

<b>Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS</b>	
<b>Semestr</b>	LS 2019/20
<b>Autor</b>	Michael, Linka, linm02
<b>Téma</b>	Strukturovaný popis škálovatelného rámce Driving Strategy, Delivering More (DSDM)
<b>Datum odevzdání</b>	11.6. 2020

## Úvod

Agilních škálovatelných rámců je na trhu dostupné velké množství. Jednotlivé z nich mají mezi sebou větší či menší rozdíly. V této práci bude podrobněji rozebrán rámec DSDM Agile Project Framework, který je jedním z nejstarších. V současnosti se ale netěší moc velké oblibě a je na agilním trhu převládán Scrumem (jako ostatně většina známých agilních rámců). Cílem práce je definování a charakteristika škálovatelného rámce DSDM a identifikace projektů, pro které je vhodné jeho použití.

## Popis rámce

- Název: DSDM Agile Project Framework
- Základní zdroje: DSDM Agile Project Framework Handbook
- Web rámce: <https://www.agilebusiness.org/page/TheDSDMAgileProjectFramework>
- Rok vzniku: 1994
- Autoři: nezisková organizace *Agile Business Consortium*
- Kategorizace dle Horlach et al.: Organization Focus - Inter-Team focus

## Charakteristika

*DSDM Agile Project Framework* je agilním škálovatelným rámcem. Jedná se o novou verzi původního DSDM Atern®. Původně se jednalo pouze o metodu vývoje softwaru, později byl ale přepracován a v současné podobě je vhodný pro projektový management a dodávání řešení obecně, není tedy už omezen pouze na vývoj SW, resp. IS/ICT projekty, ale je možné jej použít i v jiných odvětvích. Proto také došlo ke změně názvu z původního *Dynamic System Development Method* na aktuální *Driving Strategy, Delivering More* (můžeme interpretovat tak, že s se správnou údernou strategií, doručíme více). (Clifton, 2003)

Pomocí tohoto rámce můžeme řídit celý životní cyklus projektu. Jedná se ale o projektovou, organizační a strategickou úroveň, ne o technickou. Proto bývá často kombinován s jinými rámci, které se více na technickou část zaměřují jako je *Extreme Programming* a jiné. (Davis, 2020)

Oproti ostatním agilním rámcům se odlišuje hlavně pevně stanovenými základy řízení a také dohledem, pomocí prioritizačního nástroje MoSCow apod. Obecně se dá říct, že v porovnání např. se Scrumem jakožto lídrem mezi agilními rámci má DSDM definováno více principů, rolí a metodických postupů. Což může být i výhoda i nevýhoda, záleží projekt od projektu. (Clifton, 2003)

## Historie

V 90. letech se začal rozmáhat přístup RAD (*Rapid Application Development*), který v některých oblastech nahradil tradiční vodopádový přístup vývoje, který, jak dnes již víme, se nehodí na rychle se měnící typy projektů. RAD ale oproti tomu zas ztrácí ve striktnějším definování alespoň základních procesů. DSDM tedy vznikl jako reakce na oba tyto dominantní přístupy. Ne tak pomalý, neohrabaný a striktně procesní jako "vodopád", ale zároveň učesanější a disciplinovanější než RAD. (VivifyScrum, 2019)

Vznik DSDM se tedy datuje do roku 1994, kdy bylo založeno DSDM konsorcium, což byla asociace expertů a společností v oblasti SW inženýrství, s cílem vymyšlení nového rámce na základě svých společných *best practise*. Později po zveřejnění agilního manifestu, byly do DSDM přidány klíčové praktiky v souladu s nově vzniklým agilním směrem metodik. (VivifyScrum, 2019)

Roku 2006 byla vydána verze 4.2 a poslední verze rámce byla vydána roku 2014 pod aktuálním názvem *The DSDM Agile Project Framework*. Rámec je veřejně dostupný. Roku 2016 došlo k přejmenování DSDM konsorcia na *Agile Business Consortium*. (Consortium, 2014)

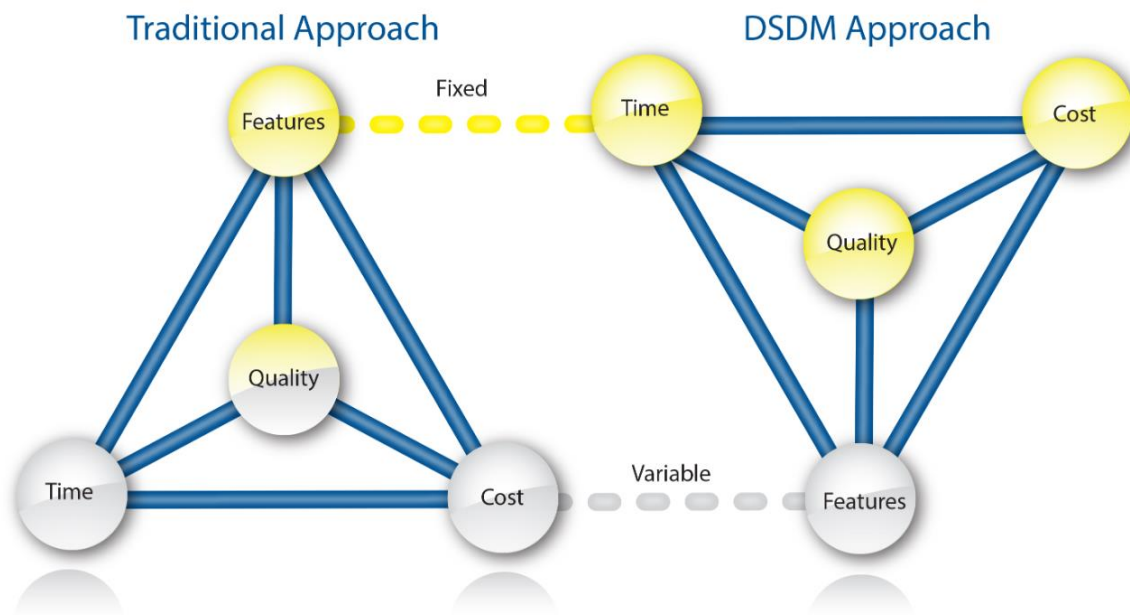
## Filozofický základ

Filozofií DSDM je: „*Nejlepší byznysové hodnoty je dosaženo, když jsou projekty směřovány ke splnění byznysových cílů, dodávány často a je zahrnuta spolupráce motivovaného týmu.*“

K dosažení této filozofie je potřeba aby všichni stakeholdeři:

- porozuměli byznys vizi a cílům,
- byli podporováni k rozhodování v rámci své oblasti expertizy,
- spolupracovali za účelem dodání odpovídajícího byznys řešení a aby dodání bylo včasné a v souladu s definovanými prioritami,
- akceptovali že změna je nezbytná pro porozumění řešení v průběhu řešení.

