**Capstone: Practical Examination Evaluation Automation Using DevOps (Chấm bài thi thực hành tự động với DevOps)**

Ứng dụng được phát triển dựa trên việc hỗ trợ việc chấm điểm bài thi thực hành. Hệ thống không hỗ trợ cung cấp một công cụ để tổ chức thi.

Hệ thống đang xây dựng một qui trình là tạo ra một công cụ để hỗ trợ chấm bài thi thực hành. Công cụ cho phép người dùng import đề thi thực hành và kịch bản để đánh giá nội dung bài thi được thực hiện được một giảng viên – tổ trưởng xây dựng.

Công cụ được xây dựng giả định việc canh thi practical test được giám sát bởi giảng viên và công cụ không đảm bảo việc quá trình thi diễn ra như thế nào. Công cụ cũng được xây dựng cho người làm bài, sinh viên để có thể kết nối với ứng dụng của giảng viên để đảm bảo rằng xác định đúng tài khoản được cho phép truy cập đến việc bắt đầu quá trình đưa bài vào hệ thống để đánh giá. Người làm bài sau khi kết nối sẽ thấy được nội dung làm bài, sau đó nộp bài theo định dạng format. Bài được đưa về hệ thống, giải nén và thực thi qui trình chấm bài xây dượng theo DevOps. Kết quả sau khi chấm xong sẽ thực hiện kiểm tra code trùng, code sao chép, … và ghi nhận kết quả để thông báo với giảng viên. Kết quả sẽ được giảng viên sẽ được công bố đến người làm bài. Ứng dụng thực hiện các chức năng như sau: Tạo ra một kỳ đánh giá với các thông tin về người tham gia thi, đề thi và kịch bản đánh giá tương ứng; Cho phép người tham gia kỳ thi đưa bài đánh giá vào hệ thống sau khi được xác thực; Đánh giá bài thi và ghi nhận kết quả; Cho phép tạo kịch bản để đánh giá bài thi; Kiểm tra code trùng và kiểm tra cấu trúc của nội dung làm bài; Kết nối giữa các ứng dụng giữa người tham gia thi, giữa giảng viên và dịch vụ đánh giá bài thi; Xây dựng bài test trên 4 môn C, OOP Java, Desktop Java và Web-based Java Application … Xây dựng qui trình dựa trên nhu cầu thực tế áp dụng công nghệ web services, giao tiếp giữa các thiết bị, cách thức đánh giá bài thi áp dụng DevOps, protocol TCP, UDP, và HTTP để giao tiếp giữa các thiệt bị. Ứng dụng sử dụng cơ chế để hệ thống có thể chạy tự động với các tác vụ để xử lý mà không cần sự tương tác của người sử dụng. Ứng dụng xây dựng trên nền tảng của ứng dụng, xây dựng các api tương tác giúp kết nối các hệ thống với nhau, kiểm tra semantic trong xử lý đánh giá, kiểm tra code trùng dựa trên GIT API, Webservices, REST …

Hệ thống đang xây dựng một qui trình là tạo ra một công cụ để hỗ trợ chấm bài thi thực hành. Công cụ cho phép người dùng import đề thi thực hành và kịch bản để đánh giá nội dung bài thi được thực hiện được một giảng viên – tổ trưởng xây dựng.

Công cụ được xây dựng giả định việc canh thi practical test được giám sát bởi giảng viên và công cụ không đảm bảo việc quá trình thi diễn ra như thế nào. Công cụ cũng được xây dựng cho người làm bài, sinh viên để có thể kết nối với ứng dụng của giảng viên để đảm bảo rằng xác định đúng tài khoản được cho phép truy cập đến việc bắt đầu quá trình đưa bài vào hệ thống để đánh giá. Người làm bài sau khi kết nối sẽ thấy được nội dung làm bài, sau đó nộp bài theo định dạng format. Bài được đưa về hệ thống, giải nén và thực thi qui trình chấm bài xây dượng theo DevOps. Kết quả sau khi chấm xong sẽ thực hiện kiểm tra code trùng, code sao chép, … và ghi nhận kết quả để thông báo với giảng viên. Kết quả sẽ được giảng viên sẽ được công bố đến người làm bài. Ứng dụng thực hiện các chức năng như sau: Tạo ra một kỳ đánh giá với các thông tin về người tham gia thi, đề thi và kịch bản đánh giá tương ứng; Cho phép người tham gia kỳ thi đưa bài đánh giá vào hệ thống sau khi được xác thực; Đánh giá bài thi và ghi nhận kết quả; Cho phép tạo kịch bản để đánh giá bài thi; Kiểm tra code trùng và kiểm tra cấu trúc của nội dung làm bài; Kết nối giữa các ứng dụng giữa người tham gia thi, giữa giảng viên và dịch vụ đánh giá bài thi; Xây dựng bài test trên 4 môn C, OOP Java, Desktop Java và Web-based Java Application … Xây dựng qui trình dựa trên nhu cầu thực tế áp dụng công nghệ web services, giao tiếp giữa các thiết bị, cách thức đánh giá bài thi áp dụng DevOps, protocol TCP, UDP, và HTTP để giao tiếp giữa các thiệt bị. Ứng dụng sử dụng cơ chế để hệ thống có thể chạy tự động với các tác vụ để xử lý mà không cần sự tương tác của người sử dụng. Ứng dụng xây dựng trên nền tảng của ứng dụng, xây dựng các api tương tác giúp kết nối các hệ thống với nhau, kiểm tra semantic trong xử lý đánh giá, kiểm tra code trùng dựa trên GIT API, Webservices, REST …

