

개발비 감정을 위한 스토리 포인트와 기능 점수의 비교

권기태*†

Comparison of Story Points and Function Points for Appraisal of Software Development Cost

Ki-Tae Kwon*†

요약

애자일 방법론은 지난 10여년 이상 소프트웨어 개발 조직 사이에서 상당한 인기를 끌었다. 애자일 방법론은 공식적인 프로세스를 최소화하지만, 개발비 감정 및 비용 추정을 위한 추정 방법론을 필요로 한다. 애자일 소프트웨어 프로젝트에서 스토리 포인트는 예상 공수와 실제 공수를 비교하여 개발 공수를 추정하는 용도로 사용되고 있지만 개발된 소프트웨어의 규모를 제공하지 않으므로 프로젝트에서 달성한 생산성이나 프로젝트 전반의 성과 비교, 그리고 조직 수준의 성과를 비교하는데 어려움이 있다. 본 연구에서는 사례 분석을 통해 애자일 관점에서 기능 점수 기반과 스토리 포인트 기반 노력 추정의 효과를 비교한 결과 기능 점수 기반 추정이 더 정확했으며 스토리 포인트로 계산한 생산성보다 편차가 적은 기능 점수 기반 생산성 계산이 가능한 것으로 관찰하였다. 이를 통해 애자일 소프트웨어 프로젝트에서 사실 상의 표준 척도인 스토리 포인트의 개선 방향을 제시하였다.

Abstract

Agile Agile methodologies have gained significant popularity among software development organizations. While agile methodologies minimize formal processes, they still require estimation methodologies for development cost appraisal. In agile software projects, story points are used to estimate development man-hours by comparing the estimated man-hours to the actual man-hours, but they do not provide the size of the software developed, which makes it difficult to compare the productivity achieved in a project, or to compare performance across projects. In this study, we compare the effectiveness of function point-based and story point-based effort estimation from an agile perspective through a case analysis and observe that function point-based estimation is more accurate and allows for function point-based productivity calculations with less deviation than productivity calculated with story points. This led to the improvement of story points as a standard measure in agile software projects.

한글키워드 : 개발비 감정, 스토리 포인트, 기능 점수, 애자일 방법론, 생산성

keywords : cost appraisals, story points, function point, agile methodology, productivity

* 강릉원주대학교 컴퓨터공학과

† 교신저자: 권기태(email: ktkwon@gwnu.ac.kr)

접수일자: 2024.06.01. 심사완료: 2024.06.10.

게재확정: 2024.06.20.

1. 서론

소프트웨어 공수 추정 작업은 추정의 정밀도

가 조금만 달라져도 조직에 상당한 손실을 초래할 수 있으므로 소프트웨어 프로젝트의 관리 및 개발비 감정에 있어 가장 중요한 활동 중 하나이다. 성숙한 소프트웨어 조직일수록 정확한 비용 추정이 비즈니스 이점을 제공하는 프로세스로 인식되고 있다. 애자일 소프트웨어 개발은 소프트웨어 개발 중 언제든지 변화를 수용하고 빠른 대응을 장려하는 반복적이고 점증적인 접근 방식이다. 조직은 빠른 응답 속도의 이점을 누리기 위해 애자일 방법론을 선택한다[1, 2].

개발비 감정 및 비용 추정을 위해 제안된 추정 기법은 다양하지만, 애자일 소프트웨어 개발에 대해 일반적으로 인정되는 표준화된 관행은 존재하지 않는다. 애자일 개발 방법론의 주요 과제 중 하나는 공수 추정이며 가장 일반적인 방식은 전문가 의견을 사용하는 것이며, 스토리 포인트는 애자일 접근 방식을 활용하는 조직에서 필요한 개발 공수를 추정하는 데 자주 사용되는 입력 값이다. 스토리 포인트를 할당하기 위해 전문가들이 모여 개발할 소프트웨어에 대한 주관적인 값을 지정하여 예상 공수와 실제 공수를 비교할 수 있다. 그러나 스토리 포인트는 개발된 소프트웨어의 규모를 명시적으로 제공하지 않으므로 프로젝트에서 달성한 생산성을 객관적으로 판단하는 데 사용할 수 없다. 또한 프로젝트 및 조직 전반의 성과를 비교하는 데 사용할 수 없다[3].

전통적으로 소프트웨어의 크기를 측정하는 방법으로는 소프트웨어 코드 라인이 많이 사용되지만, 기능 점수도 표준화된 방법이고 초기 단계에서 유용성이 높다는 장점이 있어 많이 사용된다. 전통적으로 소프트웨어의 기능 규모를 측정하는 방법을 제공하는 여러 국제 표준이 존재한다. 이러한 국제 표준은 프로젝트, 기술 및 개발 전반의 생산성 및 성능 비교 프로세스에 관한 것으로 ISO에서 채택한 5가지 기능 규모 측정(FSM) 방법으로 1세대 FSM 방법이 있고 대표적으로

IFPUG의 기능 점수가 있으며, 국내에서 개발비 감정이나 비용 추정에서 흔히 널리 사용되는 기능 점수 정통법과 간이법도 이에 해당한다. 최근에 업계에서 사용 비율이 늘어나고 있고 학계에서 관심을 끌고 있는 COSMIC 기능 점수는 2세대 방법으로 인정받은 기법이다.

