HƯỚNG DẪN TRIỂN KHAI HỆ THỐNG FACTCHECK PLATFORM

# Tài liệu dành cho IT Administrator

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## MỤC LỤC

1. [Tổng quan hệ thống](#tổng-quan-hệ-thống)

2. [Yêu cầu hệ thống](#yêu-cầu-hệ-thống)

3. [Đăng ký dịch vụ](#đăng-ký-dịch-vụ)

4. [Cài đặt môi trường](#cài-đặt-môi-trường)

5. [Cấu hình CI/CD](#cấu-hình-cicd)

6. [Triển khai hệ thống](#triển-khai-hệ-thống)

7. [Kết quả triển khai](#kết-quả-triển-khai)

8. [Video hướng dẫn](#video-hướng-dẫn)

9. [Khắc phục sự cố](#khắc-phục-sự-cố)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 1. TỔNG QUAN HỆ THỐNG

## 1.1 Kiến trúc tổng thể

FactCheck Platform là một hệ thống microservices với kiến trúc event-driven, bao gồm:

**\*Frontend:** React web application (Port 3000)

**\*API Gateway:** Điểm vào duy nhất cho tất cả requests (Port 8080)

**\*Auth Service:** Xác thực và phân quyền người dùng (Port 3001)

**\*Link Service:** Phân tích URL và phát hiện mối đe dọa (Port 3002)

**\*Community Service:** Quản lý bài đăng và tương tác cộng đồng (Port 3003)

**\*Chat Service:** Chat AI với Google Gemini (Port 3004)

**\*News Service:** Tổng hợp tin tức và fact-checking (Port 3005)

**\*Admin Service:** Dashboard quản trị (Port 3006)

**\*Event Bus Service:** Quản lý sự kiện (Port 3007)

**\*ETL Service:** Xử lý dữ liệu và analytics (Port 3008)

## 1.2 Hạ tầng cơ sở

**\*Redis:** Cache và session storage (Port 6379)

**\*RabbitMQ:** Message queue và pub/sub (Port 5672, 15672)

**\*KurrentDB:** Event sourcing database (Port 2113, 1113)

**\*Prometheus:** Thu thập metrics (Port 9090)

**\*Grafana:** Dashboard giám sát (Port 3010)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 2. YÊU CẦU HỆ THỐNG

## 2.1 Yêu cầu phần cứng tối thiểu

**\*CPU:** 4 cores

**\*RAM:** 8GB

**\*Storage:** 50GB SSD

**\*Network:** 100Mbps

## 2.2 Yêu cầu phần mềm

**\*Operating System:** Windows 10/11, Linux Ubuntu 20.04+, macOS 10.15+

**\*Docker:** Version 20.10+

**\*Docker Compose:** Version 2.0+

**\*Node.js:** Version 18+

**\*npm:** Version 8+

**\*Git:** Version 2.30+

## 2.3 Yêu cầu mạng

**\*Ports:** 3000-3010, 5672, 6379, 8080, 9090, 2113, 1113

**\*Firewall:** Mở các ports cần thiết

**\*DNS:** Cấu hình domain name (nếu có)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 3. ĐĂNG KÝ DỊCH VỤ

## 3.1 Firebase Configuration

1. \*\*Tạo Firebase Project\*\*:

* Truy cập https://console.firebase.google.com
* Tạo project mới với tên "factcheck-platform"
* Ghi lại Project ID

2. \*\*Cấu hình Authentication\*\*:

* Vào Authentication > Sign-in method
* Bật Email/Password
* Bật Google Sign-in (tùy chọn)

3. \*\*Cấu hình Firestore Database\*\*:

* Vào Firestore Database
* Tạo database ở chế độ production
* Chọn location gần nhất

4. \*\*Tạo Service Account\*\*:

* Vào Project Settings > Service accounts
* Tạo service account key
* Download file JSON
* Lưu trữ an toàn

## 3.2 API Keys Registration

1. \*\*Google Gemini API\*\*:

* Truy cập https://makersuite.google.com/app/apikey
* Tạo API key mới
* Ghi lại key

2. \*\*VirusTotal API\*\* (tùy chọn):

* Truy cập https://www.virustotal.com/gui/join-us
* Đăng ký tài khoản
* Lấy API key

3. \*\*News API\*\* (tùy chọn):

