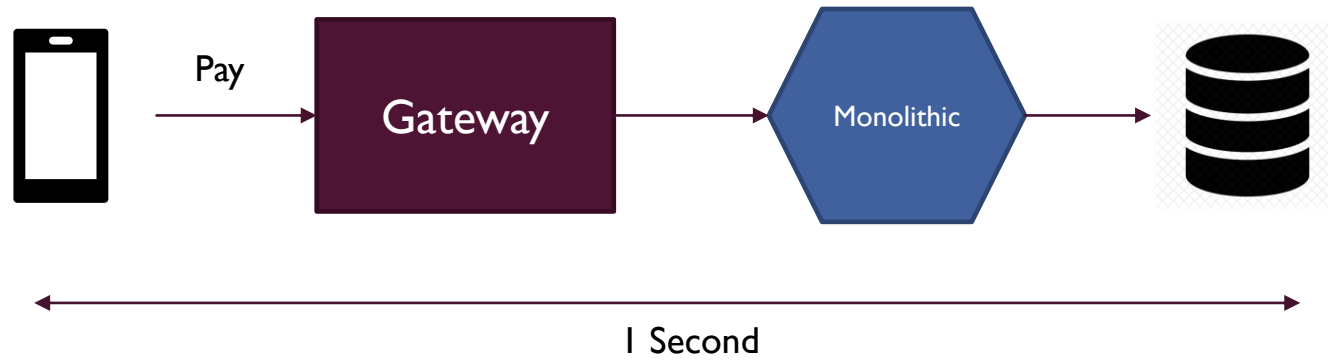


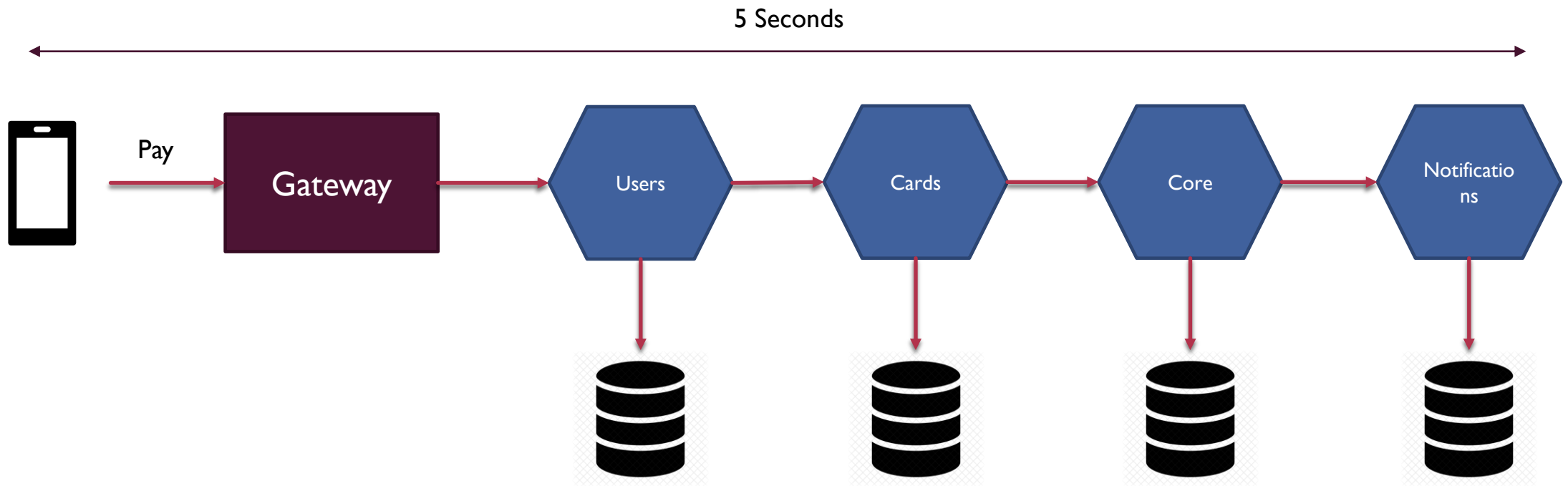


MICROSERVICES - PERFORMANCE

