

2013. 9. 2. [제65호]

# 사용자 경험(UX)과 품질 관리(QA)에 따른 사용성 확보

소프트웨어공학센터 경영지원TF팀

## C o n t e n t s

- ▶ 서론
- ▶ 애자일 개발의 네 가지 단계별 고려사항
- ▶ 결론

**Key Message**

SW 사용성(usability)은 SW 사용자 입장에서 원하는 기능과 서비스를 손쉽게 찾아서 활용할 수 있도록 기능을 제공하는지를 의미함. 최근 들어 SW 사용성은 SW 가치의 중심축 역할을 함에 따라 애자일 개발에서는 전체 개발팀이 제품의 사용성을 함께 고려하는 것이 중요함. 따라서 여기서는 사용자 경험(UX)과 품질 관리(QA)측면에서 사용성 확보를 위한 기획-개발-테스팅-배포 등의 단계별 고려사항을 제시함.

▶ **서론**

- 사용성<sup>1)</sup>의 원칙은 망치와 같은 물리적 품목들에서 소프트웨어 애플리케이션에 이르기까지 모든 유형에 적용되는데, 소프트웨어의 사용성은 SW가치의 중심 역할을 함
- 애자일 소프트웨어 개발측면에서 제품의 사용성은 팀 전체의 공유되는 관심사이자 만, 본 보고서에서는 기획, 개발, 테스트 및 배포라는 애자일 개발의 네 가지 단계를 통해 품질 관리와 사용자 경험<sup>2)</sup>이 사용성을 확보하는 방법에 초점을 맞추고자 함

▶ **애자일 개발의 네 가지 단계별 고려사항**

**1. 기획(Research)**

- 비즈니스 분석가와 설계자는 일반적으로 애자일 개발의 기획 단계를 주도하지만, 기획 단계는 품질 관리를 위해서도 중요한 요소임
- 사양(Specification)의 기본 개념은 기획단계에서 모든 관점을 포함하고 있는 요구 사항에 대한 결과물로, 일반적으로 사례 주도적 개발, 실행 요구 사항, 수용 테스트 주도적 개발, 신속한 수용성 테스트 등의 방법론들의 결과물도 모두 포함하고 있음
- 프로젝트 팀원이 공동으로 모든 테스트를 실행하고 동시에 문서를 생성할 수 있도록 요구 사항을 생성하고 유지하는 방식으로 팀은 의사소통의 오류 및 요구 사항과 디자인에 대한 혼란을 피할 수 있음

1) 사용성 : 사용자가 목표를 완수할 수 있도록 어떤 도구를 사용하는 데에 있어 용이함을 말함  
2) 사용자경험 : 사용자가 시스템의 상호작용을 인식하는 것을 말함

## 2. 개발(Development)

- 개발 단계는 다양한 활동을 포함하는데, 사용자 인터페이스를 설계하는 과정은 사용자 경험과 품질 관리 모두에 있어서 매우 중요함
  - 개발기간 동안 사용자 경험(UX)은 개발 중인 애플리케이션에 대한 표준 정보를 제공하는 스타일 가이드(style guide)를 작성하고 수정해야 하는데, 스타일 가이드는 색상에서부터 디자인 패턴에 이르기까지 다양한 범위를 포함할 수 있음
  - 애플리케이션이 사용자에게 어떻게 인식되고 작동하게 되는지에 대한 이해를 공유하기 위해 사용자 경험과 품질관리 업무가 협업하는 것은 좋은 기회라 할 수 있음
- 프로젝트 개발팀이 애플리케이션을 지속적으로 개발하고 사용자 스토리를 테스트함으로써, 사용자 경험과 비즈니스에 의해 정의되어 변경되지 않는 사용자 인터페이스의 측면들을 식별할 수 있음
  - 품질 관리 측면에서 변경되지 않는 부분이 빨리 확정되면 될 수록 품질관리자들은 모든 테스트주기 동안 반복해야 하고, 적용 가능한 표준들을 작성해야만 하는 회귀테스트(regression test)<sup>3)</sup>를 더 빨리 수립할 수 있음

