

패키지게임 S/W 품질모델 및 시험기법에 관한 연구

김훈, 최종원
(재)한국게임산업개발원 게임연구소

e-mail : nuhmik@gameinfinity.or.kr

A Study on Package Game S/W Quality Model and Testing Method

Hun Kim, Jong -Won Choi
Game Technology Lab, Korea Game Development & Promotion Institute

요 약

세계 게임시장의 급속한 팽창으로 인하여 게임 S/W 의 경쟁력 확보를 위한 여러 방안들이 제기되고 있으며 이중 품질에 대한 논의가 대두되고 있다. 그러나 게임 S/W 의 품질에 관한 국내의 연구 • 개발이 아직 미흡한 실정이며 게임 S/W 의 품질저하로 이어지고 있다. 따라서 게임산업의 발전과 아울러 패키지게임 S/W 의 품질 향상을 통해 게임 S/W 의 질적 수준을 제고할 필요가 있다. 본 연구에서는 ISO/IEC 9126 를 기반으로 패키지게임 S/W 를 시험하여 결과를 산출할 수 있는 품질모델과 시험기법을 개발하여 패키지게임 S/W 품질평가에 적용할 수 있도록 하였다.

1. 연구배경

최근 게임의 인기가 급상승하고 있으며, 그 사회적 파급효과 역시 나날이 커지고 있다. 2002 년도를 기준으로 보면 국내 게임산업의 시장규모는 온라인게임, 모바일게임, 비디오게임, 패키지게임, 아케이드게임, PC 방, 게임장, 복합유통업소 등을 모두 포함하여 총 3 조 4,026 억원에 이르는 것으로 파악되고 있으며 국내 총생산(GDP)에서 규모가 596 조 381 억원(한국은행 집계)이므로 게임산업의 GDP 대비 비중은 0.57%에 해당하는 것으로 추정할 수 있다[1].

또한 게임 S/W 가 점점 복잡하고 다양한 기능을 수행하고, 용량이 증가하고 있다. 이에 게임 S/W 를 개발하고 있는 수많은 개발사들은 여러 가지 상황에 쫓기고 있으며 수많은 경쟁사와의 무한 경쟁이 치열해지면서 게임개발사들은 더욱 신속하게 보다 높은 품질의 게임을 한정된 기간과 예산만을 이용하여 개발해야 하는 어려움에 처해 있다. 때때로 개발사들은 출시기한을 맞추기 위해 게임 S/W 의 품질을 충분히 확보하지 못하거나, 추가하기로 하였던 기능을 미처 추가

하지 못하고 제품을 출시하기도 한다. 이렇게 게임 S/W 의 품질을 충분히 확보하지 못한 경우 실제 구동될 때 많은 문제점을 일으킬 수 있다.

이처럼 게임 S/W 의 품질 문제가 업계의 심각한 공통 과제로 대두되는 상황에서 게임의 품질을 확보하고, 소프트웨어 개발 기간 및 소요 예산을 계획대로 추진하기 위해서는 소프트웨어 시험을 정확하고 계획적인 방법으로 철저하게 추진하여야 한다. 출시될 게임을 개발기간 전반에 걸쳐 엄밀한 시험을 하고, 모든 측면의 품질평가를 주먹구구식이 아닌 국제적 표준에 맞추어 수행함으로써, 게임개발사가 게임 출시 후 오류로 인한 엄청난 피해를 예방할 수 있다.

2. 관련연구

2.1 게임 S/W 의 특징

게임 S/W 는 일반적인 소프트웨어가 갖는 특성 외에도 ‘놀이’, ‘재미’ 라는 영화, 애니메이션 등과 같은 콘텐츠적 기능을 구현한 소프트웨어이기 때문에 게임 S/W 만의 많은 고유한 특성을 지닌다[2].

① 상대적으로 고사양 시스템을 요구한다.

최근의 게임 S/W 는 콘텐츠의 꽃이라고 불릴 만큼 화려한 멀티미디어 테크닉을 총동원하여 제작되다 보니 사용자들의 만족도는 향상되었으나, 반면에 요구되는 시스템의 사양은 상대적으로 매우 높다, 그렇기 때문에 게임 S/W 는 시스템의 성능에 따라서 성능이 크게 좌우된다.

② 직접적인 하드웨어 접근이 빈번하다.

게임 S/W 는 일반적으로 실시간으로 사용자 입력을 요구한다. 많은 게임의 경우 적절한 시간에 입력되지 않으면 게임이 종료되는 등 사용자는 불이익을 받는다.

