

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



ĐỒ ÁN CHUYÊN NGÀNH  
HƯỚNG CÔNG NGHỆ PHẦN MỀM (CO4029)

---

Báo cáo Đồ án – version 0.1

GRADING SYSTEM

---

Giảng viên hướng dẫn: Lê Đình Thuận

Sinh viên thực hiện: Lê Minh Châu 2010947

Tp. Hồ Chí Minh, Tháng 3/2024



## Mục lục

<b>1</b>	<b>Tổng quan dự án</b>	<b>2</b>
1.1	Bối cảnh . . . . .	2
1.2	Ý tưởng . . . . .	2
1.3	Đánh giá . . . . .	2
1.4	Phương hướng phát triển . . . . .	3

# 1 Tổng quan dự án

## 1.1 Bối cảnh

Gian lận trong thi cử luôn là vấn đề được chú ý. Đối với ngành khoa học máy tính, gian lận có thể là copy code của một ai đó, hoặc sử dụng công cụ (tiêu biểu có thể kể đến AI) trong các bài kiểm tra. Điều này sẽ làm giảm chất lượng giáo dục cũng như chất lượng học viên, và nhiều hệ lụy khác, không chỉ ảnh hưởng cá nhân đó mà cả tổ chức giảng dạy.

Mục tiêu của dự án:

- Giảm thiểu tình trạng gian lận.
- Tăng chất lượng đầu ra của học viên.
- Có thể sử dụng bởi nhiều cơ sở giảng dạy khác nhau
- Có thể sử dụng cho một server cỡ nhỏ

## 1.2 Ý tưởng

Grading System là hệ thống chấm điểm bài kiểm tra dựa trên trong ngành khoa học máy tính Github Action, đồng thời sử dụng những hệ thống đánh giá gian lận được phát triển và sử dụng bởi các tổ chức tin cậy để đánh giá bài làm của học viên.

Hệ thống được hiện thực theo quy trình sau:

1. Sử dụng Github tạo repo đề, thiết kế testcase
2. Học viên fork repo đề, giải quyết bài toán, commit lại (branch/fork ?)
3. Sử dụng Github Action để compile các commit của học viên (thời gian, cách lấy kết quả compile ?)
4. Sử dụng công cụ đánh giá gian lận kiểm tra bài làm hợp lệ (thời gian ?)
5. Hệ thống đánh giá kết quả bài làm
6. Trả về kết quả và thông tin chi tiết (điểm, testcase passed, cheating)

Các bài làm hợp lệ là các bài không bị lỗi khi compile bởi Github Action.

## 1.3 Đánh giá

Quy trình hoạt động của các online judge thường là người dùng sẽ viết code trên text editor, sau khi hoàn thành sẽ trigger function gửi code đến server. Server sẽ chạy IDE để run code với các testcase được tạo sẵn, sau đó trả về kết quả cho người dùng. Tuy nhiên, với một hệ thống nhỏ, việc run code trên IDE có thể khiến

server bị quá tải (tiêu biểu như lỗi Runtime Error sẽ tiêu tốn rất nhiều tài nguyên server nếu như không có/cơ chế chịu lỗi chưa hoàn thiện, hoặc có quá nhiều request cùng lúc).

Đặc điểm của hệ thống là tận dụng khả năng của các hệ thống khác sẵn có và được phát triển qua nhiều năm, được tin tưởng bởi cộng đồng phát triển. Đồng thời giảm tải cho server cục bộ, cũng như công sức phát triển hệ thống nhưng vẫn đạt được chất lượng nhất định. Một số ưu điểm của hệ thống:

- Đáng tin cậy

Hệ thống sử dụng các hệ thống sẵn có như Github, ... để chạy và đánh giá bài làm của học viên. Các hệ thống này thường có quy mô lớn và được cộng đồng tin tưởng. Do đó, có thể đảm bảo được việc hạn chế downtime, đồng thời giảm lỗi từ phía server khi compile bài làm.

- Ổn định, giảm tải cho server

Bài làm của học viên được compile và lưu trữ trên Github. Hệ thống không cần tốn chi phí cho những hành động này. Vì vậy, những tài nguyên đó có thể sử dụng để thực hiện các process khác.

- Hỗ trợ đa ngôn ngữ

Các bài làm có thể sử dụng nhiều loại ngôn ngữ khác nhau, tùy từng học viên. Đối với một số môn học (DSA, ...), ngôn ngữ thường không chiếm vai trò quá quan trọng, do đó, học viên có thể sử dụng ngôn ngữ phù hợp để làm bài sẽ giúp hỗ trợ việc học tốt hơn.

- Đánh giá kết quả ngay sau khi làm bài

Các tác vụ bao gồm compile và lấy kết quả kiểm tra gian lận không tốn quá nhiều thời gian. Do đó, học viên có thể xem kết quả ngay sau khi làm bài, cũng như giải trình về bài làm nếu bị phát hiện gian lận.

- Ngoài ra, học viên cũng có thể trau dồi kiến thức về các nền tảng chia sẻ code như Github.

## 1.4 Phương hướng phát triển

Lúc trước, việc gian lận có thể là copy bài làm, hoặc cùng hợp tác để vượt qua kì thi. Các công cụ hiện có thường có thể đánh giá được do đã được phát triển qua nhiều năm để đối phó với tình trạng này.

Tuy nhiên, hiện nay, rất nhiều công cụ khác rất có tiềm năng để vượt qua được những công cụ đánh giá gian lận hiện tại, tiêu biểu là AI. Dù vậy, hệ thống vẫn có thể khả thi do hiện tại, các đoạn code mà AI sinh ra thường không quá khác biệt. Nếu có một vài học viên sử dụng, hệ thống vẫn có thể nhận biết được. Nhưng tiềm năng của AI đối với việc này hoàn toàn không thể dự đoán được.

Ngoài ra, nếu AI có thể sinh code hoàn toàn khác giữa các học viên, mọi chuyện sẽ phức tạp hơn nhiều. Vì vậy, đây chính là thách thức lâu dài mà hệ thống cần phải phát triển thêm.

Ngoài ra, việc mở rộng khả năng xử lý gian lận sang nhiều lĩnh vực khác cũng là hướng phát triển quan trọng của hệ thống này.



## Tài liệu

- [1] How to use Github Action
- [2] Github Action for multiple language