

SEMINAR

# XÂY DỰNG WEBSITE BẰNG PHƯƠNG PHÁP VIBE CODING SỬ DỤNG GITHUB COPILOT

NHÓM SINH VIÊN THỰC HIỆN

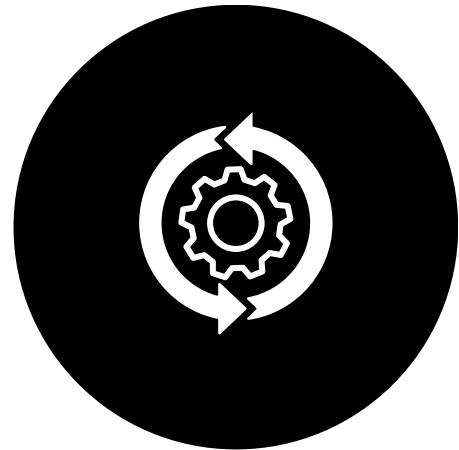
NGUYỄN HỮU ANH KHOA

VÕ THÀNH DANH

NGUYỄN LÊ QUỲNH HƯƠNG

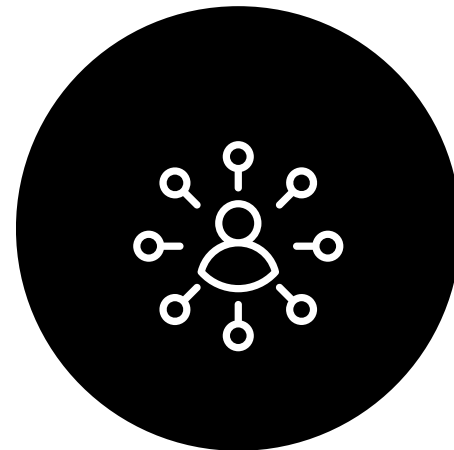
TRANG GIA HUY

# BỐI CẢNH BÀI TOÁN HIỆN NAY



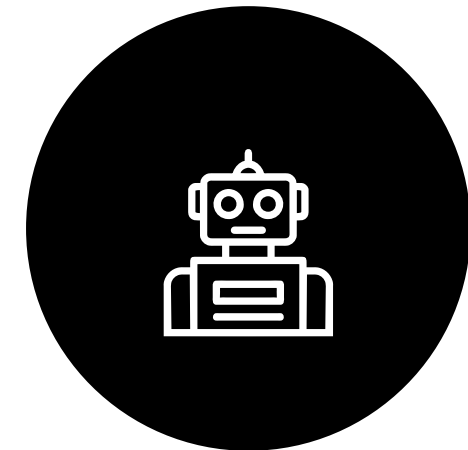
ĐỘ PHỨC TẠP TĂNG CAO

CÁC HỆ THỐNG HIỆN ĐẠI YÊU CẦU THAY  
ĐỔI CÔNG NGHỆ THƯỜNG XUYỀN



HẠN CHẾ NGUỒN LỰC

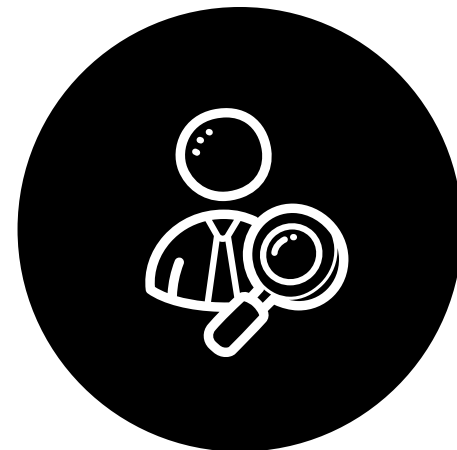
PHƯƠNG PHÁP TRUYỀN THỐNG GẶP RÀO  
CẢN VỀ THỜI GIAN VÀ CHI PHÍ BẢO TRÌ



KỸ NGUYÊN AI

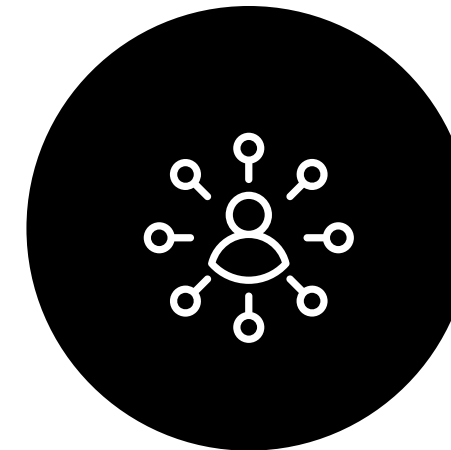
SỰ XUẤT HIỆN CỦA GITHUB COPILOT MỞ  
RA KHẢ NĂNG "VIBE CODING" ĐỘT PHÁ

# MỤC TIÊU VÀ PHẠM VI DỰ ÁN



NGHIÊN CỨU AI-ASSISTED  
DEVELOPMENT

ÁP DỤNG GITHUB COPILOT AGENT MODE  
VÀO TOÀN BỘ VÒNG ĐỜI PHÁT TRIỂN



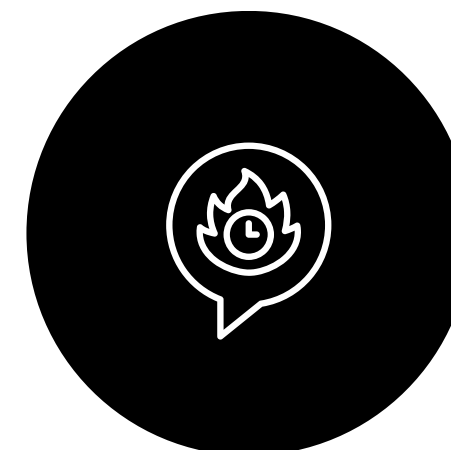
XÂY DỰNG & DI TRÚ HỆ  
THỐNG

PHÁT TRIỂN MICRO SOCIAL MEDIA VÀ THỰC  
HIỆN CHUYỂN ĐỔI CÔNG NGHỆ LINH HOẠT



ĐÁNH GIÁ PHƯƠNG PHÁP  
LUẬN

PHÂN TÍCH VAI TRÒ CỦA PROMPT DRIVEN  
DEVELOPMENT VÀ HIỆU QUẢ KỸ THUẬT THỰC TẾ



PHẠM VI GIỚI HẠN

TẬP TRUNG VÀO KHẢ NĂNG THÍCH ỨNG VÀ TRIỂN  
