

게임제작 지원도구(GDSS) 설계 및 구현

장인걸*, 우종식*, 이승훈**, 이남용**

한국게임산업개발원, 숭실대학교 컴퓨터학과

ig8961@gameinfinity.or.kr, jswoo@gameinfinity.or.kr, shlee@selab.ssu.ac.kr,

nylee@computing.soongsil.ac.kr

A Design and Implementation of Supporting System for Game Development(GDSS)

In-Geol Jang*, Jong-Sik Woo*, Seung-Hun Lee**, Nam-Yong Lee**

* Korea Game Development & Promotion Institute,

**School of Computing, Soongsil University

요 약

국내 게임 산업은 여타 문화 산업과 비교하였을 때 상업화의 시기는 가장 늦었음에도 불구하고 2002 년을 기준으로 3 조 4 천억에 이를 정도로 급성장을 하였다. 그러나 게임 제작사의 95% 이상이 적자에 시달리고 있는데 이는 게임 제작과 관련하여 개발 단계에 있어 담당자의 역할 분배 및 각 단계에 대한 적절한 관리를 하지 못해 팀원간 의사소통, 게임제작 공정 생성물 관리, 핵심 기술 및 관리 등의 문제가 발생하기 때문이다. 이러한 문제점들은 CASE 툴을 통해서 해결이 가능하다. CASE 툴을 통해 게임제작 공정에 있어 팀원간의 의사 소통과 인력 자원 관리, 공정 자료의 버전 컨트롤 및 이력관리의 정보를 체계적으로 관리할 수 있도록 한다. 이에 본 논문에서는 ISO12207 과 RUP 프로세스를 기반으로 게임제작 공정의 생산성과 효율성을 증가시킬 수 있는 게임제작 지원도구(GDSS)를 설계 및 구현하였다.

1. 서론

다른 산업에 비해 상업화가 가장 늦었음에도 불구하고 국내 게임 시장은 엄청난 속도로 성장하고 있다. 이러한 국내 게임산업의 급속한 성장은 국내 시장 활성화와 인력 수요의 창출, 외화 획득 등 다분히 긍정적인 역할을 수행하고 있다. 하지만 이러한 국내 시장의 급성장 및 활성화의 이면에는 여러 가지 문제점을 가지고 있다.

국내 게임제작사의 95% 이상이 적자에 시달리거나 제품화의 실패로 어려움을 겪고 있다. 이러한 문제점들은 체계적이지 못한 개발과 관리가 가장 큰 문제로 지적되고 있으며, 다음으로 제품의 생명주기가 짧고, 사용자들의 요구가 다양해 신속히 반영하지 못해 경쟁력이 떨어지고, 기업문화와 개발자문화, 체계적인 학문적 연구의 부족 등에 따라 게임제작을 위한 필수적인 정보 획득 작업을 하지 못하기 때문이다[5][10].

본 논문에서는 ISO12207 과 RUP 프로세스를 기반으로 게임제작 기획 단계부터 테스트를 거쳐 배포에 이르는 전 과정의 공정 관리 및 품질과 생산성을 향상시킬 수 있는 CASE 툴을 설계 및 구현하였다[4].

본 논문의 구성은 다음과 같다. 2 장에서는 시스템의 이해를 위해 기본 개념을 기술하고, 3 장에서는 시스템을 설계하며, 4 장에서는 구현, 마지막으로 5 장에서는 결론 및 향후 연구과제를 제시한다[1].

2. 관련 연구

2.1 ISO 12207

ISO 12207 은 프로세스(Process), 활동(activity) 및 태스크(Task)의 집합으로 구성되어 있으나 프로세스에 포함되어 있는 활동 및 태스크를 어떻게 수행해야 한다는 것은 명시하지 않고 있다. ISO 12207 은 특정한 생명주기 모델이나 소프트웨어 개발 방법론을 전제하지는 않으며 통일된 공통 프레임워크를 제공하기 위한 목적으로 개발되었다.

