

1

Tổng Hợp Những Câu Hỏi Phỏng Vấn Tester Thường Gặp

**Biên soạn : Trung Tâm TesterProVN Địa chỉ: Tầng 2 - nhà văn hóa dân**

**phố tổ 17 - Ngách 2 – Ngõ 76 - Duy Tân- Cầu Giấy- Hà Nội**

**Website:** <https://testerprovn.com/>

**Hotline:0973.303.357**



2

Mục Lục

[Phần I - Chuyên môn Test 6](#_bookmark0)

[**Câu hỏi 1: Quy trình kiểm thử có những bước nào?** 6](#_bookmark1)

[**Câu hỏi 2: Test case hay kịch bản test là gì ?** 6](#_bookmark2)

[**Câu hỏi 3: Theo bạn thì kiểm thử là gì?** 6](#_bookmark3)

[**Câu hỏi 4: Theo em lỗi thường xuất hiện ở giai đoạn nào là chủ yếu trong chu kỳ phát triển phần mềm?** 7](#_bookmark4)

[**Câu hỏi 5 : Lợi ích chính của kiểm thử sớm trong chu kỳ phát triển phần mềm là gì?** 7](#_bookmark5)

[**Câu hỏi 6 . Bạn biết gì về template của test case? (Hoặc biểu mẫu testcases có những nội dung nào?)** 7](#_bookmark6)

[**Câu hỏi 7: Test plan là gì? Test plan gồm những nội dung gì?** 8](#_bookmark7)

[**Câu hỏi 8 : Ở trung tâm( hoặc ở cty em đang làm ) Em thực thi test như thế nào?** 8](#_bookmark8)

[**Câu hỏi 9: Bạn biết gì về các kỹ thuật test hộp đen ( hay bạn biết những gì về các kỹ thuật thiết kế testcase? )** 8](#_bookmark9)

[**Câu hỏi 10: Mục đích sử dụng các kỹ thuật test như phân vùng tương đương hay phân tích giá trị biên để làm gì?**](#_bookmark10)

[........................................................................................................................................................................................ 9](#_bookmark10)

[**Câu hỏi 11: Có mấy mức/ giai đoạn test mà bạn biết ?** 9](#_bookmark11)

[**Câu hỏi 12: Integration testing là gì? System testing là gì?** 10](#_bookmark12)

[**Câu hỏi 13: Apha testing là gì? Beta testing là gì?** 12](#_bookmark13)

[**Câu hỏi 14: Vì sao lỗi càng phát hiện muộn thì chi phí sửa lỗi càng cao?** 12](#_bookmark14)

[**Câu hỏi 15: Testcase là gì ( Hoặc kịch bản kiểm thử là gì – Hoặc e biết gì về test case ?** 12](#_bookmark15)

[**Câu hỏi 16: Bạn biết gì về test?( Hoặc ở trung tâm bạn được học những gì? )** 13](#_bookmark16)

[**Câu hỏi 17: Re-test là gì? Regression Testing (test hồi quy) là gì?** 15](#_bookmark17)

[**Câu hỏi 18: Smoke test là gì?** 15](#_bookmark18)

[**Câu hỏi 19: Performance testing là gì/ Test hiệu năng là gì?** 16](#_bookmark19)

[**Câu hỏi 20: Test Web App/Mobile khác Test Web ở desktop/ PC như thế nào?** 17](#_bookmark20)

[**Câu hỏi 21: Test report là gì – Báo cáo kiểm thử dùng để làm gì ? Nội dung báo cáo kiểm thử gồm những gì?** 18](#_bookmark21)

[**Câu hỏi 22: Em biết những quy trình phát triển phần mềm nào ?** 19](#_bookmark22)

[**Câu hỏi 23: Mô hình Agile/Scrum là gì? Mô hình này khác với các quy trình khác ở điểm nào ?** 19](#_bookmark23)

[**Câu hỏi 24: Viết thử một bug bằng tiêng anh hoặc tiếng việt** 20](#_bookmark24)

[**Câu hỏi 25: Theo bạn để trở thành tester thì cần những gì ? ( Hoặc test thì cần gì ? )** 22](#_bookmark25)

[**Câu hỏi 26: Tester làm thế nào để xác đinh được mức độ của regression test là phải test hồi quy tất cả hay chỉ**](#_bookmark26)[**test 1 phần ?** 24](#_bookmark26)

3

[Câu hỏi 27: Test hồi quy sẽ được thực hiện như thế nào ? 24](#_bookmark27)

[Câu hỏi 28: Test API là gì ? ( Em biết gì về test API ) 24](#_bookmark28)

[Câu hỏi 29: Tại sao em lại chọn làm nghề Tester ? 24](#_bookmark29)

[Câu hỏi 30: Em đã tham gia dự án nào chưa ? có thể các bạn sẽ bị hỏi chi tiết làm gì trong dự án đó và test như](#_bookmark30)

[thế nào nếu các bạn có ghi trong CV 25](#_bookmark30)

[Câu hỏi 31: Em hãy Liệt kê testcases của chức năng Login? 26](#_bookmark31)

[Câu hỏi 32: Hoặc hỏi test boundary value ( giá trị biên cho khoảng từ 0->20? (Hoặc Em hãy liệt kê testcases cho](#_bookmark32) [dải số từ 0 đến 20?) 26](#_bookmark32)

[Câu hỏi 33: Em hãy Liệt kê testcases chức năng QUÊN Mật khẩu 28](#_bookmark33)

[Câu hỏi 33: Độ nghiêm trọng (Priority) và mức độ ưu tiên (Severity) của bug là gì đưa ra dẫn chứng cho từng loại](#_bookmark34)

[? Nêu ví dụ mà một bug có độ nghiêm trọng cao và độ ưu tiên thấp 29](#_bookmark34)

[Câu hỏi 34: Regression testing khác gì re-test ? 30](#_bookmark35)

[Câu hỏi 35: Một báo cáo công việc kiểm thử (test report) gồm những gì? Và ích lợi của bảng báo cáo này? 31](#_bookmark36)

[Câu hỏi 36: Khi nào thì nên sử dụng kiểm thử tự động? 31](#_bookmark37)

[PHẦN II: SQL 32](#_bookmark38)

[Câu hỏi 1: SQL là gì? 32](#_bookmark39)

[Câu hỏi 2: Một bảng có được 2 khóa chính không? 32](#_bookmark40)

[Câu hỏi 3: Câu lện nào dùng để truy vấn hay cần tìm 1 dữ liệu trong DB? 32](#_bookmark41)

[Câu hỏi 4: Khi cần Thêm mới 1 dữ liệu trong DB thì dùng câu lệnh gì ? 33](#_bookmark42)

[Câu hỏi 5: Khi cần sửa 1 dữ liệu trong DB thì dùng câu lệnh gì? 33](#_bookmark43)

[Câu hỏi 6: Nếu cần xóa 1 dữ liệu trong DB thì dùng câu lênh gì? 33](#_bookmark44)

[Câu hỏi 7: Nếu cần đếm số lượng trong DB thì dùng câu lệnh gì? 34](#_bookmark45)

[Câu hỏi 8: Nếu muốn join 2 bảng trong DB thì e dùng câu lệnh nào? 34](#_bookmark46)

[Câu hỏi 9: Câu lệnh Order by dùng để làm gì ? 35](#_bookmark47)

[Câu hỏi 10 : Làm thế nào để lấy được tất cả các bản ghi trong một bảng xác định? 35](#_bookmark48)

[Câu hỏi 11: Bạn thêm một cột vào bảng như thế nào? 36](#_bookmark49)

[Câu hỏi 12 : Định nghĩa câu lệnh SQL Delete 36](#_bookmark50)

[Câu hỏi 13: Primary key là gì? 37](#_bookmark51)

[Câu hỏi 19: Foreign keys là gì? 37](#_bookmark52)

[Câu hỏi 20 : Một bảng có thể có nhiều hơn một khóa ngoại được không? 38](#_bookmark53)

[Câu hỏi 21: Trường BOOLEAN có thể nhận các giá trị nào 38](#_bookmark54)

[Phần III - Câu hỏi tình huống thường gặp 38](#_bookmark55)

4

[Câu hỏi 1: Khi Dev/ Lập trình viên không công nhận bug/ lỗi của em, em sẽ giải quyết thế nào ? 38](#_bookmark56)

[Câu hỏi 2: Nếu phải test số lượng lớn testcase trong thời gian ngắn ( tức là ko đủ thời gian để test hết) thì phải](#_bookmark57) [làm thế nào? 38](#_bookmark57)

[Câu hỏi 3: Em phải test một phần mềm mới nhưng người quản lí gia hạn thời gian để test ít. Với trình độ của bạn](#_bookmark58) [cần nhiều thời gian hơn để test.Em sẽ phải làm gì? 39](#_bookmark58)

[Câu hỏi 4: Là tester mới, khi phát hiện ra bug nhưng em không chắc chắn thì em sẽ làm như thế nào ? 40](#_bookmark59)

[Câu hỏi 5: Nếu em vào làm tester của một dự án không có tài liệu nghiệp vụ thì em sẽ làm như thế nào ? 40](#_bookmark60)

[Câu hỏi 6: Nếu dự án / màn hình em được giao vào làm tester nhưng không có đủ tài liệu hoặc tài liệu nghiệp vụ](#_bookmark61) [không chi tiết thì em sẽ làm thế nào ? 40](#_bookmark61)

[Câu hỏi 7: Em đã từng bỏ qua bug nghiêm trọng nào chưa ? Nếu đã từng bạn sẽ xử lý như thế nào ? 41](#_bookmark62)

[Câu hỏi 8: Nếu em nhận được bản build có nhiều lỗi nghiêm trọng em sẽ làm như thế nào ? 41](#_bookmark63)

[Câu hỏi 9: Nếu Tester tìm thấy lỗi trên máy test nhưng do dev không nhận lỗi này để fix nguyên nhân máy dev](#_bookmark64) [không bị lỗi này thì em sẽ làm như thế nào ? 41](#_bookmark64)

[Câu hỏi 10: Khi nào thì chúng ta dừng test ? Hoặc khi nào thì việc kiểm thử phần mềm kết thúc? 42](#_bookmark65)

[Câu hỏi 11: Kiểm thử nên bắt đầu ở giai đoạn nào trong chu kỳ phát triển phần mềm? 42](#_bookmark66)

[Câu hỏi 12: Khi được hỏi em muốn mức lương bao nhiêu? 42](#_bookmark67)

[Câu hỏi 13: Định hướng của em như nào trong khoảng 2-3 năm tới ? 43](#_bookmark68)

[Câu hỏi 14: Bạn tự nhận xét mình có những ưu điểm/khuyết điểmgì? 43](#_bookmark69)

[Câu hỏi 15: Nếu làm chung dự án với developer rất nóng nảy, bạn làm thếnào để dung hòa công việc với họ? 43](#_bookmark70)

[Câu hỏi 14: Bạn tự nhận xét mình có những ưu điểm/khuyết điểmgì? 44](#_bookmark71)

[Câu hỏi 15: Nếu làm chung dự án với developer rất nóng nảy, bạn làm thếnào để dung hòa công việc với họ? 44](#_bookmark72)

[Câu hỏi 16: Tại sao bạn lại chọn công ty này mà không phải công ty khác? 44](#_bookmark73)

