Tento dokument by měl sloužit jako cheatsheet a ke každému tématu poskytovat vždycky 1-2 věty jako pick-up line, když se ho na dané téma zeptají. Např. "Cocomo - Model používaný k odhadování ceny SW. Idea je taková, že cena vývoje aplikace přímo závisí na velikosti SW."

tags: `řsss-základ`, `cheatsheet`

01 Softwarové inženýrství

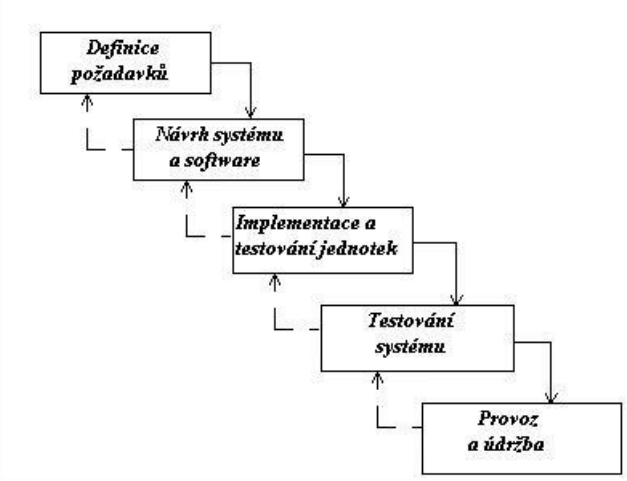
Proces vývoje SW

Vývoj software je proces, při němž jsou **uživatelovy potřeby** transformovány na **požadavky na software**, ten je transformován na **návrh**, který je implementován jako **kód**, který je testován, dokumentován a certifikován pro **operační použití**.

Modely životního cyklu

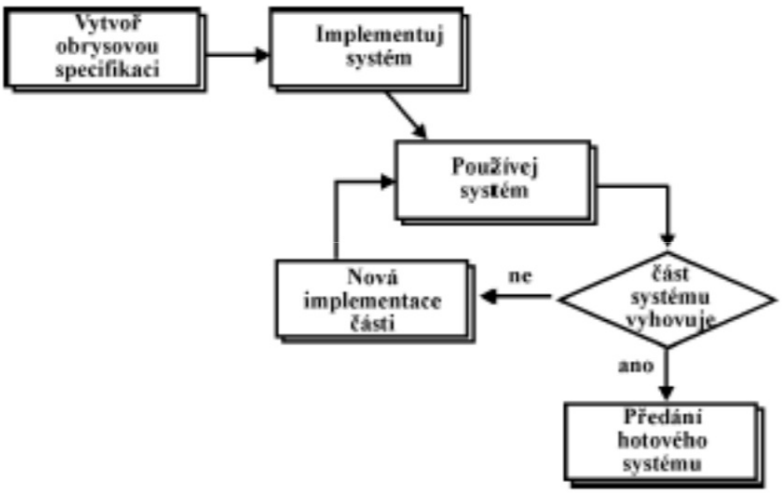
Vodopád

Model životního cyklu, jednoduše se řídí, ale postupuje se přísně sekvenčním způsobem a špatně reaguje na změnu.



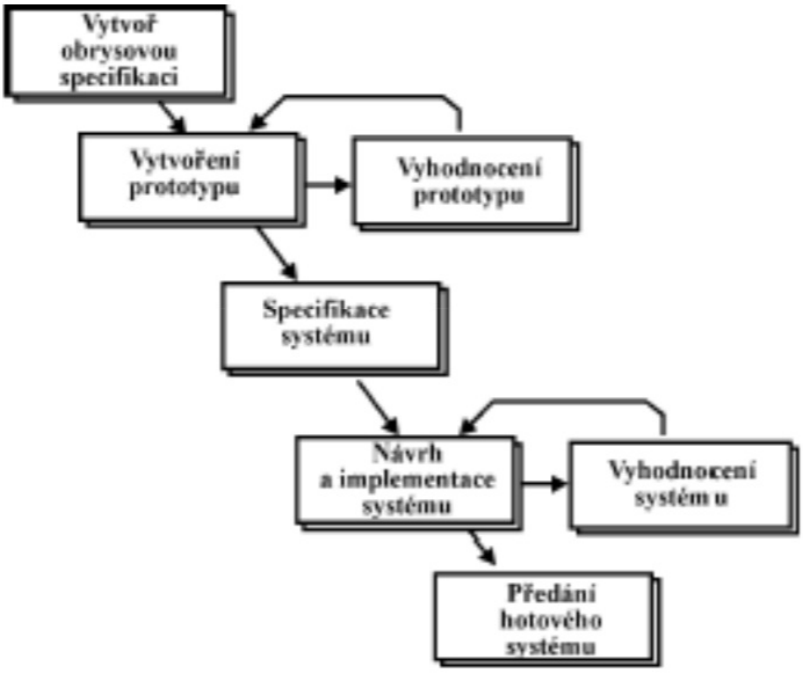
"Inkrementální"

Modifikace vodopádu. Finální projekt je rozdělen na dílčí verze - inkrementy, které postupně přidáváme, nicméně návrh nebude v takové kvalitě, jako když ho uděláme najednou. Určen pro velké projekty.



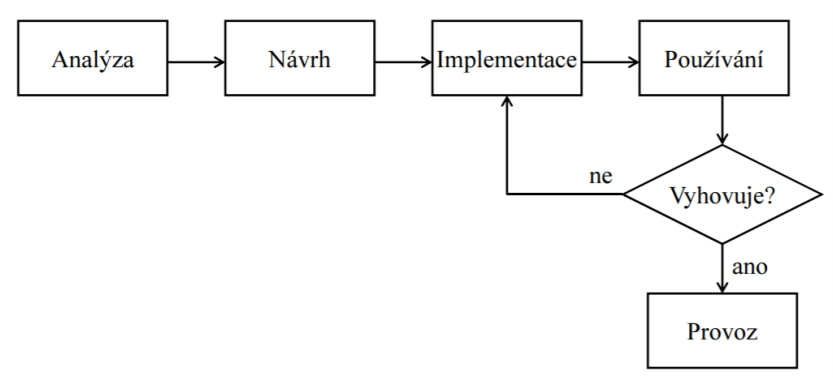
Model životního cyklu „prototypování“

Zákazník nedokáže vyjádřit své požadavky - vytvoříme mockup, na ten dá feedback, pokud se líbí, tak zahodíme prototyp a jdeme implementovat. Tvorba prototypů **probíhá za účelem získávání poznatků**, neslouží jako předloha k vývoji.