Dalším klíčovým bodem je struktura proměnných projektu. Tento model vychází z agilních metodik obecně, není nikterak specifický pro DSDM. Jedná se konkrétně o to, že v rámci projektu/dodávaného řešení chceme mít kvalitu, náklady a čas fixně stanoveny. Není možné tedy na problémy při vývoji, zdržení apod. reagovat tím, že bychom posunuli termín doručení nebo tak že bychom pro nahnání ztráty vyvinuli nějakou komponentu méně kvalitně. To, čím hýbeme jsou vždy vyvíjené funkcionality, kdy je můžeme odebírat nebo přidávat na základě aktuálního stavu projektu, samozřejmě dle byznysových priorit a v souladu s přáním klienta. Toto je jeden z důležitých bodů, kde se agilní metodiky liší od tradičního/vodopádového modelu. A pro rámec DSDM konkrétně je toto kritický pilíř. Schéma a porovnání můžete vidět na [Obr. 1](#). (Consortium, 2014)



Obr. 1- porovnání tradičního a DSDM přístupu k proměnným projektu

Filozofie je dále podporována osmi principy, na kterých celý rámec stojí a které jsou pro jeho úspěch kritické. (Consortium, 2014)

## Principy

Na základě filozofie, kterou se škálovatelný rámec řídí, bylo vytvořeno 8 principů, které tuto filozofii podporují. Do určité míry také zastupují agilní hodnoty, na kterých je DSDM založeno. Pokud společnost nebude všechny principy dodržovat, nebude možné využít (v plné míře) výhod, které rámec DSDM nabízí. Práce v souladu se všemi principy naopak umožňuje společnosti doručovat za pomoci týmové spolupráce co nejlepší řešení. (Consortium, 2014)

### 1. Zaměření na požadavky byznysu

U všech rozhodnutí v rámci projektu, musíme mít na paměti, jejich dopad na byznysové požadavky. Je nutné mít stále na paměti, že námi vytvářené řešení je pouze prostředek k vytvoření hodnoty pro zákazníka, k uspokojení jeho potřeb. Proto by nikdy neměla žádná součást projektu v něm být pouze na základě rozhodnutí interního týmu řešitelské společnosti. V rámci principu je nutné si uvědomit byznysové priority a také mít vždy na paměti doručení alespoň minimálního funkčního řešení. K dodržení tohoto základního principu nám pomáhá členění do rolí a jasné vymezení jejich zodpovědností a také některé nástroje (viz MoSCoW prioritizace) (Consortium, 2014)

### 2. Včasné doručení

Doručit projekt/řešení včas. To bývá často jednou z nejdůležitějších podmínek, abychom vůbec projekt mohli prohlásit za úspěšný. Dává nám mj. pomyslnou výhodu před konkurencí a také se díky tomu v některých případech můžeme vyhnout nedodržení některých právních norem.

Z těchto důvodů je nutné umět projekt správně řídit, aby se nám náhodou překročení maximální doby trvání nestalo. V tom nám může pomoci *timeboxing*, a také prioritizace. Včasným doručením také získáváme důvěru zákazníků a jakousi předvídatelnost. Jen díky včasnému doručení všech dílčích celků se nemusíme obávat nedodržení termínu finálního produktu (samozřejmě mohou nastat odlišné incidenty). (Consortium, 2014)

### 3. Spolupráce

Na spolupráci je kladen velký důraz. Týmový duch je i v DSDM klíčem k úspěchu a je určitě preferován před individualistickým přístupem, ať už se jedná o sebelepší jedince. Kolaborace podporuje vyšší míru sdíleného vlastnictví produktu. Je důležité také v průběhu projektu zapojovat všechny stakeholdery. Měli bychom podporovat a povzbuzovat proaktivní přístup ze strany reprezentantů byznysu. Více o spolupráci na úrovni jednotlivých rolí v kapitole *Role a zodpovědnosti* (Consortium, 2014)

### 4. Kvalita nesmí být nikdy kompromitována

Zde se jedná o základní princip všech agilních rámců. Kvalita řešení by nikdy neměla být proměnnou (spolu s časem dodání) v žádném našem projektu. Úroveň kvality by měla být definována a napevno stanovena před začátkem samotného vývoje. Někdy není v lidských silách zcela bezchybná kvalita, v tom případě je nutné se uchýlit ke konceptu GEQ (*Good Enough Quality*), tedy kvalita „pouze“ dostatečná. I tak ale musíme vždy projít pevně stanovenými akceptačními kritérii. (Consortium, 2014)

Pro dosažení požadované úrovně kvality je nutné dbát také na kvalitní testování. Proces testování by měl probíhat kontinuálně v průběhu vývoje. Kvalita se ale také vztahuje na dokumentaci a design a obecně i veškeré nefunkční požadavky, na které se nesmí zapomenout. (Consortium, 2014)

### 5. Budujte inkrementálně od základů

Jednou z klíčových odlišností oproti ostatním agilním rámcům je kladení důrazu na stanovení jasných „základů“ ještě před začátkem samotného vývoje. Je nutné, aby všichni znali náležitosti projektu a jeho přesný rozsah (v rámci celého projektu samozřejmě na vyšší úrovni, u konkrétních iterací chceme jít více do hloubky). Analýza ale nikdy nesmí být přílišná. (Consortium, 2014)

