



※ 본 아티클은 CMP MEDIA LLC와의 라이선스 계약에 의해 국문으로 제공됩니다

소규모 개발자가 대규모 게임을 제작하는데 따른 리스크를 최소화하는 법 - 1부 (Small Developers: Minimizing Risks in Large Productions - Part I)

트로이 더니웨이(Troy Dunningway)
가마수트라 등록일(2009. 11. 3)

http://www.gamasutra.com/view/feature/4180/small_developers_minimizing_risks_.php

[이 기사에서는 마이크로소프트, EA, 인섬니악(Insomniac), 유비소프트(Ubisoft)와 같은 스튜디오에서 경력을 쌓아온 숙련된 게임 개발자이자 매니저인 트로이 더니웨이씨가 소규모 개발 팀에서 대규모 개발팀으로 이행하는데 따른 주요 리스크에 대해 서술하고 이와 같은 리스크를 알아내는 방법과 없앨 수 있는 방법에 대해 서술한다. 2부로 구성된 기사 중 1부에 해당하는 이 기사에서는 더니웨이가 요약한 10 가지 리스크 중 네 가지를 다루었으며 대규모 프로젝트에 내재된 리스크를 파악할 수 있는 방법을 제시한다.]

제작 과정에 있는 프로젝트에서 나타날 수 있는 리스크는 다양하며 단순한 리스크로 인해 작품의 품질이 떨어지게 되거나 프로젝트를 취소당하거나 심지어는 회사가 해당 사업을 포기하게 될 수도 있다.

프로젝트 내 리스크는 회사 내 모든 부서에서 다양한 방식으로 영향을 끼치며 종종 프로젝트 별, 팀 별, 회사 별로 매우 다른 리스크가 존재하게 된다. 자신의 프로젝트 리스크를 객관적으로 분석하고 문제점이 어디에 있는지 알아내어 늦기 전에 조기에 문제를 해결하는 것이 중요하다.

게임 업계는 지난 수 년간 다양하게 변화되어 왔다. 소규모 스튜디오에서 규모를 키우거나 “보다 간단한” 플랫폼에서 개발하던 소규모 프로젝트에서 대규모 게임 제작으로 전환하게 되었다면 가능한 한 많은 리스크를 최소화하는 것이 중요하다. 여기서는 보다 대규모 게임을 개발할 때 공통적으로 나타나는 리스크를 없애거나 아니면 최소화하기 위한 여러 가지 팁을 알려준다.

이 글에 적힌 리스트는 이제 막 게임 업계에 진입한 사람으로 처음으로 자신의 프로젝트를 수행하거나 보다 대규모 프로젝트에 임하게 될 사람들을 위한 분명하고 일반적인 도움말이 많이 설명되어 있다. 일부 팁은 조직 내 역할을 확대하려는 기존 개발자나 프로듀서에게도 도움이 될 것이다.

지난 18 년 여의 경력을 거치면서 느껴온 바이지만 게임을 만들 때 일반적인 상식이 무시되는 경우도 많으며 대부분 서두르는 경향이 있다. 바라건대 이 도움말들이 나를 포함한 많은 사람들이 반복적으로 저질러 온 실수들을 답습하지 않도록 도움이 되었으면 한다.

리스크 개요

프로젝트에 존재하는 모든 리스크를 일일이 열거하는 것은 거의 불가능하다. 리스크는 회사 내 모든 요소, 즉 프로세스, 팀, 프로젝트 요구사항, 제작자, 기타 수많은 내부적 요소와 외부적 요소 중 자신이 제어할 수 없는 요소로 인해 일어난다.

특정 프로젝트에서 어떤 리스크를 파악해야 하는가에 대해 확고한 규칙은 없다. 모든 대규모 프로젝트에서는 거의 모든 리스크가 나타나기 때문이다. 최선의 방법은 팀 내에서 가장 경험이 많은 사람(대부분의 경우 리더)에게 주기적으로 프로젝트에 존재하는 리스크를 평가하게 하는 것이다.

