

모듈기반의 벽돌깨기 온라인 게임 개발 및 구현

임종혁*, 정화영**

목원대학교 컴퓨터공학과*, 예원예술대학교 전자상거래학과**
bestofbest99@empal.com*, jhymichael@empal.com*

Development and Implement of Module Based a Brick Destruct Online Game

Jong-hyuk Im*, HwaYoung Jeong**

Dept. of Computer engineering Mokwon University*, Dept. of Electronic Commerce Yewon Art University*

요 약

인터넷의 발달로 인해 이제는 집에서 PC로만 해오던 게임을 언제 어디서든 인터넷만 가능하다면 게임을 즐길 수 가 있게 되었다. 이 게임은 웹의 편리함과 자바의 애플릿 기술을 이용하여 인터넷 온라인 게임을 설계 및 구현하였다. 웹 상에서 이루어진다는 편리함을 가지고 있지만, 시간과 장소의 제약이 크고, 동시에 여러 사람이 함께 게임을 할 수 없다는 단점을 가지고 있다.

본 논문에서는 모듈기반의 벽돌깨기 게임을 개발 및 구현하였다. 시스템 분석 및 설계에서는 UML을 이용하여, 자바 애플릿으로 구현하여 웹 상에서도 이용 가능하였다.

1. 서 론

최근 인터넷이 급속도로 발전하면서 동시접속자수가 20만명에 이르는 온라인 게임의 등장에 비하면 초라하기 그지없다. Java기반의 게임은 다른 온라인 아케이드 게임과 같은 온라인 대전 게임으로의 발전 가능성을 충분히 앞두고 있다고 할 수 있겠다. [1,3]

본 논문은 자바 애플릿 게임인 벽돌깨기 게임의 설계 및 구현하였다. 시스템의 분석 및 설계는 UML(Unified Modeling Language)을 이용하였다. 또한, 자바 애플릿을 이용하여 모듈기반으로 구현함으로써 웹 상에서의 이용이 가능하였다.

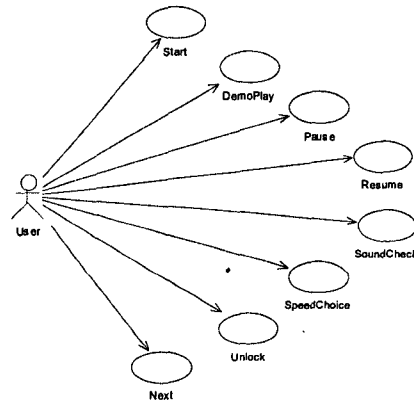
2. 온라인 게임 개발 동향

1990년대 중반까지만 해도 국내 게임시장은 아케이드 시장과 PC게임, 비디오게임이 양분하고 있었으며, 후반부에 들어서는 PC게임시장의 대약진, 온라인 게임시장은 시장성 자체가 불투명해 게임 업체들은 이 때까지만 해도 추이를 지켜보고 있었다. 그러나 엔씨소프트가 개발한 리니지의 대성공이 온라인 게임 개발의 기폭제가 되었다. 현재 서비스중인 온라인 게임들은 유/무료 전부 합하여 200개 가량 된다. 무료 서비스중인 온라인 게임은 160여가지 되는 것으로 추산되며 상용중인 온라인 게임은 대략 40여개에 이르고 있다. 그러나 이것은 공식적으로 밝혀진 현황이며 이 시간에도 노출되지 않고 물

밑에서 기획 중이거나 개발 중인 게임들이 많다.[1]