Khai, KHÔNG ĐI SÂU VÀO BẢO MẬT NÂNG CAO

# PHƯƠNG PHÁP TIẾP CẬN

**Vibe coding là gì?**

**ƯU TIÊN Ý ĐỊNH (INTENT) HƠN CÚ PHÁP (SYNTAX)**


**VIBE CODING LÀ PHONG CÁCH LẬP TRÌNH MỚI, NƠI LẬP TRÌNH VIÊN KHÔNG CẦN VIẾT TỪNG DÒNG MÃ CHI TIẾT. THAY VÀO ĐÓ, HỌ MÔ TẢ Ý TƯỞNG VÀ YÊU CẦU BẰNG NGÔN NGỮ TỰ NHIÊN.**

**GITHUB COPILOT AGENT MODE ĐÓNG VAI TRÒ NHƯ MỘT CỘNG TÁC VIÊN KỸ THUẬT, TỰ ĐỘNG CHUYỂN ĐỔI MÔ TẢ THÀNH MÃ NGUỒN HOÀN CHỈNH VÀ THỰC THI TRÊN NHIỀU FILE.**




# CODING TRUYỀN THỐNG VỚI VIBE CODING

### CODING vs VIBE CODING



- Structure
- Logic
- Scalability



- Flow
- Speed
- Creativity

Tiêu chí	Coding truyền thống	Vibe Coding
Vai trò con người	Viết mã chi tiết	Định hướng, kiểm soát
Vai trò công cụ	IDE, thư viện	AI Agent (Copilot)
Cách tiếp cận	Code-first	Prompt-first
Thời gian phát triển	Dài, phụ thuộc kinh nghiệm	Nhanh hơn, linh hoạt
Khả năng di trú công nghệ	Khó, tốn công	Dễ hơn nhờ AI hỗ trợ

# VAI TRÒ CỦA GITHUB COPILOT TRONG DỰ ÁN

**TÁC NHÂN HỖ TRỢ (AGENT MODE):** GITHUB COPILOT KHÔNG CHỈ GỢI Ý CODE MÀ CÒN CÓ KHẢ NĂNG HIỂU TOÀN BỘ NGỮ CẢNH DỰ ÁN, TỰ ĐỀ XUẤT CẤU TRÚC FILE VÀ THỰC HIỆN CÁC THAY ĐỔI PHỨC TẠP TRÊN NHIỀU TẬP TIN CÙNG LÚC.

