



유용성 테스트(Usability Testing): 두려움을 직면하고 그것을 사랑하는 법을 배우다.

작성자: 크리스 비거스 [Chris Viggers](#) [Game Design]
가마수트라 등록일 2011년 7월 14일

[블리츠 게임 스튜디오 개발 디렉터 크리스 비거스는 유저 테스트가 스튜디오 키넥트 (Kinect) 주제들의 두 가지와 함께 플레이어의 상호관계에 결함을 어떻게 발견하는지 그 방법을 설명하고 있다. 그리고 그 결함들이 개발팀의 추측에 어떠한 도전을 주었는지 설명한다.]

그것은 인기가 있다. 그리고 유행어이다. 그것은 성장산업이고 많은 개발자들은 그것을 두려워하지만, 유용성 테스트란 정확히 무엇일까. 그리고 어떻게 그것은 더욱 강력한 엔터테인먼트를 창조하기 위해 사용되어질 수 있을까?

개발자의 시각으로 본다면, 그것의 핵심에는, 유용성 테스트는 디자이너들, 아티스트, 코드를 만드는 사람들, 애니메이션 만드는 사람들, 그리고 관리자들에게 그들이 무엇을 잘못하고 있는지 말해주는 내용이 있다. 그것은 고통(정신적 혹은 육체적)의 어떤 종류를 피하기 위해 시도하는 인간적 본질이고 심지어 그러한 피드백이 더 좋은 결말을 이끌어 낼 수 있을 때, 우리 중에 많은 사람들이 비평을 받아들이려고 분투하고 있다.

본질적인 인간 본성의 문제에 더하여, 이 게임 산업은 심지어 이전보다 더 급속하게 변화하고 있는 중이고, 우리의 창조물은 더욱 더 많은 사람들에게 그리고 심지어 더욱 다양한 인구 통계에 의해 경험되어지고 있는 중이다. 우리의 게임에 그들의 소중한 시간을 사용하는 동안에 그들이 원하는 게임을 하는 방법으로 각각의 그들 자신의 기대를 가지고 있다.

왜 유용성 테스트를 모두 하려 할까?

이것을 읽은 어떤 프로듀서들은 시험을 증언할 것이므로, 당신이 그 게임으로 성취하길 진정으로 원하는 것을 성취하기 위해서는 당신은 결코 충분한 시간과 충분한 돈, 혹은 충분한 자원이 없다고 예민하게 항상 깨달으며 개발을 할 것이다.

당신의 생산 파이프라인으로 또 다른 프로세스가 추가되는 것은 그 팀과 상부 관리팀에게는 두 가지 모두 어려운 판매가 될 것이라는 것이다. 특히 그 프로세스가 플레이어들이 관심을 갖고 있는 문제들을 고치는 재작업을 야기할 가능성이 매우 높을 때에 더욱 그렇다.

그러나 진실은 당신의 게임이 결국에는 좋아할 게임인지 아닌지 곧 테스트될 것이고, 그 테스트는 매트릭스판의 점수 혹은 리뷰의 형태로 나올 것이다. 개발하는 동안에 만약 당신이 그러한 마지막 유저들의 피드백과 유용성 결과를 무시한다면, 당신은 당신의 게임과 평판에

공공연히 치명적 손상을 주는 나쁜 리뷰와 피드백을 받을 진짜 위험에 빠져드는 것이다. 이 사실을 진정으로 직면하고 이것을 당신의 개발 파이프라인과 함께 통합시키는 것이 성공에 이르는 핵심이라 하겠다.

우리 스튜디오 디자인 디렉터, 존 내쉬(John Nash)는 이미 여기서 매년 광범위하게 변하고 있는 게임을 위한 장치들을 입력하는 방법에 관하여 이야기를 해왔었고, 앞으로도 계속 할 것이다. 이것은 새로운 목표가 되는 거대하고 다양한 청중과 합쳐진다. 즉, 우리가 개발단계에서 훨씬 일찍 플레이어 입력을 수용해야 할 필요가 있을 뿐 아니라, 또한 우리는 이러한 입력을 수용하는 업무 흐름을 적용하는 것을 배워야 할 것이다.

