



Způsoby a techniky měření stavu a průběhu  
projektu za využití agilního přístupu k řízení  
projektů

Autoři:  
Martin Jirsa  
Štěpán Mazanec  
Pavel Gregor

## Obsah

1. Úvod .....	3
2. Definice pojmů .....	4
2.1. User Story .....	4
2.2. Story point .....	5
2.3. Velocity .....	6
3. Modelový příklad .....	7
4. Hockey stick chart .....	8
5. Lead&Cycle time .....	10
6. Burn down, burn up chart – Měření pokroku v projektu .....	11
6.1. Burn down a burn up chart na modelovém příkladu: .....	12
7. Product Backlog Burnup Chart .....	13
8. Závěr .....	14
9. Zdroje .....	15
10. Seznam obrázků .....	16
11. Seznam tabulek .....	16

# 1. Úvod

Cílem této seminární práce je seznámit čtenáře s technikami a přístupy, pomocí kterých lze měřit průběh a stav projektu při využití agilního přístupu pro řízení projektů. V první kapitole práce budou představeny pojmy, které jsou dále v práci diskutovány a je nezbytné, aby byl čtenář blíže seznámen s problematikou, do které tyto pojmy spadají. V druhé části práce bude čtenáři přiblížen charakter projektu a obsah prvního Sprintu, pomocí kterého prezentován nežádoucí výsledek měření, kdy Burn Up graf má tvar Hockey Stick. Dále budou diskutovány výsledky, které graf představuje a jejich negativní dopady na projekt. Ve třetí části práce již bude prezentován očekávaný průběh grafu, který představuje pozitivní průběh projektu a plnění úkolů v rámci druhého Sprintu. Současně bude představen Littleův zákon, pomocí kterého můžeme jednoduše stanovit doby sprintů a předcházet tak negativním výsledkům v podobě hockey stick grafu.

## 2. Definice pojmů

V článku jsou použity termíny, které nemusí každý ICT odborník znát, a vědět co přesně reprezentují. Jejich názvy uvedu v angličtině a vysvětlím jejich význam, případně uvedu příklad, kde se používají.

### 2.1. User Story

Termín se používá v softwarovém vývoji a v produktovém managementu víceméně jako neformální a představuje popis nějaké vlastnosti, funkcionality (feature) informačního systému nebo vyvíjeného systému. User stories jsou často vnímány z pohledu koncového uživatele nebo uživatele systému (IS). Často jsou zaznamenávány na poznámkové lístky nebo na tzv. „samolepící štítky“ nebo pomocí nějakého PR software. V závislosti na velikosti a náročnosti projektu mohou být user stories vytvářeny všemi zúčastněnými stranami a to například klienty, manažery nebo členy vývojového týmu. User stories pomáhají v komunikaci a vývojovým týmům usnadňují pochopit kontext ve vývoji produktu.

User stories mohou být zaměněny s požadavky systému. Požadavek je formálním popisem potřeby oproti tomu user story je neformální popis funkce.

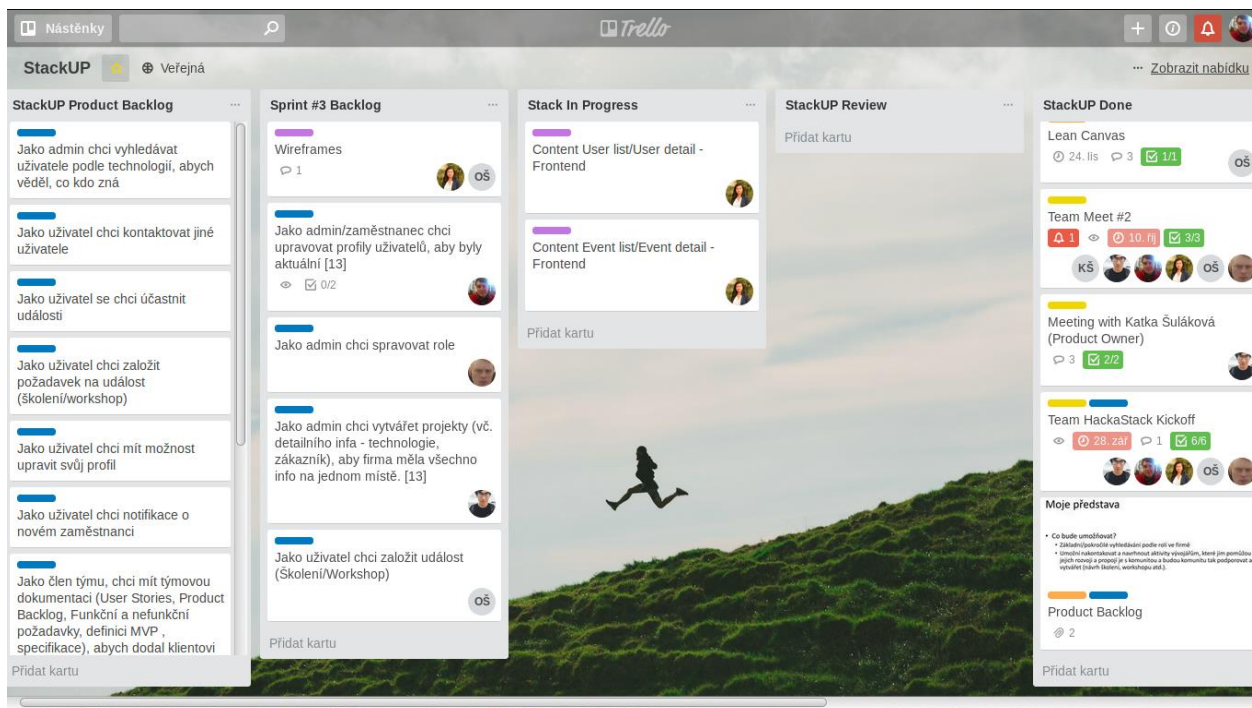
*Použití user stories v agilních metodikách [DVOŘÁK, 2010]*

