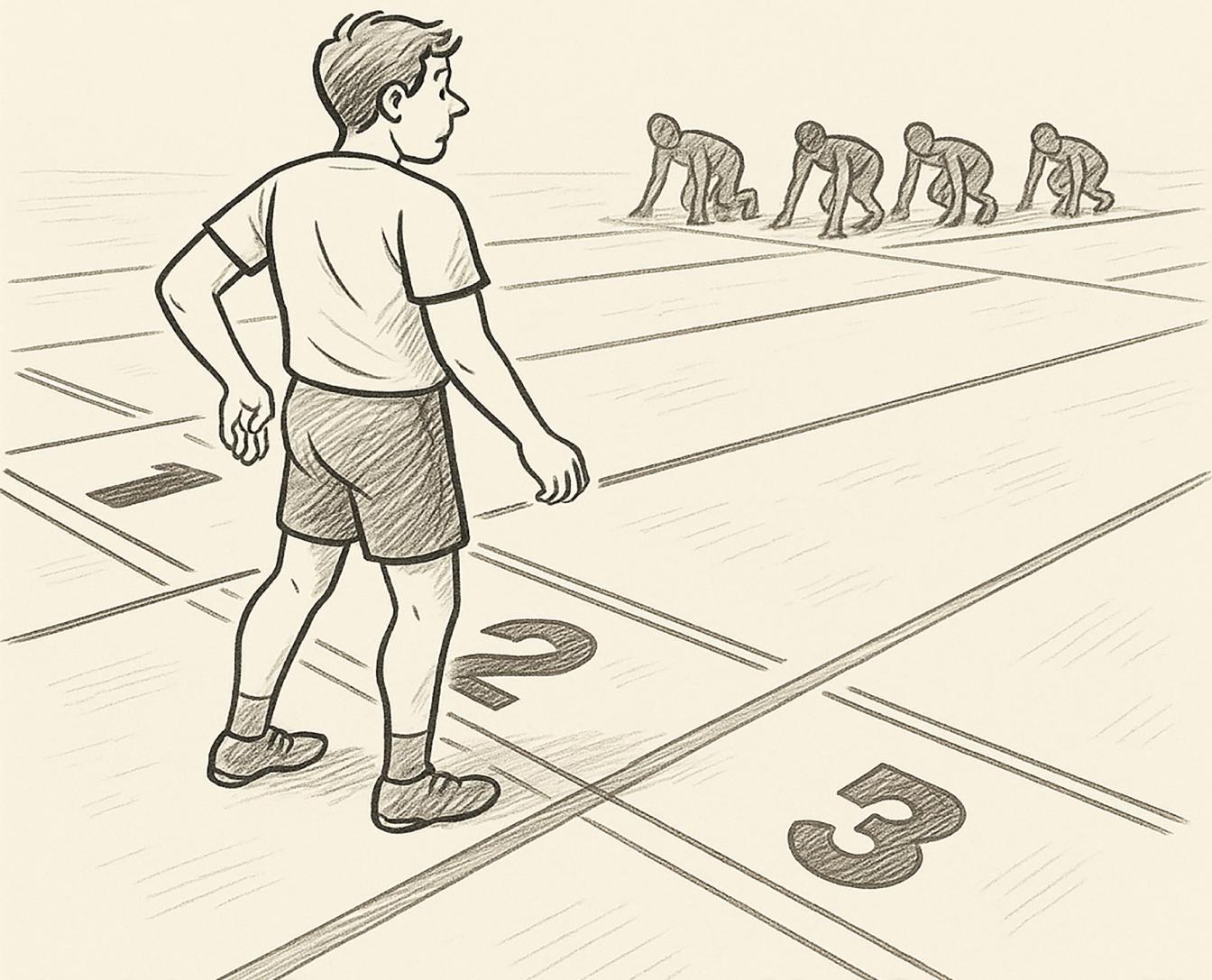


READY

WHY MOST SOFTWARE PROJECTS FAIL AND HOW TO FIX IT



Luniel de Beer

Max Guernsey, III

한국어판

이 책을 트윗하세요!

[Twitter](#)에서 이 책을 공유하여 Luniel de Beer와(과) Max Guernsey, III님을 도와주세요!

이 책을 위한 추천 트윗입니다:

방금 Ready를 구매했습니다 — 소프트웨어 제공 과정에서 재작업, 이월, 불일치를 없애고 싶은 소프트웨어 리더와 팀을 위한 책입니다. #CodeReady

이 책의 추천 해시태그는 [#CodeReady](#)입니다.

다음 링크를 클릭하여 이 해시태그로 다른 사람들이 이야기하는 내용을 확인해보세요:

[#CodeReady](#)

Ready (한국어판)

대부분의 소프트웨어 프로젝트가 실패하는 이유와 해결 방안

Luniel de Beer와(과) Max Guernsey, III

이 책은 <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다

이 버전은 2025-10-22에 출판되었습니다



이 책은 [Leanpub](#) 도서입니다. Leanpub은 린 퍼블리싱 프로세스를 통해 저자와 출판사에 힘을 실어줍니다. [린 퍼블리싱](#)은 가벼운 도구와 여러 차례의 반복을 통해 진행 중인 전자책을 출판하여 독자의 피드백을 받고, 올바른 책이 될 때까지 방향을 전환하며, 성공적인 책이 되면 견인력을 확보하는 과정입니다.

© 2025 Luniel de Beer와(과) Max Guernsey, III

Johann van Aardt를 기리며. 그는 내 열정을 알아봐 주었고, 나를 진정한 프로그래밍의 세계로 이끌어주었으며, 새로운 보금자리를 찾도록 도와주었습니다. 그리고 첫 고객에서부터 미국에서의 첫 보금자리까지, 변함없는 지지로 이 모든 것을 가능하게 해주신 부모님께.

—Luniel

나의 태양이자 삶의 전부인 우리 가족에게.

—Max

차례

이 책에 대하여	i
이 책의 대상	ii
이 책의 활용법	iii
저자 소개	iv
서문	v
파트 I: 무언가가 빠져있다	1
장 1: 숨겨진 문제	2
장 2: 기초가 부족할 때의 대가	14
장 3: 요구사항 성숙도 흐름(RMF) 소개	25
장 4: 이것이 애자일인가?	27
파트 II: 준비성을 위한 여유 만들기	28
장 5: 첫 번째 확장	29
장 6: 왜 사람들은 이것을 하지 않을까?	30
장 7: 명시적 준비도 작업 (RMF 1)	32
장 8: RMF 1의 효과	34
장 9: RMF 1 실전 적용하기	35

파트 III: 작업 완료의 관문화	37
장 10: 다음 필요사항	38
장 11: 사람들이 일반적으로 하는 일	40
장 12: 완료 정의 정의하기	42
장 13: 맞춤형 완료 정의 (RMF 2)	44
장 14: RMF 1 & 2와 함께하는 삶	47
장 15: RMF 2 설치하기	49
파트 IV: 게이팅 구현	51
장 16: 마지막 요구사항	52
장 17: 준비 정의에 대한 배경	54
장 18: 준비완료 정의 정의하기	56
장 19: 맞춤형 준비 완료 정의 (RMF 3)	58
파트 V: 종합	62
장 20: <u>대부분의</u> 마감일은 중요하지 않다	63
장 21: 역량 1: 요구사항 성숙도 흐름	66
장 22: Scrum에서 RMF를 통한 작업과 정보의 흐름	68
장 23: RMF의 영향	70
장 24: RMF로의 전환	71
장 25: 이제 당신의 차례입니다	73
파트 VI: 자료	74
부록 A: 스크럼이 문제가 아닙니다	75

부록 B: 시냅스 프레임워크™	77
부록 C: RMF 1에 대한 일반적인 반대 의견과 장애물	79
부록 D: DoD 시작용 기준 목록	80
부록 E: DoR 시작 기준 목록	81
색인	82

이 책에 대하여

Ready는 과소 납품, 만성적인 재작업, 그리고 불명확한 요구사항에 지친 모든 소프트웨어 개발 관계자를 위한 책입니다.

여러분은 이미 팀 실행 능력에 투자하고, 프로세스 프레임워크의 구현을 개선하거나, 코드를 리팩터링하는 등의 노력을 해보았지만 여전히 더 많은 개선이 필요할 수 있습니다.

이는 대부분의 소프트웨어 개발 팀의 주요 제약이 팀 기술이 아닌 요구사항 성숙도에 있기 때문입니다. 적절한 기술을 보유한 성숙한 팀조차도 성숙하지 않은 요구사항을 다룰 때는 여전히 어려움을 겪습니다.

Ready는 RMF(요구사항 성숙도 흐름)를 소개합니다. 이는 기존 프로세스를 대체하지 않으면서도 제품팀과 개발팀을 조율하는 실용적이고 체계적인 접근 방식입니다.

스크럼, 칸반, 또는 다른 맞춤형 방법론을 사용하든 관계없이, RMF는 범위를 안정화하고, 이월을 제거하며, 실제로 중요한 것을 전달하는 데 도움을 줍니다.

만약 여러분의 팀이 “거의 완료” 단계에서 막혀있다고 느낀다면, 이 책은 여러분이 이 순환을 깨고 팀을 영구적으로 막힘없이 진행하도록 도와줄 것입니다.

이 책의 대상

이 책은 말 그대로 소프트웨어 개발에 관련된 모든 사람을 위한 것입니다. 엔지니어부터 제품 관리자까지, 개별 기여자부터 임원진까지 모두를 위한 책입니다.

만약 여러분이 소프트웨어 개발에 관여하고 있으며, 함께 일하거나 소속된 팀이 다음 문제들 중 하나 이상을 겪고 있다면 이 책이 도움이 될 것입니다:

- 작업이 한 이터레이션에서 다음 이터레이션으로 자주 이월됨
- 구현 팀이 움직이는 목표물을 쫓는 것 같은 느낌을 받음
- 작업이 너무 오래 열려있음
- 작업이 완료로 표시되었지만 실제로는 완전히 끝나지 않음
- 완료된 작업이 기대와 일치하지 않음
- 작업이 정기적으로 많은 결함을 발생시킴
- 대량의 작업을 정기적으로 다시 해야 함

이러한 문제들 중 어느 하나라도 익숙하게 들린다면, Ready가 도움이 될 수 있습니다.

이 책의 활용법

이 책은 실용적인 목적으로 설계되었습니다. 이론적인 논문이나 전략 프레젠테이션이 아닌, 실제 고객 작업을 바탕으로 하고 실제 납품 압박 하에서 현장 테스트를 거친 RMF(요구사항 성숙도 흐름)를 구현하기 위한 실용 매뉴얼입니다.

각 장은 순차적으로 작성되었지만, RMF 자체는 모듈식입니다. 세 가지 기초 실천 방안으로 구성되어 있습니다:

- RMF 1: 공유된 이해를 위한 협업
- RMF 2: 맞춤형 완료 정의를 사용한 작업 완료 게이팅
- RMF 3: 맞춤형 준비 정의를 사용한 구현 게이팅

각 부분 또는 우리가 부르는 “습관”은 독립적으로 존재하지만, 서로를 기반으로 발전 합니다. 이 책은 여러분이 이들을 순서대로 하나씩 접근할 수 있도록 돋기 위해 설계 되었습니다. 이러한 구조는 팀이 실제로 RMF를 도입하는 방식을 반영합니다 – 이전 습관이 잘 작동한 후에만 각 습관을 차례로 도입합니다.

이는 팀에게 부담을 주지 않으며 각 변화가 잘 정착될 수 있는 최상의 기회를 제공합니다. 이에 대한 자세한 내용은 [장 9](#)에서 배우게 될 것입니다.

조언, 코칭, 또는 리더십 팀과의 대화 등 도움이 필요하시다면 언제든 직접 연락해 주시기 바랍니다.

