

August 20 , 2025

B.R.A.D. DEMO 3



- 
- 01** Architecture Diagram
 - 02** Deployment Model
 - 03** Live Demo
 - 04** Other Documentation
 - 05** Meet the team and contributions
 - 06** Q&A

TABLE OF CONTENT

DOCUMENTATION

Architecture diagram

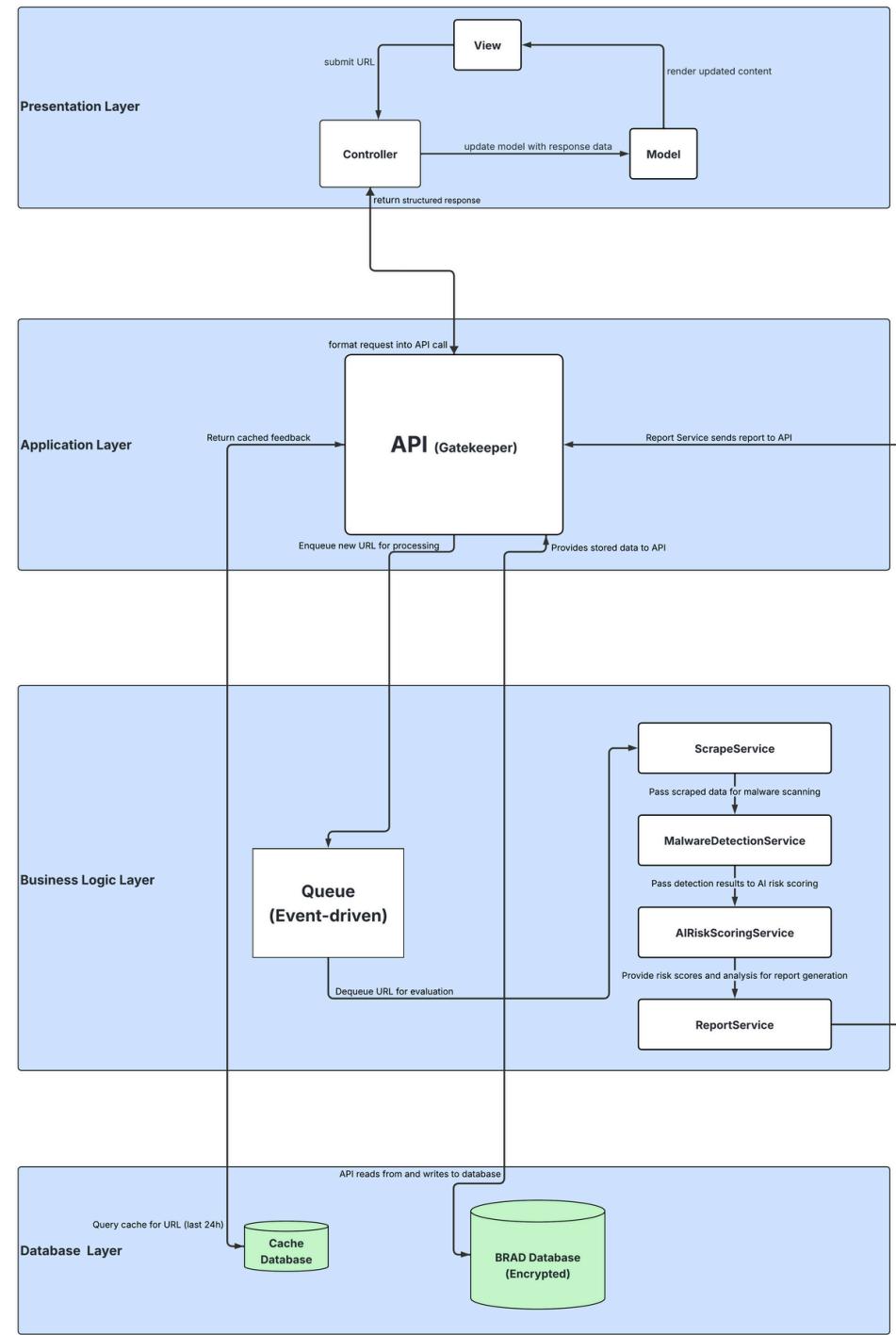
Deployment model

Coding standards

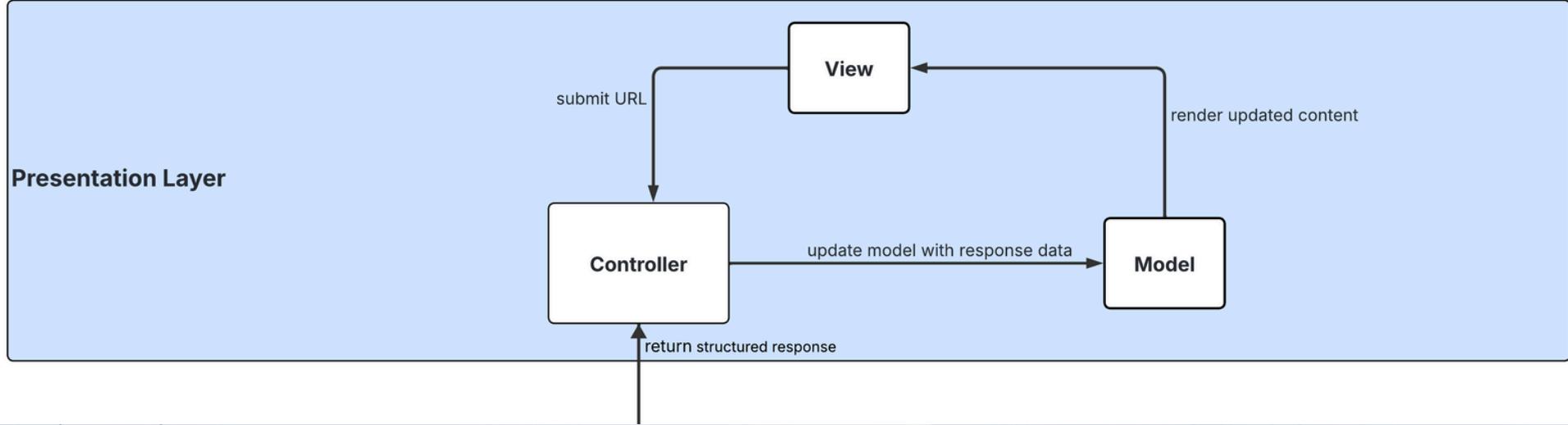
User manual

ARCHITECTURAL DIAGRAM (1/6)

