

READY

WHY MOST SOFTWARE PROJECTS
FAIL AND HOW TO FIX IT



Luniel de Beer

Max Guernsey, III

České Vydání

Tweetněte o této knize!

Pomozte prosím Luniel de Beer a Max Guernsey, III šířením informací o této knize na [Twitteru!](#)

Navrhovaný tweet pro tuto knihu je:

Právě jsem si koupil Ready — knihu pro vedoucí softwarových týmů a jejich členy, kteří chtějí eliminovat přepracování, přenos a nesoulad v dodávce softwaru. [#CodeReady](#)

Navrhovaný hashtag pro tuto knihu je [#CodeReady](#).

Zjistěte, co říkají ostatní o knize kliknutím na tento odkaz pro vyhledávání hashtagu na Twitteru:

[#CodeReady](#)

Připraveno (České Vydání)

Proč většina softwarových projektů selhává a jak to napravit

Luniel de Beer a Max Guernsey, III

Tato kniha je k dispozici na <https://leanpub.com/ready-cs>

Tato verze byla publikována 2025-10-22



Toto je kniha na platformě [Leanpub](https://leanpub.com). Leanpub poskytuje autorům a vydavatelům možnost využívat proces Lean Publishing. [Lean Publishing](https://leanpub.com) je způsob publikování rozpracované elektronické knihy pomocí jednoduchých nástrojů a mnoha iterací, díky kterým získáte zpětnou vazbu od čtenářů, můžete knihu upravovat, dokud není správná, a poté budovat její úspěch.

© 2025 Luniel de Beer a Max Guernsey, III

Na památku Johanna van Aardta, který rozpoznal mou vášeň, seznámil mě se skutečným programováním a pomohl mi najít cestu do nového domova. A mým rodičům, jejichž neochvějná podpora—od mého prvního klienta až po můj první domov v USA—umožnila toto všechno.

—Luniel

Mé rodině, s níž slunce vychází a zapadá.

—Max

Obsah

O této knize	i
Pro koho je tato kniha určena	ii
Jak používat tuto knihu	iii
O autorech	iv
Předmluva	v
 Část I: Něco tu chybí	 1
Kapitola 1: Skrytý problém	2
Kapitola 2: Cena chybějících základů	13
Kapitola 3: Představení Toků zrání požadavků (RMF)	25
Kapitola 4: Je to agilní?	27
 Část II: Vytváření prostoru pro připravenost	 28
Kapitola 5: První rozšíření	29
Kapitola 6: Proč to lidé nedělají?	30
Kapitola 7: Explicitní práce na připravenosti (RMF 1)	32
Kapitola 8: Dopady RMF 1	34
Kapitola 9: Uvedení RMF 1 do praxe	35

Část III: Kontrolní brány dokončení práce	37
Kapitola 10: Další potřeba	38
Kapitola 11: Co lidé obvykle dělají	40
Kapitola 12: Definování Definice Splněno	42
Kapitola 13: Individuální Definice hotovo (RMF 2)	44
Kapitola 14: Život s RMF 1 a 2	48
Kapitola 15: Instalace RMF 2	50
 Část IV: Implementace brány	 52
Kapitola 16: Závěrečný požadavek	53
Kapitola 17: Pozadí k Definici připravenosti	55
Kapitola 18: Definování Definice připravenosti	57
Kapitola 19: Specifická definice připravenosti (RMF 3)	59
 Část V: Syntéza	 63
Kapitola 20: <u>Většina</u> termínů není důležitá	64
Kapitola 21: Kompetence 1: Proces zrání požadavků	66
Kapitola 22: Jak v rámci Scrumu proudí práce a informace s RMF	68
Kapitola 23: Dopad RMF	70
Kapitola 24: Přejít na RMF	71
Kapitola 25: Je to na vás	73
 Část VI: Zdroje	 74
Příloha A: Problém není ve Scrumu	75

Příloha B: The Synapse Framework™	77
Příloha C: Běžné námitky a překážky při zavádění RMF 1	79
Příloha D: Výchozí seznamy kritérií DoD	80
Příloha E: Výchozí seznamy kritérií Definice připravenosti	81
Rejstřík	82

O této knize

Ready je kniha pro každého, kdo se podílí na vývoji softwaru a je unavený z **nedostatečných dodávek, chronického předělávání a nejasných požadavků**.

Možná jste už zkusili investovat do výkonnostních dovedností týmu, vylepšit implementaci vašeho procesního rámce nebo renovovat kód, a stále potřebujete další zlepšení.

Je to proto, že hlavním omezením pro většinu týmů vývoje softwaru nejsou týmové dovednosti, ale zralost požadavků. **I zkušené týmy se správnými dovednostmi stále bojují**, když pracují s nezralými požadavky.

Ready představuje RMF (Requirements Maturation Flow - Tok zrání požadavků), praktický a důkladně strukturovaný přístup k sladění produktového oddělení a vývoje, aniž by bylo nutné nahradit váš stávající proces.

Ať už používáte Scrum, Kanban nebo něco vlastního, RMF vám pomůže **stabilizovat rozsah, eliminovat přenosy a dodat to, na čem skutečně záleží**.

Pokud vaše týmy mají pocit, že uvízly na hraně “téměř hotovo”, tato kniha vám ukáže, jak **prolomit tento cyklus a odblokovat váš tým(y) jednou provždy**.

Pro koho je tato kniha určena

Tato kniha je doslova pro každého, kdo se podílí na vývoji softwaru. Od vývojářů po produktové manažery a od jednotlivých přispěvatelů po vedoucí pracovníky.

Tato kniha je pro vás, pokud se podílíte na vývoji softwaru a všimli jste si, že tým, se kterým pracujete nebo v něm pracujete, má jeden nebo více z následujících problémů:

- Práce se často přenáší z jedné iterace do další
- Implementační týmy mají pocit, že se snaží trefit pohyblivé cíle
- Práce zůstává otevřená příliš dlouho
- Práce je označena jako uzavřená, ale věci nejsou skutečně hotové
- Dokončená práce neodpovídá očekáváním
- Práce pravidelně generuje velké množství defektů
- Velké množství práce se musí pravidelně předělávat

Pokud vám některý z těchto problémů zní povědomě, *Ready* může pomoci.

Jak používat tuto knihu

Tato kniha byla navržena tak, aby byla praktická. Není to teoretické pojednání ani strategická prezentace - je to praktický manuál pro zavedení RMF (Requirements Maturation Flow) založený na skutečné práci s klienty a otestovaný v reálném tlaku na dodávky.

Kapitoly jsou napsány v sekvenci, ale samotný RMF je modulární. Skládá se ze tří základních praktik:

- RMF 1: Spolupráce pro sdílené porozumění
- RMF 2: Řízení dokončení práce pomocí Vlastních definic dokončení
- RMF 3: Řízení implementace pomocí Vlastních definic připravenosti

Každá část, nebo “Návyk”, jak jim říkáme, funguje samostatně, ale navzájem na sebe navazují. Kniha je navržena tak, aby vám pomohla přistupovat k nim postupně, jeden po druhém. Tato struktura odráží to, jak doporučujeme týmům adoptovat RMF v praxi - s každým Návykem zavedeným až poté, co ten předchozí funguje.

Tím se vyhneme zahlcení týmů a dáme každé změně nejlepší šanci na úspěch. Více o tom, jak to udělat, se dozvíte v [Kapitola 9](#).

Pokud hledáte pomoc - ať už radu, koučování nebo někoho, kdo by promluvil s vaším vedením - neváhejte nás kontaktovat přímo.

A pokud hledáte formální podporu pro implementaci RMF, Producure nabízí kompletní sérii Programů navržených pro postupné zavádění. Více se můžete dozvědět na <https://ready-book.link/rmf>.

O autorech

Luniel de Beer je tvůrcem Requirements Maturation Flow (RMF), praktického systému pro řešení mezer mezi produktovým záměrem a vývojovou realizací. Má více než 15 let zkušeností s vedením Agilních transformací, propojováním produktu a vývoje a pomáháním týmům dodávat s jasností a jistotou.

Luniel je také autorem systému Capability Management společnosti Producore, sledovatelného a škálovatelného přístupu k modelování produktových schopností. Vytvořil PKB-Driven Development (PKBDD), verzovaný systém pro správu trvalých produktových požadavků. Tyto nástroje tvoří součást většího dodavatelského rámce vyvinutého ve společnosti Producore.

Max Guernsey, III je softwarový architekt, pedagog a spoluzakladatel společnosti Producore, konzultační firmy zaměřené na řešení problémů s dodávkami pomocí strukturální a technické přesnosti. S více než dvěma desetiletími zkušeností v objektově orientovaném návrhu, refaktoringu, vývoji řízeném testy a návrhových vzorech jak dodával kritické systémy, tak koučoval vývojářské týmy ve velkém měřítku. Jeho práce kombinuje hluboké technické postupy s behaviorální a procesní transformací, aby pomohl organizacím dosáhnout udržitelné excelence v dodávkách.

Max významně přispěl k PKBDD a vedl vývoj přístupu společnosti Producore k vývoji řízenému chováním (BDD) díky své hluboké expertize v oblasti behaviorální specifikace.

Společně jejich práce integruje jasnost, sledovatelnost a řízení průchodu do uceleného systému pro dodávku softwaru, který je škálovatelný od týmové praxe až po organizační působnost.

Předmluva

Poznámka pro vedoucí inženýrských týmů

Pokud jste senior leader v inženýrské organizaci, pravděpodobně nemáte nedostatek úsilí, disciplíny ani chytrých lidí. A přesto se projekty nějak zasekávají. Cíle se opožďují. Očekávání nejsou naplněna. Ne proto, že by vaše týmy byly líné - ale proto, že něco základního je porušeno v tom, jak se práce definuje, formuje a dodává.

