

3-DAT 기반의 모바일 게임 개발 프로세스 평가 방법*

이지원*, 김경식**, 박성준**†
 호서대학교 컴퓨터공학과*, 호서대학교 게임공학과**
 gshark@nate.com*, {kskim, sjpark†}@hoseo.edu**

3-DAT Based Evaluation Method for Mobile Game Development Process

Ji-Won Lee*, Kyung-Sik Kim**, Sung-Jun Park**
 Dept. of Computer Science, Hoseo University*
 Dept. of Game Engineering, Hoseo University**

요 약

4-DAT는 일반적인 소프트웨어 개발 방법인 XP와 Scrum을 분석하는 도구이다. 본 논문에서는 3-DAT를 기반으로 모바일 게임 개발 환경에 맞는 새로운 평가 방법을 제안한다. 4-DAT 평가 방법은 모바일 게임 개발에 있어서 평가 항목 및 불필요한 요소가 있기 때문에 3차원으로 최적화된 평가 방법을 적용하였다. 본 논문에서는 새롭게 정의한 3-DAT 기반의 평가 방법을 위해 P.Abrahamsson, Boehm, B. R. Turner의 이론을 기초로 하여 5항목, 20세부 기준의 개발 상황 변수를 정의하였다. 실험 방법은 4개월 동안 진행된 5개의 모바일 게임 개발 데이터를 기반으로 개발 상황 변수와 재구성된 3-DAT에 적용하였고, 구현한 시뮬레이터를 통해 3차원 그래프를 도출하였다. 실험 내용은 개발팀의 의존성, 편중성, 유연성에 대한 요소를 분석하였다. 실험 결과로 단기간의 모바일 게임 개발 시 의존성이 높고, 편중성이 낮으며, 유연성이 높을 때 위험성이 적은 개발을 완료 할 수 있음을 알 수 있었다.

ABSTRACT

The 4-DAT is a tool for analyzing general development process such as XP and Scrum. In this paper, we suggest a new evaluation method based on 3-DAT for mobile game. Since the 4-DAT has inappropriate evaluation items and unnecessary elements for mobile game development, a revised evaluation method which was optimized using three dimension. The 3-D based evaluation method have development context variables of 5 items and 20 detailed criteria. They were defined based on the theory of P.Abrahamsson, B. Boehm and R. Turner. For the experiment, We applied to five mobile game developments which were carried out for 4 months. It analyzed the three factors of the development teams : dependence, inequity and flexibility. This study found that short-term mobile game developments can be completed with low risk when dependence is low, inequity is high and flexibility is high.

Keywords : 4-DAT, Development Process, Mobile Game, Analysis Tool

접수일자 : 2010년 08월 04일, 일차수정 : 2010년 10월 08일, 심사완료 : 2010년 11월 05일

교신저자(Corresponding Author) : 박성준 (sjpark@hoseo.edu)

※ 이 논문은 2009년 호서대학교 World Class 2030 프로젝트의 지원을 받아 수행된 연구임.

1. 서론

현재의 모바일 플랫폼 시장은 고성능, 신기술을 탑재한 스마트폰과 태블릿 PC가 잇따라 출시되고 있다. 또한 시장 점유율과 오픈 시장의 규모가 확장되고, 사용자들의 직접적인 개발과 참여로 확대되고 있다. 그러나 대부분의 높은 판매율 유지와 상위 그룹에 오랜 기간 체류하는 어플리케이션은 게임 개발 업체에서 등록한 것이다. 반대로 사용자가 개발한 모바일 어플리케이션은 아이디어가 참신하지만 단기간에 사장되는 경우가 많다. 소규모로 구성된 개발팀은 능동적인 대처와 유연한 일정 조정이 필요하다. 그러나 게임 개발 업체가 한 발 앞선 시기에 유사 게임을 등록하여 순위를 선점하거나, 품질, 인지도, 마케팅 등의 경쟁력에서 우위를 점한다. 이는 아마추어 개발팀이 개발 프로세스의 도입이 어렵고 유연함이 떨어지기 때문이다. 특히, 빠르고 잦은 시장의 요구사항 변동에 대한 미진한 대처와 개발 과정 및 일정의 문제가 있는 경우가 많다.

본 논문에서는 애자일 소프트웨어 개발 프로세스 분석 도구인 4-DAT (4-Dimensional Analytical Tool)를 기반으로 모바일 게임 개발 환경에 적합한 새로운 평가 방법을 제안한다[1]. 기존의 4-DAT가 가지는 4가지 고려사항은 개발 상황, 개발 프로세스, 관리 프로세스, 비즈니스를 평가하는 항목을 모두 포함한다. 이것은 개발 평가 측면에서 포괄적이고 모바일 어플리케이션의 단기간의 개발 과정에 적합하지 않다. 따라서 불필요한 항목과 보완 요소를 조정하여 4차원의 고려사항을 3차원으로 재정의 하였다. 3-DAT를 기반으로 한 프로세스의 객관적인 평가를 위해 P. Abrahamsson, Boehm, B, R. Turner의 이론에 기초로 개발 상황 변수를 정의하였다. 개발 상황 변수는 게임 개발 중 발생 가능한 상황을 실제 데이터를 기반으로 정의한 것이다.

