

애플리케이션 테스트 수행 part 2



결함 조치 관리

학습내용

- 결함 조치 관리
- 결함 유형 분류
- 결함 조치 우선순위 결정

학습목표

- 결함 조치 관리의 개념과 프로세스를 설명할 수 있다.
- 결함 유형을 분류할 수 있다.
- 결함 심각도 및 업무별 가중치와 우선순위를 결정할 수 있다.

결함 조치 관리

1 결함 조치 관리의 정의

결함 조치
관리

결함 발견에서부터 조치 완료될
때까지의 일련의 관리 활동

결함 발견
활동

애플리케이션 테스터들의 주요 활동



결함은 발견된 후에 여러 단계를 거치면서 조치가 됨



테스터는 상세 과정을 이해하고 있어야 제대로 된
애플리케이션 테스트를 수행할 수 있음

결함 조치 관리

1 결함 조치 관리의 정의

결함의
발견

조치
우선순위
결정

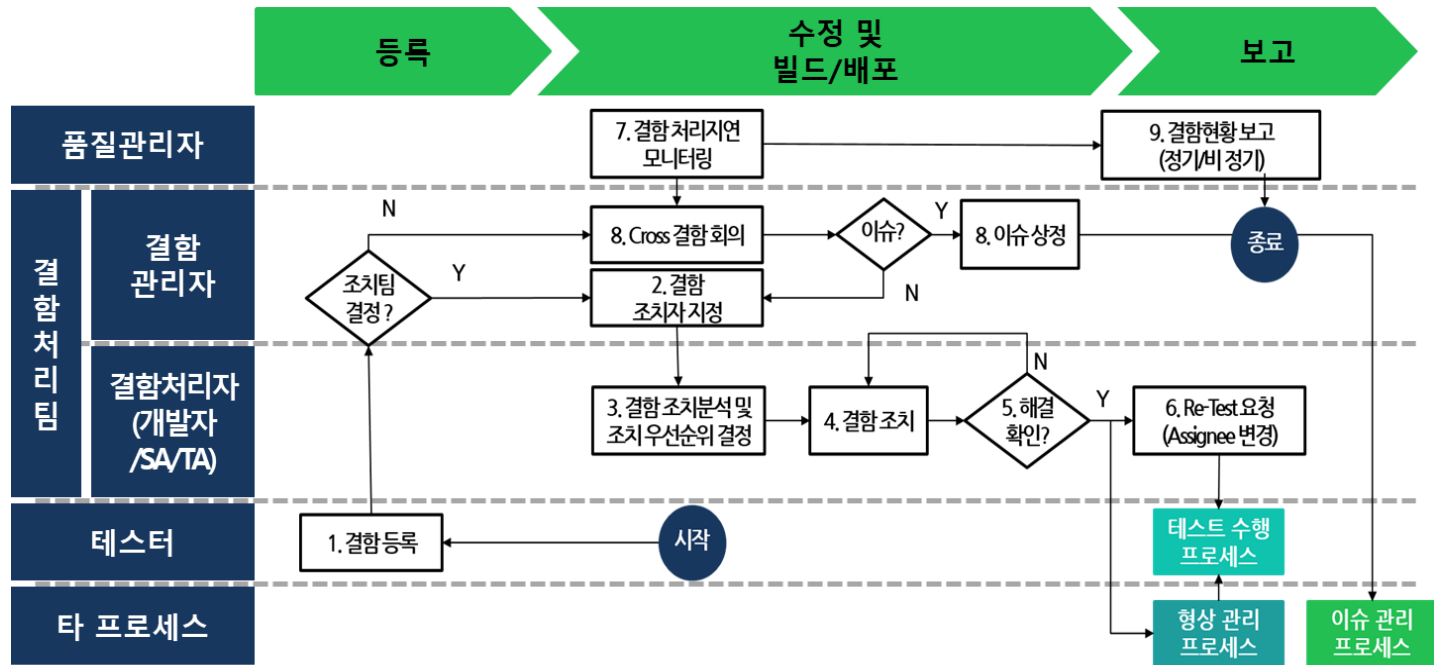
결함 분석
및 유형
분석

결함
담당자
배정

조치와
재테스트

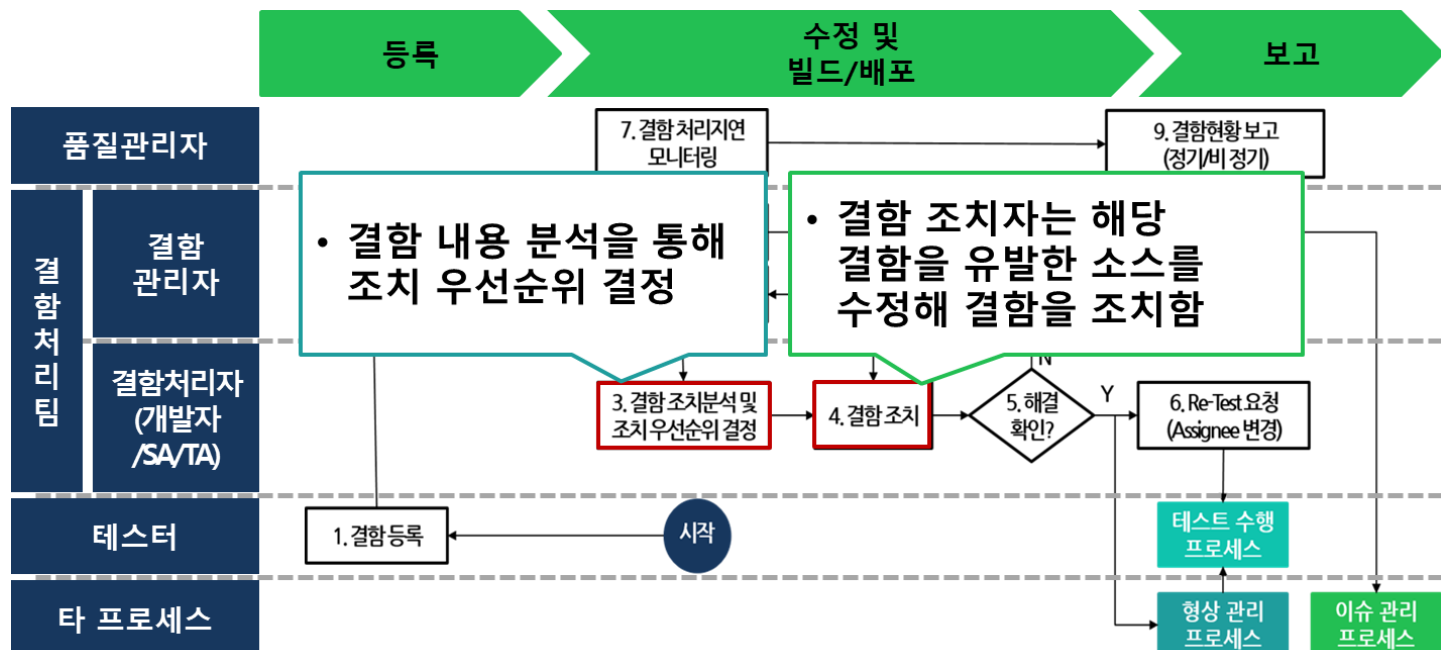
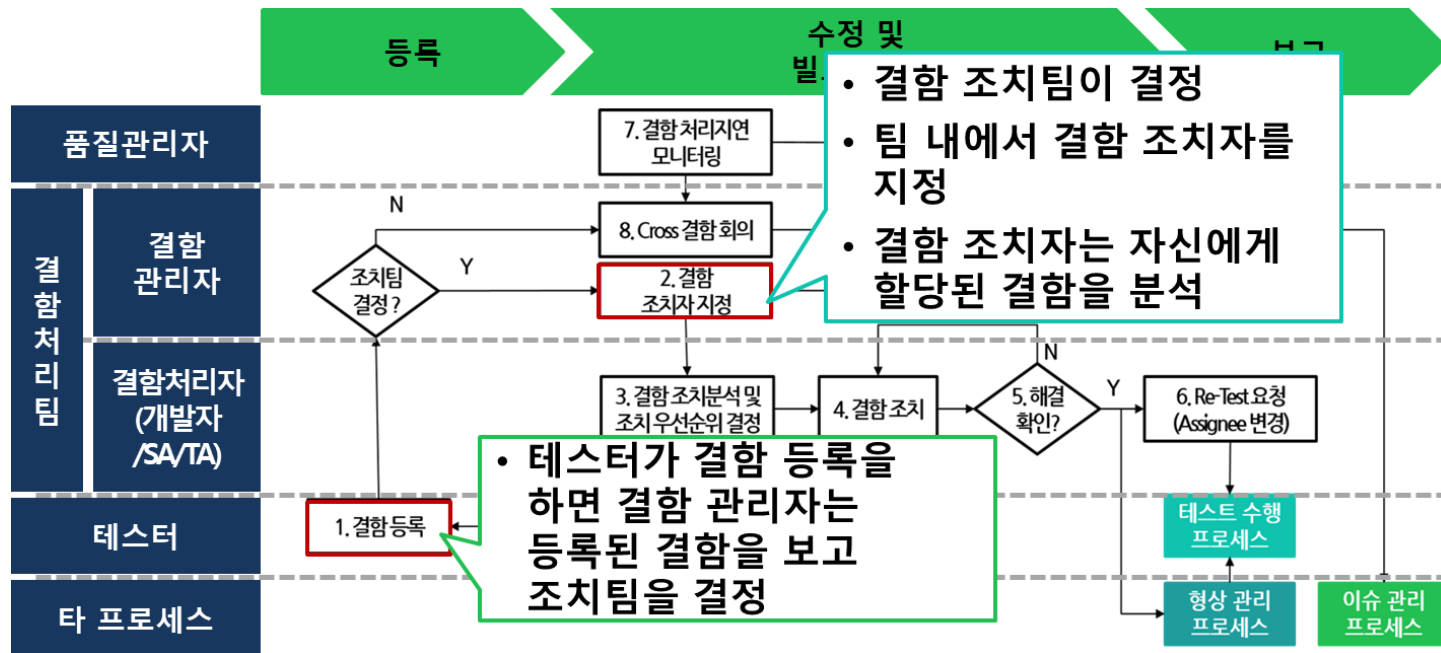
결함 조치 관리

