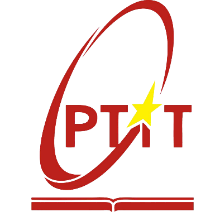
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA AN TOÀN THÔNG TIN**

-------🙞🙜🕮🙞🙜-------



**CÁC KỸ THUẬT GIẤU TIN**

**Đề bài: Tìm hiểu giấu tin bằng phương pháp BPCS với LSB cơ bản**

|  |  |
| --- | --- |
| **HỌ TÊN:** | **Nguyễn Đức Anh** |
| **MÃ SINH VIÊN:** | **B21DCAT032** |

Giảng viên hướng dẫn: PGS.TS. Đỗ Xuân Chợ

**HÀ NỘI, 03/2025**

**MỤC LỤC**

[1. Giới thiệu chung về bài thực hành 5](#_Toc196590796)

[2. Nội dung và hướng dẫn bài thực hành 5](#_Toc196590797)

[1.1. Mục đích 5](#_Toc196590798)

[1.2. Yêu cầu đối với sinh viên 6](#_Toc196590799)

[1.3. Nội dung thực hành 6](#_Toc196590800)

[3. Phân tích yêu cầu của bài thực hành 8](#_Toc196590801)

[4. Cài đặt và cấu hình máy ảo 9](#_Toc196590802)

[5. Tích hợp và triển khai 14](#_Toc196590803)

[5.1. Docker Hub 14](#_Toc196590804)

[5.2. GitHub 14](#_Toc196590805)

[6. Thử nghiệm và đánh giá 15](#_Toc196590806)

**DANH MỤC HÌNH ẢNH**

[Hình 1: Giao diện Labedit 9](#_Toc196590811)

[Hình 2: Cấu hình result 10](#_Toc196590812)

[Hình 3: Cấu hình Result 10](#_Toc196590813)

[Hình 4: Cấu hình Goal 11](#_Toc196590814)

[Hình 5: Docker container bob 11](#_Toc196590815)

[Hình 6: Docker container bob 12](#_Toc196590816)

[Hình 7: Docker container alice 13](#_Toc196590817)

[Hình 8:Docker container alice 13](#_Toc196590818)

[Hình 9: Docker 14](#_Toc196590819)

[Hình 10: Đẩy lab lên github 15](#_Toc196590820)

# **Giới thiệu chung về bài thực hành**

### **1. Giới thiệu chung về bài thực hành**

Trong bối cảnh bảo mật thông tin ngày càng trở nên quan trọng, kỹ thuật giấu tin (steganography) đã trở thành một trong những giải pháp hữu hiệu để che giấu dữ liệu nhạy cảm trong các đối tượng truyền thông như hình ảnh, âm thanh và video. Bài thực hành này hướng dẫn sinh viên triển khai một phương pháp giấu tin trong video số bằng cách sử dụng biến đổi cosine rời rạc (DCT – Discrete Cosine Transform), cụ thể là áp dụng giấu tin trong các hệ số DCT của khung hình video.

Sinh viên sẽ được làm quen với quy trình xử lý video trên môi trường lập trình Python kết hợp với thư viện OpenCV. Quy trình bao gồm việc trích xuất khung hình từ video, thực hiện biến đổi DCT để chuyển dữ liệu hình ảnh từ miền không gian sang miền tần số, sau đó nhúng thông điệp bí mật vào các hệ số DCT ít quan trọng nhằm hạn chế tác động đến chất lượng hình ảnh. Sau khi nhúng thông tin, các khung hình sẽ được tái hợp thành video đầu ra có chứa dữ liệu ẩn.

Ngoài phần giấu tin, bài thực hành cũng yêu cầu sinh viên phát triển hoặc sử dụng các script để giải mã thông tin đã nhúng bằng cách đảo ngược các bước xử lý, từ đó trích xuất chính xác thông điệp ban đầu. Điều này giúp sinh viên hiểu rõ hơn về cả hai phía của kỹ thuật giấu tin: nhúng và trích xuất.

