

Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS	
Semestr	LS2020
Autoři – jméno, příjmení, xname	Jan Klubarski – kluj03
Téma	Strukturovaný popis škálovatelného rámce Agile Capability Maturity Model Integration
Datum odevzdání	25.5.2020

1. Úvod

Cílem práce je analyzovat možnosti spojení Agilních praktik a procesního rámce CMMI, za účelem představit oblasti, které jsou výhodné řídit pomocí tohoto spojení, a uvést typy firem, pro které je spojení CMMI a Agilních praktik výhodné.

Tato práce sestává ze 3 hlavních částí:

V první části jsou představeny základní informace o Agile CMMI, a jsou taktéž přiblíženy CMMI a agilní praktiky. Přiblížení těchto pojmů je jen do takové míry, aby čtenář dokázal pochopit tuto práci a výhody spojení Agilních praktik a procesního rámce CMMI.

V druhé části práce jsou vysvětleny možné způsoby propojení Agilních praktik a CMMI, a to i s uvedením výhod těchto spojení. Jsou uvedeny praktické oblasti problémů, které splnutím Agilních praktik a CMMI je schopné efektivně řešit. Na konci druhé části práce jsou uvedeny typy firem na základě různých článků/studií, pro které je spojení Agilních praktik a CMMI vhodné.

Ve třetí části práce již jsou uvedeny konkrétní případové studie využití Agilních praktik spolu s CMMI, jsou zde uvedeny firmy využívající CMMI v praxi a konkrétní přínosy zavedení Agilních praktik s CMMI v číslech.

Tato práce nemá za cíl poskytovat detailní informace o procesech CMMI, nýbrž jen představí oblasti CMMI a jejich účely.

2. Základní informace

Agile CMMI je metodický rámec, který uvažuje o propojení agilních praktik, využívaných při řízení projektů, spolu s CMMI, určeným pro vedení celé organizace. Základní informace o tomto rámci jsou následující:

- **Název:** Agile capability maturity model integration
- **Autorka:** Cindy Shelton
- **Publisher:** Scrumalliance (<https://www.scrumalliance.org/community/member-articles/89>)
- **Rok vzniku:** 2008
- **Kategorizace** (Horlach, Bettina, Tilo Böhmman, Ingrid Schirmer, and Paul Drews. 2018. "IT Governance in Scaling Agile Frameworks." Pp. 1789–1800 in *Multikonferenz Wirtschaftsinformatik (MKWI)*.): Transformation focus -> transformation progress
- **Základní zdroje rámce:** CMMI + SCRUM
- **Kompatibilita:** CMMI + jakékoliv agilní praktiky, CMMI 2.0

3. Přiblížení CMMI a agilních praktik

Pro pochopení významu Agile CMMI, je třeba mít alespoň základní povědomí, co vlastně CMMI a agilní praktiky jsou. Tato kapitola je právě k tomuto určena – avšak je nutné uvědomit si, že tato práce není o představení CMMI či agilních praktik jako takových, tudíž jsou v této kapitole uvedeny pouze základní informace, které čtenář k pochopení Agile CMMI potřebuje.

3.1. CMMI

CMMI je Postup zlepšování procesů obsahující esenciální elementy efektivních procesů. Může být využit napříč projektem, divizí (skupina projektů) či (preferovaně) celou organizací – pomáhá s integrací rozdílných funkcí organizace, stanovuje cíle a priority ve zlepšování procesů, a vede organizaci k provozování kvalitních procesů.^[1] Proces je v CMMI modelu definován jako sada vzájemně provázaných činností, které transformují vstupy na výstupy pro dosažení definovaného účelu.^[2] Novák (2013) CMMI definuje jako „Sada best practices a cílů zlepšování procesů, které organizace využívají pro vyhodnocení a zlepšení jejich procesů“.^[3]

Důležitá součást CMMI (konkrétněji v oblasti vývoje) je stupeň zralosti procesů organizace ve stupňovité formě – neboli ohodnocení, jak má organizace vyspělé procesy. Tyto úrovně jsou následující ^[2, str. 18]:

Tabulka 1: Stupně zralosti CMMI

Úroveň	Stupeň zralosti	Popis
1	Úvodní (initial)	<ul style="list-style-type: none">• Softwarové procesy jsou náhodné a chaotické.• Organizace nemají stabilní prostředí pro podporu procesů, reagují pouze na vzniklé problémy.
2	Řízená (managed)	<ul style="list-style-type: none">• Organizace mají definovány a zavedeny postupy řízení projektu, procesy jsou disciplinované, a i v případě stresu jsou vykonávány podle popisu.
3	Definovaná (defined)	<ul style="list-style-type: none">• Procesy jsou dobře definovány a chápány a jsou popsány ve standardech, procedurách, nástrojích a metodách.• Jsou definovány i standardy pro přizpůsobování procesů podle typu projektu.

4	Kvantitativně řízená (quantitatively managed)	<ul style="list-style-type: none"> • Procesy jsou řízeny s využitím kvantitativních technik jsou definovány metriky pro vybrané podprocesy, ty jsou měřeny a vyhodnocovány. • Na základě toho je možné předpovídat výkon procesů.
5	Optimalizovaná (optimizing)	<ul style="list-style-type: none"> • Procesy se neustále zlepšují na základě chápání příčin změn v průběhu procesů.

K úrovním zralosti se úzce váží procesní oblasti organizace, které se dají začlenit do jednotlivých úrovní zralosti – díky přítomnosti/absenci těchto procesních oblastí (a jejich kvalitou) lze stanovit stupeň znalosti organizace. Všechny procesní oblasti se dají zařadit do některé z následujících kategorií: ^(2, str.17)

- **Řízení projektů**
- **Řízení procesů**
- **Návrh a realizace**
- **Podpůrné procesy**

Tabulka 2: Procesní oblasti CMMI

Úroveň	Stupeň zralosti	Procesní oblasti (v závorce kategorie)
1	Úvodní (initial)	<ul style="list-style-type: none"> • Bez procesních oblastí
2	Řízená (managed)	<ul style="list-style-type: none"> • Plánování projektů (řízení projektů) • Monitorování a řízení (kontrola) projektů (řízení projektů) • Řízení vztahu se subdodavateli (řízení projektů) • Řízení požadavků (návrh a realizace, případně řízení projektů) • Měření a analýza (podpůrný proces) • Řízení jakosti produktů a procesů (podpůrný proces) • Řízení konfigurací (podpůrný proces)
3	Definovaná (defined)	<ul style="list-style-type: none"> • Vývoj požadavků (návrh a realizace) • Technické řešení (návrh a realizace) • Integrace produktů (návrh a realizace) • Verifikace požadavků (návrh a realizace) • Validace požadavků (návrh a realizace) • Analýza rozhodování a řízení (podpůrný proces) • Definice procesů organizace (řízení procesů) • Zaměření na procesy organizace (řízení procesů) • Školení organizace (řízení procesů) • Integrované řízení projektů (řízení projektů) • Řízení rizik (řízení projektů)
4	Kvantitativně řízená (quantitatively managed)	<ul style="list-style-type: none"> • Řízení výkonnosti procesů (řízení procesů) • Kvantitativní řízení projektů (řízení projektů)
5	Optimalizovaná (optimizing)	<ul style="list-style-type: none"> • Řízení výkonnosti organizace (řízení procesů) • Rozhodování na základě příležitostných analýz možností (podpůrný proces)