본 논문에서 제안한 모바일 게임 개발 프로세스 평가 방법은 모바일 게임 개발 환경에서 프로젝트 팀이 가지는 제약성을 보완 할 수 있고, 게임 개발을 위한 프로세스의 도입에 있어 부족한 요소를 파악 할 수 있다. 본 논

문의 구성은 다음과 같다. 2장에서 4-DAT와 소프트웨어 개발 프로세스에 대한 관련 연구를 소개하고, 3장에서는 3-DAT와 개발 상황 변수에 대해 다룬다. 4장에서는 제안한 평가 방법의 실험과 결과 분석을 논한다. 끝으로 5장에서 결론 및 향후 과제로 마무리한다.

2. 관련 연구

2.1 4차원 분석도구(4-DAT)

4-DAT는 애자일 소프트웨어 개발 방법의 특정 목적을 분석하기 위해 만들어진 프레임워크 기반의 평가 도구로써 정량적 민첩성 정도를 측정하는 메커니즘이다. 4-DAT는 세부 평가를 위한 다른 관점에서의 평가 기준을 가지고 있다. [표 1]은 4-DAT가 가지는 평가 기준으로, 4가지의 기준을 가지고 각 항목을 분류하였다[2].

[표 1] 4-DAT의 특성과 평가 기준

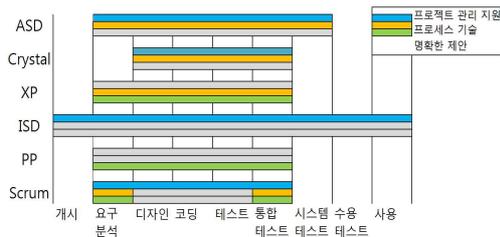
4-DAT 분류	평가 기준
1. 개발 방법의 범위 - 프로젝트와 개발 방법	
프로젝트 규모	소/중/대규모
팀의 크기	소/대그룹 (단일/복수 팀)
개발 스타일	급진적, 반복적
코딩 스타일	단순/복잡
기술적 환경	툴/컴파일러
물리적 환경	사내/재택(외주) 근무
비즈니스 문화	협력/리더중심/개인능력
메커니즘 이론	목표지향/책체지향
2. 민첩성 - 민첩성에 대한 개념	
유연성	예정된/예기치 않은 변화에 적응
속도	결과물의 산출 기간
의지	짧은 주기, 경제적/단순/품질 수단
학습	지난 프로젝트의 지식, 경험 수용
응답	개발 방법 도입 후 민감한 반응
3. 애자일 가치 - 애자일 가치와 원칙	
개인파 상호작용	개인파 상호작용/프로세스와 도구
작동하는 소프트웨어	소프트웨어/포괄적인 문서
고객과의 협력	고객과의 협력/계약 협상
변화에 대한 대응	변화에 대응/계획 유지
프로세스의 기민함	기민함 유지를 위한 실행방법
비용의 효율성	비용의 효율성을 위한 실행방법
4. 소프트웨어 프로세스 - 관례적인 개발 방법	
개발 프로세스	주 생명주기 프로세스/테스팅
프로젝트 관리 프로세스	프로젝트 관리 전반의 실행방법
제어/지원 프로세스	배치 허용 관리 프로세스
프로세스 관리 프로세스	프로세스를 관리하기 위한 제한

개발 방법의 범위 특성은 4-DAT의 첫 번째 분류 기

준으로, 프로젝트의 주요 항목의 범위를 가늠하는 척도이다. 주로 상위 레벨을 비교하는 기준이 된다. 민첩성 특성은 프로세스의 민첩성이 가지는 한계를 측정하는 기준으로, 주기와 실행방법의 두 단계로 구분되며, 예기치 않은 상황 또는 변경사항에 대한 유연성, 대응력 등을 평가하는 5가지의 특징을 가진다. 애자일 가치 특성은 애자일 선언문에서 언급된 모든 애자일 프로세스가 추구하는 네 가지의 공통적인 가치를 포함한 것으로, 다른 개발 방법들의 실행 절차로 프로세스와 개발 환경을 측정하는 항목이다. 소프트웨어 프로세스 특성은 생산 기술을 위한 프로세스와 프로세스를 관리하기 위한 프로세스로 나누고, 생산기술 프로세스는 개발과 프로젝트 관리, 지원 프로세스로 세분화되며, 다른 항목을 돕는 효율성과 실행 방법들로 평가한다.