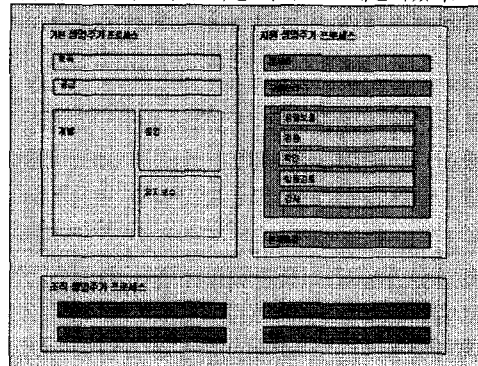


그림 1. ISO 12207 의 구조

위의 그림 1은 ISO 12207의 구조로서 소프트웨어 생명주기 동안 수행될 수 있는 활동을 기본 프로세스와 지원 프로세스 그리고 조직 프로세스로 구분하여 나타낸 것이다.

2.2. RUP(Rational Unified Process)

RUP는 반복적 개발 방법을 이용한 전통적인 Waterfall 프로세스와 비교했을 때, 초기에 위험요소를 줄일 수 있고 변경에 대한 관리가 용이하며, 보다 높은 수준의 재사용이 가능한 Rational사의 개발 방법론이다. RUP는 프로젝트의 구성을 크게 네 단계(Inception, Elaboration, Construction, Transition)로 나누며, 각 단계는 1회 이상의 반복(Iteration)으로 구성된다. 또한 각 반복은 요구사항 분석, 설계, 구현, 테스트 및 평가 과정을 포함하고 있어 자체만으로도 하나의 개발 주기를 이룬다[6].

RUP는 9개의 핵심이 되는 워크플로우(Workflows)가 존재하는데 9개의 워크플로우는 다시 6개의 엔지니어링을 위한 워크플로우와 3개의 지원을 위한 워크플로우로 나누어 진다.

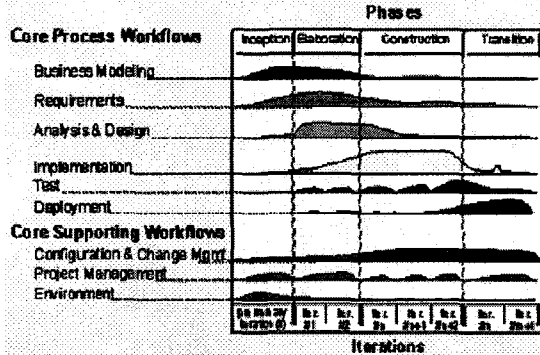


그림 2. Rational Unified Process의 구조

2.3. CASE 툴

CASE라는 용어는 소프트웨어 개발을 지원하는 시스템 또는 소프트웨어 개발 방법의 자동화를 지원하는 시스템을 말한다[7][8][11].

CASE는 소프트웨어 복잡도의 증가와 의사소통의 어려움, 소프트웨어의 위기 등으로 인하여 등장하였으며, CASE 툴은 소프트웨어의 수명주기에 대한 통합과 지원 기능, 분석 기능, 정보 저장 기능, 시스템 기술 능력의 4가지 기본 기능을 제공한다[9][11].

다음 표 1은 CASE 툴의 사용을 통해 얻을 수 있는 효과를 정리한 것이다

표 1. CASE 툴 사용 효과

소프트웨어 품질 향상
재사용성 증대
효율적인 관리
팀원간 의사소통 지원
효율적인 문서화 지원
제작 공정의 표준화 및 정형화

3. 게임제작 지원도구(GDSS) 설계

3.1. 범위

게임제작 지원도구(GDSS : Game Development Support

System)는 ISO 12207과 RUP 등의 기존 소프트웨어 개발 방법론과 표준 프로세스를 바탕으로 게임만의 특징이라고 할 수 있는 그래픽과 사운드 등의 요소 식별 및 게임제작의 특성을 고려한 후, 게임 제작에서 요구되는 관련 기술들을 고려하여 그림 3과 같이 게임제작 공정을 작성하였다.