ENG. ISMAIL ANJRINI







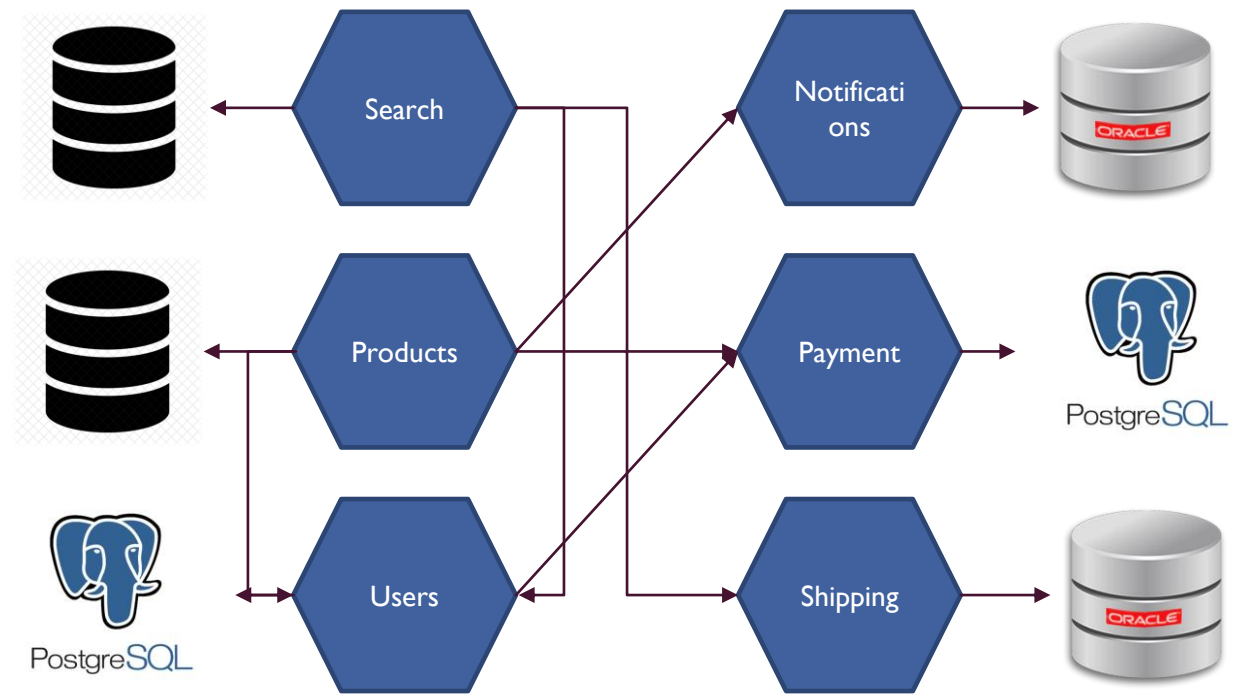
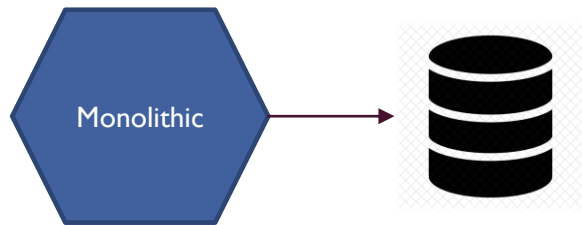