* Truy cập https://newsapi.org/register
* Đăng ký tài khoản
* Lấy API key

## 3.3 Domain và SSL (Production)

1. \*\*Đăng ký domain\*\* (nếu chưa có)

2. \*\*Cấu hình DNS records\*\*

3. \*\*Cài đặt SSL certificate\*\*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 4. CÀI ĐẶT MÔI TRƯỜNG

## 4.1 Cài đặt Docker

Windows

Tải Docker Desktop từ https://www.docker.com/products/docker-desktop

Cài đặt và khởi động Docker Desktop

Linux Ubuntu

sudo apt update

sudo apt install docker.io docker-compose

sudo systemctl start docker

sudo systemctl enable docker

sudo usermod -aG docker $USER

macOS

Tải Docker Desktop từ https://www.docker.com/products/docker-desktop

Cài đặt và khởi động Docker Desktop

## 4.2 Cài đặt Node.js

Windows

Tải Node.js từ https://nodejs.org/

Cài đặt phiên bản LTS

Linux Ubuntu

curl -fsSL https://deb.nodesource.com/setup\_18.x | sudo -E bash -

sudo apt-get install -y nodejs

macOS

brew install node

## 4.3 Clone Repository

git clone https://github.com/your-organization/factcheck-platform.git

cd factcheck-platform

## 4.4 Cấu hình Environment Variables

1. \*\*Tạo file .env\*\*:

cp .env.example .env

2. \*\*Cập nhật các biến môi trường\*\*:

Core Configuration

NODE\_ENV=production

REACT\_APP\_API\_URL=https://your-domain.com

Firebase Configuration

FIREBASE\_PROJECT\_ID=your-project-id

FIREBASE\_CLIENT\_EMAIL=your-service-account-email

FIREBASE\_PRIVATE\_KEY="your-private-key"

JWT Configuration

JWT\_SECRET=your-super-secret-jwt-key

API Keys

GEMINI\_API\_KEY=your-gemini-api-key

VIRUSTOTAL\_API\_KEY=your-virustotal-api-key

NEWSAPI\_API\_KEY=your-newsapi-key

Database URLs

REDIS\_URL=redis://:antifraud123@redis:6379

RABBITMQ\_URL=amqp://factcheck:antifraud123@rabbitmq:5672

KURRENTDB\_URL=esdb://kurrentdb:2113?tls=false

Service URLs

AUTH\_SERVICE\_URL=http://auth-service:3001

LINK\_SERVICE\_URL=http://link-service:3002

COMMUNITY\_SERVICE\_URL=http://community-service:3003

CHAT\_SERVICE\_URL=http://chat-service:3004

NEWS\_SERVICE\_URL=http://news-service:3005

ADMIN\_SERVICE\_URL=http://admin-service:3006

EVENT\_BUS\_SERVICE\_URL=http://event-bus-service:3007

ETL\_SERVICE\_URL=http://etl-service:3008

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 5. CẤU HÌNH CI/CD VÀ AUTOMATION

## 5.1 GitHub Actions CI/CD Pipeline

Project đã có hệ thống CI/CD hoàn chỉnh với file `.github/workflows/comprehensive-ci-cd.yml`:

### \*\*5.1.1 New Developer Workflow Test\*\*

Kiểm tra quy trình cho developer mới

* Test trên Ubuntu và Windows
* Test với Node.js 18 và 20
* Validate environment setup
* Test service startup capability

### \*\*5.1.2 Individual Service Deployment Test\*\*

Test từng microservice riêng lẻ

* API Gateway, Auth Service, Link Service
* Community Service, Chat Service, News Service
* Admin Service, PhishTank Service, CriminalIP Service
* Docker build và container test
* Health check validation

### \*\*5.1.3 Client Build Test\*\*

Test React client build

* npm install và build process
* Static site generation
* Environment variable injection
* Render deployment readiness

### \*\*5.1.4 Comprehensive Deployment Guide Generation\*\*

Tự động tạo deployment guide

* Generate từ CI/CD test results
* Include troubleshooting guides
* Multi-platform deployment instructions
* Environment configuration templates

## 5.2 Infrastructure as Code (IaaC)

### \*\*5.2.1 Docker Compose Configuration\*\*

docker-compose.yml - Single source of truth

version: '3.8'

services:

Infrastructure Components

redis: # Cache và session storage

rabbitmq: # Message queue

kurrentdb: # Event sourcing database

prometheus: # Metrics collection

grafana: # Monitoring dashboards

Microservices

api-gateway: # Single entry point

auth-service: # Authentication

link-service: # URL analysis

community-service: # Social features

chat-service: # AI chat

news-service: # News aggregation

admin-service: # Admin dashboard

event-bus-service: # Event management

etl-service: # Data processing

### \*\*5.2.2 Kubernetes Deployment (K8s)\*\*

k8s/ - Kubernetes manifests

* api-gateway.yml
* auth-service.yml
* link-service.yml
* community-service.yml
* chat-service.yml
* news-service.yml
* admin-service.yml
* monitoring.yml
* ingress.yml

### \*\*5.2.3 Automated Scripts\*\*

Scripts tự động hóa deployment

scripts/

├── deploy-local-all.js # Local development

├── deploy-docker-all.js # Docker deployment

├── deploy-k8s-all.js # Kubernetes deployment

├── docker-start-simple.js # Simplified Docker

├── health-all.js # Health checks

├── status-all.js # Service status

├── logs-all.js # Log aggregation

└── env-setup.js # Environment setup

## 5.3 Continuous Integration Features

### \*\*5.3.1 Automated Testing\*\*

**\*Unit Tests:** Jest framework cho từng service

**\*Integration Tests:** Service-to-service communication

**\*End-to-End Tests:** Complete workflow validation

**\*Performance Tests:** Load testing với Artillery

### \*\*5.3.2 Code Quality Checks\*\*

**\*Linting:** ESLint cho JavaScript/TypeScript

**\*Formatting:** Prettier code formatting

**\*Security:** npm audit và dependency scanning

**\*Coverage:** Code coverage reports

### \*\*5.3.3 Environment Validation\*\*

Automated environment checks

npm run env:validate # Validate .env configuration

npm run env:test # Test environment loading

npm run test:new-dev # Test new developer workflow

npm run test:workflow # Test complete deployment workflow

## 5.4 Continuous Deployment Features

### \*\*5.4.1 Multi-Environment Support\*\*

Environment-specific configurations

* Development: Local development setup
* Staging: Pre-production testing
* Production: Live deployment
* Big Data: Hadoop/Spark integration

### \*\*5.4.2 Automated Deployment Scripts\*\*

Universal deployment interface

npm run deploy:local # Local development

npm run deploy:docker # Docker containers

npm run deploy:k8s # Kubernetes cluster

npm run docker:start # Simplified Docker stack

npm run bigdata:start # Big data stack

### \*\*5.4.3 Health Monitoring\*\*

Automated health checks

npm run health # All services health

npm run status # Service status

npm run docker:health # Docker health

npm run logs # Centralized logging

## 5.5 Deployment Automation Features

### \*\*5.5.1 Cross-Platform Support\*\*

**\*Windows:** PowerShell và CMD scripts

**\*macOS:** Bash scripts

**\*Linux:** Bash scripts

**\*Docker:** Universal containerization

### \*\*5.5.2 Automated Resource Provisioning\*\*

Infrastructure provisioning

npm run infrastructure:start # Start Redis, RabbitMQ, KurrentDB

npm run infrastructure:stop # Stop infrastructure

npm run infrastructure:status # Check infrastructure status

### \*\*5.5.3 Service Discovery và Load Balancing\*\*

**\*API Gateway:** Single entry point với load balancing

**\*Service Mesh:** Inter-service communication

**\*Health Checks:** Automatic service monitoring

**\*Auto-scaling:** Kubernetes HPA configuration

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 6. TRIỂN KHAI HỆ THỐNG

## 6.1 Triển khai Development

Cài đặt dependencies

npm run setup

Khởi động infrastructure

npm run infrastructure:start

Khởi động tất cả services

npm start

Kiểm tra trạng thái

npm run status

Kiểm tra health

npm run health

## 6.2 Triển khai Production với Docker

Build và khởi động toàn bộ stack

npm run docker:start

Hoặc sử dụng Docker Compose trực tiếp

docker-compose up -d

Kiểm tra logs

npm run docker:logs

Kiểm tra health

npm run docker:health

## 6.3 Triển khai từng service riêng lẻ

Khởi động infrastructure trước

npm run infrastructure:start

Khởi động từng service

npm run start:auth

npm run start:link

npm run start:community

npm run start:chat

npm run start:news

npm run start:admin

npm run start:gateway

npm run start:client

## 6.4 Triển khai Big Data Stack

Khởi động Hadoop và Spark

npm run bigdata:start

Kiểm tra trạng thái

npm run bigdata:health

Xem logs

npm run bigdata:logs

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 7. KẾT QUẢ TRIỂN KHAI THỰC TẾ