2.2 애자일 방법론(Agile Methods)

애자일(Agile)이란, ‘민첩한’, ‘재빠른’이라는 뜻을 가진 단어로, 게임 개발 업계에서는 환경의 변화에 신속하게 대처할 수 있는 유연한 서비스의 효율적인 시스템이나 개발 프로세스를 뜻한다[3]. 개발자들에게 가장 잘 알려지고 사용되는 애자일 방법론은 XP와 Scrum으로, 다른 실행 방법과 혼합적으로 사용되거나 흡수되어 사용되는 것이 일반적이다[4].



[그림 1] 애자일 방법론의 특성 비교

[그림 1]은 애자일 개발 프로세스의 특성을 나타낸 것으로, XP와 Scrum은 기타 프로세스와는 약간의 차이점을 가진다. 프로젝트를 다루는 범위는 비슷하지만, 일관되고 명확하게 지원할 수 있도록 하는 점은 방법론적인 측면에서 차이가 있다. ASD나 Crystal 프로세스의 경우

관리와 진행에 초점을 맞추고 있고, 이를 심화한 ISD는 전 과정의 관리를 다룬다. PP는 짝 프로그래밍(2인 1개조)으로 개발 측면만을 다룬다. 그러나 XP와 Scrum은 관리와 개발을 최적화하기 위한 방법으로 관리의 측면을 줄여 개발에 치중하고, 반대로 관리에 더욱 투자하고 개발을 단순하게 진행하여 비용과 시간을 절감한다[5].

2.3 기타 개발 프로세스

애자일 방법론이 아닌 전통적인(traditional) 개발 프로세스는 대표적으로 단계적 출시 모델이나 폭포수 모델 등을 꼽을 수 있다. [표 2]와 같이 요구사항의 잦은 변화와 그로 인한 반복적인 개발 단계를 실행하는 과정이 상대적으로 기민함을 유지하기 힘들고, 위험을 관리하기 어려우며, 각 단계마다 결과물을 가시화하기 어렵다. 반대로 애자일 방법론의 XP와 Scrum은 프로세스의 기민함과 적응력, 이해도 및 민첩한 대응과 안정성이 보장되기 때문에 본 논문에서는 XP와 Scrum을 기본적인 개발 프로세스로 사용한다[6].

[표 2] 개발 프로세스 비교 평가

○:좋음 △:보통 X:나쁨

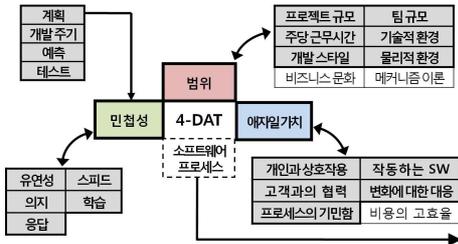
프로세스 평가	단계적 출시	폭포수	XP	Scrum
요구사항 수용	X	X	○	○
시스템 개발 안정성	○	○	○	○
위험 관리	△	X	○	○
제약사항 대응	△	△	△	○
업무 과중	△	X	△	△
고객 가시성	△	X	○	○
관리자 가시성	○	△	○	○

3. 3-DAT 및 개발 상황 변수

3.1 4-DAT의 3차원 재정의

기존의 4-DAT가 가지는 대표 기준은 고려 사항이 많기 때문에 평가 방법의 목적이 흐려질 수 있다. 특히 [표 1]의 4-DAT를 평가하는 기준 중 4번째 항목인 소프트웨어 프로세스 특성은 프로세스를 가늠하는 항목이다. 이 특성은 프로세스 관리와 지원에 해당되기 때문에

개발만을 다루는 단계와는 초점의 차이가 있다. 연구 방향의 차이는 1번째 항목인 개발 방법의 범위와 3번째 항목인 애자일 가치 항목에서도 나타난다. 비즈니스의 문화, 베키니즘 이론, 비용의 고효율을 다루는 측면 역시 차이가 있다. 또한 게임 개발의 특성상 고객 요구사항의 변동이 많고 잦으며, 시장 흐름의 변화 폭이 크기 때문에, 이에 상응하는 2번째 항목인 민첩성 특성에 집중하여 계획, 개발주기, 예측, 테스트의 항목을 추가하여 강화하였다. 각 항목을 모바일 게임 개발에 맞는 항목으로 수정하였으며, 이를 3차원으로 재정의 하였다[그림 2].



[그림 2] 새롭게 정의한 3-DAT

3.2 개발 상황 변수

3.2.1 프로세스 분석

[표 3]은 개발 상황 변수를 정의하여 프로세스를 분석하기 위한 기본적인 관점으로 P. Abrahamsson의 이론에 기초한다[4]. 각 관점은 일일 보고서나 주간 보고서를 통해 프로젝트의 개발과 진행 단계를 파악하고 분석하는 근거가 되고, 프로젝트 전체 일정의 큰 단락이 되는 생명주기를 지원하고 이끌어 나가는 내용으로 분류한다.