## 3. 테스트(Testing)

- 애자일 개발주기 상 테스트 단계에서 품질 관리는 단위 테스트, 회귀 테스트, 스모크 테스트(Smoke Test)<sup>4)</sup> 및 탐색적 테스트(exploratory test)<sup>5)</sup>를 수행하는 동안 나머지 팀 구성원은 다른 단계에 관련된 활동에 초점을 맞춤
  - 따라서 사용자 경험이 구성 요소 A를 기획하고 설계한 뒤에 개발은 그것을 코드화 하고, 최종적으로 품질 관리는 일련의 테스트를 통해 실행함
- 품질 관리팀이 테스트를 하는 동안, 개발팀은 구성 요소 B를 개발하는 작업을 하기 때문에, 사용자 경험팀은 구성요소 B를 기획하고 설계하는 것을 이미 완성하고 구성요소 C를 기획하고 설계하는 단계로 이동해야 함
- 품질 관리팀에서 구성요소 A를 테스트하고 있고 사용자 경험팀에서 구성요소 C를

3) 회귀테스트 : 오류를 제거하거나 수정한 시스템이나 시스템 컴포넌트 또는 프로그램이 오류 제거와 수정에 의해 새로이 유입된 오류가 없는지를 확인하는 일종의 반복 시험  
 4) 스모크테스트 : 본격적인 테스트의 수행에 앞서, 시스템, 컴포넌트, 소프트웨어 프로그램 등 테스트 대상이나 제품의 빌드(제품 설치 패키지)가 구축된 테스트 환경에서 테스트가 가능한지 여부를 판단하기 위해 주요 모듈이나 시스템을 간단하게 테스트 하는 것  
 5) 탐색적 테스트 : 예러 예측과 비슷하게 경험에 기반 하여 테스트를 수행하지만, 작성된 테스트 케이스 문서를 보면서 테스트를 수행하는 것이 아니라 테스트 케이스를 만들면서 테스트를 수행함

기획 개발하고 있는 경우, 애플리케이션의 구성요소들이 서로 연관될 수 있는 확률이 높아지기 때문에 다른 영역들이 함께 작업하는 기회가 증가됨

#### 4. 배포(Release)

- 프로젝트 개발팀이 코드를 기획, 개발, 테스트하고 나면 소프트웨어를 배포하게 되는데, 배포 단계에서 매우 중요한 것은 분석을 실행하고 추적하는 것임
  - 그 결과로 비즈니스 통찰력을 기르고, 사용자 행동의 패턴을 식별하며, 시스템 내 버그들을 잡을 수 있음
- 베타버전을 배포할 때, 이러한 분석은 제품 기획의도를 검증하고, 제품이 시장에 공개되기 전에 큰 실수를 하는 것을 방지할 것임
  - 시장에 제품을 배포할 때, 고객들과 신뢰를 재구축하기 위해 이러한 분석을 사용할 수 있고, 사용하는 동안 발생할 수 있는 실수나 오류를 줄일 수 있음
  - 품질 관리가 버그를 잡기 위해 사용할 수 있는 반면, 사용자 경험은 사용자 행동을 평가하기 위해 이용할 수 있음
- 팀원들은 제품이 타겟으로 하는 핵심 사용자 행동 패턴을 파악할 수 있고, 버그를 확인하거나 사용자가 직면할 수 있는 워크플로를 방해하는 요소가 없는지 확인할 수 있음
  - 팀원들이 어떤 버그를 확인한다면, 그들은 처음부터 시작해서 자신의 디자인을 재평가하고, 실제로 사용 가능한 제품을 만들기 위해 필요한 모든 변경 작업들을 수행할 수 있음

#### ▶ 결론

- 제품 개발 과정 전반에 걸쳐 사례에 따른 사양, 스타일 가이드 제작, 사용성 테스트, 분석 등 사용자 경험과 품질 관리가 협력하여 작업할 수 있는 다양한 업무들이 있음
- 품질 관리는 사용자 경험에 영향을 주기 때문에 사용자 경험과 품질 관리 기능을 같이 고려하는 것은 개발초기에서부터 종종 발생하는 업무 간(또는 프로세스 간) 이해의 차이를 줄이고 실제로 사용 가능한 제품을 제작할 수 있도록 도움을 줄 수 있음

### 참고 자료

1. <http://www.nngroup.com/articles/quality-assurance-ux/>
  2. <http://www.agilemodeling.com/essays/agileUsability.htm>
  3. <http://www.uxmatters.com/mt/archives/2013/07/user-experience-quality-assurance-usability.php>
-