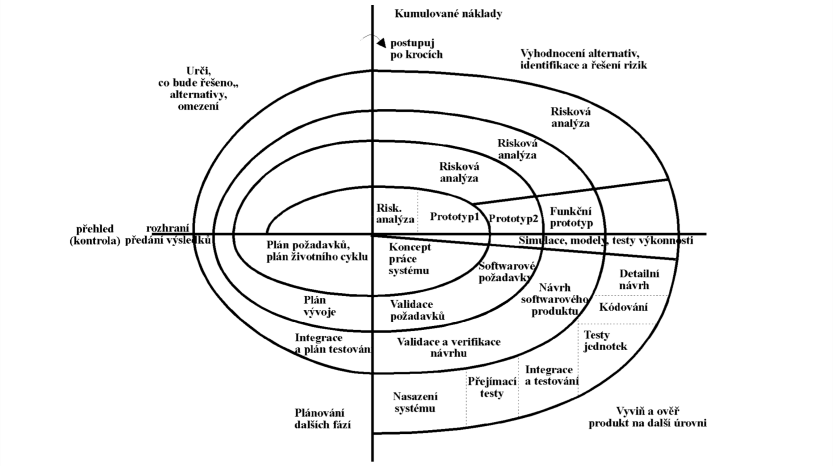
Model životního cyklu „výzkumník“

Prakticky neznáme požadavky. Většinou jsou ve stylu: "Postavte formuli, která vyhraje šampionát."



Spirálový model životního cyklu

Vývoj v cyklech/iteracích, díky kterým se přibližujeme k danému cíli.



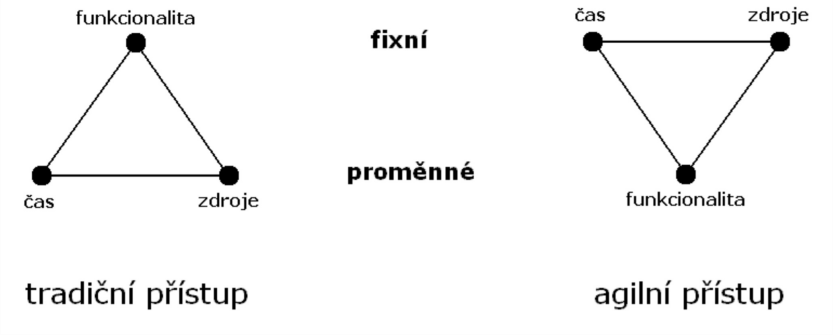
Základní přístupy k vývoji softwaru

Prediktivní (Tradiční)

- **Tradiční**
- Fixní funkcionalita, čas a zdroje se mohou měnit
- Rigidní (těžce upravovatelný)
- Zaměřuje se na proces
- **Unified process**
- Např. Jaderná elektrárna

Agile

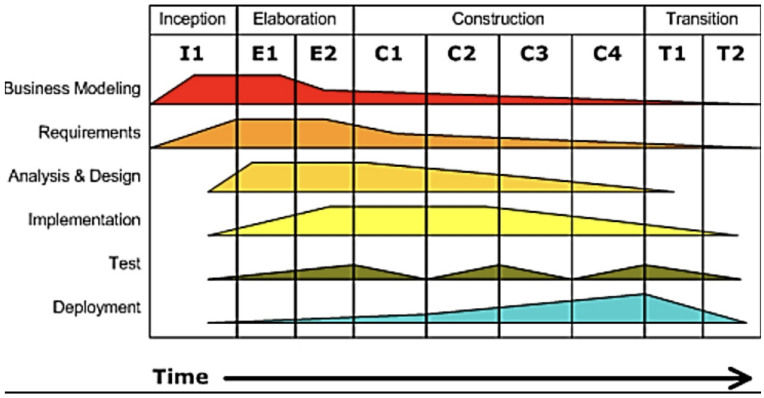
- Fixní čas a zdroje, inkrementy(funkcionalita) se přidávají dokud nevyčerpáme zdroje a čas
- Flexibilní a adaptabilní
- Zaměřuje se na lidi a mezilidské vztahy
- **SCRUM**
- Např. E-shop



Metodika Unified Process (UP)

- **Prediktivní vývojový framework, kde víme, jaký bude náš konečný výsledek. Je iterativní a inkrementální.**
- 4 fáze, 6 workflows

UP Lifecycle



Source: (Arlow, 2005)

4 fáze

Každá iterace by měla trvat max. 3 měsíce. - **Zahájení (Inception)**: Téměř předprojektová fáze. Dělá se studie proveditelnosti (je to realizovatelné?). - Milník: Cíle - UML diagramy: Activity diagram - **Příprava (Elaboration)**: Sběr požadavků. Programátoři prokopávají technologie. - Milník: Základ architektury. Podepisujeme specifikační dokument. - UML diagramy: Sequence diagram a class diagram - **Construction**: Vývoj produktu a testování - Milník: Počáteční funkcionalita - UML diagramy: Component diagram, Class diagram, Object Diagram - **Předávání (Transition)**: Předání a nasazení produktu zákazníkovi. - Milník: Release produktu - UML diagramy: Deployment diagram

6 workflows

- **Business modeling** - Activity diagram (procesní model, chceme si namodelovat proces práce s daným systémem)
- **Requirements** - Use Case diagram (používá se pro funkční požadavky)
- **Analysis & Design** - Class, Sequence, Collaboration diagram
- **Implementation** - Class, Object, Component diagram
- **Test** - Use Case, Class Activity diagram
- **Deployment** - Deployment diagram

Agilní vývoj SW

Metodiky vývoje SW, které dobře reaguje na změnu. Zákazník se účastní procesu vývoje. - Iterativní a inkrementální vývoj - Komunikace mezi zákazníkem a vývojovým týmem

