



※ 본 아티클은 CMP MEDIA LLC와의 라이선스 계약에 의해 국문으로 제공됩니다

관찰은 게임 설계하기

Jason Bay

가마수트라 등록일(2009. 7. 8)

(http://www.gamasutra.com/view/feature/4072/designing_games_that_dont_suck.php)

[상세한 설계 논문에서, Griptonite Games 의 Jason Bay 는 게이머의 게임 내 무기, 오브젝트, 퍼즐 사용을 분석하기 위해 고전적인 유용성 개념을 사용하며 이것은 어떻게 하면 모든 사람이 더 나은 비디오게임을 구축하도록 할 수 있는지를 제시한다.]

설계자의 가장 중요한 역할은 게이머를 위한 독특하고도 흥미로운 도전과제를 만들어 내는 것이다. 그러나, 쉬워야 하거나 게임플레이 중에 보이지도 않는 부분 때문에 게이머들이 예기치 않게 어려움을 겪는 것을 너무나 자주 발견한다.

이것은 사용자 인터페이스의 기능이거나, 파악하기 쉬워야 하지만 어쨌든 그렇지 않은 게임 월드 내 상호작용일 것이다. 이러한 일들이 정상적인 진행을 방해할 때 게이머들은 당황해서 게임 하는 것을 완전히 중단할 수 있다.

개발자인 우리는 그들이 더 나은 형태의 사용성에 접근하기 까지, 이러한 문제들을 일찍 발견하여 문제가 되는 대상을 뜯어고치기 위해 노력한다. 그러나 “이거 형편없는데”에서 “우와!”로 가려면 많은 작업과 반복이 필요하며 확립된 프로세스보다는 철저한 반복에 의해 목표에 도달하는 경우가 많다.

직감에 따라 운영하기 보다는 게임이 출시 되기 전에 끊임없이 잠재적 문제점을 예측하고 그에 맞춰 조정하기 위해서는 게이머들이 게임과 상호작용하는 방식을 이해할 신뢰성 있는 방법에 도움을 받을 수 있을 것이다.

고전적인 사용성 책인 일상용품의 심리학(The Psychology of Everyday Things)에서 저자 Donald A. Norman은 사람들이 과제를 수행하는 방법의 이면에 있는 철학을 고찰함으로써 사물의 사용성을 평가하는 시스템을 기술한다. 이 논의는 문이나 타이머가 있는 라디오와 같은 실제 세계의 사물에 방향이 맞춰진 것이긴 하지만 각 게임 오브젝트가 정말 직관적으로 이해할 수 있는 것인지 그리고 게이머들이 익혀서 사용하기에 쉬운지 알아 내기 위해 그의 방법론을 우리의 게임 월드에 적용할 수 있다.

이 논문에서는, 좀 더 직관적인(또는 좀 더 도전적인!) 오브젝트, 퍼즐, GUI 컨트롤 및 기타 게임 월드 상호작용을 구축하는데 이 아이디어가 어떻게 적용될 수 있는지 살펴본다.

행동의 일곱 단계

각각의 개인은 사물을 사용할 때 그가 목표를 형성하는 시간부터 그것이 성공적으로 성취되었는지 평가할 때까지 일곱 단계의 행동을 경험한다고 노먼은 정의한다.

- 1 단계: 목표를 형성하라
- 2 단계: 의도를 형성하라
- 3 단계: 행동을 명확히 하라
- 4 단계: 행동을 실행하라
- 5 단계: 세계의 상태를 인식하라
- 6 단계: 세계의 상태를 해석하라
- 7 단계: 결과에 대한 평가

이 목록을 순서에 따라 머리 속에서 훑어 보면 어떤 게임 엔티티의 사용성이라도 평가할 수 있으며 게이머가 각 단계에 참여하는 것이 가능한지 검토할 수 있다. 게이머가 이들 단계 중 하나 이상을 수행할 수 없을 것 같으면 그것은 설계를 마치기 전에 해결해야 할 사용성의 문제를 나타내는 적신호이다. 거꾸로 말하면, 어떤 것을 생략하느냐에 따라 게이머에게 흥미로운 도전과제를 만들어내기 위해 한 단계를 일부러 생략할 수도 있다.