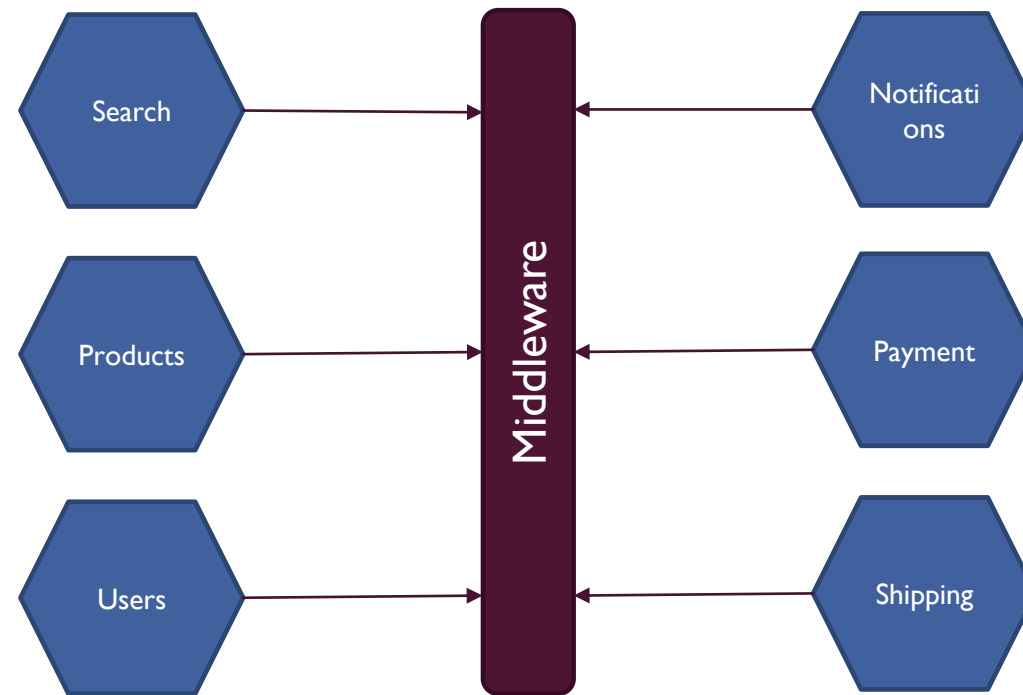
MICROSERVICES - COMPLEXITY

ENG. ISMAIL ANJRINI

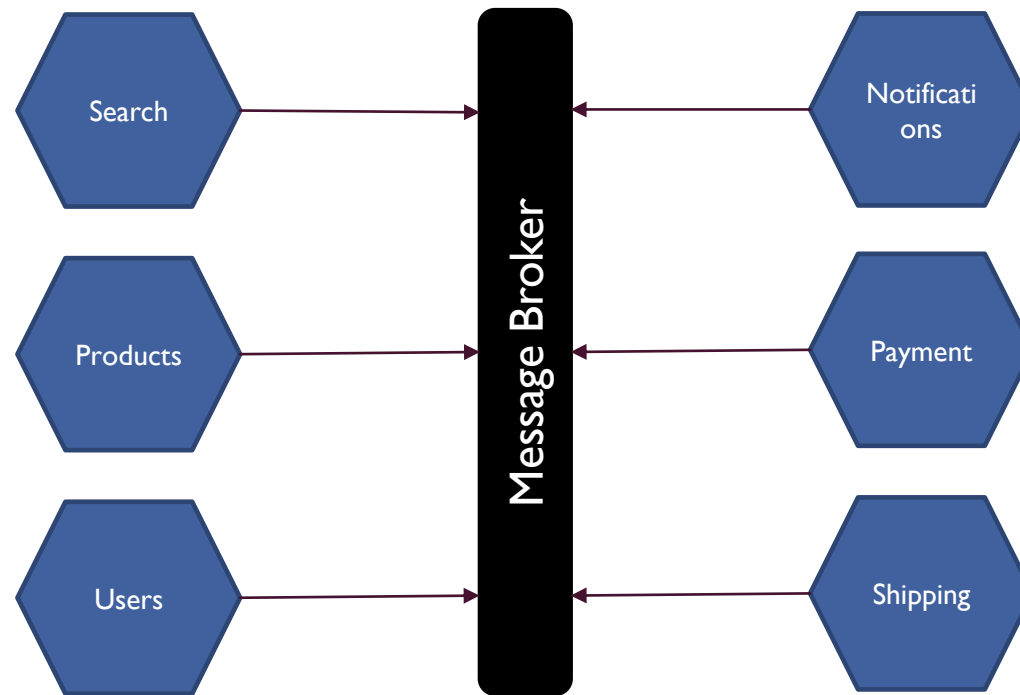