2 결함 조치 관리 프로세스

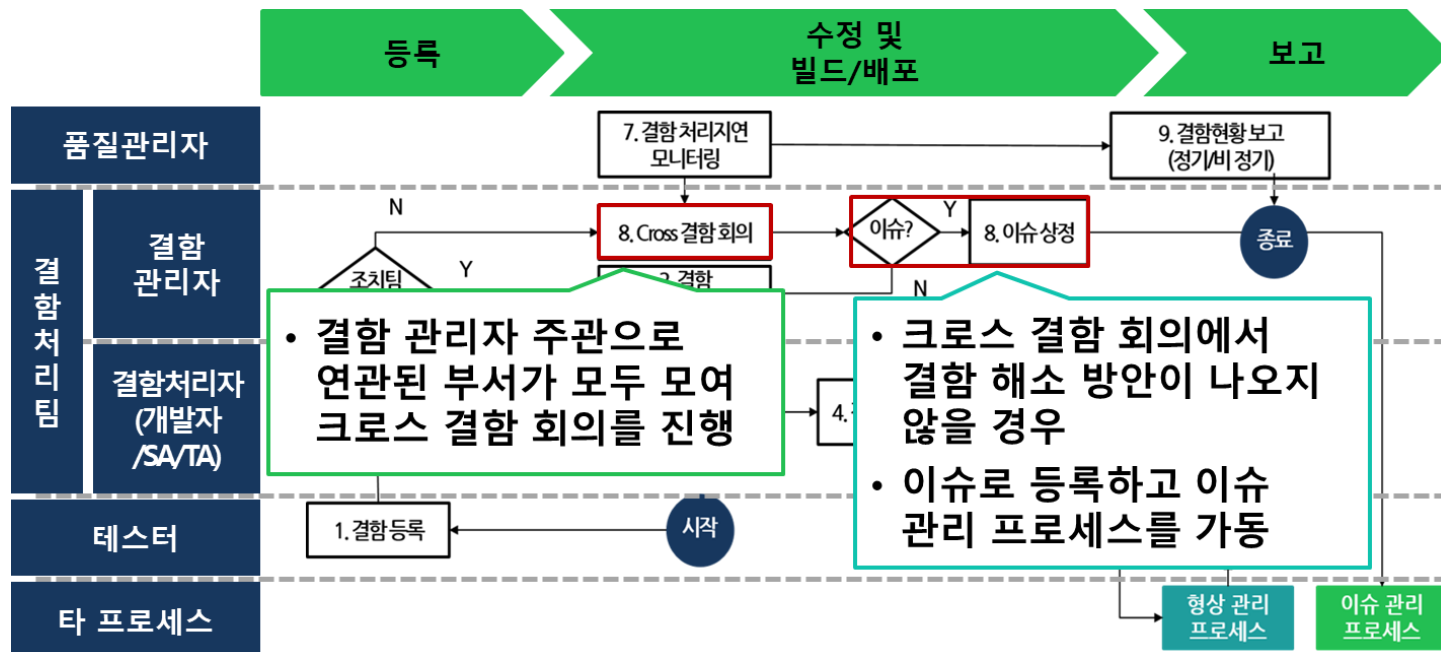


결함 조치 관리

2 결함 조치 관리 프로세스



2 결함 조치 관리 프로세스



결함 유형 분류

1 결함 유형



결함 조치자가 지정되면 결함 유형을 분류

코드
(Code)