그림 3. 게임제작 공정

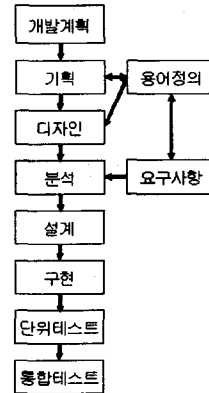
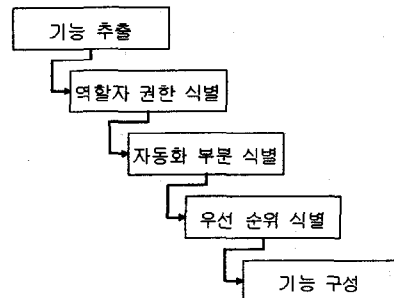


그림 3의 게임제작 공정을 살펴보면 디자인 단계에서 그래픽과 사운드에 대한 요소가 포함되어 있고 용어정의의 경우는 게임제작 전 공정에서 사용되는 모든 용어들을 관리하기 위한 기능이다. 이는 디자이너나 프로그래머와 같이 서로 다른 공정을 담당하는 구성원들 간의 의사소통 및 게임의 품질을 높이는 데 중요한 공정이라고 할 수 있다.

작성된 게임제작 공정은 그림 4와 같이 5 단계를 거치면서 GDSS에서 지원할 기능을 식별하게 되는데 이 중에서 가장 중요한 단계가 2 단계인 “역할자 권한 식별”이다. “역할자 권한 식별”이란 프로세스상의 공정을 처리할 수 있는 능력이나 기술, 책임을 잘 팀원을 배정하는 것으로 이를 통해 GDSS에 접근할 수 있고 게임을 제작하게 된다. 이 과정이 중요한 이유는 게임업체 성격상 하나의 팀원이 여러 역할을 수행해야 하는 경우가 많기 때문에 이를 효율적으로 배치하고 구성하여야 GDSS가 원활하게 운영될 수 있기 때문이다.

그림 4. GDSS 기능 식별



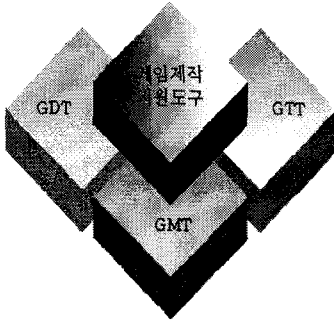
결국 GDSS는 게임제작 전 공정을 지원하는 CASE 툴이다. 그러나 CASE 툴의 기본 기능인 편리성과 효율성을 위하여, 그림 4의 3 단계 자동화 부분 식별 단계를 통하여 툴로 지원할 경우 오히려 공정이 복잡해지거나 생략 가능한 공정의 경우는 자동화 부분에 포함시키지 않았다. 따라서 게임제작 시 GDSS를 도입한다면 게임 기획단계부터 최종 테스트 단계까지 GDSS를 통하여 품질 높은 게임을

제작할 수 있을 것이다.

3.2. 구성

GDSS 는 그림 5 와 같이 GMT(Game Management Tool), GDT(Game Development Tool), GTT(Game Test Tool) 3 개의 툴로 구성이 된다.

그림 5. GDSS 구성



GMT 는 관리를 지원하는 시스템으로 기획, 용어정의, 요구사항, 개발계획으로 구성되며 세부 구성은 표 2 와 같다.

표 2. GMT 구성

구성	주요 내용 및 산출물
기획	비즈니스케이스 시놉시스, 소요제기서 위험요소목록 게임운영 및 개발 환경서
용어정의	용어정의
요구사항	게임시스템요구명세서 게임소프트웨어요구명세서
개발계획	게임개발계획서

GDT 는 게임 개발을 위한 디자인, 분석, 설계, 구현으로 구성되며 세부 구성은 표 3 과 같다.

표 3. GDT 구성

구성	주요 내용 및 산출물
디자인	시나리오, 게임요소명세서 게임규칙명세서 사용자메뉴얼
분석	시스템아키텍처기술서 소프트웨어아키텍처기술서
설계	제사용컴포넌트기술서 사용자인터페이스설계기술서 DB 설계기술서 컴포넌트설계기술서 그래픽/사운드 설계기술서
구현	그래픽/사운드 리소스 정보 컴포넌트코드 정보 DB 코드 정보

GTT 는 게임 개발 과정의 완성도와 품질을 높이기 위한 단위테스트, 통합테스트로 구성되며 세부 구성은 표 4 와 같다.