3.2. Agilní metody

Agilních metod existuje celá řada – mezi nejznámější a nejběžnější patří SCRUM, Crystal, XP (extrémní programování), Lean a FDD (Feature driven development). Co je ale nejdůležitější v agilních modách je, že se nejedná a přesně určené procesy a postupy, nýbrž o praktiky. Základ k agilním metodám slouží tzv. agilní manifesto, vydané v roce 2001. Mezi autory se řadí Kent Beck, Ken Schwaber či Jett Sutherland a mnoho dalších. Agilní manifesto pracuje se 4 prioritními hodnotami: ^[3]

- Přednost mají **Jednotlivci a interakce** před procesy a nástroji
- Užitečnější je **Fungující software** než vyčerpávající dokumentace
- Podstatnější je **Spolupráce se zákazníkem** než vyjednávání o smlouvě
- Důležitější je **Reagování na změny** než dodržování plánu

K těmto 4 prioritám se řadí i 12 důležitých principů ^[4]

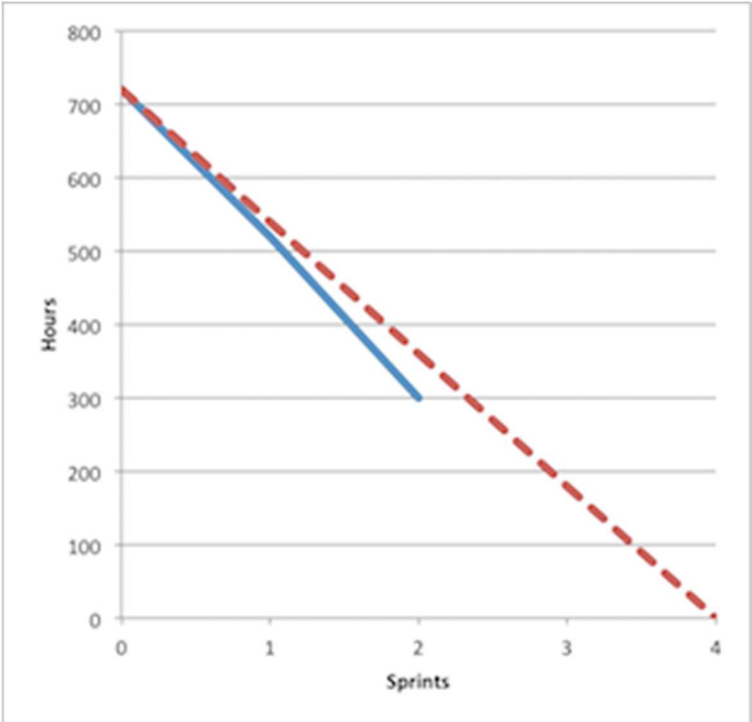
1. Nejvyšší prioritou je uspokojení zákazníka za pomoci průběžných a rychlých dodávek kvalitního software.
2. Změnové požadavky jsou vítány, dokonce i v pozdním stádiu vývoje. Agilní procesy změny zpracovávají tak, aby zákazníkovi přinášely konkurenční výhody.
3. Dodávat fungující software často, v týdenních až měsíčních intervalech. Upřednostňují se kratší intervaly dodání.
4. Lidé z businessu a vývojáři musí denně spolupracovat během projektu.
5. Pro práci na projektu vybírat motivované jedince. Dát jim prostředí a podporu, kterou potřebují, a důvěřovat jim, že práci dokončí.
6. Nejúčinnější metoda sdílení informací vývojářskému týmu (i uvnitř tohoto týmu) je osobní setkání.
7. Fungující software je primárním měřítkem progresu vývoje.
8. Agilní procesy podporují udržitelný vývoj. Sponzoři, vývojáři i uživatelé by měli být schopní dodržovat stálé tempo vývoje.
9. Průběžná pozornost věnovaná technické dokonalosti a dobrému návrhu posiluje agilní přístup.
10. Jednoduchost – umění co nejvíce práce vůbec nedělat je esenciální.
11. Nejlepší architektury, požadavky a návrhy vznikají v týmech, které se samy organizují.
12. Tým v pravidelných intervalech vyhodnocuje svou práci a upravuje své postupy tak, aby byl co nejefektivnější.

Agilní metody obsahují nespočet technik, kterými naplňují výše zmíněné principy a priority, přičemž pro tuto práci jsou nejdůležitější z nich následující (jsou ponehány v původním Anglickém znění, neboť se reálně nepřekládají) ^[5]:

Tabulka 3: Agilní techniky

Název techniky	Popis
Team Estimation Game	<ul style="list-style-type: none">• Každý člen týmu přiřazuje „body“ jednotlivým požadavkům podle náročnosti. Výsledky se zprůměrují .• Tým lze zjistit, jaké požadavky budou nejjednodušší a jaké nejsložitější.

Planning Poker	<ul style="list-style-type: none"> • Každý člen týmu přiřadí jednotlivým požadavkům body pomocí karet. Karty se odkryjí až poté, co již všichni budou mít požadavky ohodnoceny. • Tento přístup zabraňuje zaujetí vůči požadavku na základě precedentu od kolegy.
Spring Planning	<ul style="list-style-type: none"> • Schůzka, která trvá cca hodinu týdně, během které se tým dohodne, jaké požadavky je třeba během daného týdne dokončit. • Sprint je týden až měsíc dlouhý interval, ve kterém se vyvíjí (a dodává) nová verze produktu.
Backlog grooming	<ul style="list-style-type: none"> • Schůzka zákazníka a dodavatele určená k tomu, aby se reevidovaly požadavky podle priorit, a aby se určilo, že nejprioritnější požadavky jsou/budou připraveny k dodání do následujícího releasu.
Requirements Development	<ul style="list-style-type: none"> • Je proces sestávající z 6 kroků, jehož výstupem je seznam požadavků vyvíjeného produktu. • Kroky jsou: 1) Vytvořit požadavky 2) Sepsat a zdokumentovat požadavky 3) Zkontrolovat úplnost požadavků 4) Analyzovat, vylepšit a rozložit požadavky 5) Zvalidovat požadavky 6) Řídit požadavky. ^[6]
Task Estimation	<ul style="list-style-type: none"> • Je proces určený k předpovědi doby (standardně v člověkohodinách) nutné využít k tomu, aby se daný požadavek vyvinul.
Release Planning	<ul style="list-style-type: none"> • Proces, ve kterém se stanovují chtěné výsledky releasu a maximalizují se šance k tomu, aby se tyto výsledky naplnily. • Stanovují se jasné, specifické a měřitelné cíle.
Sprint Backlog	<ul style="list-style-type: none"> • Je seznam požadavků, které se mají dokončit během sprintu (výstupy ze sprintu)
Daily Standup / Daily Scrum	<ul style="list-style-type: none"> • Je krátký každodenní meeting určený k tomu, aby každý člen týmu dal zpětnou vazbu na to, co se mu povedlo/nepovedlo dokončit a na čem pracuje