본 연구는 스토리 포인트와 COSMIC 기능 점수를 사용한 노력 추정 모델의 성능을 비교하여 애자일 프로세스에서 사실 상의 표준 척도인 스토리 포인트의 개선 방안을 탐구한다. 본 논문의 구성은 다음과 같다. 2장에서는 플래닝 포커와 스토리 포인트, 그리고 COSMIC 측정 방법에 대한 개요를 설명한다. 3장에서는 다양한 사례 연구를 통해 COSMIC 기반 추정 모델과 스토리 포인트 기반 추정치를 비교한다. 4장에서는 애자일 프로세스에서 가장 널리 이용되는 스토리 포인트가 가져야 할 척도로서의 특성에 대해 기술한다. 5장에서는 결론 및 연구의 한계점과 향후 연구 방향을 제시한다.

2. 스토리 포인트와 COSMIC 기능 점수

애자일 소프트웨어 개발 방법의 기본 개념은 짧은 피드백 루프를 통해 소프트웨어를 신속하게 인도하는 것이다. 애자일 소프트웨어 개발의 장점은 적응성을 우선시하므로 개발 프로세스 후반부에도 요구사항, 설계, 우선순위를 변경할 수 있어 팀이 고객이나 시장의 요구에 신속하게 대응할 수 있다.

2.1 스토리 포인트

일반적으로 피처의 크기나 복잡도의 상대적인 측정인 스토리 포인트를 사용해 사용자 스토리를 추정한다.

표 1. 제품 백로그의 예
Table 1. Example of product backlog

업무 구분	상위 기능	ID	스토리	스토리 점수
인터넷 서점 V1.0	사용자 로그인	A1	사용자는 회원 ID와 비밀번호를 입력하고 로그인할 수 있다.	2
		A2	비회원은 회원 가입 없이 1회 로그인을 할 수 있다.	1
	도서 검색 (제목별, 저자별)	A3	사용자는 제목별, 저자별로 도서를 검색할 수 있다.	3
		A4	사용자는 선택한 책의 상세 정보(출판사, 발행일, 간단한 설명 등)를 볼 수 있다.	2
	도서 주문	A5	사용자는 배송지 주소를 입력하여 책을 구입할 수 있다.	2
		A6	사용자는 검색한 도서 목록에서 원하는 도서를 장바구니에 담을 수 있다.	3
		A7	사용자는 자신의 장바구니를 조회하고 수량을 변경, 삭제할 수 있다.	2

예를 들어 A2 스토리를 구현하는 투입공수가 표 1[4]의 제품 백로그에서 가장 적다면 이를 1점으로 설정한다. 이때 1점에 해당하는 평균 투입공수를 추정하는 것이 바람직하다. 팀에서 역량이 중간 정도인 사람이 사용자 스토리를 구현할 때 소요되는 이상적 작업 시간이 평균 투입공수를 의미한다. 1점짜리 스토리에 소요되는 투입 노력은 1점당 평균 투입공수의 추정치로 가능하다. 또한 A2 스토리를 기준으로 다른 스토리에 상대적인 점수를 부여할 수 있다.

A1 스토리를 구현하는데 들어가야 할 투입공수가 A2에 비해 2배 정도 더 필요할 것 같다면 2점을 주는 것이 직관적이다. 마찬가지로 A3 스토리를 A2 스토리 점수와 비교하여 3점을 줄 수 있다. 이때 정확한 투입공수를 추정하는 것이 아니라 상대적 비교로 추정을 한다. 스토리의 규모를 추정하는 것이 스토리 점수이지 공수를 추정하는 것은 아니다. 추정하기 쉬운 것부터 추정해 나가고 어려운 스토리는 건너뛰는 것이 바람직하다. 이런 방식으로 일관성 있게 스토리를 정한 후 해당 스토리들을 기준으로 삼는다.

경우에 따라 비교하기 어려운 스토리에 대해서는 평균 투입공수를 추정한 후 스토리 점수로 환산한다. 스토리 간에 규모의 일관성을 유지하는지 점검하고 필요하다면 다시 추정한다. 예를 들어 A5와 A6 스토리를 처음에는 3점과 2점으로

추정했지만 추후 과소 추정했다는 것을 인식하면 점수를 수정할 수 있다.

투입공수는 순수하게 해당 요구사항을 수행했을 때 소요되는 노력이다. 프로젝트에서 스토리의 규모는 변하지 않지만 투입공수는 누가 하느냐에 따라 달라질 수 있다. 따라서 스토리 포인트를 일관성 있게 유지하려면 반드시 상대평가를 하여 추정해야만 한다. 프로젝트에서 스토리 포인트를 추정할 때는 평균 투입공수를 스토리 포인트와 동일하게 가져갈 수도 있지만 나중에 새롭게 발생하는 스토리는 기준이 되는 스토리와 상대 비교를 해야 일관성을 유지할 수 있다.