[표 3] 프로세스 분석의 초점

관점	종류
소프트웨어 개발 생명주기	어느 범위의 소프트웨어 개발 생명주기를 포함하는가?
프로젝트 관리	이 방법이 프로젝트 관리 활동을 지원 하는가?
이론적 원리 / 구체화된 지시	이 방법의 이론적 원리를 대부분 신뢰할 수 있거나 구체화된 가이드를 제공하는가?
보편적 사전 정의 / 특유의 상황	이 방법이 모든 애자일 개발 상황에 본질적으로 맞춰져 있는가?
경험적 증거	이 방법이 요구에 대한 경험을 지원 할 수 있는가?

3.2.2 개발 레벨 평가

애자일 프로세스에서 가장 비중을 두는 측면은 사람으로, 개발자와 고객을 말한다. 결과물 자체에 집중하기보다 제품을 만드는 사람을 중시한다는 것을 의미한다. [표 4]는 Boehm, B와 R. Turner의 이론에 기초한 개발 레벨 평가로 개발자에 초점을 맞춰 사람, 프로세스, 프로젝트의 측면에서 평가하는 근거가 된다[7].

[표 4] 개발 레벨 평가

사람	개발자	3	전례 없는 상황을 해결 가능
		2	새로운 상황을 조절 가능
사람	개발자	1A	기능적 도구, 업무 판단 가능
		1B	간단한 기능적 도구, 테스트 실행
		-1	협업 환경의 업무 불가 또는 불능
		테스터	
프로세스	프로젝트 리더		
	팀		
	계획		
	문서화		
프로젝트	개발 프로세스		
	프로젝트 타임		
	비즈니스 요인		
		기타 프로젝트 특징	

3.2.3 개발 상황 변수

프로세스 분석과 세부 기준을 평가하는 지표를 기반으로 정의한 개발 상황 변수는 [표 5]와 같이 XP와 Scrum의 주기와 특성에 적합한 팀, 개발, 테스트, 일정, 업데이트의 5항목으로 분류하였고, 민첩함과 유연함을 기반으로 프로젝트의 완성도 및 수행 기간을 중시하여 총 20가지의 세부 기준으로 정의하였다.

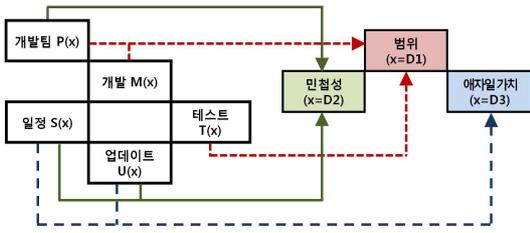
[표 5] 개발 상황 변수의 분류와 세부기준

항목	세부 기준				
	의존성	호환성	균형	실적	
팀					
개발	기술적용	대외협업	코드최적화	플레이적용	개발테크
테스트	테스트계획		테스트		집중검사
일정	납기일	새로운작업	마일스톤	적절한일정	주기수행
업데이트	유지 보수		버그 최소화		데모 활성화

3.3 개발 상황 변수의 측정

앞에서 정의한 20가지 세부기준의 개발 상황 변수가

의미하는 값은 게임을 개발하면서 발생하는 상황의 구분에 따라 나누어진다. 이 구분은 프로젝트의 진행을 문서화한 일일 보고서나 주간 보고서를 통해 알 수 있고, 이를 기초 데이터로 입력받아 각 항목의 가중치 값을 가지게 된다. 개발 상황 변수 값은 3-DAT의 각 축의 세부 항목으로 평가되고, 각 축의 데이터를 도출한다.



[그림 3] 개발 상황 변수와 3-DAT의 주요 관계

[그림 3]은 개발 상황 변수의 5가지 항목이 3-DAT의 각 축에 주요 작용하는 관계를 나타낸 것이다. 개발 상황 변수가 영향을 미치는 관계는 [표 1]의 평가 기준을 반영한 것으로, 3-DAT의 항목에 값을 측정하기 위한 척도가 된다. 3-DAT의 값을 측정하기 위해 개발 상황 변수의 가중치를 부여한다. 작업의 성격에 따라 각 항목의 채택, 주기, 실행 방법이 특정 레벨에 대응 또는 수용하는 사항을 고려한다. 개발 상황 변수의 항목이 고려 사항에 따른 결과를 반영한다면 개발 상황 변수의 항목에 1점을 부여하고, 그렇지 않으면 0점으로 한다.

$$Dx(Obj) = (1/m)\sum m Dx(Object, Phase, Practice) \quad (\text{식 1})$$

Dx = 축
 $Object(Obj)$ = 목적, 객체
 $Phase$ = 주기
 $Practice$ = 실행방법

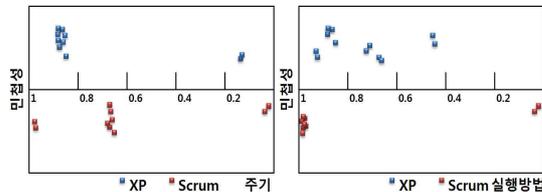
(식 1)은 3-DAT의 각 축의 세부 항목 값을 도출하는 과정으로, 개발 상황 변수의 분류 항목이 가지는 가중치에 따라 3-DAT의 세부 항목의 값이 변화하고, 3축으로 분류하여 3차원으로 결과를 도출한다.