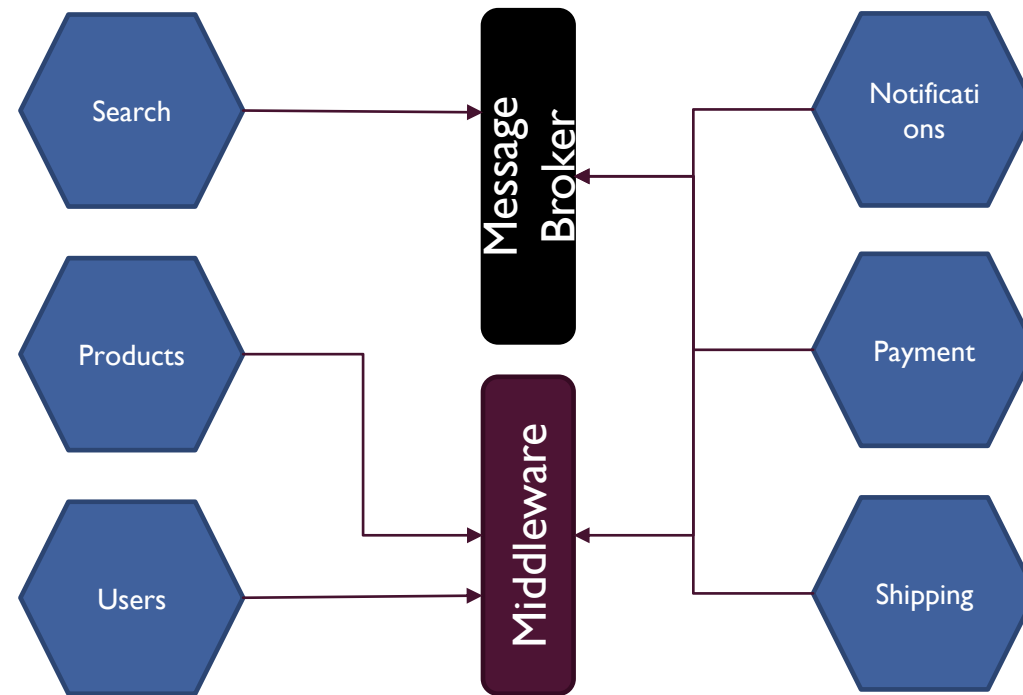
COMMUNICATION LAYER



COMMUNICATION LAYER (CONT.)



COMMUNICATION LAYER (CONT.)