Architecture Diagram

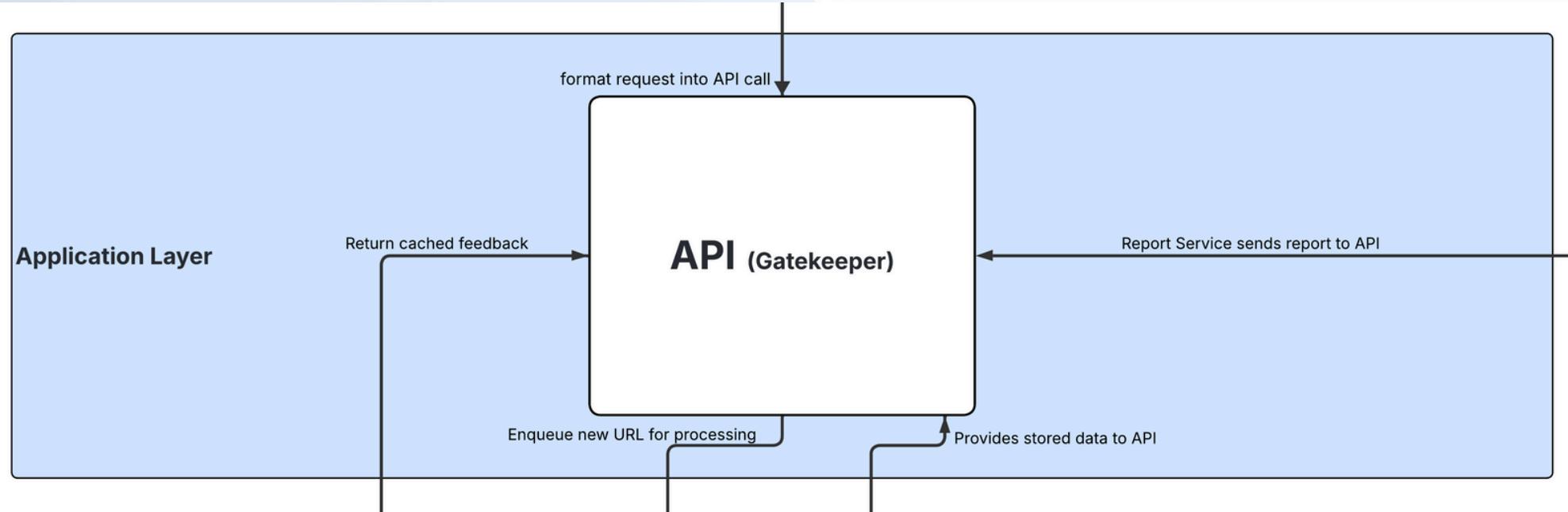


PRESENSTATION LAYER (2/6)



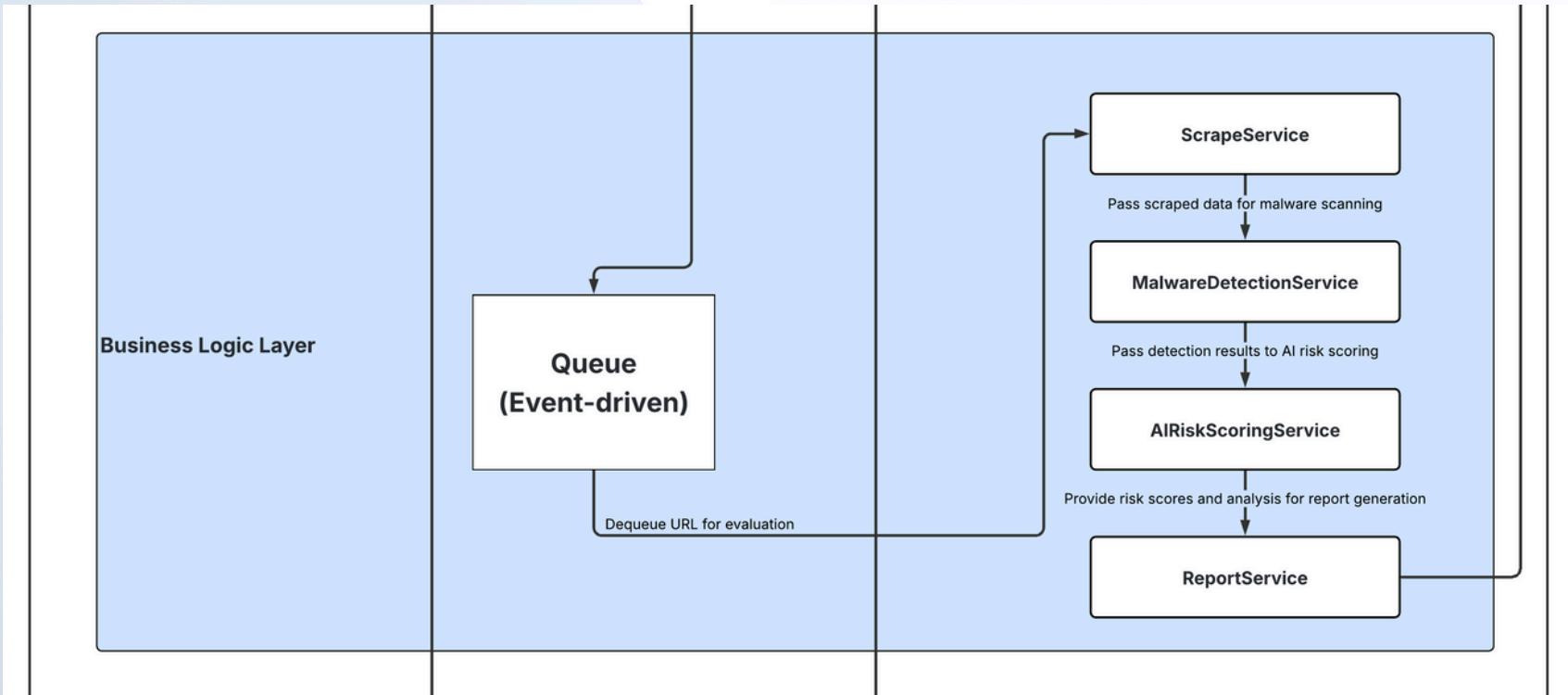
- WHERE USERS ARE ABLE TO INTERACT WITH THE SYSTEM
- **MODEL VIEW CONTROLLER APPLIED**
 - THE **VIEW** IS WHAT USERS ARE ABLE TO SEE
 - **CONTROLLER** HANDLES SUBMISSION OF URL'S AND EVIDENCE
 - **MODEL** SHOWS STATE OF DATA BACK TO USER
- MAKES OUR FRONTEND SIMPLE, RESPONSIVE AND EASY TO MAINTAIN