예를 들어 최근에 나는 개발자와 그들이 지닌 문제점에 대해 이야기를 나눈 적이 있었는데 이 때 내가 소스코드 제어 시스템에서 변경한 일부 코드를 예전 버전으로 돌려놓으라고 이야기하자 그들은 소스 제어 시스템을 사용하지 않는다고 대답했다.

경험 많은 개발자인 나도 소스 코드 제어 시스템을 사용하지 않은 적이 없는데 일부 개발자들은 아직도 전문 개발 팀이 사용하고 있으며 당연하다고 여기고 있는 최선의 방법과 간단한 규칙을 따르지 않고.

다시 말하지만 리스크를 분석할 때는 객관적으로 문제를 파악하고 분석해야 하며 외부의 도움이 필요할 때는 이를 인정할 줄 알아야 한다.

다음은 프로젝트에 존재하는 10 가지 영역의 리스크이다.

1. 비즈니스 리스크(Business Risks)
2. 제작 전 리스크(Pre-Production Risks)
3. 프로세스 리스크(Process Risks)
4. 팀 리스크(Team Risks)
5. 스케줄 리스크와 프로젝트 관리 리스크(Schedule and Project Management Risks)
6. 설계 리스크(Design Risks)
7. 예술 리스크(Art Risks)

8. 아웃소싱 리스크(Outsourcing Risks)

9. 기술적 리스크(Technical Risks)

10. 테스트 리스크(Testing Risks)

이외에도 프로젝트에는 수많은 리스크 영역이 존재하지만 위 리스트를 이용하면 대부분의 프로젝트에서 일어나는 심각한 문제를 가장 잘 다룰 수 있을 것이다. 그렇다고 해서 이것이 주의해야 할 수많은 다른 요소가 없다는 것을 의미하지는 않는다.

휴대용 게임, 모바일 게임, 캐주얼 게임과 같은 소규모 게임을 제작하고 있다면 위에 열거한 리스크 중 일부는 그다지 크게 다가오지 않거나 문제가 되지 않을 수 있다. 또한 이제까지 해온 것과 매우 유사한 종류의 또 다른 게임 시리즈를 작업하고 있다면 많은 리스크가 일어나지 않을 수도 있다. 그렇지만 항상 주의를 기울여야 한다.

게임 제작에서의 10 가지 리스크 - 1 번에서 4 번까지

1. 비즈니스 리스크

인식하고 있든 없든 간에 소규모 게임 제작자는 끊임없이 리스크에 처하게 된다. 프로젝트 하나만을 담당하는 대부분의 소규모 게임 개발자는 흔히 하나의 프로젝트를 위해 한 곳의 제작사에서만 투자를 받는 리스크를 지니고 있다.

10 년 전까지는 개발과정이나 요구사항이 훨씬 적었기 때문에 몇 일이나 몇 주 만에 새로운 프로젝트에 대한 계약이 가능했다. 5 년에서 10 년 전에는 운이 아주 좋거나 유명한 개발자가 아닐 경우에 새로운 프로젝트를 계약하는데 3 개월에서 6 개월이 소요되었다. 오늘날에는 새로운 프로젝트를 위해 12 개월에서 18 개월 넘게 협상한 팀도 있다고 들었다.

제작사들도 예전에는 프로토타입 제작을 위한 비용을 지불하곤 했으며 메이저 AAA 급 콘솔 게임을 위한 정말 훌륭한 프로토타입을 제작하려면 25 만 달러에서 50 만 달러가 소요되기도 했다.

그렇지만 이제 대부분의 제작사들이 만족할 만한 품질의 프로토타입을 제작하는 비용은 이전의 2 배, 심지어는 3 배가 되면서 제작사들은 이 비용을 지불하길 꺼려하고 있다. 이는 프로토타입 제작과 이를 위한 투자까지



개발자에게 부담 지우게 되는 것으로 대부분의 소규모 회사에게는 위험하고 힘들 일이다.

소규모 게임 개발자들은 오늘날의 게임 개발 시장에서 재정적인 리스크를 더 많이 갖게 된다. 이들은 새로운 프로젝트를 위한 컨셉을 만들어야 하는 부담을 해결해야 하고 새로운 프로젝트를 홍보해야 하며 프로토타입을 개발하고 새로운 프로젝트를 위한 계약을 따내며 현재의 프로젝트가 갑자기 취소되지 않도록 해야 한다.