Po ustanovení pevných základů probíhá vývoj inkrementálně, snažíme se vytvořit alespoň nějakou byznysovou hodnotu co nejdříve je to možné, tu poté rozšiřovat a nabalovat na ni další podrobnější a složitější komponenty. Díky tomu získáváme brzy, a hlavně průběžně, zpětnou vazbu z byznysového hlediska. Dobrou praktikou je také revaluace požadavků po každém doručeném inkrementu. (Consortium, 2014)

### 6. Vyvíjejte iterativně

Pro DSDM je typická kombinace iterativní vývoje, častých demonstrací pro byznys a retrospektiv po každé iteraci, pro zaručení včasné zpětné vazby. Musím být kladen důraz na přístupnost změnám. Málokdy se povede ideální řešení na první pokus. Proto je důležité obdržet zpětnou vazbu po každé iteraci. Uvědomit si, že správné řešení se nenajde bez experimentování a dostatečné kreativity, k tomu by měli všichni být vedeni. Změna je žádoucí a snažme se využívat

její potenciál. Vždy je třeba mít na paměti, abychom vyvíjeli opravdu to, co byznys potřebuje. (Consortium, 2014)

#### 7. Jasná a kontinuální komunikace

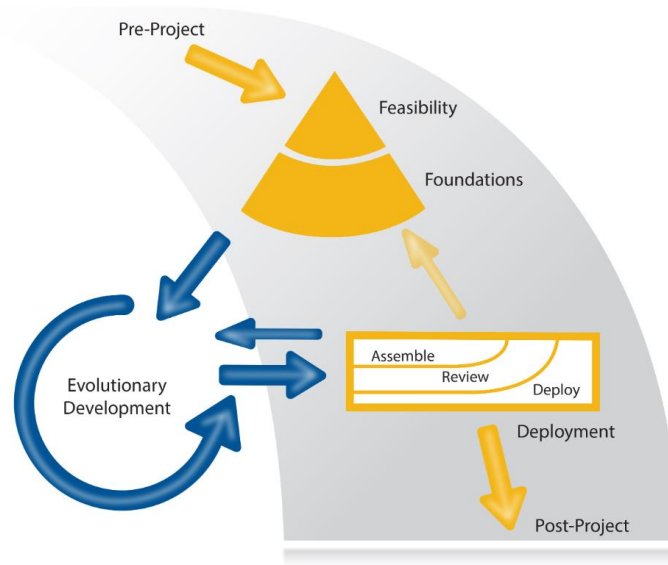
Špatná komunikace je někdy uváděna jako největší problém selhání projektů. DSDM zapojuje více praktik, abychom tuto hrozbu neutralizovali. Upřednostněna by měla být neformální komunikace z očí do očí. Dále denní *standupy* pro synchronizaci v rámci týmu. Každý člen stručně vysvětlí, co minulý den dělal, na čem aktuálně pracuje a k čemu se dostane po úspěšném dokončení aktuálně rozpracovaného úkolu. Důležité je také použití vizualizačních technik jako modelování a prototypování (viz X). Dokumentovat musíme stručně, často a jasně. Měli bychom pracovat vždy s očekáváními klienta/byznysu. V neposlední řadě je čestnost a transparence naší komunikace. (Consortium, 2014)

#### 8. Demonstrace kontroly

Při řízení projektu je nutné nad ním mít kontrolu v celém jeho průběhu také to umět v případě potřeby prokázat. Toho docílíme jasným naplánováním. Zde jsou zodpovědnými rolemi zejména team leader a projektový manažer. Plány a aktuální stav jejich naplnění by měl být dostupný všem zaměstnancům. Řízení by mělo být spíše proaktivní, než diktátorské. Úspěch by měl být měřen na základě doručených produktů, ne pouze provedených činností. Kontinuální evaluace může pomoci včas zachytit odklon od vytyčených cílů. *Timeboxy*, retrospektivy a plánování by měly projektovému manažerovi a team leaderovi v tomto snažení pomoci. (Consortium, 2014)

### Životní cyklus a proces vývoje

Na rozdíl od většiny agilních rámců má DSDM pro projektový management a produktový vývoj jen jeden model. Procesní model se skládá ze šesti provázaných fází (před-projektová fáze, proveditelnost, základy, vývoj, nasazení a po-projektová fáze). V rámci těchto fází je definovaný ideální průběh, ale je možné v reakci na průběh projektu některé fáze zopakovat a v rámci modelu se vrátet zpět, pokud to situace žádá. Výsledný životní cyklus tedy nebude vždy stejný. Model procesu vývoje můžete vidět na Obr. 2. (Consortium, 2014)



Obr. 2 - model procesu vývoje DSDM

### Před-projektová fáze (Pre-Project)

Tato fáze stojí před samotným začátkem projektu. Je v ní nutné definovat jasně cíle projektu a také rozhodnout, zda bude vůbec projekt realizován (může se při evaluaci ukázat, že třeba nebude dostatečně ziskový apod.) (Consortium, 2014)

### Fáze proveditelnosti (Feasibility)

Fáze proveditelnosti slouží také k ověření, jestli má cenu na projektu začít pracovat, ale zde se jedná o technickou a byznysovou stránku věci. Nemělo by se zdá ztrácet příliš mnoho času. Ideálně by měla být jen tak krátká, aby se rozhodlo, jestli projekt bude realizován nebo se v této fázi zruší (nemáme dostatečné kapacity, expertizu pro realizaci, ...). (Consortium, 2014)