[Câu hỏi 17: Em có câu hỏi gì cho anh/chị không ? ( Nhà tuyển dụng hỏi ứng viên ) 44](#_bookmark74)

[Câu hỏi 18 : Nghành Tester có điều gì thích thú? Mức lương hiện nay của Tester khoảng bao nhiêu? 45](#_bookmark75)

[Câu hỏi 19 : Bước chân vào nghề KTPM cần những kỹ năng 45](#_bookmark76)

[Để làm việc tốt và gắn bó lâu dài với nghề, chúng ta cần chuẩn bị những gì và hiện thức nó như thế nào? 45](#_bookmark77)

[Câu hỏi 20 : Là một Tester cần biết ngôn ngữ Lập trình cơ bản nào? Sự hấp dẫn của công việc Kiểm thử phần](#_bookmark78) [mềm (KTPM) như thế nào? Đã làm các anh trung thành với công việc KTPM? 46](#_bookmark78)

[Câu hỏi 21 : Vì sao phải kiểm tra bằng tay khi mình đã có kiểm thử tự động (KTTĐ) ? 46](#_bookmark79)

[Câu hỏi 22 : Kiểm thử tự động có những khó khăn nào? 46](#_bookmark80)

[Có người nói Tester không cần phải biết code, có người nói code rất giỏi mới test được, như vậy ý kiến nào](#_bookmark81)

[đúng? 46](#_bookmark81)

5

[Câu hỏi 23 : Theo em biết, ở VN những Tester đa phần là lập trình không tốt nên mới theo ngành Kiểm thử,](#_bookmark82)

[nhưng em thấy trong yêu cầu Tester phải viết được các tool automation thì làm sao có thể viết được? 47](#_bookmark82)

[Câu hỏi 24: Khi test 1 phần mềm, làm sao để biết rằng test thế nào là đủ? 47](#_bookmark83)

[Câu hỏi 25 : Yêu cầu cần và đủ của 1 Tester là gì? 48](#_bookmark84)

[Câu hỏi 26 : Developer và Tester không thể tồn tại nếu thiếu nhau, nhưng khi làm việc thì thường xuyên xảy ra](#_bookmark85) [xung đột. Anh/chị có thể cho em biết có quy trình hoặc phương pháp nào giúp Developer và Tester kết hợp làm](#_bookmark85) [việc hoàn hảo và tránh xung đột trong công việc? 48](#_bookmark85)

[Nhớ đọc kỹ câu 29-phần 1 nhé](#_bookmark86) [48](#_bookmark86)

# Phần I – Giới thiệu về bản thân + Kết thúc PV

Câu hỏi 1: Bạn hãy giới thiệu về bản thân mình?

Lời đầu tiên em xin chân thành cảm ơn Quý công ty đã tạo cơ hội cho em được phỏng vấn ngày hôm nay.

Em tên là Nguyễn Đức Tiến, 24 tuổi, và hiện tại em có 2 năm kinh nghiệm trong lĩnh vực kiểm thử phần mềm.

Trong thời gian làm việc tại công ty ITSOL, em đã tham gia OS tại Tập đoàn Viettel, nơi em lần đầu tiếp xúc với mô hình phát triển phần mềm Waterfall. Tại đây, em đã đảm nhận vai trò chính trong việc kiểm thử, đi demo sản phẩm với khách hàng và viết hướng dẫn sử dụng cho hệ thống.

Sau đó, em có cơ hội OS tại Ngân hàng MSB, nơi em hiểu rõ hơn về mô hình Agile và tiếp tục đảm nhiệm vai trò kiểm thử.

Gần đây, em làm việc tại Ngân hàng SHB, em cũng rất vinh dự vì em là cá nhân đầu tiên của đội tester và được join vào TTPT . Tại đây em học hỏi được rất nhiều kinh nghiệm từ manual test và automation test. Đưa kinh nghiệm đó vào các dự án.

Câu hỏi 2: Bạn hãy giới thiệu dự án?

**Mục tiêu của dự án:**

Xây dựng 1 hệ thống cho phép người dùng cấu hình và quản lý hạn mức của từng quốc gia

**Có các roll:**

1. Maker khởi tạo hạn mức gửi đến checker.
2. Tại checker sẽ kiểm tra các thông tin từ hồ sơ hạn mức:
   1. Checker phê duyệt end luồng
   2. Checker từ chối > hồ sơ sẽ trở về maker bổ sung

**Vai trò của tôi trong dự án:**

Là tester chính trong dự án và cùng BA đảm bảo chất lượng của sản phẩm trước khi ra mắt.

**Quá trình thực hiện:**

* Đọc tài liệu đặc tả + tài liều hạn mức của từng quốc gia trong năm 2020

Em có khái niệm : Ngoài kiểm thử phần mềm hệ thống ra ta còn kiểm thử cả tài liệu đặc tả

VD: Tài liệu trên app mobile:

Tài liệu đề cập đến câu messeger thông báo “Số tài khoản của quý khách chọn không phù hợp. Vui lòng nhập số tài khoản khác” và button Xác nhận >< nhưng trong hình ảnh minh họa ở dưới thì button lại là OK

* **Lập kế hoạch kiểm thử:** Tạo ra kế hoạch kiểm thử chi tiết cho các chức năng của hệ thống
* **Thực hiện kiểm thử:** Em đã kiểm thử chức năng, kiểm thử hiệu năng bằng 2 phương pháp: Load testing và Stress testing (Kiểm thử ngoài lề vì không yêu cầu), Kiểm thử tính tương thích, kiểm thử UX, UI và kiểm thử hồi quy.

Ghi nhận lỗi: Trong quá trình kiểm thử em đã phát hiện ra 18 lỗi bao gồm 1 lỗi nghiêm trọng

**Thách thức:**

* Thiếu công cụ hỗ trợ: Phần này em đã đề xuất Jira để quản lý dự án lẫn quản lý bugs. Mất thời gian nhưng phía cty yêu cầu đội tester phải lớn mạnh thì bên công ty sẽ hỗ trợ và phát triển ý kiến của em
* Chưa có phần mềm quản lý bug: Em đã đề xuất log và quản lý bug trên gg sheet để mọi người theo dõi và quản lý tránh tình trạng lọt bug or log trùng bug
* Thiếu sự hỗ trợ từ các bộ phận khác: Các nhóm phát triển trong cty có thể chưa nhận được tầm quan trong của kiểm thử trong quy trình nên dẫn đến thiếu sự hợp tác và các thông tin cần thiết. Vì họ đã quen quy trình cũ là code xong BA sẽ nhảy vào test.
* Khó khăn trong việc xây dựng niềm tin với các đội nhóm: Như em xin thông tin DB để query kiểm tra nhưng cũng mất nhiều thời gian để có
* Cảm giác của em cũng khá là cô đơn: Vì 1 môi trường chưa có quy trình test và các ý kiến của em đều bị bãi bỏ và lúc đấy cũng chưa có leader để hỗ trợ.

## Nêu rõ ra :

## Mục tiêu dự án

## Các Roll có trong hệ thống

## Vai trò của mình trong dự án

## Quá trình thực hiện

## Thách thức

## Câu kết:

## Cám ơn anh chị đã giành thời gian để phỏng vấn em

## Rất mong trong thời gian sớm em sẽ được hợp tác với quý công ty và được làm việc cùng với các anh chị



6

# Phần II - Chuyên môn Test

Câu hỏi 1: Quy trình kiểm thử có những bước nào?

## Có 5 giai đoạn

* Lập kế hoạch test ( test plan) và kiểm soát việc kiểm thử
* Phân tích yêu cầu vàThực hiện viết testcase
* Thực hiện test ( test execution)
* Đánh giá tiêu chí dừng test + Lập báo cáo kiểm thử (test report)
* Đóng hoạt động test ( Close Test)

Câu hỏi 2: Test case hay kịch bản test là gì ?

* Test case là tài tiệu dùng để mô tả một dữ liệu đầu vào, hành động và một kết quả mong đợi để xác định một chức năng của ứng dụng phần mềm hoạt động đúng hay không.
* Testcases(kịch bản test) thông thuờng đuợc Tester viết trên excel. Một file Testcase nhất định cần phải có những truờng sau: ID , mục tiêu của việc test(summary), các buớc thực hiện test (Steps), và kết quả trả về (expected result)
* Ngoài ra các bạn có thể nói thêm ý này : Để viết đuợc testcases có hiệu quả bao phủ hết các truờng hợp cần test thì testcases phải có đầy đủ hết các Nghiệp vụ mà hệ thống yêu cầu ,biết vận dụng linh hoạt các kỹ thuật thiết kế testcase (các kỹ thuật test hộp đen) để viết đuợc testcases có độ bao phủ tối đa nhất.

Câu hỏi 3: Theo bạn thì kiểm thử là gì?

- Kiểm thử phần mềm (software testing) là hoạt động nhằm tìm kiếm, phát hiện các lỗi của phần mềm

* Kiểm thử phần mềm đảm bảo sản phẩm phần mềm đáp ứng chính xác, đầy đủ và đúng theo yêu cầu của khách hàng, yêu cầu của sản phẩm đề đã đặt ra.
* Kiểm thử phần mềm tạo điều kiện cho bạn tận dụng tối đa tư duy đánh giá và sáng tạo để bạn có thể phát hiện ra những điểm mà người khác chưa nhìn thấy.

7

Câu hỏi 4: Theo em lỗi thường xuất hiện ở giai đoạn nào là chủ yếu trong chu kỳ phát triển phần mềm?

Ở giai đoạn sau khi code xong và bàn giao sang cho tester bắt đầu giai đoạn testing. Một bên test và 1 bên fix bug, đây là giai đoạn nhiều lỗi nhất trong chu kỳ phát triển phần mềm.

*( Nếu bị hỏi tại sao thì gthich thêm, nếu không hỏi thì thôi :Nguyên nhân do dev chưa đánh giá hết được mức độ ảnh hưởng nên sửa chỗ này có thể làm hỏng chỗ khác,sửa bug này lại phát sinh thêm lỗi khác hoặc lúc build code lên làm mất code của mình hoặc code của người khác…)*

Câu hỏi 5 : Lợi ích chính của kiểm thử sớm trong chu kỳ phát triển phần mềm là gì?

* Hoạt động kiểm thử nên bắt đầu càng sớm càng tốt trong chu trình phát triển mềm và cần được tập trung vào các mục tiêu xác định.
* Kiểm thử sớm, phát hiện lỗi sớm, lỗi được sửa sớm sẽ đảm bảo được dự án hoàn thành đúng tiến độ và chất lượng sản phẩm cũng được đảm bảo. Chi phí của dự án cũng sẽ không bị phát sinh.
* Lỗi phát hiện muộn, sửa muộn, nhất là dồn vào giai đoạn thời gian cuối dự án, sẽ dẫn đến sửa vội, test vội, code vội, chất lượng ko được đảm bảo, tiến độ không hoàn thành được dẫn đến phải overtime, tăng chi phí của dự án.

Câu hỏi 6 . Bạn biết gì về template của test case? (Hoặc biểu mẫu testcases có những nội dung nào?)