[그림 4]는 3-DAT의 한 축인 D2를 측정된 결과로, XP와 Scrum의 유연함, 스피드, 의지, 학습, 응답을 주기와 실행 방법으로 구분해 측정된 데이터이다[8]. 본 논문에서 강화한 민첩성 특성을 (식 1)에 적용하여 신뢰성을

검증하기 위한 과정으로, 이를 시각화하여 분석하면 [그림 5]와 같다.

민첩성 측정					Scrum								
XP	유연성	스피드	의지	학습	응답	합계	Scrum	유연성	스피드	의지	학습	응답	합계
(0) 주기							(0) 주기						
팀원	1	1	0	1	1	4	프리 게임	1	1	0	1	1	4
계획	1	1	0	1	1	4	개발	1	1	0	1	1	4
반복적인 활동	1	1	0	1	1	4	포스트 게임	0	1	0	0	0	1
생산	1	1	1	1	1	5	합계	2	3	0	2	2	9
유지	1	0	0	1	1	3	민첩성 정도	0.67	1.00	0.00	0.67	0.67	0.60
숙종	0	1	0	0	0	1	(0) 실행 방법						
합계	5	5	1	5	5	21	스크린 테스트	1	1	0	1	1	4
민첩성 정도	0.83	0.83	0.17	0.83	0.83	0.70	스크럼 팀	1	1	0	1	1	4
							(0) 실행 방법						
계획 수립	1	1	0	1	1	4	백로그 작성	1	1	0	1	1	4
짧은 주기	1	1	1	1	1	4	스크린트	1	1	0	1	1	4
매타포	0	1	1	0	0	2	스크린트 계획 미팅	1	1	0	1	1	4
심플 디자인	1	1	1	1	1	5	데일리 스크럼 미팅	1	1	0	1	1	4
데스킹	1	1	0	1	1	4	스크린트 리뷰	1	1	0	1	1	4
개구성	1	1	1	1	1	5	합계	7	7	0	7	7	28
작 프로그래밍	1	0	0	1	1	3	민첩성 정도	1.00	1.00	0.00	1.00	1.00	0.80
공동체 의식	1	0	0	1	1	3							
지속적인 통합	1	1	1	1	1	5							
40시간 근무	0	0	0	1	0	1							
온라인 고객	1	0	0	1	1	3							
표준 코딩	1	1	1	1	1	5							
합계	10	8	5	11	10	44							
민첩성 정도	0.83	0.67	0.45	0.92	0.83	0.73							

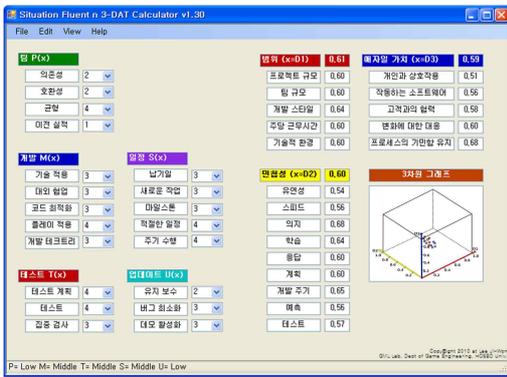
[그림 4] D2(Phase, Practice)의 측정 결과 테이블



[그림 5] D2(Phase, Practice)의 측정 결과 그래프

좌측의 그래프는 프로세스의 주기를 분석한 것이다. XP가 Scrum에 비해 실용적이고 효율적인데, Scrum에서는 개발 이후의 유지 단계를 지원하지 않기 때문이다. 우측의 그래프는 실행 방법을 분석한 것이다. Scrum이 XP에 비해 완성도가 높는데, Scrum의 프로세스가 실행 방법을 위한 구조로 디자인되었기 때문이다[그림 5].

결과는 Qumer A.와 Henderson- Sellers B.가 분석한 애자일 프로세스의 결과와 유사한 값을 가진다[2]. 같은 과정을 반복하여 각 축에 대한 종합 데이터를 구한다.



[그림 6] 시뮬레이션 툴 실행 화면

[그림 6]은 본 논문에서 제안한 평가 방법을 구현한 시뮬레이션 툴의 실행 화면이다. 시뮬레이션 툴은 크게 개발 상황 변수의 입력 처리, 3-DAT의 변환 처리, 3차원 시각화 처리의 3가지로 구성된다. 실행 단계는 다음과 같다. 결과보고서를 기반으로 가중치 값을 좌측의 개발 상황 변수 항목에 입력한다. 입력된 값은 [그림 4]의 방식을 거쳐 3-DAT의 세부 항목 값으로 계산하여 우측에 출력된다. 계산된 결과는 [그림 5]와 같은 평면 그래프를 3차원 그래프로 시각화하고, 결과를 출력하는 절차로 실행한다.