표 4. GTT 구성

구성	주요 내용 및 산출물
단위테스트	단위테스트결과보고서
통합테스트	게임소프트웨어테스트결과보고서 게임시스템테스트결과보고서

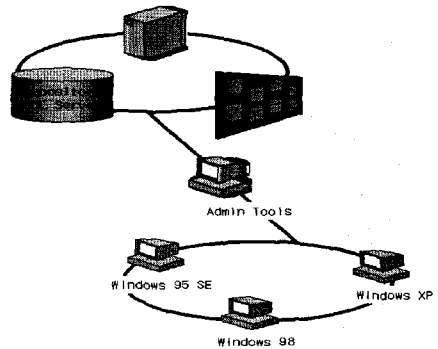
3.3. 기대효과

GDSS 는 ISO 12207 과 RUP 등의 표준 프로세스를 기반으로 CASE 툴의 특징인 자동화를 통한 고품질의 게임제작을 지원한다. GDSS 는 사용자 편리성과 GUI 를 통한 팀원간의 의사소통을 지원하고, 형상관리 기능을 기반으로 모든 자원을 관리하기 때문에 이력관리나 버전관리 등 프로젝트 진행에 있어 중요한 부분들을 자동으로 지원한다. 따라서 이를 통해 게임 개발 생산성과 품질의 향상, 효율적인 게임 개발 자원 관리가 가능하다.

4. 게임제작 지원도구(GDSS) 구현

GDSS 는 C/S 형태로 구현하였고 Windows O/S 를 지원하며, Visual C++ 6.0 과 SQL Server2000, Access 2000 등을 사용하여 개발 하였다.

그림 6. GDSS 시스템 구성도



GDSS 를 사용하기 위해서는 가장 먼저 사용자 인증 과정을 거쳐야 한다. 사용자 인증 과정은 GDSS 를 사용할 수 있는 권한으로 Admin 툴에 의해 관리 되어지는데, 그림 7 은 Admin 툴에서 사용자 권한을 설정하는 과정을 보여주고 있다.

그림 7. Admin 툴 - 사용자 권한 부여

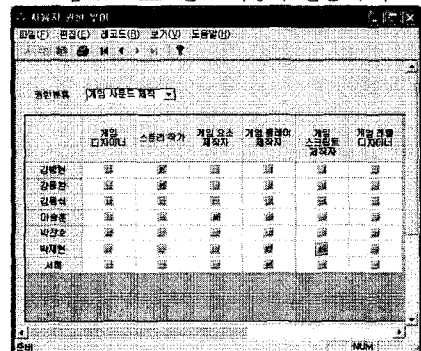


그림 7의 과정을 통해 사용자들은 게임제작 공정상에서 본인들이 진행해야 하는 역할의 권한을 부여 받게 된다. 사용자들은 다수의 권한을 가질 수 있도록 되어 있는데, 이는 프로세스 상에서의 역할과 사용자들의 역할이 일치하지 않는 것을 해결하기 위한 기능이다.

그림 8. GDSS - 시놉시스작성

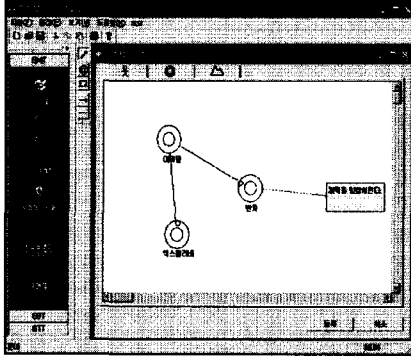


그림 8은 게임제작 공정 중에서 시놉시스를 작성하는 모습을 보여주고 있다. 재미있고 완성도 높은 게임이 제작되기 위한 기반 작업이 시놉시스작성이다. GDSS에서는 자체 정의한 GDL(Game Design Language)를 통하여 시놉시스를 GUI 형태로 작성하고, 작성된 시놉시스를 사용자들이 검토 및 의견을 작성해서 수정할 수 있는 기능을 제공한다.