2.2 플래닝 포커

플래닝 포커 기법은 애자일 개발에서 가장 많이 사용하는 추정 기법 중의 하나로 착수 준비와 추정 대상 토론, 추정 수행 단계로 구분할 수 있다. 사용자 스토리를 추정하는 일반적인 기술인 플래닝 포커 기법은 전문가 의견, 분해, 유추를 오락과 결합한 방안으로 스토리가 생길 때마다 사용한다.

표 2. 스토리 포인트의 예
Table 2. Example of story point

구분	라운드	이한나	김영태	박해자	김명진	김지홍	추정값
A 스토리	1라운드	8	8	2	5	13	8
	2라운드	8	8	5	8	8	
B 스토리	1라운드	2	3	20	5	8	13
	2라운드	3	5	13	8	8	
	3라운드	8	8	13	13	13	

표 2에서 A 스토리를 추정할 때 1라운드에서 가장 큰 값과 가장 작은 값의 차이는 11이지만 2라운드에서는 3으로 좁혀진다[4].

일반적으로 첫 번째 추정에서 큰 차이를 보이지만 추정 근거에 대해 논의하기 위해 외부 투표자들에게 물어봄으로써 스토리의 불확실성과 가정을 알게 해준다. 그 결과로 세부사항을 논의하

고 개선할 수 있다. 2~3라운드의 진행에도 불구하고 차이가 좁혀지지 않으면 생략한 후 다른 항목을 진행한다. 정보가 불충분한 경우 가정 자체가 다르기 때문이다. 다른 스토리를 진행하는 과정에서 자연스럽게 이견이 줄어들어간다.

2.3 스토리 포인트와 피보나치 수열

스토리 포인트는 스토리의 크기와 복잡도에 대한 신속하고 상대적인 추정을 제공함으로써 완벽하게 정확한 것은 아니지만, 용인할 정도의 적절한 수준으로 받아들여지고 있다. 스토리 포인트 추정에 사용할 숫자 집합을 정의하는 것이 계획에 선행되어야 한다. 사용하는 숫자 간격을 작은 숫자에서는 가깝게, 큰 숫자에서는 멀게 설정한다. 그 이유는 규모가 20점과 21점인 스토리 간의 차이를 판단하는 능력이 1점과 2점인 스토리 간의 차이를 판단하는 능력과는 다르기 때문이다.

일반적으로 유용한 스토리 포인트의 집합은 피보나치 수열로부터 유도된다. 만약 아주 적은 노력을 필요로 하는 사소한 스토리에는 0점을 부여한다[5].

2.4 COSMIC 기능 점수

COSMIC 기능 점수 방법은 소프트웨어의 기능적 사용자 요구사항을 측정하는 국제 표준(ISO 19761)으로 측정 결과는 소프트웨어 자체의 규모를 나타내는 값으로, 벤치마킹 및 추정 목적으로 사용할 수 있다. COSMIC에서 기능 프로세스는 측정 대상 소프트웨어의 기능적 사용자 요구사항을 나타내는 일련의 데이터 이동으로 데이터 그룹의 기능적 흐름 내에 포함된다. 데이터 그룹의 네 가지 이동 유형은 소프트웨어의 기능 사용자와 COSMIC 기능 사용자 간의 데이터 그룹 진입(E) 및 종료(X) 데이터로 COSMIC 기능 프로세스를 통해 소프트웨어 경계를 넘어 기능

사용자와 데이터를 교환할 수 있다. COSMIC 기능 프로세스와 영구 스토리지 간의 데이터 그룹의 읽기(R) 및 쓰기(W) 데이터 이동은 스토리지 하드웨어와의 데이터 교환을 허용한다[6]. 이러한 네 가지 유형 중 하나에 해당하는 각 데이터 이동에는 단일 크기가 할당되어 IFP가 된다.

완료된 프로젝트의 베이스라인을 COSMIC으로 측정하면 팀 생산성을 과거 데이터와 추정을 기반으로 도출할 수 있고, 선형 또는 비선형 회귀와 같은 다양한 통계 기법을 사용하여 모델을 구축할 수 있다[7].

3. 사례 연구

기존의 다양한 연구에서는 아직 스토리 포인트와 IFPUG 기능 점수 사이의 관계에 대한 결론이 내려지지 않았다. IFPUG 기능 점수와 스토리 포인트 간에 양의 상관관계가 있음을 발견한 연구도 있는 반면에 음의 상관관계를 발견한 연구도 존재하므로 IFPUG 기능 점수와 스토리 포인트 간의 관계에 대한 결론을 내리기에는 이르다는 것을 확인했다.