일곱 단계에 대한 상세 설명

게임 엔티티의 사용 편리성을 분석하기 위해 이들 단계를 점검하는 법을 설명하는데 도움을 주기 위해 간단한 예를 살펴보자. 게임 내의 폭발이 너무 시끄럽다고 게이머가 생각한다고

하자(그는 시끄러운 게 좋지만 옆방에서 자고 있는 딸을 깨울 위험이 있다). 게임의 세팅 메뉴를 분석한다면, 각각의 행동 단계에서 이 게이머는 어떤 일을 하거나 생각하고 있는 것일까?

1. 목표를 형성하라: 이것은 사용자가 성취하고 싶어하는 무엇보다 중요한 최종 목표를 형성하는 단계이다. “여기 세팅 메뉴가 있구나. 폭발음 볼륨을 줄이는 옵션을 제공할 것 같다.”

2. 의도를 형성하라: 이 단계에서 사용자는 목표를 향해 끊임없이 나아갈 방법에 관해 파고 들기 시작한다. “‘이펙트’라는 제어장치가 있네. 폭발음에 영향을 주기 위해 어떤 식으로 조종할 수 있을 거야.”

3. 행동을 명확히 하라: 이 단계에서 사용자는 할 수 있는 특정 행동을 확인한다. ‘이펙트’ 컨트롤은 수평 트랙에 있는 작은 손잡이 같다. 볼륨을 변경하려면 손잡이를 잡고 트랙을 따라 드래그 할 수 있을 것 같다. 레일의 오른쪽에는 큰 스피커 아이콘이 있고 왼쪽에는 작은 스피커 아이콘이 있으니 볼륨을 줄이려면 왼쪽으로 드래그 해야겠다.”

4. 행동을 실행하라: (포인터로 작은 손잡이를 잡고 트랙을 따라 왼쪽으로 이동한다.)

5. 세계의 상태를 인식하라: 이 단계에서 사용자는 자신이 시스템을 변경함으로써 생긴 피드백을 처음 경험한다. “손잡이를 왼쪽으로 드래그해서 저기다가 풀었다. 게임에서 샘플 음향 효과가 나오네.”

6. 세계의 상태를 해석하라: 이것은 사용자가 전 단계에서 받은 피드백의 의미를 해석하는 단계이다. “지금은 손잡이가 작은 스피커 아이콘에 가까이 있고 손잡이를 움직이기 전에 게임에서 들었던 것보다 샘플 음향이 작아진 거 같으니까, 폭발음도 더 조용해 질 거라는 의미겠지.”

7. 결과에 대해 평가하라: “소리 볼륨을 낮추고 싶었는데 해냈다. 성공!”

일반적인 사용자라면 이 단계들을 잠재의식 속에서 몇 초 만에 진행하겠지만 각 단계를 순조롭게 진행할 수 없다면 혼란스러워하고 당황할 것이다. 그런 일이 일어나지 않도록 하는 것이 당신의 역할이다.

이 실례의 단순한 메뉴가 부정확하게 설계되어 있었다면, 하나 이상의 단계가 어떻게 실패할 수 있을지 몇 가지 예를 살펴보자

1 단계와 2 단계의 문제: 일전에 그냥 “SFX”라고 적힌 사운드 컨트롤이 있는 게임을 우연히 발견했다. 게임업계에서는 “음향효과”라는 용어를 이런 식으로 종종 줄여 쓰지만, 게이머가 이 업계에 종사하지 않거나 “특수 효과” 같은, 뭔가 다른 것을 의미한다고 해석한다면 어떻게 될까?

