Software Engineering Q/A Sheet (#10)

date:11-20 number: 2018312280 name: 이상수

Questions from Prof.

1. Describe the two goals of software testing. And explain what the successful test that satisfies each goal is.

software testing은 프로그램이 의도대로 작동하는지, 프로그램의 문제점은 무엇인지 찾는 것이 목표이다.

첫번째 골은 validation testing으로 테스트한다.

이 테스트의 성공은 시스템이 의도대로 동작한다는 것을 보여준다.

두번째 골은 defect testing으로 테스트한다.

시스템이 incorrect하게 동작하게 의도하여 이 defect를 노출시키는 것이 성공적인 테스트이다.

1. Compare three development tests: Unit, Component, and System testing.

Development test는 개발자들이 테스트하는 과정으로

Unit testing : Defect test로서 역할을 하며 개별 class수준의 object나 function을 테스트하는 테스팅으로 확대시켜서 composite component도 unit으로 볼 수도 있다.

Component testing : 여러 unit이 모여 component를 이룰 때 이 component의 interface를 테스트하는 테스트로 오브젝트의 동작, 각 오브젝트들의 attributes의 값, 그리고 오브젝트들의 다양한 상황에서의 상태를 테스트한다. 이 때 상속관계를 활용한 오브젝트들은 그 상속 관계의 모든 오브젝트를 테스트해야 하기에 테스트에 어려움을 가져올 수 있다.

System testing : components들이 모여서 시스템 전체적으로 테스트하여 컴포넌트 간에 상호작용을 테스트하는 테스트

1. Describe partition testing and create an example in your term project.

Partition은 비슷한 출력이 만들어지는 data끼리의 그룹이다.

이러한 partition들 별로 대표 data를 뽑아서 테스트하여 총 테스트에 드는 비용과 시간을 줄이는 방법이다.

이를 위해서는 우선 input equivalence partitions와 output equivalence partitions를 나눠야하고, 이중 경계에 있거나 partition의 중간쯤에 있는 값을 일부분 선택하여 테스트한다.

팀프로젝트 예시 :

Input partition : 정상코드, 실행 불가 코드

Output partition : 정상 수행 시 탄소배출량 출력, 비정상 수행시 에러 출력값을 출력

1. Explain the four interface types in the component test and explain the interface errors associated with each type of interface.

Parameter interfaces : 데이터를 오브젝트 간에 주고받는 경우

Procedural interfaces : set of procedures를 서브시스템 간에 주고받는 경우

Message passing interfaces : 클라이언트-서버 처럼 서브 시스템 간에 서비스를 요청하는 경우

Shared memory interfaces : 메모리 블록이 함수나 작동에 공유되는 경우

1. Explain the benefits of test-driven development.

Code coverage : 코드를 작성하였을 때 모든 코드가 필수적인 코드가 될 확률이 높아진다.

Regression testing : 자동화된 테스트코드로 incremental development에 이전 코드들을 매번 따로 테스트해주지 않아도 된다.

Simplified debugging : 테스트가 실패하면 새로 작성한 코드가 문제가 있을 확률이 높아지고 이를 테스트를 통해 추적하기도 쉬워 디버깅이 쉬워진다.

System documentation : 테스트를 작성해두는 만큼 어느 부분이 테스트가 필요한지를 판단하는 것이기에 이는 문서 작성에도 도움이 된다.

1. Practice requirements-based testing with part of your team project like the example on slide 53, 54.

유저가 코드를 입력하면 이를 서버에서 실행시켜야한다.

-자바코드를 입력해보고 이를 서버에서 실행시켜서 결과가 출력 되는지 확인

유저가 잘못된 코드를 입력하면 에러와 console결과가 유저에게 출력되어야한다.

-자바코드를 오류가 있는 코드를 입력해보고 이 결과가 잘 출력 되는지 확인한다.

1. Compare three user tests: Alpha, Beta, and Acceptance tests.

Alpha : generic한 시스템에 테스트. 유저가 개발자 사이트에 와서 아이디어도 주고 하며 테스트의 한 팀으로써 작용하는 테스트

Beta : generic한 시스템의 테스트. 불특정 다수에게 테스트를 받고 피드백을 받는 다양한 환경과 유저에게서 일어날 에러를 잡는 테스트.

Acceptance test : custom한 시스템에 테스트. 요구사항에 맞는 고객과의 합의된 내용들을 테스트하는 테스트.

Questions from you

베타 테스트에 잘못된 피드백을 받으면 어떻게 해야하나?

해당 유저에게 제재를 가하는 등 경고를 주어 발생하지 않게 유도한다.