### Fáze základů (Foundations)

V této fázi už začínáme připravovat řešení, ale pouze na velmi vysoké úrovni abstrakce, nemělo by se jít příliš do detailů. Hlavní v této fázi je porozumění rozsahu projektu a určit zhruba jak, kým a za jakých okolností bude realizován. I u větších a komplexnějších projektů by tato fáze neměla přesáhnout pár týdnů. Je možné také rovnou naplánovat opakování fáze základů, pokud to produkt vyžaduje, po fázi nasazení. Někdy se tak stane i zcela neplánovaně, a to, pokud se nasazení nepovede tak, jak se očekávalo. V rámci této fáze by se tedy celý životní cyklus měl odsouhlasit. (Consortium, 2014)

### Fáze vývoje (Evolutionary Development)

Zde dochází k vývoji samotné řešení. Ten probíhá iterativně v jednotlivých *timeboxech*. Řešitelský tým vytváří jednotlivé inkrementy řešení. Dále zde určitě využijeme nástroje, jakými jsou modelování a prioritizace MoSCoW. Testování řešení by mělo probíhat kontinuálně v průběhu celé fáze. (Consortium, 2014)



## Fáze nasazení (Deployment)

V této fázi je hlavním úkolem uvést vyvíjené řešení do chodu, nasadit jej na produkční prostředí. Nasazení dělíme na tři menší pod-fáze: sestavení, zhodnocení, nasazení. Délka sestavovací fáze záleží na typu a komplexitě projektu, může být prakticky nulová, ale také se může jednat o sestavení aplikace, skládající se z mnoha komponent a používající desítky závislostí. To je poté dlouhý a nelehký proces. V rámci zhodnocení je na pořadu finální evaluace, zda jsme schopni řešení nasadit. Po nasazení je na řadě formální ukončení projektu. (Consortium, 2014)

## Po-projektová fáze (Post-Project)

Tato fáze nastává až po formálním ukončení. Kontrolujeme, do jaké míry se podařilo byznysové požadavky naplnit. Není samozřejmě možné evaluovat všechny benefity ihned po nasazení. Pro potvrzení některých může být nutné delší provoz řešení. Toto zhodnocení benefitů můžeme dělat po každém vydání (budou tedy součástí ještě fáze předchozí) nebo po celkovém ukončení projektu. (Consortium, 2014)

## Role a zodpovědnosti

Kromě komunikace klade DSDM velký důraz na jasné rozdělení rolí a zodpovědností z nich vyplývajících. V duchu všech agilních metodik je zde dodržován princip samo-organizace pro co nejlepší dodávané výsledky. (Consortium, 2014)

Je definované značné množství rolí (v porovnání s jinými agilními metodikami). Tyto jsou strukturovány do čtyř skupin, které reprezentují různé pohledy na projekt: byznysový, technický/řešitelský, manažerský a procesní. (Consortium, 2014)

Dále jsou role strukturovány do tří úrovní, reprezentujících, kde v projektu dané role figurují. Celou hierarchii znázorňuje Obr. 3.

### Byznys sponzor

Nejzkušenější byznysovou rolí na projektové úrovni by měl být právě byznys sponzor. Měl by být dostatečně vysoko v podnikové hierarchii, aby mohl rychle řešit vzniklé problémy a rozhodovat o rozpočtu pro projekt. Měl by být dostupný v průběhu celého projektu. Na menších projektech by se mělo jednat o jednu konkrétní osobu, ale u větších projektů může být role zastoupena více osobami, např. zvolenou komisí. (Consortium, 2014)

### Zodpovědnosti

- Vlastní *Business Case* projektu.
- Zajišťuje soulad projektu s *BC*.
- Rozhoduje o rozpočtu.
- Musí efektivně a rychle řídit eskalaci vzniklých problémů.
- Musí se účastnit demonstrací na konci každého *timeboxu*.

### Byznys Vizionář

Za vizi a její udržení v průběhu projektu je zodpovědný byznys vizionář. Z důvodu případných odklonů a zmatků, by měla být tato role zastupována pouze jedním člověkem. Jeho úkolem je dále tlumočení potřeb byznys sponzora ostatním členům týmu. Měl by také být přítomný

v průběhu celého projektu a usměrňovat jej. Po jeho ukončení je zodpovědný za nasazené řešení a také za patřičné zžitkování všech jeho výhod. (Consortium, 2014)

#### Zodpovědnosti

- Vytváří byznysovou vizi projektu a její komunikace v rámci týmu.
- Definuje a schvaluje byznys změny v průběhu projektu.
- Zajišťuje kooperaci všech *stakeholderů* v rámci rozsahu projektu.

#### Technický koordinátor

Tato role zajišťuje technickou kvalitu projektu. Dohlíží také na to, aby byla technická stránka projektu konzistentní a v případě problémů technického rázu je musí řešit. Má na starost také technické inovace. Zjednodušeně se dá říct, že to, co zajišťuje u projektu z hlediska byznysu byznys vizionář, tak to zajišťuje technický koordinátor po stránce technické. (Consortium, 2014)

#### Zodpovědnosti

- Kontroluje technologickou architekturu.
- Určuje technická prostředí.
- Koordinuje technické aktivity v rámci týmu.
- Zodpovídá za nefunkční požadavky projektu.
- S pomocí byznys analytika mění byznysové požadavky na jasnější technická řešení na nižší úrovni.

#### Projektový manažer

Projektový manažer koordinuje veškeré součásti projektu (na nejvyšší úrovni), ale měl by u toho dodržovat veškeré DSDM koncepty. Tedy je spíše stranou a snaží se členy týmu více povzbuzovat a pouze usměrňovat, kdyby se příliš odklonili od cíle. Neměl by rozkazovat a pouze striktně zadávat úkoly. Někdy mohou být na této pozici dva jedinci, často se tak děje u outsourcovaných projektů, kdy jeden zaopatroje vše na straně zákazníka a druhý na straně dodávaného řešení. (Consortium, 2014)

#### Zodpovědnosti

- Zajišťuje včasnou a efektivní komunikaci jak uvnitř týmu, tak i šíření informací ke všem zainteresovaným stakeholderům mimo něj.
- Plánuje projekt a jeho rozložení v čase, ale ne na úrovni *timeboxů*.
- Kontroluje postup projektem dle plánu doručení.
- Motivuje členy týmu.
- Monitoruje a zajišťuje efektivní komunikaci mezi členy řešitelského týmu.
- Řeší eskalované problémy od řešitelského týmu .