Theo em được biết thì một mấu template testcase thường gồm 5 phần chính như sau:

+ ID của Test Case

+ Mục đích kiểm thử ( Summary)

+ Các buớc thực hiện ( Steps to reproduce )

+ Kết quả mong muốn ( phần mềm phải chạy đúng nhu kết quả mong muốn) ( Expected resutls )

+ Kết quả thực tế ( cái này là khi nào thực hiện test thì điền là Pass/ Fail) ( Test results )



8

Câu hỏi 7: Test plan là gì? Test plan gồm những nội dung gì?

Test plan là tài liệu tổng quan về việc kiểm thử 1 project bao gồm những thông tin sau:

* Phạm vi kiểm thử, huớng tiếp cận, quy trình kiểm thử, tài nguyên và nhân lực test cần có, các chức năng/ module cần đuợc test, các công cụ và môi truờng test cần có.
* Bao gồm cả kế hoạch ai test chức năng nào, khi nào bắt đầu thực hiện viết và hoàn thành testcases, khi nào bắt đầu thực hiện test và kế hoạch hoàn thành test
* Dựa vào kế hoạch chung của dự án để lên kế hoạch cho bên kiểm thử. Trong truờng hợp khi làm thực tế thấy có khả năng không đúng nhu kế hoạch đã lên thì phải báo cáo cho cấp trên (PM, QC manager /Leader )

Câu hỏi 8 : Ở trung tâm( hoặc ở cty em đang làm ) Em thực thi test như thế nào?

## ( Hoặc execute test là gì ? Hoặc em thực hiện test như thế nào ? )

* Là tìm hiểu tài liệu đặc tả yêu cầu để hiểu hệ thống và thực hiện viết testcases. Khi lập trình code xong và đã đẩy code lên môi trường test thì em sẽ tiến hành thực hiện test dựa trên testcase đã viết.
* Trong quá trình test, tìm thấy bug/ lỗi thì log/ viết lỗi lên tool quản lý lỗi. Lỗi của Lập trình viên nào thì giao cho người coder đó. Khi nào lập trình viên fix bug xong thì tester sẽ thực hiện test lại. Nếu OK rồi thì close bug, nếu chưa thì lại re-open/mở lại lỗi để Lập trình viên thực hiện fix lại.
* Khi thực hiện test thì test luồng chính truớc, test chức năng chính truớc, luồng phụ test, giao diện test sau luồng chính.
* Ngoài ra, cũng phải đôn đốc, giục lập trình viên fix bug để kịp tiến độ nếu thấy chậm và luời fix bug
* Cuối cùng thực hiện làm báo cáo kiểm thử (Tùy theo y/c của dự án )

Câu hỏi 9: Bạn biết gì về các kỹ thuật test hộp đen ( hay bạn biết những gì về các kỹ thuật thiết kế testcase? )

Có 1 số kỹ thuật như:



9

* Phân vùng tuơng đuơng (Equivalence Patitioning )
* Phân tích giá trị biên (boundary value analysis)
* Bảng quyết định (Decision table )
* Đoán lỗi (error guessing)

#### Nếu bị hỏi chi tiết hơn về các kỹ thuật thì nói như sau :

* **Phân vùng tương đương** : chia miền đầu vào của một chương trình thành các lớp dữ liệu, từ đó suy dẫn ra các ca kiểm thử. Thiết kế ca kiểm thử cho phân vùng tương đương dựa trên sự đánh giá về các vùng tương đương với một điều kiện vào. Vùng tương đương biểu thị một tập cho các trạng thái hợp lệ hay không hợp lệ đối với điều kiện vào
* **Phân tích giá trị biên** : Tập trung vào biên của miền xác định để xây dựng test case .

Lý do là lỗi thường xảy ra ở gần các giá trị biên

* **Bảng quyết định** : là kỹ thuật quyết định số testcase tối thiểu với độ bao phủ tối đa và được áp dụng cho những trường hợp cần nhiều sự kết hợp.
* **Đoán lỗi** : Đoán lỗi là kỹ thuật dựa vào kinh nghiệm ,phán đoán của tester chứ nó không có luật rõ ràng như các kỹ thuật còn lại

Câu hỏi 10: Mục đích sử dụng các kỹ thuật test như phân vùng tương đương hay phân tích giá trị biên để làm gì?

* Giúp giảm thiểu số luợng testcase thừa, tiết kiệm thời gian viết test case và thực hiện test mà vẫn đảm bảo chất luợng phần mềm
* Giúp Tester lựa chọn đuợc đúng testcase đại diện cần test mà vẫn bảo phủ đuợc các truờng hợp

Câu hỏi 11: Có mấy mức/ giai đoạn test mà bạn biết ?

Có 4 mức

* + Unit test: Do lập trình viên/test thực hiện test
  + Integration test : Do tester thực hiện test
  + System test : Do tester thực hiện test
  + Acceptance test: khách hàng test



10

Câu hỏi 12: Integration testing là gì? System testing là gì?

1. **Integration testing:**

Là công việc kiểm thử tích hợp 1 nhóm các module riêng lẻ với nhau. Dự án phần mềm gồm nhiều module phần mềm, đuợc code bởi nhiều nguời khác nhau. Tích hợp thử nghiệm tập trung vào kiểm tra truyền dữ liệu giữa các module.

Kiểm thử tích hợp tập trung chủ yếu vào các giao diện & lưu luợng dữ liệu / thông tin giữa các module.

Nếu bị hỏi phải lấy ví dụ thì mới nói tiếp vidu sau nhé:

VD: Có 3 module là Login, Register , Phân Quyền

+ Admin tạo 1 tài khoản thành công, tài khoản có trạng thái là Active và gán quyền cho người dùng này là Member chẳng hạn

+ Tiếp theo em sẽ dùng tài khoản vừa tạo đó để đăng nhập thành công và thực hiện các chức năng theo đúng các quyền được gán để xem nó có thực hiện được không

+ Tiếp đó e lại login vào bằng Admin để chuyển tài khoản Member em vừa tạo thành trạng thái Inactive

+ Bước cuối cùng là em dùng tài khoản Member đó để đăng nhập vào hệ thông và kết quả mong muốn của em là đăng nhập không thành công.

### System testing

Kiểm thử hệ thống là kiểm thử toàn bộ chức năng và giao diện của hệ thống, đảm bảo hệ thống không có lỗi và đáp ứng theo đúng yêu cầu nghiệp vụ.

Kiểm thử hệ thống là kiểm thử hộp đen. System liên quan đến các hoạt động bên ngoài của phần mềm từ quan điểm của nguời sử dụng.

11

Các kỹ thuật kiểm thử phổ biến được sử dụng trong kiểm thử hệ thống:

* **Kiểm thử chức năng**



12

* **Kiểm thử giao diện**
* **Kiểm thử bảo mật**
* **Kiểm thử hiệu năng …**

Câu hỏi 13: Apha testing là gì? Beta testing là gì?

**Alpha testing:** là một dạng của acceptance testing. Alpha testing là 1 nhóm nguời thực hiện test tại nơi sản xuất phần mềm. Alpha testing là một hình thức kiểm thử chấp nhận nội bộ, truớc khi phần mềm đuợc tiến hành kiểm thử Beta.

**Beta test:** là một hình thức acceptance testing bởi nguời dùng ngoài. Beta test đuợc thực hiện tại địa điểm của khách hàng/ người dùng thực hiện test hay sử dụng phần mềm bên ngoài - không phải tại nơi phát triển phầm mềm.

Câu hỏi 14: Vì sao lỗi càng phát hiện muộn thì chi phí sửa lỗi càng cao?

* Kiểm thử và sửa lỗi có thể được thực hiện tại bất kỳ giai đoạn nào của vòng đời phát triển sản phẩm. Từ giai đoạn Phân tích đặc tả nghiệp vụ, Thiết kế, Coding chứ không chỉ riêng giai đoạn test hay tập trung cho tất cả giai đoạn test.
* Lỗi được phát hiện càng muộn thì chi phí cho việc sửa lỗi càng lớn, bởi vì có những lỗi sẽ phải thực hiện lại từ khâu Thiết kế, rồi coding lại và mới thực hiện test được. Nên lỗi được phát hiện càng sớm, càng ở những giai đoạn đầu dự án, thậm chí ngay từ giai đoạn làm Yêu cầu/ Nghiệp vụ giúp cho các giai đoạn sau thực hiện được chính xác, giảm được số lượng lỗi và sản phẩm hoàn thành đúng tiến độ theo kế hoạch.
* Bug phát sinh ở giai đoạn release là nghiêm trọng và tốn kém nhất. Không chỉ bị ảnh hưởng về mặt uy tín chất lượng sản phẩm, mà còn dẫn đến việc phải coding và testing lại, phát sinh chi phí về nhân lực dự án, chậm trễ tiến độ.
* Bug được phát hiện càng muộn thì chi phí sửa càng lớn. Đôi khi không chỉ tỷ lệ với thời gian mà có thể là tỷ lệ bình phương thời gian.

Câu hỏi 15: Testcase là gì ( Hoặc kịch bản kiểm thử là gì – Hoặc e biết gì về test case ?

13

* Testcase là tài liệu mô tả dữ liệu đầu vào, hành động và kế quả mong đợi ( expected result) để xác đinh một chức năng của ứng dụng/ phần mềm hoạt động có đúng hay không
* Testcase thông thường được viết trên excel . Một file testcase nhất định phải có những trường sau :

+ Testcase ID

+ Mục tiêu của việc test ( Summary )

+ Các bước thực hiện test ( Steps)

+ Kết quả mong đợi ( Expected results )

+ Kết quả sau khi thực hiện test ( Test Results)

*( Ngoài ra nếu bạn nào tự tin thì có thể nói thêm cả ý này ) Để viết được test case có hiệu quả bao phù hết các trường hợp cần test thì test case phải có đầy đủ hết các nghiệp vụ mà hệ thông yêu cầu ( các yêu cầu này được viết trong tài liệu đặc tả hệ thông , vì vậy tester không được bỏ sót bất ký nghiệp vụ nào ) .*

*Đồng thời sử dụng các kỹ thuật thiết kế testcase ( Kỹ thuật test hộp đen ) để viết testcase có độ bao phủ tối đa.*

Câu hỏi 16: Bạn biết gì về test?( Hoặc ở trung tâm bạn được học những gì? )

* Em biết quy trình kiểm thử là … ( Lý thuyết đã viết ở những câu trên)
* Em biết các kỹ thuật để viết testcase ( Phân vùng tuơng đuơng, Phân tích giá trị biên, Bảng quyết định…)



14

* Biết phân tích yêu cầu
* Biết thực hiện test , log bug bằng Jira, biết dùng những câu lệnh SQL mà tester chuyên dùng.



15

Câu hỏi 17: Re-test là gì? Regression Testing (test hồi quy) là gì?