이번 기사에서는, 어떻게 우리가 블리츠 게임 스튜디오(Blitz Games Studios)에서 이것을 해왔는지 실용적인 예시들로 개요를 서술할 것이다. 이번 경우에는 박스 제품 개발에 대한 초점을 맞추기 보다는 오히려 매우 다른 도전으로 여겨지는 소비자를 직접 대상으로 하는 개발에 중점을 두려고 한다.

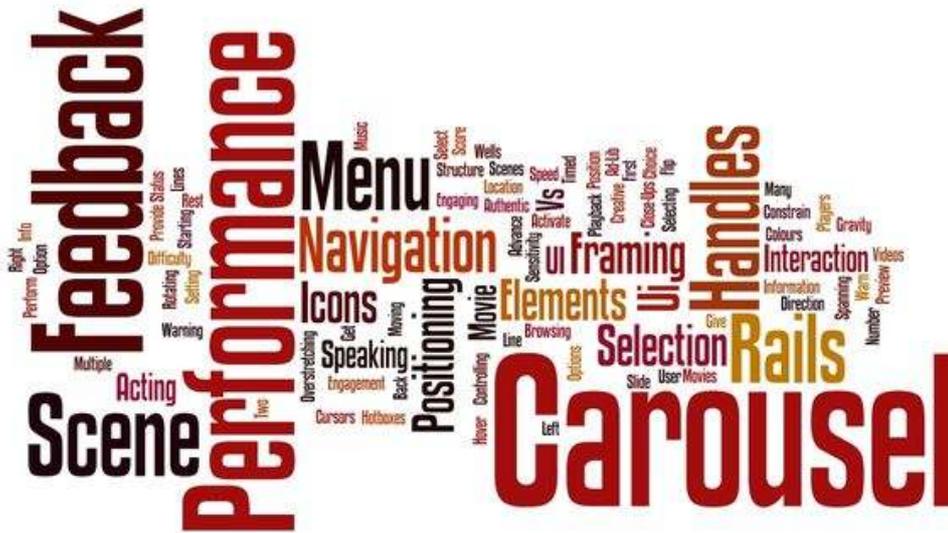
데이터의 가치

유용성 테스트는 확실히 과학적 방법론에 그 뿌리를 두고 있으며 게임을 경험한 유저들로부터 생산되어진 데이터에 의지하고 있다. 그것은 플레이어가 실제로 그 게임에 대해 어떻게 생각하는지를 독립적으로 보이는 것이다. (혹은, 그 게임에 대해 그들이 생각하는 점) 그것은 논쟁거리가 없는 어려운 통계 데이터와 그 플레이어가 실제로 그 게임을 하는 법을 입증할 것이다.

데이터 수집이 단순히 중요하다고만 하는 것이 아니다. 즉 유용성 테스트의 가치를 이해하는 것은 그 개발에 있어서 그것의 실용성을 최대화시키는 것을 가능할 것인지가 매우 논쟁적이다. 우선적으로 살펴보아야 할 점은 당신의 게임 경험과 핵심적 기둥이 되는 비전에 대항하는 그 데이터를 수집해서 쌓을 것인가의 문제이다.

예를 들면, 어떠한 분야에서는 사용자가 좌절되었는가, 오로지 마지막으로 퍼즐을 해결하여 우쭐대고 있는 것일까, 당신이 진정으로 최대화시키고 싶어 하는 것이 있는가? 혹은 특정분야에서만 그들은 두려워하고 그러한 염려 때문에 그 환경으로 인해 움직임이 느려지고 있지 않은가? 아니면 그들은 지루해졌거나 가야할 길을 잃어버린 것일까? 나는 지금 당신과, 림보(Limbo), 그리고 죽음의 공간(Dead Space)을 보고 있는 중이다!

특정한 유저 테스트가 우리의 작년 프로젝트 중에 하나였던 유스타 2 (Yoostar 2)에 도움을 주는지의 방법에 관하여 더 구체적인 예시를 제공하기 위하여, 테스트의 6개가 넘는 세션을 통하여 40개의 개별 항목이상에서 발견되었고, 그 플레이어를 위한 최대한 접근성과 경험을 보장하는 관심을 더욱 필요로 한다는 것을 발견하게 하였다.