**HỖ TRỢ DI TRÚ CÔNG NGHỆ:** COPILOT ĐÓNG VAI TRÒ THEN CHỐT TRONG VIỆC CHUYỂN ĐỔI (MIGRATION) MÃ NGUỒN TỪ PYTHON SANG JAVA SPRING BOOT HOẶC TỪ REACT SANG BLAZOR BẰNG CÁCH PHÂN TÍCH LOGIC CŨ VÀ TÁI CẤU TRÚC SANG NGÔN NGỮ MỚI.

**TỰ ĐỘNG HÓA CÁC TÁC VỤ LẶP LẠI:** SINH CÁC MÃ BOILERPLATE, CẤU HÌNH DOCKERFILE, VÀ TẠO CÁC BỘ DỮ LIỆU MẪU ĐỂ KIỂM THỬ.

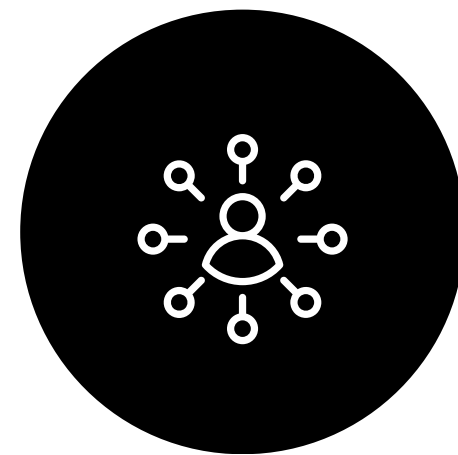
# PROMPT DRIVEN DEVELOPMENT

PROMPT DRIVEN DEVELOPMENT LÀ KỸ THUẬT PHÁT TRIỂN PHẦN MỀM DỰA TRÊN VIỆC THIẾT KẾ PROMPT RÕ RÀNG, CÓ CẤU TRÚC ĐỂ ĐIỀU KHIỂN AI SINH RA KẾT QUẢ MONG MUỐN.

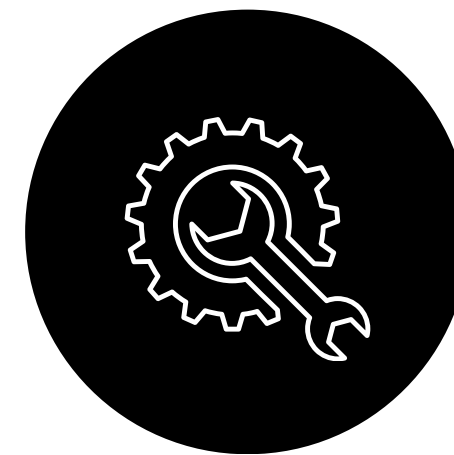
MỘT PROMPT HIỆU QUẢ THƯỜNG BAO GỒM:



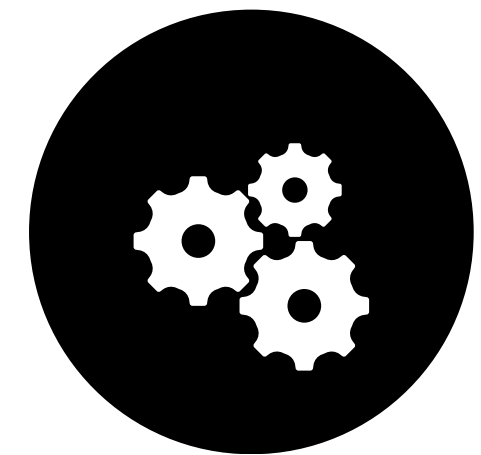
MÔ TẢ BỐI CẢNH  
(CONTEXT)



YÊU CẦU CHỨC NĂNG  
(WHAT)