### Re-test:

Re-test là thực hiện test để đóng bug/ defect / lỗi sau khi lập trình viên đã fixed bug hoặc sửa 1 chức năng nào đó rồi test lại chức năng sửa đó thì gọi là test lại hoặc 1 chức năng cần re test vài lần cho hết bug

#### Regression testing :

Test hồi quy là test lại 1 chức năng đã đuợc làm xong, đã đuợc test xong rồi, đã hết lỗi nhung do có sự sửa đổi 1 chức năng khác mà lại có ảnh huởng đến nó, thì phải test 1 chức năng đã xong rồi thì gọi là test hồi quy

* Ví dụ có 3 chức năng A B C đã hoàn thành, chức năng A và C có mối quan hệ với nhau, và chức năng A cần phải sửa đổi thêm về nghiệp vụ, và việc sửa chức năng A sẽ làm ảnh huởng đến C và việc phải test lại chức năng C thì gọi là test hồi quy hoặc khi Dev sửa 1 chức năng mà chức năng này có làm ảnh huởng đến chức năng đã xong rồi thì cũng phải thực hiện test lại chức năng đã xong rồi kia thì gọi là test hồi quy
* Hoặc ngay cả khi re test để đóng bug, mà thấy chức năng Developer sửa có thể làm ảnh huởng đến 1 chức năng khác đã xong rồi thì tester cũng nên test hồi quy lại chức năng đó để tránh có lỗi tiềm ẩn

Câu hỏi 18: Smoke test là gì?

( Lưu ý chỉ nên chọn 1 trong các ý dưới để nói thôi )

16

* Smoke Testing là 1 quá trình để kiểm tra liệu bản build có ổn định hay không? Để xem bản build có đủ ổn định để thực hiện test chi tiết hay ko. (trong truờng hợp bản build ko ổn định, lỗi luôn chức năng chính hoặc build bị lỗi thì trả lại Dev, yêu cầu sửa luôn)
  + Hay kiểm tra các tính năng quan trọng có đang hoạt động hay không
  + Nó là 1 bài test hồi quy nhỏ và nhanh của các chức năng chính.
  + Nó là 1 bài test đơn giản cho thấy sản phẩm đã sẵn sàng cho việc test chưa.

Vidu: Em được giao làm smoke test cho chức năng đăng ký thì em sẽ đi kiểm tra luồng chính của nó là nhập vào tất cả các trường ở màn hình đăng ký giá trị hợp lệ rồi nhấn nút [Đăng Ký] để xem nó có cho đăng ký thành công hay không. Nếu đăng ký thành công => quá trình smoke test cho chức năng đăng ký của em đã OK, còn nếu đăng ký không thành công thì em sẽ trả lại cho dev bản build này và yêu cầu dev sửa luôn.

Câu hỏi 19: Performance testing là gì/ Test hiệu năng là gì?

* + Performance testing là đo tải khả năng của 1 hệ thống và cách cu xử trong điều kiện hoạt động bình thuờng. Nó tìm ra điểm nguỡng tối đa chịu tải của hệ thống
  + Test hiệu năng cũng đo cả hệ thống chiếm dụng tài nguyên nhu: RAM, CPU hay Bộ nhớ. Nếu vuợt nguỡng (nhu chiếm 80% RAM) thì là hiệu năng cũng hệ thống ko đạt
  + Để test hiệu năng thì dung tool để test nhu LoadRunner hay Jmeter

Ví dụ:

+ Load test: Kiểm tra hiệu năng của 1 trang Login với số luợng user đăng nhập tăng dần là 100, 200, 300, 400, 1000… và xem response time trả về là bao nhiêu ms, tức là 1000 user đăng nhập cùng 1 lúc thì thời gian hoàn thành sẽ làm



17

bao nhiêu.

+ Stress test: Ngoài ra cũng tăng dần số luợng user truy cập cùng 1 lúc để tìm điểm nguỡng chết (peak) của hệ thống

+ Hay khách hàng cũng đưa luôn ra hiệu năng của 1 số trang như login < 3s, search (tìm kiếm) < 3 s…

Câu hỏi 20: Test Web App/Mobile khác Test Web ở desktop/ PC như thế nào?

* + Về giao diện, kích thuớc: màn hình mobile nhỏ hơn trên máy tính rất nhiều nên hay phát sinh lỗi giao diện. Trên mobile màn hình cũng có thể xoay ngang – dọc đuợc.
  + Về tính tuơng thích: Mobile thì có nhiều dòng điện thoại khác nhau

Ví dụ như android thì có nhiều version, chạy trên nhiều dòng máy đt khác nhau nhu HTC, SamSung, Sony… nên môi truờng test cũng rộng hơn trên PC

* + Về bộ nhớ, tốc độ của mobile cũng kém hơn nhiều so với PC nên phần mềm hay bị crash, treo….
  + Về trải nghiệm nguời dùng: trên mobile cũng khác PC, cách hành vi, thao tác, chức năng... trên mobile cũng đòi hỏi phải hợp lý cho 1 màn hình bé nhỏ và cấu hình cũng thấp hơn PC rất rất nhiều
  + Về xung đột với các chức năng của mobile: nhu vừa bật nhạc trên mobile vừa test app, vừa sử dụng app và có cuộc điện thoại gọi đến hay có tin nhắn…
  + Về chế độ online/ offline mode: trên mobile thì sẽ có những chức năng chạy đuợc khi có internet/ online và có những tính năng vẫn làm việc hay sử dụng đuợc khi ko có internet/ offline mode

18

Câu hỏi 21: Test report là gì – Báo cáo kiểm thử dùng để làm gì ? Nội dung báo cáo kiểm thử gồm những gì?

-Test report dùng để thống kê số lượng lỗi còn tồn( chưa được fix/ sửa) – Test report được dùng để thống kê mức độ nghiêm trọng của lỗi ( bao nhiều lỗi major, bao nhiêu lỗi high, low… )

* + Dùng để xem / báo cáo lỗi còn nhiều ở chức năng nào, lập trình viên nào còn nhiều lỗi
  + Chức năng nào trễ tiến độ bàn giao test, hay dev fix bug chậm, gây nhiều lỗi…
  + Chức năng nào đã hoàn thành test, chức năng nào chưa

19

* + ***Tóm lại*** : Dựa vào test report để đánh giá chất lượng dự án đang như thế nào . còn nhiều lỗi hay ko .lỗi đang còn nhiều ở chức năng nào, mức độ nghiêm trọng ra sao ? có kịp tiến độ bàn giao, triển khai cho kháng hàng hay ko . hoàn thành đúng theo dealine ko.

Câu hỏi 22: Em biết những quy trình phát triển phần mềm nào ?

* + Em biết một số quy trình phổ biến hiện nay như :

+ Waterfall – mô hình thác nước

+ V model – Mô hình chữ V

+ Mô hình Agile / quy trình scrum.

Câu hỏi 23: Mô hình Agile/Scrum là gì? Mô hình này khác với các quy trình khác ở điểm nào ?

* + Mô hình này không đưa toàn bộ yêu cầu nghiệp vụ của hệ thống vào code và testing cùng 1 lúc ( như các quy trình truyền thống khác ) mà sẽ chia các yêu cầu ra thành từng giai đoạn. Mỗi giai đoạn chỉ làm 1 số lượng y/c nhất định

VD : cả 1 hệ thống sẽ có 200 màn hình, nếu apply quy trình agile vào thì mình sẽ chia thành nhiều giai đoạn nhỏ / Sprint . Mỗi sprint chì làm 20 màn hình rồi demo cho KH xem.

* + Mỗi sprint chỉ kéo dài từ 1 đến 4 tuần ( không dài hơn 1 tháng)
  + Đầu sprint sẽ họp lên kế hoạch sẽ làm những y/c nào . Sau đó sẽ thực hiện code và test. Cuối sprint phải hoàn thiện xong cả code và test có thể demo cho Khách hàng
  + Sau khi hoàn thành sprint 1 thì tiếp tục làm sprint 2… cho đến khi hết các màn hình hoặc yêu cầu của KH.

20

* + Trong mỗi sprint sẽ có họp hàng ngày ( daily meetiing ). Thường chỉ họp ngắn 15-20 phút .Mỗi thành viên sẽ báo cáo với cả team : hôm qua tôi đã làm được gì- hôm nay tôi sẽ làm gì – có khó khăn gì không.
  + Do mỗi sprint rất ngắn ngày nên yêu cầu hiệu quả làm việc phải cao, đúng tiến độ để đảm bảo cuối sprint là phải hoàn thành xong đúng kế hoạch
  + Ưu điểm của quy trình này là phù hợp với những yêu cầu/nghiệp vụ hay thay đổi. Hoặc những hệ thống nghiên cứu làm theo giai đoạn ngắng ngày,có thể nhìn thấy những rủi ro và những điểm chưa phù hợp để thay đổi sớm.
  + Người điều hành dự án gọi là scrum master, Khách hàng là người đánh giá xem phần nềm đã làm theo đúng yêu cầu/nghiệp vụ chưa thì gọi là product owner

Câu hỏi 24: Viết thử một bug bằng tiêng anh hoặc tiếng việt

#### Bằng Tiếng Việt :

1. ***Summary*** (Mô tả bug ): [Màn hình - Thêm mới sản phẩm] Không hiển thị message thông báo bắt buộc nhập khi để trống trường “Mã Sản Phẩm “.
2. ***Description*** ( Mô tả các bước tái hiện bug ):
   * ***Điều kiện*** : Login vào bằng account quản trị viên ( acc: QTV / Pass : 1234567@)

#### Các bước tái hiện bug:

1. Không nhập gì mà trường “Mã Sản Phẩm “
2. Các trường còn lại nhập giá trị hợp lệ
3. Nhấn [Thêm Mới]
   * + ***Actual result*** (Kết quả thực tế ) :

3.Không hiển thị message thông báo bắt buộc nhập khi để trống trường



21

“Mã Sản Phẩm “

* + - **Expected result** (Kết quả mong đợi ): Hiển thị thông báo lỗi “ Đây là trường bắt buộc” ở dưới textbox “Mã Sản Phẩm”, highligh và focus chuột vào textbox ‘Mã Sản Phẩm “



22

### Log bug bằng Tiếng Anh:

***Summary*** : [Add New Product screen] Not show error message when “ProductID” is blank

***Description*** :

* *Account : QTV/ Pass*: 123456@

#### Steps to reproduce:

1. “ProductID” textbox is blank
2. Input valid data into remain fields
3. Click on [Add New] button

#### Actual results:

3.System does not show error message on Add New Product screen ( *see attachted image* )

#### Expected result :

3. Should showed error message “ This is required field” under “ProductID” textbox

Câu hỏi 25: Theo bạn để trở thành tester thì cần những gì ? ( Hoặc test thì cần gì ? )

* Theo em Tester ít nhất phải biết sử dụng máy tính, internet, excel , biết những thuật ngữ chuyên môn của CNTT. Nếu có nền tảng về CNTT thì càng tốt ạ
* Cần có kiến thức về phần mềm, về kiểm thử phần mềm như : Biết quy trình sx phần mềm, quy trình kiểm thử phần mềm
* Biết các kỹ thuật thiết kế test case và đôn đốc lập trình viên fix bug,biết ứng biến nhanh trong mọi tình huống …
* Cần phải biết tùy biến ,mềm dẻo, cương quyết khi cần ( Ý này nói cũng được

, ko nói cũng không sao )

* Biết dùng SQL căn bản để khi test có thể kết hợp với truy vấn dữ liệu như

23

select, Insert into, update,delete, join ,group by , order by, … ( Ý này nếu bạn nào thạo SQL mới nói, không thì ko nói )



24

Câu hỏi 26: Tester làm thế nào để xác đinh được mức độ của regression test là phải test hồi quy tất cả hay chỉ test 1 phần ?