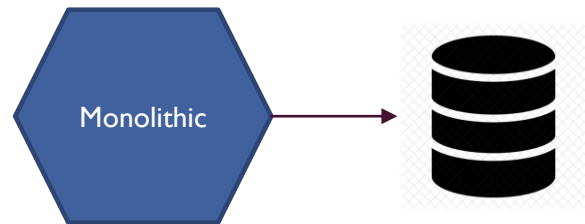




MICROSERVICES – DATA ACCESS

ENG. ISMAIL ANJRINI





Search

Notifications



Products

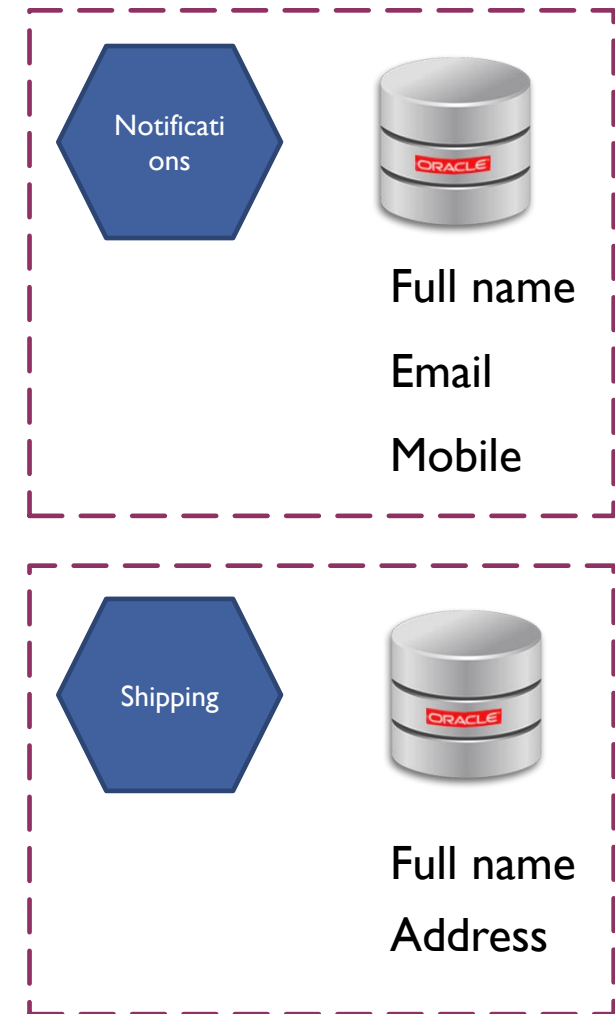
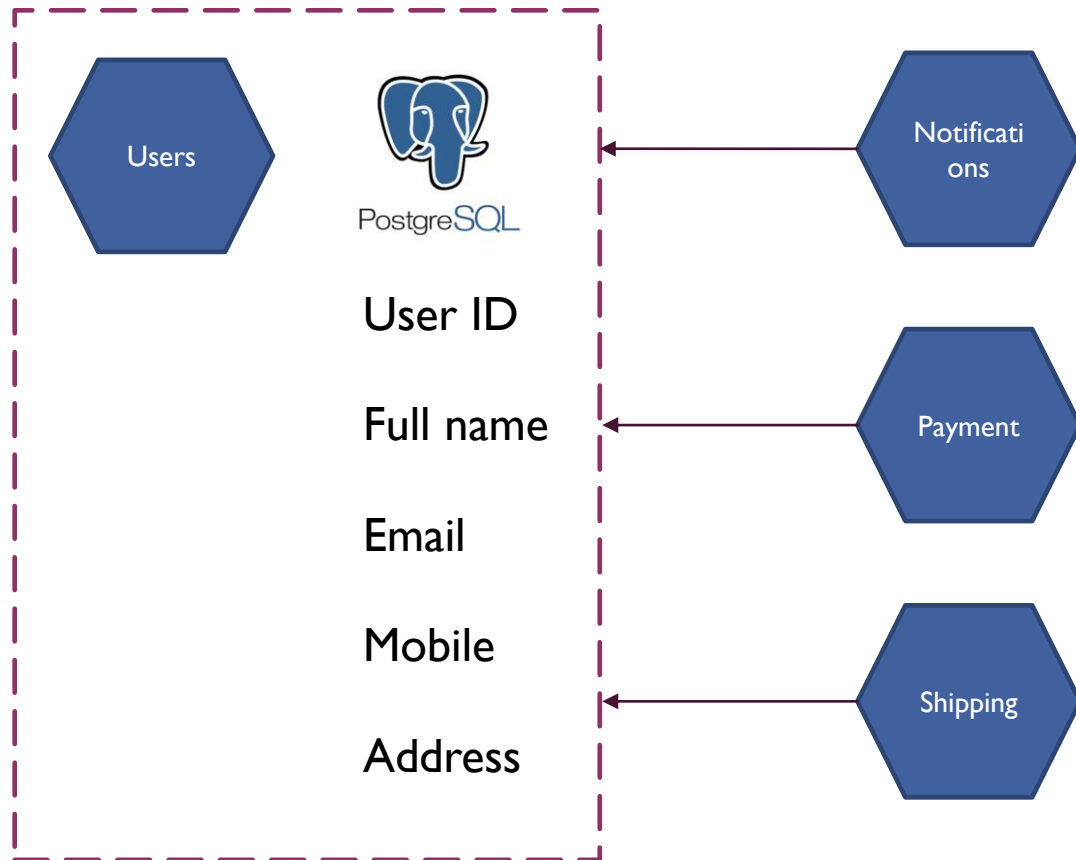
Payment