플레이 테스트를 하는 동안 한 사용자가 이 메뉴를 열어서 잠시 동안 보다가 “SFX”가 뭔지 내게 물어보는 것을 보았다! 사운드 컨트롤을 잘못 표시해서 게이머를 1 단계에서 방해했으므로 이것은 메뉴 설계 면에서 볼 때 실패작이었다. 사운드 컨트롤에 대한 바른 표시가 없다면 그는 “소리 볼륨 변경”이라는 목표를 형성할 수 없을 것이다.

게이머가 “소리 볼륨 변경”이라는 목표로 이미 무장하여 이 메뉴에 도달했다면 그는 2 단계에서 끔찍 못하게 됐을 것이다. 그 과제와 분명한 관련을 가진 것이 아무 것도 없는데 어떻게 소리를 줄이기 위해 컨트롤 중 하나를 조작한다는 의도를 형성할 수 있겠는가?

3 단계의 문제: 컨트롤이 부적절하게 설계되어 있다면, 게이머는 그것을 가지고 무엇을 해야 할 지 결정하기 어렵거나 불가능할 것이다. 수평 슬라이더 컨트롤에 “사운드”이라고 적힌 것을 본 기억이 있다.

그러나 거기에는 볼륨을 높이려면 어떤 방향으로 밀어야 되는지, 또는 “볼륨”이 과연 조절되는 속성인지에 관한 어떤 표시도 없었다(음향의 좌우 패닝 조종은 쉬웠을 것이다).

어느 쪽이 “높이는 것이고” 어느 쪽이 “낮추는 것인지” 시각적으로 보여주는 무언가가 없다면, 게이머는 그 간단한 조작으로부터 우리가 예상할 수 있듯이 쉽게 3 단계: 행동을 명확히 하리를 완수하는데 어려움을 겪을 것이다

4 단계의 문제: 조종장치로 무엇을 해야 할지 사용자가 일단 파악하더라도, 4 단계: 행동을 실행하리에 걸림돌이 될 몇 가지 장애물이 아직도 남아 있다. 몇 가지 실제 세계의 실례는 다음과 같다.

- 사용자는 키보드를 선호하는데 컨트롤은 마우스 입력만 허용하는 경우
- 사용자는 손가락이 큰데 터치 스크린 상의 컨트롤이 너무 작고 다른 컨트롤에 가까운 경우
- 사용자에게 장애가 있는데 컨트롤에는 적절한 필요한 옵션을 지원하지 않는 경우
- 사용자가 원하는 만큼 컨트롤이 정밀하지 않은 경우 (예: 사용자는 3 씩 증가시키기를 원하지만 컨트롤은 10 씩 증가시키는 것만 가능하도록 한 경우)

5 단계와 6 단계의 문제: 컨트롤이 사용자에게 피드백을 주지 못하거나(5 단계의 문제), 확실하지 않고 혼란스러운 피드백을 주는 경우(6 단계의 문제)는 흔하다. 우리가 제시한 세팅 메뉴의 실례에서, 슬라이더가 움직인 후에 볼륨 컨트롤에서 약간의 소리를 들려주지 않으면 어떻게 될까?

게이머는 볼륨이 줄어든 것을 눈으로 볼 수 있겠지만 오디오 샘플이 없다면 새로운 볼륨이 만족스러운지 알아보기 위해 게임으로 되돌아가야 할 것이다.

잘 설계된 우리의 보기 메뉴도 조금은 부적절할 수 있다. 메뉴의 소리 피드백이 예를 들어 총소리를 들려주는 것이라면 게이머는 새로 조정된 폭발음을 상대적으로 추측만 할 수 있을 뿐이다.

7 단계의 결과: 플레이어가 7 단계: 결과에 대해 평가하러에 게이머가 도달하는 때가 되면, 그를 위해 할 수 있는 일이나 그가 할 수 있는 일이 많지 않으므로 설계자인 당신은 성공하거나 실패한 것이다.

상호작용이 어떨든 간에, 사용자가 성공을 지각하도록 하기 위해 이전 단계에서 당신이 좋은 결과를 준비했다는 것을 확인하라. 그가 실패를 지각하면 왜 실패했는지 반드시 이해 시키고 상황을 바로잡기 위해 무엇을 할 수 있는지 알려주도록 하라.