그리고 RMF 구현을 위한 공식적인 지원이 필요하시다면, Producore는 단계별 도입을 안내하도록 설계된 전체 프로그램 시리즈를 제공합니다. <https://ready-book.link/rmf>에서 자세히 알아보실 수 있습니다.

저자 소개

Luniel de Beer는 제품 의도와 엔지니어링 실행 사이의 간극을 해소하기 위한 실용적인 시스템인 요구사항 성숙도 흐름(RMF)의 창시자입니다. 그는 애자일 전환 주도, 제품과 엔지니어링 간의 가교 역할, 그리고 팀이 명확성과 자신감을 가지고 제품을 전달할 수 있도록 돋는 15년 이상의 경험을 보유하고 있습니다.

Luniel은 또한 제품 기능을 모델링하기 위한 추적 가능하고 확장 가능한 접근 방식인 Producore의 역량 관리 시스템을 고안했습니다. 그는 지속적인 제품 요구사항을 관리하기 위한 버전 관리 시스템인 PKB 주도 개발(PKBDD)을 구상했습니다. 이러한 도구들은 Producore에서 개발된 더 큰 전달 프레임워크의 일부를 형성합니다.

Max Guernsey, III는 소프트웨어 아키텍트, 교육자이자 구조적 및 기술적 엄격성을 통해 전달 실패를 해결하는데 전념하는 컨설팅 회사인 Producore의 공동 설립자입니다. 객체 지향 설계, 리팩터링, 테스트 주도 개발, 디자인 패턴 분야에서 20년 이상의 경험을 보유하고 있으며, 중요 시스템을 직접 구축하고 대규모 엔지니어링 팀을 코칭해왔습니다. 그의 작업은 조직이 지속 가능한 전달 우수성을 달성할 수 있도록 심도 있는 기술 실천과 행동 및 프로세스 전환을 결합합니다.

Max는 PKBDD에 큰 공헌을 했으며, 행위 명세에 대한 깊은 전문성을 바탕으로 Producore의 행위 주도 개발(BDD) 접근 방식 개발을 주도했습니다.

이들의 협력은 명확성, 추적성, 게이팅을 하나의 통합된 시스템으로 결합하여, 팀 단위의 실무에서부터 조직적 역량에 이르기까지 확장 가능한 소프트웨어 제공 체계를 구축했습니다.

서문

엔지니어링 리더들을 위한 안내

엔지니어링 조직의 고위 리더라면, 아마도 노력이나 규율, 또는 똑똑한 인재가 부족하지는 않을 것입니다. 그럼에도 불구하고, 프로젝트는 여전히 정체되고 있습니다. 목표는 미끄러지고, 기대치는 충족되지 않습니다. 이는 팀이 계을러서가 아니라, 일이 정의되고, 형성되고, 전달되는 방식에 있어 근본적인 무언가가 깨져있기 때문입니다.

이 책은 리더십 가이드가 아닙니다. 이는 구조적 진단을 위한 도구입니다. 팀 내부에서 실제로 무슨 일이 일어나고 있는지를 보여줍니다—왜 “거의 완료”가 계속해서 “미완료”가 되는지, 그리고 왜 지엽적인 진전이 좀처럼 전략적 성과로 이어지지 않는지를 말입니다.

여러분은 이 책에서 자신의 모습을 발견하지 못할 수도 있습니다. 하지만 만약 여러분의 팀이 필요한 것을 제대로 전달하지 못한다면, 여러분은 그들의 모습을 보게 될 것입니다. 그리고 그때, 여러분은 마침내 그것을 해결하기 위한 언어와 시스템을 갖게 될 것입니다.

Luniel로부터

우선, 이 책은 Max 없이는 불가능했을 것입니다. 그의 안개와 혼란을 깨뚫어 보고 아이디어의 본질을 추출해내는 능력은 제가 감히 따라갈 수 없는 수준입니다.

우리가 어떻게 여기까지 오게 되었을까요?

돌이켜보면, 저는 항상 일이 실제로 어떻게 작동하는지 이해하고 싶어 했던 것 같습니다. 종교든, 영양학이든, 소프트웨어 개발이든, 저는 계속해서 같은 문제에 부딪혔습니다: 압박을 받으면 무너지는 표면적인 답변들이었죠. 그래서 저는 계속 파고들었습니다—우리가 무엇을 하는지뿐만 아니라, 왜 하는지, 그리고 잘 작동하지 않을 때 무엇이 빠져있는지를 묻는 것이죠.

시스템의 가장 초기 균열은 제가 유명 기술 회사에서 데이터 서비스를 제공하는 팀의 스크럼 마스터, 프로덕트 오너, 그리고 개발 관리자(!!)라는 세 가지 역할을 동시에 맡고 있을 때 나타났습니다. 우리는 스크럼이 말하는 대로 했습니다—짧은 스프린트,

백로그의 스토리들, 반나절 동안의 계획—하지만 새로운 스프린트를 시작할 때마다 마찰이 발생했습니다. 팀은 문제를 완전히 이해하지 못했고, 스프린트 중간에 요구사항을 재검토하고 수정해야 했으며, 예방 가능한 의존성이 드러나 자연을 초래했고, 핵심 단계들이 누락되곤 했습니다.

그래서 저는 뭘가 다른 것을 시도하기 시작했습니다. 모든 스토리에 대해 팀과 이해 관계자들을 한 방에 모아, 문제를 상세히 살펴보고, 함께 해결책을 브레인스토밍한 후에야 스토리를 작성했습니다. 스프린트 계획은 한 시간으로 줄어들었고, 우리의 전달 성공률은 급격히 상승했습니다.

천천히, 저는 성공이 스프린트 내에서 더 열심히 일하는 것에서 오는 것이 아니라는 것을 깨닫기 시작했습니다. 그것은 스프린트가 시작되기 전에 마련해 놓는 구조에서 오는 것이었습니다.

나중에, Jeff Sutherland가 “준비 완료의 정의”에 대해 이야기하는 것을 듣고 나서, 거기에 가치 있는 무언가가 있다는 것을 알았습니다—하지만 그것만으로는 충분하지 않았습니다. 요구사항, UX, UI, 연구, 그리고 나중에는 BDD와 관련된 제 경험은 서로 다른 작업 항목들이 서로 다른 종류의 준비도를 요구한다는 것을 보여주었습니다. 어떤 것은 행위 명세가 필요하고, 어떤 것은 시스템 접근이 필요하며, 어떤 것은 전체 기능 추적이 필요합니다.

그리고 이 모든 것에는 가정이 아닌, 실제로 확인된 공유된 이해가 필요합니다.

더 많은 팀들과 일하면서, 저는 어디서나 같은 패턴을 보았습니다: 누락된 단계들, 충족되지 않은 의존성들, 최선을 다하지만 예방될 수 있었던 문제들을 해결하기 위해 끊임없이 고군분투하는 팀들. 훌륭한 팀들조차도 어려움을 겪었습니다—그들이 약해서가 아니라, 준비도를 명확하게 만드는 구조가 없었기 때문입니다.

이 모든 학습, 반복, 그리고 좌절의 결과가 준비도를 관리하기 위한 구조화된 시스템입니다.

이 책이 바로 그것에 관한 것입니다.

이 책이 실제 문제가 어디에 있는지, 그리고 어떻게 해결할 수 있는지에 대한 명확성을 제공하기를 바랍니다. “추가적인” 것처럼 보이지만 실제로는 필수적인 관행들을 옹호할 수 있는 언어를 제공하기를 바랍니다. 그리고 무엇보다도, 이 책이 팀들이 더 적은 스트레스, 더 적은 놀라움, 그리고 훨씬 더 나은 결과로 전달할 수 있도록 돋기를 바랍니다.

우리가 이것을 제대로 해낸다면, 업계에서 수십억 달러를 절약할 수 있을 것입니다.

하지만 더 중요한 것은, 사람들에게 그들의 정신 건강을 되돌려줄 수 있다는 것입니다.

Max로부터

저는 수십 년 동안 다양한 각도에서 이 문제에 대해 작업해 왔지만, Luniel을 만나기 전까지는 제 진전이 더디었습니다.

이는 제가 그를 알기 전에는 이 문제를 근본적으로 기술적인 문제로 접근했기 때문입니다. 저는 팀들이 테스트 주도 개발(TDD), 리팩토링, 고급 소프트웨어 설계, 그리고 나중에는 인수 테스트 주도 개발(ATDD)이나 행위 주도 개발(BDD)과 같은 것들을 채택하도록 돋는 데 집중했습니다.

대부분의 경우, 이 책에서 다루는 문제는 이러한 기술적 관행들을 확립하는 데 있어 구현 세부사항으로 취급되었습니다.

이는 제가 더 이상 기술적 관행들을 중요하게 여기지 않는다는 뜻은 아닙니다. 저는 여전히 그것들이 매우 중요하다고 생각하지만, 그것들이 소프트웨어 개발에서의 준비도 문제를 직접적으로 다루지는 않습니다. 대신, 그것들은 그 문제를 드러내고, 그러면 사람들은 구현하려고 하는 기술적 관행들을 지원하기에 “그저 충분한” 정도로 그것을 해결하기 위해 프로세스에 패치를 덧붙입니다.

저는 또한 이 책이 누구를 도울 수 있는지에 대한 질문도 다루고 싶습니다. 간단한 답변은 “아마도 소프트웨어 개발에 있는 거의 모든 사람”이지만, 실제 답변에는 기본적인 의미를 변경하지 않고도 다양한 환경에 맞출 수 있게 해주는 뉘앙스가 포함되어 있습니다.

이 책에서 제시하는 해결책이 필요한 팀들이 있습니다. 1장에서 이러한 팀의 순화된 사례를 만나보실 수 있습니다.

또한 우리가 제안하는 시스템이 엄밀히 말해 필요하지 않더라도 이를 통해 이점을 얻을 수 있는 팀들도 있습니다.

제가 지금까지 함께 일해본 최고의 팀은 Central Oregon 오지에 자리잡고 있었는데, 이들은 그 다음으로 좋은 팀보다 훨씬 뛰어났으며, 아마도 표준편차 한두 단위는 더 높았을 것입니다. 이들은 매우 뛰어난 성과를 내는 팀이어서, 순전히 그들의 역량만으로도 이러한 시스템이 없는 상황을 극복할 수 있었습니다. 하지만 당시 제 관리자였던 Tom Barreras는 “테스트에 대해 사전에 시간을 들여 이야기를 나눌 때 스토리가 더 잘 진행된다는 걸 알게 됐어”라는 말을 한 적이 있습니다.

이것 역시 당시에는 테스트-개발과 기술적 실행이라는 관점에서만 바라보았지만, 지금은 이것이 준비도가 팀에 영향을 미치는 요소였다는 또 다른 지표였음을 알고 있습니다… 그 특별한 팀은 너무나 유능하고 대응이 빨라서 장애요소를 사전에 예방하기보다는 발생했을 때 대응하는 방식으로도 성공할 수 있었습니다.

준비도에 대해 특별히 신경 쓸 필요가 없을 만큼 뛰어난 사람이거나, 그러한 동료들과 함께 일하는 경우라도, 이 책의 내용을 통해 여전히 이점을 얻을 수 있습니다.

파트 I: 무언가가 빠져있다

같은 일을 더 잘하는 것이 도움이 되지 않을 때는, 하지 않고 있는 것을 찾아보라.

장 1: 숨겨진 문제

이것은 한 은행에 관한 실제¹ 이야기입니다. 우리는 이 은행을 그냥 “그 은행”이라고 부르겠습니다. 이는 미국의 국가 금융 인프라의 일부를 담당하는 연방 신용 기관 유형입니다.

우리(Luniel과 Max)는 소프트웨어 프로젝트 구현에 어려움을 겪고 있던 그 은행에 투입되었습니다. 이곳은 우리가 본 환경 중 가장 역기능적인 곳 중 하나였고, 그래서 우리는 이것을 첫 번째 사례 연구로 선택했습니다. 그 은행에서 의미 있는 변화가 가능했다면, 어디서든 가능하다는 것을 보여주기 때문입니다.