Thông qua bài lab này, sinh viên không chỉ củng cố kiến thức về xử lý ảnh và video, mà còn hiểu rõ cách thức ứng dụng steganography trong việc bảo vệ thông tin nhạy cảm. Đồng thời, sinh viên cũng sẽ rèn luyện được kỹ năng viết mã, phân tích và đánh giá hiệu quả của các kỹ thuật giấu tin trong môi trường thực tế. Đây là bước chuẩn bị quan trọng cho các nghiên cứu và ứng dụng sâu hơn trong lĩnh vực an toàn thông tin, bảo mật đa phương tiện và phân tích dữ liệu ẩn.

# **Nội dung và hướng dẫn bài thực hành**

## **Mục đích**

* Hiểu và thực hành kỹ thuật giấu tin trong miền tần số sử dụng biến đổi cosine rời rạc (DCT).
* Ứng dụng Python và OpenCV để xử lý và thao tác trên khung hình video.
* Thực hiện thao tác nhúng và trích xuất thông tin bí mật trong video mà không làm thay đổi đáng kể nội dung hiển thị.
* Làm quen với quy trình mã hóa và giải mã dữ liệu ẩn trong nội dung truyền thông đa phương tiện.

## **Yêu cầu đối với sinh viên**

* Nắm vững kiến thức cơ bản về mã hóa video và xử lý ảnh số.
* Biết sử dụng dòng lệnh Linux và thao tác trên terminal.
* Có khả năng đọc, hiểu và chỉnh sửa mã nguồn Python.
* Biết cách chạy các lệnh cơ bản với ffmpeg và sử dụng scp để truyền file qua SSH (nếu có yêu cầu).
* Hiểu cấu trúc và nguyên lý hoạt động của kỹ thuật DCT trong xử lý ảnh/video.

## **Nội dung thực hành**

Sinh viên khởi động bài lab

Chạy lệnh:

*labtainer -r stego\_code\_bpcs\_dct*

*(Chú ý: sinh viên sử dụng <TÊN\_TÀI\_KHOẢN> của mình để nhập thông tin người thực hiện bài lab khi có yêu cầu, để sử dụng khi chấm điểm.)*

Sau khi khởi động bài lab,có 2 container hiện lên là bob và alice. Sinh viên thực hiện làm lab theo yêu cầu trên 2 container này

TASK 1: Giấu tin vào khung ảnh video bằng kỹ thuật DCT

Khi container khởi động, một terminal ảo sẽ mở ra. Trong thư mục làm việc đã có sẵn các file sau:

* video.mp4: video gốc
* pbcs\_dct\_embed.py: mã nguồn để giấu tin vào video
* key.txt: lưu thông tin vị trí các bit tin mật đã nhúng

Sinh viên sử dụng Python script sau để thực hiện nhúng tin:

python3 pbcs\_dct\_embed.py

Script sẽ xử lý từng khung hình của video, sử dụng biến đổi DCT để nhúng thông điệp, rồi ghép lại thành video mới có tên:

* output\_stego.mp4: video đã giấu tin
* output\_stego\_noaudio.mp4: phiên bản không có âm thanh, dùng trong xử lý kỹ thuật

TASK 2: Kiểm tra kết quả giấu tin

Sau khi chạy script giấu tin, tiến hành kiểm tra các file đầu ra:

ls

ls -lh output\_stego.mp4 video.mp4

Sinh viên quan sát sự khác biệt về kích thước, kiểm tra sự tồn tại của file đầu ra.

Việc thay đổi kích thước nhỏ chứng tỏ dữ liệu đã được nhúng mà không ảnh hưởng lớn đến nội dung.

TASK 3: Trích xuất thông điệp từ video đã giấu tin

Sinh viên sử dụng Python script pbcs\_dct\_extract.py để giải mã thông điệp từ video đã giấu tin:

python3 pbcs\_dct\_extract.py

Script sẽ:

* Đọc vị trí khung và tọa độ bit từ key.txt
* Phân tích khung hình đã chọn từ video
* Trích xuất lại chuỗi bit chứa thông điệp
* Lưu kết quả hoặc hiển thị nội dung thông điệp ra terminal

Kết thúc lab:

* Trên terminal khởi động lab, sinh viên sử dụng lệnh:

***Stoplab***

* Khi bài lab kết thúc, một tệp lưu kết quả được tạo và lưu vào một vị trí được hiển thị bên dưới stoplab. Sinh viên cần nộp file .lab để chấm điểm.
* Để kiểm tra kết quả khi trong khi làm bài thực hành sử dụng lệnh:

***checkwork <tên bài thực hành>***

* Sinh viên cần nộp file .lab để chấm điểm.
* Kiểm tra kết quả trong quá trình làm bài:

***checkwork <tên bài lab>***

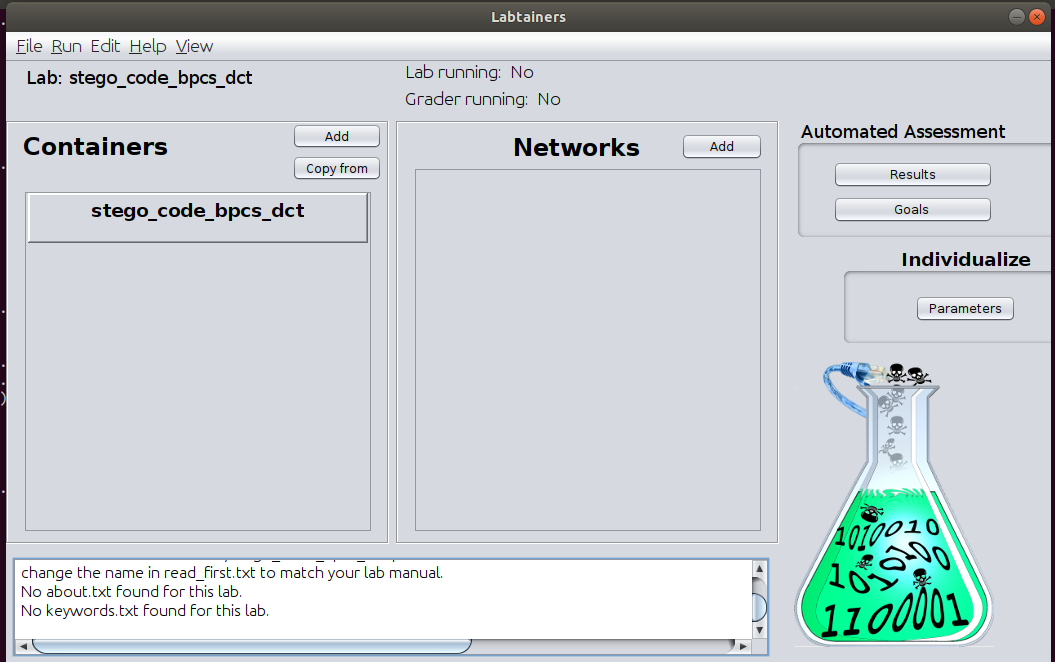
* Khởi động lại bài lab: Trong quá trình làm bài sinh viên cần thực hiện lại bài lab, dùng câu lệnh:

***labtainer -r stego\_code\_bpcs\_dct***

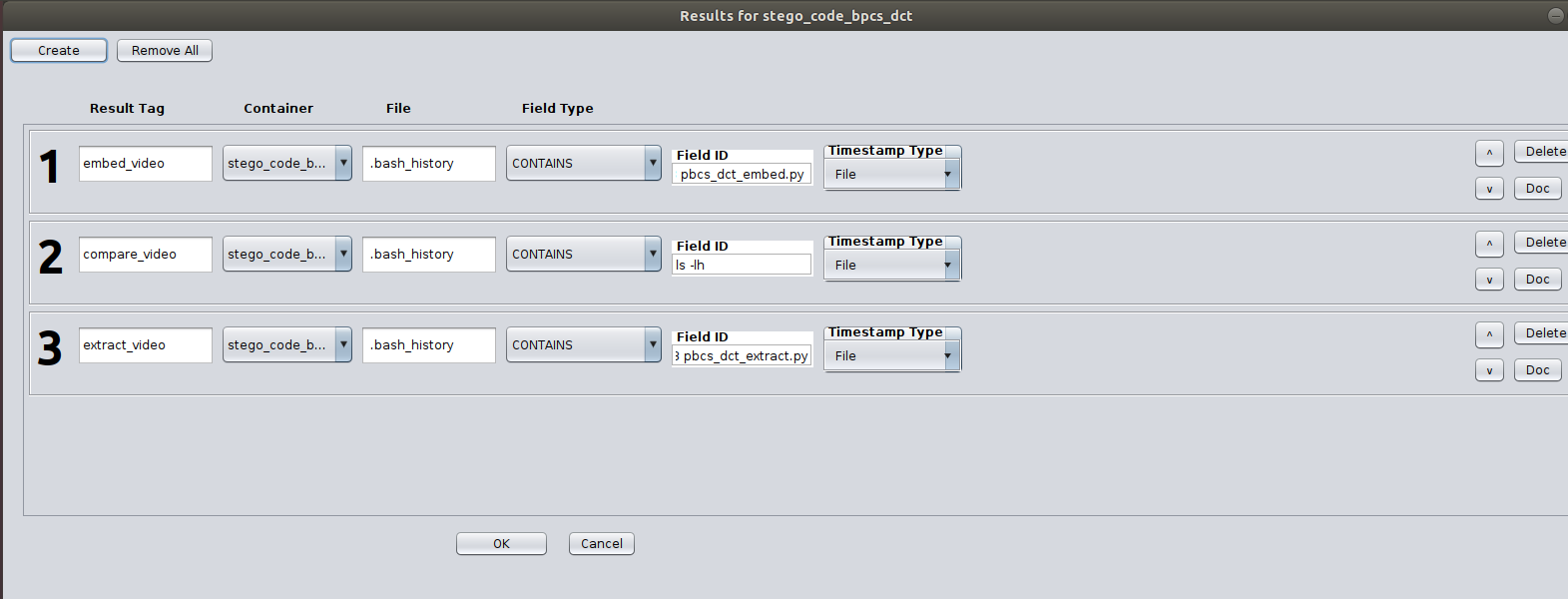
# **Phân tích yêu cầu của bài thực hành**

Bài thực hành gồm 1 máy tính ubuntu, trong đó chứa 1 video mẫu dạng mp4, 1 code giấu tin và 1 code giải mã. Để hoàn thành bài thực hành , sinh viên cần thực hiện giấu tin , giải mã và đọc tin đã giấu.