데이터
(Data)

환경
(Environment)

패키지
(Package)

클라이언트
환경
(Client
Environment)

결함 아님
(Not
Defect)

결함 유형 분류

1 결함 유형

결함 유형	세부 내용	조치자
코드 (Code)	<ul style="list-style-type: none"> 개발자가 소스 코드 개발을 잘못하여 오류를 유발한 경우 UI, 서비스, 컴포넌트, Batch, Interface 등 소스 코드 결함 	애플리케이션 개발자
데이터 (Data)	<ul style="list-style-type: none"> 데이터베이스에 저장된 기준정보, 스키마 형상 불일치, 데이터 생성 오류 등으로 인하여 데이터 불일치 결함 기준정보, 스키마 형상 불일치, Output 데이터 생성 결함 등 	DBA (DataBase Administrator), DA(Data Architect)
환경 (Environment)	<ul style="list-style-type: none"> 서버, 네트워크, 시스템 소프트웨어 등의 설정이 잘못되어 오류가 나오는 결함 소스 코드에 결함은 없는데 테스트 환경을 구성하는 과정에서 운영 환경과 다르게 구성하여 결함이 발생하는 경우 DBMS, 하드웨어, 미들웨어, 네트워크, 시스템 소프트웨어 등 환경 결함 	SA (Software Architect), TA (Technical Architect)

결함 유형 분류

1 결함 유형

결함 유형	세부 내용	조치자
패키지 (Package)	<ul style="list-style-type: none"> • Oracle, SAP 등 패키지 소프트웨어의 오류로 인해 발생하는 결함으로 패키지 제공사에서 패치를 적용해 수정함 • Siebel, Oracle, SAP 등 패키지 결함 	3rd Party Vendor
클라이언트 환경 (Client Environment)	<ul style="list-style-type: none"> • 테스터의 PC 환경 설정이 잘못되어 발생하는 결함 • 특정 테스터 PC에서만 발생하는 결함으로 다른 PC에서는 보통 발생하지 않음 • Client의 PC 환경 결함 	SA (Software Architect)
결함 아님 (Not Defect)	<ul style="list-style-type: none"> • 테스터 입장에서는 결함이라고 등록을 하였으나 조치자가 분석해 보니 결함이 아닌 경우 • 프로세스가 변화하여 변화 관리 대상이라던가 정책이 수정되어 아직 테스터까지 공유되지 못한 경우 	

결함 유형 분류

2 결함 심각도(Criticality)



결함이 비즈니스 환경에 얼마나 영향을 미칠 수 있는가를 등급으로 매긴 것

1등급, Critical

- 조치되지 않으면 테스터가 더 이상 진행이 불가능한 경우(즉시 조치)
- 주문 데이터베이스 다운: 주문 자체가 되지 않기 때문에 더 이상 테스트 진행이 안됨

2등급, High

- 테스트 진행은 가능하지만, 핵심 기능에 오류가 나는 경우(1일 이내 조치)
- 주문은 가능하지만 빌링이 되지 않아서 청구서 발행이 되지 않는 경우
- 당장 테스트는 가능하나 빨리 조치 되지 않으면 심각한 장애가 발생할 수 있는 경우

결함 유형 분류

2 결함 심각도 (Criticality)



결함이 비즈니스 환경에 얼마나 영향을 미칠 수 있는가를 등급으로 매긴 것

3등급, Medium

- 핵심 기능은 정상이나 부가 기능이 결함인 경우
- 주문, 빌링 등 핵심 기능은 정상 동작 하지만 SMS발송 기능에 오류가 생긴 경우
- SMS 발송 오류는 테스트 진행에는 문제가 되지 않지만, 부가 기능 오류임

4등급, Low

- 기능 개선이 필요한 경우
- 결함은 아니지만 개선할 경우 사용자에게 편리한 경우

결함 조치 우선순위 결정

1 업무별 가중치 부여



업무별 가중치 부여는 결함의 조치 우선순위를 정하기 위해서 시스템별 등급을 부여한 것

구분	가중치	평가 등급				
		1	2	3	4	5
업무·서비스	30%	주문관리·고객관리·사용량 조회	빌링관리·공사관리·청구서 발송관리	유통관리	리스크관리	게시판관리
시스템 의존도	20%	대체 불가	대체 불가	대체 불가	수작업 업무 가능	수작업 가능
사용자 범위	30%	상담사·대리점	빌링팀·구축작업팀·영업관리팀	유통관리 팀	리스크팀	인사팀
월 이용자 수	20%	15,000명	2,000명	1,000명	100명	30명