## 7.1 Web UI (Frontend) - Production Ready

**\*Development URL:** http://localhost:3000

**\*Production URL:** https://factcheck-platform.onrender.com

**\*Static Site URL:** https://factcheck-client.onrender.com

**\*Tính năng đã triển khai:**

* ✅ Giao diện người dùng React hiện đại với Material-UI
* ✅ Responsive design cho mobile và desktop
* ✅ Real-time updates với WebSocket và Socket.io
* ✅ Dark/Light theme với ThemeContext
* ✅ Multi-language support (Vietnamese/English)
* ✅ Progressive Web App (PWA) capabilities
* ✅ Service Worker cho offline functionality
* ✅ Lazy loading và code splitting
* ✅ SEO optimization với React Helmet

## 7.2 APIs (Backend Services) - Production Deployed

**\*API Gateway:**

* Development: http://localhost:8080
* Production: https://factcheck-api-gateway.onrender.com
* ✅ Single entry point cho tất cả requests
* ✅ Authentication middleware với JWT
* ✅ Rate limiting (100 requests/minute)
* ✅ Request routing và load balancing
* ✅ CORS configuration cho production
* ✅ Request logging và monitoring

**\*Auth Service:**

* Development: http://localhost:3001
* Production: https://factcheck-auth.onrender.com
* ✅ User registration và login với Firebase Auth
* ✅ JWT token management với refresh tokens
* ✅ Role-based access control (User, Admin, Moderator)
* ✅ Password reset functionality
* ✅ Email verification
* ✅ Google OAuth integration
* ✅ Multi-factor authentication support

**\*Link Service:**

* Development: http://localhost:3002
* Production: https://factcheck-link.onrender.com
* ✅ URL security analysis với multiple APIs
* ✅ Threat detection (VirusTotal, ScamAdviser, CriminalIP)
* ✅ Real-time scanning và scoring
* ✅ Scan history management với caching
* ✅ Screenshot capture cho visual verification
* ✅ Phishing detection với PhishTank integration

**\*Community Service:**

* Development: http://localhost:3003
* Production: https://factcheck-community.onrender.com
* ✅ Post creation và management với rich text editor
* ✅ Voting system (upvote/downvote)
* ✅ Comment functionality với threading
* ✅ User reputation system và badges
* ✅ Content moderation tools
* ✅ Real-time notifications

**\*Chat Service:**

* Development: http://localhost:3004
* Production: https://factcheck-chat.onrender.com
* ✅ AI-powered chat với Google Gemini Pro
* ✅ Real-time messaging với Socket.io
* ✅ Security recommendations và tips
* ✅ Educational content về cybersecurity
* ✅ Chat history và context preservation
* ✅ Multi-language AI responses

**\*News Service:**

* Development: http://localhost:3005
* Production: https://factcheck-news.onrender.com
* ✅ News aggregation từ multiple sources
* ✅ Fact-checking alerts và notifications
* ✅ Source credibility scoring algorithm
* ✅ Trending topics analysis
* ✅ News categorization và filtering
* ✅ RSS feed integration

**\*Admin Service:**

* Development: http://localhost:3006
* Production: https://factcheck-admin.onrender.com
* ✅ Administrative dashboard với analytics
* ✅ User management và role assignment
* ✅ System monitoring và health checks
* ✅ Content moderation tools
* ✅ Performance metrics và reports
* ✅ Audit logs và activity tracking

## 7.3 Databases - Production Infrastructure

**\*Redis:**

* Development: localhost:6379
* Production: Redis Cloud hoặc AWS ElastiCache
* ✅ Session storage với TTL management
* ✅ Cache management với LRU eviction
* ✅ Pub/Sub messaging cho real-time features
* ✅ Real-time data synchronization
* ✅ Connection pooling và failover