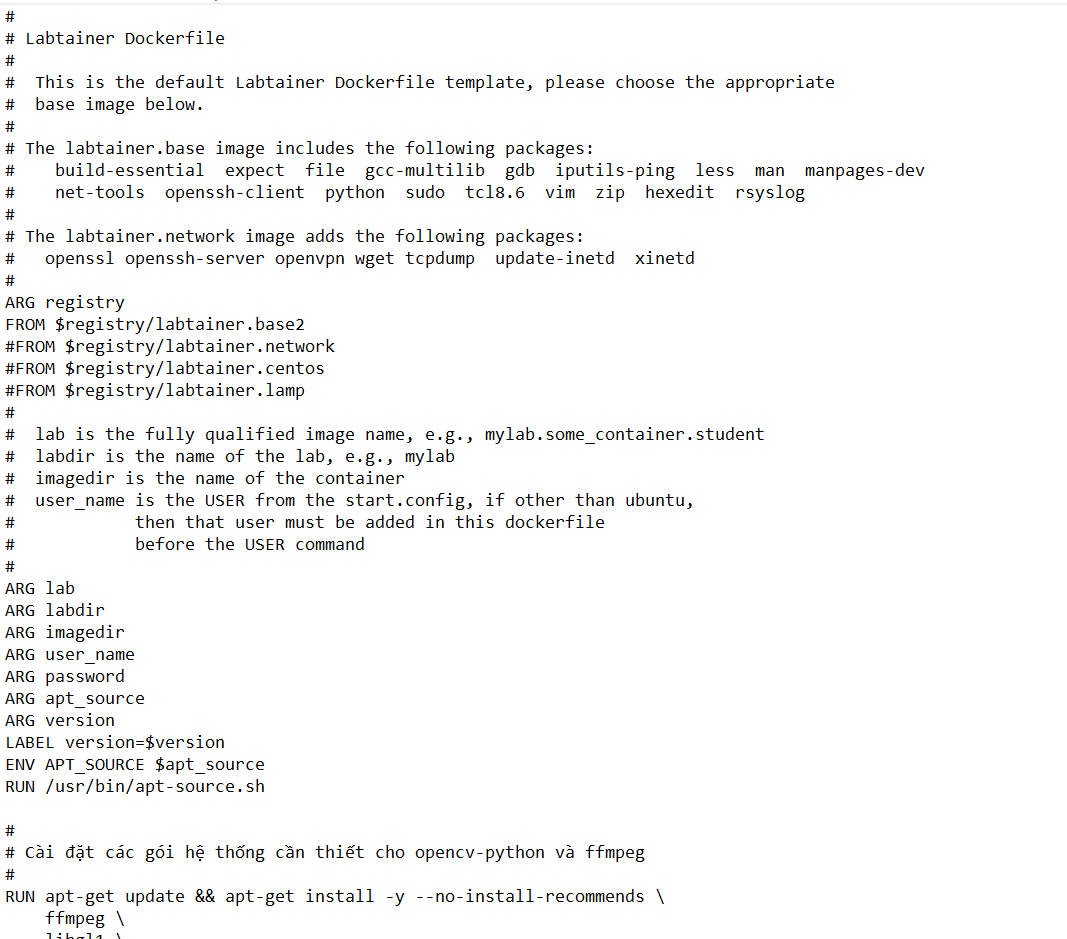
# **Cài đặt và cấu hình máy ảo**



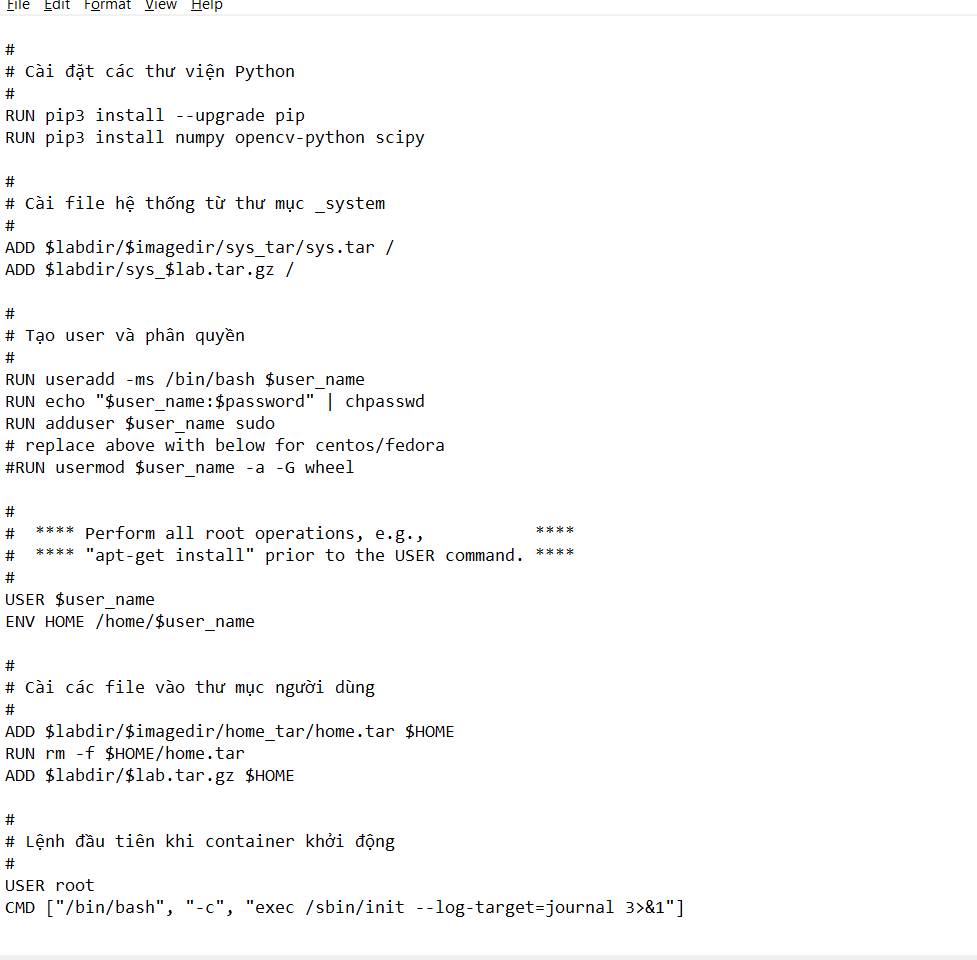
Hình 1: Giao diện Labedit



Hình 2: Cấu hình Result



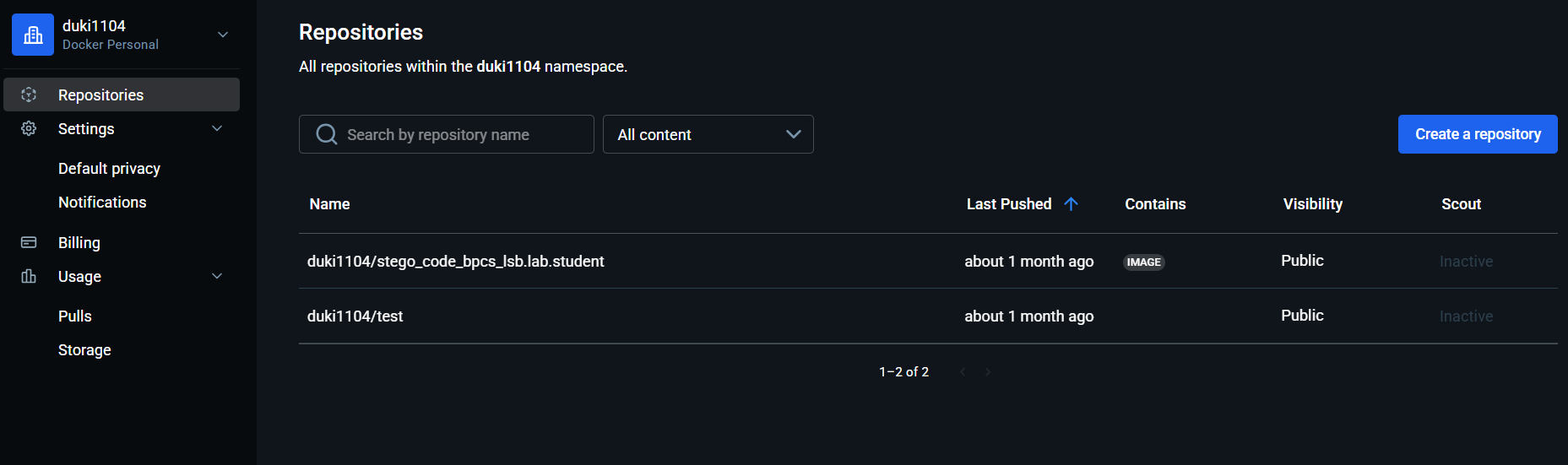
Hình 3: Docker container ubuntu



Hình 4: Docker container ubuntu