APPLICATION LAYER (3/6)



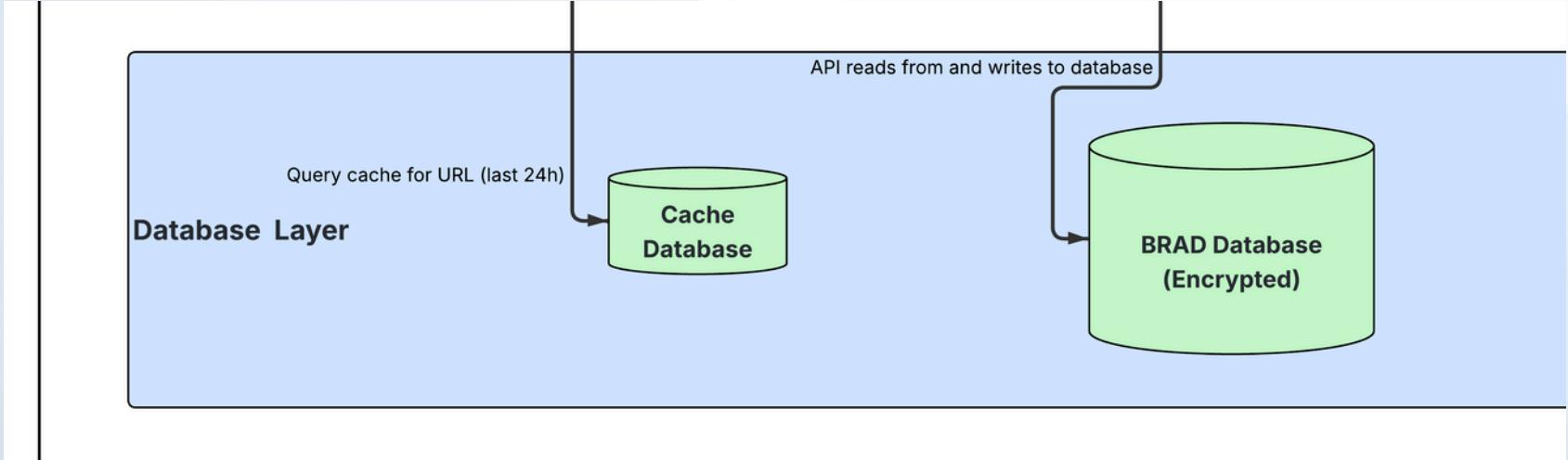
- **GATEKEEPER PATTERN IMPLEMENTED**
- API IS THE SINGLE ENTRY POINT INTO THE BACKEND
- ENFORCES ROLE BASED ACCESS CONTROL AND MFA
 - USED TO VALIDATE REQUESTS

BUSINESS LOGIC LAYER (4/6)



- COMBINATION OF **EVENT DRIVE ARCHITECTURE AND PIPE AND FILTER PATTERN**
- THE QUEUE ENABLES EVENT DRIVEN PROCESSING
 - WHEN A NEW URL IS SUBMITTED IT IS ADDED TO THE QUEUE
 - USING **PIPE AND FILTER PATTERN**, THE URL FLOWS THROUGH A SEQUENCE OF SERVICES, EACH PERFORMING A SPECIFIC TASK AND PASSING IT TO THE NEXT

DATABASE LAYER (5/6)



- CACHED DATA STORED FOR RECENTLY SUBMITTED URL'S
- ENCRYPTED RECORD STORED IN BRAD DATABASE
- ENSURES RELIABILITY, PERFORMANCE, COMPLIANCE