3.1 사례 연구 1

사례 연구 1에서는 동일한 팀의 애플리케이션에서 24개의 작업을 실행한다. 9개의 반복을 분석하며, 첫 번째 작업을 선택하고 스토리 포인트 크기를 시간 단위로 추정한다. 각 팀원은 피보나치 수열에서 선택한 값을 이용한다[8].

총 추정 노력은 1,329시간, 작업당 평균 55시간에 해당한다. 이 데이터 세트의 작업별 실제 총 노력은 1,051시간으로 작업당 평균 44시간에 해당한다.

스토리 포인트로부터 도출된 추정치와 실제 노력 간의 관계는 그림 1과 같이 선형 회귀 모델

로 나타낼 수 있으며, 여기서 스토리 포인트는 x축에, 실제 공수는 y축에 해당한다.

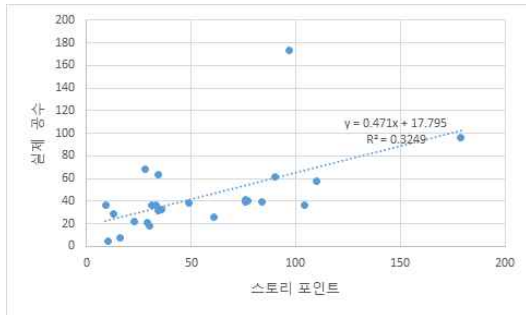


그림 1. 사례 1의 스토리 포인트 추정
Figure 1. Story point estimation of case 1

결정 계수 R^2 가 0.32에 불과하다. 즉, 이 조직에서는 스토리 포인트가 작업을 구현하는 데 필요한 노력에 대한 합리적인 추정치를 제공하지 못하며, 과소 추정과 과대 추정이 모두 심각하여 MMRE가 58%라는 큰 결과를 초래하여 산정에 적절한 척도라고 하긴 어렵다.

상기 동일한 24개 세트의 기능 규모를 COSMIC 방법으로 측정하였다. 모든 작업의 기능적 규모 측정은 동일한 전문가에 의해 수행되었다. 총 기능 규모가 COSMIC FP 측정 단위로 제시되며, 작업의 기능 규모는 2 FP에서 72 FP까지 다양, 평균 규모는 20 FP에 해당한다.

COSMIC 기능 점수를 독립 변수로, 실제 공수를 종속 변수로 하는 추정 모델을 제시하면 그림 2와 같고 COSMIC FP 규모는 x축, 실제 공수는 y축에 해당한다. 스토리 포인트를 추정 목적으로 사용했을 때의 32%와 달리 결정 계수 $R^2 = 0.782$ 는 COSMIC 규모가 공수 변화의 78%를 설명한다는 것을 알 수 있고, COSMIC 기반 추정 모델의 상대 오차(MMRE)의 평균 크기는 28%로, 스토리 포인트 추정 모델의 MMRE인 58%보다 훨씬 우수하였다.

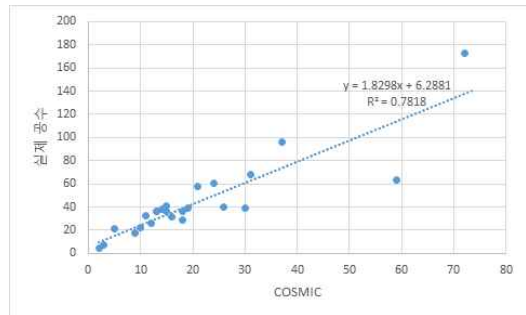


그림 2. 사례 1의 COSMIC FP 추정
Figure 2. COSMIC FP estimation of case 1

스토리 포인트는 소프트웨어 작업의 산출물을 측정하지 않으므로 단위 노력 또는 생산성 비율로 프로젝트 성과를 계산하고 비교할 수 없다. 반면에 COSMIC 기능 점수를 사용하면 각 작업의 성과를 계산할 수 있다. COSMIC 방법이 표준 측정 방법을 제공하고 있지만, 특히 이 사례 연구에서는 모든 규모 측정은 동일한 전문가에 의해 수행되었기 때문에 일관성 있는 값을 얻을 수 있었다.

3.2 사례 연구 2

2번째 사례 연구에서 분석된 데이터[9]는 웹 포털 프로젝트의 스프린트 10개이며 6명의 소프트웨어 개발자와 2명의 소프트웨어 테스터가 참여했다. 이들은 사용자 스토리에 스토리 포인트를 할당하기 위해 플래닝 포커 기법을 사용했다.

분석에 사용된 데이터에는 커밋 및 완료된 스토리 포인트 데이터와 10개의 스프린트 각각에 소요된 총 공수가 포함되며 사용자 스토리의 COSMIC 크기도 측정되었다. 커밋된 데이터에는 특정 스프린트에 대한 모든 사용자 스토리의 크기가 포함되며, 완료된 데이터에는 성공적으로 완료된 사용자 스토리만 포함된다.

