TesterVN.com

Mounting passion, sharing experience

SECTION 12
LEVEL TEST CASES AND PRIORITY BUG

TesterVN
.com
Mounting passion, sharing experience

AGENDA

1. LEVEL TEST CASES
   a. INTRODUCTION
   b. HOW TO DIVIDE
2. PRIORITY BUG
   a. INTRODUCTION
   b. HOW TO SET PRIORITY

1. LEVEL TEST CASES

- INTRODUCTION
- HOW TO DIVIDE

1.1. INTRODUCTION

- Definition:
  - Gather test case to a group and set priority to run test case

- There are levels in which each test case will fall in order to avoid duplication efforts.
- Depend on the importance of test case.
- Need to set level of test case for Automation suit.

1.2. HOW TO SET

- Level 1:
  - Primarily consists of Basic Acceptance Testing (BAT) test cases
  - To represent a small subset of Must Run/Must Pass test cases that are ran on every software release to insure confidence in the release before beginning testing
  - Should be a quick functional test to verify the release is able to be tested
  - For example: verify ability to download/install, launch the application and perform major features, etc

1.2. HOW TO SET

- Level 2:
  - To validate the features of the release based on past/present requirements
  - To represent a limited set of primarily positive use cases/test cases to verify functionality and demonstrability of the features.
  - Should be ran in a limited environment (Leverage testing between Operating System, language, region, software, hardware, etc.)

1.2. HOW TO SET

- Level 3:
  - Consists of Release Candidate Acceptance test cases (100% of test cases must run to declare RC)
  - To validate more in-depth functionality of the release based on past/present requirements
  - Should primarily be positive test cases with some negative/boundary use cases
  - These test cases verify lower level functionality of specified features, general user interface, user experience, grammar/spelling/format, translations, etc

1.2. HOW TO SET

- Level 4:
  - Consists of negative/boundary use cases, break test cases, stressing the application, and ad-hoc
  - To verify more extreme scenarios that most common users will not experience
  - Should include low visibility and low usage scenarios
  - These are lower priority test cases and ideally should be ran at least once prior to GM

1.2. HOW TO SET

- Level 4:
  - Should leverage test results from previous test releases if tested on previous versions. Only retest for regressing test cases around areas of change
  - Examples: Full HDD, multiple HDD/ODD, manual manipulation of IDF/files/registry, looping the application to stress, uninstall/install, changing system date & time, language & region changes, etc

2. PRIORITY BUG

- Introduction
- How to set priority

2.1. INTRODUCTION

- *ISTQB Definition:*

priority: The level of (business) importance assigned to an item, e.g. defect.

- Must set priority
- To define the order in which we should resolve a defect.
- Sometimes, we need base on severity and exposure to set priority

2.1. INTRODUCTION

- Priority v.s Severity
  - Priority: defines the order in which we should resolve a bug
  - Severity: defines the impact that a given defect has on the system

1.2. HOW TO SET

- Depend on the importance or urgency of fixing a defect
- Normally, bug tracking has 4 priorities:
  - P1 – Critical/ Show Stopper/ Urgent
  - P2 – Major/ Must Fix/ High
  - P3 – Moderate/ Should Fix/ Medium
  - P4 – Low/ Optional Fix/ Low

2.2. HOW TO SET

- P1 – Critical/ Show Stopper/ Urgent
  - Types of issues
    - 1. Critical reliability/performance issue*
    - 2. Critical legal issue*
    - 3. Critical business issue*
    - 4. Critical security issue*
    - 5. Core functionality** is not working; no workaround
  - Must be fixed in the next build.

2.2. HOW TO SET

- P2 – Major/ Must Fix/ High
  - Types of issues
    - 1. Major reliability/performance problem*
    - 2. Major legal issue*
    - 3. Major business issue*
    - 4. Major security issue*
    - 5. Core functionality is not working; workaround exists
  - Must be fixed in any of the upcoming builds but should be included in the release.

2.2. HOW TO SET

- P3 – Moderate/ Should Fix/ Medium
  - Types of issues
    - 1. Unpleasant but bearable reliability/performance issue*
    - 2. Legal issue of moderate importance*
    - 3. Business issue of moderate importance*
    - 4. Security issue of moderate importance*
    - 5. Functionality doesn't work (cases that don't fall into P1, P2, P4 categories)

  - May be fixed after the release / in the next release

2.2. HOW TO SET

- P4 – Low/ Optional Fix/ Low
  - Types of issues
    - Cosmetic issue* of low importance that doesn’t and won’t affect reliability/performance, or legal, business, security sides of company operation.
    - Low importance/probability functional problems.

  - May or may not be fixed at all.

SUMMARY

- Have knowledge about level test case, why should we divide test case to level.
- Priority bug: definition and type

TesterVN.com

Mounting passion, sharing experience

Q&A

?