

Open Source 커뮤니티에서의 개발 기간 산정에 관한 연구

변정원, 김지혁, 류성열

승실대학교 컴퓨터학과

A Study on Estimation of Development Duration in Open Source Community

Byun, Jung-Won, Kim, Ji-Hyeok, Rhew, Sung-Yul

Dept of Computing, Soong-Sil University

E-mail : {Jimi01, jhkim78, syrhew}@ssu.ac.kr

요약

오픈 소스 소프트웨어에 대한 관심과 함께 많은 연구가 진행 중이며, 오픈 소스 소프트웨어 개발을 지원하는 커뮤니티도 다수 존재하고 있다. 오픈 소스를 활용한 개발비용을 예측을 위해선 오픈 소스의 개발 기간을 예측하여야 하지만, OSS 개발 방법에는 개발 기간을 산정하는 것이 미비하다. 그리하여 본 연구에서는 OSS 커뮤니티에서 개발기간을 산정하는 것에 대하여 XP 개발 방법의 사용자 스토리 카드를 이용한 산정법을 적용하여 활용하는 연구를 하고자 한다. 이를 위해, XP의 산정법을 OSS 커뮤니티에 적합하게 수정하였다. 또한 제안한 방법을 실제 SourceForge.net의 프로젝트 중 버그와 추가사항의 기록이 많은 프로젝트에 적용하여 그 가능성을 입증하였다.

1. 서론

오픈 소스 소프트웨어(Open Source Software, OSS)는 비용 절감과 기간 단축이라는 점에서 최근 각광받고 있다. 이러한 OSS 개발을 지원하기 위한 많은 OSS 커뮤니티가 존재한다.

사용자가 배포된 OSS를 이용하여 추가적인 기능 개발 및 유지 보수의 비용과 일정을 수립하기 위해선, OSS 커뮤니티는 사용자의 요구사항에 대한 개발기간을 산정하고, 정보를 제공해야 한다. 이 정보를 가지고 사용자는 요구사항을 OSS 개발 방식을 통해 개발할 것인지, 직접 개발할 것인지 결정할 수 있다. 하지만 OSS 커뮤니티가 개발 기간과 관련된 정보를 제공하는 것이 미비하다[1].

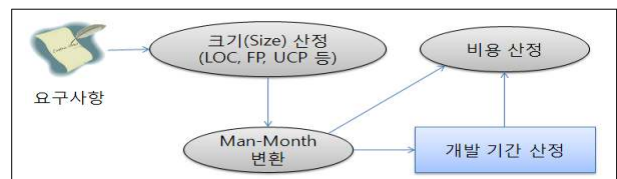
본 연구는 OSS 커뮤니티의 과거 완료한 요구사항의 개수를 기반으로 사용자가 작성한 요구사항이 언제 반영될 것인가를 산정하는 방법을 제안하는 것을 목적으로 한다. 개발기간을 산정하기 위해 OSS 커뮤니티에 기존의 개발 방법을 연구하였으며, 그 결과 XP의 개발 방법이 OSS 개발 방법과 유사하였다. 그리하여 XP의 스토리 카드

를 이용한 개발기간 산정법을 적용하고자 한다.

2. 관련 연구

2.1 개발 기간의 산정을 위한 기존의 모델

기존의 개발 기간 산정 모델[2]은 개발에 투입될 노력을 이용하여 개발 기간을 추정하는 연구로, Oligny, Kitchenham, Ferens 등의 연구가 있다. 이 중 Oligny[3]는 개발 플랫폼별로 구분하여 총 소요 노력과 개발 기간의 로그 관계로부터 개발 기간을 추정하였다. 이러한 기존의 방법은 요구사항을 여러 가지 요소로 계산하여 크기를 산정하고, 크기를 기반으로 노력을 산정하며, 노력으로 개발 기간과 비용을 산정한다. 기존의 개발 기간 산정 모델은 (그림 1)과 같다.



(그림 1) 기존의 개발 기간 산정 모델

기존의 개발 기간 산정 모델을 OSS 개발 방식에 적용할 때에 문제점은 다음과 같다.