③ 정보를 은닉한다.

게임 S/W 는 모든 정보를 바로 접근할 수 없도록 제작한다. 게임 S/W 의 목적은 기본적으로 ‘놀이’ 이기 때문에 사용자에게 궁금증을 유발하도록 하기 위해서 처음부터 모든 정보를 사용자에게 제공하지 않는다.

④ 표준 UI 가 존재하지 않는다.

게임은 참신한 아이디어 즉 독창성이 많을수록 더욱 가치가 높아지고, 일반적 기능이 존재하지 않는 만큼 UI 의 표준이 존재하기 어렵다.

2.2 ISO/IEC 9126 품질모델

ISO/IEC 9126(Information Technology-Software Quality Characteristics and Metrics)은 품질특성 및 메트릭을 정의하고 있는 표준으로, 각 품질 특성별로 세부 메트릭을 제시하고 있다. 표준화된 소프트웨어 품질모델에 관한 연구는 McCall, Boehm[3], Deutsch & Willis[4], IEEE[5]를 통해 제시되어 왔으나, 현재 국제적인 표준으로 인정 받고 있는 품질모형은 ISO/IEC 9126 이다. 최근에는 두 표준을 합치기 위한 노력이 SQuARE (Software Product Quality Requirements and Evaluation) 프로젝트로 진행되고 있다[6]. 1991 년 제정된 ISO/IEC 9126 은 소프트웨어의 내, 외부적인 특성에 따라 품질을 평가하는 모델이다. 6 개의 외부 품질특성과 적합성, 정확성 등 27 개의 내부 품질특성으로 구분할 수 있다. 각각의 특성은 소프트웨어의 품질을 좌우하는 요소로 존재한다[7].

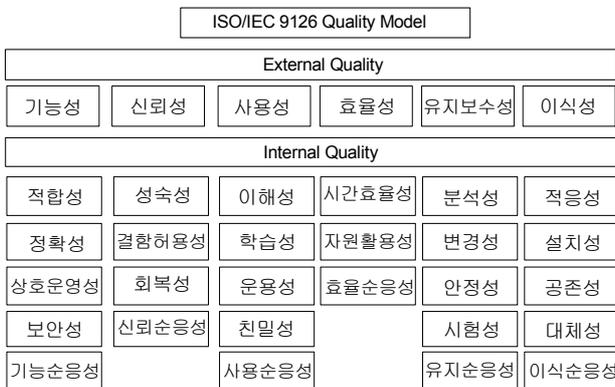


그림 1. ISO/IEC 9126 품질모델

2.3 소프트웨어 시험

소프트웨어의 시험은 크게 개발 과정에 포함되는 개발 공정으로서의 반복 시험과 개발된 소프트웨어를 시험하는 인수 시험으로 구분할 수 있다. 각 시험의 특징 및 시험 수행 시 효과는 다음과 같다.

2.3.1 개발 공정으로서의 시험

개발 공정 전 과정에서 반복적으로 수행되며, 소프트웨어의 개발을 계획적으로 추진하여 정해진 기간 내에 일정한 품질 이상의 소프트웨어 개발을 성공시키는 것이 목적이다. 개발 공정에서 시험은 소프트웨어 개발 프로젝트의 설계 단계에서부터 종료 단계까지 모든 과정에서 반복적으로 수행되며, 대부분 화이트박스 시험 방법을 이용한다.

2.3.2 인수 시험

개발된 소프트웨어를 대상으로 수행되며, 품질의 모든 요소를 시험함으로써 소프트웨어의 품질을 확보하는 것을 목적으로 한다. 대부분 블랙박스 시험 방법을 이용하여 수행하며, 소프트웨어 요구사항이 정확하게 구현되었는가를 시험하게 된다.

3. 패키지게임 S/W 품질평가기법

3.1 패키지게임 S/W 품질모델

ISO/IEC 9126 과 같은 기존의 품질 모델은 범용의 S/W 품질모델에 관한 정의로서 게임 S/W 의 특성을 특별히 반영하지 못하므로 게임 S/W 의 특성에 적합한 외부특성 및 내부특성을 추출하여 품질모델을 재구성한다. 또한 해당 특성을 평가하기 위한 정량적인 기준이 제시되어 있는 경우가 많지 않다. 정량적인 평가기준이 없는 평가결과는 주관적 평가에 의존하게 됨으로써 품질평가결과가 갖추어야 하는 반복성, 재현성, 공평성, 객관성을 보장하기 어렵게 된다. 따라서 정량적인 평가가 불가능한 품질특성보다는 정량적 평가가 가능한 품질 특성에 중점을 둔다[8]. 평가방법은 각 내부특성들을 가장 정확하고 타당하게 평가하기 위한 시험기술을 제공한다.