**\*KurrentDB (EventStore):**

* Development: localhost:2113, 1113
* Production: EventStore Cloud hoặc self-hosted
* ✅ Event sourcing database với CQRS pattern
* ✅ Event history và replay capability
* ✅ Audit trail cho compliance
* ✅ Data consistency với optimistic concurrency
* ✅ Event projections và read models

**\*Firebase Firestore:**

* Production: Google Cloud Firestore
* ✅ User data với real-time synchronization
* ✅ Content storage với automatic scaling
* ✅ Offline support với conflict resolution
* ✅ Security rules và access control
* ✅ Automatic backups và disaster recovery
* ✅ Global distribution với low latency

## 7.4 Authentication and Authorization Service - Production Ready

**\*Firebase Authentication:**

* Production: Google Firebase Auth
* ✅ Email/password authentication với secure hashing
* ✅ Google OAuth integration với SSO
* ✅ Phone number verification với SMS
* ✅ Multi-factor authentication (MFA)
* ✅ Social login (Google, Facebook, Twitter)
* ✅ Account linking và profile management
* ✅ Password strength validation
* ✅ Account recovery và security questions

**\*JWT Token Management:**

* ✅ Secure token generation với RSA256
* ✅ Token validation với signature verification
* ✅ Refresh token mechanism với rotation
* ✅ Token revocation và blacklisting
* ✅ Token expiration và automatic renewal
* ✅ Role-based access control (RBAC)
* ✅ Permission-based authorization
* ✅ Session management và concurrent login control

## 7.5 Monitoring và Analytics - Production Monitoring

**\*Prometheus:**

* Development: http://localhost:9090
* Production: Prometheus Cloud hoặc self-hosted
* ✅ Metrics collection với custom exporters
* ✅ Performance monitoring với histograms
* ✅ Alert management với notification routing
* ✅ Data visualization với Grafana integration
* ✅ Service discovery và auto-configuration
* ✅ Long-term storage với remote storage

**\*Grafana:**

* Development: http://localhost:3010
* Production: Grafana Cloud hoặc self-hosted
* ✅ Custom dashboards với templating
* ✅ Real-time monitoring với live updates
* ✅ Alert notifications qua email/Slack/Discord
* ✅ Performance analytics với APM integration
* ✅ User management và access control
* ✅ Dashboard sharing và collaboration

**\*RabbitMQ Management:**

* Development: http://localhost:15672
* Production: RabbitMQ Cloud hoặc self-hosted
* ✅ Message queue monitoring với metrics
* ✅ Queue management và policy configuration
* ✅ Performance metrics với throughput analysis
* ✅ Connection monitoring và troubleshooting
* ✅ Cluster management và high availability
* ✅ Message tracing và debugging tools

## 7.6 Infrastructure Components

**\*Message Queue (RabbitMQ):**

* Reliable message delivery
* Event-driven architecture
* Service decoupling
* Scalability support

**\*Event Bus Service:**

* Centralized event management
* Event routing
* Event persistence
* Event replay capability

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 8. VIDEO HƯỚNG DẪN

## Trang 2: Liên kết Video YouTube

\*\*Video Demo Triển khai Hệ thống FactCheck Platform\*\*

📺 \*\*Link Video\*\*: [https://www.youtube.com/watch?v=YOUR\_VIDEO\_ID](https://www.youtube.com/watch?v=YOUR\_VIDEO\_ID)

\*\*Nội dung video bao gồm\*\*:

1. \*\*Phần 1: Chuẩn bị môi trường\*\* (0:00 - 5:00)

* Cài đặt Docker và Node.js
* Clone repository
* Cấu hình environment variables

2. \*\*Phần 2: Đăng ký dịch vụ\*\* (5:00 - 15:00)

* Tạo Firebase project
* Cấu hình Authentication
* Đăng ký API keys

3. \*\*Phần 3: Triển khai Development\*\* (15:00 - 25:00)

* Khởi động infrastructure
* Deploy các microservices
* Kiểm tra health status

4. \*\*Phần 4: Triển khai Production\*\* (25:00 - 35:00)

* Docker deployment
* Domain configuration
* SSL certificate setup

5. \*\*Phần 5: Kiểm tra kết quả\*\* (35:00 - 45:00)

* Test các APIs
* Verify web UI
* Monitor system performance

6. \*\*Phần 6: Troubleshooting\*\* (45:00 - 50:00)