Projekt je rozdělen na několik desítek částí nebo scénářů (user stories) a ke každé se přiřadí odhadovaná pracnost v podobě bodů (tzv. story points). Vývoj probíhá na základě krátkých časových úseků o stejné délce nazývaných iterace nebo též sprints. Rychlost postupu se pak měří počtem bodů, které je projektový tým schopen „spálit“ za jednu iteraci (kapacitu týmu). Na začátku každé iterace je plánovací porada, kde se analyzují zjištění a poznatky z poslední iterace a na základě změřené týmové kapacity se plánuje práce pro nadcházející iteraci. Výhodou tohoto postupu je minimální režie a schopnost rychlé reakce na změnu situace. Nevýhodou je vyšší riziko a nejistota ohledně celkového času i pracnosti.

*Proč a k čemu User Stories?*

Naši práci ve vývoji schvaluje zákazník a někdy také “Product Owner”, který zprostředkuje produkt zákazníkovi.

ProductOwner neřídí jednotlivé členy v týmu ani tým samotný. Nepřikazuje co se má provést a co se musí dokončit, ale poukazuje na vlastnosti daného software, které se musí dokončit a na základě jaké priority. Především jeho práce spočívá v zprostředkování produktu zákazníkovi, jasné definice cíle, proč to tak děláme a jakým způsobem [AGILE ALLIANCE, 2018].



Obrázek 1- Příklad použití user stories v nástroji Trello

## 2.2. Story point

Příběhové body jsou měrnou jednotkou pro vyjádření odhadu celkového úsilí, které bude zapotřebí k plné implementaci položky s nevyužitým produktem nebo jakéhokoli jiného díla [COHN, 2016].

Když odhadujeme body příběhu, přidělíme každé položce bodovou hodnotu. Hrubé hodnoty, které přidělujeme, jsou nedůležité. Důležité jsou relativní hodnoty. Příběh, který je přiřazen 2, by měl být dvakrát tak velký jako příběh, který je přiřazen 1. Měli by také být dvě třetiny příběhu, který se odhaduje na 3 příběhy.

Co je zahrnuto do Story points?

Protože příběhové body představují snahu vyvíjet „příběh“, odhad týmu musí obsahovat vše, co může ovlivnit úsilí vývoje. To by může zahrnovat:

- Množství práce, kterou je třeba udělat
- Složitost práce
- Jakékoli riziko nebo nejistota při práci

Při odhadu s příběhovými body se ujistěte, že zvažíte každý z těchto faktorů. Co znamenají Body příběhu?

Příběhové body představují úsilí, které je zapotřebí k tomu, aby byla položka PBI (Product Backlog Item) plná. Každý bod příběhu představuje normální rozložení času.

Například: 1 Story Point může představovat rozsah 4-12 hodin, 2 body příběhu 10-20 hodin a tak dále. Tato distribuce času je během odhadu neznámá. Použitím relativních PBI relativních, na které lze odhadnout, není nutné vědět, kolik času trvá. Jen chcete mít hrubý údaj o tom, kolik času potřebujete k dokončení. Jaké jsou výhody používání bodů Story?

Příběhové body umožňují týmu:

Rychle odhadněte problémy. Odhad je relativní vzhledem k již dokončeným položkám produktu s nevyřízenými položkami. To je rychlejší než odhad bez jakéhokoli odkazu. Odhadněte bez závazku. Při odhadu v hodinách se přesně zavazujete. Odhad příjmů ve story points zabraňuje tomu, aby byl závazek přesný. Nikdo přesně neví, kolik hodin jste vymezil na konkrétní problém.

Přijmout nejistotu, která přichází s odhadem. Příběhové body specifikují neznámý časový rozsah. Výběrem ze specifické sekvence Bodů bodů Fibonacci umožňuje zachytit nejistotu. Přesně tak, abyste plánovali sprinty dopředu. To umožňuje lépe zvládnout časové očekávání zúčastněných stran pro budoucí práci [DALMIJN, 2017].