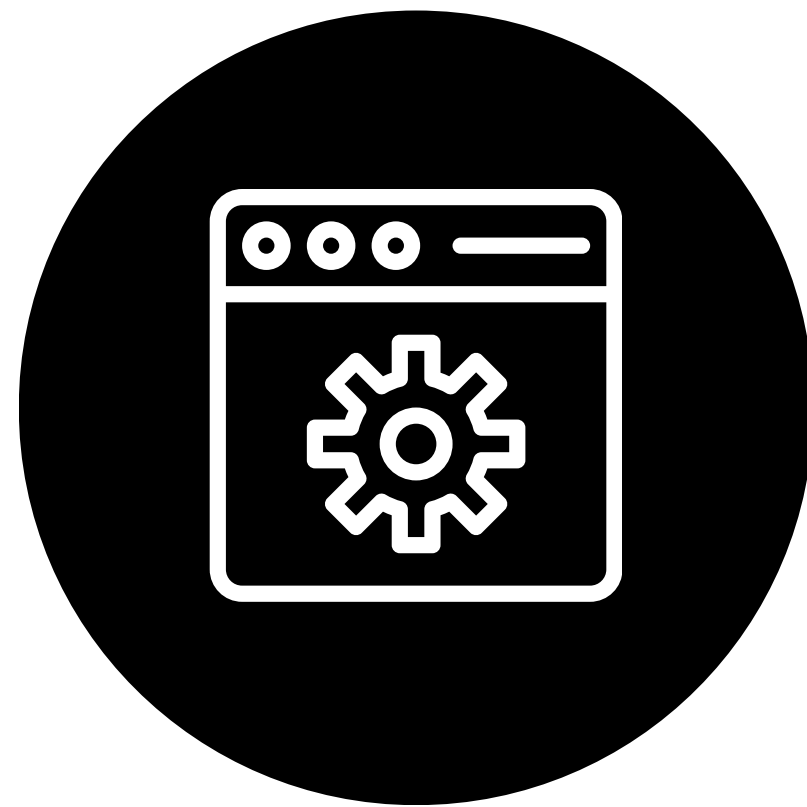


RÀNG BUỘC KỸ THUẬT  
(CONSTRAINTS)



CÔNG NGHỆ SỬ DỤNG  
(TECH STACK)

# PHÂN TÍCH HỆ THỐNG



COPILOT TRONG PHÂN  
TÍCH NGHIỆP VỤ

SỬ DỤNG COPILOT ĐỂ XỬ LÝ FILE PRODUCT-REQUIREMENTS.MD.

## TÓM TẮT YÊU CẦU CHÍNH

COPILOT GIÚP TRÍCH XUẤT CÁC MỤC TIÊU QUAN TRỌNG CỦA CÔNG TY CONTOSO, CHẴNG HẠN NHƯ XÂY DỰNG MỘT NỀN TẢNG MẠNG XÃ HỘI THU NHỎ ĐỂ QUẢNG BÁ SẢN PHẨM NGOÀI TRỜI.