<p>Release Burndown</p>	<ul style="list-style-type: none"> Graf znázorňující reálnou zbývající práci na projektu oproti ideálnímu průběhu zbývající práce. Na ose X jsou sprinty, na ose y zbývající čas k dokončení práce (např. v hodinách).  <p style="text-align: center;"><i>Figure 1: Release burndown chart. Zdroj: study.com^[7]</i></p>
<p>Sprint Burndown</p>	<ul style="list-style-type: none"> Graf znázorňující reálnou zbývající práci ve sprintu oproti ideálnímu průběhu zbývající práce. Graf je defacto identický jako Release Burndown, ovšem vztahuje se na sprint, nikoliv na celý projekt.
<p>Release on Demand</p>	<ul style="list-style-type: none"> Proces, který dodává vyvinutý požadavek/funkcionalitu produktu do produkce a vydává ji zákazníkovi ihned nebo inkrementálně v závislosti na poptávce zákazníka.
<p>Incremental Release</p>	<ul style="list-style-type: none"> Metoda softwarového vývoje, kdy je produkt dodáván po částech v různých iteracích (např. čtvrtletí).
<p>Product Backlog</p>	<ul style="list-style-type: none"> Prioritovaný seznam nových funkcionalit/požadavků, změn k již vyvinutým funkcionalitám, oprav chyb, změn architektury/infrastruktury či jiných funkcionalit určených k dodání v rámci produktu.
<p>User Stories/Epics</p>	<ul style="list-style-type: none"> Jsou krátké, jednoduché popisy požadavků od zákazníka, který tyto požadavky požaduje.
<p>Definition of Done</p>	<ul style="list-style-type: none"> Akceptační kritéria nebo podmínky, kterých software musí dosáhnout k tomu, aby uspočil zákazníka.
<p>Top 10 Features</p>	<ul style="list-style-type: none"> Seznam 10 nejdůležitějších požadavků, které produkt musí obsahovat.
<p>Sprint Demo</p>	<ul style="list-style-type: none"> Aktivita, při které se (standardně zákazníkovi) ukáží nové funkcionality/požadavky produktu, vyvinuté během sprintu.
<p>Sprint Retrospective</p>	<ul style="list-style-type: none"> Schůzka, při které tým diskutuje o předešlém sprintu – co fungovalo, co nefungovalo. Na základě toho se tým domluví, co zlepšit.

Test Driven Development	<ul style="list-style-type: none"> • Metoda softwarového vývoje, během níž mají prioritu testy. • Testy se píšou jako první, a až po vyhotovení testů k požadavku, se začíná požadavek kódovat.
Continuous Build / Continuous integration	<ul style="list-style-type: none"> • Praktika vývoje, během níž se do sdíleného repozitáře commituje kód velmi frekventovaně, ideálně i několikrát denně.
Refactoring	<ul style="list-style-type: none"> • Proces provádění změn (přepisování) kódu takovým způsobem, aby funkcionality kódu zůstala zachována. • Účelem je úprava vnitřní struktury kódu tak, aby byl kód čistší, optimalizovanější a snadnější ke čtení, a to s pouze minimálním (či, preferovaně, žádným) rizikem zanesení chyb do kódu.
Team Agreements	<ul style="list-style-type: none"> • Jsou domluvy či ustanovení, které slouží k minimalizaci sporů/napětí mezi členy týmu. • Obsahují šablony toho, co se od každého člena týmu očekává, aby dělal během pracovního dne. (Např. jeden člen pracuje na modulu reportů v aplikaci, druhý člen se zaměřuje primárně na modul obchodu v aplikaci atp.).
Vision	<ul style="list-style-type: none"> • Vize toho, čeho se zákazník a dodavatel snaží dosáhnout vývojem daného produktu.
Product Owner Meeting with Customer	<ul style="list-style-type: none"> • Schůzka, na které se zákazník (Product owner) setkává s uživateli produktu (Customer), aby zjistil, co uživatel od produktu očekává.
Pair Programming	<ul style="list-style-type: none"> • Praktika vývoje software, která se nejčastěji využívá v extrémním programování. • 2 vývojáři píšou tentýž kód společně, u jednoho monitoru. • Snižuje se riziko chyby v kódu a nutnosti refaktoringu.
Retrospectives	<ul style="list-style-type: none"> • Týmové setkání zaměřené na pohledy na události, které se již odehrály. • Mají za účel zlepšení/zefektivnění budoucích událostí podobného typu na základě zpětných vazeb na předešlé události.

4. Propojení Agile a CMMI

Propojení Agile a CMMI je výhodné z řady důvodů, např. posílení škálovatelnosti agilních praktik mimo malé oddělné týmy – přímo do celého podniku. Podle některých zdrojů (zvláště těch starších, z počátku 21. století) toto propojení není možné, a to z toho důvodu, že CMMI je striktní pohled v rámci procesů. Toto však není pravda, protože CMMI neříká, jak konkrétně se mají věci provádět, ale pouze definuje **CO** má organizace provést, aby byla úspěšná. Ačkoliv je CMMI založeno na procesech, tak proces v CMMI není chápán jako něco, co je striktně definováno, avšak je chápán jako „co je třeba dělat“, nikoliv „jak je to třeba dělat“.^[8] Agilní metody na druhou stranu neříkají co se má dělat, nýbrž **JAK** se to má dělat.^[5]

Zároveň by se do propojení CMMI a Agile daly zařadit i nějaké „zlepšovací techniky“, jakožto SixSigma, Theory of Constraints či Lean (v některých zdrojích uváděn jakožto agilní metoda). Tento postup by se dal jednoduše znázornit následujícím schématem: ^[1]

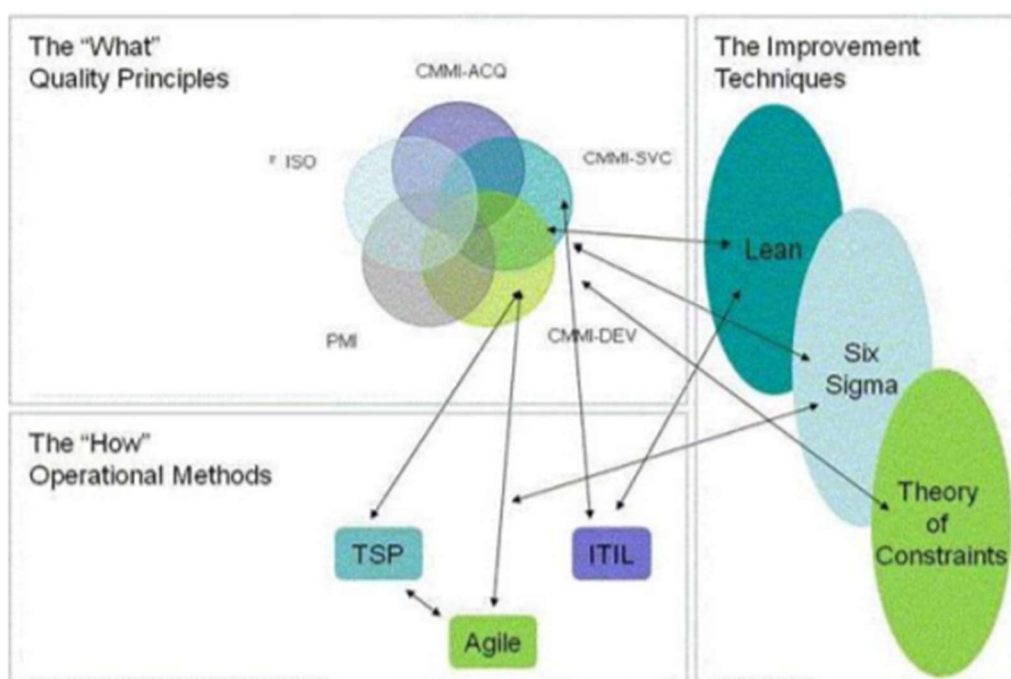


Figure 2: Propojení Agilních metod, CMMI a "zlepšovacích technik". Zdroj: scrumalliance.org ^[1]