## 2.3. Velocity

Pojem, který je často užíván jako „velocity“, zde nereprezentuje „rychlost“, ale měření, chcete-li, metriku. Na konci každé iterace tým přidá odhady své práce, které jsou spojené s „user stories“ jež byly dokončeny během této iterace. Toto celé se nazývá „velocity“. Pokud známe rychlost, tým může vypočítat, nebo revidovat odhad, jak dlouho bude trvat dokončit projekt, a to na základě odhadů spojených se zbývajících user stories za předpokladu, že rychlost zbývajících iterací bude přibližně stejná. To je obecně přesná predikce, i když málokdy přesná.

### *Očekávané přínosy*

Příklad: agilní tým začal pracovat na iteraci, plánuje dokončit příběhy A a B, odhadované na 2 body a příběh C, odhadovaný na 3 body. Na konci iterace jsou příběhy A a B 100% kompletní, ale C je dokončeno pouze 80%.

Agilní týmy obecně uznávají pouze dvě úrovně dokončení, 0% provedeno nebo 100% provedeno. Proto se C nepočítá k rychlosti a rychlost této iterace je 4 body.

Předpokládejme, že zbývající uživatelské příběhy představují celkem 40 bodů; týmová prognóza zbývajících úsilí pro projekt je pak 10 iterací.

Rychlost je také používána k omezení množství práce, která byla provedena v dalších iteracích. V našem příkladu by týmu bylo dobré doporučit v příští iteraci plánovat pouze příběhy o hodnotě 4 bodů. To nutně neznamená, že dokončí tolik ale ve skutečnosti dokončení příběhu C v příští iteraci může znamenat, že rychlost týmu bude naopak mnohem vyšší.

Agilní týmy považují oba druhy událostí za varovné znamení: nedokončují příběh nebo vidí svou rychlost "houpáním". Očekávanou reakcí je snažit se o jemnější rozložení příběhů.

Rychlost slouží jako regulační mechanismus několika způsoby [AGILE ALLIANCE, 2018].

## Možná úskalí

Výše uvedená definice rychlosti má několik dalších důsledků:

- rychlost je "měření" provedené po této skutečnosti; ačkoli to může napomoci plánovat dopředu, není samo o sobě rozpočet nebo prognóza, a fráze jako "nastavení rychlosti" odhalí základní nedorozumění
- rychlost je definována s ohledem na jednotkové hodnoty (story points) spíše než na jednotku úsilí (úlohy)
- pouze záleží na souhrnné rychlosti týmu a výraz "individuální rychlost" nemá smysl; tým je mechanismus určený k získání více než součtu jeho jednotlivých částí
- neexistuje žádné smysluplné srovnání rychlosti "mezi" různými týmy, neboť takové týmy mohou mít odlišné přístupy k odhadu, aby rychlost poskytovala předpovědi konce data projektu, je nezbytné, aby všechny uživatelské příběhy tvořící projekt byly odhadnuty konzistentním způsobem
- to lze dosáhnout jedním ze dvou hlavních způsobů:
  - odhadnout celou sadu uživatelských příběhů před zahájením projektu nebo brzy jako v prvních několika iteracích
  - použijte relativní odhad, abyste zajistili, že odhady provedené později budou shodné s odhady provedenými na začátku projektu.

## 3. Modelový příklad

V předchozí kapitole byly uvedeny a vysvětleny pojmy, které přímo souvisejí s technikami pro sledování průběhu projektu a pomocí kterých manažer získává podrobný přehled o stavu projektu. Aby v této práci mohly být tyto techniky názorně představeny, je zapotřebí pracovat s konkrétními projektovými daty. Pro účely této seminární práce, uvádíme modelový příklad, který nám poskytne potřebná data pro sestavení grafů, které jsou uvedeny v dalších kapitolách.

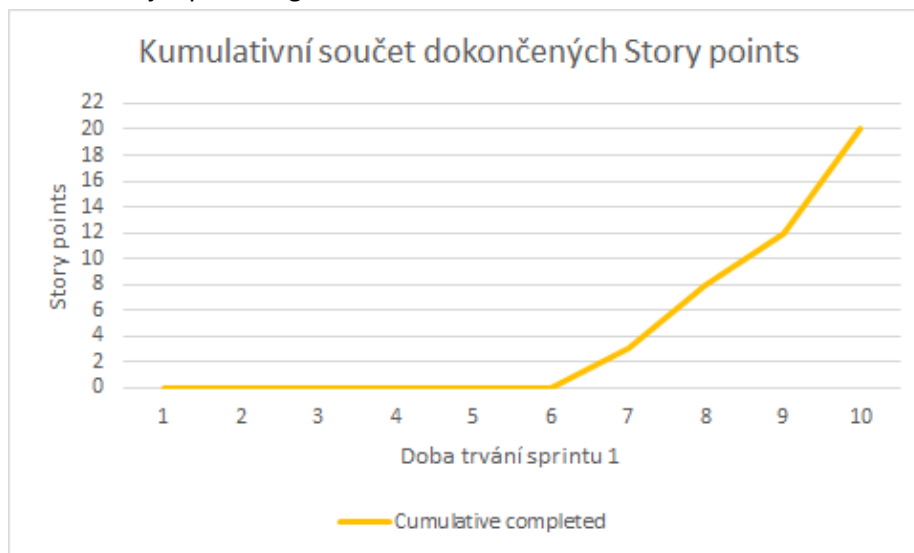
Firma zabývající se vývojem webových aplikací na míru, realizuje momentálně projekt, který dle projektového záměru má trvat 7 měsíců. Projekt již běží po dobu dvou měsíců a právě se připravuje backlog pro třetí sprint projektu. Každý sprint trvá 10 pracovních dnů a skládá se z několika stories, které představují funkcionalitu požadovanou zákazníkem a jsou ohodnoceny story pointy, dle jejich náročnosti na vývoj popřípadě priority. Tabulka 1 představuje obsah Sprintu č.1, který byl z hlediska dokončených funkcionalit neúspěšný, a který si rozebereme v následující kapitole.