소규모 인디 개발자라면 비용 문제를 하나씩 해결해 나가기 때문에 현재의 프로젝트가 취소될 경우 새로운 제작사를 찾을 여유를 갖지 못할 것이다. 개발 중인 게임을 소유할 권리를 갖고 있고 게임이 꽤 잘 만들어졌다면 쉽게 계약을 따낼 수 있을지도 모르지만 모든 것을 새로 시작해서 새로운 프로젝트를 찾으려면 4개월에서 8개월이 소요되는 경우가 허다하다.

제작사가 자신의 프로젝트 제작이 취소되는 리스크를 줄이기 위한 손쉬운 해결책은 없다. 최선의 방법은 다음 프로젝트를 지속적으로 모색하고 협상하면서 가능한 한 빨리 두 번째 팀이나 프로젝트를 시도하고 두 번째 경우를 늘리려고 노력하는 것이다. 수입원이 다원화되고 투자원이 많아질수록 회사가 더 안정적이 될 수 있다.

두 번째 프로젝트를 진행하면 일부 팀에서 너무 과도한 성장을 추구하게 되는 또 다른 리스크를 안게 되긴 하지만 그만큼 이점이 많아지기도 한다.

회사를 운영하려면 공유 공간을 구축하는 부담을 비롯하여 접수원, HR, 회계, 관리, 오디오 부서 등과 같은 제작부서가 아닌 인원의 필요성에 이르기까지 많은 비용이 필요하다. 두 개의 프로젝트를 운영하게 되면 이런 비용을 나눌 수 있으므로 회사 자금 사정을 크게 완화할 수 있다.

또한 자신의 비즈니스를 다양하게 만들 수 있는 다른 방법도 찾아보기 바란다. 더 이상 “전통적인” 대규모 상업적 콘솔 게임 제작에 매달려 있을 필요는 없다.

여러분이 알고 있는 것보다 훨씬 쉽고 빨리 게임을 제작할 수 있는 새로운 플랫폼이나 제작 옵션, 온라인 판매, 캐주얼 게임, 휴대용 게임 등이 많이 있다. 현재의 게임 역시 확장할 수 있는 수많은 옵션이 있으므로 다운로드할 수 있는 콘텐츠나 테마 팩, 기타 콘텐츠 등을 이용하여 수익을 더 창출할 수 있을 뿐만 아니라 게임의 브랜드 가치도 높일 수 있을 것이다.

2. 제작 전 리스크

제작 전 과정과 연관된 가장 큰 리스크는 적절한 제작 전 과정을 전혀 갖지 않는 것이다. 일부 팀은 곧바로 게임 제작 과정으로 돌입하여 제작을 진행하면서 계획을 수정하곤 한다. 제작 전 과정은 프로젝트에서 일어날 수 있는 리스크를 완화하고 여러분이 실행할 수 있다고 알고 있고 실행 가능하다고 증명되는 계획을 구축하는 과정이다.

불행히도 제작 전 과정에 대한 정의는 대부분 약간 모호하기 때문에 제작 전 과정에서 잘못된 일을 하거나 일을 철저히 하지 못할 가능성이 매우 크다.

예를 들어 팀들은 제작 전의 모든 시간을 새로운 게임 기술과 게임 역학을 테스트하기 위해 사용하지만 무언가 다른 것을 만들어내는 데 실패하고 만다. 이 경우 자신들이 계획한 대규모의

새롭고 멋진 것으로 기대하는 기능이 형편없으며 실행할 가치가 없다는 것을 증명할 수는 있지만 한편으로는 12 개월을 허비하면서도 보여줄 것이 하나도 없기 때문에 기본적으로 다시 시작해서 새로운 아이디어의 핵심 기능을 실행할 수 있다는 것을 증명해야 하는 상황을 맞게 된다.