분별해낸 40개 개별 항목과 해결되어진 유스타 2

심지어 모든 수집된 이러한 데이터는, 팀은 각 이슈에 뛰어들지도 고치기 위하여 시도하지 않은 것이었다. 각 분야에서 평가되었다. 그 비전(vision)의 핵심 기둥에 대항하여 평가하였고 플레이어가 게임을 경험하는 동안에 느끼길 원하는 것에 대해 평가하였다.

이러한 변화로 인한 결과는 게임이 더욱 접근할 수 있게 하거나 더욱 복잡하게 할 것이다. 또 용어의 변화는 전 세계적으로 이해되어질 것이다. 또 이러한 변경사항들은 게임 루프의 시간대 혹은 유저의 행동에 권리를 주는 능력으로 보이는 핵심적 측면에 영향을 줄 것이다. 그 게임의 어떤 부분을 고치기 전에, 심지어 당신 바로 앞에 있는 최고의 전문가의 피드백을 받기 전에 이러한 모든 질문은 당신이 스스로 자신에게 물을 필요가 있다.

비록 당신에게 돌아올 정보가 중요할지라도, 또한 당신이 받은 결과물의 일부는 일반적인 솔루션을 위해 더욱 필요하다는 점을 기억해야할 가치가 있다. 다른 말로 표현하자면, 사람들에게 혁신적인 기능을 개발함으로 일으키는 문제들의 일부는 아직 사람들에게는 낮익지 않은 것들이다. 사람들의 기대에 접근하는 것은 훌륭한 것이지만, 독특하고 혁신적인 아이디어들로 사람들을 놀라게 하는 것은 또한 달성하기에 고분 분투해야 얻을 수 있는 것이다.

유용성테스트의 결과물은 모두 존중 받으므로, 그것을 복음처럼 대우할 필요는 없다. 뿐만 아니라 당신의 본능과 시야를 신뢰해야 한다.

빠르게 도전한 것이 아니면, 가능한 일찍 테스트 하라,

유스타 2 프로젝트를 시작했을 때, 우리는 코드 한 줄을 쓰기 전에 더 많은 정보를 필요로 한다는 것을 알고 있었다.

(Kinect을) 아직 경험하지 않은 시대가 아닌 아주 새로운 하드웨어를 유용하게 하는 게임일 뿐 아니라, 그 게임을 영화장면에서 플레이어들이 스스로 주인공 역할을 하는 능력을 주기 때문이다. 청중들에게는 아주 신성하고 새로운 것이었다.

우리 스스로 물었던 첫 번째 질문은 “ 선택한 타겟층에게 호응을 받을 수 있는 게임의 핵심적 개념이 있는가?, 그 개념으로 그들이 기대하는 것은 무엇인가? 그리고 어떻게 사람들이 키네트(kinect)와 상호작용하는 것을 기대할까?” 였다.

그 추가적이고 중요한 문제가 그 게임의 외형적 스타일을 꾸미는데 피드백을 갖도록 하였다. 이러한 질문들에 대한 오로지 한 가지 방법이 있었고, 그 게임이 목표로 하는 사람들에게 질문하고자 하는 것이었다.

실제로 게임 경험을 즐기고 향유하는 것에 관해서, 우리는 즉시 유저 경험과 유용성 테스트 분야에서 전문성을 가진 회사인 버티칼 슬라이스(Vertical Slice)와 계약을 맺었다. 게임 흐름과 메뉴가 어떻게 될지에 관한 우리의 생각을 파워포인트 프리젠테이션으로 만들어 그들에게 보냈다.(코드가 없이 평이한 슬라이드를 담은 것이었다.)

우리는 또한 우리의 이상적인 목표로 하는 청중을 택하였고 파워포인트 “플레이”를 하면서 제스처어를 사용하며 콜트roller는 사용하지 않고 서서 보여주었다. 그 결과는 경직되어 있었다. 플레이어들의 기대의 개관은 아래와 같다.



기대되고 신뢰할 수 있는 몸짓(제스처)을 위한 내부적 집중 테스트

그 게임 기능에 대하여 우리가 초기의 집중하였던 테스트에서는, 수많은 흥미로운 사실들이 곧바로 명확하게 알 수 있게 되었다.

