

# 스크럼 마스터 (Scrum Master) 체크 리스트

- 원본: [http://scrummasterchecklist.org/pdf/ScrumMaster\\_Checklist\\_12\\_unbranded.pdf](http://scrummasterchecklist.org/pdf/ScrumMaster_Checklist_12_unbranded.pdf)

## An Example Checklist for ScrumMasters

- Michael James (mj4scrum@gmail.com)
- 14 September 2007 (Revised 2 Feb 2016)
- 역자: 이윤석(@pineoc), 교정: 김승호(@raccoonyy)
- 번역일: 2018.06.03

### 스크럼 마스터는 언제나 조력자(퍼실리테이터, Facilitator)인가?

좋은 스크럼 마스터는 한 번에 두세 팀을 처리할 수 있습니다. 역할을 나눠야 하는 콘텐츠가 있는 경우 회의를 만들고, 마감 시한을 정하고, 사람들이 털어놓는 불만 사항을 대응한다면 여러분은 이 역할에 대한 부분적인 관심으로 그럭저럭 해나갈 수 있습니다. 팀은 여전히 기준선을 초과하고, 조직에서의 스크럼 사전 예측을 초과할 것이며, 그리고 아마도 어떠한 재앙도 일어나지 않을 것입니다.

그러나 변화된 조직 내에서, 이전에 아무도 가능하다고 생각하지 못했던 일을 해내는 팀이 되길 꿈꾼다면 위대한 스크럼 마스터가 되어 보십시오.

위대한 스크럼 마스터는 한 번에 한 팀을 다룰 수 있습니다.

프로젝트를 시작할 때, 일곱 명 정도로 구성된 팀 하나당 전담 스크럼 마스터를 권장합니다.

해당 팀에서 수행해야 할 작업을 모두 찾지 못한 경우에는 제품 소유자, 팀, 팀의 엔지니어링 관행 및 팀 외부 조직에도 문의하십시오.

모든 사람에게 똑같은 약을 처방할 수는 없겠지만, 이 문서에서는 제가 보았던 스크럼 마스터들이 간과하고 있는 대표적인 것들에 대해 설명했습니다.

마지막 페이지에서 설명한 대로 각 항목에 √, Δ, ? 또는 N/A를 표시하기 바랍니다.

### Part 1, 제품 소유자(Product Owner)는 무슨 일을 해야 할까?

스크럼 마스터는 제품 백로그 및 릴리스 계획을 유지 관리할 수 있는 방법을 찾아 제품 소유자가 효율적으로 일하도록 돕습니다.

(제품 소유자는 우선 순위가 지정된 백로그를 담당합니다.)

- 그/그녀의 최근 생각에 따라 제품 백로그의 우선 순위가 결정되니까?
- 모든 이해 관계자의 필요 사항과 요구 사항이 제품 백로그에 포함되어 있습니까? 기억하세요: 백로그는 불시에 생겨납니다.
- 제품 백로그는 관리 가능한 크기입니까? 관리 가능한 수의 항목을 유지하려면 작은 단위의 일을 위로 올리고 큰 단위의 일을 아래 쪽으로 내리는 것을 유지해야 합니다. 제품 백로그의 상단을 지나치게 오래 분석하는 것은 생산성에 방해가 됩니다. 여러분의 요구 사항은 개발 중인 제품에 따라, 혹은 이해 관계자나 고객 간의 지속적인 대화를 통해 달라질 것입니다.
- 요구 사항 (특히 제품 백로그의 상단에 있는 요구 사항)을 독립적이며, 협상 가능하며, 가치 있고, 예측 가능하고, 작고, 테스트 가능한 **사용자 스토리(역서)**로 잘 표현할 수 있습니까?
- 제품 소유자에게 기술 부채와 이를 피하는 방법에 대해 교육한 적이 있습니까? 각 백로그 항목에 대해 "완료"의 정의에 의해 자동화된 테스트 및 리팩터링을 작성하는 것은 매우 중요합니다.