4. 실험

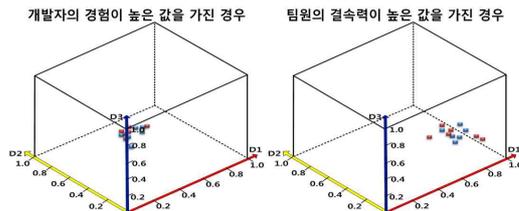
본 논문에서는 3장의 개발 상황 변수에서 도출한 값을 3-DAT에 적용하였다. 범위, 민첩성, 애자일 가치를 D1, D2, D3의 축으로 하였으며, 개발팀의 의존성, 특정 파트의 편중성, 개발 방법의 유연성을 3차원 분석하였다. 실험의 방법으로 4개월 동안 진행된 5개의 모바일 게임 개발 프로젝트의 결과 데이터를 개발 상황 변수와 3-DAT에 적용하였다.

[표 6] 조건별 개발 상황 변수의 평균 레벨

주요항목	팀	개발	테스트	일정	유지
하이스펙	높음	보통	낮음	높음	낮음
결속력	높음	보통	보통	보통	낮음
프로그램	낮음	높음	보통	보통	높음
기획	보통	높음	높음	낮음	보통
피드백	보통	보통	낮음	높음	높음
아마추어	낮음	낮음	높음	보통	보통

[표 6]은 6가지 개발 상황에 따라 나타난 개발 데이터를 분석하였고, 개발 상황 변수의 평균값에 따라 크게 세 부류로 구분한다. 첫 번째는 개발팀에 대한 의존도가 높은 경우이다. 개발 실적이 있거나 팀원의 결속력이 두터운 상황으로, 개발 상황에 익숙하거나 전체적인 스펙이 높은 상태를 말한다. 두 번째는 특정 파트에 역량이 편중된 경우로 팀원 중 특정 개발자의 경험이나 스펙이 높아 그 파트를 중심으로 개발이 진행되는 상황이다. 세 번째는 개별적인 특수한 상황으로 유지보수나 피드백을 통한 작업을 주요 업무로 하는 팀과 학생 또는 아마추어 개발자로 이뤄진 팀으로 전체적인 개발 경험이 적고, 스펙이 낮은 상태를 말한다.

4.1 개발팀의 의존성 평가

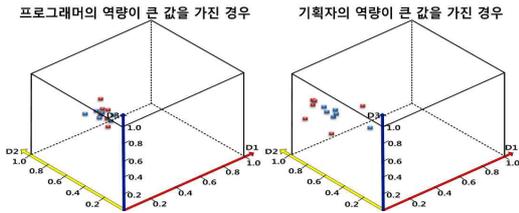


[그림 7] 개발팀의 의존성 평가 결과 그래프

[그림 7]의 좌측 그래프는 개발 경험이 많은 팀의 개발 경우이다. 개인의 의존도와 일정 관리, 기술적 환경에 높은 값을 가졌다. 반대로 상호작용과 협업 환경에 낮은 값을 가지고, 프로젝트와 팀의 규모에 영향을 적게 받아, 개발 상황 변수의 값이 바뀌어도 결과의 편차가 작았다. 팀의 결속력이 높은 개발팀의 경우 우측과 같은 그래프를 보였다. 근무 시간, 개발 스타일, 생산 주기가 높은 값을 가진다. 프로세스의 기민함과 변화에 대응하

는 값이 둔감하여 변화 폭이 작는데, 이것은 유연함과 능동적 대처항목에 영향을 준다.

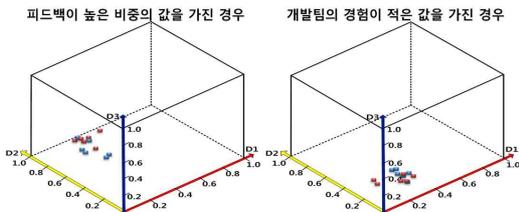
4.2 특정 파트의 편중성 평가



[그림 8] 특정 파트의 편중성 평가 결과 그래프

[그림 8]의 좌측 그래프는 프로그래밍 파트에 팀의 역량이 편중된 경우이다. 코드의 최적화, 일관성 있는 적용, 테스트의 집중 검사, 주기 관리, 다음 개발 주기의 예측 항목의 값이 높았다. 그러나 업무 과중과 위험 관리에 민감한 변화를 나타냈었는데, 프로그래머에 대한 의존도가 높아 일정 관리에 영향을 주기 때문이다. 우측 그래프는 기획 파트에 역량이 편중된 경우의 결과이다. 버전 관리와 생산 주기, 테스트와 업데이트에 높은 결과를 보였다. 기술적 환경의 변화에 민감하게 반응하여 팀원의 균형과 호환성의 값에 변화 차이가 컸다. 또한 프로그래밍 파트에 편중되었던 경우의 결과와 마찬가지로 개발 내용 변경과 위험 관리에 민감하게 반응하여, 의지와 응답의 변화의 폭이 불규칙하였다.