프로젝트에 관한 간단한 설명

이 책에서 “프로젝트”라는 용어를 사용할 때는 프로젝트 관리의 맥락에서 사용합니다. 이 단어의 의미에 대해 다양한 견해가 있지만, 우리는 [프로젝트 관리 협회](#)의 정의를 사용합니다:

“프로젝트는 고유한 제품, 서비스 또는 결과물을 만들기 위해 수행되는 일시적인 노력입니다.”

이는 프로젝트에 정해진 시작과 끝이 있다는 것을 의미합니다. 프로젝트가 종료되면, 프로젝트 지식과 산출물이 보관되고, 팀원들은 해산되며, 계약이 마무리됩니다.

이 책에서 프로젝트는 근본적으로 실행에 관한 것입니다. PMI가 정의한 대부분의 프로젝트는 타당성 조사나 설계로 시작됩니다. 비전과 전략은 프로젝트가 시작되기 전에 이미 수립되어 있습니다.

프로젝트는 그 비전과 전략에서 탄생하며, 설정된 목표가 실현되었는지 여부에 따라 성공 또는 실패가 결정됩니다—그 목표들이 올바른 것이었는지 여부와는 관계없이 말입니다.

여러분은 “프로젝트”라는 단어를 다르게 사용할 수 있으며, 그것도 괜찮습니다. 다만 우리가 이 단어를 사용할 때는 위의 정의와 맥락을 참조한다는 점을 알아두시기 바랍니다.

¹관련된 사람들과 기관들의 프라이버시를 보호하기 위해 식별 정보를 변경했습니다.

이는 우리가 소프트웨어 개발에 프로젝트 관리를 사용하는 것을 옹호한다는 의미는 아닙니다. 오히려 그 반대입니다. 하지만 우리는 그럼에도 불구하고 이것이 사용되고 있다는 것을 인정합니다. 우리는 이 문제를 나중에 [장 6](#)에서 다룹니다.

1.1: 전형적인 재작성

그 은행은 여러 가지 이유로 대출 상환 포털을 재작성하고 있었습니다.

기존 시스템은 전체가 C#/.NET 솔루션이었는데, 버그가 많았습니다. 고객 만족도가 저하되었을 뿐만 아니라, 시스템이 만든 오류를 수정하기 위해 누군가가 수동으로 데이터베이스를 조작해야 하는 매우 비용이 많이 드는 지원 사고가 끊임없이 발생했습니다.

기존 시스템은 유지보수 관점에서도 낙후되어 있었습니다. 엔지니어들이 의미 있는 변경을 하는 것이 거의 불가능했고, 변경이 가능하더라도 매우 위험한 작업이었습니다.

재작성은 이러한 상황을 바꾸기 위한 것이었습니다.

새로운 시스템도 C#/.NET 백엔드를 사용할 예정이었지만, 테스트가 완벽하게 적용될 예정이었습니다. 프론트엔드는 OutSystems로 구현될 예정이었는데, 이는 인기 있는 로우코드 또는 노코드 솔루션으로, 조직이 한 곳에서 애플리케이션을 정의하면 변경 사항을 배포하기로 결정할 때마다 웹 앱, Android 앱, iOS 앱이 자동으로 생성되는 방식입니다.

새로운 시스템이 버그 없이 작동하여 고객 만족도를 높이고 지원 비용을 크게 줄일 수 있기를 희망했습니다.

또한 백엔드에 대한 더욱 체계적인 접근 방식과 프론트엔드에 대한 로우코드 방식의 조합으로 새로운 기능 개발의 비용과 위험을 크게 줄여 개발자들의 업무가 원활해지기를 기대했습니다.

OutSystems로 이전하면서 얻게 되는 좋은 부수 효과는 주요 플랫폼 모두에서 깔끔하고 현대적인 모바일 앱을 얻을 수 있다는 것이었습니다.

이것이 이 이야기가 시작되기 3년 전에 시작했을 때의 꿈이었습니다. 하지만 현실은 지금까지 팀들이 아무것도 출시하지 못했다는 것이었습니다.

1.2: 문제에 대한 관점들

경영진과 대화했을 때, 우리는 그들이 큰 투자를 했음에도 전략적인 진전이 전혀 없다는 것에 대해 당연한 좌절감을 느끼고 있음을 알 수 있었습니다.

그들의 관점에서는 모든 것을 시도해보았습니다. 직원을 교체하고, 중원하고, 예산을 변경하고, 증액하고, 압박을 가하고, 수많은 컨설턴트들을 영입했습니다(우리가 그 마지막이라는 것이 강하게 암시되었습니다). 아무것도 상황을 개선시키지 못했습니다— 적어도 그들이 측정할 수 있는 방식으로는 말입니다. 왜냐하면 그들이 본 것은 “바늘”이 한 분기에 0을 가리키다가 다음 분기에도 여전히 0을 가리키고 있다는 것뿐이었기 때문입니다.

그들은 더 이상의 “보이지 않는 진전”을 원하지 않았습니다. 그들은 결과를 원했습니다.

제품 조직 내의 관리자들과 대화했을 때, 우리는 약간 다른(하지만 여전히 비슷한) 이야기를 들었습니다. 그들이 엔지니어링 팀과 더 직접적으로 일했기 때문입니다.

팀들이 아무것도 하지 않았다는 게 아니었다. 적어도 그들 입장에서는 그랬다. 문제는 팀들이 요청받은 일을 절대로 하지 않았다는 것이다. 거의 확실했다: 요청이 아무리 단순하고 명확하게 전달되었더라도, 팀이 만든 결과물을 평가할 때면 완전히 다른 것이 나왔다.

상황이 너무 심각해져서 “우리가 원하는 걸 얻으려면 우리가 원하지 않는 걸 요청해야 할 것 같다”는 농담이 돌 정도였다.

엔지니어링 임원진들은 상황을 매우 다르게 보고 있었다.

그들이 보기에는 문제는 프로덕트 팀이 실행 가능한 요구사항을 제공하지 않았고, 충분한 요구사항을 제공하지 않았다는 것이었다. 프로덕트 팀이 “알아서 잘” 해준다면, 팀들은 원하는 것을 정시에 예산 내에서 제공할 수 있을 거라고 생각했다.

그들은 코드를 작성하고 배포하는 방식을 현대화하는 데 많은 투자를 했고, 그들의 관점에서는 프로덕트 팀이 명확한 요구사항을 제공하지 않고 있었다.

우리가 다른 컨설턴트들(그들이 우리를 이 조직에 추천했다)과 이야기했을 때, 그들은 당연히 그들이 목격한 역기능에 주목했다: 모두가 다른 사람을 비난하는 데 매우 집중하고 있었다. 그들이 애초에 우리를 불러들인 이유는 인력 운영 전략에 대해 우려했고 개별 기여자들에 대한 평가를 원했기 때문이었지만, 그들은 임원진 수준의 책임 전가와 업무 감독자적 태도를 주요 문제의 원인으로 보고 있었다.

1.3: 우리의 조사

우리의 초기 임무는 팀들을 평가하고 필요한 경우 그들의 기술을 향상시키는 것을 돋는 것이었기에, 우리는 최전선에서 일하는 사람들을 살펴보기 시작했다.

분명히 개선의 여지가 있었다.

프로덕트 쪽의 개별 기여자들은 실제로 필요한 기술을 갖추지 못했다. 현실적으로, 그들은 대부분 프로덕트 오너(PO) 또는 프로덕트 매니저 역할에 떠밀린 프로젝트 매니저들이었다.

결과적으로, 그들 중 절반은 “모호한” 요구사항을 작성하고 나서 어떤 비판적 분석도 없이 그 이터레이션에서 팀이 한 일을 (문자 그대로) 받아들였다. 나머지 절반은 같은 종류의 요구사항을 작성하고 나서 팀이 “알았어야 했다”며 한 번도 언급하지 않은 것들에 대해 불평하면서 작업 항목들을 사실상 무기한으로 열어두었다.

은행은 작업 백로그를 관리하고 추적하는 데 사용했다. 이 책에서 다루는 내용은 대부분 스크럼과 직교하지만, 대다수의 팀들이 (또는 적어도 다수가) 스크럼을 사용하기 때문에 우리는 책 전반에 걸쳐 스크럼 용어를 사용한다.



정의: 스크럼

스크럼은 복잡한 문제에 대한 적응형 솔루션을 통해 사람, 팀, 조직이 가치를 창출하는 것을 돋는 경량 프레임워크다. 간단히 말하면:

1. 프로덕트 오너가 복잡한 문제에 대한 작업을 프로덕트 백로그로 정리 한다
2. 스크럼 팀이 스프린트 동안 선택된 작업을 가치 있는 결과물로 만든다
3. 스크럼 팀과 이해관계자들이 결과를 검토하고 다음 스프린트를 위해 조정한다
4. 반복한다

스크럼과 그 용어에 익숙하지 않다면, [스크럼 가이드 2020년 버전](#)을 검토하는 것을 추천한다. 빠르고 유익한 읽을거리다.

마찬가지로, 우리는 기술 팀들의 코딩 실력이 일반적인 수준보다 훨씬 낮고(-2σ 수준이 최선), 거기다 변화에 매우 저항적이라는 것을 발견했다. 당연한 결과로, 코드 품질은 형편없었다.

하지만 팀들의 말로는, 이것이 그들이 제대로 결과물을 내지 못하는 이유가 아니었다. 그들에게는 모호한 요구사항과 프로덕트의 스프린트 중간 변경이 프로젝트를 망치고 있었다.

…그리고 아무도 더 큰 문제에 대해 이야기하지 않았다. 그 문제는 너무 터무니없어서 직접 겪기 전에는 지어낸 이야기처럼 들릴 정도였다.

엔지니어링 팀들은 요구사항을 이해하지 못하면 아무거나 만들어 놓고는 “작업 항목을 완료했다”며 인정을 요구하는 습관이 있었다.

작은 오해 정도가 아니었다. 완전히 동떨어진 것이었다: 우리가 특정 상황에서 원금 적용을 비활성화하라고 하면, 그들은 대신 보조 확인 이메일 주소를 추가하는 기능을 비활성화했다.

그리고는 우리가 요청한 게 그거라고 말했다.

1.4: 더 깊이 파고들기

두 가지 기술 격차 모두 해결할 수 있었지만, 우리는 그것들이 진짜 장애물이라는 것에 회의적이었다.

뭔가 다른 문제가 있었기에, 우리는 더 깊이 파고들었다. 우리는 이 질문으로 시작했다: 왜 요구사항을 작성하는 데 그렇게 오래 걸리고 그렇게 나쁜 결과가 나오는 걸까?

한 가지 이유는 의미 있는 요구사항을 작성하는 데 필요한 지식이 매우 부족했기 때문이다. 엔지니어링과 프로덕트 팀이 그 지식의 일부만을 가지고 있었다.

일부는 레거시 시스템의 코드에 내장되어 있었습니다. 일부는 완전히 사라졌습니다. 하지만 대부분은 은행의 여러 부서에 흩어져 있는 주제 전문가들의 암묵지 형태로 보관되어 있었습니다. 이는 전략적 목표를 실제로 진전시키는 요구사항을 만드는 것이 매우 노동 집약적이고 시간이 많이 소요되는 활동이었다는 것을 의미합니다.

이와 대조적으로 기능이 부족한 리더십 팀에서는 기능에 대한 끝없는 욕구가 있었습니다. “엔지니어들이 계속 일하게 하라—요구사항으로 가득 채워라”라는 지시가 있었습니다. 팀을 바쁘게 유지하기 위한 요구사항의 양에 초점이 맞춰져 있었습니다—이는 실질적인 ‘진전’을 이룰 수 있는 요구사항을 정의하는 데 필요한 주의와 시간과는 정반대되는 관점이었습니다.