- 백로그는 모든 이해 관계자에게 즉각적으로 보이는 **정보 라디에이터(information radiator)**입니까?
- 백로그 관리를 위해 자동화된 도구를 사용하는 경우, 모든 사람이 그 도구를 쉽게 사용하는 방법을 알고 있습니까? 자동화된 관리 도구는 스크럼 마스터의 적극적인 공유가 없으면 **정보 냉장고(information refrigerators)**가 될 위험이 있습니다.
- 모든 사람에게 출력물을 보여줌으로써 정보를 발산할 수 있습니까?
- 눈에 보이는 큰 차트를 만들어 정보를 발산할 수 있습니까?
- 제품 소유자가 백로그 항목을 적절한 릴리스 또는 우선 순위 그룹으로 구성할 수 있도록 도왔습니까?
- 출시 계획이 여전히 현실과 일치하는지 여부를 모두 알고 있습니까? 스프린트 검토 회의가 진행되는 동안 항목이 "완료"되었다고 인정되면 제품/출시 **변 다운 차트(역서)**를 모두에게 보여줄 수 있습니다. 실제로 완료된 PBI 비율과 새로 추가된 PBI 비율을 보여주는 차트를 통해 범위나 일정이 어긋나고 있음을 조기에 발견할 수 있습니다.
- 제품 소유자가 마지막 스프린트 검토 회의 이후에 릴리스 계획을 조정했습니까? 적절한 테스트를 거친 제품을 **제 시간**에 전달하는 소수의 제품 소유자들은 모든 스프린트의 출시를 다시 계획합니다. 이 때문에 좀 더 중요한 작업이 발견되었다면, 덜 중요한 작업들은 다음번 릴리스로 연기될 수 있습니다.

## Part 2, 우리 팀은 어떻게 하고 있습니까?

팀 구성원들과 협력하여 일하는 사례를 이끌어내는 것이 좋지만, 기술적인 업무에서 길을 잃을 위험이 있습니다. 팀에 대한 주요 책임 사항을 고려하세요.

- 여러분의 팀은 몰입 상태입니까? **몰입 상태(역서)**의 일부 특성입니다.
  - 명확한 목표 (기대와 규칙은 식별 가능하며 목표는 달성 가능하며 기술의 집합과 능력에 따라 적절히 조정됩니다).
  - 집중과 주목, 제한된 관심 분야에 대한 높은 집중.
  - 자의식 감각 상실, 행동과 인식의 융합.
  - 직접적이고 즉각적인 피드백 (활동 과정에서의 성공과 실패는 명백하므로 행동이 필요에 따라 조정될 수 있습니다).
  - 능력 수준과 도전 사이의 균형 (활동이 너무 쉽지도 어렵지도 않은 상태).
  - 상황이나 활동에 대한 개인적인 통제.
  - 이 활동은 본질적으로 보람을 느끼므로 행동하기 쉽습니다.
- 팀원들이 서로를 좋아하며, 농담 따먹기하며 서로의 성공을 축하하는 것처럼 보입니까?
- 팀 구성원들은 서로의 높은 기준에 부합하며, 서로 성장하도록 도전합니까?
- 팀이 너무 **불편(역서)**하기 때문에 논의하지 않는 문제/기회가 있습니까?
- **스프린트 회고 미팅(역서)**의 다양한 형식과 장소를 시도해 보셨습니까?
- 팀은 스프린트 목표에 계속 집중했습니까? 스프린트에 커밋된 제품 백로그 항목의 허용 기준을 다시 검토하려면 스프린트 중간 점검을 수행해야 합니다.
- 스프린트 작업 보드가 팀의 실제 상황을 반영합니까? 하루 분량의 일보다 큰 비공개 과제 같은 "암흑 물질"을 조심하십시오. 스프린트 약속과 관련이 없는 작업은 장애물일 뿐입니다.
- 여러분의 팀에는 잠재적으로 실행 가능한 제품을 빌드할 수 있는 충분한 기술을 보유한 세 명에서 아홉 명 사이의 직원이 있습니까?
- 팀의 작업 보드는 최신의 상태입니까?
- 팀이 사용하기에 편리하도록 팀이 자체 관리 아티팩트들(제품 백로그, 스프린트 백로그, 제품 결과물, 옮긴이)을 볼 수 있습니까?
- 이러한 아티팩트들(제품 백로그, 스프린트 백로그, 제품 결과물, 옮긴이)이 간섭하려는 사람으로부터 적절히 보호되고 있습니까? 하루하루 팀 외부에서 지나치게 감시하려 드는 행위는 팀 내부의 투명성과 자기 관리를 저해할 수 있습니다.
- 팀원들이 스스로 일을 맡습니까?