[1] 왼쪽에서 오른쪽으로 읽는 문화는 자연스럽게 “오른쪽”을 “올린다”와 연관 짓겠지만, 오른쪽에서 왼쪽으로 읽는 언어에 메뉴가 로컬화되어 있다면 이 문화적 표준에 의해 제공되는 직관성은 더 이상 도움이 되지 않을 것이다.

게임월드에서의 사용성

노먼의 행동의 일곱 단계를 이용하면 게임 개발자인 우리는 정보에 입각한 설계 결정을 할 수 있다. 우리의 단순한 세팅 메뉴 예시에서도, 몇 가지 인터페이스 문제를 확인하고 되돌아가기 위해 이들 단계를 이용할 수 있었다.

덧붙여, 설계하는 동안 행동의 일곱 단계 모두를 적절히 다루지 못하면 프로세스가 결국은 게이머의 게임 진행을 방해한다는 것을 알게 되었다.

그러나 이 시스템의 적용가능성은 메뉴 UI 를 훨씬 능가하며 사실상 대부분 또는 모든 게임의 게임 월드 내 상호작용을 평가하는데 사용될 수 있다.

보기를 이용하여, 일곱 단계를 다루는 것이 1 인칭 액션 게임에서 설계를 결정하는데 어떻게 영향을 줄 수 있는지 잠깐 살펴본다.

게임 월드 내 액션: 플라즈마 폭탄을 던져라

1 단계: 자신이 플라즈마 폭탄을 가지고 있고 그것을 던질 수 있다는 것을 게이머가 아는가? (HUD에 그것이 표시되는가? HUD 아이콘이 명확하고 이해할 만한가?)

2 단계: 지금이 폭탄을 던질 적절한 때인지 아닌지 게이머가 이해하는가? (플라즈마 폭탄은 접근전에서 효과가 좋은가? 수중에서는 어떤가? 유효한 표적 위에서 공중 정지할 때 표적 십자선이 변경되는가?)

3 단계: 폭탄을 던지려면 어떤 버튼을 눌러야 하는지 게이머가 아는가? (개인지도 시간에 들었지만 잊어버렸는가, 그의 개가 사용자 지침을 먹어 치웠는가? 그렇다면, 게임에서는 맞춤형 사용설명서를 제공하는가?)

4 단계: 게이머는 이동이나 피하기와 같은 다른 중요한 동작을 하면서 실제로 맞는 버튼을 누를 수 있는가? 큰 컨트롤러를 사용하는 어린 아이라면, 그의 작은 손가락이 아날로그 스틱을 움직이는 동시에 플라즈마 폭탄 버튼을 누를 수 있는가?



5 단계: 플라즈마 폭탄이 정말 던져졌다는 것을 표시하는 만족스러운 피드백이 있는가? 폭탄이 공중에 날아가는 것을 그가 보면서 듣거나 보든지 듣든지 둘 중 하나를 할 수 있는가? 어둡거나 짙은 연기 속에서도? HUD 카운터가 변했는가?

게이머에게 폭탄이 없는 경우 왜 폭탄이 나타나지 않는지 알려주는 피드백이 있는가? HUD 상에서 번쩍이는 아이콘? “젠장, 플라즈마가 다 떨어졌잖아”라고 투덜거리는 그의 아바타?

6 단계: 플라즈마 폭탄이 작동했는가? (폭발이 있었는가? 다른 효과는? 적들이 부상당한 것이 눈에 보이는가? 그것이 물에 떨어져 눈에 보이게 솟하며 꺼지는가?)

7 단계: 플라즈마 폭탄이 이런 상황에서 사용하기에 성공적인 무기였는가? (이런 유형의 적에 대항하는데 효과적이었는가? 몹시 긴장하여 결국 게이머에게 부상을 입혔는가? 그것은 만족스러운 상호작용이었는가?)