첫째, 빠르게 변해야 하는 OSS 개발 방식에, 기존의 개발 기간 산정 모델의 요소나 방식이 복잡하다는 것이다. 또한 요구사항이 완료되지 않은 상태임으로, 전체적인 개발 기간을 산정하는 것이 중요한 의미를 가지지 않는다.

둘째, 요구사항이 지속적으로 변한다. 따라서 개발기간을 산정한다는 것이 중요한 의미를 가지지 않는다.

셋째, 개발 인원이 고정되어 있지 않다. 그러므로 노력(Man-Month)을 통한 방법으로 개발기간을 산정하는 것이 어렵다.

2.2 XP의 개발기간 산정

한 번에 모든 요구사항을 얻을 수는 없다. 필요한 것을, 필요한 부분만, 필요한 시점에 작성한다는 점에서 OSS 커뮤니티의 요구사항 작성 방식과 사용자 스토리 카드의 작성 방식[4]이 비슷한 특징[5]을 가진다. 그리하여 사용자 스토리 카드의 개발기간 산정 방식이 OSS 개발 방식의 기간 산정에 적합하였다. 또한 이러한 스토리 카드를 이용하여 개발 기간을 산정하고 예측할 수 있다. 사용자 스토리 카드를 활용하여 개발기간을 측정하는 방식은 개발자들의 만장일치나 회의 방식을 통해 스토리 점수를 결정하고, 이전 반복 주기의 속도에 따라 개발기간을 산정하는 방식이다. OSS 개발 방식에 적용할 때 문제점은 OSS 커뮤니티에서 자발적인 참여라는 점에서 스토리 점수를 결정을 위한 만장일치 또는 회의 방식을 지속하는 것이 어렵다는 점이다.

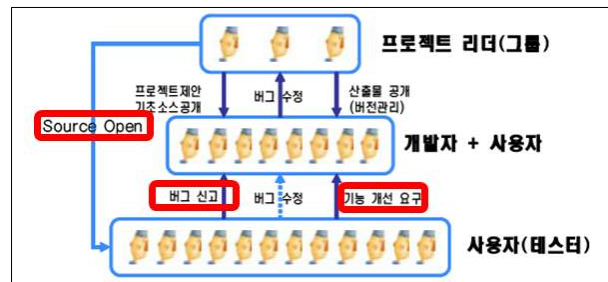
2.3 OSS 커뮤니티의 개발 기간

본 절은 The Cathedral and the Bazaar[6]와 OSS 커뮤니티(SourceForge.net)를 참고하여, 개발 기간 산정을 위한 OSS 커뮤니티의 이해당사자와 산출물을 연구하여, OSS 커뮤니티에서 개발기간에 대하여 연구하였다. (그림 2)는 OSS 커뮤니티 이해당사자와 산출물을 나타낸다.

(그림 2)에서 보는 바와 같이 사용자가 기능 개선이나 버그를 신고하면 개발자가 이를 반영하여 소스를 공개한다. 이렇게 사용자가 요구사항을 신

고한 시점부터 OSS에 이러한 요구사항이 반영될 때까지의 기간을 개발 기간으로 한다.

OSS 개발 방식에서 개발 기간을 산정할 때의 문제는 다음과 같다. 첫째, 개발자의 개발 능력을 알 수 없으며, 개발을 강제할 수 없다. 둘째, 요구사항과 변경은 계속하여 추가된다. 기존의 기간 산정 모델의 복잡한 방법을 OSS 개발 방법에 사용하기 힘들다는 뜻이며, 힘들게 구한 기간이 계속하여 변경된다는 뜻이다.



(그림 2) OSS 커뮤니티 이해당사자와 산출물[7]

3. OSS 커뮤니티에서의 개발기간 산정

관련 연구에서 나타난 문제점을 개선하고, OSS 커뮤니티에 사용자 스토리 카드의 개발기간 산정 방법을 적용하기 위하여, 본 장은 OSS 커뮤니티에서 몇 가지 항목을 추가하고, 개발기간을 산정하는 방법을 적용하였다.