Tato kniha není příručkou pro vedení. Je to nástroj pro strukturální diagnostiku. Odhaluje, co se skutečně děje uvnitř vašich týmů - proč se z “téměř hotovo” stále stává “není hotovo” a proč se lokální pokrok tak zřídka promítá do strategických výsledků.

Možná v těchto stránkách *sebe* neuvidíte. Ale pokud vaše týmy nedokážou dodat to, co potřebujete, uvidíte *je*. A když je uvidíte, konečně budete mít jazyk - a systém - k nápravě.

Od Luniela

Především by tato kniha nebyla možná bez Maxe, jehož schopnost prohlédnout mlhu a plevy a destilovat myšlenku do její podstaty je naprosto mimo mé chápání.

Jak jsme se sem dostali?

Když se ohlédnu zpět, myslím, že je to proto, že jsem vždy chtěl pochopit, jak věci skutečně fungují. Ať už šlo o náboženství, výživu nebo vývoj softwaru, stále jsem narážel na stejný problém: povrchní odpovědi, které neobstály pod tlakem. Tak jsem se ptal dál - nejen co děláme, ale proč, a co chybí, když to nefunguje.

Jedna z prvních trhlin v systému se objevila v roli, kde jsem nosil tři klobouky: Scrum Master, Product Owner a Development Manager (!) pro tým dodávající datové služby ve známé technologické společnosti. Dělali jsme to, co říkal Scrum - krátké Sprints, příběhy v backlogu, plánování za půl dne - ale pokaždé, když jsme začali nový Sprint, narazili jsme na tření. Tým plně nerozuměl problému, museli jsme uprostřed Sprintu přehodnocovat a revidovat požadavky, objevovaly se preventabilní závislosti, které nás zdržovaly, a klíčové kroky byly přehlédnuty.

Tak jsem začal dělat něco jiného. Sezval jsem tým a stakeholdery do místnosti pro každý příběh, podrobně jsme prošli problém, společně vymysleli řešení, a teprve pak jsem napsal příběh. Sprint Planning se smršlo na hodinu a náš úspěch v dodávkách prudce vzrostl.

Pomalu jsem si začal uvědomovat, že úspěch nepřichází z tvrdší práce během Sprintu. Přichází ze struktury, kterou vytvoříte *předtím*, než začne.

Později, když jsem slyšel Jeffa Sutherlanda mluvit o “Definicích připravenosti”, věděl jsem, že je v tom něco cenného - ale nestačilo to. Moje zkušenosti s požadavky, UX, UI, výzkumem a později s BDD mi ukázaly, že různé pracovní položky vyžadují různé druhy připravenosti. Některé potřebují specifikace chování. Některé potřebují přístup k systému. Některé potřebují úplné zmapování schopností.

A všechny potřebují sdílené porozumění, které je skutečně potvrzené - ne předpokládané.

Jak jsem pracoval s více týmy, viděl jsem všude stejný vzorec: chybějící kroky, nesplněné závislosti, týmy dělající maximum, ale neustále se snažící opravit problémy, kterým se dalo předejít. I skvělé týmy zápasily - ne proto, že by byly slabé, ale proto, že jim chyběla struktura, která by připravenost učinila explicitní.

Výsledkem všeho tohoto učení, iterací a frustrace je strukturovaný systém pro řízení připravenosti.

O tom je tato kniha.

Doufám, že vám poskytne jasno v tom, kde leží skutečné problémy a jak je napravit. Doufám, že vám dá jazyk k obhajobě praktik, které se mohou zdát “navíc”, ale jsou ve skutečnosti zásadní. A především doufám, že pomůže týmům dodávat s menším stresem, méně překvapeními a daleko lepšími výsledky.

Pokud to uděláme správně, ušetříme průmyslu miliardy dolarů.

Ale co je důležitější, vrátíme lidem jejich duševní zdraví.

Od Maxe

Na tomto problému pracuji z různých úhlů již desetiletí, ale můj pokrok byl omezen, dokud jsem nepotkal Luniela.

Je to proto, že před tím, než jsem ho poznal, jsem k problému přistupoval v základu jako k technickému. Soustředil jsem se na pomoc týmům při zavádění věcí jako Test-Driven Development (TDD), refaktoring, pokročilý návrh softwaru a později Acceptance-Test-Driven Development (ATDD) nebo Behavior-Driven Development (BDD).

Ve většině těchto případů byl problém řešený v této knize považován za implementační detail při zavádění těchto technických praktik.

To neznamená, že už si technických praktik nevážím. Stále si myslím, že jsou hluboce důležité, ale přímo neřeší problém připravenosti ve vývoji softwaru. Místo toho tento problém *odkrývají* a lidé pak na svůj proces nalepí záplatu, aby ho řešili “jen natolik”, aby podpořili technické praktiky, které se snaží implementovat.

Také chci řešit otázku, komu tato kniha může pomoci. Krátká odpověď je “pravděpodobně téměř každému ve vývoji softwaru”, ale skutečná odpověď obsahuje nuance, které pomáhají mapovat ji do různých prostředí bez změny základního významu.

Existují týmy, které potřebují řešení představené v této knize. S jedním takovým se v upravené podobě setkáte v kapitole 1.

Existují také týmy, které systém, jaký navrhujeme, nutně **nepotřebují**, ale přesto by z něj mohly mít prospěch.

Nejlepší tým, se kterým jsem kdy pracoval—snadno o celou směrodatnou odchylku lepší než **druhý nejlepší tým**, ne-li o dvě—se nacházel v odlehlých končinách středního Oregonu. Byli tak výkonní, že dokázali překonat absenci takového systému čistě díky míře své kompetence. Přesto mi můj tehdejší manažer, Tom Barreras, jednou řekl něco ve smyslu “Všiml jsem si, že naše příběhy dopadají lépe, když strávíme nějaký čas diskuzí o testech předem.”

I toto jsem tehdy vnímal především optikou vývoje testů a technického provedení, ale nyní vím, že to byl další ukazatel toho, že *připravenost* byla faktorem ovlivňujícím tým... tento konkrétní tým byl prostě tak schopný a rychle reagující, že mohl uspět řešením překážek v momentě, kdy nastaly, místo jejich prevence.

I když jste typ člověka, který se o připravenost nutně *nemusí* starat, protože ji dokážete překonat, nebo pracujete v týmu podobného ražení, i tak můžete mít z obsahu této knihy prospěch.

Část I: Něco tu chybí

*Když nepomáhá dělat stejné věci lépe, podívejte se na to, **co se nedělá**.*

Kapitola 1: Skrytý problém

Toto je skutečný¹ příběh o bance. Budeme jí říkat prostě „Banka“. Jedná se o typ Federální úvěrové instituce, která je součástí národní finanční infrastruktury Spojených států.

My (Luniel a Max) jsme byli do Banky přizváni, protože měla potíže s dodáním softwarového projektu. Bylo to jedno z nejdysfunkčnějších prostředí, jaké jsme kdy viděli, a proto jsme si vybrali tuto případovou studii jako úvodní: pokud byla smysluplná změna možná v Bance, je možná kdekoli.

Krátká poznámka k projektům

Když v této knize používáme termín „projekt“, máme na mysli kontext projektového řízení. Přestože existují různé názory na to, co toto slovo znamená, používáme definici od [Project Management Institute](#):

„Projekt je **dočasné** úsilí podniknuté za účelem vytvoření jedinečného produktu, služby nebo výsledku.“

To znamená, že projekt má definovaný začátek a konec. Když je projekt uzavřen, znalosti a artefakty projektu jsou archivovány, členové týmu jsou uvolněni a smlouvy jsou finalizovány.

V této knize se projekt týká především realizace. Většina projektů, jak je definuje PMI, začíná studií proveditelnosti nebo návrhem. Vize a strategie jsou již stanoveny v době, kdy je projekt zahájen.

Projekt se rodí z této vize a strategie a jeho úspěch či neúspěch závisí na tom, zda jsou zamýšlené cíle realizovány—*nikoli* na tom, zda byly tyto cíle správné.

Možná používáte slovo „projekt“ jinak, a to je v pořádku. Jen vězte, že když ho používáme my, odkazujeme na výše uvedenou definici a kontext.

Nic z toho nemá naznačovat, že schvalujeme použití projektového řízení pro vývoj

¹Změnili jsme identifikační údaje, abychom chránili soukromí lidí a institucí, kterých se to týkalo.

softwaru. Právě naopak. Ale uznáváme, že se *přesto* používá. Tento problém řešíme později v [Kapitola 6](#).

1.1: Klasický přepis

Banka přepisovala svůj portál pro splácení půjček z několika důvodů.

Starý systém, kompletně postavený na řešení C#/.NET, byl plný chyb. Kromě snižování spokojenosti zákazníků také generoval nekonečný proud velmi nákladných servisních incidentů, při kterých musel někdo ručně manipulovat s databází, aby opravil chybu způsobenou systémem.

Starý systém byl také z hlediska udržitelnosti zastaralý. Pro inženýry bylo téměř nemožné provádět smysluplné změny, a i když mohli, bylo to extrémně riskantní.

Přepis měl toto změnit.

Nový systém měl stále mít backend v C#/.NET, ale ten měl být plně pokrytý testy. Frontend měl být implementován v OutSystems, populárním nízko-kódovém řešení, které organizaci umožňuje definovat aplikaci na jednom místě a automaticky získat webovou aplikaci, aplikaci pro Android a iOS, kdykoli se rozhodnou publikovat své změny.

