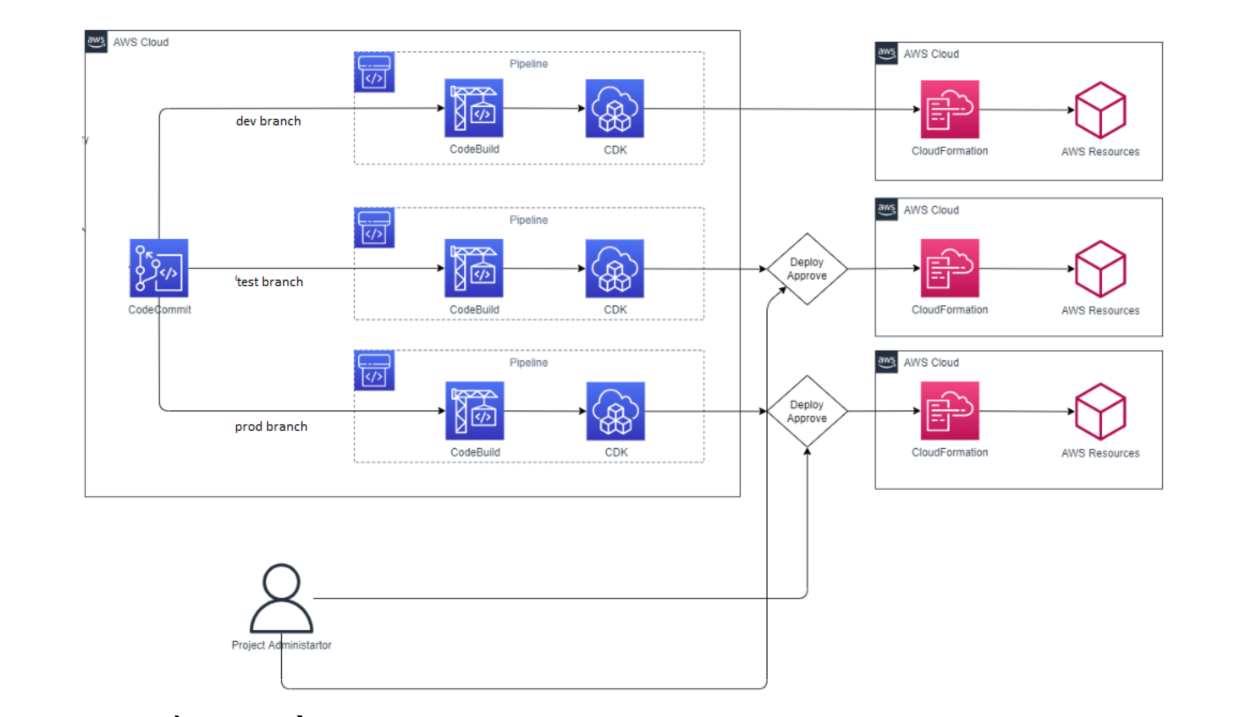
1. TỔNG QUAN DỰ ÁN
   1. Mục tiêu:

- Triển khai một ứng dụng Java Tomcat trên AWS Cloud. Ứng dụng cần chạy trên cả ba môi trường: dev, test và prod.

- Tích hợp Elastic Load Blancer (ELB) để phân phối lương truy cập. Xây dựng CI/CD để triển khai ứng dụng lên EC2.



Hình 1. Diagram dự án

* 1. Công nghệ sử dụng:

- Ngôn ngữ: Python.

- Công cụ triển khai: AWS Cloud Development Kit (CDK).

- IDE: Pycharm.