Zdánlivý konflikt mezi CMMI a Agilními metodami pramení díky tomu, že Agilní metody se zaměřují na produkt a jeho výstupy, zatímco výstupy, které napřímo nepřispívají produktu, se v agilních metodách berou jako „nadbytečnost“ neboli „plýtvání“. Nejlepší perspektiva, jak vyvrátit tento zdánlivý konflikt je řízení rizik (risk management) – projekty a programy by měly cílit toho, aby dodaly produkt (produkty) s minimálním rizikem k projektu, programu a organizaci. ^[5]

Schéma efektivního řízení rizik by se dalo zobrazit následovně:^[5]

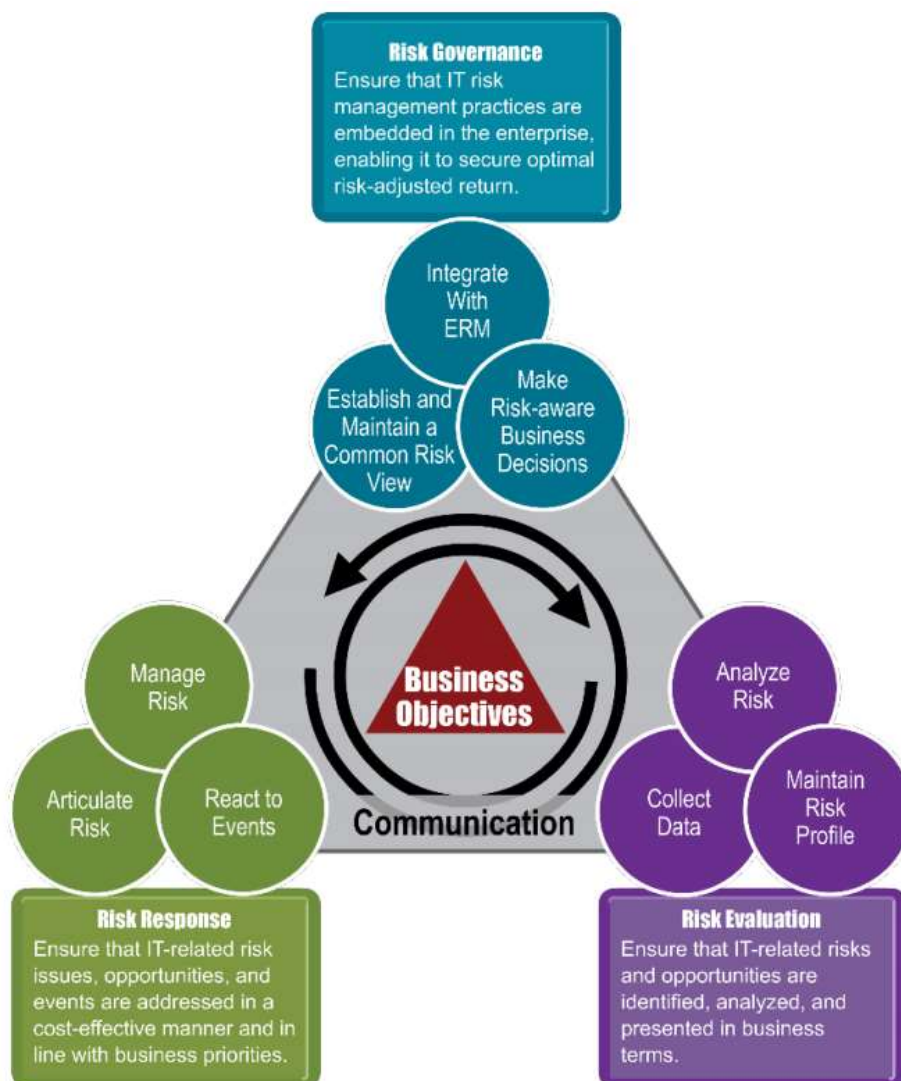


Figure 3: Risk IT Framework for Management of IT Related Business Risks (ISACA). Zdroj: selectcomputing.com ^[5]

Následující procesní oblasti CMMI se v agilních metodách mohou jevit jako „plýtvání“, avšak podporují efektivní řízení rizik (Stupeň zralosti – Maturity Level je uveden ve zkratce jako ML) ^[5]:

- **Měření a analýza (Measurement and Analysis) – ML2**
 - Tato procesní oblast je určena k prioritizaci backlogu – neboli neustálému hodnocení stories a prioritizaci překážek/chyb.
- **Monitorování a řízení (kontrola) projektů (Project Monitoring and Control) – ML2**
 - Tato oblast má za cíl porozumění postupu projektu a reálným vychýlením od požadavků a plánu. Tato vychýlení pomáhají identifikaci rizik, které mohou dosáhnout až na úroveň celé organizace.
- **Měření jakosti produktů a procesů (Process and Product Quality Assurance) – ML2**
 - Procesní oblast je určena ke kontrole, že tým dodržuje metodologii a dodává dílčí výstupy produktu.
- **Analýza rozhodování a řízení (Decision Analysis and Resolution) – ML3**
 - V této oblasti se definují výběrová kritéria a porozumívá se jim, taktéž se analyzují možné alternativy.

- **Řízení rizik (Risk Management) – ML3**
 - Tato oblast identifikuje potenciaální problémy, kterým se může předejít či které mohou být zmírněny.
- **Řízení výkonnosti procesů (Organisational Process Performance) – ML4**
 - Tato procesní oblast má za cíl analyzovat organizační výkonnost a zlepšovat schopnosti organizace provádět projekty.
- **Kvantitativní řízení projektů (Quantitative Project Management) – ML4**
 - V této procesní oblasti se porovnávají výkonnosti projektů napříč organizací. Výsledky přispívají k odpovědi, zda je určitý projekt na správné cestě k jeho úspěšnému zakončení.
 - Výsledky této odpovědi jsou důležité i pro agilně řízené projekty, a to z toho důvodu, že mohou pomoci určit, jak či zda budou budoucí projekty podobného rázu úspěšné.

