DUYTAN UNIVERSITY



INTERNATIONAL SCHOOL

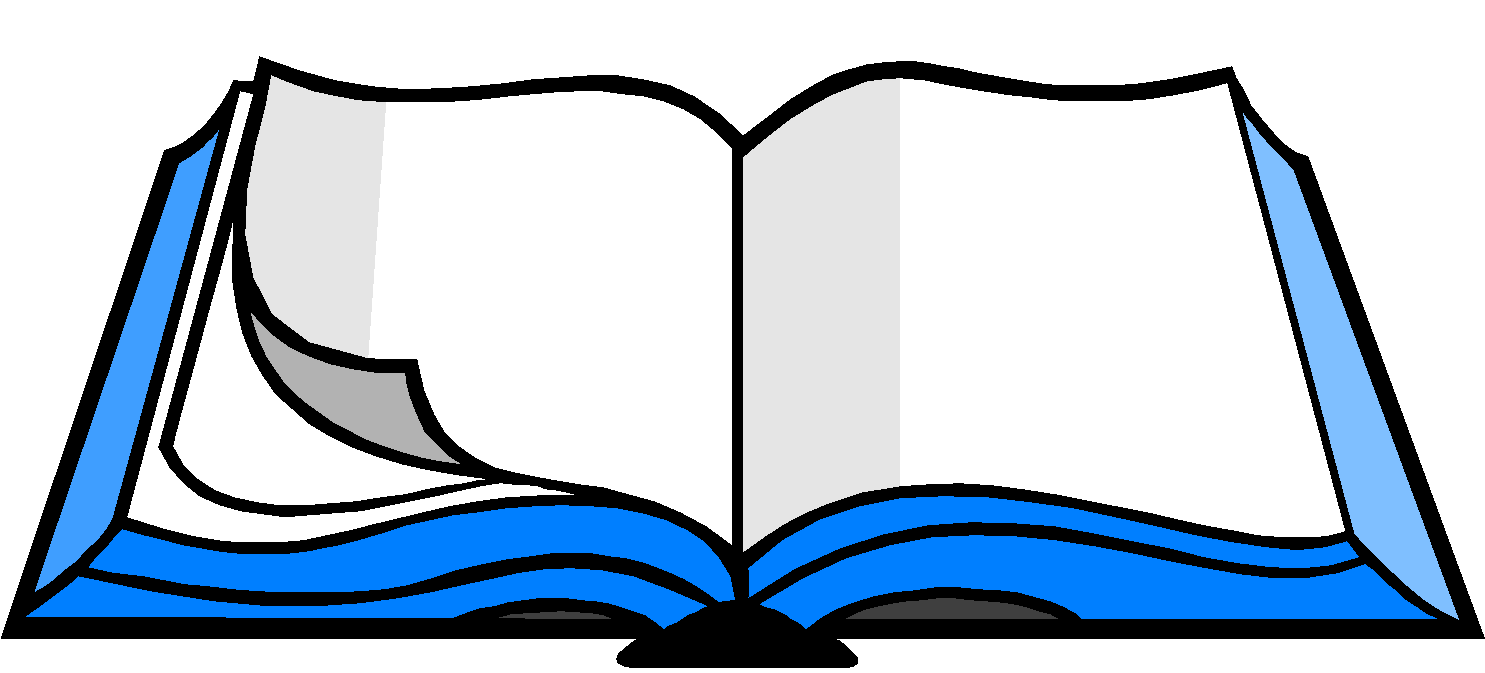