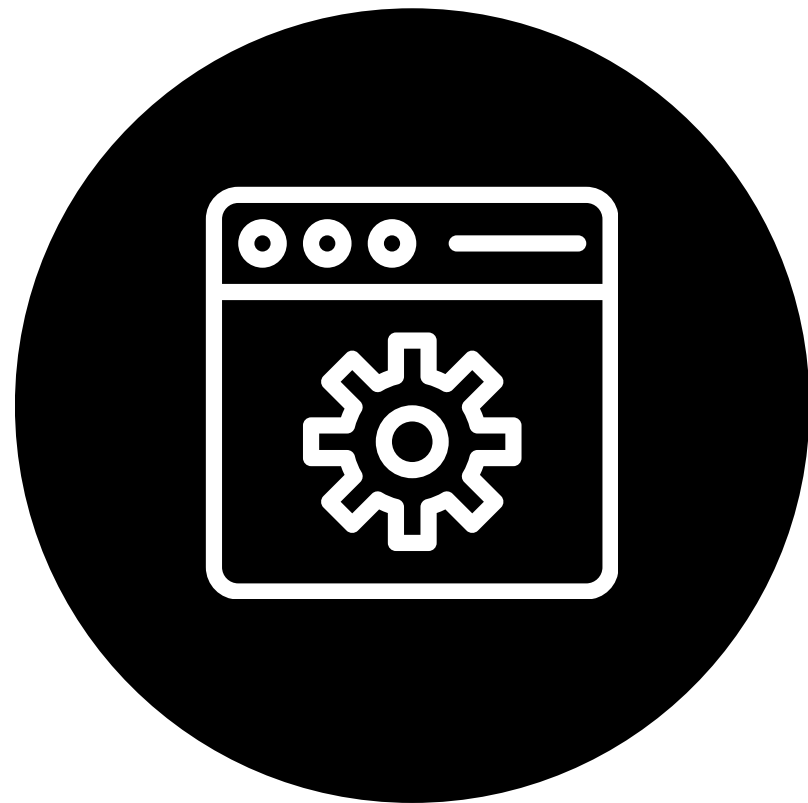
## XÁC ĐỊNH CÁC THỰC THỂ NGHIỆP VỤ

AI HỖ TRỢ LIỆT KÊ CÁC ĐỐI TƯỢNG CỐT LÕI CẦN QUẢN LÝ NHƯ: NGƯỜI DÙNG (USER), BÀI VIẾT (POST), BÌNH LUẬN (COMMENT).



# PHÂN TÍCH HỆ THỐNG

## Giai đoạn 1: Khởi tạo hệ thống



COPILOT TRONG THIẾT KẾ  
HỆ THỐNG

### MỤC TIÊU

XÂY DỰNG NHANH PHIÊN BẢN HOẠT ĐỘNG (MVP) CHO MICRO  
SOCIAL MEDIA.

### CÔNG NGHỆ

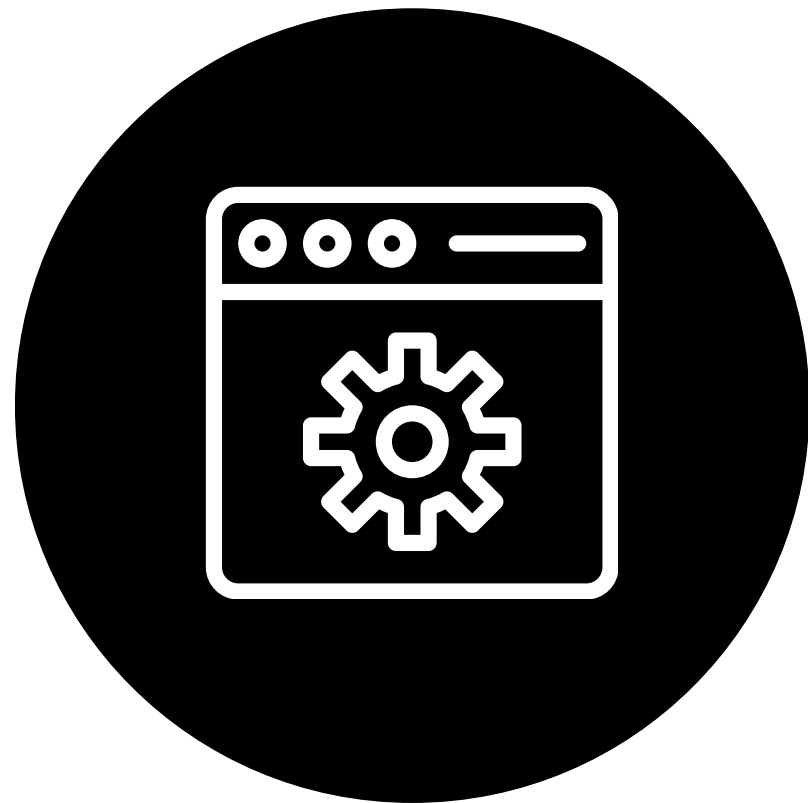
PYTHON (FASTAPI) & JAVASCRIPT (REACT).

### VAI TRÒ COPILOT

HỖ TRỢ SINH MÃ NGUỒN NHANH, ĐỀ XUẤT CẤU TRÚC DỰ ÁN  
VÀ XÂY DỰNG CÁC TÍNH NĂNG NỀN TẢNG.