스프린트에 대한 COSMIC 기반 생산성 비율

은 특정 스프린트에 대해 완료된 사용자 스토리의 COSMIC 기능 규모를 해당 스프린트에 소요된 총 노력으로 나누어 계산한다. 마찬가지로 스토리 포인트에 대한 생산성 비율은 특정 스프린트에 대한 스토리 포인트를 해당 스프린트의 완료된 사용자 스토리에 소요된 총 공수로 나누어 계산했다. SP와 COSMIC FP 생산성 비율의 표준 편차를 비교한 결과, FP의 값이 SP 생산성 비율의 값보다 3배 이상 큰 것으로 나타난 결과는 COSMIC 기능점수가 SP에 비해 더 신뢰할 수 있는 추정 입력임을 보여준다.

터로 FP와 SP의 평균 생산성 값 간의 편차를 비교한 결과 FP 생산성은 0.005(평균: 0.11), SP 생산성은 0.023(평균: 0.16)으로 나타났다.

3.3 사례 연구 3

3번째 사례 연구의 데이터[9]는 애자일 개발 접근법을 사용하는 대규모 소프트웨어 조직에서 유도되었다. 이 사례 연구는 실제 사례에서 COSMIC FP와 전문가 판단을 기반으로 한 추정치의 효율성을 비교하는 것을 목표로 한다. 스토리 포인트 값은 공수 추정치로 직접 변환되므로 이 사례 연구에서는 SP 값 대신 노력 추정치를 사용했다.

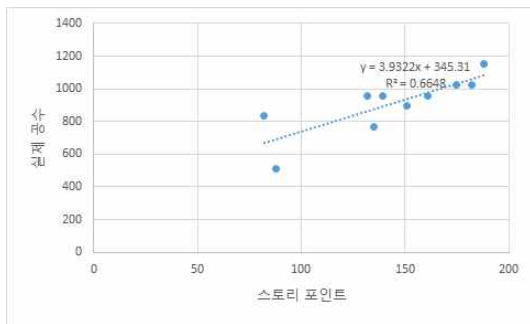


그림 3. 사례 2의 스토리 포인트 추정
Figure 3. Story point estimation of case 2

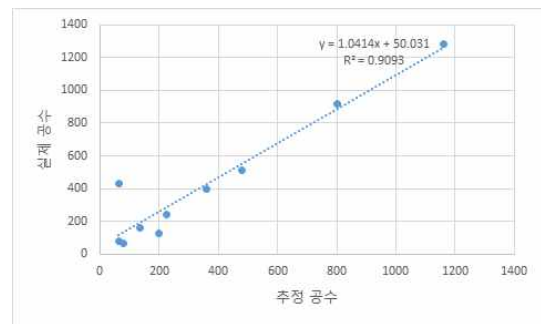


그림 5. 사례 3의 공수 추정
Figure 5. Effort estimation of case 3

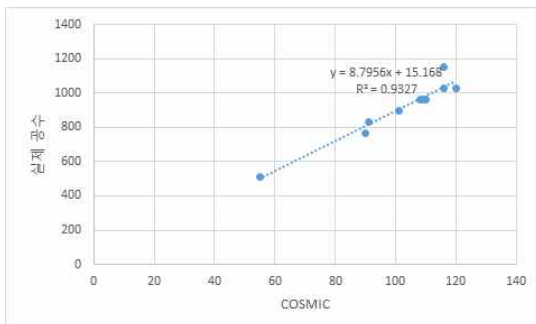


그림 4. 사례 2의 COSMIC FP 추정
Figure 4. COSMIC FP estimation of case 2

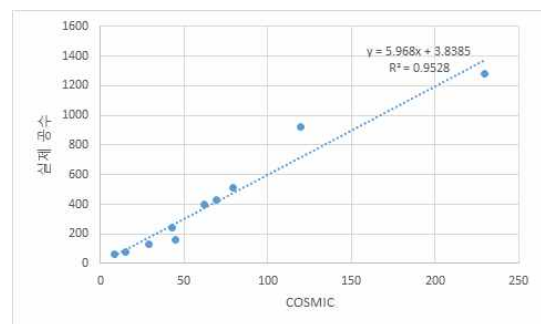


그림 6. 사례 3의 COSMIC FP 추정
Figure 6. COSMIC FP estimation of case 3

실제 공수에 대한 SP 및 FP 값을 그래프로 그려 분석한 결과는 그림 3, 4와 같다. 주어진 데이

그림 5, 6의 회귀 모델이 유도되었는데, 하나는 COSMIC FP를 독립 변수로 사용하고 다른 하나는 예측된 공수 독립 변수로 사용했으나 모두 실제 공수 값을 종속 변수로 사용했다. 두 모델 모두 만족할 만한 결과를 제공했지만, COSMIC 크기를 사용한 모델이 더욱 유의미하고 MMRE 및 PRED(30) 값이 더 나은 결과를 보여주었다.