Story	Odhad	Priorita
User_story_1	9	1
User_story_2	3	2
User_story_3	3	3
User_story_4	7	4
User_story_5	8	5
<b>Celkem:</b>	<b>30</b>	<b>X</b>

Tabulka 1 - Obsah sprintu 1 (zdroj:Autor)

## 4. Hockey stick chart

Jednou z možných technik měření průběhu projektu, je měření pomocí dokončených story points, respektivě dle jejich kumulativního součtu v průběhu daného sprintu. Pokud bychom graficky znázornili proběhlý sprint č. 1 pomocí story points a jejich kumulativní součet dokončených story points, získáme následující podobu grafu.



Obrázek 2 - Kumulativní součet dokončených Story points (zdroj:Autor)

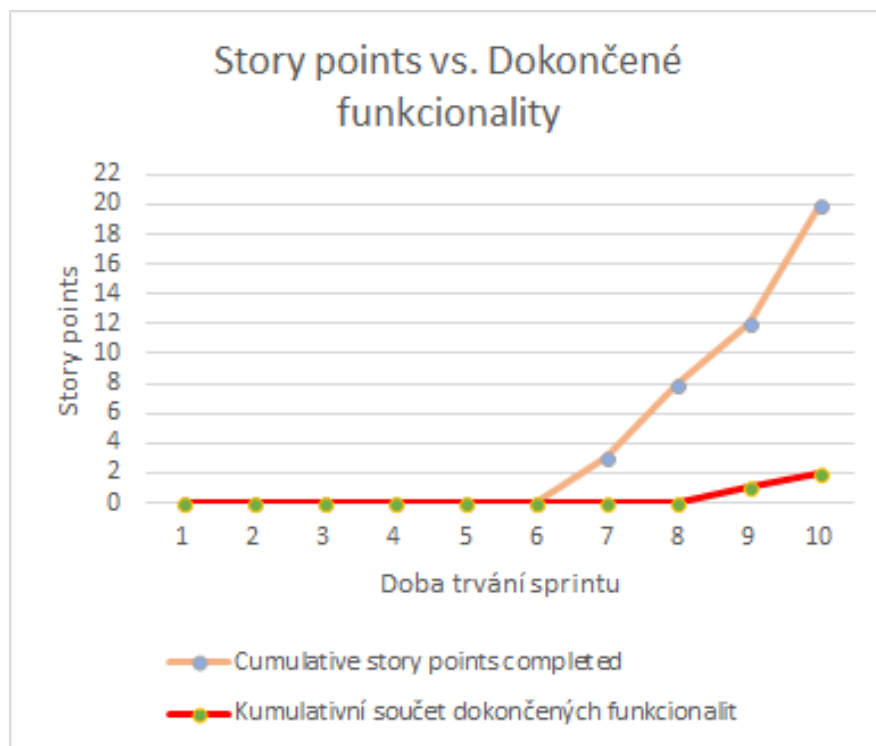
Takový to tvar grafu, respektivě Burn up chartu, který připomíná s trochou představivosti tvar hokejky, se nazývá hockey stick chart, jehož průběh nám v našem pojetí říká, že se úkoly dokončují na poslední chvíli a dochází tak k řadě nežádoucích efektů. Současně takový tvar obvykle vyznačuje nízkou kvalitou dodávaného produktu.

Vezme-li v úvahu skutečnost, že po skončení sprintu 1, byla řada úkolů ve stavu in-progress a sprint tak nebyl dokončen celý, takové výsledku mohou již předurčit zpoždění projektu z důvodu, že do dalšího sprintu se přebírají nedokončené user stories z předchozího sprintu. Z grafu můžeme dále vyčíst, že bylo dokončeno celkem 20 story points ze 30. Nejvyšší nárůst story points byl mezi 9. a 10. dnem sprintu, tedy k posledním dnům sprintu. Dalším znepokojivým ukazatelem může být, že první



dokončené story points jsou zaznamenány až od šestého dne projektu, tedy více než po uplynutí poloviny sprintu a do té doby nebylo dokončeno žádné user story [ROTHMAN,2017].