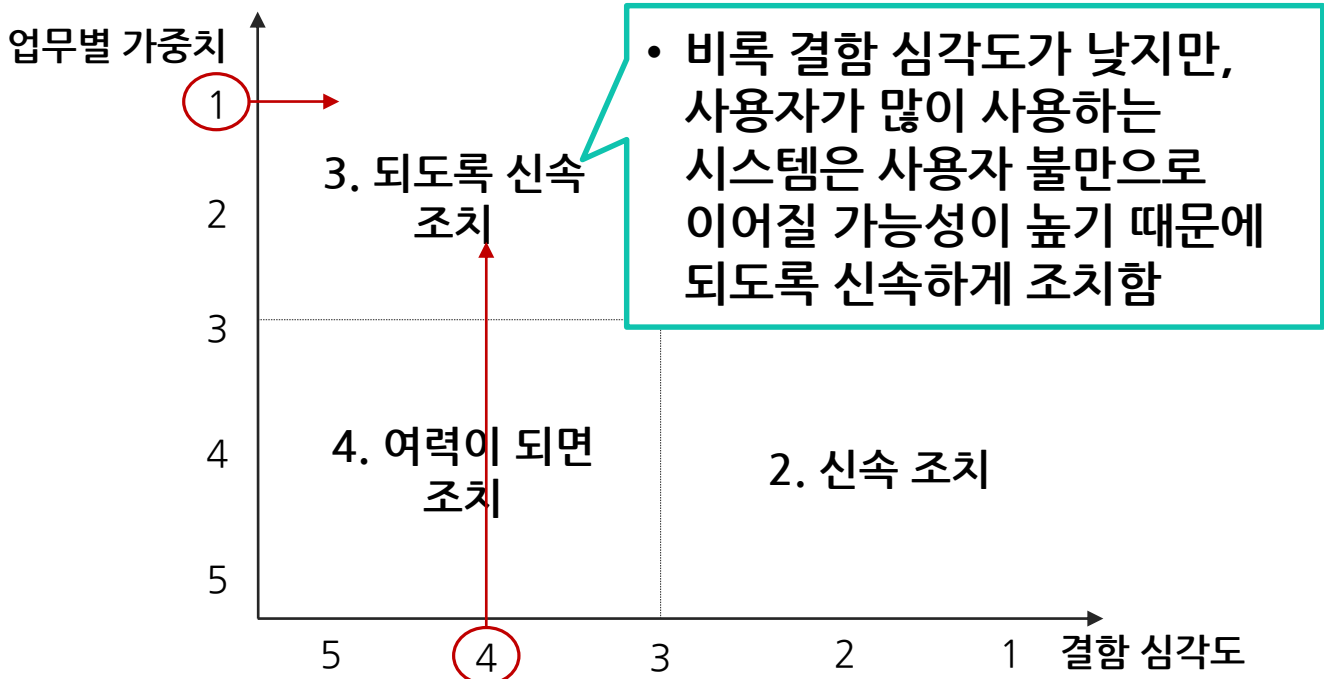
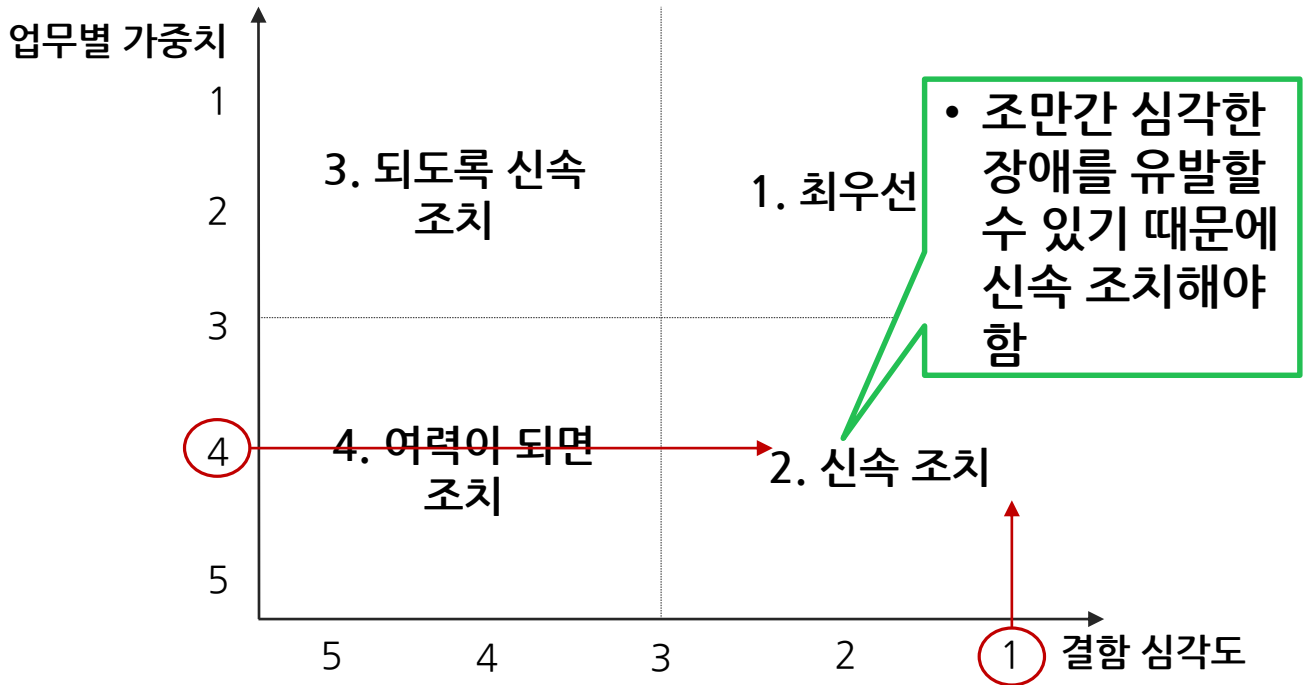
결함 조치 우선순위 결정