Z těchto procesních oblastí je možné odvodit, že CMMI a Agilní přístupy mají vlastně hodně společného, a to i díky společnému základu na demingovém cyklu (PLAN-DO-CHECK-ACT). Dále je také možné určit, že úroveň zralosti 2 formalizuje Risk Response, úroveň zralosti 3 formalizuje Risk Evaluation, a úroveň zralosti 4 formalizuje Risk Governance (viz obrázek 3).^[5]

4.1. Zlepšení výkonnosti organizace propojením Agile a CMMI

Procesní oblasti CMMI zlepšují schopnosti projektového týmu vykonávat agilní praktiky v závislosti na různé výzvy/problémy v projektu. Následná tabulka, adoptovaná z CMMIInstitute.com, znázorňuje časté business problémy, kde CMMI a Agilní praktiky poskytují „návody“ a rady ke zlepšení. V prvním sloupci jsou uvedeny typy problémů, na které organizace může narazit během jejich projektů, ve druhém sloupci jsou agilní techniky (především ze SCRUMu) určené k tomu, jak by se dalo daný problém vyřešit. Třetí sloupec referencuje CMMI procesní oblasti, které se zaměřují na nutné schopnosti organizace, jež jsou potřebné pro aplikaci daných agilních technik. Organizace tedy využívají CMMI procesní oblasti k tomu, aby posílily svou schopnost vykonávat agilní techniky k zajištění škálovatelnosti a pružnosti/odolnosti projektů.^[5]

Tabulka 4: Problémy a jejich řešení díky propojení Agilních technik a CMMI procesních oblastí

Problém	Agilní Techniky	CMMI oblasti (v závorce úroveň zralosti)
Projektové odhady jsou nerealistické nebo neznámé	<ul style="list-style-type: none"> • Team Estimating Game • Planning Poker • Sprint Planning • Backlog Grooming • Requirements Development • Task Estimation • Release Planning • Sprint Backlog 	<ul style="list-style-type: none"> • Plánování projektů (ML2) • Integrované řízení projektů (ML3) • Řízení požadavků (ML2) • Vývoj požadavků (ML3) • Měření a analýza (ML2) • Kvantitativní řízení projektů (ML4)

Projekty nejsou dodávány včas	<ul style="list-style-type: none"> • Daily Standup/ Daily Scrum • Release Burndown • Sprint Burndown • Task Estimation • Release on Demand • Incremental Release 	<ul style="list-style-type: none"> • Monitorování a řízení (kontrola) projektů (ML 2) • Měření a analýza (ML 2) • Kvantitativní řízení projektů (ML 4)
Požadavky se neustále mění	<ul style="list-style-type: none"> • Backlog Grooming • Sprint Planning • Product Backlog • User Stories/ Epics • Definition of Done (for User Stories) • Top "10" Features • Release Planning 	<ul style="list-style-type: none"> • Řízení požadavků (ML 2) • Monitorování a řízení (kontrola) projektů (ML 2) • Vývoj požadavků (ML3) • Měření a analýza (ML2) • Verifikace požadavků (ML3)
Zákazník se nesvěří projektu	<ul style="list-style-type: none"> • Release Planning • Sprint Planning 	<ul style="list-style-type: none"> • Řízení požadavků (ML2) • Plánování projektů (ML 2)
Zákazníci jsou frustrováni průběhem projektu	<ul style="list-style-type: none"> • Release Planning • Sprint Planning • Sprint Demo • Sprint Retrospective 	<ul style="list-style-type: none"> • Integrované řízení projektů (ML 3) • Validace požadavků (ML 3) • Plánování projektů (ML 2) • Monitorování a řízení (kontrola) projektů (ML 2) • Měření a analýza (ML 2)
Zákazníci nejsou spokojeni s finálními produkty	<ul style="list-style-type: none"> • Definition of Done • User Stories/ Epics • Test Driven Development • Sprint Demo 	<ul style="list-style-type: none"> • Validace požadavků (ML3) • Verifikace požadavků (ML3) • Vývoj požadavků (ML3) • Řízení požadavků (ML2)
V projektu či post-projektu se vyskytuje příliš mnoho chyb (bugů)	<ul style="list-style-type: none"> • Test Driven Development • Continuous Build/ Continuous Integration • Refactoring 	<ul style="list-style-type: none"> • Validace požadavků (ML3) • Verifikace požadavků (ML3) • Integrace produktů (ML3) • Technické řešení (ML3)

Neschopnost zajistit či udržet projektové zdroje	<ul style="list-style-type: none"> • Team Agreements • Release Planning • Daily Standup • Vision 	<ul style="list-style-type: none"> • Integrované řízení projektů (ML3) • Plánování projektů (ML2) • Řízení rizik (ML3) • Monitorování a řízení (kontrola) projektů (ML2) • Školení organizace (ML3)
Nedostatek vnitřní či vnější podpory projektu	<ul style="list-style-type: none"> • Team Agreements • Release Planning • Sprint Planning • Daily Standup • Product Owner Meeting with Customer 	<ul style="list-style-type: none"> • Integrované řízení projektů (ML 3) • Řízení rizik (ML3) • Plánování projektů (ML2) • Monitorování a řízení (kontrola) projektů (ML2)
Špatná komunikace v projektu	<ul style="list-style-type: none"> • Daily Standup • Release Planning • Sprint Planning • Sprint Demo • Sprint Retrospective • Product Owner Meeting with Customer 	<ul style="list-style-type: none"> • Monitorování a řízení (kontrola) projektů (ML2) • Plánování projektů (ML2) • Integrované řízení projektů (ML3)
Požadavky jsou příliš obecné	<ul style="list-style-type: none"> • User Story/ Epic • Definition of Done • Backlog Grooming 	<ul style="list-style-type: none"> • Řízení požadavků (ML2) • Vývoj požadavků (ML3)
Odpovědnosti v rámci týmu nejsou zřejmé	<ul style="list-style-type: none"> • Sprint Planning • Release Planning • Incremental Release 	<ul style="list-style-type: none"> • Plánování projektů (ML2)
Neadekvátní znalosti projektu člena týmu v rámci požadavku	<ul style="list-style-type: none"> • Release Planning 	<ul style="list-style-type: none"> • Školení organizace (ML3) • Plánování projektů (ML2)
Neschopnost úspěšně naplánovat projekt	<ul style="list-style-type: none"> • Release Planning • Sprint Planning • Backlog Grooming 	<ul style="list-style-type: none"> • Plánování projektů (ML2) • Řízení požadavků (ML2) • Kvantitativní řízení projektů (ML4)

Neschopnost předvídat potenciální problémy	<ul style="list-style-type: none"> • User Stories/ Epics • Sprint Planning • Release Planning • Daily Standup • Release Burndown • Sprint Burndown • Sprint Demo • Vision 	<ul style="list-style-type: none"> • Monitorování a řízení (kontrola) projektů (ML2) • Integrované řízení projektů (ML3) • Řízení rizik (ML3) • Měření a analýza (ML2) • Verifikace požadavků (ML3)
Nízká kvalita kódu	<ul style="list-style-type: none"> • Pair Programming • Test Driven Development • Continuous Build/ Integration • Retrospectives • Definition of Done 	<ul style="list-style-type: none"> • Technické řešení (ML3) • Verifikace požadavků (ML3) • Validace požadavků (ML3) • Integrované řízení projektů (ML3) • Řízení jakosti produktů a procesů (ML2)
Výkon organizace se nezlepšuje	<ul style="list-style-type: none"> • Retrospectives 	<ul style="list-style-type: none"> • Integrované řízení projektů (ML3) • Zaměření se na procesy organizace (ML3) • Definice procesů organizace (ML3) • Školení organizace (ML3) • Řízení jakosti produktů a procesů (ML2) • Řízení výkonnosti procesů (ML4) • Kvantitativní řízení projektů (ML4) • Rozhodování na základě příležitostných analýz možností (ML5)