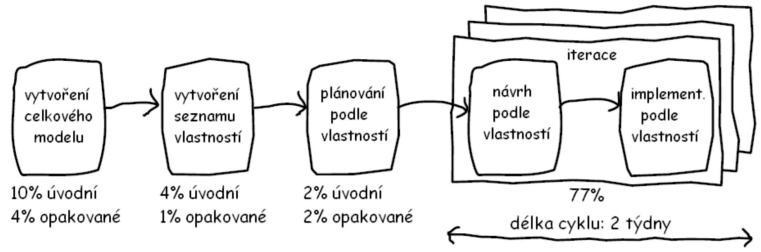
Extreme Programming (XP)

Metodika přizpůsobená programátorům, největší hodnota je kvalitní kód. Hodí se pro menší projekty a malé týmy. Používá běžné principy a postupy, které dotahuje do extrému - např. pokud se osvědčí revize kódu, bude se pořád revidodat, jestli testování, tak se bude testovat.

Feature-Driven Development

Metodika zaměřená na vývoj po malých kouscích (inkrementech). 5 fází, první 3 jsou sekvenční, poslední 2 iterativní.

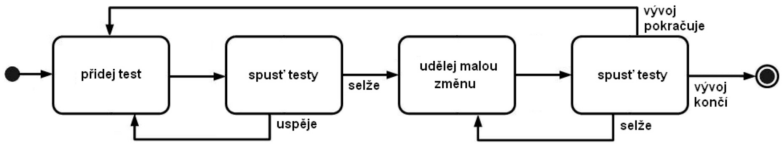
Fáze metodiky z pohledu rozdělení času:



Test-Driven Development

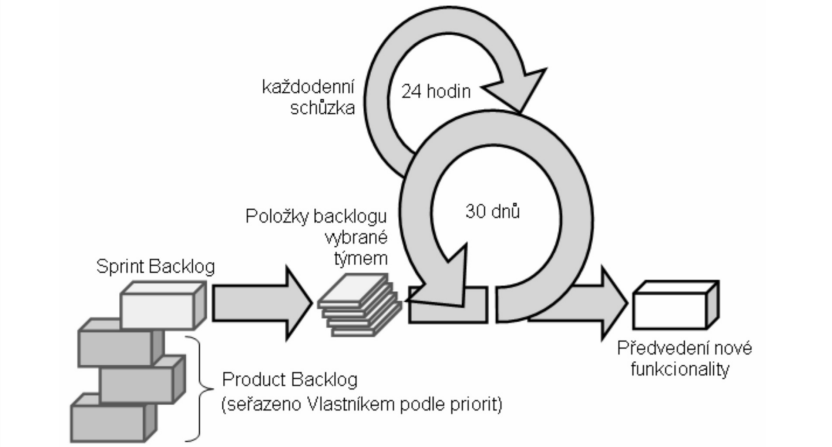
Metodika s principem "co projde testem, je v pořádku".

Postup metodiky:



SCRUM

Iterativně inkrementální způsob řízení vývoje SW. Na konci každé iterace by měl vzniknout nový, funkční inkrement. - **5 Events** - Sprint - **Sprint Planning** - Plánování sprintu. Vybírají se položky z product backlogu. - **Daily SCRUM (standup)** - 10 minutový každodenní meeting - Kdo na čem dělá, na čem jsem se zasekl - **Sprint Review** - Scrum team a zákazník - Kontroluje se inkrement, prohlíží se product backlog a rekalkuluje se progres a datum dodání. - **Sprint retrospective** - Pouze scrum team - Probírá se uběhlý sprint - mezilidské vztahy, procesy a tooly - co fungovalo, co zlepšit - **3 Artifacts** - Product Backlog, Sprint Backlog, Product Increment - **3 Role** - Product Owner, SCRUM Master, Team of Developers

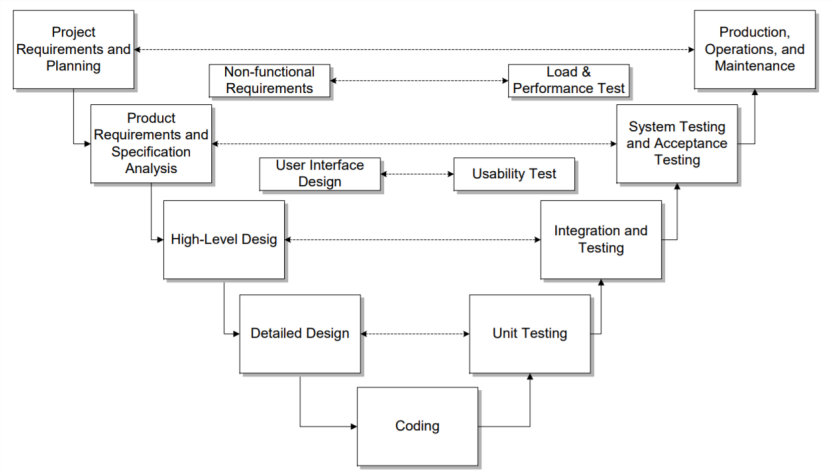


Fáze testování a typy testů

Testování je proces spuštění programu s cílem nalézt chyby. Typy testů: - Vizuální testování - Unit testy - Integrační testy - Systémové testování (performance, penetrační, akceptační, beta testy)

V-Model

Model životního cyklu softwaru, kde na levé straně jsou zobrazené jednotlivé fáze vývoje softwaru a na pravé jsou testy ke každé fázi. Idea je taková, že nejzásadnější chyby, které stojí nejvíc peněz se odhalí až jako poslední. Chyby nejčastěji vznikají při sběru požadavků a v návrhu. Akceptace a výš - black-box. Integrační testy a níž - white-box.



Validace vs. Verifikace

Validace: Test proti specifikovaným funkcím. Dělat správné věci. Black-Box **Verifikace**: Test proti vnitřní činnosti. Dělat věci správně. White-Box

Black-box testování

Funkční testování zaměřené pouze na vstupy a výstupy.

White-box testování

Testování, které zohledňuje strukturu programu.