2 결함 조치 우선순위



결함 조치 우선순위 결정

2 결함 조치 우선순위



결함 조치 우선순위 결정

2 결함 조치 우선순위



결함 조치 우선순위 결정

2 결함 조치 우선순위

결함 심각도가 낮지만, 업무별 가중치가 높은 결함을 먼저 조치

결함 심각도가 높고, 업무별 가중치가 낮은 제품을 나중에 조치

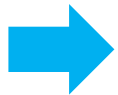
프로젝트의 상황에 따라서 우선순위를 바꿔서 결함 조치를 할 수 있음

결함 조치 우선순위 결정

3 결함 조치 일정 설정



가장 우선순위가 높은 결함을 먼저 조치하고,
우선순위가 낮은 결함은 나중에 조치



자연스럽게 조치 기한 결정



개발자들은 조치 기한 내에 조치해 재테스트를 요청

결함 ID	결함명	결함 유형	결함 설명	우선 순위	조치자	테스터	조치 기한
IT_FRM_01	다국어 번역 오류	코드	다국어 번역에 오류가 있음 (한국 → 영어)	1	개발자1	테스터1	'18.5.10
IT_FRM_02	금액 불일치	데이터	매입 금액의 합계와 밑에 있는 셀의 합계 금액이 불일치함	2	개발자2	테스터2	'18.5.12
IT_FRM_03	보고서 안 보임	환경	정산서 출력 시 Report Designer 화면이 보이지 않음	3	개발자3	테스터3	'18.5.15
IT_FRM_04	ActiveX 설치오류	클라이 언트 환경	테스터 PC에 ActiveX가 설치되지 않아서 세금계산서 발행 안 됨	4	개발자4	테스터4	'18.5.19

결함 조치 우선순위 결정

4 결함의 가이드라인

① 환경 결함의 가이드라인

컴파일, 테스트, 다른 지원 시스템 문제

- 테스트 과정에서 만들어진 결함
- 예 : 테스트 도구 결함, 테스트 스크립트 오류, 버그가 아닌 것을 버그로 간주

Configuration, 타이밍, 메모리 오류

- 서버가 다운된 경우
- 프린터, 관련 장비 등 디바이스의 연결 오류
- 네트워크 오류가 발생한 경우

결함 조치 우선순위 결정

4 결함의 가이드라인

② 개발 표준 결함의 가이드라인

용어사전,
응용프로그램
코딩가이드,
성능가이드,
응용프로그램
명명가이드 등

- StringBuffer를 String으로 선언한 경우 (for문 안에서만 적용됨, 성능 가이드 참조)
- 클래스 변수(Class Member Variable)를 설명하는 주석문 미작성(응용프로그램 코딩 가이드 참조)
- String 변수명의 접두어가 str로 시작되지 않은 경우(응용프로그램 명명 가이드 참조)

철자 오류,
format,
스펠링, 구두점
등

- 윈도우 제목, 화면 표시 항목 등에 철자 오류가 있을 경우

버튼,
레이어명,
메시지, UI
등이 표준
미준수

- UI스타일 가이드 표준을 준수하지 않았거나 화면 간 일관성이 결여된 경우
- 기능 수행 시 표시되는 오류 메시지의 표시 형식이나 내용이 다른 화면의 그것과 다른 경우
- 기능 표준을 준수하지 않았거나 동일 기능이 화면 간에 일관성 있게 구현되지 않은 경우