4.2. Kdy propojovat Agilní metody s CMMI

C. Sheltonová ve svém článku z roku 2008 tvrdí, že nejlepší synergie mezi CMMI a agilními praktikami nastává právě ve 3. úrovni zralosti (ML3), a to z toho důvodu, že pokrok agilně řízených projektů je viditelný - je možné kontrolovat cenu, plán i scope projektu. V případě 1. stupně úrovně zralosti (ML1), se procesy mění na základě uživatele či události – agilní projekty mohou být úspěšné, ale ne opakovatelné. ^[1]

Ve 2. úrovni zralosti (ML2) jsou sice některé procesy opakovatelné, avšak ne pro celou organizaci. Agilní praktiky přinášejí strukturu a disciplínu potřebnou ke zvýšení úrovně zralosti na 3. úroveň, ale implementace agilních praktik jsou obvykle velmi nákladné. Nicméně tato implementace dokáže zdůraznit efektivitu agilních praktik. ^[1]

Ve 4. a 5. úrovni zralosti jsou již procesy napříč organizací stejné, a tak je flexibilita agilních praktik limitována. Výkon/provedení procesů je standardně kontrolováno pomocí statistických a jiných kvantitativních metod. Avšak autorka tvrdí, že nikdy neměla zkušenosti se zaváděním agilních praktik v organizacích dosahujících 4. a vyšší úroveň zralosti.^[1]

Článek na selectcomputing.com z roku 2017 naopak tvrdí, že nejvyššího efektu splynutí CMMI s Agilními praktikami je dosaženo právě od 4. stupně zralosti (ML4). Je to z toho důvodu, že 4. úroveň zralosti vytváří kvantitativní cíle pro kvalitu a výkonnosti procesů organizace, které jsou využity jako kritéria pro řízení projektů. Díky statistickým a kvantitativním technikám je možné vytvářet realistické predikce výkonnosti budoucích projektů, což zapříčiní možnost využití agilních přístupů i na velkých projektech.^[5]

Z těchto dvou článků lze tedy určit, že propojení CMMI a agilních praktik je možné již od 2. stupně zralosti (ML2), avšak za cenu vysoké ceny implementace agilních metod. Při 3. stupni zralosti (ML3) nastává vysoká efektivita tohoto spojení, která vrcholí ve 4. stupni zralosti (ML4). Ze statistiky plyne, že propojení CMMI a Agilních metod je vhodný pro naprostou většinu organizací využívajících CMMI, neboť značná většina těchto organizací dosahuje právě 3. úrovně zralosti, přičemž značná část organizací dosahuje 5. úrovně znalosti – i když v této úrovni pravděpodobně není propojení Agilních metod a CMMI tak výhodné, jako ve 4. úrovni zralosti.

Appraisal Trend by Maturity Level 2008–2018*

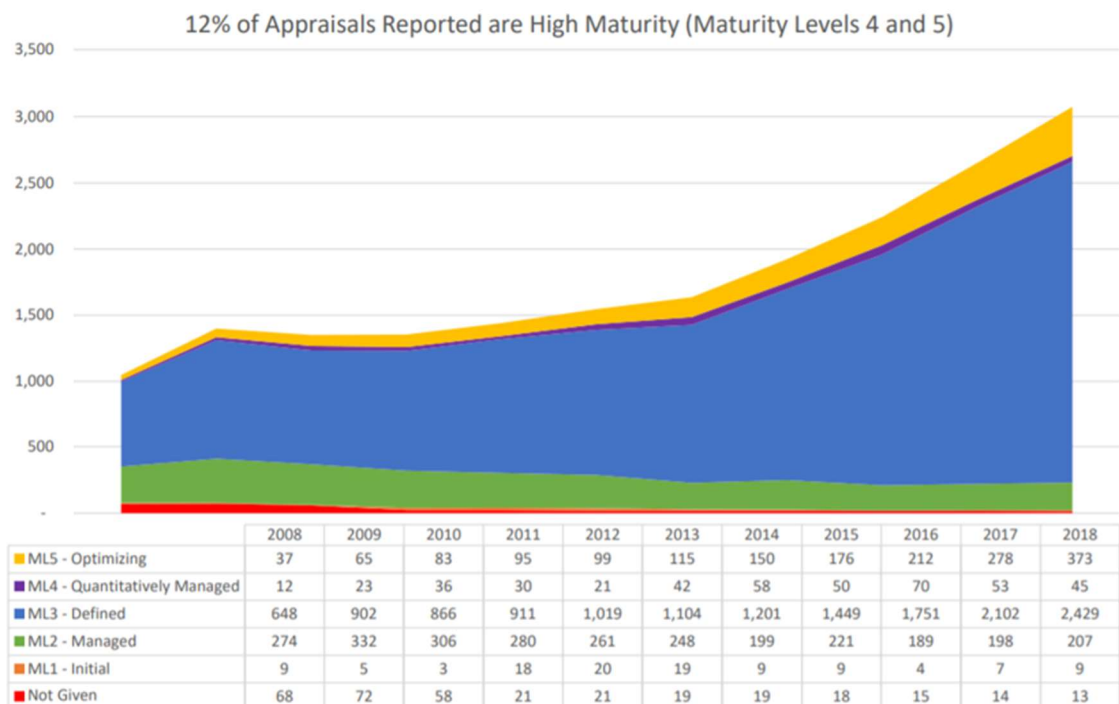


Figure 4: Graf podílu úrovní zralostí napříč organizacemi využívajícími CMMI. Zdroj: cmminstitute.com^[9]

5. Reference a případové studie

V roce 2019 byla vydána verze CMMI 2.0 – ta se zaměřuje konkrétně na využívání CMMI spolu s agilními metodami, konkrétněji se SCRUM metodikou. CMMI 2.0 není veřejně přístupná, nicméně z velké části popisuje výhody splynutí CMMI a agilních metod obdobně jako článek ze Selectcomputing z roku 2017. Mezi hlavní výhody CMMI 2.0 patří: ^[10]

- CMMI V2.0 posiluje Agilitu organizace SCRUM procesy za pomoci přímého vedení obsaženého v core practice areas.
- CMMI V2.0 zlepšuje Agilní nasazení škálováním agilních přístupů napříč celou organizací.
- CMMI V2.0 pro Agilitu pomáhá uživatelům udržovat konstatní tempo vývoje díky využíváním agilních metod, technik, a celé capability area zaměřující se na best practices pro řízení a udržování pokroku.