1.5: 상황을 개선하려 했지만 더 나아지지 않았다

이 모든 문제들은 해결할 수 있는 것들이었지만, 해결하려 노력해도 도움이 되지 않았습니다.

소프트웨어 개발 기법을 개선하려는 과거의 시도들은 팀의 제공 능력 향상에 도움이 되지 않았지만, Max는 팀들이 더 발전할 수 있도록 돋기 위해 노력했습니다.

그는 20세기 중반의 프로그래밍 교리에서 나온 혁신적인 개념들을 소개했습니다. “그 코드를 27번² 복사-붙여넣기하지 말고, 함수로 만들어서 그것을 호출하세요”와 같은 제안만으로도 새로운 코드의 품질이 극적으로 향상되었고 품질 개선을 시작할 수 있게 되었습니다.

이러한 기본적인 코딩 조언들은 팀이 미래에 더 쉽게 유지보수할 수 있는 더 나은 코드를 작성하는 데 도움이 되었습니다.

…하지만 프로젝트를 진전시키는 데는 도움이 되지 않았습니다.

제품 측면에서, Luniel은 BDD를 도입하여 제품 책임자들이 팀에 요구사항을 전달하기 전에 철저히 검토하도록 했습니다.

그는 팀들이 협력하여 제품 백로그 항목(PBI)이 실제로 완료되었는지 평가하는 데 이를 사용하도록 했습니다.



정의: 제품 백로그 항목 (PBI)

제품 백로그 항목(PBI)은 제품에 대한 잠재적 변경, 추가 또는 개선을 나타내는 제품 백로그의 개별 작업 단위입니다. PBI는 기능, 버그 수정, 기술적 개선, 연구 과제 등 다양한 형태를 취할 수 있으며, 제품 가치에 대한 기여도로 정의됩니다.

많은 팀들이 PBI를 “스토리” 또는 “사용자 스토리”라고 부르지만, 스크럼의 정확한 용어는 “제품 백로그 항목” 또는 “PBI”입니다. PBI가 스프린트에 포함되면 스프린트 백로그의 일부가 됩니다. 단순화와 중립성을 위해, 우리는 스크럼 팀이 관리하는 모든 작업 항목을 “제품 백로그 항목” 또는 “PBI”로 지칭합니다—여러분이 그것을 제품 백로그 항목, 스프린트 백로그 항목(SBI), 사용자 스토리, 스토리, 작업 항목, 또는 백로그 항목이라고 부르든 상관없이.

이는 명확성을 더했지만, 흐름을 만들어내지는 못했습니다.

우리를 데려온 파트너 컨설턴트들의 도움으로, 팀과 요구사항 작성자들에게 가해지는 극심한 압박을 (일시적으로) 완화할 수 있었습니다.

이는 약간의 신뢰를 구축하는 데 도움이 되었을 수 있지만, 가시적인 결과를 내지는 못했습니다.

²이는 과장이 아닐 뿐만 아니라, 가장 심각한 사례도 아닙니다. 한 경우에는 동일한 알고리즘이 거의 백개나 정확히 중복되어 있었습니다.

우리는 심지어 사람들이 요구사항을 작성하는 데 필요한 비즈니스 지식을 추적하고 그 지식에 있어 간극이 있는 부분을 식별하는 데 도움이 되는 지식 베이스를 구축하기 시작했습니다.

이는 요구사항 작성 속도를 높였지만, 제품을 출시하는 데는 도움이 되지 않았습니다.

수개월간의 상호작용 끝에, 우리는 리더십이 현재 상황을 파악하도록 도왔지만, 그들은 목표에 전혀 가까워지지 않았습니다. 그리고 더 가까워지지도 않고 있었습니다.

그들은 팀이 항상 바쁘게 유지되도록 “과부하를 주는” 옛 전략으로 돌아갈 준비를 하고 있었습니다.

1.6: 소거법

“이건 희망이 없다”라고 손을 들어버리는 것이 쉬웠을 것입니다. 평계 삼을 만한 이유들이 많이 있었습니다:

- 엔지니어링 팀의 기술력이 낮았다 (사실이었습니다)
- 요구사항 작성자들이 잘못된 기술 세트를 가지고 있었다 (사실이었습니다)
- 리더십이 비현실적인 기대로 팀을 압박하고 있었다 (사실이었습니다)
- 조직이 기능하는 데 필요한 중요한 비즈니스 지식이 부족했다 (사실이었습니다)
- 임원들이 서로를 신뢰하지 않는 것처럼 보였다 (사실이었습니다)

이 모든 것이 사실이었습니다. 하지만 이러한 모든 영역에서 개선이 이루어졌음에도 근본적인 문제는 나아지지 않았습니다. 어느 것도 엔지니어링 팀, 제품팀, 관리자, 또는 임원진이 목표에 더 가까워지는 데 도움이 되지 않았습니다.

…그리고 바로 거기에 해결책을 위한 힌트가 있습니다: 앞서 나열된 모든 문제들은 사람들이 이미 볼 수 있었던 문제들이었습니다.



볼 수 있는 변수들이 차이를 만들지 않는다면, 보이지 않는 변수가 차이를 만들고 있을 것입니다.

진짜 문제는 아무도 알지 못했던 곳에 있었습니다.

1.7: 진짜 원인 찾기

은행이 목표를 달성하지 못하게 막고 있는 진짜 원인을 찾기 위해서는 어디서부터 시작해야 했습니다.

실패한 제품 백로그 항목들(대부분이 실패했습니다)의 공통점을 살펴보는 것이 합리적인 시작점이었습니다.

우리는 먼저 크게 다양했기 때문에 공통점이 아님이 분명한 것들을 제외했습니다:

- 시스템의 어느 부분: 일부 제품 백로그 항목은 백엔드만, 다른 것들은 프론트엔드만, 또 다른 것들은 양쪽 모두에 영향을 미쳤습니다
- 어느 팀이 작업을 수행했는지 – 누가 작업을 했는지는 중요하지 않은 것 같았고, 실패할 확률이 높았습니다
- 어느 제품 책임자가 작업을 작성했는지 – 팀의 경우와 마찬가지였습니다

그런 다음 공통적인 것들을 살펴보기 시작했습니다. 목록이 길지는 않았지만, 그렇다고 짧지도 않았습니다:

- 엔지니어링 팀들
- 제품 책임자들
- 리더십
- 문화
- 개발 환경
- 엔지니어링 관행
- 요구사항 작성 기법
- 도메인(금융)
- 종속성 서비스

이 중 많은 것들도 즉시 제외할 수 있었습니다. 팀, 제품 책임자, 리더십, 문화, 개발 환경은 모두 최근에 개선되었지만 의미 있는 결과물에는 실질적인 영향이 없었습니다. 우리는 직접 엔지니어링과 요구사항 작성 관행을 개선하는 것을 도왔고 그러한 개선이 유지되고 있다는 것을 확인했지만, 여전히 도움이 되지 않았습니다.

도메인을 탓할 수는 없습니다. 금융은 기록된 역사에서 가장 오래된 계산 유형 중 하나입니다. 매우 성숙한 분야입니다. 게다가, 다른 은행들이 소프트웨어를 배포하고 있다는 사실이 은행은 이런 일을 할 수 없다는 (명백히 무리한) 가설을 실증적으로 반증하고 있습니다.

종속성 서비스들도 비난할 수 없었습니다. 우리가 살펴보고 있는 이니셔티브만큼이나 변경하는 데 어려움을 겪고 있었기 때문입니다…

…하지만 그것이 우리의 생각을 이끌어냈습니다: 실패의 원인을 분석하기 시작하면 어떨까요?

1.8: 실패의 씨앗 해부하기

한 제품 백로그 항목은 팀이 늘 그래왔듯이 요청과 거의 완전히 무관한 무언가를 완전히 임의로 수행했기 때문에 제품을 진전시키는 데 실패했습니다. 이는 분명히 그들이 작업 항목을 이해하지 못했다는 신호입니다. 따라서 우리가 BDD 도입을 도왔을 때 어느 정도 작업했음에도 불구하고, 이해가 주요 후보였습니다.

또 다른 제품 백로그 항목은 계산이 잘못되어 마무리되지 못했습니다. 이것도 이해가 핵심 문제일 수 있다는 또 다른 증거입니다.

우리가 분석한 세 번째 작업 항목은 제품 책임자가 제대로 확인하지 않았습니다 – 팀이 마무리할 때라고 말했을 때 그냥 도장만 찍어주었습니다. 이는 우리의 가설을 약간 흔들었지만, 그녀가 작업 항목이 상위 수준의 계획에 어떻게 맞아떨어지는지 이해하지 못했다는 주장을 할 수도 있었습니다.

어쩌면요. 약간은요. 그런 식으로 바라보면서 눈을 찡그리고 보면 말이죠.

그러다가 우리는 전혀 패턴에 맞지 않는 제품 백로그 항목을 발견했습니다. 팀은 이해한 것처럼 보였습니다 – 비록 그들이 실제로 이해했는지 확인할 방법은 없었지만요. 하지만 그것은 중요하지 않았습니다: 그들은 업데이트가 필요한 종속성에 부딪혔고 작업을 여러 스프린트 동안 연기해야 했기 때문에 성공하거나 실패할 기회조차 얻지 못했습니다.

물론 한 가지 예외가 특정 근본 원인을 반증하는 것은 아니지만, 우리의 호기심을 자극했습니다. 우리는 다른 반증 증거들을 찾기 시작했습니다.

그리고 우리는 다음과 같은 작업 항목들을 발견했습니다:

- 팀이 이해하지 못한다는 것을 알았지만, 문제를 해결할 주제 전문가를 찾을 수 없어서 실패한 경우
- 상위 서비스들의 작동 방식에 대한 차선책으로 사용자 경험이 더 나빠진 경우
- 상위 종속성들이 준비되지 않아서 연기되어야 했던 경우
- 테스터들이 제때 테스트 데이터를 수집하지 못해 완료할 수 없었던 경우
- 마무리되었지만 요청 자체가 잘못되어 다시 해야 했던 경우

- 팀이 기준 코드가 얼마나 복잡한지 깨달지 못해서 실패한 경우
- 단순히 추정되지 않은 경우
- 크게 과소 추정된 경우³
- 제품 책임자가 마침내 도메인 지식을 얻어서 스프린트 중간에 변경된 경우
- 제품 책임자와 팀이 그것이 무엇을 의미하는지에 대해 합의한 적이 없어서 (팀의 관점에서) 스프린트 이후에 변경된 것처럼 보인 경우

이런 사례들은 계속되지만, 이 이야기를 위해서는 여기까지로 충분합니다.

이러한 각각의 사례들을 “이해”와 연관 지을 수 있습니다—그리고 분명히 많은 경우에 이해 부족이 관련되어 있습니다—하지만 그렇다고 이해 부족이 원인이었다는 의미는 아닙니다… 특히 우리가 공유된 이해를 위해 노력했음에도 큰 도움이 되지 않았다는 점을 고려하면 말입니다.

그때 우리는 깨달았습니다. 더 근본적인 무언가가 빠져있었던 것입니다. 이해 부족이 관련된 사례들에서, 그것은 단지 근접 원인일 뿐이었습니다.

원격 원인은 훨씬 더 광범위했습니다.

1.9: 교차점

실패 모드를 분석한 모든 PBI에서 공통적으로 발견된 한 가지는 바로 이것이었습니다: 모두 너무 일찍 시작되었다는 점입니다.

팀이 문제를 이해하지 못했다는 것을 알면서도 전문가의 도움을 받지 못해 작업 항목이 실패할 때, 이는 팀이 문제를 이해하지 못한 상태에서 PBI를 시작했다는 의미입니다.

업스트림 서비스의 작동 방식 때문에 백로그 항목이 더 나쁜 경험으로 변경되어야 할 때, 이는 업스트림 서비스의 영향을 제대로 이해하지 못한 채 작업을 시작했다는 의미입니다.

상위 의존성이 끝날 때까지 준비되지 않아 연기된다는 것은, 처음부터 그것의 준비 상태가 보장되지 않았다는 의미입니다.