Díky předchozímu grafu víme, že práce na jednotlivých stories trvala příliš dlouho a nejvyššímu nárůstu dokončených story points docházelo ke konci sprintu, který tak nebyl dokončen v celém svém objemu. Abychom zjistili, poměr dokončených funkcionalit v rámci sprintu oproti kumulativnímu součtu story points, využijeme Burn Up chart, kde si tyto 2 měřené veličiny zobrazíme a porovnáme (graf. 2).



Obrázek 3 - Story points vs. dokončené funkcionality (zdroj:autor)

Nyní již získáváme ucelený pohled na problém Sprintu 1. Z grafu č.2 vidíme, že byly dokončeny pouze 2 funkcionality a to k samotnému konci projektu, zatímco v kumulativním součtu bylo dokončeno 20 story points z 30. Výsledkem těchto Burn Up chartů je zjištění, že aby byl býval Sprint č.1 dokončen v celém svém objemu, musely by být dokončené user stories ohodnoceny po 15. story pointech každý, případně by daná funkcionality musela být rozdělena na více user stories, který by tvořily backlog pro Sprint 1. Hlavním problémem Hockey Stick chartu je skutečnost, že představuje nedodané funkcionality na konci sprintu, což vede k nespokojenosti product ownera a hrozí případné eskalace na vedení podniku [ROTHMAN, 2017].

V předchozím textu jsme si popsali co Hockey stick představuje a co říká manažerovi, popřípadě celému vývojovému týmu. Nyní bych chtěl uvést několik rad a tipů, jak takové situaci předcházet, na co si dát pozor a jaké jsou důsledky hockey stick grafu pro vývojový tým. Níže uvedené body ve svém článku: "Burnup looking like a Hockey Stick?" popisuje autor SONI, 2016.

Jak předcházet Hockey stick chartu

1. Rozdělit velké user stories na více částí
2. Posílit tým, popřípadě sloučit týmy v rámci projektu
3. Hlídat a kontrolovat počet otevřených úkolů a úkolů ve stavu inprogress

#### 4. Jasně vymezit a definovat akceptační kritéria

Nežádoucí efekty Hockey stick chartu

1. Nespokojenost product ownera
2. Přepřacovanost
3. Pokles morálky
4. Nabalování práce do dalších sprintů
5. Zpoždění projektu

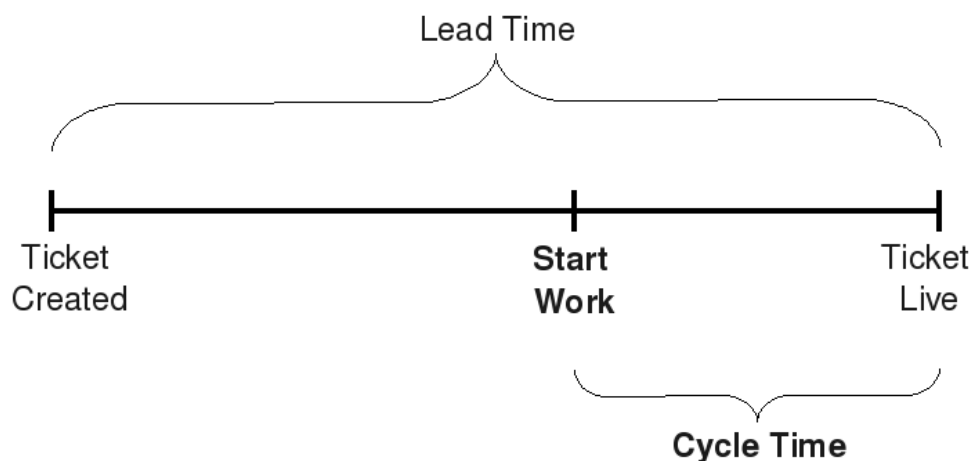
V této kapitole jsme si představili negativní výsledky získané měřením story points v projektu. V navazující části bude uveden průběh Burn Up chartu, který vyznačuje správný průběh sprintu a budou diskutovány další možné techniky měření pro získání potřebných informací o stavu a průběhu projektu.

## 5. Lead&Cycle time

**Lead&Cycle time:** jde o termíny, které je se používá v metodikách Kanban a Scrum a jim podobným. Tyto termíny se používají, abychom v rámci vývoje byli schopni informovat své obchodní partnery, kdy mohou očekávat svůj produkt, za jakou cenu a hlavně co očekávat. Jde spíše o to, aby se oba partneři snadněji domluvili, dále napomáhají lepší komunikaci v týmu a k upřesnění doby jednotlivých sprintů.

**Lead time:** se začíná měřit hned, jak se objeví požadavek a je umístěn na tabuli. Může jít o žádost o nový produkt, funkci, nebo jakoukoliv jinou práci v rámci firmy. Tato lhůta končí v době, kdy je veškerá práce na této položce dokončena a předána klientovi. Jde tedy o celkový čas, kdy klient čeká na dodání produktu.