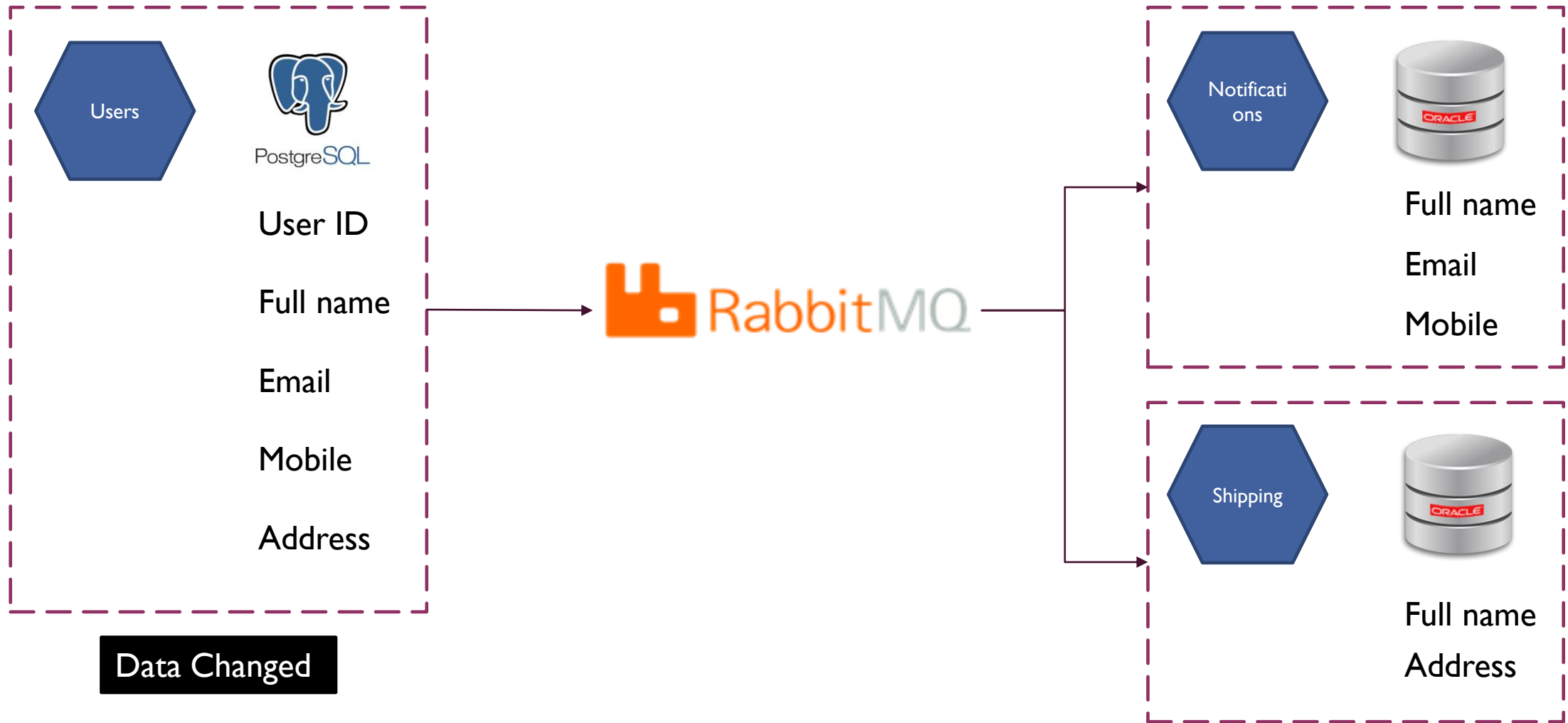


Users

Shipping





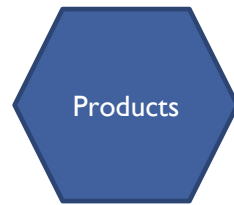


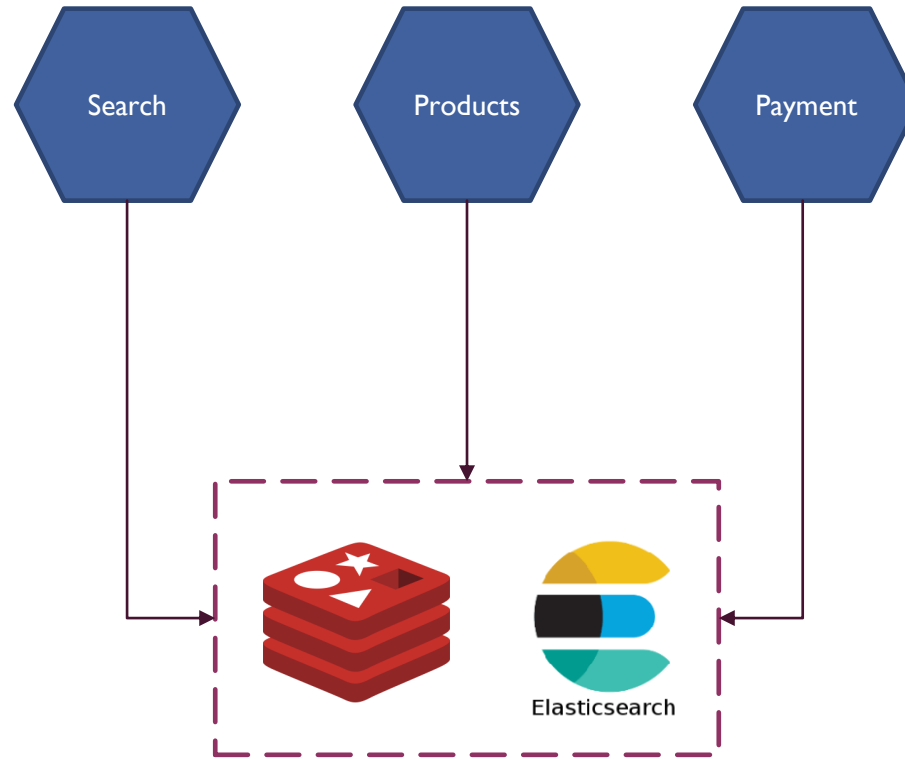