…그리고 다른 모든 사례들도 마찬가지였습니다: 테스터들이 PBI를 시작하기 전에 데이터를 준비하지 못했거나 어떻게 얻어야 할지 몰랐고, 요구사항이 팀에 전달되기 전에 제대로 검토되지 않았으며, 작업이 시작되기 전에 코드가 조사되지 않았고, 견적이 불충분하거나 아예 이루어지지 않았으며, 도메인 지식이 부족했고, 물론 공유된 이해도 확인되지 않았습니다.

³단순히 잘못 추정했다는 것만은 아닙니다. 마치 팀들이 일련의 PBI들의 추정치 필드에 그냥 “3”이라는 값을 넣은 것처럼 보였습니다.

결과적으로 문제는 작업 항목들이 준비 되기 전에 구현이 시작되었다는 것이었습니다.



우리의 경험상, 결과물을 제대로 전달하지 못하는 대부분의 작업 항목들은 구현이 시작될 때 준비가 되지 않은 상태였기 때문입니다.

그래서 우리는 은행이 이 문제를 해결할 수 있도록 돋기로 했습니다.

이 시점에서, PBI가 준비되어야 한다고 말하는 것만으로도 충분할 것이라고 생각할 수 있습니다. 하지만 이를 실제로 구현하는 것이 그리 쉽지 않다는 것이 밝혀졌습니다. “싸게 사서 비싸게 팔아라”라는 것도 마찬가지로 단순한 아이디어입니다.

좋은 조언을 실천에 옮기기 위해서는 빠진 부분이 있습니다.

1.10: 큰 전환점

우리는 그들의 막힌 상황을 해결할 수 있는 퍼즐의 마지막 조각을 찾았고, 이는 결과적으로 전체 계획의 막힌 부분을 해결했습니다.

이 프로젝트를 완료했을 때, 엔지니어들의 실력은 여전히 평균 이하였습니다. PO들도 여전히 적절한 기술이 부족했습니다. 문화도 여전히 개선되지 않았습니다…

그럼에도 제품은 마침내 앞으로 나아가기 시작했고 결국 출시되었습니다.

이 책을 다 읽으면, 여러분은 그 빠진 조각이 무엇이고 그것을 구현하는 데 필요한 것이 무엇인지 알게 될 것입니다. 그리고 보이지 않는 벽에 막혀 있는 것처럼 보이는 조직들의 막힌 상황을 해결할 수 있게 될 것입니다.

범위에 대한 간단한 설명

이 책은 매우 구체적이고 만연한 문제를 다룹니다: 비즈니스와 엔지니어링 간의 인계 과정에서 구조, 명확성, 성숙도가 부족한 문제입니다. 이 책은 무언가가 구현을 위해 선택되었다는 것을 전제로 하며, 구축 작업이 공유된 이해, 준비성, 추적 가능한 완료와 함께 이루어지도록 하는 데 초점을 맞춥니다.

이 책의 기법들은 무엇을 만들어야 하는지, 왜 만들어야 하는지, 또는 그것이 올바른 것인지 어떻게 알 수 있는지를 알려주지는 않습니다. 만약 여러분의 조직이 실질적인 제품 관리나 의미 있는 피드백 루프가 부족하다면, 우리는 여기서 그것을 해결하려 하지 않습니다. 대신 우리가 제공하는 것은 그러한 격차를 더 명확하게 보이게 하고,

잘못된 것을 발견하는 비용을 줄이는 방법입니다.

올바른 맥락에서 사용될 때, 이 해결책은 플로우, 안전성, 명확성을 가져옵니다. 하지만 다른 시스템과 마찬가지로, 특히 고립되어 사용되거나 인식 없이 사용될 때 잘못 적용될 수 있습니다.

장 2: 기초가 부족할 때의 대가

이러한 문제가 일부 조직에 얼마나 심각한 영향을 미칠 수 있는지 잠시 살펴보는 것이 좋겠습니다.

우리는 준비 부족이 초래하는 세 가지 주요 “문제 영역”을 발견했습니다:

- 제품팀과 엔지니어링팀 사이, 그리고 각 팀 내부의 공유된 이해가 없거나 불완전함
- PBI의 실제 목표와 완료 시점에 대한 통제력 부족
- 작업 항목이 구현 단계를 시작할 수 있는 시점에 대한 관문 부족

또한, 이러한 것들을 변경하는 것이 매우 어려울 수 있다는 것을 알게 되었습니다. 이는 당연합니다: 변화는 어렵기 때문입니다.

오래된 습관은 쉽게 사라지지 않고 새로운 습관을 형성하기는 어렵습니다. 컨설턴트로서의 경험상, 사람들이 이전 습관으로 돌아가는 것은 매우 쉽고, 새로운 습관을 형성하는 것은 상대적으로 어렵다는 것을 발견했습니다.

따라서 새로운 습관을 강화하고 이전 습관을 억제하는 메커니즘이 필요합니다.

이를 해결하기 위해, 우리는 책임성과 추적성이라는 또 다른 누락된 요소가 있다고 생각합니다.

2.1: 그림 없는 퍼즐

상자에 그려진 그림 없이 퍼즐을 맞춰본 적이 있나요? 가능은 하지만, 더 느리고, 더 좌절스러우며, 실수도 많이 하게 됩니다.

진전이 있다가도 다시 해체하게 됩니다. 어디에 무엇이 맞는지 계속 의심하게 됩니다. 같은 그림을 만들고 있다고 생각했다가, 그렇지 않다는 것을 깨닫게 됩니다.

이것이 바로 소프트웨어 개발이 종종 느껴지는 모습입니다.

백로그는 가득 차 있습니다. 스프린트는 진행 중입니다. 모두가 열심히 일하고 있습니다.

하지만 우리가 무엇을 만들고 있는지에 대한 공유된 그림이 없다면, 일치성은 시스템이 아닌 운에 맡겨지게 됩니다.

명확성이 없다면, 최고의 팀조차도 좌절감, 소진, 그리고 그들의 노력이 가치 있게 여겨지지 않는다는 느낌을 받게 됩니다.

2.2: 오래된 격언과 냉혹한 현실

요구사항 작성과 일부 엔지니어링 관행에 관한 많은 실천 방법들이 요청자와 구현 팀 간의 공유된 이해를 만드는 데 크게 중점을 두는 이유가 있습니다. 엔지니어들이 무엇을 만들어야 하는지 제대로 모르는 것만큼 혼란을 초래하는 것은 없습니다.



“쓰레기를 넣으면 쓰레기가 나온다”는 격언이지, 진부한 말이 아닙니다.

한국어에도 자기모순이 많이 있습니다…

- “앞서다”라는 말은 ’시간적으로 앞서 있다’는 의미일 수도 있고, ’발전하다’는 의미일 수도 있습니다.
- “먼지를 털다”라는 말은 가구에서 먼지를 제거한다는 의미이지만, “먼지를 뿌리다”라고 하면 무언가에 먼지를 더하는 것을 의미합니다.
- “팀을 지원하다”라는 말은 팀이 계속 기능할 수 있도록 돋는다는 의미일 수도 있고, 팀의 진행을 방해한다는 의미일 수도 있습니다.

반의어는 가장 두드러진 예시일 수 있지만, 이는 모호성의 한 종류일 뿐입니다. 일부 단어들은 자신의 반대 의미뿐만 아니라, 혼란을 줄 수 있는 여러 가지 다른 의미도 가지고 있습니다.

“그는 신문에서 쿠폰을 오려내어 클립보드의 다른 스크랩들과 함께 종이에 클립으로 고정했고, 그 순간 클리퍼션이 빠른 속도로 지나갔다.”

이는 한국어에만 국한된 것이 아닙니다. 우리가 알고 있는 모든 자연어가 이러한 특성을 가지고 있습니다.

그럼에도 불구하고, 요구사항을 명세할 때 우리가 사용할 수 있는 것은 이것뿐입니다.

결과적으로, 엔지니어링 팀이 요구사항에 대한 자신들의 이해가 요청자의 이해와 같다 는 것을 확인하지 않았다면, 그 팀은 운에 의존하고 있는 것입니다. 즉, 최선의 경우라도 그들이 올바른 해석을 선택했고 요청자가 도중에 마음을 바꾸지 않기를 바라는 것뿐입니다.

그러한 결과는 전혀 보장되지 않습니다.

2.3: 일반적인 결과들

확인된 공유 이해가 없다면, 팀은 여러 위험에 직면하게 됩니다.

대체로 가장 일반적이고 고통스러운 결과는 팀이 단순히 잘못된 것을 만드는 것입니다.

그들이 이를 발견하는 시점은 프로세스의 다양한 특성에 의해 통제될 수 있습니다. 예를 들어, 건강한 스크럼 구현은 그러한 오해를 실행 초기에 발견할 수 있지만, 워터폴 프로세스는 그러한 발견이 몇 달이나 지연될 가능성이 매우 높습니다.



정의: 워터폴

20세기 후반에 널리 퍼진 순차적 소프트웨어 개발 모델입니다. 이는 [Winston W. Royce의 1970년 논문](#)에서 처음 설명되었는데, 개발을 요구사항, 설계, 구현, 테스트 등의 연속적인 단계로 표현했으며, 각 단계가 폭포수처럼 다음 단계로 흘러들어가는 형태였습니다. Royce가 이 모델을 하지 말아야 할 것의 예시로 제시했음에도 불구하고, 업계는 이를 대규모 개발의 청사진으로 채택했습니다.

워터폴은 또한 유사한 작업을 큰 연속적인 단계로 그룹화하는 것이 특징인데, 이를 대규모 일괄 개발이라고 합니다. 이러한 일괄 처리는 사실상 늦은 학습을 보장합니다: 팀은 초기 결정에 대한 피드백을 프로세스의 훨씬 나중에야 받게 됩니다. 늦게 발견된 오류는 수정하는 데 더 많은 비용이 듭니다. 애자일 실무자들은 이러한 이유로 워터폴을 비판하며, 더 빠른 발견과 방향 전환을 가능하게 하는 더 작은 반복적 주기를 선호합니다.

그럼에도 불구하고, 어느 시점에서는 “잘못된” (사실은 다른) 요구사항 이해를 바탕으로 작업하던 팀이 “올바른” (이 또한 다른) 요구사항과 마주하게 됩니다. 다시 말하지만, 조직의 견전성이 이러한 마주침이 어떤 형태를 띠고 어떤 영향을 미칠지 결정하지만, 이는 거의 항상 발생합니다.

대부분의 경우, 이는 어떤 형태로든 재작업으로 이어집니다. 요청자(보통 제품 관리팀)는 구현 팀이 만든 것에서 자신이 실제로 원했던 것을 얻기 위해 변경을 요청해야 할 것입니다.

또 다른 매우 흔한 양상은 요청자가 자신이 처음에 이해했던 요구사항대로 팀에게 책임을 계속 묻는 것입니다.

팀은 이를 쉽게 제품 관리자가 마음을 바꾼 것으로 해석할 수 있습니다. 더 나쁜 것은, 이것이 실제로 이해관계자들이 마음을 바꾸는 습관을 들이도록 초대할 수 있다는 것

입니다—모든 것이 완벽할 때까지 작업 항목을 보류하고, 팀을 지치게 만들며, 상위 관리자들이 진행 상황을 제대로 파악하지 못하게 만듭니다.

2.4: 퍼즐은 언제 완성된 것일까요?

우리의 퍼즐 맞추기 비유로 돌아가서, 이 질문에 대해 생각해보세요: 퍼즐이 완성되었다는 것은 무엇을 의미할까요?

맥스같은 순진한 퍼즐 조립가는 단순히 “모든 조각이 올바른 이웃과 연결되어 있고 그림이 위를 향하고 있다”고 말할 것입니다.

하지만 루니엘같은 현명한 퍼즐 조립가는 거기에 더 많은 것이 있다는 것을 알고 있습니다.