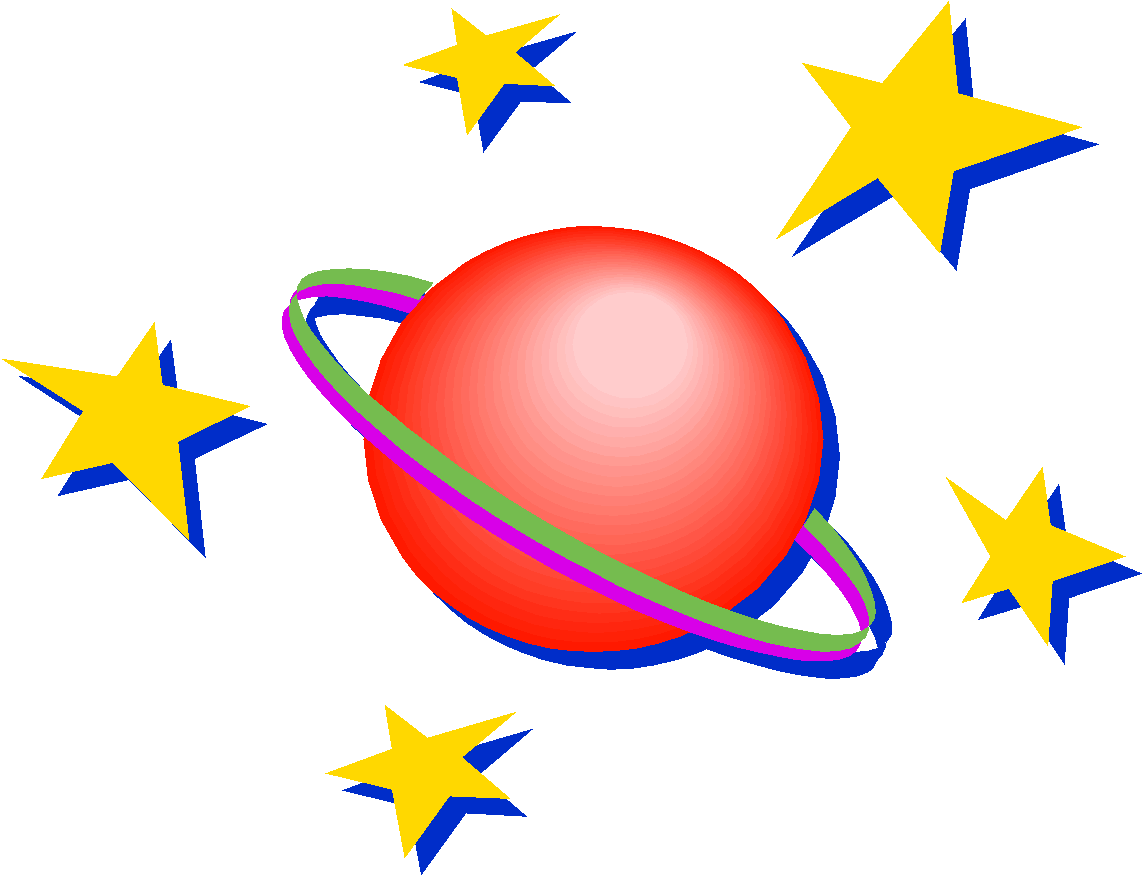
--------------🙖🙐✰🙖🙐-------------