- 경력 모드(Career mode)는 1인용 플레이 게임을 하기 위하여 매우 동기를 부여하는 이유로써 만장일치로 보여지게 되었다.
- 플레이어들은 정확하게 그 장면을 복사하기 위해 그 선택을 원했고, 창의적이고 설명하는 공연이 되길 원했다. 그러나 결과에 대한 그들의 기대치는 다양하였다. 정확하게 복사하길 원했던 플레이어들은 어떠한 점수를 원했다. 반면에 애드립을 하는 것을 원했던 플레이어들은 희화된 모방 작품을 창조함으로써 단순하게 동기부여가 되고 있었다.
- 점수들은 또한 다시 플레이할 것을 매기는 가장 중요한 가치가 될 것으로 보고되어졌다. 플레이어들은 그들이 일인용 게임 모드를 하고 있을 때 점수가 향상되고 있는 것을 알 수 있기를 원했다. 또는 멀티플레이 모드에서 그들의 친구들을 이기고 있는지 알기 원했다.
- 더 나아가 필름 클립을 위해 DLC의 가능성이 더 잘 수용되어졌다.
- 플레이어들은 외적인 스타일의 일부가 가상의 할리우드 스타일로 될 수 있기를 기대하였다.

특이하게도 키네트를 사용하는 법에 대한 사람들의 기대에 관하여, 조사된 내용은 이러하

다. :

- 사람들은 전통적인 패드 네비게이션 보다 훨씬 느린 컨트롤링 커서를 찾았다.
- 사람들은 많은 다른 선택들 보다는 스크린 상에서 몇 개의 단순한 선택사항을 기대하고 있었다.

우리는 이러한 결과들을 팀에 가져갔고 게임 흐름과 게임 디자인을 만드는데 정직하게 그 정보를 염두해서 일을 착수 하였다. 또다시, 생각해보면 이 정보는 상식적인 것으로 보여 지지만, 이 당시에는 이러한 요점은 목표로 하는 청중과 정말 연관이 있는 무언가 창조하는데 매우 가치있는 것이었다.

두 개의 세계가 충돌 할 때 (안할 때)

나는 항상 게임을 만드는 세상에서 게임 개발자들이 가장 나쁜 사람들이라고 농담을 하였다. 우리는 객관적 판단을 하는 것이 어렵도록 하는 것에 아주 열정을 가지고 있고, 그 프로세스에 매우 가깝게 있다. 만약에 그것이 불가능하다면, 우리 안에 거품이 있을 때이다. 우리가 거의 완벽한 방법으로 열정의 모든 중요한 밸런스를 설명하기 위해 유용성 테스트를 발견하였다. 이것의 단순한 예로는, 유스타(Yoostar2)에서 키네트 'rail'(철도) UI 요소들 중에 하나의 방향을 지시하는 것을 결정하였을 때, 우리는 하나의 간단한 화살표를 사용하였다. :

아주 완벽하게 간단하지 않은가? 우리가 UI를 테스트 할 때까지만 해도, 우리도 그렇게 생각했었다. 유저의 90퍼센트는 버튼을 누르는 것을 원했다. VCR에 있는 플레이 버튼 같이 똑같이 생긴 것을 말한다. 게임에서 그 첫 번째 버튼과 열 사람들 중에 아홉은 지나간 첫 번째 화면을 얻을 수 없을 것이다. 바로 그것이 문제이다.

여러 주에 걸쳐 게임 안에 이 문제가 있었지만, 스튜디오에 있는 사람들 가운데 아무도 그것에 대해 질문하지 않았다. 키네트가 작동했던 법을 우리 모두 알고 있었고 “ 이것은 rail이어야 하니까 당신이 오른쪽으로 당신의 손으로 휘둘러야 한다.” 고 생각했다. 우리 자신에 대항하여 키네트의 모든 자세한 기술적 지식과 즉시 게임이 작동하는 법을 생각하였다.

이것은 간단한 예이고 고맙게도 놀랍도록 간단하게 팀이나 그 프로젝트 스케줄에 작은 영향을 주어 문제를 고칠 수 있는 것이었다. 그러나 진짜 유저의 관점에 대항하는 게임 개발자들의 생각의 위험성을 그저 아름답게 설명하고 있는 것 같다.