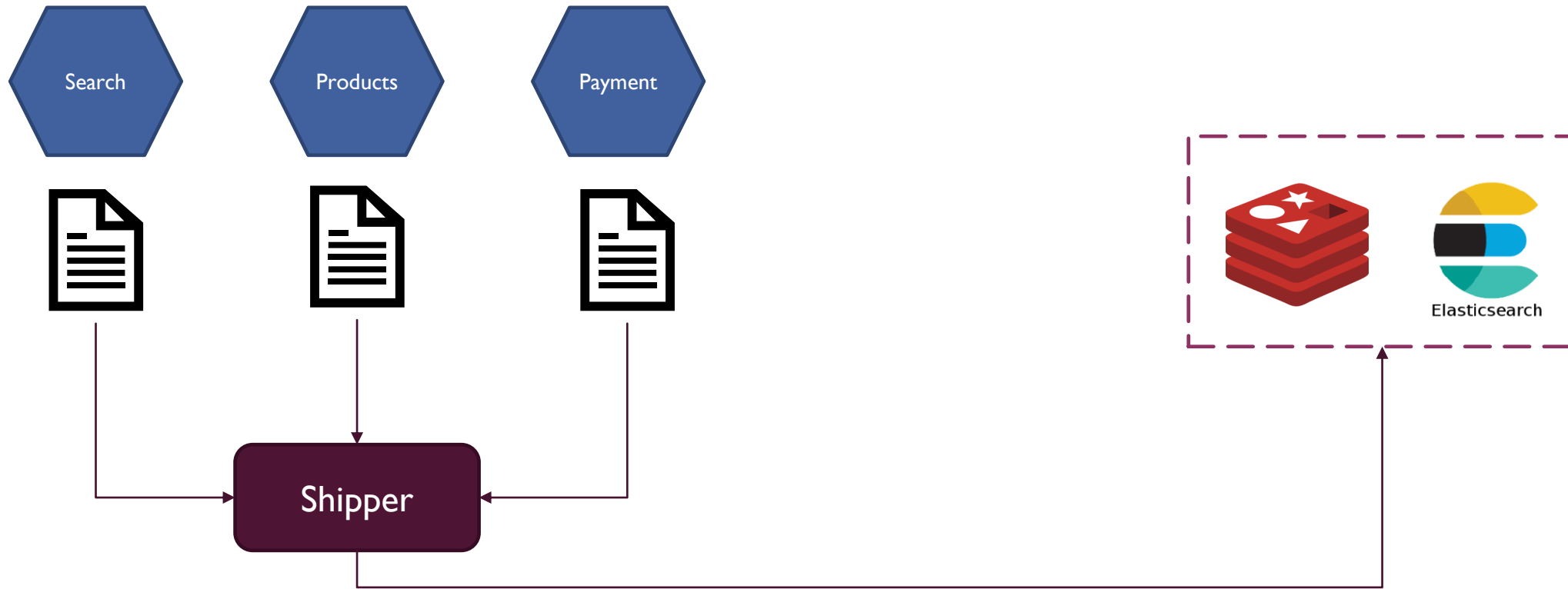
MICROSERVICES – LOGGING

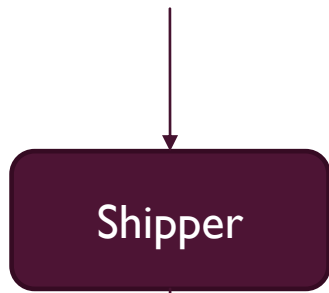