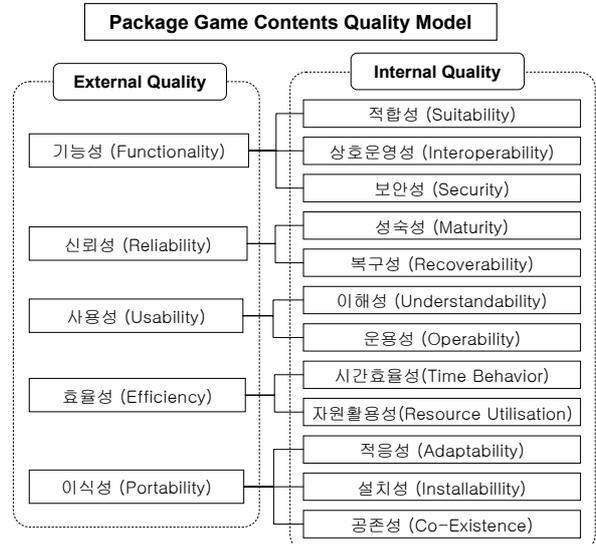


그림 2. 패키지게임 S/W 품질모델

3.1.1 기능성

게임의 기능에 관련이 있는 부분을 평가하는 품질 평가 항목으로서, 게임 기획서, 요구사항 분석서, 매뉴얼 등을 참조하여 명시된 요구와 내재된 요구를 평가 수행한다.

① 적합성(Suitability)

게임내에서 구현하고자 하였던 모든 기능이 의도한 대로 구현되었는가를 말한다.

② 상호운영성(Interoperability)

입출력장치, 플랫폼 및 라이브러리 에서 의도한 모든 기능이 작동하는가를 말한다.

③ 보안성(Security)

게임 S/W 에 사용자가 임의대로 변경 또는 영향력을 행사하지 못하도록 함을 말한다.

3.1.2 신뢰성(Reliability)

게임 S/W 에 명세된 조건하에서 사용할 때 성능수준을 유지할 수 있는지를 평가하는 항목이다.

① 성숙성(Maturity)

게임내에서 에러, 버그 등으로 인하여 결함이 발생하지 않도록 대처할 수 있는 능력을 말한다.

② 복구성(Recoverability)

결함 발생시 명세된 성능 수준을 재유지하고 직접적으로 영향받은 데이터를 복구하는 능력을 말한다.

3.1.3 사용성(Usability)

게임을 플레이할 때 나타나는 사용상의 적합성을 평가하는 항목이다.

① 이해성(Understandability)

사용자 매뉴얼을 적절히 구성하는 것과 게임의 기능에 대한 설명, 정의, 방법을 기술하며 가독성이 있어야 함을 말한다.

② 운용성(Operability)

게이머가 게임을 플레이하기 위해 필요한 조작의 일관성과 편의성이 존재하는가

3.1.4 효율성(Efficiency)

컴퓨터 시스템은 많은 자원으로 구성된다. 게임 S/W 는 이러한 자원을 소비하면서 실행이 된다. 효율성 품질평가는 컴퓨터 자원에 따른 게임의 성능을 평가한다.

⑤ 시간효율성(Time Behavior)

게이머가 게임을 플레이할 때 적절한 반응 및 처리시간과 처리율을 제공함을 말한다.

⑥ 자원효율성(Resource Behavior)

명세된 환경에서 다양한 자원을 적절하게 운영함을 말한다.

3.1.5 이식성(Portability)

다양한 장치 및 윈도우 버전의 환경 변화에 따라 게임 S/W 가 적응하는지를 검증하는 평가 항목이다.

① 적응성(Adaptability)

시스템 환경이 변경되어도 게임 S/W 가 작동

가능함을 말한다.

② 설치성(Installability)

명세된 환경에서 게임 S/W 가 정상적으로 설치되고 제거됨을 말한다.

③ 공존성(Co-Existence)

공유시스템 리소스를 독점 또는 변경하지 않음을 말한다.

3.2 패키지게임 S/W 의 시험기법

실제 시험 단계에서는 패키지게임 S/W 품질모델에 대하여 모든 시험이 일률적으로 대입되는 것은 아니며, 패키지게임 S/W 의 품질 요구사항에 따라 해당되는 패키지게임 S/W 품질모델을 선별적으로 수행하게 된다.