이 게임월드 액션의 각 요소를 연결하면 노먼의 심리학 기반 접근법에는 문, 오븐, 타이머가 달린 시계의 기능을 훨씬 능가하는 적용가능성이 있다는 것을 더 잘 설명된다.

이것은 UI와 전투 뿐 아니라 퍼즐, 보스 전투, 게이머가 당신의 게임으로 할 거의 모든 상호작용에 적용될 수 있는 탄력적이고 유용한 발견적 도구이다.

도전과제 만들어내기

게임 설계자들은 게이머에게 상호작용이 좀 더 힘들도록 고의로 만들기 위해 의도적으로 일곱 단계 중 한 개 이상을 힘들게 만들 수 있다. 이것은 생략이 어떻게 좋은 난제를 만들어 낼 수 있는지에 대한 아이디어를 주는 귀중한 설계 도구이며, 여기 몇 가지 예가 있다.

3 단계를 생략하라: 많은 종류의 퍼즐은 게이머에게 목표(1 단계)와 의도(2 단계)를 제공하지만 3 단계: *행동을 명확히 하라*를 생략함으로써 퍼즐을 푸는 방법을 숨긴다. 일단 게이머가 비밀을 알아내고 나면, 도전과제를 완료하기 위해 퍼즐을 조작하는 것(4 단계: *행동을 실행하라*)은 별 일 아니다. 도전과제가 성공적으로 극복되었기 때문에 게임은 남은 단계에 대한 좋은 피드백을 제공할 수 있다

4 단계를 생략하라: 퍼즐의 또 다른 표준 형태는 목표(1 단계), 의도(2 단계), 수단(3 단계)을 제공하지만 속임수는 실행에 있다(4 단계: *행동을 실행하라*의 어려움)

액션/플랫폼 장르의 점핑 퍼즐에서 이것은 흔한 예이며, 게이머는 왜 그리고 어떻게 일련의 점프를 완수해야 하는지는 알지만 죽지 않고 거리와 타이밍을 실행하기는 어렵다는 것을 알게 될 것이다.

5 단계를 생략하라: 변경을 활성화했지만 게임 월드에서 무엇이 바뀌었는지 모른 채 게임을 한 적이 있는가? 그렇다면 그것은 게임의 설계자가 의도했든 아니든 변경이 실행된 후 당신이 *세계의 상태를 인식하도록* 도움을 준 어떤 피드백도 제공하지 않았기 때문이다. 이런 종류의 도전과제는 상황에 따라 좋을 수도 있고 나쁠 수도 있다.

6 단계를 생략하라: 위의 예를 이용해보자. 변경을 실행한 후 복도 저 아래 어딘가에서 길면서 굵고 낮은 삐걱거리는 소리를 듣는다면 어떻게 될까? 크고 무거운 돌로 된 물체가 움직인 소리 같았다.

이 때 당신은 게임 월드의 변화(5 단계)를 인지했지만 그것이 무엇을 의미하는지, 플랫폼이었는지 문이었는지 등 정확히 해석하지 못할 수도 있다. 이것은 6 단계: *세계의*

상태를 해석하러의 어려움과 같다. 새로운 상태를 해석하기 위해 더 많은 조사가 정당화된다.

머리의 지식과 세계의 지식

노먼이 발표한 또 하나의 중요한 개념은 “머리의 지식”과 “세계의 지식”의 차이이며, 사용자는 세계가 보여주는 것에 근거하여 무엇을 해야 할지 알거나, 스스로 그 상황에 떠올린 지식에 기초하여 무엇을 해야 할지 안다는 의미이다. 이것을 오해하면 의도하지 않게 게이머를 방해하는 상황을 만들 수 있기 때문에 어떤 지식은 이미 알고 있는 게이머에게 의지하고 어떤 정보는 그들에게 가르쳐야 하는지 결정하는 것은 중요하다

세계에서

인터페이스로부터 어떤 도움도 받지 않고 이미 알고 있기를 기대하는 것(또는 알아내도록 내버려두는 것)과는 반대로, “세계의 지식”은 액션을 수행하기 위해 게이머에게 필요한 정보를 게임에서 활발하게 주고 있다는 것을 의미한다. 달리 말하면, 참고를 위해 필요한 정보는 이미 게임 월드 안에 들어 있다.