3.1 OSS 커뮤니티 적용을 위한 스토리 카드의 개선

기존의 스토리 카드를 OSS 커뮤니티에 적용할 때의 문제를 해결하기 위하여, 스토리 카드의 항목을 결정하는 방식을 만장일치 또는 회의 방식에서 투표의 방식으로 변환하였다. 이러한 투표를 통하여 얻는 장점은, 대수의 법칙에 따라, 개개인의 반응에서 나타난 측면(개인성, 우연성)이 상쇄된다는 점이다[8]. 대수의 법칙의 전제조건이 많은 개인이 구성된 전체 집단이라는 점에서, OSS 커뮤니티는 이러한 점을 만족한다.

OSS 커뮤니티와 스토리 카드의 항목을 수집하였으며, 다음의 조건을 고려하고, 항목을 선별하여 <표 1>의 항목을 도출하였다. 첫째, 기존의 OSS 커뮤니티의 사용자 및 개발자가 작성하기 쉽고, 단순해야 한다. 둘째, 반복적이고 점진적인

<표 1> OSS 커뮤니티의 개발기간 산정을 위한
요구사항 추가 또는 변경할 항목

항 목	사 용 목 적	작성자
우 선 순 위	개발의 순서이다. 개발 리스트의 출력순서에 영향을 준다.	사 용 자 (투 표)
스토리 점수	측정과 예측에 사용될 측정지표이다.	개 발 자 (투 표)
버전과 완료주기	적용된 버전과 완료된 주기를 표현한다.	커뮤니티
상태	요구사항에 대한 구현 여부와 현재 상태를 표현한다.	프로젝트 리더와 개 발 자

OSS 개발 방법에 적합한 항목이어야 한다.

<표 1>의 작성자는 이 항목을 작성하는 대상이다. 요구사항을 작성하는 사람은 사용자이지만, 개발자와 사용자 모두가 활용할 것이기에 모든 항목을 사용자가 작성할 필요는 없다.

“커뮤니티”라고 하는 것은 직접 작성하는 것이 아니라, OSS 커뮤니티에서 유추할 수 있는 정보임으로 실제 개발자나 사용자가 작성하지 않아도 되는 항목이다.

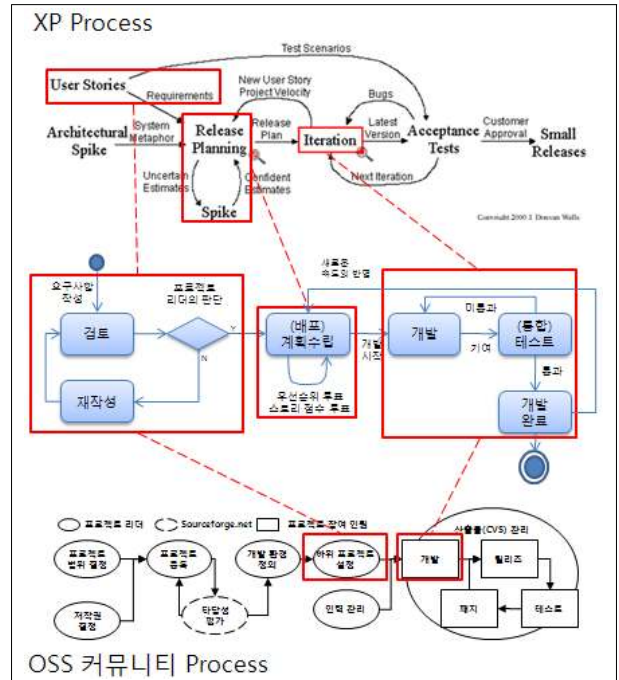
“투표”라고 쓰인 것은 사용자나 개발자가 이러한 항목에 각자가 생각하는 점수나 수치 및 이유를 입력하면, 커뮤니티는 이러한 것을 집계하여 평균한 결과를 사용한다는 것을 의미한다.