# PHÂN TÍCH HỆ THỐNG

## Giai đoạn 2: Di trú công nghệ



COPILOT TRONG THIẾT KẾ  
HỆ THỐNG

### MỤC TIÊU

ĐÁNH GIÁ KHẢ NĂNG HỖ TRỢ CỦA GITHUB COPILOT AGENT  
MODE TRONG VIỆC CHUYỂN ĐỔI HỆ THỐNG

### CÔNG NGHỆ

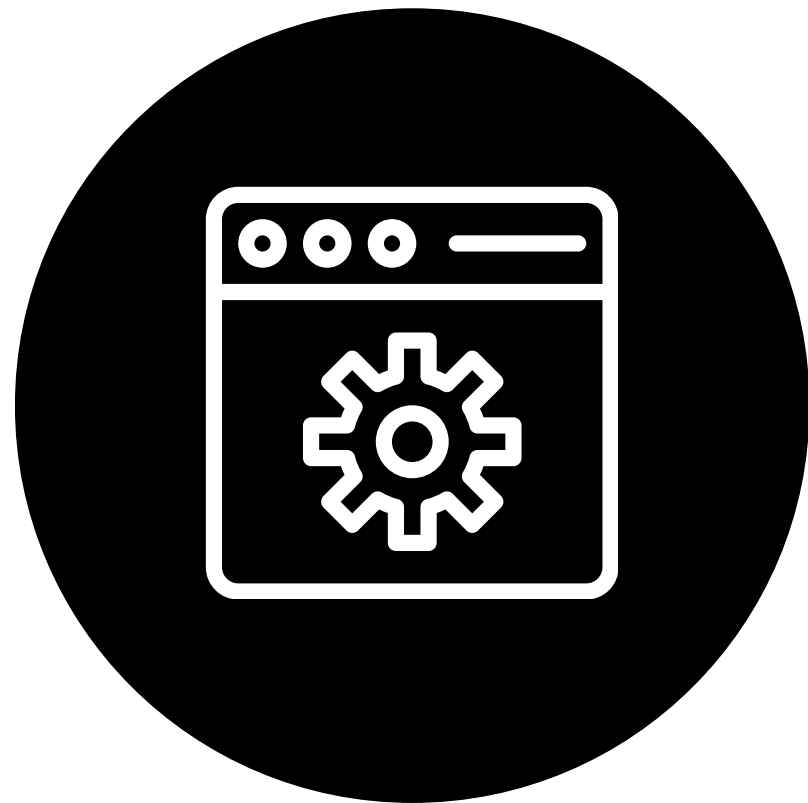
BACKEND: FASTAPI → JAVA SPRING BOOT.  
FRONTEND: REACT → .NET BLAZOR.

### VAI TRÒ COPILOT

PHÂN TÍCH MÃ NGUỒN CŨ, ĐỀ XUẤT CẤU TRÚC TƯƠNG ĐƯƠNG  
VÀ TỰ ĐỘNG SINH MÃ CHUYỂN ĐỔI.

# PHÂN TÍCH HỆ THỐNG

## Thiết kế Containerization



COPILOT TRONG THIẾT KẾ  
HỆ THỐNG

### THỰC THI

SỬ DỤNG DOCKER ĐỂ ĐÓNG GÓI TOÀN BỘ THÀNH PHẦN  
(FRONTEND, BACKEND, DATABASE)

### HỖ TRỢ TỪ COPILOT

TỰ ĐỘNG TẠO DOCKERFILE VÀ DOCKER-COMPOSE.YML

### LỢI ÍCH

ĐẢM BẢO TÍNH NHẤT QUÁN GIỮA CÁC MÔI TRƯỜNG VÀ SẴN  
SÀNG CHO MỞ RỘNG

# TRIỂN KHAI BẰNG VIBE CODING VỚI COPILOT



BACKEND PYTHON FASTAPI

## MỤC TIÊU

HIỆN THỰC HÓA NHANH YÊU CẦU NGHIỆP VỤ VÀ KIỂM CHỨNG THIẾT KẾ API.