예를 들면 GUI 에서 이런 저런 제품이 무엇을 하는지에 관한 지식이 다음의 몇 가지 방식으로 전달된다.

- 그것이 무엇을 하는지 설명하는 텍스트 라벨 제공
- “도구 도움말” 또는 그 밖의 문맥 감지형 도움말 제공
- 어떤 것을 어떻게 조작해야 하는지에 관한 시각적 단서 제공 -- 눌러야 할 것들은 버튼처럼 보여야 하고 밀어야 하는 것에는 그림 아이콘이 있어야 하며, 잡아야 하는 것에는 손잡이가 있어야 한다.

게임 월드의 실체에 관해서는, 특정 상황에 따라 지식을 집어넣는 무수한 방법이 있다. 몇 가지 예로는:

- RTS 에서는 구조물을 건축할 수 있는지 알아내기 위해 게이머가 클릭해서 각 유닛의 속성 시트를 읽도록 하지 않는다. 그 대신 유닛의 그래픽이 구조물을 건축할 수 있는 무언가처럼 *보이도록* 설계한다. 유닛이 손에 톱을 들고 있거나 벨트에 해머가 있으면 게이머는 바로 알 것이다.
- FPS에서는 어떤 핫 키가 산탄총으로 바뀌는지 게이머가 암기하도록 하지 않는다. 관련된 키 매핑을 가지고 HUD 상에서 사용할 수 있는 무기를 보여준다[2].

- 게이머는 그 영토에 정통하기 때문에 친절한 **NPCs** 는 복잡성을 제고한 힌트와 도움말을 줄 수 있다.

물론, 게이머를 위한 도전과제를 만들려고 할 때 게임 월드에서 지식을 빼내고 싶을 것이다. 그러나 게이머들은 접근 가능해 보이지만 그럴지 않은 암봉 같이, 그럴 의도가 없었던 것들에 도전 받는 경우가 너무 많고 그것은 그들을 좌절하게 하며 더 심각한 일은 이 때문에 당신의 게임이 거추장스럽게 투박하고, 사용자 친화적이지 않으며 또는 불필요하게 가파른 학습곡선을 가지고 있다고 인식하게 된다는 것이다. 필요한 곳에는 어디든지 적절한 단서를 게임 월드에 집어넣음으로써 게이머들이 게임을 이해할 수 있도록 도와야 한다.

[2] 이것이 당신의 아름다운 미니멀리스트 **UI**를 얼마나 볼품없게 만들 것이라 놀리고 있다면, 필요한 키를 그들이 일단 외운 후에 즉, 지식이 그의 머리에 들어 간 후, 그 표시장치를 숨기는 옵션을 제공할 수 있다. 그 동안에는 예쁘지는 않아도 편리한 리마인더에 당신에게 감사할 것이다.

머리에서

“머리의 지식”은 게임 월드가 활발히 또는 지속적으로 게이머들이 액션을 수행하는데 필요한 정보를 제공하지 않는다는 것을 의미한다. 그 대신 게이머들은 도움말이나 리마인더 없이 과제를 행하는 법을 직관적으로 알고, 기억하고 알아낼 것으로 기대된다

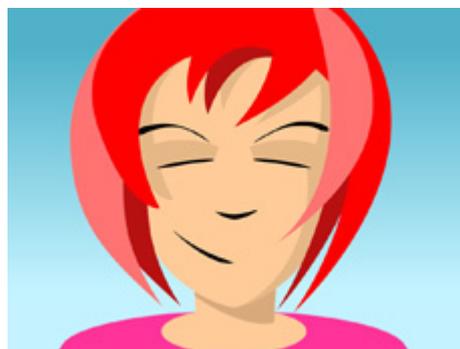
게임의 **GUI**에서는 다음을 행함으로써 게이머가 이미 알고 있는 지식에 영향을 줄 수 있을 것이다.