* Em sẽ dựa kinh nghiệm và sự hiểu biết của tester đối với nghiệp vụ của hệ thống để quyết đinh test hồi quy tất cả hay chỉ test 1 phần
* Ngoài ra em có thể trao đổi thêm với leader hoặc PM hoặc dev để quyết định

Tóm lại : Test hồi quy 1 phần hay thực hiện test hồi quy tất cả tùy thuộc vào mức độ nghiêm trọng và thời gian cho phép

Câu hỏi 27: Test hồi quy sẽ được thực hiện như thế nào ?

* Đầu tiên sẽ sẽ tập trung vào nơi nào mà sự thay đổi bị xảy ra
* Tiếp theo em sẽ cân nhắc test hồi quy cho những chức năng chính, nghiệp vụ chính của hệ thông
* Test hồi quy 1 phần hay thực hiện test hồi quy tất cả tùy thuộc vào mức độ nghiêm trọng và thời gian cho phép

Câu hỏi 28: Test API là gì ? ( Em biết gì về test API )

Khi test API là giao diện không có nên việc thực hiện test API là sẽ không thực hiện test trên giao diện mà phải dùng tool như SoapUI để gửi đi request tới API đó và nhận response là kết quả trả về để so sánh kết quả mà website mình lấy về để hiển thị trên web có đúng không.

**Câu hỏi 28.1: API Testing là gì:**

Là một loại kiểm thử tập trung vào xác minh chức năng >> nó liên quan đến việc KT giao tiếp giữa các phần mềm khác nhau >> đảm bảo trao đổi và tích hợp dữ liệu

**Câu hỏi 28.2: APIs làm việc như thế nào**

API đóng vai trò trung gian và là cầu nối giữa các hệ thống với nhau để giao tiếp và trao đổi dữ liệu với nhau >> cho phép các ứng dụng truy cập và sử dụng các chức năng hay dịch vụ của phần mềm khác

**Câu hỏi 28.3: Các công cụ được sử dụng nhiều nhất cho API Testing là gì?**

Postman, SoapUI, RestAssured, JMeter và Assertible

**Câu hỏi 28.4: Các phương thức HTTP được REST hỗ trợ là gì (4 methods)**

GET: Gọi dữ liệu

POST: cập nhật dữ liệu

PUT: Tạo mới tài nguyên

DELETE: Xóa tài nguyên

**Câu hỏi 28.5: URI là gì?**

**Câu hỏi 28.6: Mã trạng thái HTTP là gì?**

Các mã tiêu chuẩn được sử dụng trong HTTP tương ứng với các trạng thái hoàn thành của máy chủ

Mã HTTP thường gặp:

+ 200 >> Hoàn thành

+ 400 >> cú pháp trong yêu cầu và yêu cầu bị từ chối >> do thiếu dữ liệu hoặc lỗi xác thực

+ 401 >> FORBIDDEN >> Người dùng không có quyền truy cập (Hệ thống từ chối quyền)

+ 404 >> LỖI chỉ ra rằng không thể tìm thấy phương thức được yêu cầu

+ 500 >> Lỗi sự cố từ máy chủ



25

### Câu hỏi 29: Tại sao em lại chọn làm nghề Tester ?

### + Nếu bạn nào học CNTT thì trả lời như sau :

Em học cntt nhưng khả năng code của em chưa thực sự tốt và em có thiên hướng muốn theo nghề tester vì em tự thấy mình có những tố chất , khả năng trở thành một tester giỏi ( tố chất : chăm chỉ, tỉ mỉ ,cẩn thận, có tư duy phân tích vấn để ,năng động, sáng tạo,nhanh nhạy, học hỏi nhanh, hiểu nhanh … )

### + Nếu bạn học trái ngành thì nên trả lời như sau :

* Em rất yêu thích lĩnh vực cntt ,nhưng ngày trước em thi đại học lại không vào học được khoa cntt như em mong muốn. Sau khi ra trường em vẫn ấp ủ ước mơ theo đuổi được làm việc trong lĩnh vực cntt . Và sau một thời gian tìm hiểu thì em đã biết đến nghề tester. Em cảm thấy rất vui vì đã tìm ra được một nghề khá là phù hợp với mong ước và khả năng của em.
* Bởi bản thân em thấy em có khá nhiều tố chất để có thể trở thành một tester chuyên nghiệp , đó là em đã tìm học một khóa học cơ bản về testing, em cũng tự học/tự tìm hiểu thêm về nghề, tự trao đồi kiến thức/ thuật ngữ hay dùng trong cntt.
* Ngoài ra em thấy mình có những tố chất của một tester ( tố chất : chăm chỉ, tỉ mỉ

,cẩn thận, có tư duy phân tích vấn để ,năng động, sáng tạo,nhanh nhạy, học hỏi nhanh, hiểu nhanh … )

Câu hỏi 30: Em đã tham gia dự án nào chưa ? có thể các bạn sẽ bị hỏi chi tiết làm gì trong dự án đó và test như thế nào nếu các bạn có ghi trong CV .

Lưu ý là các dự án, bài tập testcase ,thực hành test mà ghi trong CV các bạn phải kiểm soát được để trả lời nếu bị hỏi đến.

Thông thường đi phỏng vấn hay bị hỏi những gì có ghi trong CV của bạn. Nếu những gì mà bạn không chắc chắn thì không nên ghi vào CV.

Nếu được hỏi thì bạn cứ mạnh dạn trả lời là có tham gia dự án rồi ( và nói tên dự án là những dự án mà mình đã được viết testcase , thực hành ở trên lớp cũng ok

) .

Bạn nên nói là được tham gia dự án …. , em được tiếp cận,phân tích requirement

, rồi được giao viết testcase một số màn hình và thực hiện test theo testcase của

26

em đã viết và testcase của tester khác trong dự án viết ạ.

Cuối ngày em viết email report để báo cáo kết quả công việc hàng ngày của em cho leader ( vidu như ngày hôm nay em viết testcase cho những màn hình/ function nào ,hoàn thành được bao nhiêu % . Hoặc execute màn hình/ function nào : bao nhiêu case passed, failed/N/A … có bao nhiêu bug high/ normal /low hoặc đã close được bao nhiêu bug, bao nhiêu bug vẫn open … )

Câu hỏi 31: Em hãy Liệt kê testcases của chức năng Login?

**Case1:** Đăng nhập thành công với tài khoản tồn tại trong DB

**Case 2:** Đăng nhập với 1 tài khoản ko tồn tại trong DB -> hệ thống Thông báo Tên đăng nhập hoặc mật khẩu sai/ Highlight và focus vào textbox **Case 3:** Nhập username hợp lệ (tồn tại trong DB) và sai password (

ko tồn tại trong DB) -> hệ thống Thông báo Tên đăng nhập hoặc mật khẩu ko đúng/ Highlight và focus vào textbox

**Case 4:** Nhập username sai và password hợp lệ -> hệ thống Thông báo Tên đăng nhập hoặc mật khẩu không đúng/ Highlight và focus vào textbox

**Case 5:** Username và password để trống -> hệ thống Thông báo Bạn chua nhập tên đăng nhập/ Bạn chua nhập mật khẩu / Highlight và focus vào textbox **Case 6:** Nhập ký tự đặc biệt cho username -> hệ thống Thông báo Tên đăng nhập ko đúng/ Highlight và focus vào textbox

**Case 7:** Nhập khoảng trắng đầu và cuối cho username và mật khẩu hợp lệ -> Đăng nhập thành công, trim space khoảng trắng đầu và cuối username.

**Case 8:** Đăng nhập với username viết hoa viết hoa và mật khẩu hợp lệ -> Đăng nhập thành công. Hệ thống ko phân biệt username là chữ hoa, thuờng

**Chú ý:** là chỉ username viết HOA thì đăng nhập mới thành công, còn password viết hoa là sẽ ko thành công do password phân biệt chữ hoa và thường.

Câu hỏi 32: Hoặc hỏi test boundary value ( giá trị biên cho khoảng từ 0->20? (Hoặc Em hãy liệt kê testcases cho dải số từ 0 đến 20?)

Các giá trị cần test: là 0, 20, 21, -1

27

Chỗ dưới này là giải thích thêm thôi nhé, hỏi đến thì mới nói ra: Áp dụng kỹ thuật phân tích giá trị biên ta chọn đuợc các case sau:

Case 1: Nhập giá trị là -1 => hiển thị lỗi Case 2: Nhập giá trị là 0 => pass

Case 3: Nhập giá trị với 10 => pass Case 4: Nhập giá trị với 20 => pass



28

Case 5: Nhập giá trị với 21 => hiển thị lỗi

Case 6: Để trống không nhập gì hay nhập ký tự không phải dạng chữ => hiển thị lỗi

Câu hỏi 33: Em hãy Liệt kê testcases chức năng QUÊN Mật khẩu

*Nghiệp vụ : Khi User quên mật khẩu, User sẽ click vào để chọn Re-set lại mật khẩu , để có mật khẩu mới*

Case 1: Test tạo mới mật khẩu mới thành công khi Nhập Mật khẩu mới + Nhập lại mật khẩu mới trùng nhau

Case 2: Test tạo mật khẩu mới ko thành công khi nhập Mật khẩu mới và Nhập lại mật khẩu mới ko trùng nhau => Hệ thống hiện thị thông báo, highlight và focus vào textbox Nhập lại mật khẩu

Case 3: Test maxlength 2 textbox, nhập quá max length cho 2 text box

Case 4: Test tính bắt buộc của 2 textbox, để trống tất cả, click vào button Tạo mật khẩu, hệ thống hiện thị thông báo bắt buộc nhập, có cả highlight và focus

Case 5: Nhập mật khẩu có 1 ký tự thì hệ thống thong báo Mật khẩu tối thiểu là 6 ký tự ( tất nhiên là cái này cũng tùy thuộc vào nghiệp vụ yêu cầu của khách hàng)

Case 6: Test mã hóa của mật khẩu, hiện thì là \*\*\*\*\* khi nhập mật khẩu

Case 7: Kiểm tra Rule của mật khẩu mới ở màn reset MK phải giống với rule khi tạo mật khẩu mới ở màn Đăng ký tài khoản mới ( Vidu: Ở Màn ĐK Tài Khoản mới, rule tạo mk phải gồm min là 8 ký tự + bao gồm cả chữ hoa chữ thường thì rule ở màn reset MK phải gồm min8 ký tự + bao gồm cả chữ hoa chữ thường )

…

29

Câu hỏi 33: Độ nghiêm trọng (Priority) và mức độ ưu tiên (Severity) của bug là gì đưa ra dẫn chứng cho từng loại ? Nêu ví dụ mà một bug có độ nghiêm trọng cao và độ ưu tiên thấp

1. **Độ nghiêm trọng (Priority)** :Đã bug thì sẽ phải sửa .Tuy nhiên , đội phát triển khó có thể sửa hết tất cả các bug một lượt cũng như không đáng để sửa hết các bug.Do đó, đội phát triển sẽ phải cần đến độ ưu tiên của con bug để biết được bug nào cần được sửa trước , bước nào sau.

