**BÀI THỰC HÀNH PHP**

TRONG 3-5 NGÀY TẠI NHÀ,

Ứng viên có thể xin thêm thời gian nếu muốn.

Ứng viên không bắtc buộc phải làm hết, nếu không

đủ thời gian.

**Bắt đầu:**

Dùng php framework Yii2 để làm bài kiểm

tra này.

Cơ sở dữ liệu có thể dùng: Mysql, mysqlite

Hệ thống gồm có các chức năng cần thiết sau.

**PHẦN 1. KỸ NĂNG CƠ BẢN**

Với các chức năng trong phần này, bạn không cần

bắt lỗi đầu vào.

1. Đăng ký tài khoản đăng nhập cho người sử dụng.

Người sử dụng tự đăng ký. Có 2 lựa chọn:

- nhập email và password

- hoặc đăng ký bằng tài khoản facebook hay google có

sẵn.

2. Đăng nhập. Có thể đăng nhập bằng email và

password đã đăng ký. Hoặc đăng nhập bằng tài khoản

facebook và google đã đăng ký ở chức năng 1.

3. Sau khi đăng nhập, người sử dụng sẽ được dẫn

vào trang thông tin cá nhân. Tại trang này, người sử

dụng có thể cập nhật các thông tin cá nhân sau:

- upload 1 ảnh đại diện

- tên họ

- tên gọi

- số điện thoại

- ngày tháng năm sinh (nên dùng datepickers nếu có thể)

4. Trên menu chính của hệ thống có các item sau:

- trang chủ: dẫn về trang thông tin cá nhân

- kinh nghiệm: cho phép người sử dụng vào trang để

nhập các việc mà mình đã từng làm.

Trang kinh nghiệm có các chức năng chi tiết sau: mỗi

việc cần có tháng và năm bắt đầu, tháng và năm

kết thúc, tên công ty, mô tả công việc. Người sử

dụng có thể nhập nhiều kinh nghiệm.

6. Làm giao diện cơ bản hợp mắt cho các chức năng

sử dụng css. Bạn cũng có thể dùng css có sẵn của

framework mà bạn lựa chọn. Không được pha trộn css

với html hay php. Bạn cần đặt css ra 1 file .css riêng.

Bạn cũng có thể dùng .less nếu muốn.

PHẦN 2. BẮT LỖI ĐẦU VÀO

7. Xử lý bắt lỗi đầu vào cho chức năng 1, và 4.

Tuỳ theo suy luận của bạn để xác định cần bắt

lỗi gì.

8. Với chức năng 5, thêm hạn chế kích cỡ file không

quá 5MB, và file chỉ được thuộc loại .doc, .docx,

.pdf, .png, .jpeg, và .jpeg.

PHẦN 3. DÙNG CÔNG CỤ QUẢN LÝ PHIÊN BẢN CODE (SVN,

GIT...)

9. Bạn tự tạo tài khoản của mình miễn phí trên

[github.com](http://github.com/) [1 [1]] và upload code của bạn lên đó.

PHẦN 4. PHÂN QUYỀN

10. Đảm bảo hệ thống cho phép nhiều người khác

nhau đăng ký tài khoản. Và tài khoản của người nào

chỉ được phép xem thông tin của người đó.

11. Commit phiên bản code mới nhất của bạn lên github

mà bạn tạo ở câu 9, nếu có thể.

PHẦN 5. PHÂN TÍCH GIẢI PHÁP HỆ THỐNG

12. Câu này không yêu cầu code. Bạn chỉ cần mô tả

giải pháp. Bạn có thể dùng đồ thị nếu muốn.

Tình huống như sau: sản phẩm cần có thêm khả năng

cho admin xem danh sách người đăng ký, xem thông tin cá

nhân của họ và các văn bản họ đã upload. Phần

admin này do 1 đội ngũ lập trình viên khác tham gia

làm, khác với đội ngũ làm các chức năng mô tả

trước câu hỏi này. Theo bạn nên thiết kế giải

pháp này ra sau. Vui lòng chọn a,b hoặc c kèm theo giải

thích lý do.

a. Cho 2 đội ngũ này chia sẻ cùng code, cùng cơ sở

dữ liệu và đội nào làm chức năng của đội đó.

Nếu có liên quan thì cùng nhau phân tích ảnh hưởng

và cùng tìm hướng giải quyết chung.

b. Tạo một hệ thống mới cho admin. Hệ thống mới

này chia sẻ thông tin với hệ thống hiện tại thông

qua Web Service API.

c. Tạo một hệ thống mới cho admin. Hệ thống mới

này lấy thông tin của hệ thống hiện tại thông qua

một phương pháp tối ưu hơn cả Web Service API (bạn

vui lòng nói rõ phương pháp nào trong giải thích của

bạn)

NỘP BÀI

Nếu bạn không làm được câu 9 thì khi xong bài, bạn

vui lòng zip code lại và gửi qua email cho chúng tôi.

Nếu làm được câu 9, bạn chỉ cần email cho chúng

tôi địa chỉ github của bạn.

Cảm ơn bạn.