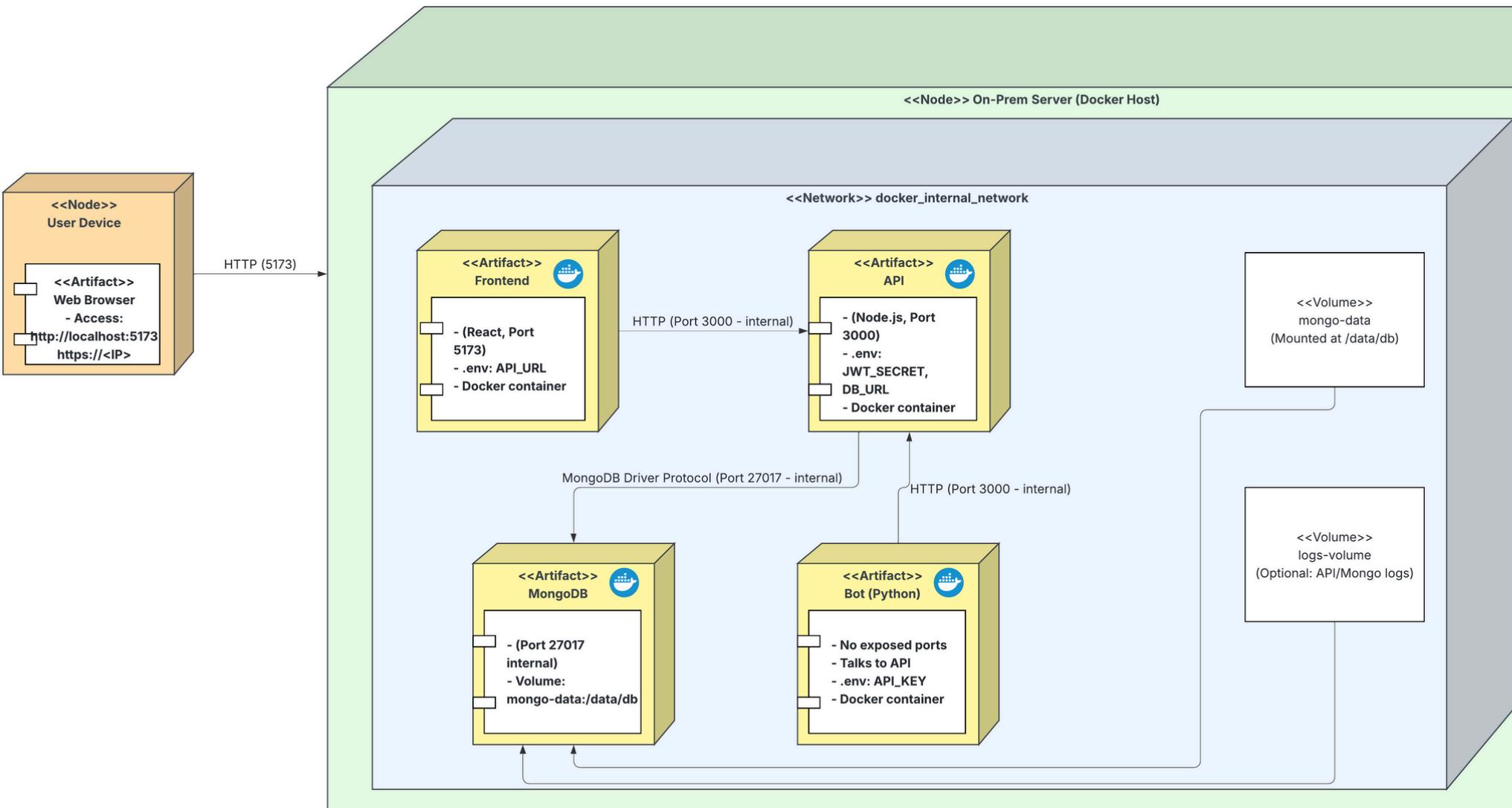
ARCHITECTURAL DIAGRAM (6/6)

THE COMBINATION OF THESE LAYERS AND
PATTERNS OUR ARCHITECTURE ENSURES:

- SECURITY
- SCALABILITY
- RELIABILITY
- MAINTAINABILITY

WHILE KEEPING THE DESIGN CLEAN AND
STRUCTURED

DEPLOYMENT MODEL

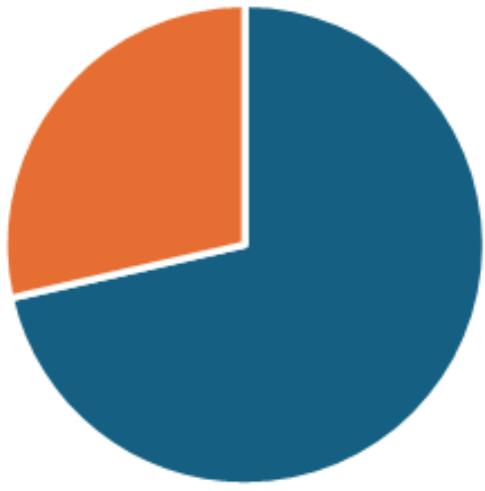


LIVE DEMO

We would like to show you everything we
have implemented for B.R.A.D.

CORE REQUIREMENTS PROGRESS

Core requirements



■ DONE ■ IN PROGRESS

Core Requirements	Progress
User Submission Portal	DONE
Scraping & Malware Detection	IN PROGRESS
Forensic Data Collection	DONE
AI-Powered Analysis	IN PROGRESS
Incident Reporting System	DONE
Investigator Dashboard	DONE
Secure Data Storage	DONE

Progress	COUNT
DONE	5
IN PROGRESS	2

OPTIONAL REQUIREMENTS PROGRESS

Optional Requirements



■ TO DO ■ IN PROGRESS ■ DONE

Optional Requirements	Progress
Threat Intelligence Integration	TO DO
Automated WHOIS & Reverse DNS Lookup	DONE
AI-Based Domain Similarity Detection	TO DO
Real-Time Alerting System	IN PROGRESS
Historical Analysis	DONE
Multi-Language Support	TO DO

Progress	Count
TO DO	3
IN PROGRESS	1
DONE	2

Meet Our Team



Ethan Vletter

Project Manager,
DevOps



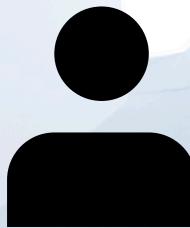
Megan Pretorius

UI/UX, API



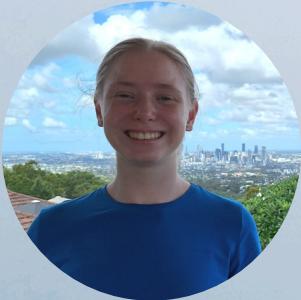
**Tebatso
Mahlathini**

API



Salome Kalaka

Architecture , API



Carinda Smith

BOT,API

Testing and CI/CD Pipeline

Actions

New workflow

All workflows

.github/workflows/deploy-no-docker.yml

CI/CD Pipeline

CI/CD Pipeline

Management

Caches

Attestations

Runners

Usage metrics

Performance metrics

CI/CD Pipeline

deploy.yml

73 workflow runs

- Merge pull request #203 from COS301-SE-2025/ui**
CI/CD Pipeline #73: Commit [431fff8](#) pushed by [carinda-smith](#) dev
- Merge pull request #202 from COS301-SE-2025/deployment-proxy**
CI/CD Pipeline #72: Commit [a88fc7e](#) pushed by [EthanVletter](#) dev
- Merge pull request #200 from COS301-SE-2025/settings**
CI/CD Pipeline #71: Commit [a588a0b](#) pushed by [EthanVletter](#) dev
- Merge pull request #198 from COS301-SE-2025/ui**
CI/CD Pipeline #70: Commit [fe6b949](#) pushed by [EthanVletter](#) dev
- Merge pull request #196 from COS301-SE-2025/stats-frontend**
CI/CD Pipeline #69: Commit [1852d56](#) pushed by [carinda-smith](#) dev
- Merge pull request #194 from COS301-SE-2025/notifications**
CI/CD Pipeline #68: Commit [c7acd1b](#) pushed by [carinda-smith](#) dev
- Merge pull request #192 from COS301-SE-2025/ui**
CI/CD Pipeline #67: Commit [3a50305](#) pushed by [carinda-smith](#) dev

QGA

**THANK
YOU**