패키지게임 S/W 의 시험은 모든 경우의 수를 따져가며 작업하는 매우 방대하며 단순 반복적인 작업이다. 그렇기 때문에 패키지게임 S/W 시험은 항상 체계적인 프로세스 및 시험 도구를 이용하여 수행하게 된다. 시험의 단순 반복과정을 자동화한다면 패키지게임 S/W 의 시험에 소요되는 노력의 양은 획기적으로 줄어들어 시험을 매우 효율적으로 수행할 수 있으며, 시험의 프로세스가 잘 정립되어 있다면 시험을 체계적으로 수행함으로써 쓸데없는 시간 낭비를 최소화할 수 있다. 그러나 많은 경우 시험 결과의 판단은 오직 인간만이 수행할 수 있기 때문에 시험과정 중 판단이 필요한 대부분의 단계에서는 사람이 직접 수동으로 시험을 수행하는 경우가 많다.

3.2.1 기능성

기능성을 시험하기 위해서는 테스트 대상 게임소프트웨어에 적합하게 만들어진 기능성 체크리스트를 작성하여야 한다. 기능성 체크리스트는 게임소프트웨어의 기획서 등 개발 문서를 참고하여 작성할 수 있으며, 개발팀의 회의를 거쳐 확정할 수 있다.

레이싱 시뮬레이션 게임에서 ‘브레이크’ 기능이 구현되었다고 명시되었을 때에는 브레이크가 반드시 구현되어야 하며, 브레이크 구현에 필요한 제반 사항들도 함께 구현되어야 한다. 이러한 모든 기능이 구현되었을 때, 해당 게임콘텐츠는 기능성이 있다고 한다. 또한 각종 그래픽카드, DirectX 등 에서 충돌이 발생하지 않아야 하며 지원 불가능한 특정사항에 대해서는 명시되어야 한다.

3.2.2 신뢰성

신뢰성을 시험하기 위해서는 소프트웨어가 설치될 수 있는 여러 가지 환경을 구성하여 해당 환경에서 테스트하여야 한다. 패키지게임 또는 온라인게임의 경우 모든 환경을 직접 구성하지 않고도 베타테스트를 통하여 게임 소프트웨어의 안정성을 높일 수 있다. 신뢰성을 시험하기 위한 방법은 다음과 같다.

- 다양한 플랫폼에서 플레이 수행
- 발생 가능한 에러에 대한 처리여부 검증
- 메모리 누수 검증

3.2.3 사용성

사용성 시험은 사용자가 게임소프트웨어를 사용함에 있어 불편한 점이 없는가를 검증하고, 불편한 점을 밝혀내는데 목적이 있다. 그렇기 때문에 사용성 테스트는 자동화하기 매우 까다로우며, 특정한 목적을 가진 프로그램이 아닌 일반 응용 프로그램에 사용될 수 있는 사용성 시험 자동화도구는 아직 개발되지 않았다.

a) 이해성

매뉴얼은 게이머가 게임을 이해하는데 도움을 주며, 게임의 요소를 설명함으로써 게임에 흥미를 갖도록 유발하는 매체이다. 가령 게임의 시리얼 키에 찍어 있는 숫자 1 과 영어 소문자 l, 숫자 0 과 영어 대문자 O 의 구분이 불가능한 폰트로 찍어 있어 사용자는 이로 인하여 상당한 혼란을 겪을 수 있다.

b) 운용성

전략시뮬레이션의 경우 많은 커맨드들이 존재하므로 이를 빨리 처리할 수 있는 단축키의 지원, 카레이싱 게임일 경우 입력장치에서 휠을 지원할 수 있어야 한다.

3.2.4 효율성

효율성을 검증하기 위한 간편한 방법으로서 최소 사양 및 권장 사양의 적정성을 시험하는 방법이 있다. 최소 사양은 게임이 동작하기 위한 최소 사양이며, 권장 사양은 게임이 원활하게 작동하기 위한 최소 사양이다. 최소 사양 및 권장 사양에서 게임을 동작시킬 때 기대한 만큼의 성능이 구현되는가를 검증함으로써 게임의 효율성을 시험할 수 있다.

3.2.5 이식성

게임소프트웨어가 설치될 OS 환경하에서 설치 및 제거 과정을 진행하여 본다. 다음의 사항을 검증함으로써 설치성을 시험할 수 있다.