ENG. ISMAIL ANJRINI

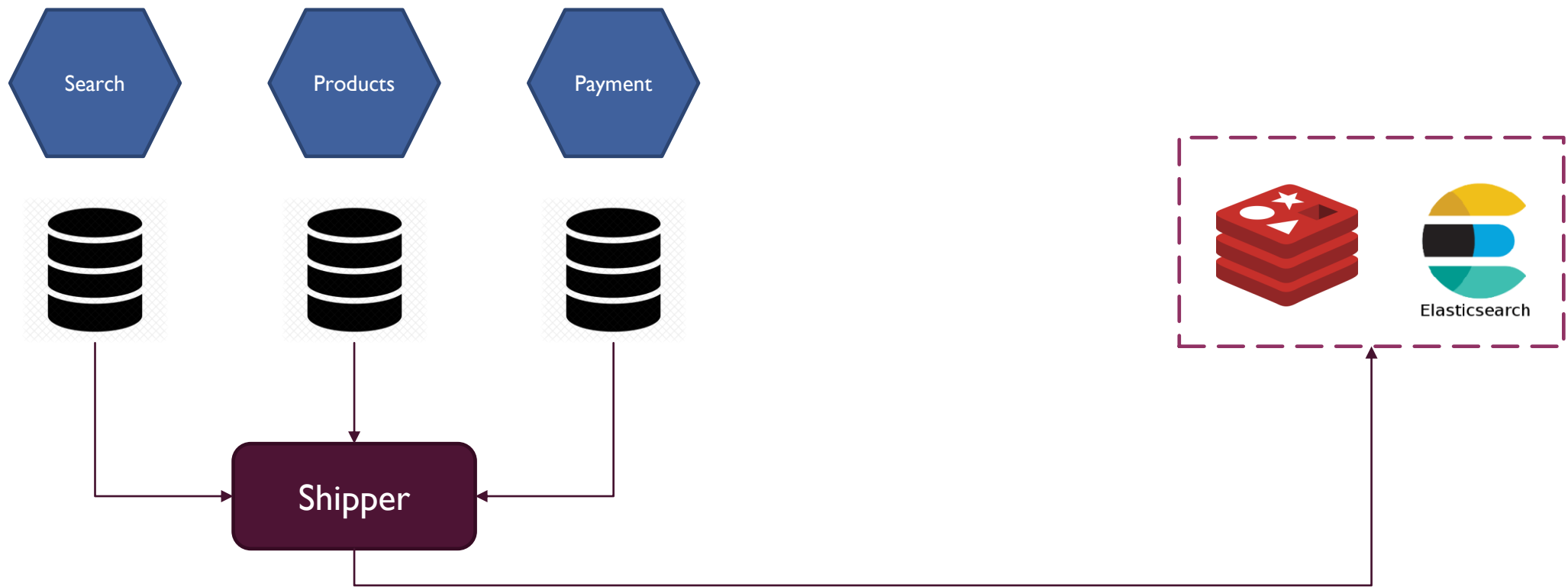














MICROSERVICES – MICRO FRONTEND

ENG. ISMAIL ANJRINI