3. 온라인 벽돌깨기 게임 분석 및 설계

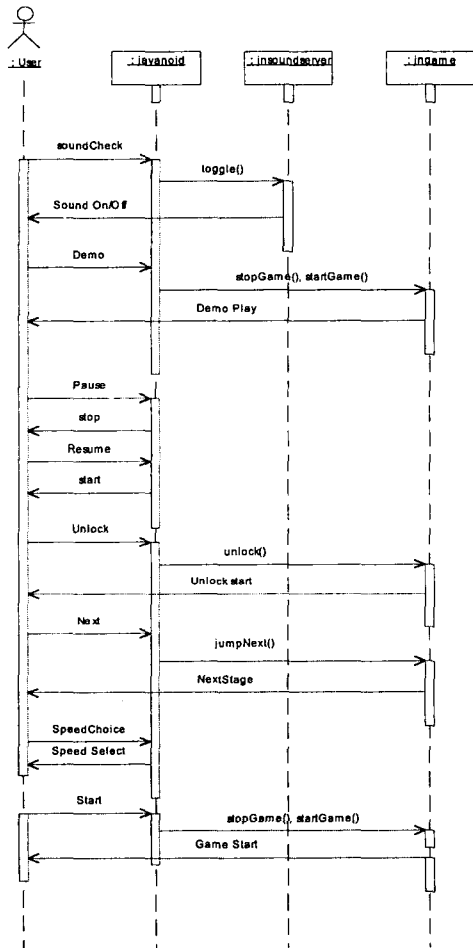
벽돌깨기는 게임에 필요한 클래스들을 모듈별로 세분화하였고, 애플릿 게임이기 때문에 웹 상에서 게임을 즐길 수가 있다.



<그림 1> UseCase Diagram

또한 사용자는 게임을 시작하기 전에 게임의 속도, Sound On/Off를 정할 수가 있고, DemoPlay 감상과 다음 스테이지를 볼 수가 있다. 게임을 시작하고 난 후에는 일시정지를 통해 중간에 게임을 멈췄다가 다시 재

개 할 수도 있다. 또한 각 모듈의 효율적인 동작을 위해 멀티 스레드를 사용하였다. <그림 1>은 유즈 케이스를 통한 간단한 구조를 나타낸다. 또한, <그림 2>는 각 모듈 객체들간의 이벤트 흐름을 나타내는 시퀀스 다이어그램이다.



<그림 2> Sequence Diagram

4. 온라인 벽돌깨기 게임 구현

벽돌깨기는 SDK1.4를 이용하여 자바 애플릿으로 구현되어 있으며 사용자가 게임을 하기 위해 필요한 것들은 메인 클래스에 구현되어 있거나 메인 클래스를 통하여 필요한 모듈을 얻을 수가 있고, 각각의 클래스들을 통제 관리한다. 다음은 본 게임 로직의 중요부분인 스레드와 이벤트 핸들러 구현부분을 나타낸다.

```

// 스레드와 이벤트 핸들러 구현 부분[2]
public void stop() {
    if (myThread != null) myThread.stop();
    myThread = null;
}

public void start() {
    if (myThread == null) {
        myThread = new Thread(this);
        myThread.setPriority(Thread.MIN_PRIORITY);
        myThread.start();
    }
}

// Run method
public synchronized void run() {
    while (Thread.currentThread() == myThread) {
        try {
            if (game != null) { game.live(); game.repaint();
                if (sndServer != null) sndServer.iterate();
                Thread.sleep(animationDelay);
            }
        } catch (InterruptedException e) { break; }
    }
}

// Event handler (panel)
public boolean action(Event evt, Object arg) {
    if (evt.target == speedChoice) {
        animationDelay = 5
        * speedChoice.getSelectedIndex();
    }
    else if (evt.target == soundCheck) {
        if (sndServer != null) sndServer.toggle();
    }
    else if ("START".equals(arg)) {
        if (game != null) { game.stopGame();
            game.startGame(false); }
    }
    else if ("DEMO".equals(arg)) {
        if (game != null) { game.stopGame();
            game.startGame(true); }
    }
    else if ("PAUSE".equals(arg)) stop();
    else if ("RESUME".equals(arg)) start();
    else if ("UNLOCK".equals(arg)) {

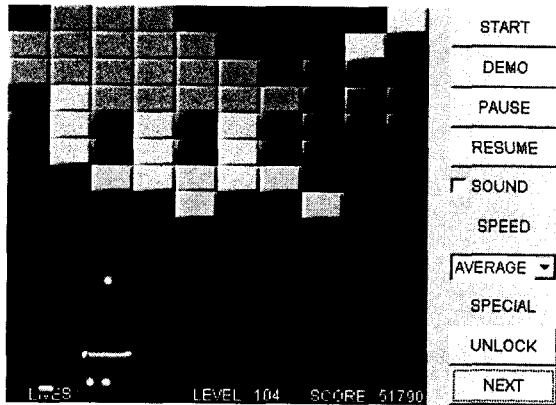
```

```

        if (game != null) game.unlock();
    }
    else if ("NEXT".equals(arg)) {
        if (game != null) game.jumpNext();
    }
    return true;
}