- ▶ 설치 및 제거가 원활하게 동작하는가를 검증
- ▶ 설치와 제거 수행 후 설치 이전 상태로 완벽하게 되돌아왔는가를 검증
- ▶ 설치 도중 일어날 수 있는 문제를 최대한 발생시켜 각각의 상황에서 설치 프로그램이 적합한 동작을 하는가를 검증

4. 고 찰

ISO9126 은 품질모델에 따른 여섯가지 품질요소를 제시하고 있으며 표현모델과 측정모델 간의 구조적 일관성이 우수하며, 특히 서비스 품질에 대한 개념을 도입하여 개발단계, 시험 및 인수단계, 운용단계 전반에 걸쳐 적용이 가능한 품질모델을 구축함으로써 평가 체계의 완전성을 확보하였다는 측면에서 우수하다고 할 수 있다[8]. 그러나 소프트웨어의 품질을 어떤 시각으로 보는가에 따라 여러 가지 품질 모델이 있을 수 있으며 품질 모델이 달라짐에 따라 품질요소도 달라진다. 품질평가는 품질요소 단위로 수행하는 것이 바람직하다. 그렇지만 실제로 지금까지 진행된 많은 연구들은 하위 품질요소들을 종합하여 단일한 값을

얻기 위해서 노력하였다. 하지만 소프트웨어의 품질을 단일 결과로 표현하거나 소프트웨어 전체 품질에 대해서 품질평가를 수행하는 것은 바람직하지 못하다.

따라서 편의성에 대한 평가, 가용성에 대한 평가, 성능에 대한 평가와 같이 소프트웨어 품질평가는 최종 사용자가 이해 가능한 소프트웨어 개별 품질요소들에 대해서 단위별 평가를 수행하여야 하며 품질요소별 우선순위에 따른 합리적인 가중치 결정방식을 추가하여 품질요소별로 영향력을 차등화시켜야 할 것이다. 또한 상기의 품질특성을 바탕으로 반복성, 재현성, 공평성, 객관성을 확보하기 위해 활동, 제약조건 그리고 자원을 포함할 수 있는 품질평가 프로세스를 정립하며 상용 게임콘텐츠에 대한 시험평가를 지속적으로 실시하여 본 논문에서 제안한 품질평가방법이 좀더 유용한 모델이 되도록 보완해야 할 것이다.

5. 결 론

본 연구를 통해 궁극적으로 도달하고자 하는 목적은 게임콘텐츠 품질향상에 기여하는데 있다. 게임콘텐츠는 게임엔진, 애니메이션, 그래픽, 캐릭터, 시나리오, 사운드 등 다양한 디지털 콘텐츠를 결합한 복합적인 산업이다. 이 때문에 다양한 디지털 콘텐츠 분야를 모두 고려해야만 적절한 품질평가를 수행할 수 있다. 다양한 게임을 분석하다 보면 몇 가지 공통적인 특징을 발견할 수 있다. 이 중 특히 게임콘텐츠에 영향을 주는 요소로는 콘텐츠적 요소, 기술적 요소, 서비스적 요소 등을 들 수 있다. 이 요소들을 각 부문의 종사자들이 이해하고, 적용할 수 있도록 품질에 관한 정량적 결과를 제공하는 것이 필요하다. 또한, 품질에 대한 정량적 결과를 바탕으로 게임콘텐츠 업체가 보다 우수한 게임 S/W 를 제작하는데 활용할 수 있는 정보를 제공하며 게임콘텐츠의 품질향상에 도움이 되고자 한다.

참고문헌

- [1] (재)한국게임산업개발원 “2003 년 대한민국 게임 백서” 2003
- [2] 문화관광부 법률 제 6552 호, “음반, 비디오물 및 게임물에 관한 법률,” 일부개정 2001.12.29
- [3] Boehm, B. W., “Software Engineering Economics” Prentice-Hall, NJ, 1988.
- [4] Deutsch, M. and Willis, R. “Software Quality Engineering” Prentice Hall, NJ, 1988
- [5] IEEE Computer Society, IEEE-STD-1061 “IEEE Standard for a Software Quality Metrics Methodology” 1992.
- [6] Motoei AZUMA, SQuaRE (Software Product Quality Requirements and Evaluation), 2001. Jan.
- [7] ISO, ISO/IEC 9126 “Information Technology - Software Quality Characteristics and Metrics” 1996.
- [8] Yingxu Wang, “Software Engineering Standards : Review and Perspectives” Handbook of Software Engineering and Knowledge Engineering, 2001