3.4 사례 연구 4

4번째 사례 연구의 데이터[9]는 주로 금융 및 은행 산업을 위한 소프트웨어를 개발하며 스크럼 방법론을 활용하고 있는 조직에서 유도되었다. COSMIC 기능점수를 사용하여 유도된 공수 추정 모델을 스토리 포인트를 사용하여 유도된 공수 추정 모델과 비교한다. 스토리 포인트 값은 공수 추정치와 직접적으로 관련이 있지만 추정된 공수값은 제공되지 않는다.

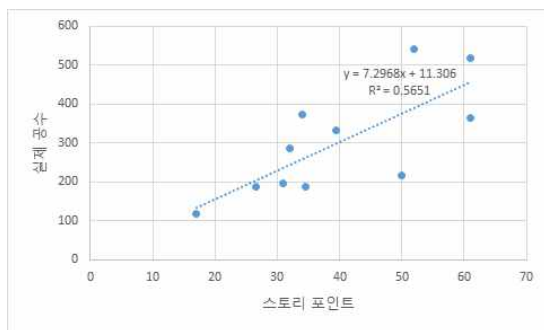


그림 7. 사례 4의 스토리 포인트 추정
Figure 7. Story point estimation of case 4

분석 결과는 그림 7과 그림 8과 같다. 다른 사례와 마찬가지로 FP와 실제 공수를 비교하면 더 나은 선형 방정식을 얻을 수 있지만, SP와 FP 모두 이전 사례에 비해 훨씬 작은 R² 값을 제공한다. 그래프를 보면 프로젝트의 분포가 다소 분산되어 있음을 알 수 있다.

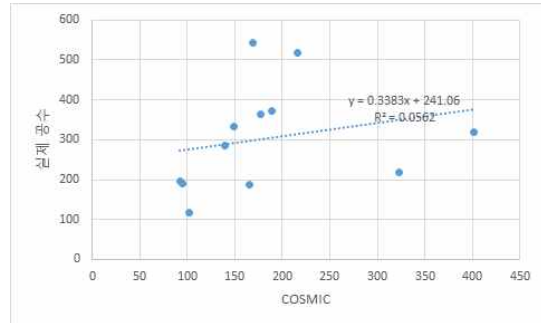


그림 8. 사례 4의 COSMIC FP 추정
Figure 8. COSMIC FP estimation of case 4

사례 4의 경우 FP 생산성의 표준편차는 0.17 (평균: 0.56)이고 SP 생산성은 0.03(평균: 0.14)이다. 이 값은 두 경우 모두 FP 생산성이 SP 생산성보다 덜 분산된 분포를 따르고 있음을 보여준다.

4. 스토리 포인트의 개선 과제

4.1 스토리 포인트의 문제

소프트웨어 프로젝트의 개발비 감정 및 비용 추정에서 가장 중요한 요소는 감정 대상 소프트웨어의 규모이다. 소프트웨어 규모에 대한 공통된 기준이 없으면 프로젝트를 추정하기는 어렵고 관리하기도 매우 어렵다.

앞의 사례 연구들에서 살펴본 것처럼, 스토리 포인트와 COSMIC 기능 점수를 비교하면 스토리 포인트는 규모 추정의 일관성을 유지하기 어렵다[10]. 스토리 포인트 추정치는 작업을 수행하는 팀에 따라 달라질 수 있다. 스토리 포인트는 실제로 어떤 작업을 수행하기 위한 예상 노력의 대리인으로 규모에 대한 느낌을 제공한다. 팀원들이 어떤 작업을 수행하는 데 필요한 노력의 양에 대한 의견은 서로 다를 수 있으므로 스토리 포인트의 규모가 달라질 수 있다. 따라서 스토리

포인트의 규모 추정치는 다른 프로젝트나 조직 또는 다른 업계는 물론 다른 팀과 비교하는 데에도 의미가 없을 수 있다. 대부분의 조직에서 프로젝트의 특성을 다른 프로젝트와 비교하거나 학습하기 위해 스토리 포인트를 사용하는 것은 사실상 불가능하다.

사례 연구에서 분석에 이용된 데이터의 경우 스토리 포인트는 애자일 프로젝트를 수행하는 각각 상이한 사람들에 의해 추정된 값들이지만, COSMIC FP의 경우 동일한 사람들에 의해 추정된 값이라는 점도 분석 결과에 어느 정도 영향을 주었다. 물론 COSMIC FP나 전통적인 FP 혹은 국내 대가산정 가이드라인의 간이법의 경우 표준적인 산정 방안이 제시되었으므로 상이한 사람들이 추정하더라도 일관성을 어느 정도 유지할 수 있다는 장점을 가지고 있다. 스토리 포인트의 경우도 객관적인 척도 기준 및 측정 방안을 제시할 필요가 있다.