* 1. Tên dự án: java-world

Được sử dụng để nhận biết tài nguyên trong dự án:

env-projectName-resourceType

Ví dụ: dev-java-world-vpc

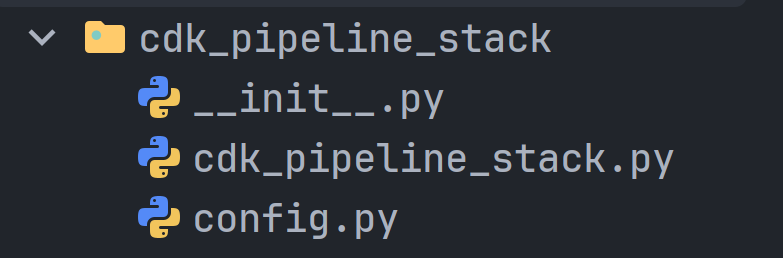
1. TÀI NGUYÊN
   1. CDK Pipeline:

Xây dựng một pipeline cho cdk triển khái các môi trường.

Tạo một folder chứa CDK code dựng pipeline.

- config.py: tập trung thông tin cấu hình cho stack.

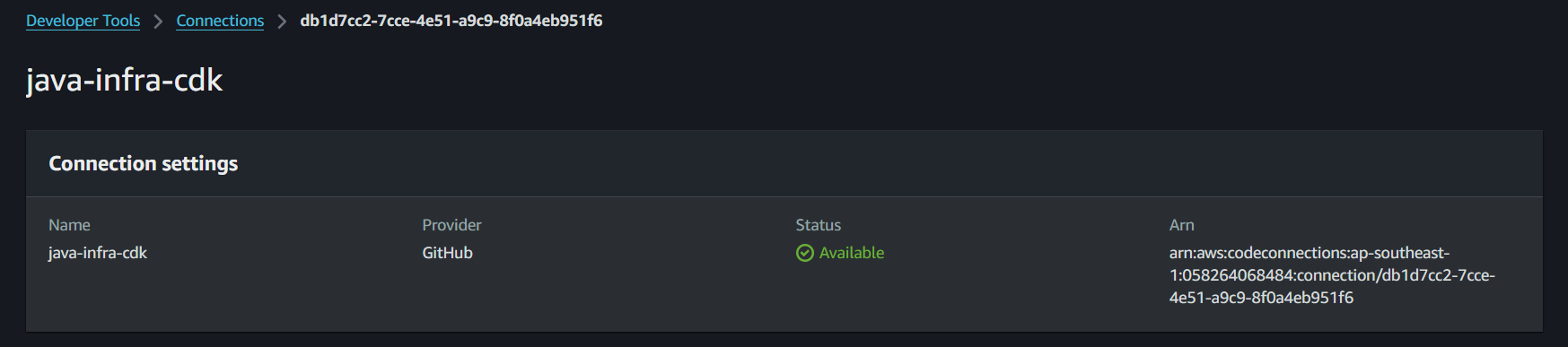
- cdk\_pipeline\_stack.py: chứa code CDK tạo pipeline cho CDK app.



Hình 2. Thư mục code CDK pipeline stack

* 1. Kết nối với Github:

- Tạo kết nối AWS account với Github account.