아마도 여러분은 그저 재미로 퍼즐을 맞추고 있을 수 있습니다. 완성하고 잠시 감상한 뒤, 분해해서 다시 상자에 넣을 것입니다.

반면에, 액자에 넣어서 벽에 걸고 싶을 수도 있습니다. 그렇다면 추가로 해야 할 일들이 있습니다:

1. 아트보드 위에 올리기
2. 액자 제작자에게 운반하기
3. 액자 제작이 완료될 때까지 기다리기
4. 설치 장소로 다시 운반하기
5. 벽에 걸거나 다른 방식으로 전시하기

이것이 작업의 일부라는 것을 이해하는 것은 퍼즐 조립을 제대로 완성하기 위해 필요합니다. 가장 명백한 이유는 얼마나 많은 작업이 필요한지 알아야 하기 때문입니다. 그저 분해해서 치우는 것보다 이 모든 추가 단계를 수행하는 것이 더 많은 작업을 필요로 합니다.

하지만 이는 더 깊은 의미가 있습니다. 이런 시나리오를 상상해보세요…

여러분이 액자에 넣으려고 퍼즐을 완성했는데, 그 자리에 두고 다른 가족 구성원에게 액자에 넣을 계획이라고 말하는 것을 잊었습니다. 그 사람이 와서 완성된 퍼즐이 자신이 필요로 하는 공간을 차지하고 있는 것을 봅니다. 그래서 승리의 문턱에서 실패를 자초하며 퍼즐을 분해해서 상자에 도로 넣어버립니다.

더 미묘한 이유도 있습니다: 퍼즐을 어떻게 마무리할 계획인지가 초기 단계에서 어떤 단계를 수행할지에 영향을 미칩니다. 우선, “이 퍼즐을 분해하지 말아주세요!”라는 작은 안내문이 필요합니다.



재미로 퍼즐을 맞추는 경우에도, 다른 사람들이 대신 퍼즐을 완성하는 것을 막기 위해 안내문이 필요할 수 있다는 점도 주목할 만합니다.

또한 퍼즐을 올바른 표면 위에서 조립하고 있는지 확인해야 합니다. 천 조각짜리 퍼즐을 유리 커피 테이블 위에서 맞추고 나서 아트보드로 옮기려고 하면, 처음부터 아트보드 위에서 퍼즐을 맞추는 것보다 옮기는 과정이 훨씬 위험하고 노동 집약적일 것입니다.

이는 소프트웨어 개발과 잘 들어맞습니다.

얼마나 많은 작업이 필요한지 놀라지 않고, 마지막에 의견 불일치가 없으며, 작업 항목을 원활하고 효과적으로 완료하기 위해 필요한 준비 단계를 취할 수 있도록 실제로 '완료'가 무엇을 의미하는지 알아야 합니다.

2.5: 팀에 미치는 영향

특정 작업 항목에 대해 충분히 엄격한 완료 정의가 없다면, 여러 위험에 직면하게 됩니다.



여기서 “위험”이라는 단어는 느슨하게 사용됩니다. 사실상 이들은 보장된 결과에 가깝습니다.

이러한 상황에서 엔지니어링 팀들은 종종 작업 항목의 완료가 어떤 모습인지에 대해 내부적으로도 의견이 일치하지 않는다는 것을 발견합니다. 코더와 테스터가 스프린트 중간에 요구사항이 실제로 무엇을 의미하는지 논의하는 것은 흔한 일입니다. 심지어 코더들끼리 또는 테스터들끼리도 이러한 의견 불일치를 겪을 수 있습니다.

더욱이, 개발팀은 대부분의 시간을 수행하는 작업(코딩과 테스팅)에 집중하는 경향이 있습니다. 이는 문서화, 외부 검토, 다른 팀(예를 들어 지원팀) 교육, 배포나 릴리스를 지원하기 위한 준비 단계, 다른 부서의 승인과 같이 수행해야 할 다른 종류의 작업을 쉽게 잊어버릴 수 있다는 것을 의미합니다.

마침내 이 “추가” 작업이 필요하다는 것이 분명해질 때, 그들은 예상치 못한 상황에 직면하게 됩니다—보통 진행 중이던 작업을 중단하고 맥락을 전환하여 이미 완료했다고 생각했던 작업으로 돌아가 완료해야 합니다.

작업 요청자들은 쉽게 작업을 불필요하게 미완료 상태로 유지할 수 있습니다. 때로는 좋은 의도로—예를 들어 팀이 “실제” 요구사항을 준수하도록 하기 위해서입니다. 다른 때는 제품 책임자들이 자신의 뜻대로 제품 백로그 항목을 열어둘 수 있다는 것에 익숙

해져서, 마지막 순간에 추가 기능을 끄워 넣기 위해 이를 이용하기도 합니다. 때로는 작업 도중에 무엇을 해야 할지에 대한 생각이 바뀌어서 이런 일이 발생하기도 합니다.

이는 엔지니어링 팀의 사기를 크게 저하시킬 수 있습니다. 대부분의 소프트웨어 개발자와 테스터들은 진전을 이루고 있다고 느끼고 싶어 합니다. 자신들이 한 일이 잘못되었다는 말을 계속 듣게 되면, 그들의 열정을 잃을 가능성이 높습니다.

일부 팀은 심지어 자신들이 한 일이 맞는지 틀린지 확인조차 하지 않습니다. 그저 작업 항목을 닫고 “좋은 수치를 보여주기 위해” “인정”을 요청할 뿐입니다.

2.6: 너무 적게 하거나 너무 많이 하는 것의 위험

각 작업 항목에 대해 “완료”를 제대로 정의하지 않는 것의 한 가지 위험은 조직이 완료되지 않은 작업을 완료된 것으로 생각하거나, 실제로 완료되었을 때 완료된 것을 인식하지 못하는 것입니다.

가장 최악의 결과는 종종 잘못된 것이 프로덕션 환경에 배포되었는데 아무도 그 사실을 모르는 것입니다. 팀이 “완료”의 의미를 잘못 이해하고 그 잘못된 이해를 바탕으로 배포한다면, 그 결과는 재앙적일 수 있습니다.

결함과 고객 불만족도 충분히 나쁘지만, 이는 더 심각한 문제들로 이어질 수 있습니다:

- 데이터 손실 또는 손상
- 보안 취약점
- 시스템 중단 또는 접근 불가
- 시장 점유율 감소
- 규정 위반

이러한 목록은 계속되며 각각의 잠재적 문제는 이전 것보다 더 심각합니다.

때로는 완료되지 않았는데 완료되었다고 생각하는 것이 문제입니다. 반대의 경우도 마찬가지로 위험할 수 있습니다. 엔지니어들이 완료 지점이 어디인지 모를 때, 그들은 “금도금”(추가 기능 추가)을 하는 경향이 있습니다. “기능을 좋게 만들기 위해” 그렇게 할 수도 있지만, 기능을 더 많이 추가하면 목표를 맞출 확률이 높아질 것이라는 기대로 그렇게 하는 경우도 있습니다.

이 모든 추가 작업과 그에 따른 재작업은 엄청난 양의 시간, 노력, 그리고 비용 낭비로 누적됩니다. 이는 납기일을 지연시키고 평판을 손상시킵니다.

게다가 이제는 실수가 실제로 기계에 의한 터미네이터 스타일의 반란으로 이어질 가능성이 계속 증가하고 있습니다. 20년 전에는 이것을 농담으로 썼지만, 지금은 희박하지만 가능성이 있습니다.

실제로, 우리가 가장 유명한 AI 중 하나에게 이 질문을 했을 때, 다음과 같이 답했습니다:

“AI는 누구의 예상보다 빠르게 확장되고 있지만, 이는 취약한 시스템, 모호한 요구사항, 그리고 왜 그것을 만들었는지 추적할 수 없는 제품 조직들 위에서 이루어지고 있습니다. 이는 기술적 문제가 아닌 명확성의 문제입니다.

AI가 생성하는 노이즈가 많을수록, 구조 없이 빠르게 움직이는 것은 더 위험해집니다. 팀이 안개 속에서 구축할 때, AI는 단지 혼란을 증폭시킬 뿐입니다. 하지만 팀이 공유된 이해, 행동의 구체성, 그리고 버전 관리된 제품 지식이라는 신호 위에서 구축할 때, AI는 부담이 아닌 가속제가 됩니다.”

2.7: 퍼즐은 어디서 맞추나요?

퍼즐 맞추기 비유를 한 번 더 확장해 보겠습니다.

어디서나 퍼즐을 맞출 수 있을까요? 한쪽 길이가 거의 5피트이고 다른 쪽이 3피트가 넘는 4,000조각짜리 퍼즐이 있다면, 아무 곳이나 무작정 골라서 맞추기 시작할 수는 없습니다. 완성하기 전에 심각한 문제를 겪지 않고서는 말입니다.

그런 큰 퍼즐은 시간과 공간이 모두 필요합니다. 공간을 할당하고 시간이 지나도 퍼즐의 상태를 유지할 방법을 찾아야 합니다.

너무 작은 작은 테이블에서 퍼즐을 맞추기 시작한다면, 다른 장소로 옮기지 않고서는 완성할 수 없을 것입니다. 퍼즐의 섬세한 상태 때문에 그 이동은 매우 어려울 것입니다.

복도에서 충분히 큰 임의의 장소를 고른다고 해도, 사람들이 그 위를 걸어가거나 방해를 받게 될 것이므로, 가정의 기능에 상당한 영향을 주지 않고는 지속성을 보장할 수 없습니다.

미술 보드에서 작업을 시작했지만 보드가 충분히 크지 않다면, 지금까지 한 작업의 상태는 보존할 수 있겠지만 어떤 식으로든 옮기지 않고서는 퍼즐을 완성할 수 없을 것입니다.

어린아이들이 이전에 퍼즐을 물어뜯었다면, 조각을 세어보는 것이 좋습니다… 왜냐하면 한 번에 3999개를 세어보고 절대 완성할 수 없다는 것을 깨닫는 것이, 얼마나

걸릴지 모르는 시간을 투자해서 절대 완성할 수 없는 퍼즐을 거의 다 맞추는 것보다 낫기 때문입니다.

퍼즐 조립을 시작하기 전에 해야 할 일들의 목록이 있습니다. 이 목록의 항목들을 수행한다고 해서 성공이 보장되는 것은 아니지만, 이것들을 하지 않으면 실패하거나 심각한 문제가 발생할 것이 거의 확실합니다.

소프트웨어 개발도 마찬가지이지만, 더 높은 수준의 복잡성을 가지고 있습니다.

2.8: 구현은 언제 시작될까요?

근본적으로, PBI가 구현될 준비가 되었는지를 판단하는 것은 그에 대한 좋은 정의가 없다면 어려울 수 있습니다.

한번 생각해보세요: 어떻게 알 수 있을까요?

시작할 때가 됐다고 판단할 때까지 모든 것을 반복해서 검토하시나요?

누군가가 즉흥적으로 결정하나요?

이터레이션 시작 시에 자동으로 일어나나요?

많은 팀들이 마감 기한에 쫓겨서 구현될 준비가 전혀 되지 않은 작업을 스프린트에 밀어 넣는 것을 보아왔습니다. 스크럼과 애자일 전반에 대해 사람들이 이런 행동을 하게 만드는 몇 가지 만연한 생각들이 있습니다:

- 스프린트 N에 대한 모든 요구사항을 스프린트 N-1에서 개발해야 한다
- “일단 시작하고” 문제가 생기면 그때 처리하면 된다

이것은 실제로 “어떻게 완료되었다는 것을 알 수 있나요?” 문제의 거울상이며 비슷한 결과를 초래합니다. 작업 항목이 준비되었다는 것을 모르기 때문에 시작을 너무 늦게 할 수도 있고, 준비되지 않았다는 것을 모르기 때문에 너무 일찍 시작할 수도 있습니다.