- 일을 완료했다는 말에 기술 부채를 줄였다는 의미가 명확히 포함되어 있고, 점차적으로 코드가 동작 하기에 더 쾌적한 곳이 되었습니까?
- 팀 구성원이 합의된 작업(테스팅, 문서, 등)의 모든 측면을 총괄적으로 책임지고 자신의 역할을 팀이 사용하는 공간 입구에 남기고 있습니까?

### Part 3, 당사의 엔지니어링 실무는 어떻게 진행되고 있습니까?

- 개발 중인 시스템에 "테스트 푸시" 버튼이 있어(같은 팀이나 다른 팀의) 회귀 실패(이전 작업 기능 중단)를 일으킨 경우 누군가가 편리하게 감지할 수 있습니까? 일반적으로 이것은 xUnit 프레임워크(JUnit, NUnit 등)를 통해 이루어집니다.
- 자동화된 end-to-end 시스템 테스트("기능 테스트"라고도 함)와 자동화된 유닛 테스트 사이에서 적절히 균형을 이루고 있습니까?
- 팀이 개발 중인 시스템과 동일한 언어로 시스템 테스트와 유닛 테스트를 모두 작성하고 있습니까? 협업은 독점적 스크립팅 언어 또는 팀의 일부만 유지 관리 방법을 알고 있는 캡처 재생 도구에 의해 강화되지 않습니다.
- 팀이 시스템 테스트와 유닛 테스트 사이의 유용한 중간 영역을 발견했습니까?
- 누군가 7개의 회귀 실패를 일으킬 경우 지속적인 통합(CI) 서버가 자동으로 알람을 울려줍니까? 이 피드백 루프를 몇 시간 또는 몇 분으로 줄일 수 있습니까? ("일일 빌드는 겁쟁이들이나 하는 것입니다." -- 켄트 벡)
- 모든 테스트가 지속적인 통합 서버 결과에 반영됩니까?
- 8개의 Big Design Up Front(구현 전 완벽한 디자인) 대신에 지속적인 디자인과 지속적인 리팩터링(역사)의 즐거움을 팀원들이 발견했습니까? 리팩터링의 정의는 엄격합니다. 즉, 외부의 행동을 변경하지 않고 내부 구조를 변경하는 것입니다. 리팩터링은 중복 코드, 복잡한 조건 로직(들여쓰기가 많다거나 길이가 긴 메서드), 이해할 수 없는 이름, 객체 간 결합도가 높을 때 등 시간마다 수차례 진행해야만 합니다. 자동화된 테스트가 존재하는 경우에만 리팩터링을 신뢰할 수 있습니다. 리팩터링을 등한시하면 향후 제품을 변경하기가 어려워집니다. 특히 나쁜 코드에서 기꺼이 개발할 좋은 개발자를 찾기 어려울 수 있습니다.
- 각 제품 백로그 항목의 "완료"에 대한 정의에 완전히 자동화된 테스트 적용 범위와 리팩터링이 포함되어 있습니까? 테스트 중심의 개발(TDD)을 학습하면 이 목표를 달성할 확률이 높아집니다.
- 팀원들이 대부분의 시간을 짝 프로그래밍 방식으로 진행합니까? 짝 프로그래밍은 코드 유지 관리 가능성을 획기적으로 높이고 버그 발생률을 줄일 수 있습니다. 그것은 사람들의 한계에 도전하고 때로는 더 오래 걸리는 것처럼 보입니다. (우리가 전달할 기능보다는 코드 라인으로 측정하는 경우) 팀원들과 짝을 이루는 작업을 시작하는 것을 예시로 들어 알려줍니다. 그들 중 일부는 이런 식으로 일하는 것을 더 좋아할 것입니다.