#### Byznys analytik

Byznys analytik slouží jako spojka mezi řešitelským týmem a projektovou úrovní rolí. Nesmí ale pouze komunikaci zprostředkovávat, ale měl by zajistit, aby spolu subjekty komunikovali navzájem na denní bázi při veškerých záležitostech týkajících se vyvíjeného řešení. (Consortium, 2014)



### Zodpovědnosti

- Asistuje byznys vizionáři při formulaci byznys vize.
- Identifikuje rizika, příležitosti a dopad celého řešení.
- Pomáhá na projektové úrovni formulovat byznys případ.
- Kontroluje kvalitu a formulaci požadavků a také jejich správnou interpretaci v řešitelském týmu.
- Dohlíží, aby seznam priorit byl vždy aktuální a aby byl dodržován.
- Kontroluje dopad byznys rozhodnutí s ohledem na samotný projekt.
- Zodpovídá za dodržení požadavků ve fázi akceptace.

### Team leader

Team leader zodpovídá za řešitelský tým a veškeré jeho fungování. Zajišťuje jeho koherenci. Také pracuje přímo s týmem a diriguje ho na úrovni *timeboxů*. V ideálním případě by se mělo jednat o člena týmu, který zároveň zastává nějakou jinou roli v řešitelském týmu a měl by být zvolen ostatními členy. Je možné tedy tuto roli střídát na bázi jednotlivých iterací. (Consortium, 2014)

### Zodpovědnosti

- Zajišťuje snažení řešitelského týmu směrem ke konečnému produktu.
- Vede, povzbuzuje a podporuje řešitelský tým.
- Zodpovídá za včasné provedení veškerého testování a retrospektiv.
- Případné problémy postupuje výše projektovému manažerovi, byznys vizionáři nebo technickému koordinátorovi dle povahy problému.
- Pořádá denní *standupy*.

### Byznys ambasador

Tato role je zástupcem byznysového pohledu uvnitř řešitelského týmu. V úvodní fázi je důležitý při stanovení požadavků a při jejich prioritizaci. V průběhu vývoje poté detailně kontroluje denní přírůstky s ohledem na naplnění byznysových cílů. Při samotném vývoji je také zodpovědný za řešení byznysových otázek, je tedy důležité, aby měl v tomto ohledu dostatečné zkušenosti a respekt a důvěru ostatních spolupracovníků na byznys úrovni. Standardně se nejedná o pozici na plný úvazek a ambasador si tedy vyčleňuje pouze určitý čas na daný projekt. (Consortium, 2014)

### Zodpovědnosti

- Přispívá k tvorbě požadavků, vzhledu a retrospektivním sezením.
- Pomáhá řešit každodenní problémy z byznysového pohledu.
- Každodenní zajištění správného vývoje projektu.
- Organizuje a kontroluje akceptační testování.
- Zodpovídá za dokumentaci.

### Vývojář řešení

Vývojář řešení spolupracuje s ostatními členy týmu a s jejich pomocí přetváří byznysové požadavky na řešení, v konkrétní iteraci vývoje. (Consortium, 2014)

### Zodpovědnosti

- Iterativně vyvíjí řešení projektu, modely potřebné pro kontrolu vývoje a modely a dokumentaci nutnou k podpoře nasazeného řešení.
- Testuje výstupy své práce (izolovaně).
- Dodržuje technické standardy a *best practice*.

### Tester řešení

Hlavní úlohou testera je testování celého řešení v průběhu projektu. Činí tak na základě domluvené testovací strategie. (Consortium, 2014)

### Zodpovědnosti

- Na základě vstupů od byznys rolí vytváří testovací scénáře a testovací případy.
- Testuje řešení jako celek po technické stránce.
- Pomáhá utvářet akceptační kritéria.
- Reportuje výsledky testů technickému koordinátorovi a informuje o nich team leadera.

### Byznys poradce

Byznys poradce má expertní znalosti v dané oblasti. Stojí mimo řešitelský tým a pomáhá jeho členům s vývojem a testováním. (Consortium, 2014)

### Zodpovědnosti

- Poskytuje rady ohledně požadavků, návrhu a retrospektiv.
- Pomáhá s byznys scénáři pro definování testů.
- Pomáhá s dokumentací pro finální řešení.

### Technický poradce

Obdobou byznys poradce, akorát v technické sféře je poradce technický. Pomáhá řešitelům s technickými problémy při každodenní práci, zejména v oblastech managementu změn, operativní podpory, údržby řešení apod. (Consortium, 2014)

### Zodpovědnosti

- Poskytuje rady v oblasti požadavků, návrhu a retrospektiv.
- Řeší otázky operativy nebo podpory za účelem vytvoření testovací strategie.
- Zajišťuje, že se vývoj řešení ubírá správným směrem.
- Pomáhá s akceptačním testováním.

### Facilitátor workshopu

Tato role patří do skupiny podpůrných a nemá vazbu přímo na řešitelský tým. Facilitátor je zodpovědný za workshopy, což jsou v DSDM nezávislá jednání, s přesně zadaným tématem a zvolenou úzkou skupinou osob, které se ho mají zúčastnit. Facilitátor poté figuruje jako „moderátor“ diskuze a pomáhá zúčastněným dospět k vytyčenému cíli, ať už je to rozhodnutí ohledně rozpočtu, technického řešení nebo čehokoliv v rámci projektu. (Consortium, 2014)