3.1.1 요구사항의 분할

개발기간에 사용할 측정지표인 스토리 점수가 큰 경우 개발 기간을 산정하는 신뢰성에 영향을 줄 수 있다. 스토리 점수가 큰 경우 요구사항을 분할하여 작은 스토리 점수들로 나누고 이를 활용하여야 한다. 스토리 점수를 나누어야 할 대상에는 1회 반복주기에 할 수 없을 정도로 큰 요구사항인 경우, 복잡하거나 복합적인 요구사항인 경우가 있다.

3.1.2 요구사항의 상태 변화

(그림 3)과 같이, 상단의 그림인 XP의 프로세스 [4]와 하단의 SourceForge.net의 프로세스를 비교하여, 요구사항의 상태(검토, 재작성, 계획수립, 개발, 테스트, 개발 완료)를 도출하였다. 이 상태는 사용자에게 제공할 정보이며, 또한 개발기간 산정에 활용한다.



(그림 3) 요구사항의 상태 항목의 변화

3.2 개발기간 산정을 위한 OSS 커뮤니티의 개선

개발기간을 산정하기 위하여 기존의 OSS 커뮤니티에 <표 1>의 항목과 개발 리스트, 반복 주기 개념을 도입하여 개선하였다. 이는 2.2 절에서 제시한 문제점을 해결하기 위해 개발자의 개발 능력을 기간을 산정하는 기준으로 하는 것보다 “과거의 개발 이력이 앞으로도 지속될 것이다”라는 가정을 한 것이다. 이렇게 하여 (그림 1)에서 프로젝트의 크기나 Man-Month를 계산하여 기간을 산정하는 것이 아닌, 과거의 기록을 이용하여 빠르고 쉽게 개발기간을 산정한다.

3.2.1 요구사항 항목의 추가

OSS 개발 기간을 산정하기 위해, <표 1>의 항목을 추가한다. 이는 XP에서 사용자 스토리를 통한 기간의 산정법을 적용하기 위함이다.

3.2.2 개발 리스트의 추가

빠르게 변하는 OSS의 요구를 충족시키기 위하여 장기간의 정확한 개발기간을 산정하는 것보다, 필요한 때에 가장 중요한 것을 우선 개발하는 것이 필요하다. 이를 위해, 커뮤니티는 우선순위와

상태 항목을 활용하여 개발자 리스트를 만들고, 개발자의 개발 순서를 적정한 수준에서 제어할 수 있어야 한다.

3.2.3 반복 주기 개념의 추가

OSS 커뮤니티에 반복 주기에 대한 개념을 추가한다. 이는 OSS 커뮤니티의 지속적이고 반복적인 방식에 일정한 기간을 단위로 하는 개념이다.

3.3 개발기간 산정 방법의 적용

개발기간을 산정하기 위해 우선 개발 속도를 계산하여야 한다. 개발 속도를 구하는 공식은 아래와 같다. 괄호는 단위를 나타낸다.

$$\text{속도(점수/일)} = \frac{\text{현재 완료한 점수 합(점수)} / \text{개발 기간(일)}}{\text{일}}$$

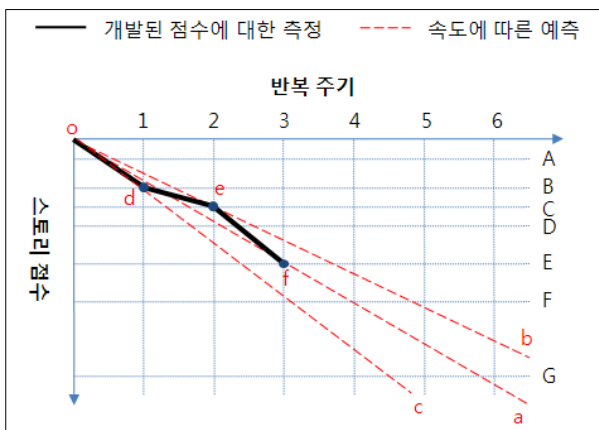
개발기간을 산정하는 공식은 아래와 같다. 괄호는 단위를 나타낸다.

$$\text{개발기간(일)} = \frac{\text{남은 스토리 점수의 합(점수)} / \text{속도(점수/일)}}{\text{일}}$$

남은 스토리 점수의 합을 구하는 방법은 요구사항의 상태 항목을 활용하여 개발 상태와 테스트 상태의 요구사항의 스토리 점수를 합산한다.