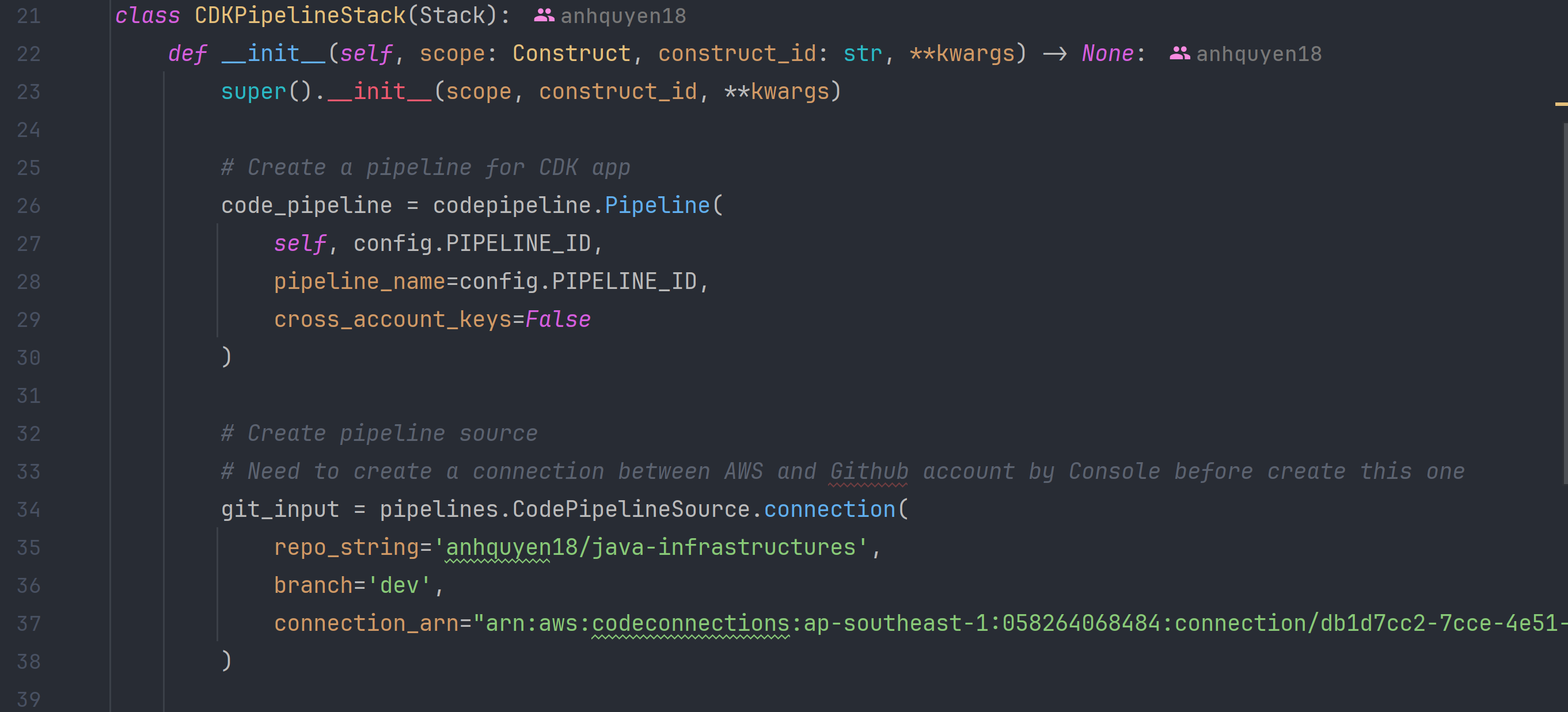


Hình 3. Github connection

- Có thể sử dụng github-token để kết nối, sử dụng Secrets Manager để lưu token và gọi ra khi cần thiết tránh rủi ro về bảo mật.

* + 1. Tạo CodePipeline:

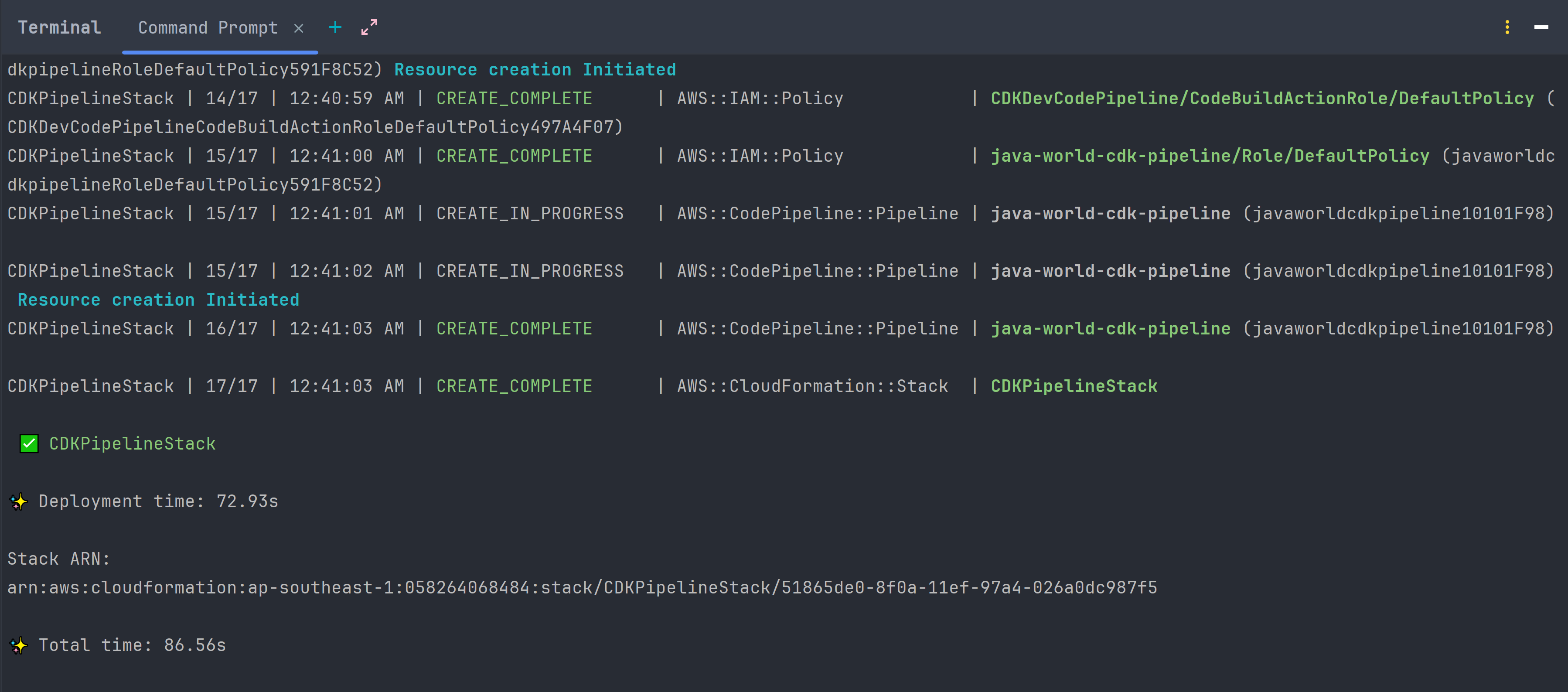
Tạo một pipeline gồm: Source -> Build -> Assets -> Deploy stack