```

본 벽돌깨기 게임은 사용자와 모듈간의 상호작용과 모듈과 모듈간의 상호 작용도 함께 동작한다. 즉, 사용자는 메인 클래스를 통하여 필요한 모듈을 얻지만 각각의 모듈들은 또 다른 모듈과의 연결을 통하여 게임을 구성하고 있다. 다음은 모듈별 기능을 나타내며 <그림 3>은 구현된 화면을 나타낸다.



<그림 3> 게임 실행 화면

- 1) **jgame** : 모든 게임 objects의 생성과 삭제, 생명과 총들을 처리하고, 게임과 사용자간의 이벤트를 담당한다.
- 2) **jstatus** : 게임의 상태를 화면에 유지시켜준다. 전 문가 즉, 고득점 자를 화면에 보여주고 상태를 화면에 나타내는 역할을 담당한다.
- 3) **jlevel** : 레벨에 맞는 맵과 속도등을 관리한다.
- 4) **jball** : 게임의 공들을 관리, 담당한다..
- 5) **jnpill** : 게임에서 나오는 아이템을 관리해준다.
- 6) **jbullet** : 게임에 사용되는 패들이 어느 특정 아이템을 먹었을 경우 총이 나갈 수 있게 해준다.
- 7) **jnpaddle** : 게임에서 마우스로 움직일 수 있는 패들을 담당한다.
- 8) **jnblock** : 게임 화면에 나오는 블록을 관리하기 위한 클래스이다.
- 9) **jnbuffer** : 3중의 버퍼를 정렬하고 인플리먼츠 하기

위한 클래스이다. 업데이트 항목등을 관리한다.

- 10) **jnmovingobject** : 움직임을 담당하는 클래스이다. 패들, 공, 등의 움직임을 처리한다
- 11) **jnobject** : 벽돌깨기 게임의 기본이 되는 오브젝트이다.
- 12) **jnimageserver** : 게임에 사용되는 이미지들을 관리해준다. 이 클래스를 로드 함으로써 로딩 시간을 단축 시킬 수 있다.
- 13) **jnsoundserver** : 이 클래스를 로드하면 사운드를 처리한다.

5. 결론

벽돌깨기 게임은 모듈 기반의 구조를 가지고 있으며 애플릿 프로그램이기 때문에 웹 상에서 언제든지 실행 가능하고, 조작법 또한 간편해서 누구든지 처음 하는 사람이라도 쉽게 게임에 적응 할 수 있다는 장점이 있다.

그러나, 온라인 상에서 다른 사용자들과 함께 즐길 수 없다는 점은 앞으로도 이 게임의 발전 방향을 제시해 주고 있다. 앞으로 서버-클라이언트의 통신모듈의 개발로 온라인 상에서 다른 사용자들과의 대전 모드로의 개발과 메신저 기능도 추가함으로써 온라인 아케이드 게임으로 만들어 볼 계획이다. 지금은 이와 같은 사항만 제시 할 뿐이지만 앞으로 훌륭한 온라인 아케이드 게임으로 활용 될 수 있을 것이다.

참고문헌

- [1] 조성삼, 정문경, 온라인 게임 개발 현황, 한국 정보처리 학회, VOL.99, No.03, pp0024~0033, 2002. 05.
- [2] 최재영, 최종영, 유재우, 프로그래머를 위한 JAVA2 총론 과학 출판사
- [3] 이재찬, 박성근, 최준영, 배재학, 핸드폰에서 할 수 있는 모바일 게임의 설계 및 구현, 정보처리학회 2002년 추계 학술대회, VOL.09, No.02, 2002. 10.