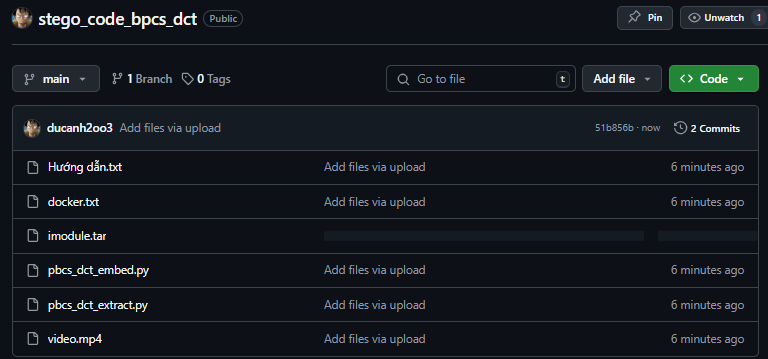
# **Tích hợp và triển khai**

## **Docker Hub**



Hình 9: Docker

## **GitHub**



Hình 10: Đẩy lab lên github

# **Thử nghiệm và đánh giá**

* Tải bài lab về từ Github và chạy bài lab:
* Thực hiện Task 1:
* Thực hiện task 2:
* Thực hiện task 3:
* Thực hiện task 4:
* Checkwork: