



# Docker Development Environment - Hoàn tất!

---

✓ Đã tạo xong môi trường Docker Development đầy đủ

📁 Các file đã tạo:

## Scripts khởi động (Windows):

- ✓ `start-dev.bat` - Khởi động tất cả (SCRIPT CHÍNH)
- ✓ `stop-dev.bat` - Dừng tất cả services
- ✓ `reset-dev.bat` - Reset hoàn toàn (xóa data)

## Scripts khởi động (Linux/Mac):

- ✓ `docker-dev.sh` - Script khởi động đầy đủ với logs
- ✓ `docker-dev.ps1` - PowerShell script (alternative)

## Docker Configuration:

- ✓ `docker-compose.yml` - Đã cập nhật với tất cả services
- ✓ `Dockerfile` - Đã cập nhật cho môi trường dev
- ✓ `docker/init-db/01-init.sql` - Script khởi tạo database
- ✓ `docker/pgadmin/servers.json` - Cấu hình pgAdmin tự động
- ✓ `docker/wait-for-it.sh` - Script đợi database ready

## Documentation:

- ✓ `DEV-QUICKSTART.md` - Hướng dẫn nhanh (ĐỌC ĐẦU TIÊN!)
- ✓ `DOCKER-DEV-README.md` - Hướng dẫn chi tiết đầy đủ
- ✓ `.env.example` - Template cho environment variables



## CÁCH SỬ DỤNG

Windows (Đơn giản nhất):

```
start-dev.bat
```

Chỉ cần double-click file `start-dev.bat`!

Hoặc dùng Docker Compose trực tiếp:

```
docker-compose up -d --build
```

---

## Điều gì sẽ xảy ra khi chạy?

### ☒ Lần chạy đầu tiên:

1. **Build application** từ source code
2. **Tạo database** `p2p_trading_dev` tự động
3. **Chạy tất cả migrations** Flyway tự động
4. **Seed master data** (currencies, payment methods...)
5. **Khởi động services:**
  - PostgreSQL Database (port 5432)
  - pgAdmin Web UI (port 5050)
  - RabbitMQ + Management (port 5672, 15672)
  - Application API (port 9000)

### ☒ Lần chạy tiếp theo:

- Database và data **vẫn còn** (không mất)
- Migrations mới **tự động chạy**
- Nếu có thay đổi code, **tự động rebuild**
- Chỉ mất ~30 giây thay vì 2-3 phút

---

## Truy cập các services

### Application API

```
http://localhost:9000/api
```

### pgAdmin (Quản lý Database qua Web)

```
http://localhost:5050  
Login: admin@p2p.local / admin123
```

Server "P2P Trading DB (Dev)" đã được cấu hình sẵn - chỉ cần click vào!

### PostgreSQL Database

```
Host: localhost:5432  
Database: p2p_trading_dev  
User: postgres  
Password: postgres123
```

```
http://localhost:15672  
Login: guest / guest
```

---

## Đọc gì tiếp theo?

1. **Quick Start:** Đọc [DEV-QUICKSTART.md](#) để biết workflow và troubleshooting
2. **Chi tiết kỹ thuật:** Đọc [DOCKER-DEV-README.md](#) để hiểu sâu hơn
3. **Test API:** Xem [docs/api-spec.md](#) để test các endpoints

---

## Các tính năng chính

- ✓ **Tự động tạo database** nếu chưa có
- ✓ **Tự động chạy migrations** mỗi lần start
- ✓ **Tự động seed data** lần đầu tiên
- ✓ **Web UI quản lý database** (pgAdmin) - không cần cài thêm gì
- ✓ **Data persistent** - không mất khi restart
- ✓ **Healthchecks** - đảm bảo services sẵn sàng trước khi start app
- ✓ **Port 9000** cho dev (dễ nhớ)
- ✓ **Một lệnh khởi động tất cả** - không cần setup gì thêm

---

## Troubleshooting nhanh

Application không chạy?

```
docker logs -f p2p-app
```

Port bị chiếm?

Sửa file [docker-compose.yml](#), section [ports](#)

Cần reset lại từ đầu?

```
reset-dev.bat  
start-dev.bat
```

---

## Workflow Development

## Sửa code Java:

1. Sửa code
2. `docker-compose restart app` (hoặc `start-dev.bat`)
3. Container tự động rebuild và restart

## Thêm migration mới:

1. Tạo file `VXXX__description.sql` trong `p2p_repository/src/main/resources/db/migration/postgres/`
2. `docker-compose restart app`
3. Migration tự động chạy

## Debug:

```
# Xem logs real-time
docker logs -f p2p-app

# Vào database console
docker exec -it p2p-postgres psql -U postgres -d p2p_trading_dev

# Xem tất cả containers
docker ps
```

---

## Tóm lại

Bạn giờ có một môi trường development **hoàn chỉnh** với:

- ☒ Database tự động setup
- ☒ Migrations tự động chạy
- ☒ Seed data tự động
- ☒ Web UI để quản lý database
- ☒ Message queue (RabbitMQ)
- ☒ Application chạy trên port 9000
- ☒ Tất cả trong **một lệnh duy nhất**

Chi cần chạy `start-dev.bat` và bắt đầu code! 

---

## Support

Nếu có vấn đề:

1. Xem logs: `docker logs -f p2p-app`
2. Đọc troubleshooting trong `DEV-QUICKSTART.md`
3. Reset: `reset-dev.bat` rồi `start-dev.bat`