소프트웨어 프로젝트를 관리할 때 주요 관심사는 소프트웨어 규모, 공수, 시간, 품질 등으로 조직은 이들에 대한 일관된 척도가 필요하다. 규모를 알면 품질, 일정 및 자원에 대한 적절한 지표를 신속하게 결정할 수 있으며, 이 중 가장 중요한 지표는 규모이다. 스토리 포인트는 소프트웨어 규모와 공수에 관한 뚜렷한 관점을 제공하지 않고 있으므로 이러한 측면에서도 일관된 관점을 제시할 필요가 있다.

애자일 프로세스의 표준 척도에 해당하는 스토리 포인트는 외주 개발 계약에는 적합하지 않다. 예를 들어, 스토리 포인트의 규모에 대한 일관된 기준이 없기 때문에 특정 분량의 스토리 포인트를 제공하기로 계약할 수 없다. 그 결과 고객은 가시성을 확보할 수 없고 통제할 방법도 부족하므로, 마찬가지로 일관된 기준을 제시해야 한다.

4.2 스토리 포인트의 가능성

스토리 포인트 합의 형성 과정에서 얻을 수 있는 이점이 있다. 소프트웨어 프로젝트에 관련된 모든 분야의 실무자를 한자리에 모아 추정할 때 모든 사람이 자유롭게 자신을 표현하고 제품 관리자와 개발 팀 간의 제품 관리자와 개발팀 간의 교류를 촉진할 수 있다. 팀에서 가장 영향력 있거나 목소리가 큰 사람의 의견에 좌우되지 않고 열린 토론을 장려한다. 이를 통해 모든 팀원의 경험을 활용할 수 있다.

반면에 몇 가지 약점은 대부분 주관적인 특성과 관련이 있다. 동일한 백로그가 다른 스크럼 팀에 제공될 때 추정치가 달라질 수 있으며 여러 팀이 동일한 백로그에 대해 함께 작업하는 경우 의견 충돌이 발생할 수 있다. 추정치는 기술과 경험에 따라 달라져 숙련도와 경험이 많은 사람은 해당 분야를 처음 접하는 사람보다 더 낮은 값을 제공할 가능성이 높으며, 이로 인해 추정치를 왜곡할 수 있다. 아직 잘 문서화된 경험적 연구와 실험적 연구가 부족하다. 본 연구와 같이 스토리 포인트 추정치를 실제 프로젝트 공수와 비교하거나 스토리 포인트를 COSMIC FP와 같은 다른 추정 기법과 비교하여 스토리 포인트의 개선 사항을 찾는 연구가 필요하다.

스토리 포인트는 성과나 진행 상황을 효과적으로 측정하는 데 도움이 되는 특징을 가져야 한다. 스토리 포인트가 비용 추정을 위한 지표가 되기 위해서는 비용 추정 목표와 직접적으로 일치해야 하며 원하는 결과에 실질적인 영향을 미치는 것을 측정해야 한다. 이해하기 쉽고 해석하기 쉬워야 한다. 이해하기 어렵거나 복잡한 지표는 혼란과 오해를 불러일으킬 수 있다. 스토리 포인트가 실시간 또는 정기적으로 업데이트가 가능하다면 신속한 조정 및 의사 결정과 연계될 수 있다. 스토리 포인트는 서로 다른 시간대, 팀 또는 부서 간에 일관되고 비교 가능한 일관성을 제

공해야 하며 스토리 포인트를 수집하고 분석하는데 지나치게 많은 자원이 소모되지 않아야 한다. 스토리 포인트는 조직의 보다 광범위한 전략적 목표와 연계되어야 중요한 목표를 향한 진행 상황을 측정하는 데 도움이 될 수 있다. 스토리 포인트를 둘러싼 맥락을 이해하는 것이 중요하다. 스토리 포인트가 성공적인 척도가 되기 위해서는 스토리 포인트를 통해 얻은 결과가 개선으로 이어지고, 이는 다시 향후에 영향을 미치는 긍정적인 피드백을 촉진할 수 있다.

5. 결론

본 연구에서는 애자일 프로세스의 대표적인 추정 기법인 스토리 포인트와 COSMIC 기능 점수를 이용한 비용 추정 모델을 분석했다. 기능 점수 기반 추정 모델을 구축하면 비교 가능한 객관적인 규모 측정을 제공할 뿐만 아니라 신뢰성을 높일 수 있다. 사례 연구를 통해 상이한 조직의 추정치와 추정치를 사용하여 스토리 포인트와 COSMIC 기능 점수를 이용한 추정 모델을 사용하는 효과를 분석했다.