### Zodpovědnosti

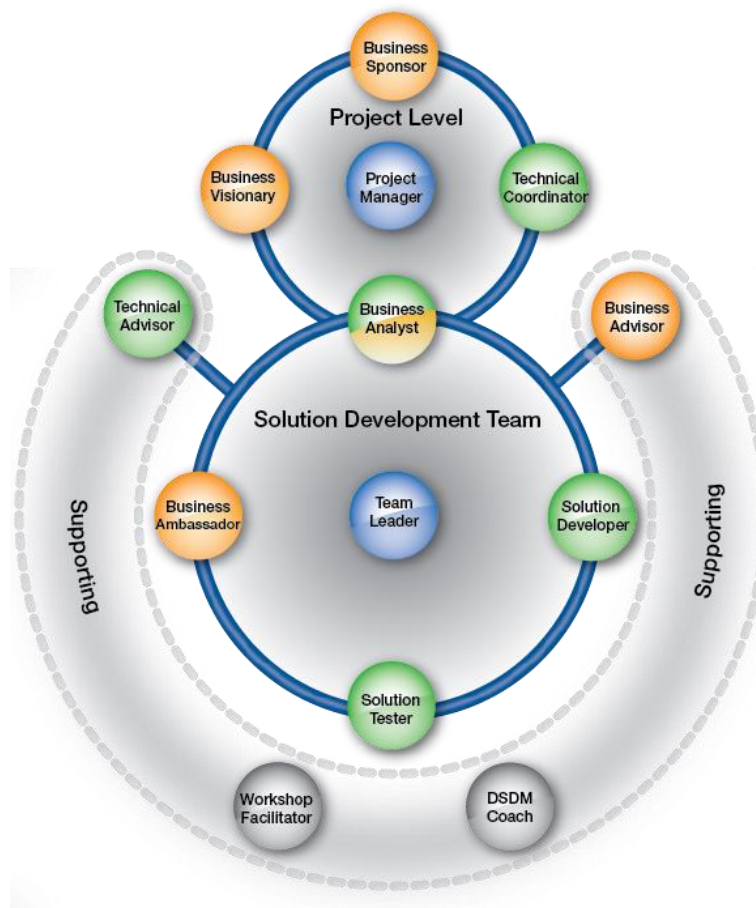
- Zaopatřuje a organizuje workshopy.
- Zajišťuje, aby se workshop dobral ke svému cíli.
- Je zodpovědný za předání výsledku workshopu rolím, které mají být informovány.

### DSDM kouč

DSDM kouč pomáhá ostatním členům v rámci týmu v oblastech dodržování rámce. Měl by na tuto pozici být certifikován. (Consortium, 2014)

### Zodpovědnosti

- Poskytuje všem detailní informace o rámci.
- V případě problému s nasazením konkrétní DSDM techniky v projektu by měl dotyčným pomoci, buďto upravit požadavek nebo přizpůsobit DSDM techniku, aby se dala použít. (Consortium, 2014)



Obr. 3 - role v DSDM a jejich vazby a úrovně

## Škálovatelnost rámce

DSDM je jakožto škálovatelný rámec možno upravovat dle potřeb konkrétního projektu.

Škálovat můžeme zejména týmy a jednotlivé role v nich. Pokud máme projekt menšího rozsahu bude nám pravděpodobně stačit pouze jeden tým, jedna osoba může zastávat více rolí a některé role také můžeme vynechat. Naopak u rozsáhlých komplexních projektů budeme mít na některých rolích více než jednu osobu, ale zároveň týmů jako na *Obr. 3* můžeme mít současně více než jen jeden. (Consortium, 2014)

Další s čím můžeme pohybovat dle potřeby je samotný životní cyklus, který jak už bylo zmiňováno nemusí být reprezentován přesně jako v oficiálním modelu, ale můžeme si ho do jisté míry upravit ke svým vlastním potřebám a například přidat nasazení do každé iterace nebo volit nelineární průchod. (Consortium, 2014)

## MoSCoW prioritizace

Z důvodu nutnosti dodržení stanovených termínů je důležité umět prioritizovat práci a jednotlivé úkoly. V DSDM k tomu slouží prioritizační technika nazvaná MoSCoW. Nejčastěji je používána na požadavky nebo *User Stories*, ale dá se využít také pro testy, produkty, akceptační kritéria nebo *use case*. Tuto techniku vymyslel v roce 1994 zaměstnanec společnosti Oracle Dai Clegg a věnoval ji rámci DSDM. Jejím účelem je pochopení a stanovení priorit, konkrétně ve 4 možných stupních:

- M – Must have,
- S – Should have,
- C – Could have,
- W – Won't have this time. (Consortium, 2014)

*Must have* označuje požadavek, který musí být dodán bezpodmínečně za každou cenu. Typicky vyvíjené řešení by bez těchto požadavků by vůbec nefungovalo, nedávalo by smysl, a proto by ho nemělo cenu doručovat. Pokud je ale možné, aby projekt bez této funkcionality fungoval (i kdyby to mělo být za cenu pracovního odstranění vzniklých potíží), měl by být spíše v druhé kategorii priorit. (Barbir, 2018) (Consortium, 2014)

Tou je kategorie *Should have*, tedy požadavky, které by měly být splněny, ale nejsou pro projekt kriticky důležité. Nezáleží tedy, kolik problému jejich vynechání způsobí, ani jak těžké by bylo je odstranit, pokud je vynechat lze, neměli by být *Must have*. Nicméně je třeba brát zřetel na to, že toto stanovení priorit musí být bráno vždy z hlediska byznysu a požadavků zákazníka, ne z hlediska technického. Někdy se může jednat o funkcionality, bez které je systém/produkt plně funkční, ale pokud je označena zákazníkem za kriticky nutnou, musí s ní být dle toho zacházeno i při interním stanovení priorit. (Barbir, 2018)

Třetí kategorie je *Could have*. Sem patří požadavky, které by bylo pěkné splnit, ale jejich priorita je nejnižší. V porovnání s *Should have*, mají menší dopad při vynechání. Při odlišení těchto dvou kategorií je dobré evaluovat právě zmiňovaný dopad (přidaná hodnota pro byznys, počet ovlivněných uživatelů, ...), při nenaplnění požadavku. V obecné rovině jsou *Could have* požadavky naplněny pouze pokud je vývoj opravdu v tom nejlepším možném scénáři a vše