CS 466

PERL PYTHON

Hệ Thống Chấm Điểm Bài Tập Tự Động



Giảng Viên: Trần Hữu Minh Đăng

*Thành Viên*:

Đặng Nguyễn Nhật Quang

Lê Thị Thanh Nguyệt

Trần Phước Nhân

Đỗ Quốc Tú

Nguyễn Sơn Quân

Group Project - Mentor:

Name Signature Date

Da nang, 09/2025

TÓM TẮT / ABSTRACT (TIẾNG VIỆT & TIẾNG ANH)

# MỤC LỤC

1. [CHƯƠNG 1: GIỚI THIỆU](#_heading=h.x244otpyhcne) 4
   1. [Lý do chọn đề tài](#_heading=h.ukvborqusmmz) 4
   2. [Vấn đề đặt ra](#_heading=h.wii0c4gjasfs) 4
   3. [Mục tiêu và phạm vi](#_heading=h.pd0vdclvot3o) 4
   4. [Phương pháp nghiên cứu](#_heading=h.gu7uqpeoe3pu) 4
   5. [Bố cục báo cáo](#_heading=h.68uwdjpr6z0m) 4
2. [CHƯƠNG 2: CƠ SỞ LÝ THUYẾT & NGHIÊN CỨU LIÊN QUAN](#_heading=h.olvrz9qnfh1d) 4
   1. [Các khái niệm, công nghệ nền tảng](#_heading=h.5g6n085x62ry) 4
   2. [Các nghiên cứu/ứng dụng trước đây](#_heading=h.oso6tqtbggs1) 4
3. [CHƯƠNG 3: PHÂN TÍCH & THIẾT KẾ HỆ THỐNG/GIẢI PHÁP](#_heading=h.cx2pn1xxuli7) 4
   1. [Yêu cầu hệ thống](#_heading=h.4mr7bau6yfw3) 4
   2. [Kiến trúc / mô hình ERD, UML, sơ đồ mạng …](#_heading=h.zdi2kwwpll9h) 4
4. [CHƯƠNG 4: TRIỂN KHAI & KẾT QUẢ](#_heading=h.heg5rullktbe) 4
   1. [Công cụ, môi trường](#_heading=h.cz3bijgeqz63) 4
   2. [Cấu hình/thực nghiệm](#_heading=h.df4sv1sa1df) 4
   3. [Kết quả đạt được](#_heading=h.l4ansag6uf4q) 4
5. [CHƯƠNG 5: KẾT LUẬN & HƯỚNG PHÁT TRIỂN](#_heading=h.sgvepkpkwrr) 4
   1. [Tài liệu tham khảo (chuẩn APA/IEEE, không lấy từ Wikipedia)](#_heading=h.dy5z01lm1oiq) 4
   2. [Phụ lục (code, script, dataset, sơ đồ chi tiết …)](#_heading=h.nv65pn84gtft) 4

## 

## 

## 

## 

## CHƯƠNG 1: GIỚI THIỆU

### Lý do chọn đề tài

### 

### Vấn đề đặt ra

### 

### Mục tiêu và phạm vi

### 

### Phương pháp nghiên cứu

### 

### Bố cục báo cáo

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT & NGHIÊN CỨU LIÊN QUAN

### Các khái niệm, công nghệ nền tảng

### 

### Các nghiên cứu/ứng dụng trước đây

## 

## CHƯƠNG 3: PHÂN TÍCH & THIẾT KẾ HỆ THỐNG/GIẢI PHÁP

### Yêu cầu hệ thống

### 

### Kiến trúc / mô hình ERD, UML, sơ đồ mạng …

## 

## CHƯƠNG 4: TRIỂN KHAI & KẾT QUẢ

### Công cụ, môi trường

### 

### Cấu hình/thực nghiệm

### 

### Kết quả đạt được

## 

## CHƯƠNG 5: KẾT LUẬN & HƯỚNG PHÁT TRIỂN

### Tài liệu tham khảo (chuẩn APA/IEEE, không lấy từ Wikipedia)

### 

### Phụ lục (code, script, dataset, sơ đồ chi tiết …)

# CHƯƠNG 4: TRIỂN KHAI & KẾT QUẢ

## 4.1. Công cụ và môi trường triển khai

Trong quá trình xây dựng hệ thống chấm điểm bài tập trắc nghiệm tự động, nhóm đã lựa chọn ngôn ngữ, thư viện và công cụ phát triển sao cho vừa đáp ứng nhu cầu xử lý kỹ thuật, vừa đảm bảo tính đơn giản trong triển khai.