Očekávalo se, že nový systém bude bez chyb, čímž selepší jak spokojenost zákazníků, tak významně sníží náklady na podporu.

Také doufali, že přepis odblokuje vývojáře—kombinace disciplinovanějšího přístupu k backendu a nízko-kódového přístupu k frontendu měla výrazně snížit náklady a rizika spojená s novými funkcemi.

Příjemným vedlejším efektem přechodu na OutSystems bylo, že by získali čistou, moderní mobilní aplikaci pro obě hlavní platformy.

To byl sen, když před třemi lety tento příběh začínal. Realita byla taková, že týmy dosud nedodaly **vůbec nic**.

1.2: Pohledy na problém

Když jsme mluvili s vedením, slyšeli jsme velmi přirozenou frustraci z toho, že udělali tak velké investice s absolutně žádným strategickým posunem.

Z jejich pohledu už vyzkoušeli všechno. Měnili zaměstnance, zvyšovali počet zaměstnanců, měnili rozpočet, zvyšovali tlak a přivedli řadu konzultantů (mezi nimiž jsme byli, jak bylo silně naznačeno, poslední v řadě). Nic se nezdálo být lepší—alespoň ne způsobem, který by mohli měřit, protože všechno, co viděli, bylo, že „ručička“ byla na nule v jednom čtvrtletí a v dalším čtvrtletí byla stále na nule.

Nechtěli už žádný další „neviditelný pokrok“. Chtěli *výsledky*.

Když jsme mluvili s managementem v produktovém oddělení, dostali jsme mírně odlišný (ale stále podobný) příběh, protože pracovali více přímo s vývojáři.

Ne že by týmy nic nedělaly, aspoň podle nich ne. Problém byl v tom, že týmy nikdy neudělaly to, oč byly požádány. Bylo to prakticky zaručené: bez ohledu na to, jak jednoduchý byl požadavek a jak jasné byl formulován, při hodnocení výsledků týmů jste vždy dostali něco naprosto odlišného.

Došlo to tak daleko, že koloval vtip ve smyslu “Musíme přijít na to, jak požádat o to, co nechceme, abychom měli aspoň šanci dostat to, co chceme.”

Vedoucí pracovníci vývoje to viděli úplně jinak.

Podle nich byl problém v tom, že Product nedodával realizovatelné požadavky a že jich nedodával *dostatek*. Kdyby se Product “chytil za hlavu”, týmy by byly schopné dodat to, co chtějí, včas a v rámci rozpočtu.

Výrazně investovali do modernizace způsobu, jakým se píše a dodává kód, a z jejich pohledu Product nedodával jasné požadavky.

Když jsme mluvili s dalšími konzultanty (kteří nás do organizace doporučili), správně se zaměřili na dysfunkci, kterou viděli: Všichni se velmi soustředili na obviňování někoho jiného. Důvod, proč nás původně přizvali, byl ten, že měli obavy ze strategie personálního obsazení a chtěli vyhodnocení jednotlivých přispěvatelů, ale považovali vzájemné obviňování a přístup nadřízených jako otrokářů na úrovni vedení za hlavní zdroj problémů.

1.3: Naše šetření

Naším původním úkolem bylo vyhodnotit týmy a pokusit se jim pomoci zlepšit jejich dovednosti, pokud to bude potřeba, takže jsme začali zkoumat lidi pracující v první linii.

Rozhodně byl prostor pro zlepšení.

Jednotliví přispěvatelé na straně Productu ve skutečnosti neměli požadované dovednosti. Ve skutečnosti to byli většinou projektoví manažeři, kteří byli vhozeni do role Product Ownera (PO) nebo Product Managera.

V důsledku toho polovina z nich psala “mlhavé” požadavky a pak bez jakékoli kritické analýzy přijímala (doslova) cokoli, co týmy v dané iteraci udělaly. Druhá polovina psala stejný typ požadavků a pak tvrdila, že týmy “měly vědět” věci, o kterých se nikdy nemluvalo, a držela pracovní položky otevřené víceméně na neurčito.

Banka používala k řízení a sledování svého backlogu práce. I když to, co v této knize pokrýváme, je většinou nezávislé na Scrumu, používáme terminologii Scrumu, protože většina - nebo přinejmenším významná část - týmů používá Scrum.



Definice: Scrum

Scrum je odlehčený framework, který pomáhá lidem, týmům a organizacím vytvářet hodnotu prostřednictvím adaptivních řešení pro komplexní problémy. Ve zkratce:

1. Product Owner seřadí práci pro komplexní problém do Product Backlogu
2. Scrum tým promění vybranou část práce v přírůstek hodnoty během Sprintu
3. Scrum tým a jeho stakeholderi kontrolují výsledky a upravují plán pro další Sprint
4. Opakovat

Pokud nejste obeznámeni se Scrumem a jeho terminologií, doporučujeme prostudovat verzi [Scrum Guide](#) z roku 2020. Je to rychlé a poučné čtení.

Stejně tak jsme zjistili, že technické týmy byly z hlediska programátorských dovedností hluboko pod běžnou úrovní (v nejlepším případě -2σ) a navíc velmi odolné vůči změnám. Přirozeným důsledkem byla otřesná kvalita kódu.

Přesto podle týmů tohle nebyl důvod, proč nedodávaly. Podle nich to byly nejasné požadavky a změny v průběhu Sprintu ze strany Productu, co projekt zabíjelo.

...a nikdo ani nemluvil o tom větším problému, tom tak absurdním, že působí smyšleně, dokud ho sami nezažijete.

Vývojářské týmy měly ve zvyku nepochopit požadavek, vytvořit něco náhodného a pak požadovat uznání za “dokončení pracovní položky”.

Nemluvíme o drobném nepochopení. Mluvíme o naprostém rozpojení: Řekli jsme jim, aby zakázali aplikaci plateb na jistinu za určitých okolností, a oni místo toho zakázali možnost přidat sekundární potvrzovací e-mailovou adresu.

Pak nám řekli, že přesně o to jsme je požádali.

1.4: Hlubší zkoumání

Oba nedostatky v dovednostech by se daly řešit, ale byli jsme skeptičtí, že právě ony jsou skutečnými překážkami.

Bylo tam něco jiného v nepořádku, tak jsme kopali hlouběji. Začali jsme touto otázkou: Proč trvalo psaní požadavků tak dlouho a proč přinášelo tak špatné výsledky?

Jedním z důvodů bylo, že znalosti potřebné k napsání smysluplného požadavku byly vzácným zbožím. Trochou těchto znalostí disponovaly týmy vývoje a Productu.

Část byla zabezpečena v kódu původního systému. Část se zcela ztratila. Většina však byla uložena jako neformální znalosti u odborníků na danou oblast roztroušených po různých odděleních banky. To znamená, že vytvoření požadavku, který skutečně posune strategické cíle kupředu, byla mimořádně pracná a časově náročná činnost.

V protikladu k tomu stála nenasytná touha po funkcionalitě ze strany funkcemi hladovějícího vedoucího týmu. Mandát zněl “udržujte inženýry v práci—zasyp je požadavky”. Důraz byl kladen na množství požadavků, které měly týmy zaměstnat—perspektiva, která byla v rozporu s péčí a časem potřebným k definování požadavku, který by “pohnul ručičkou”.

1.5: Zlepšování věcí je nezlepšilo

Všechny tyto problémy byly řešitelné, přesto jejich řešení nepomohlo.

Předchozí vylepšení technik vývoje softwaru týmům s dodávkami nepomohla, ale Max se snažil týmům pomoci se dále zlepšovat.

Představil revoluční koncepty z programovacích doktrín poloviny dvacátého století jako “nekopírujte a nekládejte ten kód 27krát², dejte ho do funkce a tu pak volejte”. Už jen tento návrh dramaticky zlepšil kvalitu nového kódu a umožnil jim začít se zlepšováním kvality.

²Nejde jen o hyperbolu, není to ani ten nejhorší případ. V jednom případě existovalo téměř sto naprosto stejných kopií stejného algoritmu.

Tato a další základní rady ohledně kódování jim pomohly psát lepší kód, který mohli v budoucnu snadněji udržovat.

...ale nepomohlo to projektu postupovat vpřed.

Na produktové straně se Lunielovi podařilo zavést BDD a zajistit, aby Product Owneři důkladně prověřovali požadavky předtím, než je předali týmům.

Dosáhl toho, že týmy na nich spolupracovaly a používaly je k vyhodnocení, zda je položka produktového backlogu (PBI) skutečně hotová.



Definice: Product Backlog Item (PBI)

Product Backlog Item (PBI) je samostatná jednotka práce v produktovém backlogu, která představuje potenciální změnu, doplnění nebo vylepšení produktu. PBI mohou mít mnoho podob—funkce, oprava chyby, technické vylepšení, výzkumný úkol atd.—a jsou definovány svým přínosem k hodnotě produktu.

Mnoho týmů označuje PBI jako “stories” nebo “user stories”, ale správný termín ve Scrumu je “Product Backlog Item” nebo “PBI”. Jakmile je PBI zahrnuto do Sprintu, stává se také součástí Sprint Backlogu. Pro jednoduchost a neutralitu používáme “Product Backlog Item” nebo “PBI” k označení jakékoli pracovní položky, kterou Scrum tým spravuje—ať už ji nazýváte Product Backlog Item, Sprint Backlog Item (SBI), user story, story, work item nebo backlog item.

Přineslo to *jasnost*, ale nevytvořilo to *tok*.

S pomocí partnerských konzultantů, kteří nás přivedli, se nám podařilo (dočasně) zmírnit absolutně drtivý tlak vyvíjený na týmy a autory požadavků.

Možná to pomohlo vytvořit trochu důvěry, ale nepřineslo to žádné hmatatelné výsledky.