결함 조치 우선순위 결정

4 결함의 가이드라인

③ 프로그램 일관성 결함의 가이드라인

가이드라인	결함 내용
구현되지 않은 누락된 기능	<ul style="list-style-type: none"> • 입력 필드에 대한 Validation이 누락된 경우 • 반드시 필요한 기능 버튼이 누락되어 있는 경우 • 기능 버튼을 클릭했으나 아무 Action이 없는 경우
모듈구조가 설계 구조와 미일치(코딩 문제)	<ul style="list-style-type: none"> • 설계 로직은 $A+B=C$로 되어 있으나, 구현 로직은 $A-B=C$가 된 경우
분석, 설계 산출물 (Input 산출물) 오류로 인한 코딩 에러	<ul style="list-style-type: none"> • 컴포넌트 설계서에 명시된 패키지 오류로 코딩 오류 발생 • 프로그램 설계서에서 Default 값이 잘못 설계된 경우

결함 조치 우선순위 결정

4 결함의 가이드라인

④ 데이터 결함의 가이드라인

부적절한 데이터 변수 선언, 정의, 초기화, 중복

- 변수 자릿수를 30자로 선언해야 하는데 10자리로 선언한 경우
- 변수의 default 값을 true로 설정해야 하는데 false로 한 경우

시험 데이터의 부적절로 인한 오류

- 시험 데이터 미준비
- A 업무 테스트 시 시험 데이터가 B 업무에 대한 데이터가 준비된 경우

SQL을 사용한 DB 질의·갱신 오류

- 조회되는 결과에 정렬 기준이 없어 데이터가 랜덤하게 표시되는 경우
- 입력, 수정, 삭제하였으나, DB 데이터에 입력, 수정, 삭제가 안 되어 있는 경우

연계 화면 간의 데이터 전달 오류

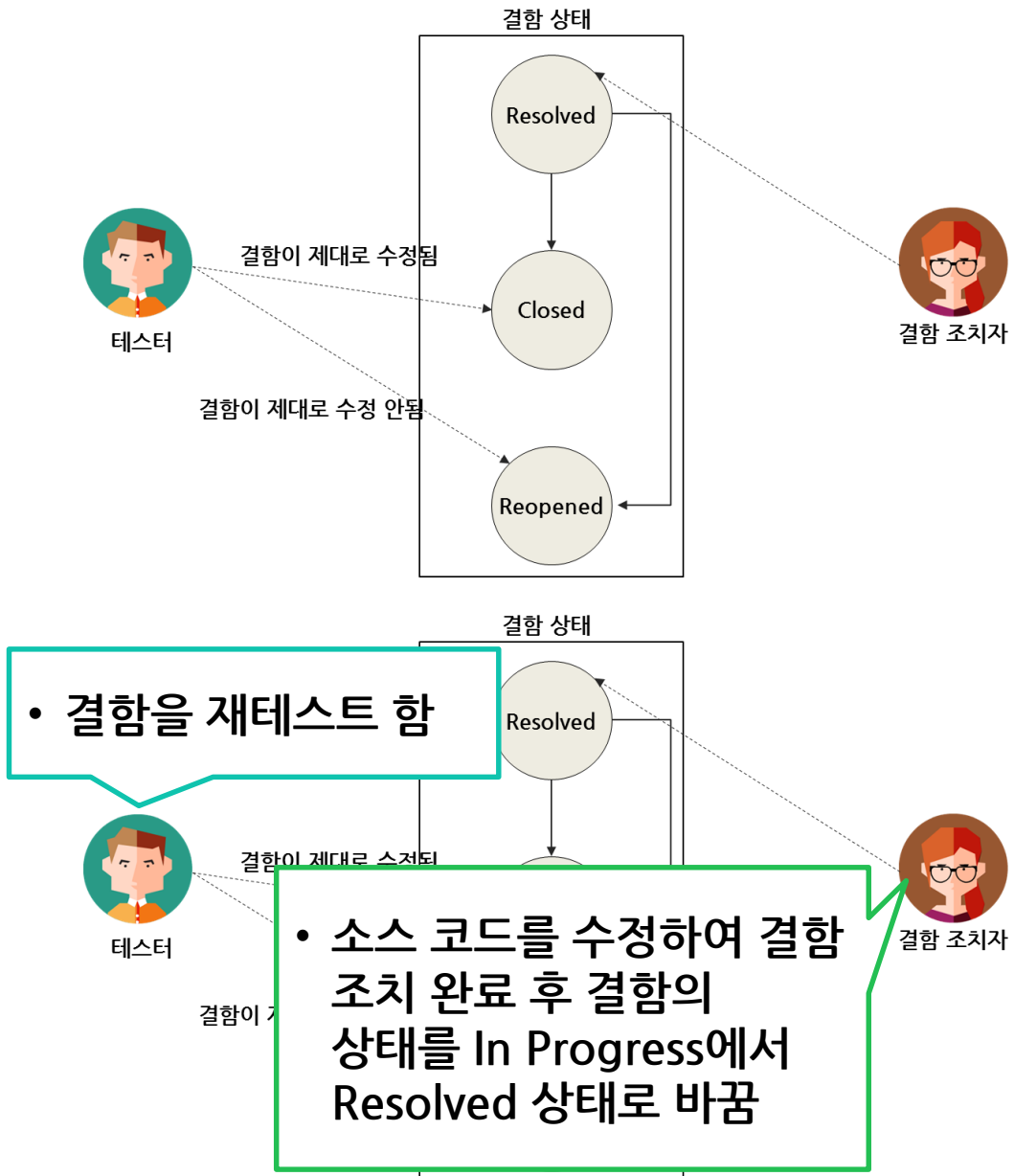
- 타 업무 화면으로 데이터를 전달했으나, 해당 화면에 데이터가 제대로 전달되지 않은 경우