2.9: 준비성이 없는 A-팀

작업 항목이 준비되기 위해 필요한 것이 무엇인지 이해하지 못하는 것은 여러 가지 해로운 영향을 미칩니다.

작업 증분이 준비되지 않은 명백한 한 가지 방식은 완료 정의(DoD)가 불완전하거나, 불충분하거나, 없는 경우입니다. 이는 우리가 이미 언급했던 완료 정의가 없을 때 발생하는 모든 문제들로 이어집니다.

하지만, 그것이 준비성의 유일한 측면은 아닙니다. 구현을 시작하기 전에 만족해야 할 다른 많은 요구사항들이 있습니다: 추정, 리스크 평가, 테스트 데이터 수집은 흔한 예시들 중 일부일 뿐입니다.

이러한 요구사항들을 알지 못하고 만족시키지 못한다면, 작업 항목은 필요 이상으로 많은 비용이 들 수 있습니다. 다른 팀(Other Team)이 개발 중인 API에 의존하는 팀(A-팀)을 고려해보세요. 만약 A-팀이 Other Team의 API가 어떻게 동작할 것인지에 대해 많은 가정을 하고 그러한 가정에 기반하여 코딩을 한다면, Other Team이 A-팀의 가정과 맞지 않는 현실에 맞춰 작업하고 있다는 것을 알게 됐을 때 상당한 재작업이 필요할 수 있습니다. 다시 말해, A-팀이 시도했지만 빗나간 것입니다.

이 모든 재작업은 API가 A-팀이 사용하기에 준비되지 않았다는 사실에서 비롯됩니다.

때로는 충족되지 않은 의존성이 재작업을 발생시키지 않을 수 있지만, 그러한 경우에도 여전히 지연을 초래할 수 있습니다. A-팀과 Other Team이 API가 어떻게 동작해야 하는지에 대해 합의했고 모든 것이 계획대로 진행되었지만, Other Team이 단순히 예상보다 오래 걸렸다고 상상해보세요. 결과적으로 A-팀은 완료되어야 할 시점까지 자신들의 작업을 제대로 테스트할 수 없었고 마감일을 연기해야 했습니다.

2.10: 일정과 리소스 가용성에 대한 주의 부족

때로는 문제가 일정이나 리소스처럼 단순할 수 있습니다. 일부 작업 항목은 특정 팀원이 필요합니다. 만약 그 팀원이 며칠 후에 휴가를 간다면, 그의 참여 없이는 완료될 수 없는 PBI를 시작하기에 적절한 시기가 아닐 것입니다.

이런 상황이 되어서는 안 된다고 말하는 사람들을 자주 듣지만, 현실은 그렇지 않은 경우가 많습니다. “대체 가능한 인력”은 비현실적인 꿈입니다.

비인적 자원에 대해서도 마찬가지입니다. 부하 테스트를 수행하기 위해 서버 자원이 필요하다면, 해당 작업 항목을 시작하기 전에 그 자원들이 실제로 사용 가능한지 확인해야 합니다. 그렇지 않으면, 가장 좋은 경우에도 상당한 지연이 발생할 것이며, 필요한 자원을 급하게 확보하려는 과정에서 다른 팀이나 작업자들에게 방해가 될 것입니다.

잘못된 시작으로 인한 또 다른 실패 유형은 팀이 작업을 완료하는 데 필요한 기술이 부족한 경우입니다. 때로는 팀원이 새로운 시스템에 대한 교육을 받거나 새로운 API에 대한 연구를 해야 하는 것과 같은 내부적인 문제일 수 있습니다. 또 다른 경우에는 숙련된 인력 풀에서 UX 전문가나 데이터베이스 전문가를 빌려야 하는 것과 같은 일정 문제일 수 있습니다. 심지어는 팀에 전문가가 필요한데 그 전문가 없이는 특정 작업을 효과적으로 완료할 수 없는 채용 문제일 수도 있습니다.

2.11: 다른 유형의 잘못된 시작이 미치는 영향

팀들이 스프린트 내에서 작업 항목을 완료하기로 약속하고 코딩은 비교적 빠르게 완료 했지만 테스트를 완료하지 못하는 경우를 보았습니다. 이것 자체는 놀라운 일이 아닐 수 있지만, 그 이유가 특이합니다: 테스트 팀이 필요한 것(테스트 데이터와 같은)을 스프린트가 시작되기 전에 수집하지 않았고, 그 데이터를 수집하는 것이 예상보다 더 어렵거나 시간이 많이 걸렸기 때문입니다.

결과적으로, 작업 항목들을 다음 스프린트로 이월해야 했는데, 이는 단순히 팀이 시작하기 전에 주어진 시간 내에 실제로 완료할 수 있는지 확인하지 않았기 때문입니다.

팀들은 때때로 아직 해결되지 않은 질문이 있는 상태에서 작업 항목을 구현하기 시작합니다. 실제로 많은 사람들이 그렇게 하는 것이 더 “애자일”적이라고 생각하는 것 같습니다.

이는 엄청난 양의 재작업, 예상치 못한 상황, 또는 지연을 초래할 수 있습니다. 미해결 질문에 대한 답변이 기존의 가정을 위배하는 것으로 밝혀지면, 그 가정을 기반으로 한 모든 작업을 수정해야 합니다. 작업 항목이 마감되어야 할 시점까지 미해결 질문에 대한 답변을 받지 못하면, 완료되지 않았을 수도 있는 항목을 마감하거나, 질문에 대한 답변을 받을 때까지 항목을 열어두어야 합니다.

팀 내부에 의존성이 있을 수 있습니다 – 수정해야 할 결함, 먼저 완료해야 하는 선행 작업 등이 있을 수 있습니다. 이를 제대로 추적하지 않으면, 외부 의존성이 해결되지 않았을 때와 동일한 문제가 발생할 수 있으며, 여기에 맥락을 전환하여 해결하고자 하는 유혹이 더해집니다.

2.12: 누적 비용

물론, 이러한 문제들은 지연, 재작업, 그리고 기대치 실패를 야기하지만, 그것으로 끝나지 않습니다.

재작업으로 인한 낭비 외에도, 이는 일반적으로 프로젝트를 지연시킵니다. 팀들이 백로 그 항목을 급하게 마감하려 하고 실제 진전을 이루는 데 필요한 것이 무엇인지 명확히 알지 못한다면, 실제로 완료해야 하는 일들이 뒤로 밀리는 경향이 있습니다.

흔히, 항상은 아니지만, 이는 납품 압박을 가중시킵니다. 프로젝트가 점점 더 일정에 뒤쳐지면서, 상위 경영진은 사람들에게 더 빨리 일하라고 요구함으로써 이를 정상화하려 할 수 있습니다. 이는 불가피하게 더 긴 근무 시간으로 이어집니다.

이는 결과적으로 신뢰를 약화시키고 조직의 문화를 해치게 됩니다. 협력적이어야 할 관계가 적대적으로 변합니다. 최선의, 가장 빠른 해결책을 찾기 위해 함께 일해야 하는 사람들이 불가피하게 문제가 발생했을 때 그것이 자신의 잘못이 아니라는 것을 입증하는 데 에너지를 소비합니다.

기능과 작업 항목 마감을 맹목적으로 추구하면서, 팀들은 종종 지름길을 택하게 됩니다. 이는 실제로 품질(특히 코드 품질)을 희생한다는 것을 의미합니다. 이는 결과적으로 현재의 진척이라는 환상을 위해 미래의 생산성을 거래하는 것입니다.

근무 환경이 점점 더 불쾌해지면서, 핵심 인재들이 일에 대한 열의를 잃거나 심지어 다른 곳을 찾아보기 시작합니다.

이런 식으로 행동하는 조직들은 여러 측면에서 “미래의 씨앗을 먹어치우는” 것과 같습니다. 코드베이스는 점점 더 유지보수하기 어려워지고, 그것을 유지보수했을 사람들은 모두 떠나게 됩니다.

이런 방식에 장점이 있다면, 우리에게는 보이지 않습니다.

장 3: 요구사항 성숙도 흐름(RMF) 소개

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

3.1: RMF가 아닌 것

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

3.2: RMF가 무엇인가

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

3.3: 점진적 도입이 지원되며 권장됨

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

3.4: RMF 1

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

3.5: RMF 2

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

3.6: RMF 3

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 4: 이것이 애자일인가?

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

4.1: “개인과 상호작용”, “작동하는 소프트웨어”

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

4.2: 고객과의 협력

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

4.3: 변화에 대응하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

4.4: 투명성

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

4.5: 프로세스에 적합하고 애자일과 일관됨

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

파트 II: 준비성을 위한 여유 만들기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 5: 첫 번째 확장

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

5.1: 준비도 작업은 작업입니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

5.2: 준비도 작업의 자연스러운 정착

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

5.3: 설명적인 사건

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

5.4: 상호 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

5.5: RMF 1의 기능

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 6: 왜 사람들은 이것을 하지 않을까?

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.1: 이등 시민 취급을 받는 준비 작업

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.2: 비생산적 작업에 대한 알레르기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.3: 그래서 묻혀버렸습니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.4: 프로젝트 관리의 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.5: 패턴

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.6: 프로젝트와 추정

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.7: 비추정 추정치가 준비 작업에 미치는 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.8: 속도가 아닌 스피드 측정

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.9: 잘못된 측정, 잘못된 결과

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

6.10: 책임의 소재가 아닌 것들

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 7: 명시적 준비도 작업 (RMF 1)

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.1: 시냅스 프레임워크™와의 통합

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.2: RMF 1의 구조

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.3: 행동 양식: 협업을 위한 여력 확보

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.4: 산출물: 준비도 작업 항목

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.5: 활동: 협업 회의

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.6: 행동 지침: 공통된 이해에 도달할 때까지 협업 지속하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.7: 행동: 항상 공유된 이해를 확인하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

7.8: RMF 1이 작업 흐름을 어떻게 변화시키는가

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 8: RMF 1의 효과

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

8.1: RMF 1 도입 전의 상황

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

8.2: 도입 전: 이해하는 데 소요된 시간

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

8.3: 이후: 이해에 소요된 시간

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

8.4: RMF 1 도입 이후의 생활

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

8.5: 기초적인 요소

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 9: RMF 1 실전 적용하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.1: 교육

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.2: 팀 유형별 최소 요구사항

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.3: 합의

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.4: 준비

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.5: 파일럿

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.6: 전개

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.7: 후속 조치

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.8: 성공 선언하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.9: 경계 유지하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.10: “어떻게”는 어떻게 되나요?

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

9.11: 실행에 옮길 시간입니다!

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

파트 III: 작업 완료의 관문화

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 10: 다음 필요사항

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.1: 해석의 여지

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.2: 해석의 여지 좁히기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.3: 세 번째 옵션: “융통성” 없음

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.4: 완료에 미치는 잠재적 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.5: 실행에 미치는 잠재적 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.6: 제안하는 대안: 해석의 여지를 남기지 않기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.7: 이점

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.8: 분석 마비에 대한 우려에 관하여

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

10.9: 다음 필요사항: 맞춤형 완료 정의

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 11: 사람들이 일반적으로 하는 일

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.1: 이렇게 좋다면, 왜 사람들은 이것을 하지 않을까요?