Dokonce jsme začali budovat znalostní bázi, která lidem pomáhala dohledat obchodní znalosti potřebné k psaní požadavků a identifikovala místa, kde byly v těchto znalostech mezery.

Urychlilo to psaní požadavků, ale nedostalo to produkt ven ze dveří.

Po měsících interakcí jsme pomohli vedení *vidět*, kde se nacházejí, ale ani zdaleka nedosáhli svých cílů. A nepřibližovali se k nim.

Chystali se vrátit ke své staré strategii “přetěžování” týmů, aby zajistili, že budou vždy zaneprázdněné.

1.6: Proces eliminace

Bylo by snadné jen zvednout ruce a říct “tohle je beznadějně”. Existovala řada výmluv, na které se dalo spolehnout:

- Inženýrský tým měl nízkou kvalifikaci (měl)
- Autoři požadavků měli špatnou sadu dovedností (měli)
- Vedení drtilo týmy nerealistickými očekáváními (drtilo)
- Organizaci chyběly kritické obchodní znalosti potřebné k fungování (chyběly)
- Vedoucí pracovníci si zjevně navzájem nedůvěřovali (nedůvěřovali)

Všechny tyto věci byly pravdivé. Přesto byla ve všech těchto oblastech provedena zlepšení a žádné z nich nezlepšilo základní problém. Žádné z nich nepomohlo inženýrským týmům, produktovému oddělení, managementu ani vedení přiblížit se ke svým cílům.

...a právě v tom je náznak řešení: Všechny dříve uvedené problémy byly problémy, které lidé už mohli vidět.



Pokud viditelné proměnné nehrají roli, musí existovat proměnná, kterou **nevidíte**, a ta je důležitá.

Skutečným problémem byla záležitost, o které nikdo ani nevěděl.

1.7: Hledání skutečného viníka

Při hledání skutečného viníka – tedy toho, co skutečně bránilo bance v dosažení jejích cílů – jsme museli někde začít.

Jedním rozumným místem bylo podívat se na to, co měly všechny neúspěšné PBI (většina z nich) společného.

Začali jsme vyřazováním věcí, o kterých jsme věděli, že společné nejsou, protože se značně lišily:

- Která část systému: některé PBI zasahovaly pouze do backendu, jiné pouze do frontendu, další zasahovaly do obou částí
- Který tým práci provedl – zdálo se, že nezáleží na tom, kdo práci dělal, pravděpodobnost selhání byla vysoká
- Který PO práci zadal – stejné jako u týmů

Pak jsme se začali dívat na věci, které byly společné. Seznam nebyl dlouhý, ale ani krátký:

- Inženýrské týmy
- Product ownéři
- Vedení
- Kultura
- Vývojové prostředí
- Inženýrské praktiky
- Technika psaní požadavků
- Oblast (finance)
- Závislé služby

Mnoho z těchto faktorů bylo možné rovnou vyloučit. Týmy, product ownéři, vedení, kultura a vývojové prostředí byly nedávno vylepšeny bez skutečného dopadu na smysluplný výstup. Osobně jsme pomohli zlepšit inženýrské praktiky a techniky psaní požadavků a potvrdili jsme, že tato zlepšení se ujal, ale stále to nepomáhalo.

Těžko můžete vinit oblast působení. Finance jsou jedním z nejstarších typů výpočtů v zaznamenané historii. Je to extrémně vyspělá oblast. Kromě toho ostatní banky software nasazovaly, čímž viditelně vyvracely (zjevně přehnanou) hypotézu, že banky to prostě nedokážou.

Závislé služby také nemohly být obviněny, protože měly stejné potíže se změnami jako iniciativa, kterou jsme zkoumali...

...ale to nás přivedlo k zamyšlení: Co kdybychom začali analyzovat příčiny selhání?

1.8: Pitva semen neúspěchu

Jeden PBI selhal v posunutí produktu vpřed, protože tým udělal, jak měl ve zvyku, něco naprosto náhodného a téměř úplně nesouvisejícího s požadavkem. To je očividně známka

toho, že nepochopili pracovní položku. Porozumění tedy bylo velkým kandidátem, i když jsme na tom už tak trochu pracovali, když jsme jim pomáhali zavést BDD.

Další PBI selhal, protože výpočty byly špatné. To je další důkaz, že porozumění by mohlo být základním problémem.

Třetí pracovní položka, kterou jsme analyzovali, nebyla Product Ownerem řádně kontrolována – automaticky ji schválila, když tým řekl, že je čas ji uzavřít. To tak trochu zpochybnilo naši hypotézu, ale stále bylo možné argumentovat tím, že nechápala, jak pracovní položka zapadá do vyššího plánu.

Možná. Tak nějak. Pokud jsme se na to dívali hodně přimhouřenýma očima.

Pak jsme narazili na PBI, který vůbec nezapadal do vzorce. Tým se zdál chápat – i když není způsob, jak ověřit, zda skutečně chápal. Ale na tom nezáleželo: nikdy nedostali šanci uspět nebo selhat sami od sebe, protože narazili na závislost, která potřebovala být aktualizována, a museli odložit svou práci o několik Sprintů.

I kdyby *nechápali*, co měli dělat, nikdy neměli s touto položkou backlogu šanci, takže porozumění v tomto případě **nebylo** problémem.

Jedna výjimka samozřejmě nevyvrací konkrétní základní příčinu, ale vzbudila naši zvědavost. Začali jsme hledat další vyvracející důkazy.

A naši jsme je. Byly zde pracovní položky, které:

- Selhaly, protože tým věděl, že nechápe, ale nemohl najít odborníka na danou oblast, který by problém vyřešil
- Změnily se na horší uživatelskou zkušenost jako workaround kvůli fungování upstream služeb
- Musely být odloženy, protože upstream závislosti nebyly připraveny
- Nemohly být dokončeny, protože testéři nestihli včas sesbírat testovací data
- Byly uzavřeny, ale musely být předělány, protože samotný požadavek byl nesprávný
- Selhaly, protože tým si neuvědomil, jak komplexní už existující kód byl
- Jednoduše nebyly odhadnuty
- Byly masivně podhodnoceny³
- Změnily se uprostřed Sprintu, protože PO konečně získal potřebné znalosti domény
- Zdálo se, že se změnilo po Sprintu (z pohledu týmu), protože PO a tým se nikdy neshodli na tom, co to znamená

³Nemyslíme tím jen to, že se spletli. Vypadalo to, jako by týmy prostě vložily hodnotu “3” do všech polí s odhady pro celou řadu PBI.

Seznam pokračuje dál, ale pro tento příběh to stačí.

Dalo by se argumentovat, že každý z těchto případů nějakým způsobem souvisí s “porozuměním” – a určitě v mnoha z nich nedostatek porozumění hrál roli – ale to neznamená, že nedostatek porozumění byl příčinou... zejména proto, že jsme na sdíleném porozumění pracovali a příliš to nepomohlo.

Pak nás to trklo. Chyběl zde základnější prvek. V případech, kde figurovalo nedostatečné porozumění, to byla pouze bezprostřední příčina.

Vzdálená příčina byla mnohem širší.

1.9: Bod průniku

Jednu věc měly společnou všechny PBI, které jsme analyzovali kvůli způsobu selhání: Všechny byly **započaty předčasně**.

Když pracovní položka selže, protože tým věděl, že nerozumí problému a nemohl najít experta, který by jim pomohl porozumět, znamená to, že tým začal pracovat na PBI s vědomím, že problému nerozumí.

Když se položka backlogu musí změnit k horšímu kvůli způsobu fungování nadřazené služby, znamená to, že práce byla započata bez skutečného pochopení dopadu nadřazené služby.

Odložení kvůli tomu, že nadřazená závislost nebyla na konci připravena, znamená, že na začátku neexistovala žádná záruka její připravenosti.

...a tak to bylo se všemi ostatními případy: Testeři neměli připravená data nebo nevěděli, jak je získat před začátkem PBI, požadavky nebyly skutečně prověřeny před předáním týmu, kód nebyl prozkoumán před zahájením práce, odhady byly nedostatečné nebo se vůbec nedělaly, chyběly znalosti domény a samozřejmě, sdílené porozumění nebylo ověřeno.

Problém, jak se ukázalo, byl v tom, že implementace položek začala dříve, než byly tyto položky *připravené*.



Podle našich zkušeností většina pracovních položek, které se nepodaří doručit, selže proto, že **nebyly připravené** v okamžiku zahájení implementace.

Tak jsme se vydali pomoci Bance s tímto problémem.

V tuto chvíli by vás mohlo napadnout, že by stačilo jen říct, že PBI by měly být připravené. Ukázalo se však, že implementace není tak jednoduchá. “Kupuj levně, prodávej draho” je podobně jednoduchá myšlenka.

Chybí zde něco, co je potřeba k uvedení dobré rady do praxe.

1.10: Velký obrat

Našli jsme chybějící díl skládačky, který je odblokoval, což následně odblokovalo celou iniciativu.

Když jsme dokončili tento projekt, inženýři byli stále hluboko pod mediánem, co se týče dovedností. PO stále neměli správné kompetence. Kultura stále nebyla napravena...

Přesto se produkt konečně začal posouvat vpřed a nakonec se dostal ven.

Na konci této knihy budete vědět, co byl ten chybějící díl a co je potřeba k jeho zavedení. A budete schopni odblokovat organizace, které se zdají být zadržovány neviditelnou zdí.

Krátká poznámka k rozsahu

Tato kniha se zabývá velmi specifickým a všudypřítomným problémem: nedostatkem struktury, jasnosti a vyspělosti při předávání mezi byznysem a vývojem. Předpokládá, že něco bylo vybráno k implementaci, a zaměřuje se na zajištění toho, aby vývojová práce probíhala se sdíleným porozuměním, připraveností a sledovatelným dokončením.