피드백? 최선을 다하기

진정으로 유용성 테스트의 중요한 요소와 많은 사람들이 간과하고 있는 당신의 팀, 특히 당신의 게임 디자이너들에게 준비할 필요가 있는 것은 정기적으로 외부에서 피드백을 받는 것이다. 열린 대화와 피드백, 열린 마음과 전문가적인 태도를 수용하고 지지하는 팀 문화를 보장함으로써, 익명의 외부에 있는 제 3자 회사가 추가적 업무 리스트로 작성한 것으로 여기지 않고, 심지어 이렇게 피드백 받기를 환영할 수 있을 것이다. 우리는 사람들에게 피드백 받는 것을 격려하고 게임 비전을 설립하는 것과 관련된 데이터 사용하는 것을 배우기를 격려할 수 있을 것이다.

많은 팀 멤버들이 그 실제적 유용성 세션에 자신 스스로를 포함하는 것은 구매 결정을 장려하는 훌륭한 방법이고, 또한 스튜디오 안에서 그 피드백 프로세스에 대해 열광하는 옹호하는 지지자들과 대사자들을 만들어 낼 것이다.

물론 어려움 곡선에 따라 게임을 테스트 영역이 몇몇 미묘한 문제를 제기할 수 있다. 실제

플레이어들이 게임 기술을 쌓는 것을 기대하는 게임 흐름 안에서 더 나아가 테스트를 하는 것이 중요하다. 그래서 어려운 매트릭스를 신뢰하고 실제 유용성 문제와 어려움의 문제 사이에서 이 둘을 명확히 구별해내야 한다. 그 도전의 수준에 따라서 당신의 게임이 일반적인 상품이 되는 것을 피하기 위해 이런 점들을 인지하도록 당신의 디자이너들을 훈련하는 것이 매우 중요하다.

광택내기의 가치

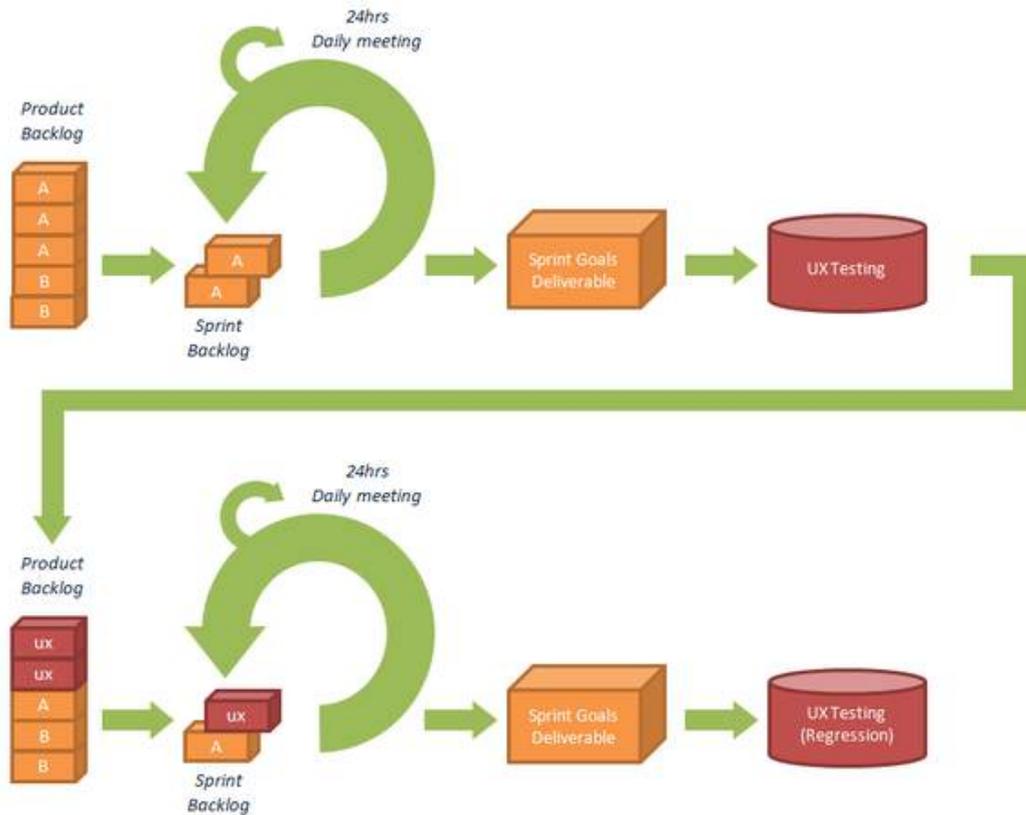
마지막 게임 자산으로 게임에 광택을 주어 세련되게 하는 것은 가치 있는 것임을 우리 모두 알고 있으나, 더불어 기능성을 테스트할 때, 당신이 테스트 하는 영역에 윤을 내는 것이 그 기능이 성공할 수 있는 핵심이다.