CMMI Institute v roce 2019 publikovalo případovou studii o využívání SCRUM společně s CMMI. Ve studii stojí hlavně že ^[11]:

- 80% firem využívajících CMMI se snaží využívat agilní praktiky.
- Avšak až 80% firem nedokáže plně využívat agilních praktik.
- 60% organizací má méně než polovinu týmů využívajících agilních praktik.
- 47% týmů nemá zkušenosti s využíváním agilních metod a technik.
- CMMI 2.0 je specificky stvořené k tomu, aby podporovalo agilní přístupy.
- Agilní přístupy je možné díky CMMI jednoduše a kvalitně škálovat.

5.1. Využití CMMI a Agilních přístupů v praxi

V této podkapitole jsou uvedeny výsledky firem, které využily splynutí agilních technik a CMMI. Zavedení Agile CMMI mělo za následek ^[8]:

Tabulka 5: Přínosy zavedení Agile CMMI v praxi

Organizace + popis	Přínosy zavedení Agile CMMI v praxi
Honeywell Technology Solutions (HTS) <ul style="list-style-type: none">• Vyvíjí software, produkty a služby pro klíčové oblasti podnikání firmy jako jsou například vzdušný prostor či automatizovaná a řídicí řešení.	<ul style="list-style-type: none">• O 12-15% méně defektů v aplikacích.• 15% zlepšení implementace strategie Kaizen za 4 roky.• Snížení doby nutné k zaučení/zaškolení zaměstnanců.• Zvýšení sdílení informací napříč organizací.
Minacs <ul style="list-style-type: none">• Nabídka outsourcingových řešení především bankám. IT oddělení firmy sídlí v Indii.	<ul style="list-style-type: none">• O 30-40 % vyšší dosažení cílů během sprintů.• O 30% vyšší počet implementací nových funkcionalit během sprintu.• O 7% vyšší zisk firmy.
Perficient <ul style="list-style-type: none">• Konzultace a pomoc ostatním firmám v IT oblasti.	<ul style="list-style-type: none">• 80% všech projektů funguje na konzistentních agilních praktikách (k roku 2016).• Reportování neshod na projektech kleslo o 70%.

Dle CMMI Institute existuje přes 10 000 organizací využívajících CMMI. Z nich aspoň 80% využívalo agilních praktik/metod v roce 2019 – nárůst z 28% v roce 2009. ^[10] Z těchto firem to jsou například následující: ^[9,12]

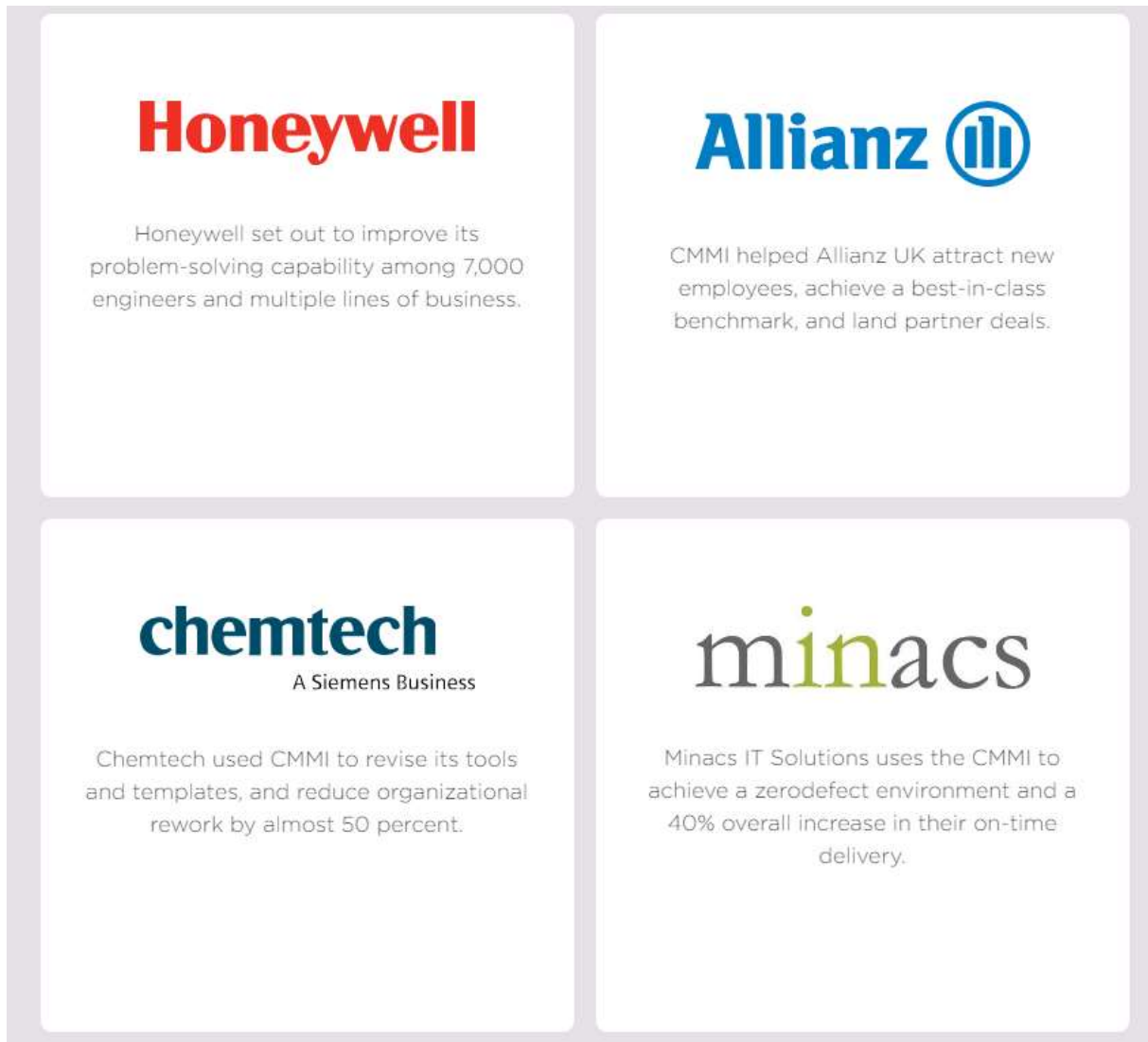


Figure 5: Společnosti využívající CMMI 2.0. Zdroj: cmmiinstitute.com^[12]

Aerospace and Defense

	RANK	COMPANY	REVENUES (\$M)
	24	Boeing	\$94,571
	50	United Technologies	\$57,244
	56	Lockheed Martin	\$50,658
	90	General Dynamics	\$31,353
	114	Northrop Grumman	\$24,508
	116	Raytheon	\$24,069
	200	Textron	\$13,788
	220	Arconic	\$12,394
	273	L3 Technologies	\$10,597
	300	Huntington Ingalls Industries	\$7,068
	393	Spirit AeroSystems Holdings	\$6,793
	492	Rockwell Collins	\$5,259

Information Technology Services









	RANK	COMPANY	REVENUES (\$M)
	32	IBM	\$79,919
	59	Hewlett Packard Enterprise	\$50,123
	199	CDW	\$13,982
	205	Cognizant Technology Solutions	\$13,487
	379	Computer Sciences	\$7,106
	381	Leidos Holdings	\$7,043
	481	Booz Allen Hamilton Holding	\$5,406