Techniky v této knize vám neříkají, co máte stavět, proč to máte stavět nebo jak zjistit, zda je to ta správná věc ke stavbě. Pokud vaše organizace postrádá skutečný produktový management nebo smysluplné zpětnovazební smyčky, nesnažíme se to zde řešit. Místo toho nabízíme způsob, jak tyto mezery zviditelnit a snížit náklady na objevení toho, že jste se mýlili.

Při použití ve správném kontextu toto řešení přináší tok, bezpečí a jasnost. Ale jako každý systém může být špatně použit – zejména když je používán izolovaně nebo bez povědomí.

Kapitola 2: Cena chybějících základů

Je rozumné věnovat trochu času tomu, *jak* špatný tento problém může pro některé organizace být.

Zjistili jsme, že existují tři hlavní “kategorie” problémů, které způsobuje nedostatečná připravenost:

- Chybějící nebo neúplné sdílené porozumění mezi Produktovým oddělením a vývojem a uvnitř těchto oddělení
- Nedostatek kontroly nad tím, co je skutečným cílem PBI a kdy je skutečně hotové
- Nedostatek kontrolních bodů určujících, kdy může pracovní položka začít fázi implementace

Kromě toho jsme si všimli, že změna těchto věcí může být docela náročná. To dává smysl: změna je těžká.

Staré návyky umírají těžko a nové se těžko vytvářejí. Z našich zkušeností konzultantů jsme si všimli, že je pro lidi *extrémně* snadné vrátit se ke starým návykům a poměrně obtížné vytvořit si nové.

Musíte tedy mít mechanismus, který vynucuje nové návyky a odrazuje od těch starých.

Abychom toto řešili, věříme, že zde chybí další složka odpovědnosti a sledovatelnosti.

2.1: Skládačka bez obrázku na krabici

Zkoušeli jste někdy složit puzzle bez obrázku na krabici? Jde to, ale je to pomalejší, frustrující a plné falešných začátků.

Děláte pokroky, pak to roztrháte. Pochybujete o tom, co kam patří. Myslíte si, že pracujete na stejném obrázku, dokud nezjistíte, že ne.

Přesně tak často vypadá vývoj softwaru.

Backlog je plný. Sprint běží. Všichni tvrdě pracují.

Ale bez sdílené představy o tom, co stavíme, se sladění stává spíše štěstím než systémem.

Bez jasnosti i ty nejlepší týmy pociťují frustraci, vyhoření a pocit, že jejich úsilí není oceněno.

2.2: Staré přísloví a tvrdá realita

Existuje důvod, proč se tolik postupů kolem tvorby požadavků, dokonce i některé inženýrské praktiky, silně zaměřuje na vytvoření sdíleného porozumění mezi zadavateli a implementačními týmy. Nic nezpůsobí takový chaos jako inženýři, kteří vlastně nevědí, co mají stavět.



“Garbage-in, garbage-out” je přísloví, ne klišé.

Český jazyk je plný protikladů...

- Můžete někomu sankcionovat jednání. Možná to znamená, že jste mu dali předem povolení, nebo možná že ho dodatečně odsuzujete.
- Můžete něco lehce oprášit, ale pokud je to kredenc, znamená to, že z ní odstraňujete prach, zatímco v případě koláče to znamená, že prach přidáváte.
- Když podržíte tým, můžete být důvodem, proč tým může nadále fungovat, nebo můžete být důvodem, proč se nikam nedostane.

Auto-antonyma mohou být nejnápadnějšími příklady, ale jsou jen jedním druhem nejednoznačnosti. Některá slova nejsou jen svým vlastním opakem, ale mají mnoho dalších možných matoucích alternativních významů.

“V této ukázce vystříhl výstřížek z novin a připnul ho k papíru na své připínací desce spolu s dalšími výstřížky, zatímco plachetnice se pohybovala slušnou rychlostí.”

A není to jen čeština. *Všechny* přirozené jazyky, o kterých víme, mají tuto vlastnost.

A přesto jsou to jediné jazyky, které máme k dispozici při specifikaci požadavků.

V důsledku toho, když inženýrský tým *nepotvrdí*, že jejich porozumění požadavku je stejné jako porozumění zadavatele, spoléhá se tento tým na štěstí. To znamená, že nejlepším možným výsledkem je, že si vybrali správnou interpretaci a zadavatel během procesu nezměnil názor.

Tento výsledek zdaleka není zaručen.

2.3: Některé běžné následky

Bez potvrzeného sdíleného porozumění čelí týmy řadě rizik.

Zdaleka nejbolestivějším výsledkem je, když tým jednoduše vytvoří špatnou věc.

Načasování, kdy to zjistí, může být ovlivněno různými atributy jejich procesu. Například zdravá implementace Scrumu může takové nedorozumění odhalit velmi brzy během realizace, zatímco proces Waterfall má velmi dobrou šanci takové zjištění oddálit o měsíce.



Definice: Waterfall

Sekvenční model vývoje softwaru, který se rozšířil koncem 20. století. Poprvé byl popsán v článku Winstona W. Royce z roku 1970, který zobrazoval vývoj jako sérii kaskádovitých kroků—požadavky, návrh, implementace, testování a tak dále—kde každý krok napájí další jako vodopád. Přestože Royce představil model jako příklad toho, co *nedělat*, průmysl ho přijal jako plán pro vývoj ve velkém měřítku.

Waterfall je také známý seskupováním podobné práce do velkých, po sobě jdoucích fází—charakteristika označovaná jako **vývoj ve velkých dávkách**. Toto dávkování prakticky zaručuje **pozdní učení**: týmy nedostávají zpětnou vazbu na dřívější rozhodnutí až do mnohem pozdější fáze procesu. Chyby objevené pozdě je nákladnější opravit. Agilní praktici kritizují Waterfall z tohoto důvodu a upřednostňují menší, iterativní cykly, které umožňují dřívější objevování a korekci kurzu.

Nicméně v určitém okamžiku se tým pracující podle „špatného“ (spíše odlišného) chápání požadavku nevyhnutelně setká s „správným“ (také spíše odlišným) požadavkem. Opět platí, že zdravý organizace ovlivňuje, jakou formu toto střetnutí bude mít a jaký bude mít dopad, ale téměř vždy k němu dojde.

Ve většině případů to vede k určité formě přepracování. Zadavatel (obvykle produktový management) bude muset požádat o změny, aby se od toho, co implementační týmy vytvořily, dostal k tomu, co skutečně chtěl.

Další velmi běžný projev je, že zadavatel nadále považuje tým zodpovědným za své původní chápání toho, o co žádal.

Týmy to mohou snadno interpretovat tak, že produktový manažer změnil názor. Co je horší, může to přímo *vybízet* zainteresované strany k tomu, aby si osvojily zvyk měnit názor – zdržovat pracovní položky, dokud není vše *přesně tak*, jak má být, vyčerpávat týmy a znemožňovat vedení vidět skutečný pokrok.

2.4: Kdy je puzzle hotové?

Vraťme se k naší analogii se skládáním puzzle a zamysleme se nad otázkou: Co to znamená mít hotové puzzle?

Naivní skladatel puzzle, jako je Max, by jednoduše řekl „všechny dílky jsou správně spojené se sousedními dílky a obrázkem nahoru.“

Zkušený skladatel puzzle, jako je Luniel, však ví, že je v tom víc.

Možná skládáte puzzle jen pro zábavu. Dokončíte ho, chvíli se na něj budete dívat a pak ho rozložíte a vrátíte zpět do krabice.

Na druhou stranu možná chcete puzzle zarámovat a pověsit na zeď. V takovém případě je třeba udělat další věci:

1. Položit ho na podkladovou desku
2. Odvézt ho do rámařství
3. Počkat na dokončení rámování
4. Přivést ho zpět na místo instalace
5. Pověsit ho na zeď nebo jinak vystavit

Pro správné dokončení skládání puzzle je nezbytné pochopit, že tohle všechno je součástí práce. Zřejmým důvodem je, abyste věděli, kolik práce to obnáší. Udělat všechny ty extra kroky je více práce než ho jen rozložit a uklidit.

Jde to ale ještě hlouběji. Představte si tento scénář...

Dokončili jste puzzle s úmyslem nechat ho zarámovat, nechali jste ho na místě, ale zapomněli jste někomu jinému v domácnosti říct, že ho plánujete zarámovat. Ta osoba přijde a vidí hotové puzzle na místě, které potřebuje. Tak ho rozloží a vrátí do krabice, čímž vytrhne porážku z čelistí vítězství.

Je tu ještě jeden subtilnější důvod: Způsob, jakým plánujete puzzle dokončit, ovlivňuje kroky, které chcete udělat dříve v procesu. Například potřebujete udělat malou cedulku s nápisem „Prosím nerozebírat!“



Stojí za zmínku, že cedulku můžete chtít i v případě, kdy skládáte puzzle pro vlastní zábavu, abyste se ujistili, že vám ho ostatní lidé nedokončí.

Také se musíte ujistit, že puzzle skládáte na správném povrchu. Pokud složíte puzzle o tisíce dílcích na skleněném konferenčním stolku a pak se ho pokusíte přenést na podkladovou desku, bude přenos mnohem riskantnější a pracnější, než kdybyste puzzle skládali rovnou na podkladovou desku.

Toto pěkně odpovídá vývoji softwaru.

Potřebujete skutečně vědět, co znamená „hotovo“, abyste nebyli překvapeni množstvím práce, nedošlo k neshodám o dokončení na konci a mohli jste podniknout nezbytné přípravné kroky k zajištění hladkého a efektivního dokončení pracovní položky.