4.3 개발 방법의 유연성 평가



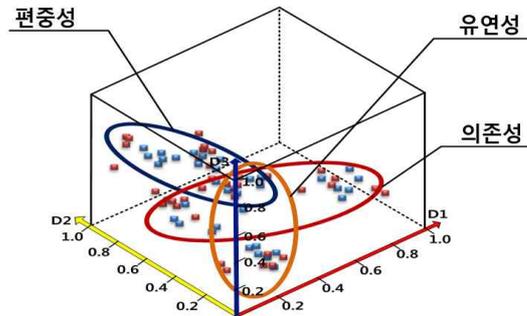
[그림 9] 개발 방법의 평가 결과 그래프

[그림 9]의 좌측 그래프는 피드백에 의한 품질관리가 익숙한 개발자나 유지보수를 주로 하는 팀원이 개발한 경우이다. 새로운 또는 발견된 추가 작업의 양, 팀의 규

모, 개발 스타일의 값이 높았다. 고객과의 의사소통과 요구사항 수용 항목에 특히 민감한 결과를 나타냈는데, 이것은 납기일의 준수와 요구사항의 변동 항목에 영향을 준다. 우측 그래프는 아마추어 개발자로 이뤄진 팀의 경우이다. 팀원의 협업, 공유, 주기 계획이 높은 값을 가졌고, 호환성, 개인에 대한 의존도가 낮은 값을 가졌다. 테스트와 의존도에 민감하게 반응하는 결과를 나타냈는데, 이 항목이 불규칙할 때 주기 수행과 적절한 일정, 새로운 작업 항목의 변화 폭이 불규칙했다.

4.4 종합 분석

개발팀의 의존성, 특정 파트의 편중성, 개발 방법의 유연성 평가의 결과를 종합하여 3차원 그룹화 하였다. 분석한 결과는 각 축의 값의 고저로 구분한다[그림 10]. 의존성은 개발자의 성향에 의한 값의 차이를 보인다. 일차적으로 개발 실적과 주기 수행 경험에 의해 나뉘고, 이차적으로 팀의 상호작용과 결속력이 클수록 높은 수치를 가진다.



[그림 10] 종합 결과 그래프

편중성은 대내외 협업 공유와 밀접한 관련이 있고, 특정 파트에 역량이 치우칠수록 테스트와 생산 주기, 일정 등에 불규칙한 값을 가진다. 즉, 강세를 띄는 파트의 변동에 의해 팀의 개발 일정과 유연성이 급격히 변화하고, 잠재적인 위험요소로 작용한다. 유연성은 프로세스의 기민함 유지와 대내외 협업 공유가 주요 작용을 하여 높은 수치를 가지게 되고, 요구사항의 변동이나 새롭게 발견된 작업의 양이 많을수록 낮은 수치를 가진다.

[표 7] 축에 영향을 주는 항목과 의미

구분	주요 항목						값의 의미
D1(x축)	많은영향	주기수행	유지보수	개발실적	의존성	테스트	개발팀, 개발자
	적은영향	데모활성화	기술적용	플레이적용	집중검사	납기일	
D2(y축)	많은영향	협업공유	마일스톤	개발테크	기술적용	새로운작업	일정, 개발
	적은영향	유지보수	데모활성화	테스트계획	의존성	주기수행	
D3(z축)	많은영향	협업공유	플레이적용	호환성	납기일	데모활성화	기민함, 유연성
	적은영향	개발테크	팀의균형	테스트	버그최소화	적절한일정	

[표 7]은 개발 상황 변수의 값이 3-DAT의 각 축에 많은 영향을 주거나 적은 영향을 주는 주요 항목이다. D1은 개발 상황 변수 항목 중 주기수행, 유지보수, 개발 실적, 의존성, 테스트 등의 항목에 우선적인 영향을 받는다. 이는 D2의 항목과 반대되는 개념으로 D2에서는 적은 영향을 주거나 반대로 저해 요소가 되기도 한다. 또한 D1의 의미는 개발팀과 개발자에 해당하며 이 축의 값이 적을 경우 주요 항목을 체크함으로 보완 할 수 있다. D2는 협업공유, 마일스톤, 개발 테크, 기술 적용, 새로운 작업 등의 항목에 많은 영향을 받고, 이는 D3의 항목과 반대되는 개념으로 D3에서는 영향력이 작거나 저해요소가 된다. D2의 의미는 일정과 개발에 해당하며, D1과 마찬가지로 주요 항목을 조절함으로 보완 할 수 있다. D3은 협업공유, 플레이적용, 호환성, 납기일, 데모 활성화 등의 항목에 많은 영향을 받고, 이는 D1과 반대되는 개념으로 D1에서는 크게 작용하지 않는다. D3 값의 의미는 프로세스의 기민함과 유연함을 의미한다.