Figure 6: Společnosti využívající CMMI. Zdroj: cmmiinstitute.com [9]

Computer Software

	RANK	COMPANY	REVENUES (\$M)
	28	Microsoft	\$85,320
	81	Oracle	\$37,047
	326	salesforce.com	\$8,392
	406	Activision Blizzard	\$6,608
	443	Adobe Systems	\$5,854

Computers/Office Equipment








	RANK	COMPANY	REVENUES (\$M)
	3	Apple	\$215,639
	41	Dell Technologies	\$64,806
	61	HP	\$48,238
	409	NCR	\$6,543
	663	Pitney Bowes	\$3,407
	672	Diebold Nixdorf	\$3,341
	884	Super Micro Computer	\$2,216

Figure 7: Společnosti využívající CMMI. Zdroj: cmmiinstitute.com [9]

Semiconductors/Other
Electronic Components



Insurance – Life,
Health (Stock)



Wholesalers
- Healthcare



Industrial
Machinery



Construction &
Farm Machinery



Motor Vehicles
& Parts



Wholesalers –
Electronics &
Office Equipment



Electronics/Electrical
Equipment



Financial Data Services



Telecommunications



Healthcare – Insurance/
Managed Care



Commercial
Banks



Engineering/Construct
ion



Network & Other
Communications
Equipment



Figure 8: Společnosti využívající CMMI. Zdroj: cmminstitute.com [9]

6. Závěr

Cílem práce bylo analyzovat možnosti spojení Agilních praktik a procesního rámce CMMI, za účelem představit oblasti, které jsou výhodné řídit pomocí tohoto spojení, a uvést typy firem, pro které je spojení CMMI a Agilních praktik výhodné.

Cíl je možné považovat za splněný v jeho celé výši. Bylo zjištěno, že CMMI definuje, **CO** je třeba dělat, aby byla organizace úspěšná, zatímco Agilní praktiky říkají, **JAK** toho lze dosáhnout. Nejvyšší opodstatnění spojení Agilních praktik a CMMI je v řízení rizik, a to řešením či eliminováním možných problémů jako např. že projekty nejsou dodávány včas, požadavky se neustále mění, zákazníci nejsou spokojeni s finálními produkty, špatná komunikace v projektu apod. Spojení Agilních praktik a CMMI je velmi výhodné pro organizace dosahujících 3. a vyšší úrovní zralosti, přičemž nejlepšího výsledku je dosaženo pravděpodobně ve 4. úrovni zralosti (dle Selectcomputing.com z roku 2017).

V první části jsou představeny základní informace o Agile CMMI, a jsou taktéž přiblíženy CMMI a agilní praktiky. Přiblížení těchto pojmů je jen do takové míry, aby čtenář dokázal pochopit tuto práci a výhody spojení Agilních praktik a procesního rámce CMMI.

V druhé části práce jsou vysvětleny možné způsoby propojení Agilních praktik a CMMI, a to i s uvedením výhod těchto spojení. Jsou uvedeny praktické oblasti problémů, které splnutí Agilních praktik a CMMI je schopné efektivně řešit. Na konci druhé části práce jsou uvedeny typy firem na základě různých článků/studií, pro které je spojení Agilních praktik a CMMI vhodné.

Ve třetí části práce již jsou uvedeny konkrétní případové studie využití Agilních praktik spolu s CMMI, jsou zde uvedeny firmy využívající CMMI v praxi a konkrétní přínosy zavedení Agilních praktik s CMMI v číslech.

Problémy ve zpracování této práce byly například nedostatek informací o CMMI 2.0 a nízký počet relevantních článků/prací pojednávajících o propojení Agilních praktik a CMMI. Avšak i přes tyto nedostatky bylo možné práci vyhotovit a cílů dosáhnout.

Literatura a zdroje

- 1) **Shelton, Cindy, 2008.** Agile and CMMI: Better Together, 7/2008. scrumalliance.org [Online] Cit. xx.05.2020. Dostupné z: <https://www.scrumalliance.org/community/member-articles/89>
- 2) **BuchalcevoVá, Alena, 2020.** Zlepšování procesů budování IT – CMMI (přednáška), 4/2020. vse.cz [Online] Cit. xx.05.2020. Veřejnosti nedostupné.
- 3) **Beck, Kent, Schwaber, Ken, Sutherland, Jett a spol, 2001.** Agile manifesto, 2/2001. agilemanifesto.org [Online] Cit. xx.05.2020. Dostupné z: <https://agilemanifesto.org/iso/cs/manifesto.html>
- 4) **Beck, Kent, Schwaber, Ken, Sutherland, Jett a spol, 2001.** Agile manifesto, 2/2001. agilealliance.org [Online] Cit. xx.05.2020. Dostupné z: <https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/>
- 5) **Huff, Arnold (Selectcomputing), 2017.** CMMI and Agile: Apply Both for Project Success, 6/2017. selectcomputing.com [Online] Cit. xx.05.2020. Dostupné z: http://www.selectcomputing.com/doc/SCi_CMMI-Agile_Apply%20Both_White%20Paper_to%20CEEF_09182017_fnl_1.pdf
- 6) **Acqnotes.com, 2018.** Requirements Development Steps, 6/2018. acqnotes.com [Online] Cit. xx.05.2020. Dostupné z: <http://acqnotes.com/acqnote/tasks/requirements-development-steps>
- 7) **Bugajenko, Olga, 2016.** Release Burndown Chart in Scrum: Example & Overview, 9/2016. study.com [Online] Dostupné z: <https://study.com/academy/lesson/release-burndown-chart-in-scrum-example-overview.html>
- 8) **Menčík, Michal, Hemžal, Adam, Šeda, Matěj, 2017.** Helping Organisations Scale Agile, 11/2017. spicenter.vse.cz [Online] Cit. xx.05.2020. Dostupné z: https://spicenter.vse.cz/wp-content/uploads/2018/08/semestrální_práce/17_12/HelpingOrganisationsScaleAgile.pdf
- 9) **CMMI Institute, 2018.** CMMI Maturity Profile Report, 6/2018, cmmiinstitute.com [Online] Cit. xx.05.2020. Dostupné z: <https://cmmiinstitute.com/getattachment/b9ce903c-92af-4e9f-a339-4600459556c0/attachment.aspx>
- 10) **CMMI Institute, 2020.** CMMI and Agile, 5/2020, cmmiinstitute.com [Online] Cit. xx.05.2020. Dostupné z: <https://cmmiinstitute.com/agile>
- 11) **CMMI Institute, 2019.** Using Agile with Scrum and CMMI, 28/6/2019, cmmiinstitute.com [Online] Cit. xx.05.2020. Dostupné z: <https://cmmiinstitute.com/getattachment/fb076975-67c3-4cff-b373-2d32f3653786/attachment.aspx>
- 12) **CMMI Institute, 2020.** Who uses CMMI, 5/2020, cmmiinstitute.com [Online] Cit. xx.05.2020. Dostupné z: <https://cmmiinstitute.com/who-uses-cmmi>