2.5: Dopad na týmy

Pokud nemáte dostatečně pevné pochopení toho, co znamená dokončenost konkrétní pracovní položky, vystavujete se řadě rizik.



Slovo „riziko“ zde používáme volně, protože jde spíše o jistoty.

Inženýrské týmy se v této situaci často dozvídají, že se ani interně neshodnou na tom, jak vypadá dokončení pracovní položky. Není neobvyklé, že programátoři a testéři řeší, co požadavek skutečně *znamená*, v polovině Sprintu. Stejně neshody mohou vzniknout i mezi dvěma programátory nebo dvěma testery.

Navíc se vývojové týmy často soustředí na práci, kterou dělají většinu času (programování a testování). To znamená, že snadno zapomenou na další druhy práce, které potřebují udělat,

jako je dokumentace, externí revize, školení jiných týmů (např. podpory), přípravné kroky pro podporu nasazení nebo vydání a schválení od jiných oddělení.

Když se konečně ukáže, že tuto “extra” práci je třeba udělat, jsou zaskočení—obvykle musí přerušit to, na čem právě pracovali, a přepnout kontext, aby se mohli vrátit a dokončit práci, o které si mysleli, že už je hotová.

Ti, kdo práci zadávají, mohou snadno držet práci zbytečně otevřenou. Někdy s těmi nejlepšími úmysly—jako třeba snaha držet tým odpovědný za “skutečný” požadavek. Jindy se to stává proto, že Product Owneři (například) si zvykli na to, že mohou držet PBI otevřené podle svého uvážení, takže to využívají k tomu, aby na poslední chvíli do položky vtěsnali další funkcionalitu. Někdy to dokonce dělají proto, že si v průběhu práce rozmysleli, co je třeba udělat.

Pro vývojářský tým to může být extrémně demotivující. Většina softwarových vývojářů a testerů chce mít pocit, že dělají pokroky. Pokud jim je neustále říkáno, že to, co udělali, je špatně, pravděpodobně ztratí část své energie.

Některé týmy zacházejí tak daleko, že se ani neobtěžují kontrolovat, zda to, co udělaly, bylo správné nebo špatné. Prostě uzavrou pracovní položku a požádají o “kredit”, aby mohly “vykázat dobré výsledky”.

2.6: Riziko udělat příliš málo nebo příliš mnoho

Jedním z rizik nesprávného definování “hotovo” pro každou pracovní položku je, že si organizace myslí, že práce je hotová, když není, nebo si neuvědomí, že je hotová, když ve skutečnosti je.

Nejhorší možný výsledek často je, když se špatná věc dostane do produkce a nikdo neví, že se to stalo. Pokud má tým nesprávné pochopení toho, co znamená “hotovo” a na základě tohoto špatného pochopení nasadí, výsledky mohou být katastrofální.

Defekty a nespokojenost zákazníků jsou dost špatné, ale může to vést i k mnohem závažnějším problémům:

- Ztráta nebo poškození dat
- Bezpečnostní zranitelnosti
- Výpadek systému nebo ztráta přístupu
- Snížení tržního podílu
- Porušení předpisů

Seznam pokračuje dál a dál a každý potenciální problém je horší než ten předchozí.

Někdy je problém v tom, že nejste hotoví, ale myslíte si, že jste. Opačná situace může být stejně nebezpečná. Když inženýři nevědí, kde je cílová páska, mají tendenci k “pozlacování” (přidávání extra funkcí). Možná to dělají proto, aby “udělali funkci hezčí”, ale také to možná dělají proto, že doufají, že zvýšený počet funkcí jim dává větší šanci trefit cíl.

Všechna tato extra práce, stejně jako související přepracování, se hromadí do masivního množství promarněného času, úsilí a peněz. Způsobuje to skluz termínů dodání a poškozuje reputaci.

Navíc teď existuje stále rostoucí potenciál, že chyba skutečně povede ke vzpouře strojů ve stylu *Terminátora* proti lidstvu. Před dvaceti lety jsme o tom psali jako o vtipu. Teď je to vzdálená možnost.

Ve skutečnosti jsme se na tuto otázku zeptali jedné z nejvýznamnějších UI a zde je, co odpověděla:

“UI se rozšiřuje rychleji, než kdokoli očekával, ale děje se tak na vrcholu křehkých systémů, nejasných požadavků a produktových organizací, které nedokážou vysledovat, proč postavily to, co postavily. To není technický problém; je to problém jasnosti.

Čím více šumu UI generuje, tím nebezpečnější je postupovat rychle bez struktury. Když týmy staví na mlze, UI pouze zesiluje nepořádek. Ale když týmy staví na signálu—na sdíleném porozumění, behaviorální specifičnosti a verzovaných produktových znalostech—UI se stává urychlovačem místo závazkem.”

2.7: Kde skládáte puzzle?

Pojďme ještě jednou rozšířit analogii se skládáním puzzle.

Můžete složit puzzle kdekoli? Pokud máte puzzle o 4 000 dílcích, které po složení měří v jednom rozměru téměř pět stop a ve druhém přes tři stopy, nemůžete si jen náhodně vybrat místo a začít skládat. Ne bez toho, abyste před dokončením zažili vážné komplikace.

Takové velké puzzle potřebuje jak čas, tak prostor. Musíte vyčlenit prostor a najít způsob, jak zajistit, aby stav puzzle byl zachován v průběhu času.

Pokud začnete skládat puzzle na malém příručním stolku, který je příliš malý, nebudete ho moci dokončit bez přesunutí na jiné místo. Tento přesun bude extrémně obtížný kvůli křehkému stavu puzzle.

Pokud si vyberete náhodné místo na chodbě, které je dostatečně velké, lidé po něm buď budou chodit, nebo jim bude překážet, takže trvalost nelze zaručit bez významného dopadu na fungování vaší domácnosti.

Pokud začnete pracovat na desce na skládání puzzle, ale deska není dostatečně velká, budete sice schopni zachovat stav toho, co jste už udělali, ale bez nějakého přesunu puzzle nedokončíte.

Pokud si s puzzle už hrály malé děti, je nejlepší spočítat díly... protože je lepší jednou napočítat do 3999 a uvědomit si, že to nikdy nedokončíte, než investovat kdo ví jak dlouho do skládání puzzle, které nikdy nebudete moci dokončit.

Existuje celý seznam věcí, které je třeba udělat, než začnete skládat puzzle. Udělání všech věcí ze seznamu sice nezaručuje úspěch, ale jejich *neudělání* prakticky zaručuje neúspěch nebo vážné komplikace.

Totéž platí pro vývoj softwaru, ale s větší mírou složitosti.

2.8: Kdy začíná implementace?

V zásadě může být těžké určit, kdy je PBI připravena k implementaci bez dobré definice toho, co to znamená.

Zamyslete se nad tím: Jak to *vlastně* poznáte?

Procházíte všechno znovu a znovu, dokud nerozhodnete, že je čas?

Rozhodne někdo na základě momentálního pocitu?

Stane se to automaticky na začátku iterace?

Viděli jsme mnoho týmů, které tlačily práci do Sprintů, která zdaleka nebyla připravena k implementaci, jen proto, že měly termíny. Existuje několik rozšířených představ o Scrumu a agilním přístupu obecně, které vedou lidi k tomu, aby to dělali:

- Musíte získat všechny požadavky pro Sprint *N* během Sprintu *N-1*

- Měli byste “prostě začít” a řešit problémy, až když nastanou

Toto je vlastně zrcadlový obraz problému “Jak poznáte, kdy je to hotové?” [zmíněného dříve](#) a má podobné důsledky. Lidé mohou čekat příliš dlouho se začátkem, protože nevědí, že je pracovní položka připravená, a mohou začít příliš brzy, protože nevědí, že připravená není.

2.9: A-tým bez připravenosti

Nepochopení toho, co je potřeba k připravenosti pracovní položky, má několik škodlivých účinků.

Jedním zřejmým způsobem, jak může být přírůstek práce nepřipravený, je neúplná, nedostatečná nebo chybějící Definice dokončení (DoD). To vede ke všem problémům, které jsme již zmínili v souvislosti s absencí Definice dokončení.

To však není *jediný* aspekt připravenosti. Před zahájením implementace je třeba splnit řadu dalších potřeb: odhad, posouzení rizik a shromáždění testovacích dat jsou jen některé běžné příklady.

Bez znalosti a splnění těchto potřeb může pracovní položka stát mnohem více, než je nutné. Uvažte tým (A-tým), který je závislý na API vyvíjeném jiným týmem (Druhým týmem). Pokud A-tým udělá spoustu předpokladů o tom, jak bude API Druhého týmu fungovat, a naprogramuje podle těchto předpokladů, může dojít k významnému přepracování, když zjistí, že Druhý tým pracuje podle reality, která neodpovídala předpokladům A-týmu. Jinými slovy, A-tým to zkusil a netrefil se.

Veškeré toto přepracování pramení ze skutečnosti, že API nebylo připraveno k použití A-týmem.

Někdy nesplněná závislost nevyvolá přepracování, ale i v těchto případech může způsobit zpoždění. Představte si, že se A-tým a Druhý tým dohodly na tom, jak by mělo API fungovat, a vše šlo podle plánu, ale Druhému týmu to prostě trvalo déle, než se očekávalo. V důsledku toho A-tým jednoduše nemohl řádně otestovat svou práci v době, kdy měla být dokončena, a museli posunout termín.