그림 9. GDSS - 게임규칙명세서

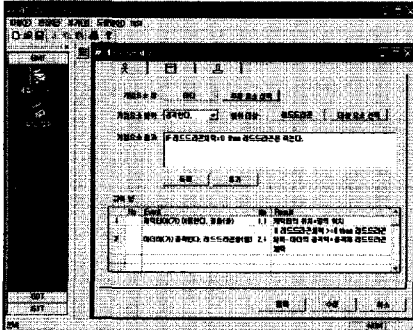


그림 9는 시놉시스에서 추출된 게임 요소에 게임규칙을 할당하는 과정을 보여주고 있다. 그림 8에서 작성된 GDL 자료를 기반으로 각각의 게임규칙을 생성하는데, 모든 과정이 자동화되어 지원되기 때문에 사용자는 몇 번의 마우스 클릭만으로도 작업을 진행할 수 있다.

5. 결론 및 향후 연구

게임산업의 급성장은 국내 게임시장의 활성화와 신규인력 수요의 창출, 외화 획득 등 긍정적인 역할을 수행하고 있지만, 그 이면에는 대다수 업체의 경영난 악화와 게임 생산성 및 품질의 저하라는 문제들을 가지고 있었다.

본 논문에서는 이러한 문제들을 해결할 수 있는 CASE 툴인 GDSS를 통하여 게임의 생산성과 품질을 향상시킬 수 있는 방법을 제시하였다. 이러한 CASE 툴의 사용은 단기적으로는 관리적인 장점이 많지만 향후 레퍼지토리 구축을 통하여 게임의 가장 중요한 요소중의 하나인 재미 요소의 정

량화라든가 보다 더 재미있고 경쟁력이 있는 게임을 제작할 수 있는 장점을 제공할 수 있을 것이다.

향후 연구되어야 할 부분으로는 게임제작시 생성된 정보의 레퍼지토리 구축, 웹 연동으로 외부에서도 시스템에 접근하여 작업이 가능할 수 있도록 웹 기반의 인터페이스 기능을 제공할 수 있어야 한다. 또한 보다 많은 게임제작 업체의 의견을 받아들여 표준화된 게임제작 프로세스를 바탕으로 표준화된 게임제작 지원도구로 성능개선이 이루어져야 한다.

6. 참고문헌

- [1] Roger S. Pressman, *Software Engineering, A Practitioner's Approach*, 3rd Ed. McGraw Hill, 1997.
- [2] Alka R. Harriger *Computer Programming with Visual Basic 6 A Problem-Solving Approach*, Que E&T, 1999.
- [3] Thomas M.Pigoski, *Practical Software Maintenance*, Wiley, pp. 37-50, 1997.
- [4] N.T. Fletton and M. Munro, *Redocumentating S/W Systems using Hypertext Technology*, IEEE Intl conf. Software Maintenance, 1988.
- [5] H. Mili, F. Mili, and A. Mili, *Reusing Software : Issues and Research Directions*, IEEE Transactions on Software Engineering, Vol. 21, No 6, pp. 528-561, June 1995.
- [6] ISO 12207, "ISO/IEC Standard for Information Technology - Software life cycle process", ISO/IEC/ JTC/SC7, March, 1998.
- [7] McGill, R., *Reverse Engineering and Reengineering - not yet?*, In Software Reuse and Reverse Engineering in Practice, Hall, P.A.V, Chapman&Hall, 1992.
- [8] 이헌기, 신규상, 역공학에서 재문서화를 지원하는 도구, 정보처리학회 추계 학술발표논문집, 제 5 권 제 2 호, pp. 433-436, 1998.
- [9] 김인성, "CASE 의 기본 개념", 포스테이터즈, pp. 11-491993
- [10] "2002 대한민국 게임백서", 한국게임산업개발원, 2002
- [11] "게임제작 공정관리 자동화도구 개발기술", 한국게임산업개발원, 연구보고서 02-005, 2002.7