Độ ưu tiên của con bug được chia thành 4 cấp độ :

+ Mức độ 1 (Immediate/Critical ) Bug sẽ phải sửa ngay

+ Mức độ 2 (High) độ ưu tiên cao, công việc bị ngăn trở rất nhiều nếu như lỗi vẫn chưa đc sửa

+ Mức độ 3 (Medium) : độ ưu tiên trung bình; công việc sẽ gặp vài khó khăn nếu như lỗi vẫn chưa đc sửa

+ Mức độ 4 (low) độ ưu tiên thấp nhất; công việc không bị ảnh hưởng nhưng lỗi vẫn phải đc sửa

1. **Mức độ ưu tiên (Severity Bug)**: thường chỉ mức độ tác động của con bug đó đến sản phẩm

+ Mức độ 1 (Critial): rất nghiêm trọng , có thể làm phần mềm chết cứng và không sử dụng được

+ Mức độ 2 (Major): Nghiêm trọng ,ảnh hưởng đến chức năng chính của sản phẩm

+ Mức độ 3 (Minor): ảnh hưởng đến các chức năng phụ của sản phẩm

+ Mức độ 4 (Cosmetic): không ảnh hưởng đến chức năng hay hiệu năng nhưng ảnh hưởng về vấn đề thẩm mỹ hoặc thông báo sai chính tả.

-Ví dụ một bug có **độ nghiêm trọng cao(Severity Bug) và độ ưu tiên thấp (Pirority Bug)** : 1 app sau khi thực hiện rất nhiều thao tác bị crash độ nghiêm trọng rất cao nhưng độ ưu tiên sẽ thấp hơn một chức năng chính nào đó bị sai không vào được màn hình hoặc không thực hiện được một chức năng nào đó.

* Ví dụ một lỗi có **độ nghiêm trọng thấp (Severity Bug) và độ ưu tiên cao (Pirority Bug)** Khi mở một trang web của công ty tên công ty hiển thị bị sai thì mức độ nghiêm trọng của nó không cao nhưng độ ưu tiê n của nó đc đặt lên hàng đầu .



30

Câu hỏi 34: Regression testing khác gì re-test ?

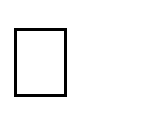
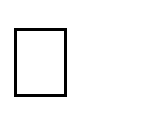
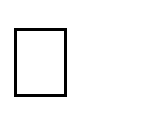
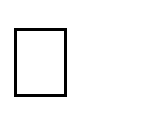
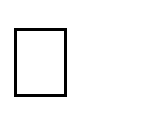
31

Regression testing là test các chức năng liên quan có bị ảnh hưởng bởi lỗi sau khi đc sửa xong hay không còn re-test là test chức năng mình log bug đã được fix hay chưa.

Câu hỏi 35: Một báo cáo công việc kiểm thử (test report) gồm những gì? Và ích lợi của bảng báo cáo này?

Test Report t hường chứa các nội dung như:

Số lượng test cases đã viết/ số lượng testcase đã test Số lượng test cases passed/failed



Số lượng defects tìm ra và status, severity của defects Số lượng defects trên từng module

Các vấn đề liên quan đến testing, bản build, tiến độ sửa lỗi…

Báo cáo test cũng thể hiện tiến độ kiểm thử, tiến độ sửa lỗi và số lượng lỗi được tìm thấy hay còn tồn của dự án. Báo cáo kiểm thử cũng là công cụ để phục vụ cho đánh giá hay giám sát dự án có kịp tiến độ hay không, có thể bàn giao hay release cho khách hàng hay không và các vấn đề cần giải quyết khi mà số lượng lỗi còn nhiều, gây ra các rủi ro về tiến độ hoàn thành của dự án để có những điều chỉnh kịp thời

Câu hỏi 36: Khi nào thì nên sử dụng kiểm thử tự động?

* Dự án cần phải test đi test lại nhiều gây tốn nguồn lực kiểm thử mà còn tạo cảm giác nhàm chán.
* Chỉ nên áp dụng auto test khi có tính ổn định, và không có sự thay đổi về mặt chức năng/nghiệp vụ.
* Không phải cái gì cũng đều test tự động được hết cả mà vẫn cần đến test manual đóng vai trò quan trọng trong dự án. Trong trường hợp mà nghiệp vụ sửa đổi thay đổi nhiều thì không nên sử dụng auto test do việc sửa script liên tục cũng mất nhiều thời gian.

Khi thực hiện performance testing/ load testing thì test tự động là sự lựa chọn duy nhất.

# PHẦN II: SQL

Câu hỏi 1: SQL là gì?

Trả lời:

SQL là viết tắt của cụm từ "Structured Query Language." - ngôn ngữ truy vấn mang tính cấu trúc. Nó được thiết kế để quản lý dữ liệu trong một hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS).

SQL là ngôn ngữ cơ sở dữ liệu, được dùng để tạo, xóa, lấy các hàng và sửa đổi các hàng,

…

Tất cả hệ quản trị cơ sở dữ liệu (Database Management System - DBMS) như MySQL, Oracle, MS Access, Sybase, Informix, Postgres và SQL Server đều sử dụng SQL như là ngôn ngữ cơ sở dữ liệu chuẩn.

Câu hỏi 2: Một bảng có được 2 khóa chính không?

Trả lời : Theo em được biết thì 1 bảng chỉ có 1 Khóa Chính và có thể có nhiều khóa phụ

Câu hỏi 3: Câu lện nào dùng để truy vấn hay cần tìm 1 dữ liệu trong DB?

Select \* from table\_name where column\_name=’value’; Vidu: Select \* from User where username = ‘hang’;



33

Câu hỏi 4: Khi cần Thêm mới 1 dữ liệu trong DB thì dùng câu lệnh gì ?

Insert into table\_name (column\_name1, column\_name2) values (‘value1’, “value2”);

Vidu:

Insert into San\_Pham ( ten\_sp,gia) values (“Dieu Hoa SamSung “ , “12000000”)

Câu hỏi 5: Khi cần sửa 1 dữ liệu trong DB thì dùng câu lệnh gì?

Update table\_name set column\_name= ‘giá trị mới’ where column\_name= ‘giá trị’; Vidu:

Update San\_Pham set gia = ‘15000000” where ID =’1253’;

Câu hỏi 6: Nếu cần xóa 1 dữ liệu trong DB thì dùng câu lênh gì?

Delete from table\_name where column\_name=’values’ Vidu:

Delete from SanPham where ID =’12242’



34

Câu hỏi 7: Nếu cần đếm số lượng trong DB thì dùng câu lệnh gì?

Select count (\*) from table\_name; Vidu:

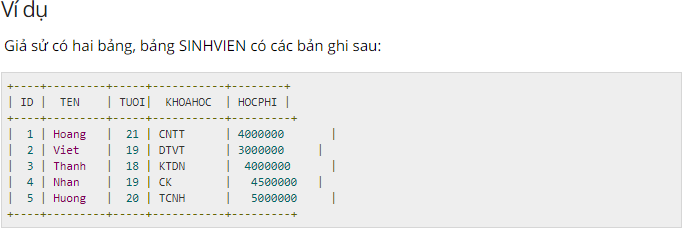
Select count (column\_name) from table\_name;

Câu hỏi 8: Nếu muốn join 2 bảng trong DB thì e dùng câu lệnh nào?

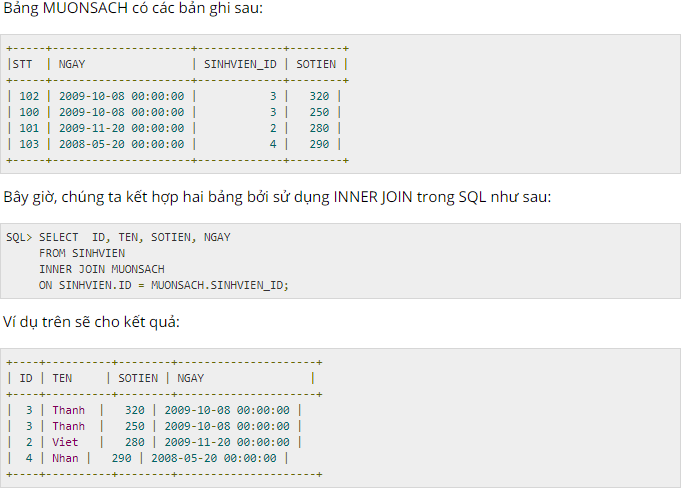
Cần join 2 bảng hay nhiều bảng với nhau thì sẽ sử dụng câu lênh JOIN nhé

Select \* from table1 inner join table2 on table1.column\_name1 = table2.column\_name1; Vidu:

Select table1.column\_name from table1 inner join table2 on table1.values1 = table2.values1;



35



Câu hỏi 9: Câu lệnh Order by dùng để làm gì ?

Trả lời : Dùng để sắp xếp lại dữ liệu trả về từ câu truy vấn cho dễ nhìn ạ

Câu hỏi 10 : Làm thế nào để lấy được tất cả các bản ghi trong một bảng xác định?

Trả lời: Để lấy được tất cả các bản ghi từ một bảng, chúng ta cần sử dụng câu lệnh dưới đây:

Select \* from table\_name;



36

Câu hỏi 11: Bạn thêm một cột vào bảng như thế nào?

Trả lời: Để thêm một cột vào trong bảng thì câu lệnh dưới đây được sử dụng:

ALTER TABLE table\_name ADD (column\_name);

Câu hỏi 12 : Định nghĩa câu lệnh SQL Delete.

Trả lời: Câu lệnh "DELETE" được sử dụng để xóa một dòng hoặc nhiều dòng từ một bảng dựa trên những điều kiện do người dùng chỉ định. Cú pháp cơ bản như sau:

DELETE FROM table\_name

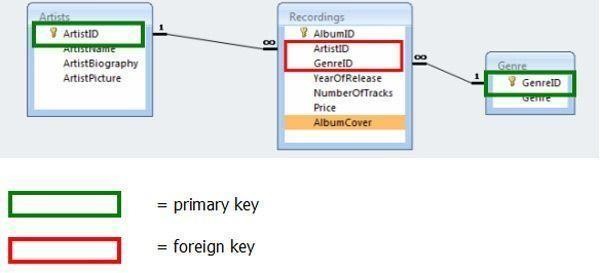
WHERE <Condition>



37

Câu hỏi 13: Primary key là gì?

Trả lời: Một PRIMARY KEY- Khóa chính là một trường trong một bảng mà nhận diện một cách duy nhất mỗi hàng/bản ghi trong một bảng dữ liệu. Các PRIMARY KEY phải chứa các giá trị duy nhất, không được sử dụng lại. Một cột là PRIMARY KEY thì không được phép có giá trị NULL. Một bảng chỉ cho phép tối đa một PRIMARY KEY, **bao gồm một trường đơn hay nhiều trường**. Mỗi bảng đều cần có khóa chính.



Câu hỏi 19: Foreign keys là gì?

Trả lời: FOREIGN KEY - Khóa ngoại là một trường trong bảng cơ sở dữ liệu, đó là khóa chính trong một bảng khác được thêm vào tạo ra sự liên quan giữa hai bảng. Trong bảng, giá trị của FOREIGN KEY có thể chấp nhận giá trị NULL, cũng như các giá trị trùng nhau.



38

Câu hỏi 20 : Một bảng có thể có nhiều hơn một khóa ngoại được không?

Trả lời: Có, một bảng có thể có nhiều khóa ngoại và chỉ có một khóa chính.