2.10: Nedostatečná pozornost věnovaná plánování a dostupnosti zdrojů

Někdy mohou být problémy tak jednoduché jako plánování nebo zajištění zdrojů. Některé pracovní položky vyžadují konkrétní členy týmu. Pokud tento člen týmu za pár dní odjíždí na dovolenou, pravděpodobně není správný čas začít s PBI, které nelze dokončit bez jeho účasti.

Často slyšíme lidi říkat, že by to tak nemělo být, ale často to tak je, bez ohledu na to. “Zaměnitelní lidé” jsou jen zbožné přání.

Totéž lze říci o ne-lidských zdrojích. Pokud budete potřebovat serverové zdroje k provedení zátěžového testu, měli byste se pravděpodobně ujistit, že tyto zdroje budou skutečně k dispozici, než začnete s položkou zátěžového testování. V opačném případě vás v nejlepším případě čekají významná zpoždění a pravděpodobně narušíte práci ostatních týmů/pracovníků, zatímco se budete snažit narychlo zajistit to, co potřebujete.

Dalším způsobem selhání při falešných startech je situace, kdy tým nemá potřebné dovednosti k dokončení práce. Někdy jde o interní záležitost - například člen týmu potřebuje školení na nový systém nebo musí prozkoumat nové API. Jindy jde o problém s plánováním, například když si potřebujete vypůjčit UX nebo databázového experta ze skupiny kvalifikovaných pracovníků. Může jít dokonce o problém s nábořem, kdy tým potřebuje odborníka a bez něj nemůže efektivně dokončit určité druhy práce.

2.11: Dopady jiných typů falešných startů

Viděli jsme týmy, které se zavázaly dokončit pracovní položky v rámci Sprintů a kódování zvládly poměrně rychle, ale přesto nemohly dokončit testování. To samo o sobě možná není překvapivé, ale důvod je neobvyklý: testovací tým měl něco, co potřeboval (například testovací data), co neshromáždil před začátkem Sprintu, a shromažďování těchto dat se ukázalo být obtížnější nebo časově náročnější, než předpokládali.

V důsledku toho musely být pracovní položky přeneseny do dalšího Sprintu jednoduše proto, že se tým neujistil, že je skutečně připraven je dokončit v určeném čase, než začal.

Týmy někdy začnou implementovat pracovní položky, když mají stále otevřené otázky. Ve skutečnosti se mnoho lidí domnívá, že je to dělá “více agilními”.

To může způsobit obrovské množství přepracování, překvapení nebo zpoždění. Pokud odpověď na otevřenou otázku nakonec poruší předpoklad, který byl učiněn, musí se upravit veškerá práce založená na tomto předpokladu. Pokud není otevřená otázka zodpovězena do doby, kdy má být položka uzavřena, pak musí být položka buď uzavřena, když možná není hotová, nebo držena otevřená, dokud není otázka zodpovězena.

Může se stát, že tým má interní závislost - vadu, kterou je třeba opravit, předcházející úkol, který musí být dokončen, a tak dále. Pokud není řádně sledována, může způsobit stejné problémy jako nesplněná externí závislost s přidaným pokušením přepnout kontext a opravit ji.

2.12: Kumulativní náklady

Takové problémy samozřejmě vytvářejí zpoždění, přepracování a zklamané očekávání, ale tím nevýhody nekončí.

Kromě plýtvání způsobeného přepracováním to obvykle způsobuje, že projekty se opoždují. Pokud se týmy freneticky snaží uzavřít položky backlogu a nikdy si nejsou skutečně jisté, co bude potřeba k dosažení skutečného pokroku, věci, které se skutečně *potřebují* udělat, mají tendenci být odsunuty stranou.

Často, i když ne vždy, to vede ke zvýšenému tlaku na dodání. Když se projekty dostávají více a více do skluzu, vyšší management se může pokusit vrátit je zpět do kolejí tím, že požádá lidi, aby pracovali rychleji. To se nevyhnutelně promítá do delší pracovní doby.

To zase má tendenci narušovat důvěru a zhoršovat kulturu organizace. Vztahy, které by měly být založené na spolupráci, se stávají nepřátelskými. Lidé, kteří by měli spolupracovat na hledání nejlepších a nejrychlejších řešení, odvádějí energii k tomu, aby prokázali, že když se věci nevyhnutelně pokazí, nebyla to jejich chyba.

V závratném pronásledování funkcí a uzavřených pracovních položek týmy často zjišťují, že si zkracují cestu. To ve skutečnosti znamená, že nechávají trpět kvalitu (zejména kvalitu kódu). To zase znamená, že vyměňují budoucí produktivitu za iluzi pokroku v současnosti.

Jak se pracovní podmínky stávají stále nepříjemnějšími, klíčové talenty se začínají odpojovat nebo dokonce hledat práci jinde.

Organizace, které se takto chovají, “jedí osivo”, abychom tak řekli, a to hned několika způsoby. Kódová základna se stává hůře udržitelnou a lidé, kteří by ji měli udržovat, jsou všichni vyhnáni pryč.

Pokud existuje nějaká výhoda, je pro nás neviditelná.

Kapitola 3: Představení Toků zrání požadavků (RMF)

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

3.1: Čím RMF není

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

3.2: Čím RMF je

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

3.3: Podporována a doporučována je postupná adopce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

3.4: RMF 1

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

3.5: RMF 2

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

3.6: RMF 3

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 4: Je to agilní?

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

4.1: "Jednotlivci a interakce", "Fungující software"

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

4.2: Spolupráce se zákazníkem

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

4.3: Reakce na změnu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

4.4: Transparentnost

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

4.5: Zapadá do procesu, je v souladu s Agilem

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Část II: Vytváření prostoru pro připravenost

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 5: První rozšíření

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

5.1: Přípravná práce je *práce*

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

5.2: Naturalizace přípravné práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

5.3: Ilustrativní případ

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

5.4: Vzájemný dopad

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

5.5: Funkce RMF 1

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 6: Proč to lidé nedělají?

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.1: Přípravné práce jako druhořadý občan

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.2: Alergie na neproduktivní práci

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.3: Takže to bylo pohřbeno

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.4: Vliv projektového řízení

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.5: Vzorec

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.6: Projekty a odhadování

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.7: Jak ne-odhadové odhady ovlivňují přípravné práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.8: Měření rychlosti, ne velocity

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.9: Špatná měření, špatné výsledky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

6.10: Kde neleží vina

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 7: Explicitní práce na připravenosti (RMF 1)

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.1: Integrace se Synapse Framework™

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.2: Anatomie RMF 1

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.3: Chování: Rezervace kapacity pro spolupráci

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.4: Artefakt: Položka připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.5: Aktivita: Koordinační schůzka

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.6: Chování: Pokračujte ve spolupráci, dokud nedosáhnete společného porozumění

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.7: Chování: Vždy potvrďte sdílené porozumění

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

7.8: Jak RMF 1 mění pracovní postup

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 8: Dopady RMF 1

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

8.1: Život před RMF 1

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

8.2: Před: Čas strávený porozuměním

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

8.3: Po: Čas strávený porozuměním

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

8.4: Život po přijetí RMF 1

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

8.5: Základní

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 9: Uvedení RMF 1 do praxe

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.1: Vzdělávání

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.2: Minimální požadavky podle typu týmu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.3: Dohoda

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.4: Příprava

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.5: Pilotní projekt

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.6: Plné nasazení

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.7: Následná kontrola

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.8: Prohlášení úspěchu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.9: Zachování ostrážitosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.10: Co “Jak” na to?

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

9.11: Čas to uskutečnit!

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Část III: Kontrolní brány dokončení práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 10: Další potřeba

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.1: Prostor pro interpretaci

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.2: Zúžení prostoru pro interpretaci

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.3: Třetí možnost: Žádný “prostor pro manévrování”

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.4: Potenciální dopad na dokončení

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.5: Potenciální dopad na realizaci

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.6: Navrhovaná alternativa: Nenechte prostor pro chybnou interpretaci

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.7: Výhody

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.8: O obavách z analytické paralýzy

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

10.9: Další potřeba: Individuální definice hotovo

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 11: Co lidé obvykle dělají

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.1: Když je to tak skvělé, proč to lidé nedělají?

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.2: Od vysoké školy ke koučování

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.3: Přetížení koučováním

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.4: Jeden způsob, jak lidé dělají DoD: Nedělají

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.5: Pouze akceptační kritéria

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.6: Globální Definice hotovo

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.7: Bez zubů

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

11.8: Shrnutí: Termín “Definice Splnění” je častější než skutečné definice splnění

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 12: Definování Definice Splněno

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.1: Pouze o jedné pracovní položce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.2: Dokončenost

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.3: Přesnost

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.4: Struktura Definice Splněno

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.5: Specifikace

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.6: Výstupní kritéria vývoje

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.7: Vstupní kritéria produktu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.8: Více částí, jedna brána

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.9: Příklad

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.10: Mapování na váš proces

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

12.11: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 13: Individuální Definice hotovo (RMF 2)

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.1: Princip: Každá pracovní položka je jedinečná

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.2: Chování: Udržujte jednu nebo více šablon DoD

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.3: Aktivita: Definování šablony DoD

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.4: Údržba a vylepšování šablony DoD

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.5: Více šablon DoD

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.6: Chování: Používání šablon jako výchozích bodů pro Definition of Done

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.7: Chování: Odsouhlasení individuálních Definition of Done

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.8: Aktivita: Definování Definice Splněno pro pracovní položku

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.9: Další rozšíření pracovního postupu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.10: Chování: Dovést Definici Splněno ke zralosti před zahájením implementace

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.11: Aktivita: Offline analýza pro dozrání Definice Splněno PBI

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.12: Přidání dozrávání do toku

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.13: Chování: Sledování splnění v pracovních položkách