* Common issues
* Debug techniques
* Performance optimization

\*\*Thông tin bổ sung\*\*:

**\*Thời lượng:** 50 phút

**\*Ngôn ngữ:** Tiếng Việt

**\*Độ phân giải:** 1080p

**\*Tác giả:** FactCheck Platform Team

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 9. KHẮC PHỤC SỰ CỐ

## 9.1 Common Issues

### Port Conflicts

Kiểm tra ports đang sử dụng

netstat -an | findstr :3000

netstat -an | findstr :8080

Dừng services xung đột

npm run docker:stop

### Container Won't Start

Kiểm tra logs

docker-compose logs service-name

Rebuild container

docker-compose up --build service-name

Remove và recreate

docker-compose down

docker-compose up -d

### Memory Issues

Kiểm tra resource usage

docker stats

Clean up unused resources

docker system prune -a

Tăng memory limit trong Docker Desktop

## 9.2 Health Check Commands

Kiểm tra tất cả services

npm run health

Kiểm tra từng service

curl http://localhost:8080/health # API Gateway

curl http://localhost:3001/health # Auth Service

curl http://localhost:3003/health # Community Service

Kiểm tra infrastructure

npm run infrastructure:status

## 9.3 Reset Everything

Nuclear option - clean everything

docker-compose down -v --remove-orphans

docker system prune -a --volumes

npm run docker:start

## 9.4 Performance Optimization

Scale services

docker-compose up -d --scale community-service=3

Monitor performance

docker stats

Check logs for errors

docker-compose logs -f

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 10. BẢO TRÌ VÀ NÂNG CẤP

## 10.1 Backup Strategy

Backup databases

docker-compose exec redis redis-cli BGSAVE

docker-compose exec kurrentdb eventstore backup

Backup configuration

cp .env .env.backup

cp docker-compose.yml docker-compose.yml.backup

## 10.2 Update Procedures

Pull latest changes

git pull origin main

Update dependencies

npm install

Rebuild containers

docker-compose up --build -d

Run migrations

npm run migrate

## 10.3 Monitoring Alerts

**\*CPU Usage:** > 80%

**\*Memory Usage:** > 85%

**\*Disk Usage:** > 90%

**\*Service Health:** Any service down

**\*API Response Time:** > 5 seconds

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 11. TÀI LIỆU THAM KHẢO

## 11.1 Documentation Links

* [Docker Documentation](https://docs.docker.com/)
* [Node.js Documentation](https://nodejs.org/docs/)
* [Firebase Documentation](https://firebase.google.com/docs)
* [React Documentation](https://reactjs.org/docs/)

## 11.2 Support Resources

**\*GitHub Issues:** [Project Issues](https://github.com/your-org/factcheck-platform/issues)

**\*Documentation:** [Project Wiki](https://github.com/your-org/factcheck-platform/wiki)

**\*Community:** [Discord Server](https://discord.gg/your-server)

## 11.3 Emergency Contacts

**\*Technical Lead:** tech-lead@company.com

**\*DevOps Engineer:** devops@company.com

**\*System Administrator:** sysadmin@company.com

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 12. PHỤ LỤC

## 12.1 Command Reference

Development

npm start # Start all services

npm run dev # Development mode

npm run docker:start # Docker deployment

npm run bigdata:start # Big data stack

Monitoring

npm run status # Service status

npm run health # Health checks

npm run logs # View logs

Management

npm run stop # Stop all services

npm run restart # Restart all services

npm run clean # Clean up resources

## 12.2 Configuration Files

* `.env` - Environment variables
* `docker-compose.yml` - Docker services
* `package.json` - Dependencies và scripts
* `nginx.conf` - Web server configuration

## 12.3 Log Locations

**\*Application Logs:** `logs/` directory

**\*Docker Logs:** `docker-compose logs`

**\*System Logs:** `/var/log/` (Linux)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\*\*Tài liệu này được cập nhật lần cuối\*\*: [Ngày hiện tại]

\*\*Phiên bản\*\*: 2.0.0

\*\*Tác giả\*\*: FactCheck Platform Team

\*\*Phê duyệt\*\*: IT Administrator

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\*Lưu ý: Đây là tài liệu nội bộ, vui lòng không chia sẻ ra ngoài tổ chức.\*