5. 결 론

본 논문에서는 소프트웨어 개발 프로세스 분석 도구인 4-DAT를 3차원으로 재정의하였다. 모바일 게임 개발에서 발생할 수 있는 데이터를 개발 상황 변수로 정의하여, 프로젝트의 실제 개발 일정 데이터를 적용하고 분석하였다. 도출된 상황을 개발팀의 의존성, 특정 파트의 편중성, 개발 방법의 유연성으로 그룹화 하였다. 각 그룹은 수치의 고저에 따라 개발팀(x축), 일정 및 개발(y축), 유연함 및 기민함(z축)으로 구성된다.

실험 방법은 4개월 동안 개발한 5개의 모바일 게임 개발팀의 실험 단계와 개발 방법을 분석하였고, 분석된 데이터를 개발 상황 변수에 적용하였다. 실험의 결과로

각 축은 팀의 상호작용과 결속력, 개발일정 및 유지보수, 개발 경험의 정도와 대외 협업 능력 등의 차이에 의한 결과를 나타내었다. 결과를 분석하여 단기간의 모바일 게임 개발 시 의존성이 높고, 편중성이 낮으며, 유연성이 높을 때 위험성이 적은 개발을 할 수 있다는 것을 알 수 있었다.

본 논문에서 제안한 게임 개발 프로세스 평가 방법은 실험을 통해 모바일 게임 개발의 요구 사항과 필요 요소를 도출하였다. 이는 모바일 게임을 개발하는 소규모 팀 혹은 아마추어 팀에 필요 요소와 보강 요소를 추출하여 적용 및 보완 할 수 있을 것으로 예상된다. 향후 과제로 본 논문에서 분석한 내용을 실제 모바일 게임 개발에 적용하여 소, 중규모 이상의 팀에서 작용하는 효율성과 신뢰도를 검증하고 분석하며, 기타 플랫폼의 개발 프로세스에 적용해야 할 것이다.

참고문헌

- [1] Qumer A. and Henderson-Sellers B., "Measuring agility and adoptability of agile methods: A 4-Dimensional Analytical Tool", IADIS, pp 503-507, 2006.
- [2] Qumer A. and Henderson-Sellers B., "Comparative Evaluation of XP and Scrum using the 4D Analytical Tool (4-DAT)", EMCIS, 2006.
- [3] J. A. Highsmith, Manifesto for Agile Software Development, AgileManifesto, 2001.
- [4] P. Abrahamsson, J. Warstab, M. T. Siponenb and J. Ronkainen, "New Directions on Agile Methods: A Comparative Analysis", ICDE03, IEEE, 2003.
- [5] T. Dyba and T. Dingsoyr, "Empirical studies of agile software development: A systematic review", IST, Vol.50, pp. 833 - 859, 2008.
- [6] K. Beck, Extreme Programming Explained: Embrace Change, 2000.
- [7] Boehm, B and R. Turner, "Using Risk to Balance Agile and Plan-Driven Methods", Computer, pp. 57-66, 2003.
- [8] Schwaber K. and Beedle M, "Agile Software Development with SCRUM", Prentice Hall,

- 2002.
- [9] Michael C. and Shawn B, “The Impact of Agile Methods on Software Project Management”, ECBS05, IEEE, 2005.
 - [10] Cockburn, A. and J. Highsmith, “Agile Software Development: The People Factor”, Computer, pp. 131-133, 2001.
 - [11] Boehm B and Turner R, “Balancing agility and discipline: evaluating and integrating agile and plan-driven methods”, 26th ICSE, IEEE, pp. 718-719, 2004.
 - [12] 박천일, 이수정, 양혜술, “기민한 개발 방법을 적용한 게임 개발 프로세스 개선”, 한국컴퓨터 게임학회논문지, 제 12호, pp. 3-9, 2008.
 - [13] 고병희, 김순곤, “모바일 3D 게임 제작환경과 개발프로세스에 관한 연구”, 한국콘텐츠학회논문지, 제 1권, 제 1호, pp. 23-30, 2005.



이 지 원 (Lee, Ji Won)

2008년 2월 호서대학교 게임공학과 (공학사)
2009년 3월-현재 호서대학교 컴퓨터공학과 (공학석사)

관심분야 : 기능성게임, 개발 프로세스, 게임 시스템



김 경 식 (Kim, Kyung Sik)

1982년 서울대학교 전산기공학과 (학사)
1984년 서울대학교 전산기공학과 (석사)
1990년 서울대학교 컴퓨터공학과 (박사)
1984년-1991년 한국전자통신연구원 선임연구원
1991년-현재 호서대학교 게임공학과 교수
2005년-현재 KAIST CT대학원 겸직교수

관심분야 : 게임 엔진, 기능성 게임, 노인용 게임



박 성 준 (Park, Sung Jun)

1997년 2월 호서대학교 컴퓨터공학과 (공학사)
1999년 2월 건국대학교 컴퓨터공학과 (공학석사)
2005년 2월 건국대학교 컴퓨터공학과 (공학박사)
2006년 3월-현재 호서대학교 게임공학과 조교수

관심분야 : 게임, 가상현실, HCI, Computer Vision