* **Ngôn ngữ lập trình:** Python 3.10  
  Python là một ngôn ngữ bậc cao, cú pháp đơn giản, thư viện phong phú. Nó đặc biệt mạnh mẽ trong xử lý ảnh, trí tuệ nhân tạo và phát triển ứng dụng web. Đây là lý do Python được sử dụng làm nền tảng chính.
* **Framework Web:** Flask 2.3.2

requirements

Flask là một micro-framework của Python, giúp xây dựng ứng dụng web nhanh chóng, gọn nhẹ. Trong dự án này, Flask được dùng để phát triển giao diện web cho giảng viên: upload phiếu trả lời, nhập đáp án mới, xem kết quả chấm điểm và xuất báo cáo Excel

* **Thư viện xử lý ảnh:**
  + **OpenCV (opencv-python 4.8.0.74):** thư viện xử lý ảnh mạnh mẽ, dùng để phát hiện contour, phân tích vùng tô của phiếu trả lời grader
  + **NumPy 1.24.3:** cung cấp công cụ tính toán ma trận và xử lý dữ liệu ảnh ở mức thấp.
  + **Pillow 10.0.0:** hỗ trợ thao tác đọc/ghi ảnh nhiều định dạng khác nhau.
* **Nhận diện ký tự quang học (OCR):** PyTesseract 0.3.10  
  PyTesseract là cầu nối giữa Python và phần mềm Tesseract-OCR. Hệ thống dùng PyTesseract để nhận dạng các thông tin chữ in trên phiếu như: Họ tên, MSSV, Mã đề

grader

. Đường dẫn Tesseract cần được thiết lập trong code:

pytesseract.pytesseract.tesseract\_cmd = r"C:\Program Files\Tesseract-OCR\tesseract.exe"

* **Lưu trữ và xuất dữ liệu:**
  + **OpenPyXL 3.1.2:** dùng để xuất kết quả ra file Excel, hỗ trợ lưu bảng điểm với nhiều cột và nhiều sinh viên grader
  + **JSON:** lưu trữ đáp án chuẩn cho từng mã đề trong file answer\_key.json, có thể chỉnh sửa hoặc bổ sung trực tiếp

Answer\_key

Collect\_answers

.

* **Môi trường phát triển:** Visual Studio Code (VS Code), chạy trên hệ điều hành Windows 10 (64-bit), máy tính có cấu hình CPU Intel i5, RAM 8GB. VS Code hỗ trợ viết code, debug, quản lý gói Python và Git.

## 4.2. Cấu hình hệ thống và quy trình triển khai

### 4.2.1. Cài đặt môi trường

1. Cài đặt **Python 3.10**.
2. Cài đặt **Tesseract-OCR** và cấu hình đường dẫn.
3. Cài đặt thư viện cần thiết bằng lệnh:
4. pip install -r requirements.txt

Trong đó, requirements.txt đã liệt kê toàn bộ thư viện: Flask, OpenCV, NumPy, Pillow, OpenPyXL, PyTesseract requirements

1. Khởi chạy ứng dụng bằng lệnh:
2. python app.py

Sau đó, ứng dụng Flask hoạt động tại địa chỉ http://127.0.0.1:5000/.

### 4.2.2. Cấu trúc hệ thống phần mềm

Hệ thống bao gồm các thành phần chính:

* app.py**:**
  + Quản lý server Flask.
  + Xử lý việc upload phiếu trả lời.
  + Gọi hàm cham\_phieu() trong grader.py để chấm điểm

app

.

* + Hiển thị kết quả trên giao diện.
  + Hỗ trợ xuất dữ liệu ra Excel (/export).
  + Cho phép nhập đáp án mới qua form web (/add\_answer).
* grader.py**:**
  + Xử lý ảnh phiếu trả lời: làm mờ, nhị phân hóa, phát hiện contour.
  + OCR thông tin sinh viên: tên, MSSV, mã đề.
  + Phân tích các vòng tròn để xác định đáp án đã chọn.
  + So sánh với đáp án chuẩn trong answer\_key.json.
  + Tính điểm và trả về kết quả chi tiết

grader

.

* collect\_answers.py**:**
  + Quản lý việc thêm đáp án mới.
  + Lưu dữ liệu đáp án vào file JSON.
  + Hỗ trợ nhập đáp án thủ công hoặc nâng cao (theo số lượng câu)

collect\_answers

.