흔히 나타나는 또 다른 리스크는 다양한 기능의 프로토타입을 제작하면서 코드를 모두 쪼개놓는다는 것이다. 이런 코드는 다시 짜야 하지만 이제 갑작스럽게 모두가 모든 것이 “제대로 작동한다”고 느끼면서 팀에서 실제로 의도한 것보다 훨씬 빨리 게임을 만들 수 있다고 생각하게 된다.

제작 전 과정에서 무엇을 하는 가는 어떻게 그 일을 하는지와 마찬가지로 중요하다.

컨셉 개발 리스크

많은 팀들은 아직도 제작 전 과정과 컨셉 개발 과정에 대해 제대로 이해하지 못하고 있다. 다른 프로젝트 작업을 진행하면서 새로운 프로젝트의 컨셉을 제대로 개발하기는 대단히 힘들다. 대부분의 팀들은 이전 프로젝트를 마무리하면서 새로운 프로젝트의 컨셉을 구축하기 위한 팀을 따로 할당할 만큼 자원이 넉넉하지 않다. 이는 실제로 오늘날의 게임 개발에서 가장 문제가 되고 있는 사항 중 하나이다. 이는 많은 프로젝트들이 시작부터 잘못되었음을 의미하며 첫 날부터 잘못된 길로 나아감을 의미한다.

게임을 완벽하게 개발하려면 새로운 프로젝트를 위한 컨셉을 개발하는 팀은 비교적 소규모여야 하며 아이디어를 제대로 적을 수 있는 시간 여유가 주어줘서 제대로 된 팀이 본격적으로 일에 돌입하기 전에 아이디어를 검증해야 한다.

현실에서는 하나의 프로젝트가 끝나면 팀 구성원들이 과다 업무 처리로 지쳐있어도 짧은 휴식만을 취한 후 다시 다음 프로젝트에 투입되어야 하는 경우가 많다.

첫 날에 일어날 수 있는 가장 나쁜 일은 팀에게 출시 날짜, 팀 규모, 예산 프로젝트 요구사항들을 못박는 것이다. 다른 일이 일어나지 않는다면 이 모든 것이 엄격하게 지켜지고 팀을 옳아매게 될 것이다.

대규모 게임에서는 보다 대규모 팀이 작업을 시작하기 전 컨셉 작업에 3 개월에서 6 개월 정도가 필요한 경우가 많다. 특정 팀이 하나의 프로젝트에서 다른 프로젝트로 전환해서 일을 해야 한다면 이런 전환에 대해 준비가 되어 있어야 한다. 설계 팀은 첫 날부터 너무 많은 부담을 안게 되기 때문에 절차를 무시하게 될 것이며 게임 일부분이 제대로 설계되지 않거나 후에 재 작업을 해야 하는 경우가 발생하게 될 것이다.

프로젝트의 제작 전 과정과 컨셉 구축과 관련된 문제는 일반적으로 인원 분배와 스케줄 조절의 문제로 인해 생긴다. 새로운 아이디어 작업을 전담하게 하고 필요할 경우 제작 과정에 참여할 준비를 갖춘 소규모 컨셉 구축 팀을 할당하거나 팀을 재구축해서 팀 구조를 보다 유연하게 만들어 준비가 되지 않은 채 제작 과정에 뛰어들지 않게 만들면 이런 문제를 피할 수 있다.

아웃소싱된 노동력이나 회사 내 노동력(특히 아티스트)를 많이 활용하고 있는 경우에는 대규모 엔진 프로그래밍 팀이 필요하지 않도록 표준화된 특정 기술을 구축하고 다양하게 활용할 수 있는 유연한 설계팀을 만들어 훨씬 더 쉽게 여러 곳에 적응할 수 있는 팀을 만드는 것이 좋다.

프로토타입 리스크

컨셉 리스크와 제작 전 단계 리스크와 더불어 프로토타입 구축과 연관된 리스크도 많다. 프로토타입이 제대로 구축되지 못할 경우에는 훨씬 많아질 것이다.