속도를 계산하는 방법은 개발된 스토리 점수의 합을 개발 기간으로 나눈 것이다. 개발 시작시의 속도는 0 / 0 임으로 0으로 가정한다. (그림 4)의 선분(oa, od 등)의 기울기와 같다.

사용자는 우선순위, 스토리 점수, 속도를 이용



(그림 4) 3회 반복 결과를 이용한 측정과 예측

하여, 요구사항이 언제 개발 완료될 것인지에 대한 대략적인 개발기간을 산정할 수 있다. (그림 3)에의 반복 주기 3까지의 속도(oa)에 따르면, 스토리 카드 F는 반복 주기 4에서 완료되며, 스토리 카드 G는 반복 주기 6에서 완료됨을 산정할 수 있다. 이러한 산정 방식은 각 반복 주기의 결과가 누적 될수록, 결과를 신뢰할 수 있다는 점이다. 각각의 속도(od, de, ef) 중 무작위로 뽑은 속도를 측정에 사용할 때에 비해, 속도의 평균을 측정에 사용하는 것이 더욱 신뢰할 수 있으며, 대상의 수는 많을수록 신뢰도가 높다는 ‘벨파이 효과’[3]로 설명할 수 있다.

3.4 개발기간을 활용한 배포 계획의 수립

산정한 개발기간을 활용하여 배포 계획을 수립할 수 있다. 개발 리스트와 반복 주기를 이용하여, 다음 주기의 대략적인 배포 계획에 대한 수립도 가능해진다. 배포 계획은 개발자가 지켜야 하는 강제적인 사항은 아니다. 배포 계획은 사용자에게 다음 배포에 어떠한 요구사항이 반영될 수 있는가를 알려주는 정보이고, 과거의 배포 주기에 어떠한 요구사항이 반영되었는가에 대한 기록이기도 하다. 커뮤니티는 작성된 각 항목들을 이용하여 자동적으로 배포 계획을 수립할 수 있다.

3.5 전체 스토리 점수의 예측

(그림 5)의 12월 전체 스토리 점수(프로젝트의 크기)는 다음과 같은 공식으로 계산한다.

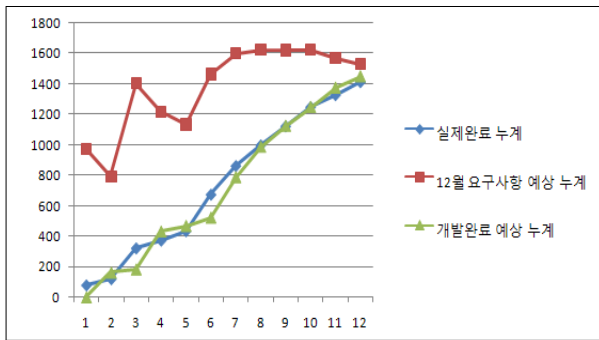
$$\text{전체 요구사항의 예상 스토리 점수} = \text{완료한 스토리 점수} + \text{속도(점수/일)} \times \text{남은 기간(일)}$$

속도는 점차 시간이 지날수록 신뢰도가 높아지고, 또한 남은 기간이 줄어들수록 요구사항의 예측 결과의 신뢰도가 높아짐으로, 공식의 결과는 시간이 지날수록 점차 예측의 신뢰도가 높아진다.

4. 사례적용 및 검증

(그림 5)는 SourceForge.net의 Notepad++라는 프로젝트의 2007년 1월부터 12월까지의 버그와 기능개선 요구의 실제 완료한 개수를 가지고 예측한 그림이다. 버그나

기능의 스토리 점수가 1이고, 반복 주기는 한 달로 가정하여 계산하였다.