인스펙션의 정의 및 개요

5 결함 재테스트



결함 조치자가 소스 코드를 수정하여 결함 조치를 완료하면 **테스터는 결함을 재테스트**해야 함



인스펙션의 정의 및 개요

5 결함 재테스트



결함 조치자가 소스 코드를 수정하여 결함 조치를 완료하면 **테스터는 결함을 재테스트**해야 함

- 결함이 제대로 수정되었으면 결함의 상태를 Closed로 바꾸고 종료



테스터

결함이 제대로 수정됨

Closed

결함이 제대로 수정 안됨

Reopened

- 결함이 제대로 수정되지 않았으면 결함의 상태를 Reopened로 바꾸고 결함 조치자에게 다시 조치해 줄 것을 요청함

01. 결함 조치 관리

- 결함 조치 관리 정의
 - ✓ 결함 발견에서부터 조치 완료될 때까지의 일련의 과정을 관리하는 활동
 - ✓ 결함 등록, 결함 분석 및 유형 분류, 결함 조치 우선순위 결정, 결함 담당자 배정, 결함 조치, 재테스트

02. 결함 유형 분류

- 결함 유형 분류
 - ✓ 코드성 결함(Code) : 개발자가 소스 코드 개발을 잘못하여 오류를 유발한 경우
 - ✓ 데이터 결함(Data) : 데이터베이스에 저장된 기준정보, 스키마 형상 불일치, 데이터 생성 오류 등으로 인하여 데이터 불일치 결함
 - ✓ 환경 결함(Environment) : 서버, 네트워크, 시스템 소프트웨어 등의 설정이 잘못되어 오류가 나오는 결함
 - ✓ 패키지 결함(Package) : Oracle, SAP 등 패키지 소프트웨어의 오류로 인하여 발생하는 결함으로 패키지 제공사에서 패치를 적용하여 수정함
 - ✓ 클라이언트 환경 결함(Client Environment) : 특정 테스터 PC에서만 발생하는 결함으로 다른 PC에서는 보통 발생하지 않음

02. 결함 유형 분류

- 결함 심각도 분류
 1. Critical : 조치되지 않으면 테스트가 더 이상 진행이 불가능한 경우(즉시 조치)
 2. High : 테스트 진행은 가능하나 핵심 기능 결함인 경우(1일 이내 조치)
 3. Medium : 핵심 기능은 정상이나 부가 기능 결함인 경우(2일 이내 조치)
 4. Low : 기능 개선이 필요한 경우(3일 이내 조치)

03. 결함 조치 우선순위 결정

- 업무별 가중치
 - ✓ 사용자 수가 많고, 시스템 의존도가 높은 시스템에 업무별 가중치를 높여 줌
- 결함 조치 우선순위
 - ✓ 결함 심각도가 높고, 업무별 가중치가 높은 결함은 최우선 조치 함
- 결함 가이드라인
 - ✓ 환경 결함의 가이드라인, 개발 표준 결함의 가이드라인, 프로그램 일관성 결함의 가이드라인, 데이터 결함의 가이드라인
- 결함 재테스트
 - ✓ 결함 조치자가 소스 코드를 수정하여 결함 조치가 완료되면 테스터는 결함을 재테스트해야 함