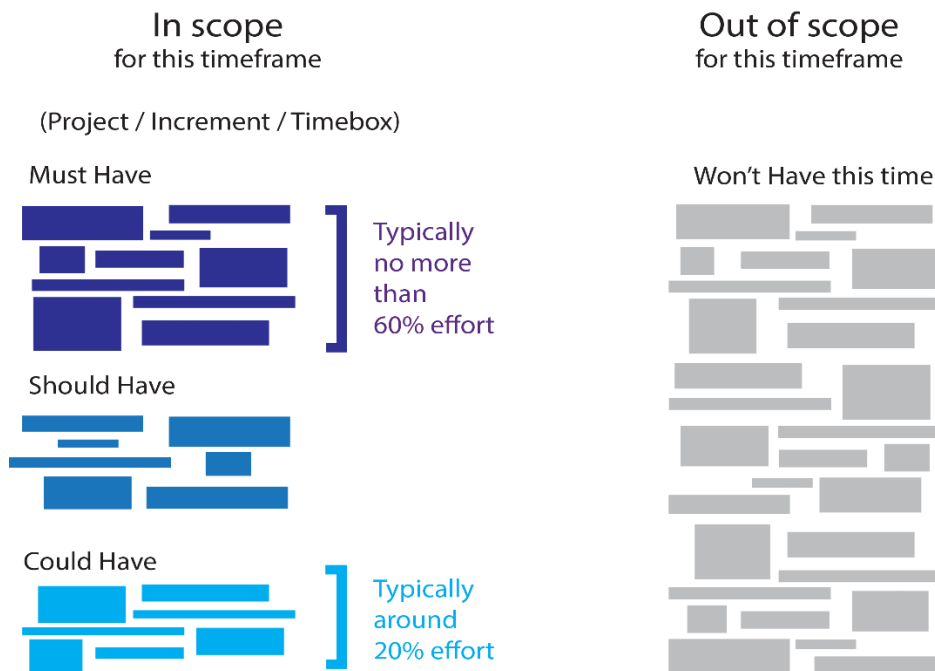
z prvních dvou úrovní se do dodání stíhá s předstihem. Naopak jsou prvním kandidáty na vyřazení v případě, kdy vše nejde tak hladce a podle plánu. (Barbir, 2018)

Poslední kategorií je *Won't have this time*. Pokud se bavíme o projektu jako celku, označení znamená, že tyto požadavky nebudou uspokojeny při vydání, ale mohou být přidány později formou opravy nebo v rozšiřující verzi. Když uvažujeme pouze inkrement vývoje, znamená toto označení, že požadavky nebudou uspokojeny pouze v tomto konkrétním cyklu. Jejich uvádění se může zdát na první pohled zmatečné, ale díky jejich určení se nám snáze udržuje daný rozsah projektu/iterace a přesně víme, co v něm není a s čím bychom neměli ztrácet čas. (Barbir, 2018)

Jak prioritizaci efektivně použít

Ačkoliv se přiřazení 4 písmen k požadavkům může zdát jako banální úkol, v praxi tomu tak není. Určení priorit je nesmírně důležité pro hladký průběh projektu. Jedním z doporučení je, že by *Must have* kategorie neměla tvořit více než 60 % celkově prováděného objemu práce. Důvodem je dostatek flexibility a také důvěra ve splnění projektu, která může být při překročení této hranice negativně ovlivněna, což může mít dopad i na pracovní výkonnost celého týmu. (Consortium, 2014)

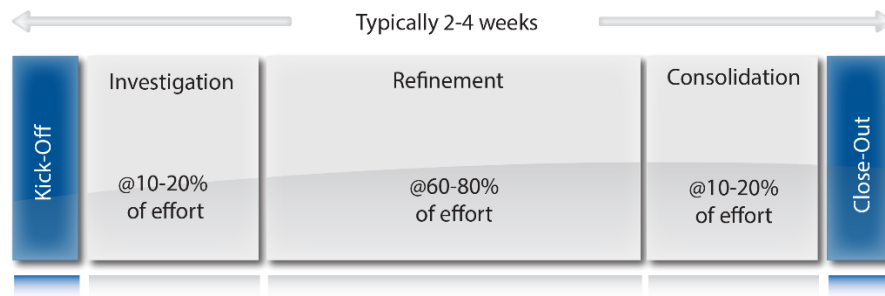
Dalším doporučením je držet *Could have* kategorii okolo 20 % objemu. Nemělo by smysl, mít zde objem práce větší, jelikož se jedná o ty nejpostradatelnější požadavky. Rozložení objemu můžete pro představu vidět na Obr. 4.



Obr. 4 - rozložení skupin prioritizace v rámci projektu

## Timeboxing

*Timebox* je v DSDM základní jednotkou iterace. Typicky trvá 2-4 týdny a skládá se ze tří oddělených částí: *investigace*, *refinementu* a *konsolidace*. Dále na začátku máme *Kick-off* poradů a na konci *Close-out* poradů. Jednotlivé procentuální rozdělení můžete vidět na Obr. 5.



Obr. 5- rozdělení dílčích částí v rámci timeboxu

*Kick-off* slouží primárně k seznámení řešitelského týmu s cílem daného *timeboxu* a odsouhlasení jeho realističnosti (porada by neměla trvat déle jak 3 hodiny).

Část investigace slouží k projití všech požadavků, které mají být doručeny. Také se rozhodne o akceptačních kritériích jednotlivých požadavků a také jak bude měřena úspěšnost daného *timeboxu*.

*Refinement* je část při které dochází k samotnému vývoji a testování řešení.

Konsolidace je závěrečnou částí, kdy dochází k formální kontrole dodaného řešení.