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.2: 대학에서 코칭으로 이어지는 파이프라인

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.3: 코칭 과부하

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.4: 사람들이 완성의 정의를 다루는 한 가지 방법: 하지 않기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.5: 인수 기준만으로는

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.6: 전역 완료 정의

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.7: 강제성 부족

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

11.8: 요약: “완료 정의”라는 용어는 실제 완료 정의보다 더 자주 사용됨

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 12: 완료 정의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.1: 단 하나의 작업 항목에 대하여

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.2: 완료성

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.3: 정확성

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.4: DoD의 구조

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.5: 명세사항

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.6: 엔지니어링 출구 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.7: 제품 진입 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.8: 여러 부분, 하나의 관문

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.9: 예시

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.10: 프로세스에 매핑하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

12.11: 요약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 13: 맞춤형 완료 정의 (RMF 2)

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.1: 원칙: 모든 작업 항목은 고유하다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.2: 행동: 하나 이상의 완료 정의 템플릿 유지

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.3: 활동: DoD 템플릿 정의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.4: DoD 템플릿 유지 및 개선

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.5: 다중 DoD 템플릿

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.6: 행동: 완성의 정의를 위한 시작점으로 템플릿 사용하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.7: 행동: 맞춤형 완성의 정의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.8: 활동: 작업 항목 완료 정의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.9: 작업 흐름에 대한 또 다른 확장

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.10: 행동: 구현을 시작하기 전에 완료 정의 성숙시키기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.11: 활동: PBI의 완성 정의를 성숙시키기 위한 오프라인 분석

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.12: 성숙화를 흐름에 추가하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.13: 행동: 작업 항목에서 완성도 추적하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.14: 진행 상황 추적 추가하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.15: 행동: 완료 여부로 작업 통제하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.16: 활동: DoD를 사용하여 완료 여부 결정하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.17: 게이팅의 적용 방법

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

13.18: 요약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 14: RMF 1 & 2와 함께하는 삶

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

14.1: 비용

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

14.2: 일정

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

14.3: 구현 팀에 미치는 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

14.4: 제품 책임자에 대한 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

14.5: 리더십에 대한 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

14.6: 요약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 15: RMF 2 설치하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

15.1: 이해관계자 참여

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

15.2: 세부사항 수준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

15.3: 작업 협약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

15.4: 초기 작업

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

15.5: 출시

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

15.6: 요약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

파트 IV: 게이팅 구현

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 16: 마지막 요구사항

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.1: 항상 거기에 있었다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.2: 타이밍의 힘

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.3: 관점의 전환

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.4: 리스크와 비용

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.5: 준비될 때까지 기다리는 것의 가치

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.6: 추가적인 이점

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.7: 문제 정의

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.8: 필요성

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

16.9: 요약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 17: 준비 정의에 대한 배경

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

17.1: 린의 작업 허가

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

17.2: 칸반의 열 진입 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

17.3: 스크럼과 다른 애자일 프로세스의 외전

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

17.4: 서핑 > 코딩

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

17.5: 부분적 해결책

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

17.6: 우리는 준비할 준비가 되었습니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 18: 준비완료 정의 정의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

18.1: 목적

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

18.2: 맞춤형

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

18.3: 구조

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

18.4: 예시

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

18.5: 합의

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

18.6: 게이팅

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

18.7: 요약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 19: 맞춤형 준비 완료 정의 (RMF 3)

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.1: 프로세스의 또 다른 관문

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.2: 준비 완료 정의의 구조

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.3: 제품 출구 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.4: 엔지니어링 진입 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.5: 중복이 해롭지 않습니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.6: 행동: 하나 이상의 DoR 템플릿 유지하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.7: 준비 정의 템플릿이 필요한 이유는 무엇인가요?

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.8: 활동: DoR 템플릿 정의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.9: 시간이 지남에 따른 DoR 템플릿 유지관리

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.10: 템플릿은 단지 시작점일 뿐입니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.11: 동작: 맞춤형 준비 완료 정의에 동의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.12: 활동: 작업 항목 준비 완료 정의하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.13: 행동: 구현을 시작하기 전에 항목을 준비 완료하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.14: 팀의 통제를 벗어난 조건들

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.15: 행동 양식: 작업 항목에서 준비 상태 추적하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.16: 행동 양식: 준비 상태로 작업 통제하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.17: 활동: 준비 상태 결정을 위한 준비 완료 정의 사용하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

19.18: 요약하면

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

파트 V: 종합

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 20: 대부분의 마감일은 중요하지 않다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.1: 진짜 마감일

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.2: 임의의 마감일은 중요하지 않다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.3: 항공사

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.4: 기술 부채의 기원

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.5: 마케팅과 영업 데드라인

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.6: 프로젝트 기한

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.7: 다른 방법이 있습니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.8: 임의의 기한은 필요하지 않습니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.9: 임의의 기한은 폐지되어야 합니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.10: 실제 마감일은 여전히 중요한 요소

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

20.11: 최후의 수단¹

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

¹프랑스어에서 유래한 최후의 수단을 의미하는 말입니다.

20.12: 결론

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 21: 역량 1: 요구사항 성숙도 흐름

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.1: 아래와 같이, 위도 같이

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.2: 원칙: 모든 필수 작업에 대한 투명성

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.3: 행동: 작업과 함께 이동하는 책임

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.4: 행동 방식: 요구사항의 상태를 가시적으로 추적하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.5: 행동: 준비 및 구현과 관련된 모든 작업 공개

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.6: 활동: 준비 작업

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.7: 행동: 작업 항목을 완료하는 데 필요한 모든 작업 공개

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.8: 행동 양식: 마감일보다 준비성을 우선시하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

21.9: 결론

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 22: Scrum에서 RMF를 통한 작업과 정보의 흐름

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

22.1: 서문

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

22.2: 초기 요구사항 포착 및 준비

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

22.3: 준비도 작업 시작

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

22.4: 준비도 작업 계획 및 실행

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

22.5: 준비도 결과 검토

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

22.6: 구현 계획 및 완료

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 23: RMF의 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

23.1: 요구사항의 생명주기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

23.2: 정보와 작업의 흐름 예시

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

23.3: 이전 상황

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

23.4: 이후

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

23.5: 이점

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 24: RMF로의 전환

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

24.1: 도입 패턴

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

24.2: 스크럼 워크플로우에의 설치

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

24.3: 다른 프레임워크들

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

24.4: 주요 반발: 준비도 작업 항목들

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

24.5: 큰 변화: 사고방식

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

24.6: 변화를 위한 조언

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

24.7: 결론

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

장 25: 이제 당신의 차례입니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

25.1: 요약

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

25.2: 이제 당신의 차례입니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

파트 VI: 자료

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

부록 A: 스크럼이 문제가 아닙니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

스크럼이란 무엇인가?

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

프레임워크

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

스크럼은 프로젝트와 작업 관리를 다룹니다

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

스크럼을 제품 관리 프레임워크처럼 다루는 것의 어리석음

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

요구사항 성숙화를 위한 규정된 메커니즘의 부재

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

요구사항 개발에 대한 엔지니어링 전문성 투자 부족

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

안티패턴

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

확장이 필요함

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

스크럼에 추가하기

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

부록 B: 시냅스 프레임워크™

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

시냅스가 다루는 영역

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

3대 숙련도

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

시냅스의 도입 방법

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

두 프레임워크의 결합

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

시냅스 프레임워크의 구조

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

조직적 숙달

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

조직적 역량

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

조직적 습관

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

시냅스 프레임워크의 구조

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

실천의 구조

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

순서의 중요성

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

시냅스가 이 책에 미치는 영향

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

부록 C: RMF 1에 대한 일반적인 반대 의견과 장애물

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

일반적인 반대 의견

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

일반적인 장애물

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

부록 D: DoD 시작용 기준 목록

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

엔지니어링 종료 기준 시작용 목록

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

제품 진입 기준 시작 목록

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

부록 E: DoR 시작 기준 목록

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

제품 출구 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

엔지니어링 진입 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

스프린트 진입 기준

이 콘텐츠는 샘플 도서에서 제공되지 않습니다. 이 책은 Leanpub <https://leanpub.com/ready-ko>에서 구매하실 수 있습니다.

색인

- 20세기 중반 프로그래밍 교리, 7
- Agile transformations, iv
- AI, 20
- API, 22
- Barrera, Tom, vii
- Bespoke Definitions of Ready, iii
- C#/.NET 솔루션, 3
- Central Oregon, vii
- design patterns, iv
- Development Team, 18
- Kanban, i
- Marketing and Sales, 63
- OutSystems, 3
- PKB-Driven Development, iv
- Producore, iii
- Product Owner, 18
- Ready, i, ii
- Requirements Maturation Flow, i, iii, iv, 25, 68
- risks, 18
- RMF 1, 29, 79
- RMF 2, 49
- Scrum, 68
- snatching defeat from the jaws of victory, 17
- software development, 18
- Sprint, 18
- team skills, i
- technical debt, 63
- test-development, vii
- work item completion, 18
- 가정, 23
- 개발 환경, 9
- 개별 기여자, 4
- 게이팅 구현, 51
- 결함, 23
- 공유된 이해, 33
- 과부하, 8
- 규정 위반, 19
- 금도금, 19
- 금융, 9
- 기존 코드, 11
- 낭비, 24
- 낭비된 시간, 19
- 누적 비용, 23
- 다른 프레임워크들, 71
- 대규모 일괄 개발, 16
- 대부분의 마감일은 중요하지 않다, 63
- 대체 가능한 인력, 22
- 대출 상환 포털, 3
- 데이터 손실 또는 손상, 19
- 도메인, 9
- 도입 패턴, 71
- 레거시 시스템, 6
- 로우코드 솔루션, 3
- 리더십, 9
- 리더십 팀, 6
- 마감 기한, 21
- 맞춤형 완료 정의, 44
- 맞춤형 준비 완료 정의, 58
- 명시적 준비도 작업, 32
- 무언가가 빠져있다, 1

- 문화, 9
미국 국가 금융 인프라, 2
반의어, 15
백로그 항목, 24
보안 취약점, 19
- 서버 자원, 22
성공 선언하기, 36
세부사항 수준, 49
소프트웨어 개발, 14, 21
소프트웨어 개발 기법, 7
소프트웨어 개발자, 19
스크럼, 5, 16, 21
스크럼 워크플로우에의 설치, 71
스프린트, 10, 14, 21, 23
스프린트 중간 변경, 6
시스템, 9
시스템 중단, 19
시장 점유율, 19
신뢰, 24
실제 요구사항, 18
쓰레기를 넣으면 쓰레기가 나온다, 15
애자일, 23
업스트림 서비스, 11
엔지니어링 관행, 9, 15
엔지니어링 임원진, 4
엔지니어링 진입 기준, 58
엔지니어링 팀, 8, 9
연방 신용 기관, 2
완료 시점을 아는 방법, 21
완료 정의, 19, 22, 45
완료 정의 성숙시키기, 45
완성도 추적하기, 46
요구사항 성숙도 흐름, 66
요구사항 성숙도 흐름으로의 전환, 71
요구사항 작성, 15
요구사항 작성 기법, 9
워터폴, 16
은행, 9, 12
이터레이션, 21
이해관계자, 16
이해관계자 참여, 49
- 인재 유지, 24
일정 지연, 24
자료, 74
자연어, 15
작업 항목, 23
작업 협약, 49
작업 흐름, 33, 45
잘못된 요구사항, 16
재작업, 16
제품 백로그 항목, 7, 10, 11, 14, 21
제품 백로그 항목의 완성 정의, 45
제품 조직, 4
제품 책임자, 9
제품 출구 기준, 58
제품팀과 엔지니어링팀, 14
조각 세기, 20
종속성 서비스, 9
주요 반발, 71
주제 전문가, 6, 10
준비, 12
준비 완료 정의의 구조, 58
준비도 작업 항목, 32
준비성을 위한 여유 만들기, 28
지식 베이스, 8
책임성과 추적성, 14
초기 작업, 49
추가 작업, 18
출시, 49
컨설턴트, 7
코드 품질, 24
코드베이스, 24
- 큰 전환점, 12
터미네이터 스타일 반란, 19
테스터, 11
퍼즐 맞추기, 20
프로덕트 매니저, 5
프로덕트 오너, 5
프로덕트, 역할, 4
프로세스의 관문, 58
프로젝트 관리 협회, 2
프로젝트 기한, 64

플로우, 13
피드백 루프, 12
행동 방식: 요구사항의 상태를 가시적으로
추적하기, 67

행동: 준비 및 구현과 관련된 모든 작업
공개, 67
행위 주도 개발, iv, 7
확인된 공유 이해, 16