개발 리스크에 어떤 것이 존재하는지는 프로토타입 구축 단계에서 증명되어야 한다. 프로토타입에서는 리스크가 있는 설계 요소, 새롭거나 리스크가 있는 기술, 제작 파이프라인이나 게임 제작 체제가 본격적으로 시작되기 전에 증명되어야 할 요소들을 모두 판명해 내야 한다.

프로토타입을 구축하는 것만으로는 특정 프로젝트에 나타나는 리스크를 충분히 완화할 수 없다. 어떤 것을 프로토타입으로 구축하는가를 신중하게 평가해서 가능한 한 많은 리스크를 최소로 만들어야 한다. 프로젝트 리스크와 팀 리스크에 어떤 것이 있을지 평가해야 하며 이들이 제대로 파악되었는지 확인해야 한다. 설계 측면에서 살펴보면 재미있지 않게 만들어질 리스크와 실제로 제작 상 측면이나 기타 측면에서 너무 어렵게 설계될 수 있는 리스크가 있는지 알아낼 수 있는 방법이 필요하다.

프로토타입 구축이 완료되면 제작 과정에 나설 준비가 되어 있음을 확인해야 하며 프로젝트에 기타 파악하고 있는 리스크를 다룰 계획을 갖고 있는지 확인할 수 있어야 한다.

3. 프로세스 리스크

간단히 말해서 게임 개발 프로세스는 게임이 개발될 방법을 규정한 것이다. 모두 빠짐없이 실행되는지, 계획된 순서대로 제작되고 있는지를 확인하기 위해 취해지는 과정들이다. 이 과정에서는 문서화 작업이 얼마나 이루어져야 하는지, 작업이 어떻게 허가되는지, 팀이 어떻게 구축되는지, 전체적인 작업흐름이 어떻게 되는지, 기민한 개발 모델을 사용할 것인지 등을 지정하게 된다.

개발 프로세스가 얼마나 제대로 운영되는가가 리스크를 회피하는 데 있어 중요한 역할을 한다는 것을 이해하는 것도 중요하다. 어떻게 작업을 이끌 것인지, 목표를 어떻게 이룰 것인지에 대한 분명한 이정표를 갖지 못한다면 큰 리스크에 부딪히게 될 것이다. 매우 분명한 프로세스를 만든다는 것은 개별적인 기능에 대해 문서화하는 것뿐만 아니라 어떻게 게임이 만들어지고 어떻게 의사결정이 이루어질 것인가 등에 관한 전체적인 파이프라인을 구축하는 것도 포함된다. 프로세스를 갖추게 되면 표준화 작업을 해야 하며 시간과 예산에 맞춰 순서대로 작업이 이루어지기 위해 작업을 어떻게 해나가야 하는가에 대한 관점을 갖게 된다.

개발 프로세스는 또한 제작사, 라이선스 권리자, 기타 프로젝트와 관련된 외부 요인의 요구사항의 관계를 이해하기 위해 중요하다. 게임에 대한 최종 권한이 있는 외부인들의 요구사항도 종종

이므로 그들이 필요한 사항과 게임 제작 중 언제 이러한 것이 필요한지를 이해하는 것 역시 중요한 사항이다.

예를 들어 특정 E3 데모를 만들거나 게임이 출시되기 전 플레이어들에게 배포될 데모를 만들기 위해 시간을 투자하지 않는 경우가 많다. 제작자에게 건네져야 할 데모가 얼마나 많아야 하는지 매월(또는 특정 시간간격마다) 제작자에게 피드백을 주고 기다려야 하는 시간이 얼마나 소요되는지에 대해 과소평가하는 경향이 많다. 예전에 제작자와 정기적인 리뷰를 해야 하는 경우가 있었는데 리뷰 시간 때마다 게임 외관이 보기 좋지 않거나 개발상황이 제대로 진척되지 않을 경우에는 게임이 무산될 위기에 처하기도 했다.

내가 프로젝트를 진행할 때에는 전체 프로젝트 자원과 인원 중 약 20% 정도를 실제 게임에는 거의 아무 도움도 되지 않을 데모 만들기에 할당했다. 소규모 데모 전담 팀에게 프로젝트를 계속 유지시키기 위해 데모만을 만들게 하는 것이 쉽지는 않았다. 그렇지만 수개월마다 리뷰를 위해 데모를 만들기 위해 2,3 주 가량을 전체 팀이 허비하게 되는 것보다는 나았다.