(그림 5) 요구사항의 실제 완료 누계와 개발완료 예상 누계

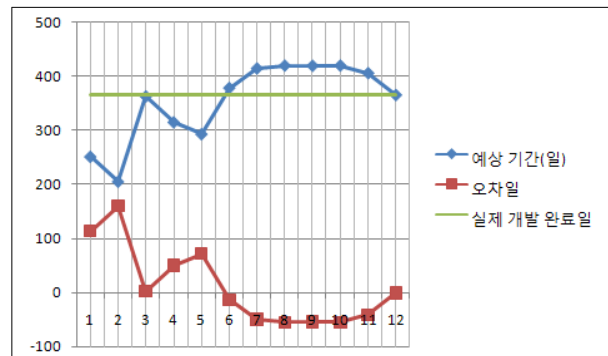
◆ 선분은 실제 완료한 누계이고, ▲는 3.3 절의 속도 공식을 이용하여 계산한 개발완료 예상 누계이다. ■ 선분은 3.5 절의 공식을 이용한 12월까지의 요구사항 예상 개수이다. (그림 5)에서 보는 바와 같이 실제완료 누계와 개발완료 예상 누계가 비슷함을 알 수 있다. 이는 속도가 시간이 지남에 따라 신뢰도가 높아진다는 뜻이다.

(그림 6)은 12월까지의 요구사항 예측 개수를 기준으로 개발기간을 산정하고, 실제 개발기간과 얼마나 차이가 있는가를 나타내는 그래프이다. 양의 수치는 개발기간의 산정이 실제보다 작게 예측했다는 뜻이며, 음의 수치는 개발기간의 산정한 수치가 실제보다 높음을 뜻한다.

(그림 6)에 따른 결과를 보면, 프로젝트 초반에는 개발기간이 작게 예측되지만, 시간이 지남에 따라 요구사항이 늘어가고 버그가 발생함에 따라 개발 기간이 크게 예측됨을 알 수 있다. 그리고 시간이 지남에 따라 점차 실제 결과와 비슷하게 예측됨을 알 수 있다.

5. 결론

본 논문에서는 스토리 카드의 개발기간 산정 방법을 이용하여 OSS 커뮤니티에서 사용자의 요구사항의 OSS에 반영되는 시기를 산정하는 방법과 활용을 제안하였다. 이러한 방법은 다음과 같은 특징을 가진다. 첫째, 기존 커뮤니티의 사용 방법을 크게 변화시키진 않았다. 둘째, 참여적인



(그림 6) SourceForge.net의 사례를

이용한 개발 기간 예측 결과 방법과 과거의 자료를 활용한 방법이다. 마지막으로 지속적으로 변하는 OSS의 요구사항에 따라 일정을 빠르고 단순하며, 커뮤니티에 의해 자동적으로 계산할 수 있다.

향후 연구에서는 Agile의 방법을 적용하여 다른 OSS 개발 방법의 문제를 해결하는 것이다. 그리고 이러한 연구를 실제 OSS 커뮤니티에 반영하여 검증하는 연구이다.

[참고문헌]

- [1] McConnell, "Open Source Methodology: Ready for Prime Time?", IEEE Software 16, pp.6-8, 1999
- [2] 이상운, "소프트웨어 개발기간 추정 모델", 정보과학회논문지, 제 31권, 제 1호, pp.20-28, 2004
- [3] Oligny, S., Bourque, P. and Abran, A., "An Empirical Assessment of Project Duration Models in Software Engineering," In The Eight European Software Control and Metrics Conference (ESCOM'97), Berlin Germany, 1997.
- [4] M. Cohn, User Stories Applied, INSIGHT, 2006
- [5] S. Koch, "Agile Principles and Open Source Software Development", XP 2004, LNCS 3092, pp.85-93, 2004
- [6] Eric S. Raymond, The Cathedral and the Bazaar, O'reilly, 1999
- [7] 홍성욱, "정당과 시장, 오픈소스공동체의 현재와 미래", 정보과학회지, 제 24권, 제 6호, pp.48-58, 2006

[8] 이준구, 이창용, 경제학원론 제 3판, 법문사,
2007