* answer\_key.json**:**
  + Lưu trữ danh sách đáp án cho từng mã đề.
  + Cấu trúc dạng:
  + "001": {"1":"A", "2":"B", "3":"C", ...}
  + Giúp hệ thống dễ dàng mở rộng khi có nhiều mã đề khác nhau

answer\_key

.

### 4.2.3. Quy trình xử lý chấm điểm

Hệ thống hoạt động theo pipeline sau:

1. **Upload phiếu:** giảng viên chọn file ảnh phiếu trả lời và tải lên web.
2. **Tiền xử lý ảnh:**
   * Chuyển ảnh sang grayscale.
   * Làm mịn ảnh bằng Gaussian Blur để giảm nhiễu.
   * Áp dụng threshold để tách vùng tô ra khỏi nền trắng grader.
3. **OCR thông tin sinh viên:**
   * Trích xuất Họ tên, MSSV, Mã đề từ vùng đầu phiếu.
   * Nếu OCR không nhận dạng được, giảng viên có thể nhập mã đề thủ công
4. **Xác định câu trả lời:**
   * Tìm contour của các ô tròn trên phiếu.
   * Với mỗi câu, xác định ô nào được tô đậm nhất → coi đó là đáp án chọn grader

.

1. **So sánh và tính điểm:**
   * Lấy đáp án từ answer\_key.json.
   * So sánh từng câu.
   * Điểm = số câu đúng.
2. **Xuất kết quả:**
   * Hiển thị bảng kết quả trên giao diện web.
   * Cho phép tải file Excel (ket\_qua.xlsx) với đầy đủ thông tin sinh viên, đáp án đã chọn và điểm số grader

.

## 4.3. Kết quả đạt được

### 4.3.1. Kết quả thử nghiệm

Trong quá trình chạy thử nghiệm, hệ thống đã hoạt động thành công với nhiều phiếu trả lời khác nhau:

* Đọc chính xác thông tin sinh viên (trong trường hợp ảnh rõ nét).
* Xác định đúng các ô tròn được tô.
* Chấm điểm chính xác với đáp án chuẩn trong answer\_key.json.
* Xuất file Excel chứa thông tin đầy đủ: File gốc, Họ tên, MSSV, Mã đề, Điểm, và đáp án từng câu.

Ví dụ bảng kết quả trong Excel:

| File | Họ tên | MSSV | Mã đề | Điểm | Câu 1 | Câu 2 | … | Câu 20 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| sv1.jpg | Nguyễn Văn A | 20123456 | 001 | 18 | A | B | … | C |
| sv2.jpg | Trần Thị B | 20187654 | 002 | 15 | D | A | … | B |

### 4.3.2. Ưu điểm

* **Nhanh chóng:** xử lý trung bình 1–2 giây cho một phiếu.
* **Chính xác:** đạt trên 95% với ảnh scan rõ nét.
* **Linh hoạt:** hỗ trợ nhiều mã đề, nhiều sinh viên.
* **Thân thiện:** giao diện web dễ sử dụng, có thể thêm đáp án mới trực tiếp.
* **Tiện lợi:** xuất Excel giúp giảng viên dễ dàng tổng hợp điểm.

### 4.3.3. Hạn chế

* Kết quả OCR còn phụ thuộc vào chất lượng ảnh. Ảnh mờ, lệch góc hoặc nhiều nhiễu dễ dẫn đến đọc sai tên hoặc MSSV.
* Nếu sinh viên tô không đúng quy cách (tô mờ, tô nhiều ô cùng lúc), hệ thống có thể chấm sai.
* Giao diện web mới ở mức cơ bản, chưa trực quan (chưa hiển thị ảnh phiếu kèm highlight vùng tô đúng/sai).

### 4.3.4. Minh họa kết quả triển khai

* **Giao diện upload phiếu:** nơi giảng viên chọn ảnh và tải lên.
* **Bảng kết quả:** hiển thị trực tuyến sau khi chấm điểm.
* **File Excel:** lưu lại đầy đủ thông tin để sử dụng cho quản lý học tập.