Hình 4. CDK pipeline stack

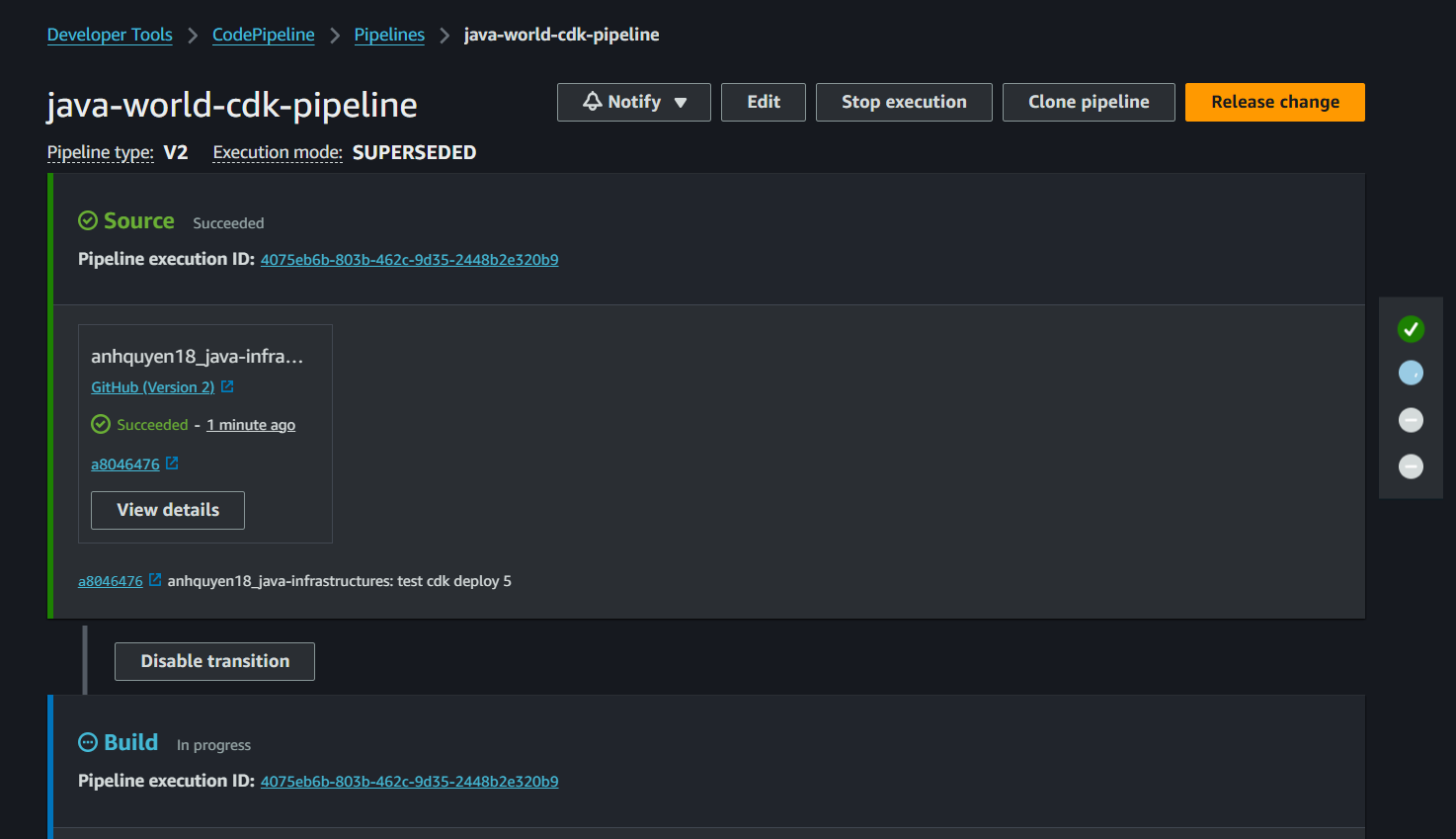
* + 1. Kiểm tra pipeline sau khi deploy:

- Chạy lệnh: cdk deploy --all.



Hình 5. Deploy CDK pipeline stack thành công.

- Kiểm tra pipeline trên môi trường AWS.



Hình 6. CDK pipeline trên console

* 1. Cấu hình network:

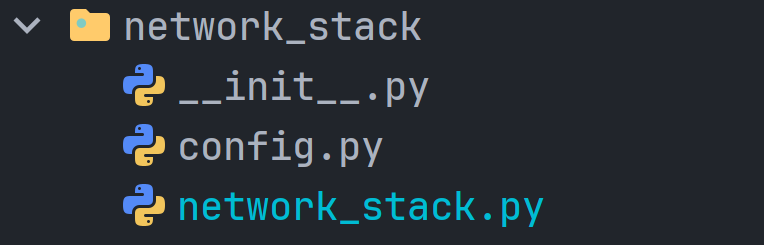
Tạo một VPC tên java-world và các thành phần cần thiết của network như Internet Gateway, Subnet, Route Table…



Hình 7. Network Stack

- config.py: tập trung thông tin cấu hình cho stack.

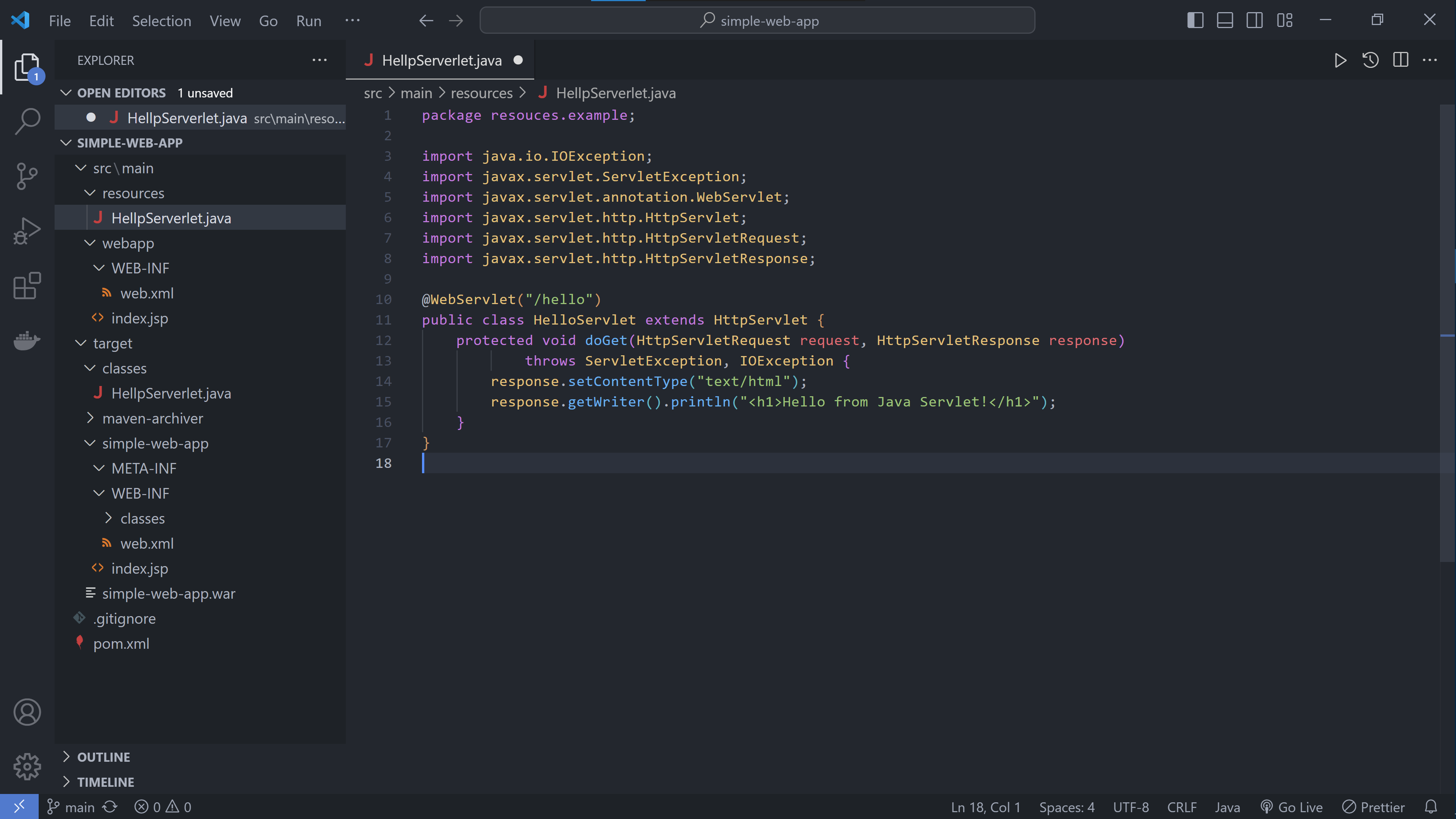
- network\_stack.py: chưa code CDK network stack.



Hình 8. Thư mục code Network stack

* + 1. *Co:*
  1. Tạo dự án Java:

- Tạo một dự án java đơn giản và một github repo gồm 3 branch dev, test, prod dành cho việc kiểm tra sự hoạt động của CI/CD.



Hình 9. Dự án mẫu