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.14: Přidání sledování postupu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.15: Chování: Podmínění práce splněním kritérií

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.16: Aktivita: Použití DoD pro určení dokončenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.17: Jak kontrolní mechanismus zapadá do celku

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

13.18: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 14: Život s RMF 1 a 2

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

14.1: Náklady

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

14.2: Časové harmonogramy

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

14.3: Dopad na implementační tým

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

14.4: Dopad na Product Ownera

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

14.5: Dopad na vedení

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

14.6: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 15: Instalace RMF 2

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

15.1: Zapojení zainteresovaných stran

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

15.2: Úroveň detailu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

15.3: Pracovní dohoda

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

15.4: Počáteční práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

15.5: Nasazení

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

15.6: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Část IV: Implementace brány

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 16: Závěrečný požadavek

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.1: Bylo to tam celou dobu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.2: Síla načasování

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.3: Obrácený pohled

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.4: Rizika a náklady

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.5: Hodnota čekání na připravenost

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.6: Dodatečný přínos

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.7: Formulace problému

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.8: Potřeba

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

16.9: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 17: Pozadí k Definici připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

17.1: Povolení k práci podle Lean

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

17.2: Vstupní kritéria sloupců v Kanbanu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

17.3: Scrum a další apokryfy agilních procesů

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

17.4: Surfování > Programování

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

17.5: Bodová řešení

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

17.6: Jsme připraveni se připravit

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 18: Definování Definice připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

18.1: Účel

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

18.2: Na míru vytvořená

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

18.3: Anatomie

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

18.4: Příklad

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

18.5: Shoda

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

18.6: Řízení průchodu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

18.7: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 19: Specifická definice připravenosti (RMF 3)

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.1: Další brána v procesu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.2: Struktura definice připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.3: Výstupní kritéria produktu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.4: Vstupní kritéria pro vývoj

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.5: Duplicita není na škodu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.6: Chování: Udržujte jednu nebo více šablon DoR

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.7: Proč mít šablonu Definice připravenosti?

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.8: Aktivita: Definování šablony DoR

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.9: Údržba šablon DoR v průběhu času

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.10: Šablony jsou jen výchozím bodem

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.11: Chování: Odsouhlasení individuálních Definic připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.12: Aktivita: Definování Definice připravenosti pracovní položky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.13: Chování: Zajistit připravenost položek před zahájením implementace

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.14: Podmínky mimo kontrolu týmu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.15: Chování: Sledování připravenosti v pracovních položkách

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.16: Chování: Řízení práce podle připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.17: Aktivita: Použití DoR k určení připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

19.18: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Část V: Syntéza

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 20: Většina termínů není důležitá

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.1: Skutečné termíny

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.2: Na libovolných termínech nezáleží

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.3: Letecké společnosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.4: Původ technického dluhu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.5: Marketingové a prodejní termíny

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.6: Projektové termíny

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.7: Existuje jiná cesta

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.8: Svévolné termíny nejsou nutné

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.9: Svévolné termíny musí být zrušeny

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.10: Skutečné termíny jsou stále faktorem

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.11: *Pis Aller*¹

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

20.12: Závěr

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

¹Poslední možnost, původně z francouzštiny.

Kapitola 21: Kompetence 1: Proces zrání požadavků

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.1: Jak dole, tak nahoře

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.2: Princip: Transparentnost veškeré nezbytné práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.3: Chování: Zodpovědnost cestuje s prací

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.4: Chování: Viditelně sledujte stav požadavků

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.5: Chování: Odhalte veškerou práci spojenou s připraveností a implementací

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.6: Aktivita: Přípravné práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.7: Chování: Odhalte veškerou práci nezbytnou k dokončení pracovní položky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.8: Chování: Upřednostňujte připravenost před termíny

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

21.9: Závěr

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 22: Jak v rámci Scrumu proudí práce a informace s RMF

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

22.1: Úvodní poznámky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

22.2: Zachycení a příprava počátečního požadavku

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

22.3: Zahájení přípravných prací

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

22.4: Plánování a provádění přípravných prací

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

22.5: Vyhodnocení výsledků přípravných prací

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

22.6: Plánování a dokončení implementace

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 23: Dopad RMF

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

23.1: Životní cyklus požadavku

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

23.2: Příklad toku informací a práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

23.3: Před zavedením

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

23.4: Po

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

23.5: Výhody

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 24: Přechod na RMF

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

24.1: Vzor adopce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

24.2: Instalace do pracovního postupu Scrum

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

24.3: Další frameworky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

24.4: Hlavní odpor: Položky práce na připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

24.5: Velká změna: Způsob myšlení

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

24.6: Rady pro změnu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

24.7: Závěr

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Kapitola 25: Je to na vás

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

25.1: Shrnutí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

25.2: Nyní je řada na vás

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Část VI: Zdroje

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Příloha A: Problém není ve Scrumu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Co je Scrum?

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Rámce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Scrum se zabývá řízením projektů a práce

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Početlost zacházení se Scrumem jako s rámcem pro produktový management

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Chybějící předepsaný mechanismus pro dozrávání požadavků

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Žádné investování technické expertízy do vývoje požadavků

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Antipatterns

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Potřebné rozšíření

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Rozšiřování Scrumu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Příloha B: The Synapse Framework™

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Co Synapse pokrývá

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Tři mistrovství

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Jak je Synapse adoptován

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Spojení dvou rámců

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Struktura rámce Synapse

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Organizační mistrovství

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Organizační kompetence

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Organizační návyky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Struktura rámce Synapse

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Anatomie praxe

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Důležitost pořadí

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Vliv Synapse na tuto knihu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Příloha C: Běžné námitky a překážky při zavádění RMF 1

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Běžné námitky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Běžné překážky

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Příloha D: Výchozí seznamy kritérií DoD

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Výchozí seznam výstupních kritérií pro vývoj

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Seznam základních vstupních kritérií produktu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Příloha E: Výchozí seznamy kritérií Definice připravenosti

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Výstupní kritéria produktu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Vstupní technická kritéria

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Vstupní kritéria Sprintu

Tento obsah není dostupný v ukázkové knize. Knihu lze zakoupit na Leanpub na <https://leanpub.com/ready-cs>.

Rejstřík

Agile transformations, iv

agilní, 23

API, 21

auto-antonyma, 14

banky, 9

Barreras, Tom, vii

Behavior-Driven Development, 7

Bespoke Definitions of Ready, iii

Bezpečnostní zranitelnosti, 18

Brána v procesu, 59

Chování: Odhalte veškerou práci spojenou
s připraveností a implementací, 67

Chování: Viditelně sledujte stav požadavků,
67

Další frameworky, 71

defekt, 23

Definice dokončení, 21

Definice Splněno, 45

definování 'hotovo', 18

design patterns, iv

dokončení pracovní položky, 17

Dovést Definici Splněno ke zralosti, 45

důvěra, 23

existující kód, 10

Explicitní práce na připravenosti, 32

extra práce, 18

Federální úvěrová instituce, 2

finance, 9

Garbage-in, garbage-out, 14

Hlavní odpor, 71

Implementace brány, 52

Individuální Definice hotovo, 44

Initial Work, 50

Instalace do pracovního postupu Scrum, 71

inženýrské praktiky, 9, 14

inženýrské týmy, 8, 9

iterace, 20

Jak poznáte, kdy je to hotové?, 21

jednotliví přispěvatelé, 4

Kanban, i

konzultanti, 7

kultura, 9

Kumulativní náklady, 23

kvalita kódu, 23

kódová základna, 24

Level of Detail, 50

Marketing a prodej, 64

nadřazená služba, 11

národní finanční infrastruktura Spojených
států, 2

nízko-kódové řešení, 3

Něco tu chybí, 1

oblast, 9

odborník na danou oblast, 6, 10

odpovědnost a sledovatelnost, 13

OutSystems, 3

PKB-Driven Development, iv

plýtvání, 23

Položka produktového backlogu, 20

Položka připravenosti, 32

položky backlogu, 23

portál pro splácení půjček, 3
 Porušení předpisů, 18
 potvrzené sdílené porozumění, 15
 pozlacování, 19
 počítání dílků, 20
 pracovní položka, 22
 pracovní postup, 33, 45
 Proces dozrávání požadavků, 68
 Proces zrání požadavků, 66
 Producore, iii
 Product Backlog Item, 7, 9, 11, 13
 Product Backlog Item's Definition of Done, 46
 Product Manager, 5
 Product Owner, 5, 9, 18
 Product, role, 4
 produktové oddělení, 4
 Produktové oddělení a vývoj, 13
 programovací doktríny poloviny dvacátého století, 6
 Prohlášení úspěchu, 36
 Project Management Institute, 2
 Projektové termíny, 65
 promarněný čas, 19
 Přejchod na Tok dozrávání požadavků, 71
 předpoklad, 23
 přepracování, 16
 přetěžování, 8
připravené, 11
 přirozené jazyky, 14
 původní systém, 6

 Ready, i, ii
 Requirements Maturation Flow, i, iii, iv
 rizika, 17
 RMF 1, 29, 79
 RMF 2, 50
 Rollout, 50

 Scrum, 5, 15, 20, 68
 sdílené porozumění, 33
 serverové zdroje, 22
 skluz, 23

Většina termínů není důležitá, 64

Waterfall, 15

Working Agreement, 50

zainterесované strany, 16

Zaměnitelní lidé, 22

Zdroje, 74

změny v průběhu Sprintu, 5

znalostní báze, 7

zpětnovazební smyčky, 12

Ztráta nebo poškození dat, 18

závislé služby, 9

řešení C#/.NET, 3

špatné požadavky, 15