### Part 4, 조직은 어떻게 되어가고 있습니까?

- 팀 간 적절한 커뮤니케이션이 이루어지고 있습니까? "Scrum of Scrums"은 이것을 달성하기 위한 유일한 방법이지만, 최선은 아닙니다.
- 아키텍처 경계를 넘어서(참고), 팀이 독립적으로 동작하는 기능을 생산할 수 있습니까?
- 여러분의 스크럼 마스터가 서로 만나서 조직적인 장애물(impediments) 목록을 작성하고 있습니까?
- 필요한 경우, 조직적인 장애물이 개발 책임자 사무실 벽에 붙어 있습니까? 시장 진출 시간 낭비, 품질 저하 또는 고객 기회 상실과 같은 것을 재화(달러)로 정량화할 수 있습니까? (그러나 Ken Schwaber의 실수로부터 배우십시오: "A dead ScrumMaster is a useless ScrumMaster."(참고))
- 여러분의 회사는 경력 개발과 팀의 공동 목표가 일치되는 몇 안 되는 기업 중 하나입니까? 테스트, 테스트 자동화 또는 사용자 문서를 희생하고 프로그래밍 또는 아키텍처 작업을 수행할 동기가 있는 경우 "아니오"로 대답하십시오.

- 여러분의 회사는 업계 언론 또는 기타 독립 언론으로부터 가장 일하기 좋은 기업 또는 업계 리더로 인정받았습니까?
- 학습 조직을 만들고 있습니까?

## 결론

만약 여러분이 이 목록들의 대부분을 기록했고, 시간을 내줄 수 있다면, 여러분으로부터 의견을 듣고 싶습니다.

인간의 독창성을 창조할 수 있는 판에 박힌 공식은 없습니다. 이 문서는 상황에 도움이 될 수도 있고 그렇지 않을 수도 있는 항목을 나열할 뿐입니다.

일단 여러분이 변화를 만들기 위해 무엇을 할 수 있는지 깨닫기 시작하면, 두려움을 느낄지도 모릅니다. 이것은 올바르게 나아가고 있다는 신호입니다.

## 첨부

### 조직의 방해물 양식

- 표면적 문제:
- 근본 원인 (5번의 "왜?"를 사용하십시오):
- 사업적 영향:
- 감정적 영향:
- 명확하고 실행 가능한 요청:

### 지침

만일 여러분이 훈련 과제로 이 체크 리스트를 받았고, 여러분의 현재(또는 가장 최근의) 고용주가 스크럼 같은 것을 시도하고 있다면, 이것을 그곳에서 본 것에 적용하십시오. 다음 중 하나를 사용하여 각 항목에 표시하십시오:

- √ ("잘한 것")
- Δ ("개선 될 수 있으며 시작하는 방법을 알고 있음")
- ? ("개선 될 수 있으나, 어떻게 하는지 모름")
- N/A ("해당 사항 없음" 또는 "아무런 도움이 되지 않을 것")

또는 현재(또는 가장 최근) 고용주가 스크럼과 같은 것을 시도하지 않았다면 각 항목을 다음과 같이 기록하십시오:

- √ ("잘한 것" 또는 "잘할 수 있는 것")
- Δ ("도전해볼만 하고, 시작하는 법을 알고 있음")
- ? ("도전해볼만 하지만, 시작하는 법을 모름")
- N/A ("해당 사항 없음" 또는 "아무런 도움이 되지 않을 것")

모든 항목에 기록하였다면, 첨부된 조직적 장애물 기입란에 둘에서 여섯 개 정도의 조직적 장애물을 정리해 보십시오. 이 점검 목록에서 파생되었는지 여부는 상관없습니다. 바꿀 수 있다는 희망을 적어도 1%는 가지고 있는 장애물을 선택하십시오.