Câu hỏi 21: Trường BOOLEAN có thể nhận các giá trị nào.

Trả lời: Giá trị của trường BOOLEAN có thể nhận 1 trong 2 giá trị sau: 1(TRUE) và 0(FALSE).

# Phần III - Câu hỏi tình huống thường gặp :

Câu hỏi 1: Khi Dev/ Lập trình viên không công nhận bug/ lỗi của em, em sẽ giải quyết thế nào ?

Em sẽ thực hiện tái hiện lại lỗi cho Dev xem, nếu vẫn ko công nhận thì em sẽ confirm/ xác nhận lại với test leader hoặc PM (QTDA) hoặc BA ( ng làm giải pháp) xem đấy có đúng là lỗi ko

Câu hỏi 2: Nếu phải test số lượng lớn testcase trong thời gian ngắn ( tức là ko đủ thời gian để test hết) thì phải làm thế nào?

* Thực hiện test các testcases thông luồng, chức năng chính, chức năng nghiệp vụ truớc. Các testcase quan trong là phải đuợc test truớc. Đảm bảo phần chức năng chính là chạy đuợc.
* Sau đó tùy vào tình hình dự án để làm thêm giờ hay làm cả cuối tuần để hoàn thành đuợc deadline công việc.

39

Câu hỏi 3: Em phải test một phần mềm mới nhưng người quản lí gia hạn thời gian để test ít. Với trình

độ của bạn cần nhiều thời gian hơn để test.Em sẽ phải làm gì?

* Thực hiện test các test thông luồng, chức năng chính, chức năng nghiệp vụ truớc. Các testcase quan trong là phải đuợc test truớc. Đảm bảo phần chức năng chính là chạy đuợc.



40

* Thực hiện làm overtime để hoàn thành đuợc công việc.
* Trao đổi thêm với nguời quản lý với thời gian ngắn nhu vậy, có thể tăng cuờng thêm nhân sự test để hỗ trợ trong giai đoạn quan trọng, cần hoàn thành gấp để đảm bảo chất luợng và hoàn thành đúng tiến độ.

Câu hỏi 4: Là tester mới, khi phát hiện ra bug nhưng em không chắc chắn thì em sẽ làm như thế nào

?

* Em sẽ đi confirm với test leader hoặc BA của đự án hoặc hỏi mọi người có kinh nghiệm hơn trong dự án mình.

Câu hỏi 5: Nếu em vào làm tester của một dự án không có tài liệu nghiệp vụ thì em sẽ làm như thế nào ?

Nếu dự án không có tài liệu nghiệp vụ nhưng đã có hệ thống cũ của khách hàng thì em sẽ vào hệ thống cũ của khách hàng để dùng thử, để học và hiểu hệ thống cũ đó , đặt câu hỏi cho BA/ PM hoặc Khách Hàng để em có thể hiểu sâu hơn về nghiệp vụ hệ thống.

Câu hỏi 6: Nếu dự án / màn hình em được giao vào làm tester nhưng không có đủ tài liệu hoặc tài liệu nghiệp vụ không chi tiết thì em sẽ làm thế nào ?

* Việc đầu tiên em sẽ đi xác nhận lại vấn đề tài liệu đặc tả nghiệp vụ không đầy đủ với cả BA và PM của dự án
* Trong trường hợp hiện tại chưa thể có thêm nghiệp vụ hay do đang vừa làm vừa nghiên cứu thì em sẽ dựa trên luồng nghiệp vụ hiện tại để đọc hiểu, phân tích và thiết kế testcase, thực hiện test . Đồng thời em sẽ make question cho BA/PM để họ có thể giúp em làm sáng tỏ những ý mà em màn hình em được giao làm . Hoặc em sẽ tạm thời viết test case theo luồng nghiệp vụ hiện tại đang có.
* Trường hợp chưa có người tham gia phân tích hay viết tài liệu chi tiết hơn thì em có thể tham gia hỗ trợ dự án bổ xung , viết hay thu thập nghiệp vụ cho dự án.



41

Câu hỏi 7: Em đã từng bỏ qua bug nghiêm trọng nào chưa ? Nếu đã từng bạn sẽ xử lý như thế nào ?

* Em chưa gặp trường hợp bỏ qua lỗi nghiêm trọng nào . Nhưng nếu e gặp phải tình huống này em sẽ nhận đây là lỗi của em và sẽ rút kinh nghiệm cho lần sau ạ. Em sẽ cố gắng hơn để không lặp lại lỗi này nữa.
* Tiếp theo e sẽ yêu cầu dev fix bug nghiêm trọng này luôn cho em và thực hiện test lại chức năng bị lỗi và đánh giá việc sửa lỗi này có ảnh hướng đến chức năng đã test xong không. Nếu có ảnh hưởng em sẽ thực hiện test hồi quy lại chức năng bị ảnh hướng đó.

Câu hỏi 8: Nếu em nhận được bản build có nhiều lỗi nghiêm trọng em sẽ làm như thế nào ?

* Khi nhận được 1 bản build từ dev bao giờ e cũng thực hiện smoke test , test qua những chức năng chính, quan trọng để kiểm tra độ ổn định của hệ thống sau mỗi bản build mới.Để kiểm tra độ ổn định của ứng dụng có đủ để bắt đầu thực hiện test hay không.
* Trong quá trình test nếu phát hiện bản build bị mắc nhiều lỗi nghiêm trọng hoặc động chỗ nào cũng lỗi, hay những chức năng trước đó đã chạy ok rồi mà trong bản build này lại bị mắc lỗi thì em sẽ trả lại bản build này cho dev và yêu cầu kiểm tra lại và cho bản build khác.

Câu hỏi 9: Nếu Tester tìm thấy lỗi trên máy test nhưng do dev không nhận lỗi này để fix nguyên nhân máy dev không bị lỗi này thì em sẽ làm như thế nào ?

* Thứ nhất : em sẽ yêu cầu dev chạy trên môi trường test là link test ở trên server , không được chạy bản trên máy local của dev để cùng chung môi trường.
* Ngoài ra, cũng có thể do môi trường khác nhau về trình duyệt giữa máy dev là test là không giống nhau, nên cần phải kiểm tra là cũng chạy trên cùng 1 trình duyệt giống nhau, version giống nhau.

42

* Đồng thời e cũng nhờ tái hiện bug này trên 1 máy cua tester / dev khác nữa để xem có xảy ra lỗi không
* Còn trong trường hợp ko giải quyết được thì em có thể hỏi test leader/ BA/PM để giải quyết vấn đề.

Câu hỏi 10: Khi nào thì chúng ta dừng test ? Hoặc khi nào thì việc kiểm thử phần mềm kết thúc?

Dừng test có thể theo yêu cầu của PM quản lý dự án dừng test vì lý do nào đấy. Dừng test khi tất cả các chức năng đã đuợc test xong, không còn tồn tại lỗi phần mềm nữa. Đảm bảo tất cả các chức năng đã chạy đúng theo Đặc tả phần mềm và toàn bộ lỗi đã đuợc sửa và test xong hết.

Câu hỏi 11: Kiểm thử nên bắt đầu ở giai đoạn nào trong chu kỳ phát triển phần mềm?

* Kiểm thử thực hiện càng sớm càng tốt. Theo quy trình phát triển phần mềm thì kiểm thử đuợc thực hiện sau giai đoạn coding. Code xong thì Dev sẽ build và bàn giao cho Tester thực hiện test.
* Thực tế thì tester tham gia sớm hơn, **chỉ cần có tài liệu Đặc tả Yêu cầu/ nghiệp vụ** là tester thực hiện tìm hiểu nghiệp vụ dự án và thực hiện viết testcases.

Câu hỏi 12: Khi được hỏi em muốn mức lương bao nhiêu?

* Với một sinh viên mới ra truờng chua đi làm thực tế hay vị trí ứng tuyển là Fresher Tester thì các bạn cứ đề xuất mức luơng là 6 triệu ( đây là mức sàn)
* Tuy nhiên cũng tìm hiểu xem tin tuyển dụng của công ty có ghi mức luơng ko, vì cũng nhiều nơi họ ghi luơng từ 7 triệu cho vị trí chua có kinh nghiệm và trả đúng thế thật. Nếu có ghi mức luơng thì ko nên order mức thấp hơn.
* Ngoài ra, trong buổi phỏng vấn thấy mình trả lời tốt, thì cứ mạnh dạn đề xuất mức luơng cá nhân mong muốn.
* Vì sao? Em thấy khả năng của mình phù hợp với mức luơng này. Em cũng sẽ cố gắng cống hiến và chăm chỉ làm việc và làm thực sự hiệu quả để xứng đáng với những gì công ty đã chi trả ạ.

43

* Ngoài ra nếu nhà tuyển dụng hỏi em là mức đó theo em là có quá cao không thì em cứ mạnh dạn nói rằng : em muốn đi làm để có tiền trang trải cuộc sống, đồng thời muốn có thêm tích lũy một chút phòng lúc ốm đau hoặc giúp đỡ gia đình ( nuôi em ăn học… ) .Nếu cty trả e mức lương như vậy e hoàn toàn có thể yên tâm cống hiến, công tác , không phải lo lắng vì vấn đề kinh tế nữa ạ.

Câu hỏi 13: Định hướng của em như nào trong khoảng 2-3 năm tới ?

* Em định sẽ học thêm tiếng anh, trau dồi thêm kinh nghiệm, kiến thức về testing
* Thi lấy chứng chỉ ISTQB , học thêm về automation để có thể tiến xa hơn nữa trong công việc ạ

Câu hỏi 14: Bạn tự nhận xét mình có những ưu điểm/khuyết điểmgì?

Em tự nhận thấy em có những ưu điểm sau ạ: Em rất cẩn thận, tỉ mỉ, luôn sẵn sàng lắng nghe, học hỏi từ mọi người và em học khá nhanh, dễ dàng thích nghi với công việc và môi trường mới. Ngoài ra em khá vui vẻ , dễ hòa đồng với mọi người.

Nhược điểm của em là em nói hơi nhiều , ( ý này bạn có thể nói như vậy vì nó ko ảnh hưởng đến cv của em , mà tester nói hơi nhiều cũng ko sao cả ) hoặc có thể nói em bị bạn bè nói là hay săm soi lời người khác nói , hay nghĩ ra quá nhiều giả thiết, trường hợp cho 1 chuyện tưởng là đơn giản ( – bởi Tester thường hay săm soi bug, hay phải nghĩ ra tất cả các case có thể xảy ra nên bạn nói những ý này ra là nhược điểm của bạn thì khá là ok )

Câu hỏi 15: Nếu làm chung dự án với developer rất nóng nảy, bạn làm thế nào để dung hòa công việc với họ?

* Đầu tiên em sẽ cố gắng để làm quen, nói chuyện ngoài lề công việc với dev đó để xây dựng mối quan hệ tốt hơn
* Khi nói chuyện công việc , hoặc liên quan đến bug em sẽ vui vẻ, mềm mỏng với bạn dev này .
* Nếu bị gây khó dễ , vướng mắc, mâu thuẫn mà tự bản thân e ko thể giải quyết được với bạn dev này thì em sẽ raise lên test leader hoặc PM , dev leader ạ



44

Câu hỏi 14: Bạn tự nhận xét mình có những ưu điểm/khuyết điểmgì?

