

Source: C:\Users\Admin\Desktop\CMC\ms1_email\ngestion\api\webhook_app.py

```
"""
webhook_app.py
FastAPI application riêng cho webhook notifications
Chạy trên port riêng (8100) và dùng ngrok tunnel riêng
"""
from fastapi import FastAPI, Request
from fastapi.responses import PlainTextResponse, JSONResponse

app = FastAPI(title="Email Webhook Service")

# Import webhook service (lazy import để tránh circular)
webhook_service_instance = None

def get_webhook_service():
    """Lazy load webhook service"""
    global webhook_service_instance
    if webhook_service_instance is None:
        from core.webhook_service import webhook_service
        webhook_service_instance = webhook_service
    return webhook_service_instance

@app.get("/health")
async def health_check():
    """Health check endpoint"""
    return {"status": "ok", "service": "webhook"}

@app.post("/webhook/notifications")
async def webhook_notifications(request: Request):
    """
    Microsoft Graph webhook notification endpoint
    """
    try:
        # Handle validation request
        validation_token = request.query_params.get("validationToken")
        if validation_token:
            return PlainTextResponse(validation_token, status_code=200)

        # Handle notification
        body = await request.json()
        webhook_service = get_webhook_service()
        result = webhook_service.handle_notification(body)

        return JSONResponse(result, status_code=202)
    except Exception as e:
        return JSONResponse(
            {"status": "error", "error": str(e)},
            status_code=500
        )

@app.get("/webhook/status")
async def webhook_status():
    """Lấy trạng thái webhook service"""
    webhook_service = get_webhook_service()
    return webhook_service.get_status()

if __name__ == "__main__":
    import uvicorn
    uvicorn.run(app, host="0.0.0.0", port=8100)
```