Hệ thống đang xây dựng một qui trình là tạo ra một công cụ để hỗ trợ chấm bài thi thực hành. Công cụ cho phép người dùng import đề thi thực hành và kịch bản để đánh giá nội dung bài thi được thực hiện được một giảng viên – tổ trưởng xây dựng.

Công cụ được xây dựng giả định việc canh thi practical test được giám sát bởi giảng viên và công cụ không đảm bảo việc quá trình thi diễn ra như thế nào. Công cụ cũng được xây dựng cho người làm bài, sinh viên để có thể kết nối với ứng dụng của giảng viên để đảm bảo rằng xác định đúng tài khoản được cho phép truy cập đến việc bắt đầu quá trình đưa bài vào hệ thống để đánh giá. Người làm bài sau khi kết nối sẽ thấy được nội dung làm bài, sau đó nộp bài theo định dạng format. Bài được đưa về hệ thống, giải nén và thực thi qui trình chấm bài xây dượng theo DevOps. Kết quả sau khi chấm xong sẽ thực hiện kiểm tra code trùng, code sao chép, … và ghi nhận kết quả để thông báo với giảng viên. Kết quả sẽ được giảng viên sẽ được công bố đến người làm bài. Ứng dụng thực hiện các chức năng như sau: Tạo ra một kỳ đánh giá với các thông tin về người tham gia thi, đề thi và kịch bản đánh giá tương ứng; Cho phép người tham gia kỳ thi đưa bài đánh giá vào hệ thống sau khi được xác thực; Đánh giá bài thi và ghi nhận kết quả; Cho phép tạo kịch bản để đánh giá bài thi; Kiểm tra code trùng và kiểm tra cấu trúc của nội dung làm bài; Kết nối giữa các ứng dụng giữa người tham gia thi, giữa giảng viên và dịch vụ đánh giá bài thi; Xây dựng bài test trên 4 môn C, OOP Java, Desktop Java và Web-based Java Application … Xây dựng qui trình dựa trên nhu cầu thực tế áp dụng công nghệ web services, giao tiếp giữa các thiết bị, cách thức đánh giá bài thi áp dụng DevOps, protocol TCP, UDP, và HTTP để giao tiếp giữa các thiệt bị. Ứng dụng sử dụng cơ chế để hệ thống có thể chạy tự động với các tác vụ để xử lý mà không cần sự tương tác của người sử dụng. Ứng dụng xây dựng trên nền tảng của ứng dụng, xây dựng các api tương tác giúp kết nối các hệ thống với nhau, kiểm tra semantic trong xử lý đánh giá, kiểm tra code trùng dựa trên GIT API, Webservices, REST …

* *- Ứng dụng thực hiện được các chức năng như sau*
* *Thùng rác được thiết kế có khả năng phân loại rác, cung cấp tình trạng hoạt động – kết nối – ghi nhận giao dịch*
* *Thùng rác có khả năng xử lý và handle được các tình huống tùy theo vật phẩm người dùng đưa vào*
* *Thùng rác có ngăn phân loại rác thu thập và rác thải ra ngoài*
* *Thùng rác có tính trực quan với người sử dụng kể cả người không có account trong hệ thống*
* *Ứng dụng mobile của người sử dụng có account giúp người dùng ghi nhận quá trình thu thập rác, tích điểm, feedback, …*
* *Ứng dụng web của người vận hành có thể quản lý được trạng thái, vị trí hoạt động của thùng rác*
* *Ứng dụng web của người vận hành có thể hỗ trợ xử lý feedback và chuyển feedback phù hợp sang nhân viên kỹ thuật để cải thiện việc nhận dạng của hệ thống*
* *Ứng dụng web của người người kỹ thuật có thể hỗ trợ họ trong việc xác định các trường hợp để tăng cường thông tin cải thiện sự nhận dạng của hệ thống*
* *Ứng dụng web của người vận hành có thể hỗ trợ cung cấp thông tin về transaction, các số liệu tổng hợp, …*
* *…*
* *Xây dựng qui trình dựa trên nhu cầu thực tế thu thập trực tiếp từ chủ dự án, đề xuất ý tưởng thiết kế thùng rác, giải pháp cho phần mềm và triển khai thực tế để thu thập số liệu và cải tiến hệ thống …*
* *Ứng dụng sử dụng cơ chế để hệ thống có thể chạy tự động với các tác vụ để xử lý mà không cần sự tương tác của người vận hành*
* *Ứng dụng xây dựng trên nền tảng của ứng dụng Web dành cho người vận hành, nhân viên kỹ thuật … Ứng dụng mobile dành cho người sử dụng có account*