스토리 포인트를 사용한 추정치는 MMRE가 58%에 달하는 등 과소 및 과대 추정하는 것으로 나타났다. 또한 스토리 포인트는 소프트웨어 작업의 산출물 크기를 측정하지 않기 때문에 작업 성과를 계산하고 비교하는 데 사용할 수 없다는 점을 확인하였다. 완성된 프로젝트의 기능 규모를 COSMIC 기능 점수 방식으로 측정했을 때, 단위 노력 또는 생산성 비율을 사용하여 생산성을 계산하고 비교하고 그 후 더 나은 추정을 위해 구축된 모델을 스토리 포인트 추정과 비교하였다. COSMIC 기반 추정 모델의 MMRE가 28%로 분산이 훨씬 더 작음을 알 수 있다. 그 결과 COSMIC 기능 점수를 사용한 추정 모델이 편차

가 훨씬 작다는 결론을 내릴 수 있다. 또한 스크럼 환경에서 COSMIC 기능 점수를 활용하면 한 작업에서 다른 작업으로 생산성을 객관적으로 비교할 수 있다. 마지막으로, COSMIC 기능 점수를 사용한 추정 모델은 주관적인 판단이 배제된 객관적인 모델로 COSMIC 기반 추정 모델을 사용할 수 있다.

스토리 포인트는 현재 애자일 커뮤니티에서 널리 인정받고 사용되지만, 스토리 포인트에 기반한 추정은 프로젝트 참여자의 경험에 직접적으로 의존하기 때문에 객관적인 추정치를 제공하지 못하므로 개선 방안을 찾을 필요가 있다.

본 연구에는 몇 가지 한계점을 가지고 있다. 본 연구에서 분석한 사례는 서로 독립적으로 설계되었기 때문에 데이터 수집 및 평가 방식에 차이가 있다. 스프린트를 기반으로 한 사례와 프로젝트를 기반으로 한 사례로 인해 제시된 프로젝트와 스프린트의 분해 수준이 동일하지 않을 수 있다. 또한 모델을 테스트하기 위한 추가 프로젝트가 없었기 때문에 모델이 구축된 프로젝트와 동일한 프로젝트에서 모델을 테스트했다.

향후 연구에서는 기존 문헌에서 수집한 데이터가 아닌 실제 현장에서 수집한 데이터의 표본 크기를 늘리고 이를 이용하여 스토리 포인트, COSMIC 기능 점수, IFPUF 기능 점수 등의 다양한 규모 척도를 사용하여 추정 모델에 영향을 미칠 수 있는 애자일 프로젝트 속성을 파악하기 위해 유사한 분석을 수행한 후 스토리 포인트의 추정 정확도를 향상시킬 수 있는 개선 방안을 제시할 계획이다.

참고 문헌

- [1] Mike Cohn, Agile Estimating, Prentice-Hall, 2006, ISBN: 0-13-147941-5

- [2] Haithem Kassem, Khaled Mahar, Amani A. Saad, Story Point Estimation Using Issue Reports With Deep Attention Neural Network, *e-Informatica Software Engineering Journal*, 17(1), 2023, DOI: 10.37190/e-Inf230104
- [3] Christopher W. H. Davis, *Agile Metrics in Action*, MANNING, 2015, ISBN: 9781617292484
- [4] Jaewang Lee, *Understanding Agile & Scrum Project Management*, Gilbut, 2016. ISBN: 9761187345015
- [5] Clinton Keith, *Agile Game Development with Scrum*, Addison-Wesley, 2010. ISBN: 9780133943030
- [6] Sang-Ki Park, Man-Gon Park, A Model to Estimate Software Development Effort Based on COSMIC-FFP Using System Complexity, *Journal of Korea Multimedia Society* 13(11), Nov. 2010, pp. 1575-1585, <https://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE01606799>
- [7] Eduardo Rodríguez Sánchez, Eduardo Filemón Vázquez Santacruz and Humberto Cervantes Maceda, Effort and Cost Estimation Using Decision Tree Techniques and Story Points in Agile Software Development, *Mathematics* 2023, 11(6), <https://doi.org/10.3390/math11061477>
- [8] Christophe Commeyne, Alain Abran, Rachida Djouab, Effort Estimation with Story Points and COSMIC Function Points - An Industry Case Study, *Software Measurement News*, 21,(1), pp 25-36, 2016, <https://isbsg.org/wp-content/uploads/2017/09/Comyene-Abran-Estimation-model.pdf>
- [9] Erdir Urgan, Numan Çizmeli, Onur Demirörs, Comparison of Functional Size Based Estimation and Story Points, Based on Effort Estimation Effectiveness in SCRUM Projects, 40th EUROMICRO Conference on Software Engineering and Advanced Applications, Nov. 2014, DOI: 10.1109/SEAA.2014.83
- [10] Murat Salmanoglu, Tuna Hacaloglu, Onur Demirörs, Effort Estimation for Agile Software Development: Comparative Case Studies Using COSMIC Functional Size Measurement and Story Points, Conference: IWSM/Mensura '17At: Gothenburg, Oct. 2017, DOI: 10.1145/3143434.3143450

— 저 자 소 개 —



권기태(Ki-Tae Kwon)

1986.2 서울대학교 계산통계학과 졸업
 1988.2 서울대학교 계산통계학과 석사
 1993.8 서울대학교 계산통계학과 박사
 1995.8-1996.7 University. of Southern California. Post-Doc.
 1990.9-현재 : 강릉원주대학교 교수
 <주관심분야> 소프트웨어공학, 데이터사이언스