Integrační postupy

Shora dolů (TDT)

Nejdříve implementuji jádro (kostru) systému a postupně přidávám moduly (neexistující moduly nahrazují "protézou"). Odhaluje chyby analýzy a návrhu.

Zdola nahoru (BUT)

Nejdříve implementuji moduly, které spojím pomocí "drivers".

Softwarové metriky

Software měříme primárně kvůli tomu, abychom dokázali určit kvalitu projektu/procesu. Taktéž můžeme měřit velikost a složitost. Celé je to o tom, že chceme zvyšovat kvalitu SW.

Míra

Kvantitativní(číselně vyjádřená) údaj o množství, rozměrech, kapacitě, nebo velikosti nějakého atributu, produktu nebo procesu. - **Přímá míra**: Počet řádků kódu (LOC), rychlost výpočtu, velikost paměti, počet chyb - Jednoduše vyčíslitelné - **Nepřímá míra**: Funkčnost, kvalita, složitost, pracnost, spolehlivost, schopnost údržby

Metrika

Kvantitativní (číselně vyjádřená) míra, tj. ukazatel do jaké míry se nějaký atribut vyskytuje v systému, komponentě, nebo procesu. (Počet chyb na 1000 řádků). - **Procesní metriky:** - Činnosti související s procesem vývoje softwaru - Např. Kolik LOC bylo během procesu napsáno - **Produktové metriky:** - Týkající se produktu - Např. Počet řádků kódu hotového produktu - **Metriky zdrojů:** - Týkající se zdrojů(lidí) - Počet lidí, jejich vzdělání a certifikáty; HW prostředky

Metriky založené na velikosti kódu

- LOC, KLOC
- Počet chyb/KLOC
- Cena/KLOC
- Počet stránek dokumentace/KLOC
- **LOC je závislá na programátorovi a programovacím jazyku**

Funkčně orientované metriky

Metriky na základě funkčních bodů (FP) Odvozené pomocí empirických vztahů založených na spočitatelných vlastnostnech systému. - Chyby/FP - Cena/FP - Dokumentace/FP - FP za člověkoměsíc

Funkční body produktu: Funkční body, které zůstanou v aplikaci na konci vývoje
Funkční body projektu: Funkční body, které prošly týmu rukama.

Metriky složitosti

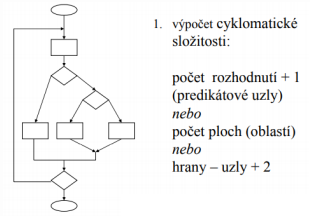
Kvantitativní metrika složitosti SW. Jsou závislé na programátorovi a programovacím jazyku.

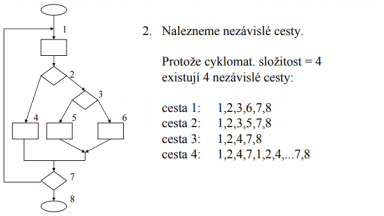
Halsteadova metrika

Softwarová metrika výpočtu složitosti na základě statistické analýzy kódu. - unikátní operátory/operandy vs. celkový počet operátorů/operandů - Estimated length = $n_1 \log_2 n_1 + n2 \log_2 n_2$ - Purity ratio: Ñ / N. Pokud PR>1 -> kód je zbytečně krátký.

Cyklomatická složitost

Softwarová metrika výpočtu složitosti, kde počítáme počet možných průchodů grafem, nebo počet nezávislých větví. Spojuje se s white-box testy. Pokud V(G) > 10 - velký chybový potenciál a chceme modul dekomponovat. - V(G) = P + 1 - V(G) = E - N + 2





Coupling

Coupling metriky vyjadřují to, jak máme propojené dva moduly, jinak řečeno - míra složitosti integrace. Čím je nižší, tím lepší (low/loose coupling) = nízká míra provázanosti. Čím je vyšší, tím horší (high/tight coupling) = vysoká míra provázanosti. Sčítám všechny parametry (input/output data/control + global data/control variables + počet modulů, které volám a které volají mě). 1/všechno = míra provázanosti.

Důvody pro měření metrik

- **plánování projektu**
- **kontrola kvality produktu**
- **odhad produktivity**
- **zdokonalení práce**

Refaktoring kódu

Refaktorování je disciplinovaný proces provádění změn v softwarovém systému takovým způsobem, že nemají vliv na vnější chování kódu, ale vylepšují jeho vnitřní strukturu s minimálním rizikem vnášení chyb. - "Bad smells": Duplicitní kód, dlouhá metoda, velká třída, middle man, lazy class, …

Systémový re-engineering

Znovu napsání celé části systému bez účelu změnit její funkcionalitu.

Odhadování úsilí

COCOMO (Constructive Cost Model)

Model používaný k odhadování ceny SW. Idea je taková, že cena vývoje aplikace přímo závisí na velikosti SW. - Hlavní indikátor velikosti SW je LOC - Vstupní hodnotou je odhad velikosti SW (LOC)

3 úrovně detailu: Podle toho, v jaké fázi projektu jsem. - Základní model - Hrubý odhad. Dělá se v úvodních fázích. - Střední model - Od něj a výše používá F_c - koreční faktor - součin 15 atributů (od velmi nízký až po extrémně velký) specifických pro vývojový proces (**atributy SW produktu, HW, vývojového týmu, projektu**). Pokud je všechno

"normální", tak je F_c = 1. Pokud máme slabší tým, tak je F_c>1$, protože potřebujeme trochu kompenzovat. - Pokročilý model - Bere v úvahu vlivy vývojové etapy.

3 vývojové módy: Podle toho, jak náročný vývoj bude. - Organický mód - Pro malé projekty, kde nás nemůže nic moc překvapit - Bezprostřední mód - Pro střední projekty, které už mají definované omezení v rozhraní, závislost na speciálním HW, etc. - Vázaný mód - Pro projekty všech velikostí, kde máme hrubé představy o cílech, striktní omezení na rozhraní, velkou závislost na speciálním HW, etc.

Úsilí a čas: E(effort) = a * ($KSLOC$)^b T(time) = c * E^d KSLOC odhadnu z empirických zkušeností; a,b,c,d najdu v tabulkách pro zvolený model a mód.