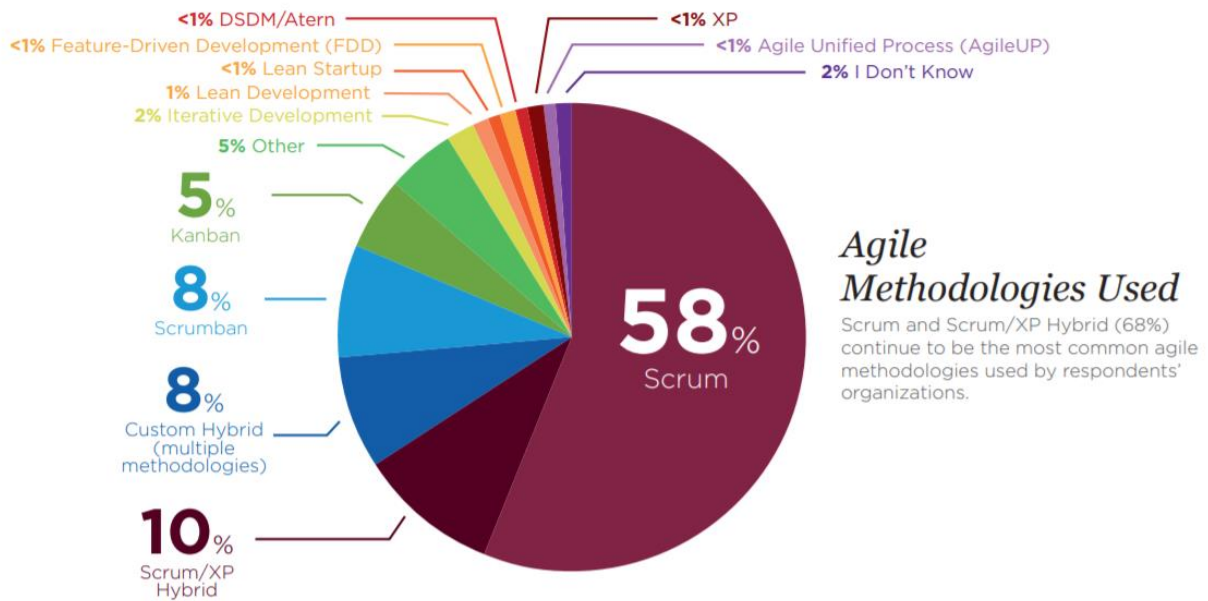
*Close-out* porada poté slouží k akceptaci řešení byznys vizionářem a technickým koordinátorem. (Consortium, 2014)

## Data z průzkumů

Jediná data o použití, která jsem našel, jsou informace o použití rámce DSDM v porovnání s ostatními agilními rámci na trhu. Jedná se o každoročně vydávaný *Annual State of Agile Report*. Bohužel ale byl DSDM zmíněn jako samostatná kategorie naposledy v roce 2017 s méně než jedním procentem podílu trhu (ještě rok předtím to bylo alespoň to celé procento) viz [Obr. 6](#). Později toto číslo asi ještě více klesalo a přestal být zobrazován samostatně, ale jeho tržní podíl je jen součástí kategorie ostatní.



# AGILE METHODS AND PRACTICES



Obr. 6 - tržní podíly agilních rámců

## Případové studie o použití rámce

Podařilo se mi bohužel najít pouze jednu případovou studii o použití, a to konkrétně na *e-Commerce* pro prodej zemědělský produktů. Z informací ve studii ani nebyl jednoduché pochopit, do jaké míry byl vlastně rámec při vývoji opravdu použit. (Mashal Alqudah, 2016)

## Použití pro projekty v praxi

Není tak jednoduché definovat, pro které projekty se rámec hodí a pro které ne. Flexibilita a škálovatelnost co se týče velikosti týmu je z mého pohledu lehce upravitelná jak směrem nahoru, tak dolů. Data z praxe bohužel ukazují, že význam rámce v době psaní této práce je téměř nulový a bez patřičného přepracování asi není reálné očekávat v blízké budoucnosti změny k lepšímu.

## Závěr

Rámec DSDM poskytuje zajímavou variantu agilního rámce. Snaží se do něj vnést alespoň nějaký řád. Možná ale právě to je jeho kamenem úrazu, jeho tržní podíl bude aktuálně hluboko pod jedním procentem. Charakteristika jakožto jeden z cílů práce ale byla bez problémů splněna. Valnou většinu jsem čerpal z oficiální příručky *DSDM Agile Project Framework Handbook* která je velmi přehledně zpracovaná a poskytuje podle mého názoru dostatečné informace pro zavedení rámce ve společnosti a celkem dobře také vyzdvihuje jeho kvality oproti konkurenci. Druhým dílčím cílem bylo identifikování projektů, pro které je vhodné DSDM použít bohužel

ale nebyl zcela naplněn. Důvodem je nedostatek získaných informací ohledně použití tohoto rámce v praxi.

Osobně se mi například líbí systém prioritizace a s velkou pravděpodobností se pokusím ho zavést u nás ve společnosti, konkrétně na produktové požadavky.

## Seznam použité literatury

Barbir, M., 2018. *MoSCoW Prioritization*, California: ProductPlan.

Clifton, M., 2003. *What Is DSDM?*, s.l.: Code Project.

Consortium, A. B., 2014. *Agile Business Consortium*, United Kingdom: s.n.

Davis, H., 2020. *DSDM: Agile for 'projects'*, s.l.: DeesonAgency.

Mashal Alqudah, R. R., 2016. A Review of Scaling Agile Methods in Large Software Development. p. 11.

VivifyScrum, 2019. *What is DSDM Agile Project Management?*, Novi Sad: VivifyScrum.

## Seznam obrázků

Obr. 1- porovnání tradičního a DSDM přístupu k proměnným projektu .....	3
Obr. 2 - model procesu vývoje DSDM .....	6
Obr. 3 - role v DSDM a jejich vazby a úrovně .....	11
Obr. 4 - rozložení skupin prioritizace v rámci projektu.....	13
Obr. 5- rozdělení dílčích částí v rámci timeboxu.....	14
Obr. 6 - tržní podíly agilních rámců.....	15

Zdroj Obr.1-5: DSDM Agile Project Framework Handbook

Zdroj Obr. 6: 11th Annual State of Agile Report