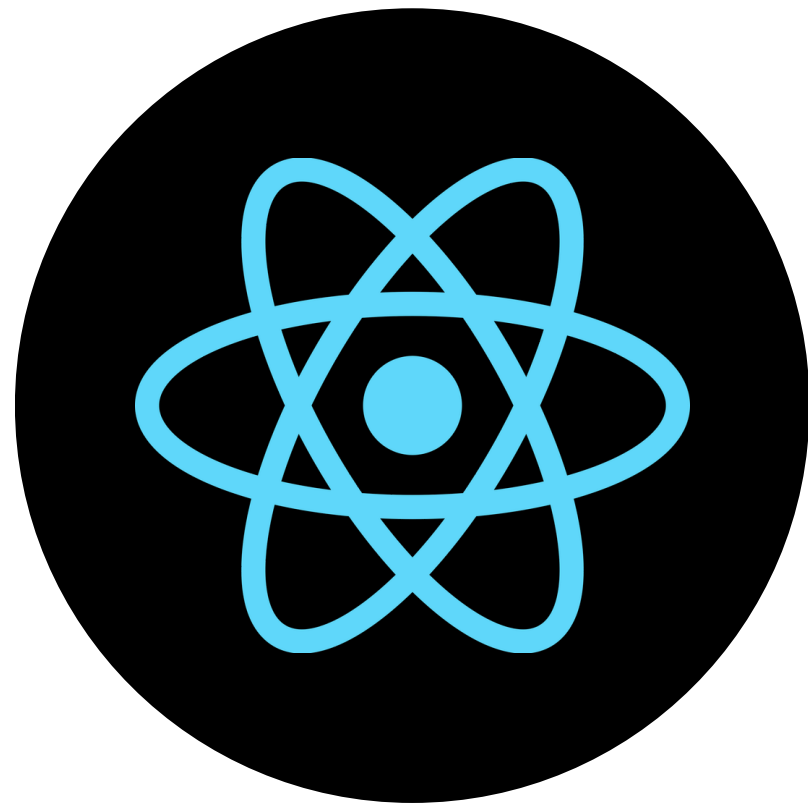
## CÁCH TIẾP CẬN VIBE CODING

LẬP TRÌNH VIÊN MÔ TẢ MỤC TIÊU, RÀNG BUỘC QUA PROMPT. COPILOT HỖ TRỢ SINH CẤU TRÚC DỰ ÁN VÀ ENDPOINT TỪ TÀI LIỆU OPENAPI.

## KẾT QUẢ

BACKEND CÓ CẤU TRÚC SẠCH, DỄ KIỂM TRA VÀ SẴN SÀNG DI TRÚ.

# TRIỂN KHAI BẢNG VIBE CODING VỚI COPILOT



FRONTEND REACT

## MỤC TIÊU

XÂY DỰNG GIAO DIỆN NGƯỜI DÙNG (UI) VÀ KIỂM CHỨNG LƯỒNG DỮ LIỆU QUA REST API.

## ỨNG DỤNG COPILOT

PHÂN TÍCH THIẾT KẾ, TỰ ĐỘNG SINH CÁC REACT COMPONENTS VÀ LOGIC GỌI API.

## KẾT QUẢ

TẠO RA MỘT FRONTEND CÓ CẤU TRÚC ĐỦ RÕ RÀNG ĐỂ PHỤC VỤ CHO QUÁ TRÌNH DI TRÚ SANG NỀN TẢNG KHÁC

# TRIỂN KHAI BẢNG VIBE CODING VỚI COPILOT



BACKEND SPRING BOOT

## MỤC TIÊU

ĐÁP ỨNG THAY ĐỔI VỀ CÔNG NGHỆ VÀ NGUỒN LỰC HỆ THỐNG.

## CHIẾN LƯỢC AI-ASSISTED MIGRATION

CUNG CẤP TOÀN BỘ NGŨ CẢNH (MÃ NGUỒN CŨ + OPENAPI) CHO COPILOT AGENT MODE.

AI HỖ TRỢ ÁNH XẠ (MAPPING) LOGIC, MODEL DỮ LIỆU SANG SPRING BOOT.

## KẾT QUẢ

SỬ DỤNG OPENAPI NHƯ "HỢP ĐỒNG BẤT BIẾN" ĐỂ HÀNH VI HỆ THỐNG KHÔNG THAY ĐỔI SAU KHI DI TRÚ.

# TRIỂN KHAI BẰNG VIBE CODING VỚI COPILOT



FRONTEND BLAZOR

## MỤC TIÊU

ĐỒNG BỘ HÓA CÔNG NGHỆ VỚI HỆ SINH THÁI .NET CỦA DOANH NGHIỆP.

## ỨNG DỤNG COPILOT

COPILOT PHÂN TÍCH CẤU TRÚC COMPONENT REACT ĐỂ CHUYỂN ĐỔI TƯƠNG ĐƯƠNG SANG BLAZOR

ƯU TIÊN CÁC TÍNH NĂNG NATIVE BLAZOR, CHỈ DÙNG JSINTEROP KHI CẦN THIẾT.

## KẾT QUẢ

GIẢM ĐÁNG KỂ THỜI GIAN HỌC FRAMEWORK MỚI, GIỮ VỮNG ĐỘ ỔN ĐỊNH CỦA GIAO DIỆN.