* Em tự nhận thấy em có những ưu điểm sau ạ: Em rất cẩn thận, tỉ mỉ, luôn sẵn sàng lắng nghe, học hỏi từ mọi người và em học khá nhanh, dễ dàng thích nghi với công việc và môi trường mới. Ngoài ra em khá vui vẻ , dễ hòa đồng với mọi người.
* Nhược điểm của em là em nói hơi nhiều , ( ý này bạn có thể nói như vậy vì nó ko ảnh hưởng đến cv của em , mà tester nói hơi nhiều cũng ko sao cả ) hoặc có thể nói em bị bạn bè nói là hay săm soi lời người khác nói , hay nghĩ ra quá nhiều giả thiết, trường hợp cho 1 chuyện tưởng là đơn giản ( – bởi Tester thường hay săm soi bug, hay phải nghĩ ra tất cả các case có thể xảy ra nên bạn nói những ý này ra là nhược điểm của bạn thì khá là ok )

Câu hỏi 15: Nếu làm chung dự án với developer rất nóng nảy, bạn làm thế nào để dung hòa công việc với họ?

* Đầu tiên em sẽ cố gắng để làm quen, nói chuyện ngoài lề công việc với dev đó để xây dựng mối quan hệ tốt hơn
* Khi nói chuyện công việc , hoặc liên quan đến bug em sẽ vui vẻ, mềm mỏng với bạn dev này .
* Nếu bị gây khó dễ , vướng mắc, mâu thuẫn mà tự bản thân e ko thể giải quyết được với bạn dev này thì em sẽ raise lên test leader hoặc PM , dev leader ạ

Câu hỏi 16: Tại sao bạn lại chọn công ty này mà không phải công ty khác?

Em đã tìm hiểu qua internet/bạn bè và biết cty mình có môi trường làm việc khá là tốt và rất chuyên nghiệp , nhân viên có cơ hội phát triển bản thân. Nếu em có cơ hội được làm việc ở đây em sẽ cố gắng hết sức để làm việc,học hỏi và trau dồi thêm kiến thức để có thể cống hiến nhiều nhất có thể cho cty

.

Câu hỏi 17: Em có câu hỏi gì cho anh/chị không ? ( Nhà tuyển dụng hỏi ứng viên )

45

- Bạn có thể hỏi 1 vài thông tin mà bạn muốn biết, không nên bảo ko có câu hỏi gì. Vidu bạn có thể hỏi cty anh/chị có bao nhiêu tester ? Các dự án có dùng tool gì để test không ?

-Nếu bạn pvan Fresher thì bạn có thể hỏi là bên anh/chị đào tạo fresher trong bao lâu, lộ trình đào tạo ntn ….

Lưu ý: Trước khi đi pvan nên tìm hiểu sơ qua về cty mình sắp đến pvan

Câu hỏi 18 : Nghành Tester có điều gì thích thú? Mức lương hiện nay của Tester khoảng bao nhiêu?

Trả lời:

(KTPM) giúp bạn luôn không ngừng học hỏi, biết thêm kiến thức trong nhiều lĩnh vực khác nhau, bạn sẽ là những người tiên phong được trải nghiệm, được "phá" những sản phẩm mà chưa ai biết, thật tuyệt vời .

Lương là chỉ số đánh giá hiệu quả công việc của bạn trong Doanh nghiệp (DN). DN trả lương vì những gì bạn đóng góp, những gì bạn làm chứ không phải là những gì bạn biết. Do đó, mức lương của bạn là do bạn và tình hình DN. Mức lương sẽ được xác định cụ thể ngay khi bạn phỏng vấn tại DN bạn nhé.

Câu hỏi 19 : Bước chân vào nghề KTPM cần những kỹ năng.

Để làm việc tốt và gắn bó lâu dài với nghề, chúng ta cần chuẩn bị những gì và hiện thức nó như thế

nào?

Trả lời:

Bước chân vào nghề KTPM, bạn cần biết cách hoạt động của ứng dụng phần mềm, bạn đam mê cái mới, thích ""phá"" cho hư,… thì bạn hoàn toàn có thể gia nhập đội ngũ Tester.

Để làm việc tốt và gắn bó lâu dài, bạn cần liên tục học hỏi thêm kiến thức trên nhiều lĩnh vực, trao dồi thêm kỹ năng mềm, nâng cao khả năng tiếng Anh. Như vậy bạn nhất định sẽ thành công...

46

Câu hỏi 20 : Là một Tester cần biết ngôn ngữ Lập trình cơ bản nào? Sự hấp dẫn của công việc Kiểm thử phần mềm (KTPM) như thế nào? Đã làm các anh trung thành với công việc KTPM?

Trả lời:

Bạn nên chọn một ngôn ngữ lập trình chính cho mình, sau đó, khi gia nhập dự án Test, tùy theo yêu cầu, công ty sẽ có training thêm để bạn có thể làm tốt công việc của Tester. Công việc KTPM giúp bạn luôn không ngừng học hỏi, biết thêm kiến thức trong nhiều lĩnh vực khác nhau, bạn sẽ là những người tiên phong được trải nghiệm, được ""phá"" những sản phẩm mà chưa ai biết, thật tuyệt vời

Câu hỏi 21 : Vì sao phải kiểm tra bằng tay khi mình đã có kiểm thử tự động (KTTĐ) ?

Trả lời:

thường được dùng kết hợp với kiểm thử bằng tay để tận dụng hiệu quả nguồn tài nguyên (kiểm thử tự động có thể chạy 24h/ngày) và giả lập được các tình huống mà nếu sử dụng kiểm thử bằng tay sẽ tốn rất nhiều chi phí (VD: 1000 connection cùng 1 thời điểm,…).

Cả 2 phương pháp này đều cần thiết trong dự án và không thể nói phương pháp nào quan trọng hơn. Kiểm thử bằng tay có lợi thế là được sử dụng sớm trong dự án, tính tùy biến cao (bạn có thể chuyển hướng nếu cách đang đi ít có khả năng phát hiện lỗi ...) và là phương pháp chính để phát hiện lỗi. Kiểm thừ tự động chỉ được sử dụng hiệu quả khi phần mềm đã phần nào ổn định, không có nhiều thay đổi lớn và được sử dụng để test hồi quy. Hơn nữa chi phí đầu tư ban đầu cho kiểm thử tự động đôi khi rất lớn.

Câu hỏi 22 : Kiểm thử tự động có những khó khăn nào?

Có người nói Tester không cần phải biết code, có người nói code rất giỏi mới test được, như vậy ý kiến nào đúng?

Trả lời:

Kiểm thử tự động sẽ không có sự linh động, tùy biến như kiểm thử tay. Tester sẽ viết script để yêu cầu các bước test sẽ như thế nào, kết quả mong muốn là

47

gì, nếu có gì bất ngờ khác với những kết quả mà Tester ghi nhận thì máy không thể tự nhận xét và đưa ra

gợi ý như người làm Test manual được. Bạn không biết code nhưng bạn biết những đặc điểm chính của ngôn ngữ đó, cách ứng dụng được xây dựng và hoạt động trên môi trường ngôn ngữ đó là bạn có thể test được rồi. Thực tế là có khá nhiều bạn thành công với nghề Tester mặc dù là tốt nghiệp không chuyên CNTT (Kinh tế, XHNV, ...) Dĩ nhiên khi bạn code giỏi thì bạn sẽ am tường môi trường, bạn sẽ có thêm nhiều Test case chuyên sâu hơn nhưng quan trọng của Tester là ""mindset"" thử ""phá"" xem thế nào vẫn là quan trọng nhất nha bạn.

Câu hỏi 23 : Theo em biết, ở VN những Tester đa phần là lập trình không tốt nên mới theo ngành Kiểm thử, nhưng em thấy trong yêu cầu Tester phải viết được các tool automation thì làm sao có thể viết được?

Trả lời:

Đó là quan điểm hoàn toàn không chính xác và hiện nay thực tế đã chứng minh đó là một hướng đi khác, một lựa chọn khác của các bạn trong lĩnh vực CNTT. Yêu cầu của Tester không phải là phải viết được Tool Automation, mà yêu cầu của Tester là làm sao phải kiếm được Bug, giúp phần mềm chất lượng hơn, không làm mất uy tín - thương hiệu của nhà sản xuất. Nếu bạn là tester mà giỏi lập trình, bạn có thể đi theo hướng automation test và xây dựng framework cho test tự động. Nếu bạn không giỏi lập trình bạn vẫn có thể test bằng tay và sử dụng những framework để hỗ trợ việc test. Test tự động hay test bằng tay vẫn có sự phát triển riêng.

Câu hỏi 24: Khi test 1 phần mềm, làm sao để biết rằng test thế nào là đủ?

Trả lời:

Có nhiều dạng lỗi phần mềm (giao diện, chức năng, hiệu suất,…) nên trước khi bắt đầu 1 dự án Test, cả KH và công ty sẽ thống nhất mục tiêu của dự án Test là gì để tập trung vào các phần cần Test. Do đó, để biết test như thế nào là đủ phải xác định kết quả dự án Test mà KH mong muốn là gì mà có định lượng phù hợp.



48

Câu hỏi 25 : Yêu cầu cần và đủ của 1 Tester là gì?

Trả lời:

Yêu cầu cần và đủ của Tester là kiến thức, kinh nghiệm và lòng đam mê với nghề. Đây là hành trang để bạn có thể gia nhập đội ngũ Tester chuyên nghiệp.

Câu hỏi 26 : Developer và Tester không thể tồn tại nếu thiếu nhau, nhưng khi làm việc thì thường xuyên xảy ra xung đột. Anh/chị có thể cho em biết có quy trình hoặc phương pháp nào giúp Developer và Tester kết hợp làm việc hoàn hảo và tránh xung đột trong công việc?

Trả lời:

Thông thường, cách làm việc ổn thỏa nhất giữa Tester và Developer là "nói có sách - mách có chứng", tôn trọng nhau và vì lợi ích chung của dự án, của công ty, ai cũng bớt đi một chút cái "tôi" thì sẽ có tiếng nói chung.

Nếu tester có sự hòa đồng, thường xuyên giao lưu với mọi người trong team thì khi làm việc sẽ dễ dàng trao đổi, hỗ trợ nhau để cùng làm ra sản phẩm tốt, đem lại hài lòng cho KH.

### Lưu ý : Nếu bị hỏi đến cái gì mà mình không biết,thay vì ngồi im không nói gì ( nếu mà ngồi im ,mất bình tĩnh là dễ bị fail luôn ) thì em nên nhanh nhẹn đáp là : Em biết dự án công ty đang cần

**người có hiểu biết về (nó) ,nhưng nếu các anh chị cho e cơ hội được làm việc ở công ty mình ,thì em sẽ tự tìm hiểu và tự học để**

**khi đi làm có thể đáp ứng được yêu cầu của công cty , có thể làm được việc . Em sẽ cố gắng hết sức để không phụ sự kỳ vọng của anh chị đã tạo điều kiện cho em được vào làm ở cty ạ…**

Nhớ đọc kỹ câu 29-phần 1 nhé

Tài liệu lưu hành nội bộ của trung tâm TesterProVN – Hotline : 0973.703.357