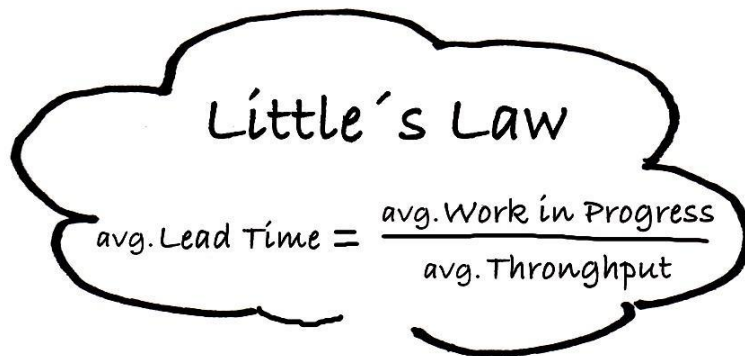
**Cycle time:** je doba kterou vynaložil tým na realizaci daného úkolu. Proto se tato doba začíná měřit až ve chvíli, kdy položka vstoupí do sloupce „working“.



Obrázek 4 - Lead vs Cycle Time

Pro vývojové týmy je dobré tyto hodnoty, tedy doby cyklů sledovat a snažit se je snižovat. Tím, že se zaměříme na zlepšení doby cyklů, dovolíme tím vyšit propustnost jednotlivých funkcí a zvýšit kvalitu dodávaného softwaru.

Jak si stanovit doby jednotlivých cyklů? Podle Littleova zákona se průměrný Lead time vypočítá jako podíl průměrné částky „work in progress“ k průměrné průchodnosti.



Little's Law

$$\text{avg. Lead Time} = \frac{\text{avg. Work in Progress}}{\text{avg. Throughput}}$$

Obrázek 5 - Little's Law

Work in progress rozumíme všechny položky, na kterých byla práce zahájena, ale dosud nebyla dokončena. Průchodnost zde představuje rychlost, s jakou se položky zpracovávají.

Jeden z důvodů, proč se zaměřujeme na snížení Lead&Cycle time, je to, že čím déle trvá dokončení jednotlivých položek, tím se zmenšuje hodnota, která je dodána zákazníkovi [Roock, 2012].

## 6. Burn down, burn up chart – Měření pokroku v projektu

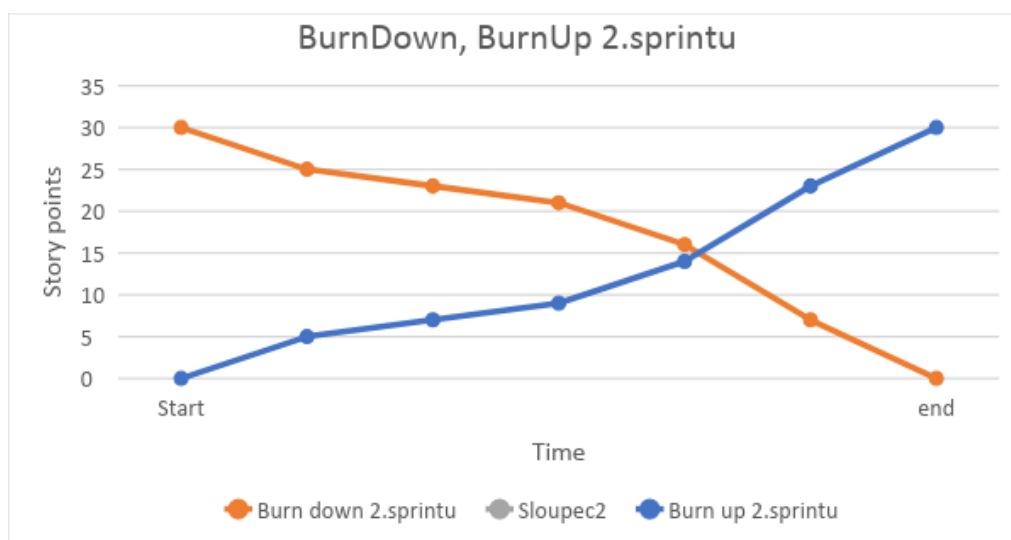
Burn down a burn up chart jsou dva typy grafů, které používají projektový manažeři pro sledování průběhu projektu. Burn down chart ukazuje, kolik práce nám na projektu ještě zbývá, zatímco burn up chart ukazuje, kolik práce bylo dokončeno a také celkový objem práce [ROTHMAN, 2016].