Atributy F_c:

- Mohou nabývat 6 možných hodnot ve stupnici: velmi nízký až extrémně velký
- Skupiny
  - Atributy SW produktu
  - HW Atributy
  - Atributy vývojového týmu
  - Atributy projektu

COCOMO2

Podobné jako COCOMO 3 různé modely: - ACM (Application Composition model) - EDM (Early Design Model) - PAM (Post Architecture Model)

2 společné vlastnosti COCOMO a COCOMO2 - Oba způsoby při odhadu ceny zahrnují jistou množinu faktorů, která ji ovlivňuje - Oba způsoby využívají stejný druh modelů na rozlišení výpočtu

3 rozdíly COCOMO a COCOMO2 - COCOMO2 zahrnuje některé nové atributy na měření odhadu ceny, které vznikly kombinací předchozích z COCOMO - Modely v COCOMO2 jsou na rozdíl od COCOMO zaměřené spíše na vývojovou etapu projektu - Při odhadování nákladů na úpravu aplikace využívá COCOMO2 i tzv. AA a SU koeficienty - Assessment and Assimilation (AA) - práce potřebná pro určení, zda a v jakém rozsahu může být existující modul použit beze změn. - Software Understanding (SU) - čitelnost a "uchopení" - jak dobře je kód čitelný, jestli je dobře zdokumentovaný, …

Funkční body (FP)

Normalizovaná metrika softwarového projektu, která měří aplikační oblast. Zaměřuje se na pohled ze strany uživatele a oproti COCOMO odbourává závislost na programátorovi a programovacím jazyku a dokážeme je spočítat s vysokou přesností už ve fázi analýzy.

Typy funkčních bodů vztažené k transakčním funkcím:

- **Externí vstupy (EI)**
  - UPDATE/DELETE/INSERT
  - Formulář, kde uživatel zadává data.
- **Externí výstupy (EO)**
  - SELECT
  - Přečte se něco v databázi a zobrazí se to v aplikaci.
- **Externí dotazy (EQ)**
  - Unikátní vstupně/výstupní kombinace, kde vstup je příčinou a generuje výstup.
  - Např. zadám nové tel. číslo a jako výstup se mi zobrazí "Staré číslo X změněno na nové Y."

Typy funkčních bodů vztažené k datovým funkcím

- **Vnitřní logické soubory (ILF)**
  - Data, které si systém uchovává sám.

- **Soubory vnějšího rozhraní (EIF)**
  - Data, které si systém neuchovává, ale přistupuje k nim prostřednictvím integrace jiného systému.

Matice složitosti

Na všechny funkční body jsou stejné, proto je musím upravit přes matici složitosti. Matice složitosti vstupů/výstupů a souborů podle vah (nízká, průměrná, vysoká.) - **FTR** (File Types Referenced) - Počet dotčených záznamů/tabulek/entit - **DET** (Data Element Type) - Počet atributů - **RET** (Record Element Type) - Počet datových elementů v ILF nebo EIF

Obecné charakteristiky systému

14 charakteristik hodnocených podle stupně vlivu na aplikace (rychlost, bezpečnost, objemy dat) na stupnici 0-5.

Počet funkčních bodů

Princip: Spočítám funkční body -> roztřídím do skupin podle vah na základě matice složitosti (získám neupravené funkční body) -> vypočítám obecné charakteristiky: - **Počet funkčních bodů** = [0.65 + (0.01 × charakteristiky systému)] × (neupravené funkční body)

Výsledné číslo hovoří o tom, jaká je velikost z pohledu funkcionality, co systém dokáže.

Softwarová fyzika

Vyjádření vztahů mezi základními veličinami (čas, pracnost, velikost softwaru v FP nebo LOC) v softwarovém inženýrství.

- **N** - délka programu (počet řádek, SLOC)
- **T** - spotřeba práce (člověkoměsíce, MM)
- **P** - produktivita P=N/T
- **D** - doba realizace programu
- **S** - průměrný počet řešitelů

Práce a délka programu

S množstvím řádku kódu roste pracnost - tzn. čím složitější program, tím trvá výroba déle. (Těžce se v něm orientuje, větší tým, víc komunikace)

Produktivita

S rostoucí délkou programu klesá produktivita programátorů.

Putnamova rovnice

Vztah mezi délkou kódu (N), provedené práce (T) a dobou řešení (D). Vyplývá z ní, že programy psané ve spěchu jsou delší a při zkrácení termínu na 83% je pracnost dvojnásobná. - $N=c\times T^{\frac{1}{3}}\times D^{\frac{4}{3}}$ - *c* je konstanta vyjadřující produktivitu firmy (zkušenost, nástroje, etc.)

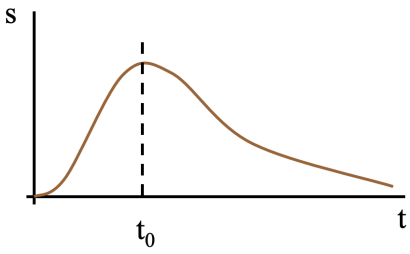
Rozložení řešitelské kapacity v čase

Na SW projektu pracuje nejvíce lidí a je utracena většina budgetu přibližně v 40% času.

Model rozložení aproximován pomocí vlny s(t) (Rayleigh)

s(t) - počet řešitelů v čase *t*

$$s(t) = T\frac{t}{t_0}e^{\frac{-t^2}{2t_0}}$$

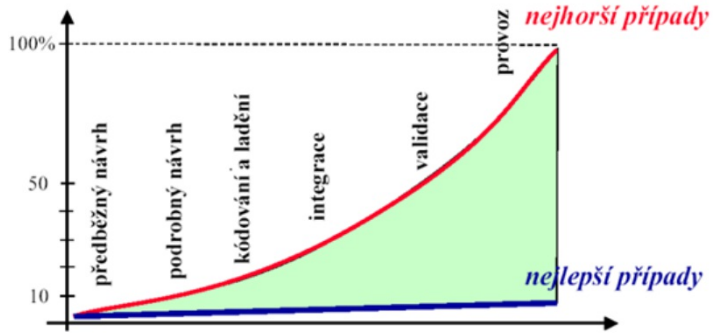


Celková spotřeba práce T:

$$T = \int_0^\infty s(t)dt$$

Údržba a znovupoužitelnost

Údržba je modifikace SW produktu po předání zákazníkovi za účelem opravy chyb, zvýšení výkonnosti a přizpůsobení měnícímu se okolí.