언제, 어디서, 어떻게 개발 프로세스가 잘못될지에 대해 이해하는 것 역시 중요하다. 개발 프로세스를 이제까지 일해오던 대규모 회사로부터 가져왔거나 아니면 지난번 프로젝트에서 사용했던 것이었을 수도 있다. 프로세스가 어디에서 비롯되었는지에 상관없이 프로세스가 잘못되어 실제로 프로젝트에 해를 주고 있다고 판단되었을 경우 고지식하게 그 프로세스에 집착해서는 안 된다는 것을 명심해야 한다.

훌륭한 프로세스를 갖기 위한 정답이나 오답이 있는 것은 아니지만 프로세스가 제대로 정의되거나 다듬어지지 않는다면 이로 인해 큰 리스크가 초래된다는 것을 배우게 되었다. 또한 제대로 문서화 작업이 이루어지지 않고 사후 지원이 되지 않는 프로세스 역시 리스크 요소가 된다는 것을 알게 되었다. 그러므로 결론은 진정으로 개발 모델과 프로세스를 이해해야 한다는 것으로 그렇지 않으면 프로젝트를 진행하면서 여러 측면에서 큰 리스크에 부딪히게 될 것이다.

4. 팀 리스크

대부분의 소규모 팀에서 나타나는 가장 큰 리스크 중 하나는 팀 구조로 인한 것이다. 대규모 회사 팀은 백 명이 넘는 사람들을 쉽게 모을 수 있지만 정말 소규모의 아이폰 팀이나 캐주얼 팀은 5 명이 채 되지 않는 경우도 있다. 어떤 팀은 팀 구성원 모두가 개발자인 반면 다른 팀은 내부의 기타 부서 팀, 외부 팀, 아웃소싱 회사, 컨설턴트의 도움을 받기도 한다.

어떻게 팀을 구성한다고 해도 그 결정에는 단기 리스크와 장기 리스크가 따르게 마련이다. 보다 큰 팀은 매월 팀 유지 비용이 많이 들기 때문에 이들이 계속 바쁘게 일하여 비용의 효율성을 높이기 위해 더 많은 프로젝트에 투여되어야 한다. 그렇지만 내부의 대규모 팀에서는 더 많은 인재들을 고용할 있기 때문에 제품의 품질이 향상된다.

외부 팀 구성원을 선발할 경우도 있으며 그래야만 하는 상황에 처하게 되면 외부 인력을 사용하는 것에 대한 리스크를 평가해야 한다. 이 때 비용 감소라는 이점과 프로젝트와 회사에 주는 부담을 함께 고려해야 한다.

그렇기 때문에 팀 구성을 어떻게 하든 간에 팀과 관련된 리스크를 단기적 안목과 장기적 안목으로 살펴봐야 한다.

직원 리스크

잘못된 사람을 고용하거나 필요한 사람을 고용하지 못할 경우, 핵심 인력이 빠질 경우, 또는 기타 직원 문제로 인해 프로젝트에 심각한 리스크가 초래될 수 있다. 현재 수많은 인재들이 있다고는 하지만 자격을 갖춘 사람을 찾는 것이 쉽지는 않다.

어떤 자리에 적합한 수십, 수백 명의 구직자를 발견할 수 있겠지만 그 중 소수만이 진정 다른 사람을 도울 수 있다. 제대로 된 사람을 발견한 경우 이들을 고용하고 올바른 자리에 배치하여 적응하게 하기 위해 이전에 고려하지 않았던 몇 개월을 허비해야 할 수도 있다. 많은 팀들이 직원을 고용하는 것이 얼마나 어렵고 오래 걸리며 비용이 소모되는지 고려하지 못하고 있다.