## 6.1. Burn down a burn up chart na modelovém příkladu:

Pro modelový příklad byly zvoleny následující hodnoty:

Story	Odhad	Priorita
User_story_1	2	1
User_story_2	5	2
User_story_3	6	3
User_story_4	9	4
User_story_5	3	5
<b>Celkem:</b>	<b>30</b>	<b>X</b>

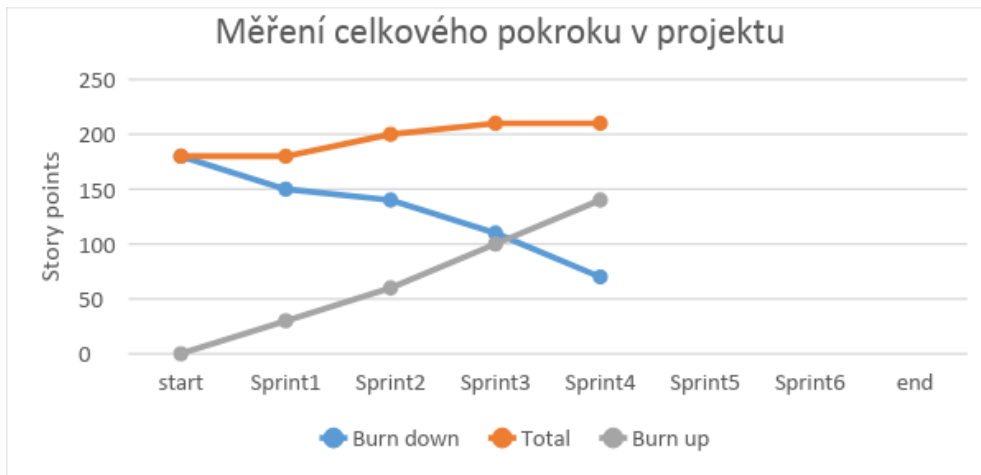
Tabulka 2 - Obsah sprintu 2 (zdroj:autor)



Obrázek 6 - BurnDown, BurnUp 2. sprintu (zdroj:autor)

Graf výše znázorňuje průběh spotřebování jednotlivých story points v průběhu času, v zásadě platí to, abychom dodrželi, že na konci sprintu budeme mít dokončeno to, co jsme si na začátku stanovili a neposouvali vývoj do dalších sprintů.

Na následující příkladu si představíme, co se stane s grafem, když v průběhu projektu přidává product owner další funkcionality.



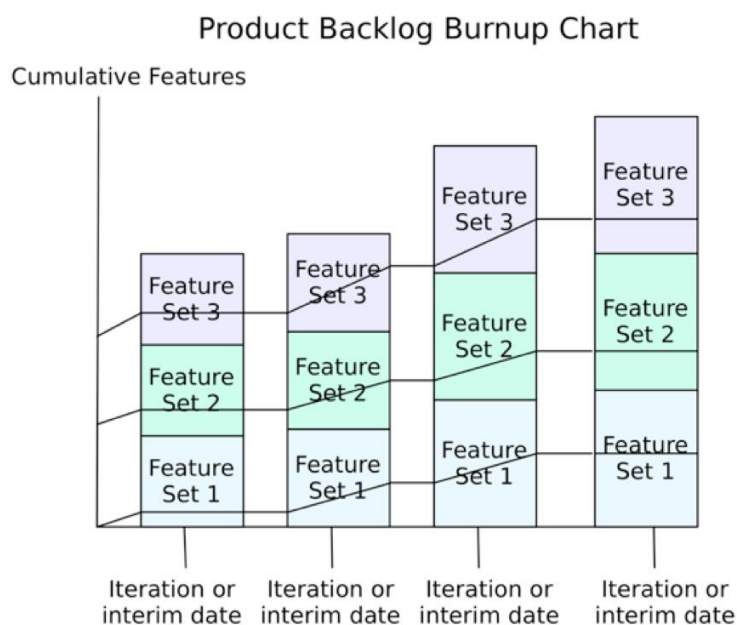
Obrázek 7 - Měření celkového pokroku v projektu (zdroj:autor)

Na grafu výše máme znázorněný průběh projektu po 4. sprintu. V první dva sprinty proběhly s hodnotou 30 story points, další 2 s hodnotou 40 story points. Po druhém sprintu nám však product owner navýšil přidání nové funkcionality celkovou hodnotu product backlogu o 20 story points a následně po třetím sprintu o dalších 10. Graf tedy znázorňuje postup průběhu projektu a bere v úvahu přidávání nových funkcionalit do product backlogu.

Graf slouží také pro projektového manažera k tomu, aby mohl vedení ukázat, v jakém stádiu se projekt nachází v souvislosti s přidáváním nových funkcionalit product ownerem.

## 7. Product Backlog Burnup Chart

Pokud budeme měřit průběh každou iterací, zjistíme, že sada funkcí poroste do té doby, dokud máme projekt aktivní a product owner přidává nové funkce do product backlog. Může se také stát, že se sady funkcí snižují a to v případě, kdy funkce pro zákazníka již nemá hodnotu, nebo je hodnota příliš nízká.