Pokud si dobře rozmyslíte strukturu nového SW a vše uděláte „pořádně“, bude vývoj sice o něco dražší, ale vynaložené náklady se vrátí v období provozu SW systému, kdy děláte jeho údržbu a na přání zákazníka provádíte modifikace.

Znovupoužitelnost

Hlavní výhodou je několikanásobné finanční ohodnocení jednou vyvinuté komponenty.

Úrovně znovupoužitelnosti: - Abstrance - Objekty - Komponenty - Systém

Lehmanovy zákony

Zákony se zabývají fází evoluce, popisují rovnováhu mezi novými požadavky a údržbou na straně jedné a zvyšující se složitostí, snižující se “business value” na straně druhé. - Zákon trvalé změny - Zákon rostoucí složitosti - Zákon vývoje programu - Zákon invariantní spotřeby práce - většina budgetu se utratí v 40% času projektu. - Zákon omezené velikosti přírůstku

Brooksův zákon

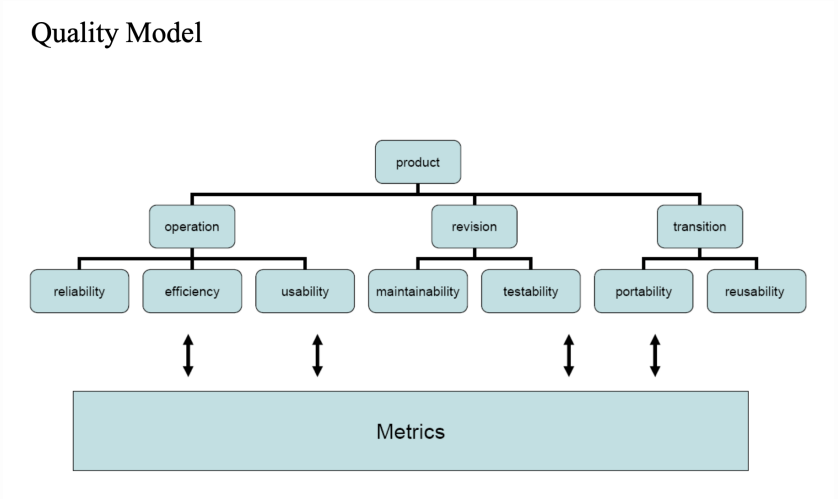
Přidání programátora do opožděného projektu může zvětšit jeho zpoždění.

Kvalita softwaru

KVALITA se rovná dodržení explicitně stanovených funkčních a výkonových požadavků, dodržení explicitně dokumentovaných vývojových standardů a implicitních charakteristik, které jsou očekávány u profesionálně vyrobeného software.

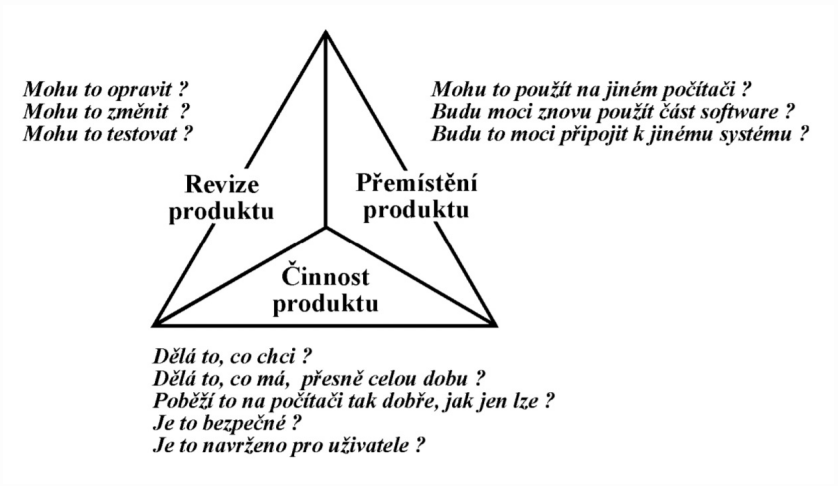
Kvalita podle IEEE - Stupeň do jaké míry systém, komponenta nebo proces splňuje specifikované požadavky.

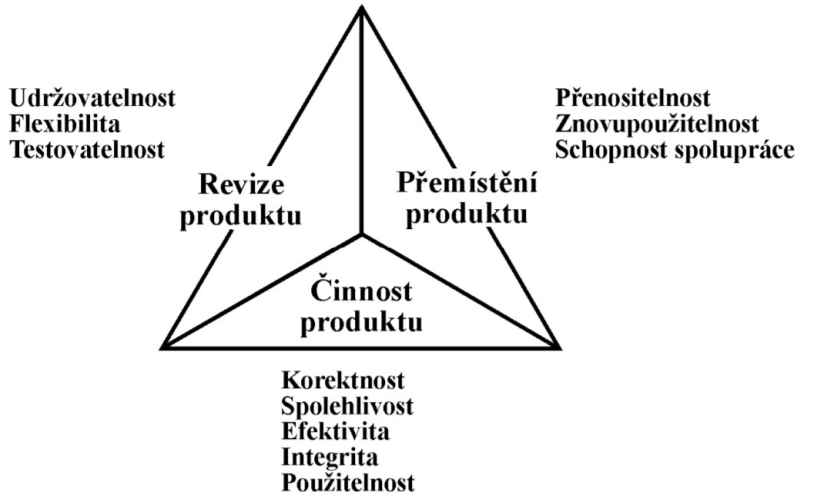
Aspekty kvality: - Odchylky od požadavků na software - Nedodržení standardů - Odchylky od běžných zvyklostí



operation - funkčnost: **Počet chyb / FP revision** - schopnost akceptovat změny: **Cyklomatická složitost transition** - přenositelnost na jiné prostředí, znovupoužitelnost: **Dokumentace / KLOC**

Faktory kvality - McCall et al. (1997)





Globální hodnocení kvality výroby

CMM - Capability Maturity Model

Hodnotí vyspělost organizací podle stupně a kvality využívání SW procesů. Od úrovně 0, kde organizace nevyužívá žádné procesy až po úroveň 5, kde investuje do optimalizace procesů.