이것은 키네트와 같은 모션 컨트롤러(motion controller)와 함께 중요하다. 그것은 모두 너무 쉬워서 전문적으로 당신이 테스트 할 수 있는 기능적 패스(pass)에 기능을 얻게 할 수 없다. 그러나 만약 기능이 음향 효과, 그래픽 효과, 컨트롤, 음악 등과 같이 기본적으로 윤 내기가 부족한 것이라면, 대부분의 유저는 해당 기능에 문제점을 가지거나 거부 할 것이다.

그다음에 이것은 “실패” 한 기능으로 보여지는 데이터를 제공하며, 그 팀은 그 기능을 고치려 하기 위해 더욱 복잡한 솔루션을 내기 위해 열심히 일을 하거나 다함께 그 기능을 버리기로 할 것이다.

(이것은 다시 때때로 개발자의 생각과 플레이어의 기대 사이의 위험한 이분법을 제공하게 한다.)

만약 당신이 빠른(Agile) 개발 방법론을 사용하고 있다면, 앞에 닦친 것을 최대한 철저하게 생각하는 한 당신의 프로세스에 이것을 통합하는 것은 매우 간단하게 할 수 있다. 상품기능(Feature A)을 완성하자마자 그것을 인증하는 것은 비교적 쉽다. 최소한 당신이 그 기능 테스트를 한다고 했을 때에 그러하다. 일단 그 기능이 테스트를 받으면, 당신은 그 결과물을 받고 프로젝트를 위한 당신의 상품 수주잔량(backlog)으로 곧바로 이것을 추가하게 된다. 그리고 나서 그 다음의 스프린트(sprint)를 하는 동안 관련 항목을 처리하면 된다.



유저 테스트 피드백을 스프린트 계획으로 통합시키기

이 방법에서 전통적인 병목현상은 발생하는데 팀이 같은 기능의 다음 단계로 이동하고자 할 때 (계속 반복하며) 생긴다. 그러나 유저 테스트 단계에서는 아직 리포트를 받지 못하고 있는 상황이 된다.

기능 A 사용자 테스트 결과가 기록되고 논의 되고 있는 동안, 대부분의 경우에 이것을 완화시키는 간단한 방법으로는 스프린트 팀이 새로운 상품 기능(기능 B)로 이동하게 만드는 것이다. 그리고 나서 다시 마지막 스프린트를 위해 기능 A로 되돌아 가서, 해당 스프린트에 관한 백로그에 직접 사용자 테스트 결과를 통합시킨다.

반복 및 확인

개별 사용자 테스트 세션을 개최하는 것은 유용하지만, 특히 당신이 만든 변경사항을 테스트하는 전용회귀 세션을 후속으로 갖는 것도 중요하다.

단순히 실제 문제를 악화시키는 것을 찾겠다고, 제한된 피드백과 그 문제에 대해 정확히 수정한다고 추정하는 생각에 기초하여 변경사항을 만드는 함정에 쉽게 빠질 수 있다. 또 다시, 이것은 그 자체 기능을 통해 어떠한 특정 기능에 광택을 내는 분량으로 인해 발생할 수 있으며 그 가능성도 언급되어야 할 것이다.