Obrázek 8 – Product Backlog Burnup Chart (Rothman)

## 8. Závěr

V textu této práce byly představeny pojmy user stories, story point a velocity, které představují základní měrné jednotky pro měření průběhu a stavu projektu v agilním přístupu pro řízení projektů. Následně byly tyto pojmy zakomponovány v praktickém příkladu, který vycházel z dat pro modelový podnik popsány v textu práce. V prvním příkladu byl představen průběh grafu ve tvaru hockey stick, který v našem případě vyjadřoval problém v podobě nedokončeného sprintu a nedokončených funkcionalit v rámci prvního sprintu. Modelová data byla následně diskutována z hlediska vzniklých problémů pro projektový tým a jeho nekvalitní dodávku prvního sprintu. V rámci druhého příkladu byla uvedena již data, která představují očekávaný průběh projektu a plnění jednotlivých user stories dle plánu projektu. Dále byl představen Littleův zákon, pomocí kterého můžeme vytvářet přesnější odhady na doby trvání cyklů a předcházet tak negativním efektům, které byly popsány v prvním příkladu. V závěru práce je uveden product backlog chart, v rámci kterého je popsán stav kdy jsou přidávány či odebrány dodatečné funkcionality pro jednotlivé sprinty.

## 9. Zdroje

1. AGILE ALLIANCE. User Stories [online]. [cit. 2018-03-12]. Dostupné z: <https://www.agilealliance.org/glossary/user-stories/>
2. AGILE ALLIANCE. Velocity [online]. [cit. 2018-02-12]. Dostupné z: <https://www.agilealliance.org/glossary/velocity/#q=~>
3. COHN, Mike. What are story points. [online]. [cit 2016-08-03]. Dostupné z: <https://www.mountaingoatsoftware.com/blog/what-are-story-points>
4. DALMIJN, Maarten, 12 common mistakes made when using Story Points [online]. [cit. 2017-11-04]. Dostupné z: <https://medium.com/bynder-tech/12-common-mistakes-made-when-using-story-points-f0bb9212d2f7>
5. DVOŘÁK, Drahošlav. Softwarová podpora řízení projektů: Bez metodických pravidel to ale nejde. [online]. [cit. 2010-04-07] Dostupné z: [https://www.systemonline.cz/business-intelligence/softwarova-podpora-rizeni-projektu-1.htm?co=&id=&chci\\_mesic=04&chci\\_rok=2016&chci\\_mesic1=&chci\\_rok1=](https://www.systemonline.cz/business-intelligence/softwarova-podpora-rizeni-projektu-1.htm?co=&id=&chci_mesic=04&chci_rok=2016&chci_mesic1=&chci_rok1=)
6. ROOCK, Stefan. Cycle Time vs Lead Time . KANBAN TOOL.[online]. [cit. 2018-05-02]. Dostupné z: <https://kanbantool.com/kanban-cycle-time-vs-lead-time>
7. ROOCK, Arne. Playing with Little's Law [online]. 2012 [cit. 2018-05-02]. Dostupné z: <http://www.software-kanban.de/2012/09/playing-with-littles-law.html>
8. ROTHMAN, Johanna. Customize Your Agile Approach: Start with Results You Want [online]. In: 1.listopadu 2017 [cit. 2018-05-02]. Dostupné z: [https://www.infoq.com/articles/agile-approach-results?utm\\_source=infoqWeeklyNewsletter&utm\\_medium=WeeklyNL\\_EditorialContent\\_culture-methods&utm\\_campaign=11072017news&utm\\_content=other](https://www.infoq.com/articles/agile-approach-results?utm_source=infoqWeeklyNewsletter&utm_medium=WeeklyNL_EditorialContent_culture-methods&utm_campaign=11072017news&utm_content=other)
9. ROTHMAN, Johanna. Value of Burndown and Burnup Charts. Rothman consulting [online]. 2. února 2016 [cit. 2018-05-02]. Dostupné z: <https://www.jrothman.com/mpd/project-management/2016/02/value-of-burndown-and-burnup-charts/>
10. SONI, Nimesh. Burnup looking like a Hockey Stick?. Nimeshsoni [online]. 3.května 2016 [cit. 2018-05-02]. Dostupné z: <http://www.nimeshsoni.com/burnup-looking-like-hockey-stick/>

## 10. Seznam obrázků

Obrázek 1- Příklad použití user stories v nástroji Trello.....	5
Obrázek 2 - Kumulativní součet dokončených Story points (zdroj:Autor).....	8
Obrázek 3 - Story points vs. dokončené funkcionality (zdroj:autor) .....	9
Obrázek 4 - Lead vs Cycle Time .....	10
Obrázek 5 - Little´s Law .....	11
Obrázek 6 - BurnDown, BurnUp 2. sprintu (zdroj:autor) .....	12
Obrázek 7 - Měření celkového pokroku v projektu (zdroj:autor) .....	13
Obrázek 8 – Product Backlog Burnup Chart (Rothman) .....	13

## 11. Seznam tabulek

Tabulka 1 - Obsah sprintu 1 (zdroj:Autor).....	8
Tabulka 2 - Obsah sprintu 2 (zdroj:autor) .....	12