Software Quality Assurance (SQA) - Zajišťování kvality SW

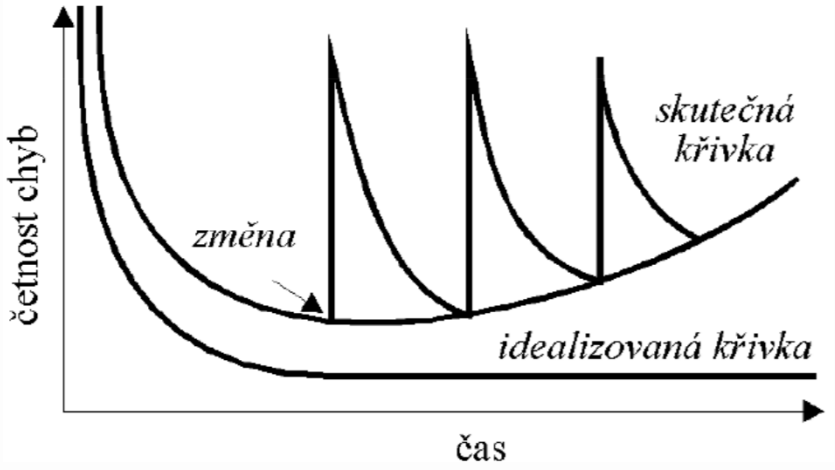
Recenze softwaru z pohledu zajištění kvality. Zahrnuje monitorování všech procesů, metod a pracovních produktů softwarového inženýrství. Definuje a dokumentuje politiku kvality, zodpovědností, autorit a vztahů mezi všemi osobami, které svojí prací mohou ovlivnit kvalitu. **Přínos SQA:** Většina aktivit by měla být podrobena inspekci, protože to vede k úspoře peněz. Například chyba odhalená v provozu stojí až 100x víc, než když ji odhalíme při návrhu. Efektivita přezkoušení roste s její formálností.

Závažnost chyb, defektů

- Kritické
  - Defekty, které mohou způsobit pád systému, vznik chybných výstupů či chování nebo narušit uživatelská data. Není známa cesta, jak se těmto defektům vyhnout.
- Vážné
  - Defekty, které způsobují chybné výstupy či chování a je známa cesta, jak se těmto defektům vyhnout. Zasažena je významná část systému.
- Středně závažné
  - Defekty ovlivňující omezenou část funkcionality, kterým je možné se vyhnout nebo je ignorovat.
- Málo závažné
  - Defekty, které mohou být opomenuty bez narušení funkčnosti

Chyby a opotřebení SW

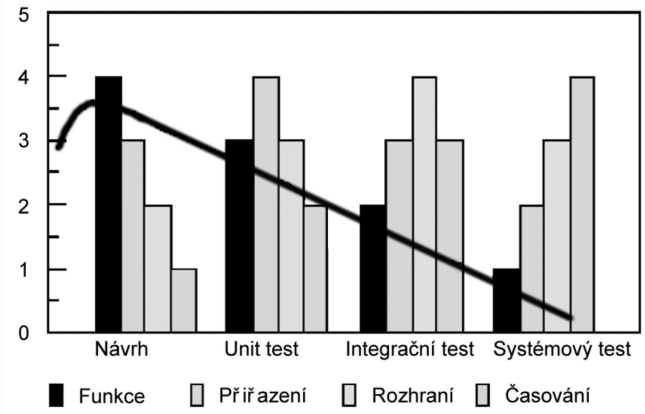
Je naivní si myslet, že s postupem času odstraníte v SW všechny chyby a zmizí tak veškeré problémy.



IBM ortogonální klasifikace defektů

Klasifikaci defektů děláme kvůli tomu, že na základě kategorie dokážeme určit fázi vývoje, kdy chyba vznikla.

Funkční selhání podle etapy



Celkové defekty a defekty typu *Funkce*

- Typy testů odhalují různé typy chyb - Například: Typ chyby "funkce" se nejvíc objevují v návrhu

---

02 Informační bezpečnost

Audit, řízení bezpečnosti, řízení rizik, protiopatření.

Audit

Bezpečností audit je nezávislé posouzení stavu bezpečnosti v organizaci. Audit posuzuje, zda procesy a opatření definované v bezpečnostní politice jsou správně implementovány a používány.

Řízení bezpečnosti

Bezpečnost je soubor opatření a činností k zajištění ochrany aktiv a funkčnosti a

spolehlivosti infrastruktur a technologií. Řízení bezpečnosti spočívá v plánování, organizování, přidělování pracovních úkolů s cílem dosáhnout požadované úrovně bezpečnosti.

Jako vstup k řízení bezpečnosti je vyžadováno: - Vypracovaná Bezpečnostní Politika (BP) - Management na všech úrovních prosazuje BP - Má být implementován měřící systém

ISMS Information Security Management System

Systém řízení bezpečnosti informací je dokumentovaný systém, ve kterém jsou chráněna definovaná informační aktiva, jsou řízena rizika bezpečnosti informací a zavedená opatření jsou kontrolována.

Řízení rizik

Jako řízení rizik jsou označovány procesy vedoucí k redukci rizik na akceptovatelnou úroveň. - **Ohodnocení rizik** (risk assessment) se skládá z: - **Analýza rizik** (risk analysis) - **Vyhodnocení rizik** (risk evaluation) - **Zmírnění rizik** (risk mitigation) - Výběr a implementace opatření snižující rizika - **Akceptace rizik** (risk acceptance) - Rozhodování o přijatelnosti rizik dle stanovených kritérií - **Informování o rizicích** (risk communication) - Sdělení informace všem, kdo mohou rizika ovlivnit nebo být jimi ovlivněni

Ohodnocení rizik je systematické zkoumání aktiv, hrozeb a možných útočníků. Cílem je dosáhnout vyrovnanosti časových a finančních nákladů na ochranu a provoz. Výstupem bývá tabulke s aktivy, hrozbami a míry rizika. - Prvním krokem je orientační ohodnocení rizik. Postupuje se buď pomocí *elementární*, *neformální*, *detailní* (formální) nebo *kombinované* metodologie.