# TRIỂN KHAI BẰNG VIBE CODING VỚI COPILOT



CONTAINERIZATION

## MỤC TIÊU

SỬ DỤNG DOCKER ĐỂ ĐÓNG GÓI TOÀN BỘ HỆ THỐNG (SPRING BOOT & BLAZOR)

## ỨNG DỤNG COPILOT

ÁP DỤNG MULTI-STAGE BUILD ĐỂ GIẢM DUNG LƯỢNG IMAGE.  
DÙNG DOCKER COMPOSE ĐỂ THIẾT LẬP MẠNG NỘI BỘ VÀ BIẾN MÔI TRƯỜNG.

## KẾT QUẢ

AI ĐÓNG VAI TRÒ TRỢ LÝ DEVOPS, CON NGƯỜI QUYẾT ĐỊNH VỀ KIẾN TRÚC TRIỂN KHAI VÀ BẢO MẬT.



# ĐÁNH GIÁ VÀ KẾT LUẬN

## ĐÁNH GIÁ HIỆU QUẢ CỦA VIBE CODING

- TĂNG TỐC ĐỘ PHÁT TRIỂN BAN ĐẦU
- HỖ TRỢ DI TRÚ CÔNG NGHỆ HIỆU QUẢ
- GIẢM KHỐI LƯỢNG CÔNG VIỆC LẶP LẠI
- TĂNG KHẢ NĂNG THỬ NGHIỆM VÀ THAY ĐỔI

## HẠN CHẾ VÀ THÁCH THỨC

- PHỤ THUỘC VÀO CHẤT LƯỢNG PROMPT
- CẦN KIẾN THỨC NỀN TẢNG VỮNG
- KHÓ KIỂM SOÁT TRONG CÁC CHỨC NĂNG PHỨC TẠP
- CHƯA THAY THẾ HOÀN TOÀN LẬP TRÌNH VIÊN

# ĐÁNH GIÁ VÀ KẾT LUẬN

## BÀI HỌC RÚT RA

- VIBE CODING HIỆU QUẢ NHẤT KHI SỬ DỤNG CHO KHỞI TẠO HỆ THỐNG, SINH MÃ BOILERPLATE VÀ DI TRÚ CÔNG NGHỆ.
- PROMPT CẦN ĐƯỢC THIẾT KẾ RÕ RÀNG, CÓ CẤU TRÚC VÀ ĐẦY ĐỦ BỐI CẢNH ĐỂ ĐẠT KẾT QUẢ TỐT.
- LẬP TRÌNH VIÊN VẪN PHẢI ĐÓNG VAI TRÒ KIỂM SOÁT KIẾN TRÚC, ĐẢM BẢO TÍNH NHẤT QUÁN VÀ CHẤT LƯỢNG HỆ THỐNG.
- OPENAPI ĐÓNG VAI TRÒ QUAN TRỌNG NHƯ MỘT “HỢP ĐỒNG KỸ THUẬT” GIÚP ĐẢM BẢO HÀNH VI HỆ THỐNG KHÔNG THAY ĐỔI SAU KHI DI TRÚ.
- CONTAINERIZATION GIÚP HỆ THỐNG DỄ TRIỂN KHAI, TÁI LẬP VÀ HỖ TRỢ QUÁ TRÌNH PHÁT TRIỂN ĐA MÔI TRƯỜNG.

## HƯỚNG PHÁT TRIỂN TRONG TƯƠNG LAI

- TỐI ƯU HIỆU NĂNG VÀ KHẢ NĂNG MỞ RỘNG (SCALABILITY) CỦA HỆ THỐNG.
- BỔ SUNG CÁC CƠ CHẾ BẢO MẬT NÂNG CAO NHƯ OAuth2, RATE LIMITING, LOGGING VÀ MONITORING.
- TRIỂN KHAI TRÊN MÔI TRƯỜNG CLOUD THỰC TẾ (AWS, AZURE HOẶC GCP).
- TÍCH HỢP CI/CD PIPELINE ĐỂ TỰ ĐỘNG HÓA BUILD, TEST VÀ DEPLOY.
- NGHIÊN CỨU SÂU HƠN VỀ VAI TRÒ CỦA AI TRONG KIỂM THỬ TỰ ĐỘNG (AI-ASSISTED TESTING).

**THANK YOU FOR  
YOUR ATTENTION**