직원들은 너무 힘들게 몰아붙이면 직원이 떠나게 될 수도 있다. 어느 시점에서든 핵심 인재를 잃게 되는 것은 비용 소모적으로 프로젝트에 큰 타격을 준다. 프로젝트에 투입된 사람들을 모두 평가하고 그들이 이직하거나 더 이상 일하지 못하게 될 경우 어떻게 될 것인가를 이해해야 할 필요가 있다.

어떤 경우에는 일부 팀 구성원에게 자신의 프로세스나 기술, 지식 등에 대해 문서화 작업을 하게 하여 다른 사람이 백업을 하거나 대체 투입될 경우 이런 문서를 통해 배우고 훈련 받을 수 있도록 할 필요가 있다. 한 사람이 핵심적인 사항을 모두 독식하도록 하는 것은 피해야 한다. 그래야만 이 직원도 휴가를 갈 수 있고 일이 잘못되어도 프로젝트가 완전히 붕괴되는 일을 막을 수 있다.

직원 리스크를 줄이기 위해서는 조기에 직원을 고용하여 이들을 행복하게 해 주고 업무량이 너무 많거나 너무 없는 상황이 없도록 신경 써야 하며 지식들을 문서화하고 가능하다면 예비 인력에게 지식을 전수하는 것이 좋다.

고용 리스크에 대한 대처

아웃소싱 외에 고용 분야에서 할 수 있는 것은 핵심 팀이 아닌 사람을 프로젝트에 투입하는 것이다. 이는 외부 디자이너, 프로그래머, 음향 디자이너 등을 고용하는 것을 포함한다.

“고용 문제”라는 말은 종종 정규 팀원을 고용할 경우 쓰이긴 하지만, 계약 직원 고용에도 적용될 수 있다. 게임을 다른 플랫폼으로 이식하는 경우거나 게임 모드 전체(멀티플레이어)를 바꾸는 작업을 할



경우 등의 일부 경우에서 회사 전체가 프로젝트에 부분적으로 고용되는 경우도 있다.

외부 회사의 도움을 받을 경우에는 항상 리스크가 존재한다. 아웃소싱과 마찬가지로 그 회사가 어떤 일을 할지 전혀 알지 못한다는 것이다. 같이 작업하게 될 회사와 직원들에 대해 조사하고 이들이 좋은 평가를 받고 있는지, 작업을 망치지 않을 지에 대해 확인해야 한다. 각 팀이 실패했을 경우 팀이 프로젝트에서 무슨 일을 할 수 있는지 이해하는 것도 중요하다.

또한 개발자나 계약 직원들을 잘 관리해서 데드라인을 적절하게 설정하고 있는지 확인하는 것도 중요하다. 가장 훌륭한 팀과 인재를 공급하기로 약속하고는 다른 사람을 보내거나 여러분의 프로젝트에 투입되어야 할 직원을 다른 프로젝트에도 투입하여 품질을 낮추게 되는 경우도 종종 있다.

외부 인력을 사용할 때 이와 연관된 모든 리스크를 완화하기는 불가능하다. 여러분이 할 수 있는 최선의 방법은 이들이 정해진 이정표를 충실히 지키고 있는지 확인하고 끊임없이 대화를 나누며 자주 만나는 것이다.

2 부의 내용

기사의 1 부에 해당하는 이 글에서 다뤄진 리스크는 보다 일반적이고 광범위한 것이다. 2 부에서는 제작 과정 중 프로젝트에 투입되는 여러 팀에 영향을 주는 보다 세세한 리스크에 대해 설명한다. **소규모 개발자가 대규모 게임을 제작하는데 따른 리스크를 최소화하는 법 - 2 부**에서는 리스크를 없애는 방법과 우선적으로 리스크가 일어나지 않도록 하는 방법에 대해 보다 자세하게 살펴본다.

이 기사가 도움이 된다고 여긴 사람 중 내가 쓴 *앞날을 위한 제작과정의 세계화(Globalizing Production for the Future)*라는 기사를 읽지 않은 사람은 이 기사를 한번 읽어보기 바란다. 이 기사에서도 팀과 프로젝트를 구성하는 방법과 대규모 게임 제작 과정에서 나타나는 리스크를 최소화하는 방법에 대한 많은 의견이 담겨있다.