- 드롭다운 목록이나 스크롤바 같이, 게이머가 인지할 확립된 표준을 따른다.
- 일반적인 사용자가 이미 친숙해진 버튼이나 폴더 탭과 같은 실제 세계에 있는 어떤 것의 외형이나 움직임을 그것에 부여한다.
- 기억을 환기시키기 위하여 친숙한 그림/아이콘으로 컨트롤에 표시를 한다.

게임 월드 자체에서는, 게이머가 이미 알고 있는 지식에 의존하는 것은 위험하다. 알 것이라고 당신이 예상하는 것을 그가 모르고 있다면 중점과제를 수행하는데 어려움을 겪을 것이다.

많은 게임들은 개인지도 단계에서 필요한 정보를 플레이어의 머리 속에 “미리 담아두며” 이것은 좋은 출발이지만 말해 준 것을 게이머가 잊어버리거나 별 신경을 쓰지 않고 띄엄띄엄 본다면 문제가 생기고, 이는 불행히도 흔한 일인 듯하다.

현대의 많은 게임은 아주 복잡해서 게이머들은 많은 시간을 게임에 접속하고 나서야 필요한 모든 정보를 암기한다.



잊어버리면 다시 배우는 것이 어렵거나 불가능한 중요한 몇 가지 지식을 자주 기억하고 있으라고 요구하면 게이머들에게 좌절감을 일으킬 수 있다.

- “그래 그 과학자가 그 차단문과 조합이 되는 걸 말해줬어. 하지만 나는 지난 6월 이후론 게임을 안 했고 그가 뭐라고 했는지 잊어버렸어!”
- “이 탱크의 공격율에 관한 리마인더 하나 얻자고 유닛 통계 화면을 정말 클릭질 해야 되나?”
- “저 보스 괴물이 연속해서 수 십 번이나 나를 죽였어. 이거 뭐냐. 내가 뭘 잘못하고 있는 거지?”

이러한 문제에 대처하기 위해서는, 외울 필요성을 덜어주는, 게임 월드 안에서 이용할 수 있고 풍부한 지식을 제공해야 하며 그럴지 않으면 게이머의 기억을 자극하기 위해 중요지점에 리마인더를 준비해야 한다. 위의 사례에 도움을 줄 수 있는 몇 가지 아이디어는 다음과 같다.

- **NPCs** 가 게이머에게 준 중요한 정보를 저장할 방법을 제공한다(노트 화면, 퀘스트 메뉴 등등).
- 게이머가 유닛 위에 커서를 정지하면 영상 통계 요약 정보 팝업을 제공하거나 현재 화면상의 모든 유닛에 대한 전체 통계를 보여주는 사이드바를 제공한다.
- 게이머가 반복적으로 과제를 수행하지 못할 때를 감지해서 유용한 힌트를 주거나 기존 힌트를 좀 더 뚜렷하게 한다.

결론

모든 발달 주기에 대해, 게임 팀은 게이머들이 이해하고 실행하기 쉽도록 만들기 위해 게임 내 액션을 수정하고 재수정하느라 많은 시간을 보낸다. 그 문제들은 원하는 액션을 성공적으로 수행하는데 필요한 정보를 가지고 있지 못한 사용자들로 인한 경우가 많았다.

개발자들이 가능하면 많은 지식을 게임 월드에 집어 넣으려는 노력을 할 수 있다면 -- 특히 노먼의 행동의 일곱 단계 완수를 용이하게 하는 정보 -- 게이머들은 좀 더 쉽게 조종을 하게 되고, 혼란스러워 하거나 당황하는 횟수가 훨씬 적어지는 것을 알게 될 것이다.

이 단순한 설계 도구로, 좀 더 직관적인 상호작용을 구축하고 우리의 게임이 항상, 매번 “이거 형편없는데”에서 “우와!”가 될 수 있도록 도울 수 있다.