환상적인 애완동물(Fantastic Pets) 키네트 게임으로 이러한 예를 들수 있는데, 그 게임은 공을 애완동물에게 던져주는데 문제가 발생한다고 많은 젊은 층의 사용자들로부터 처음 테스트 결과에 대한 보고를 받았었다. 특정적으로 그 문제는 공을 집어 들고 놓는데 문제가 있었고 또한 공은 기대되는 시점에 손에 공에서 공을 떨어뜨리는데도 문제가 있었다. (이것은 끈끈한 공 현상-Sticky ball syndrome이라 부른다)

우리는 우리가 사용했던 코드와 시스템에서 또다른 것을 살펴보았는데, 사용자가 젊은 사람 이든지 더 나이가 많은 사람이든지 관계없이, 그것은 모든 사용자들이 키네트 오른쪽에서 제스처(몸짓) 인식을 하는데 매우 까다로울 수 있었기 때문이었다.

우리는 시스템에서 몇 가지를 수정하려고 했고 그것을 게임 팀과 함께 내부적으로 테스트 하였고, 특히 공을 가지고 시작하는 문제에 있어 향상된 것을 보게 되었다. 그러므로 우리는 다음 테스트 세션으로 가기 위하여 준비되어야 부분이 향상되어졌음을 믿게 되었다.

두 번째 테스트 세션에서는 우리는 플레이어들이 여전히 그 공을 집어 올리는데 문제가 여전히 문제가 있는 것을 보고 두려움에 떨게 되었다. 더 심도 있는 조사를 한 후에, 우리는 그 문제가 실제로 플레이어의 기대에 기반 되어진 것을 깨닫게 되었다. 플레이어들은 애완 동물이 그 공을 플레이어에게 되돌려주기를 기다리기 보다는, 공을 차례로 직접 공을 던질 수 있어야 한다고 기대하고 있었다.



애완 개가 공을 잡으러 가는 동안에 고정된 자세로 카메라가 있기 때문에 플레이어들은 이렇게 생각하게 되었다. 그래서 그들은 다른 공을 또 던질 수 있다고 기대하게 된 것이다. 공이 되돌아오는 동안에 그 개를 따라가는 영화적 카메라를 간단하게 추가하여, 플레이어가 또 다시 공을 들어 던질 수 있기 전에, 플레이어가 화면이 기본 카메라로 되돌아 올 때까지 기다릴 필요가 있다고 이해하게 하였다.

이러한 회귀 세션이 없었다면, 우리는 우리가 만든 코드 수정에서 문제를 고칠 수 있고, 실제 근본 원인을 해결하지 않고서 그 게임을 출하 할 수 있는 것을 가정할 수 있게 되었다.

요약

블리츠에서, 우리는 게임개발을 위한 빠른(Agile) 방법론을 모두 읽고 실행하였다. 그것은 민첩한 과정(Agile process)으로 정기적인 시간간격을 두고 유용성 테스트를 통합하는 것이

우리에게 다음 단계로 쉽게 넘어가게 해주는 것을 알았다. 그리고 미래에 최대화하기 위하여 그것을 계속해야 하는 것을 깨달았다. 게임 개발 내에서 유용성 테스트는 그 역할이 비교적 새로운 과학이지만, 우리는 유용성 테스트가 끊임없이 성장하고 확대되는 고객을 위해 성공적인 엔터테인먼트를 창조하는 최상의 잠재력을 제공하고 있다고 믿고 있다.

당신의 파이프라인으로 유용성 테스트의 고려하는 것은 더 많은 일을 하는 것이란 의미가 되는가? 그렇다. 당신의 프로젝트에 더 많은 비용이 추가되게 하는가? 물론 그렇다. 최종결과가 당신 회사와 팀에 가치가 있는가? 의심할 바가 없다. 블리체 게임 스튜디오에서 우리의 열정은 비평적으로도 상업적으로도 모두가 즐길 수 있는 성공적인 게임을 창조해내었다. 그리고 문제를 삼는 사람들과 함께 우리의 게임들을 테스트 하는 것이 미래에 이것을 달성하기 위한 열쇠라고 믿는다.

스트루안 로버트슨(Struan Robertson)과 스티브 스탱스(Steve Stopps)에게 이 기사를 쓰는데 많은 도움을 주셔서 감사하다는 말을 전하고 싶다.