Forma **analýzy rizik** je buď **kvantitativní** nebo **kvalitativní**.

Protiopatření

Bezpečnostní politika (BP) je soubor pravidel specifikující účinný způsob uplatňování opatření (implementované adekvátními mechanismy) potřebných pro dosažení požadované úrovně akceptovatelných rizik. BP říká: - co se chrání - bezpečnostní cíle - jak se ochrana uplatňuje - způsob dosažení bezpečnostních cílů

Hodnocení bezpečnosti, hodnotící kritéria a procesy.

Hodnocení bezpečnosti se provádí, aby se zjistila dosažená úroveň bezpečnosti. K hodnocení bezpečnosti se využívají ISO/IEC normy.

Hodnotící kritéria

Seznam podmínek, které vyvíjený/kupovaný produkt nebo systém má být schopný (musí) splnit, resp. kterým musí vyhovět.

Common Criteria (CC)

Common Criteria poskytují framework, který umožňuje aby uživatel specifikoval požadavky na bezpečnost produktu, výrobce specifikoval vlastnosti produktu a nezávislý hodnotitel posoudil zda výrobek odpovídá požadavkům.

Dělí se na 3 části: - Part 1: Introduction and general model - Part 2: Security functional requirements - Part 3: Security assurance requirements

Specifikační dokument CC

- **Profil ochran**

    - Identifikuje požadavky na bezpečnost pro jisté prostředí
        - Použití čipových karet pro nepopíratelnost u podepisování, síťové firewally pro řízení přístupu
- **Bezpečnostní cíl**
    - Dokument definující bezpečnostní vlastnosti produktu/systému.

Důležité pojmy z CC

- **Předmět hodnocení** (TOE - target of evaluation)
    - Např. smartcard
- **Specifikace bezpečnosti** (ST - security targe)
    - Cílová kombinace komponent spojených s konkrétním produktem nebo systémem
        - Např. že smartcard je tamper-resistant
- **Profil bezpečnosti** (PP - protection profile)
    - Implementačně nezávislá skupina bezpečnostních požadavků určité skupiny TOE
        - Např. použití čipových karet pro nepopíratelnost u podepisování

Standardy v IT bezpečnosti a kryptografie, legislativa týkající se kryptologie

Standard

Standard (neboli norma, doporučení) je úmluva o technické specifikaci, nebo o jiném podobně přesně stanoveném kritériu. Standardy se dělí na de iure (schválena uznávanou institucí) na de facto (v rámci jisté komunity, např.: RFC) a firemní (proprietární) standardy.

Kryptografické standardy

- **Symmetric crypto** - DES, AES
- **Asymmetric crypto - encryption, signatures, key exchange and transfer**
    - **IEEE P1363** - Factoring-based, Discrete log based, Elliptic curve
    - **NIST FIPS 186-3** - Digital Signature Standard
- **Hash functions** - SHA-3 (Secure Hash Algorithm) (Keccak)
- **PKCS** - norma vyvinutá formou RSA Security pro snadné používání kryptografie s veřejnými klíči

Digitální podpis - konstrukce, legislativa, správa veřejných klíčů, certifikační autority a infrastruktura veřejného klíče

Digitální podpis

Digitální podpis je typ elektronického podpisu, který je použit k zajištění integrity a autenticity zasílané zprávy.

Legislativa - ČR

Elektronický podpis

Elektronickým podpisem se rozumí údaje v elektronické podobě, které jsou připojené k datové zprávě nebo jsou s ní logicky spojené a které slouží jako metoda k jednoznačnému ověření identity podepsané osoby ve vztahu k datové zprávě.

Zaručený elektronický podpis

Ověřuje integritu zprávy ale ne identitu podepsané osoby (certifikát může být vydaný kýmkoliv - např. si vygeneruju vlastní klíče)

Uznávaný elektronický podpis

Je zaručený elektronický podpis založený na kvalifikovaném certifikátu vydaném akreditovaným poskytovatelem certifikačních služeb a obsahujícím údaje, které umožňují jednoznačnou identifikaci podepisující osoby.

Kvalifikovaný elektronický podpis

Založen na kvalifikovaném certifikátu a je definovám nařízením Evropského parlamentu eIDAS. Jedná se o nejvyšší úroveň elektronického podpisu.

Elektronická pečeť

Stejné jako zaručený elektronický podpis, ale podepisujícím je právnická osoba, organizační složka státu, atd., která drží prostředek pro vytváření elektronických pečetí a označuje datovou zprávu elektronickou pečetí.

Certifikační autorita (CA)

Certifikační autorita je v asymetrické kryptografii subjekt, který vydává digitální certifikáty, čímž usnadňuje využívání PKI tak, že svojí autoritou potvrzuje pravdivost údajů, které jsou ve volně dostupném veřejném klíči uvedeny.

Certifikát

Veřejný klíč uživatele podepsaný soukromým klíčem důvěryhodné třetí strany.

Public key infrastructure

Pomocná infrastruktura pro správu veřejných klíčů založena na prvcích: - Bezpečnostní politika - Procedury - Produkty - Autority

Komponenty PKI - Certifikační autorita (CA) - Registrační autorita (RA) - Adresářová služba

Autentizace uživatelů v počítačových systémech - tajné informace, tokeny, biometrie.

Autentizace - proces ověření identity uživatele Autorizace - proces udělení určitých práv a určení povolených aktivit

Autentizační metody: - Co kdo zná - heslo, PIN - Co kdo má - token, čipová karta - Co kdo je - Biometrie

Identifikace vs. Autentizace * *Identifikace* * Určení totožnosti osoby (1:N) - "Kdo jsi?" * "Pozitivní autentizace" * Hůře dosažitelné (malá skupina uživatelů, nízká přesnost, výjimka: sken duhovky) * *Autentizace* * "Opravdu jsi ten, kdo tvrdíš, že jsi?" * Verifikace (ověření) tvrzení osoby o její totožnosti (1:1) * Jednodušší než identifikace

---

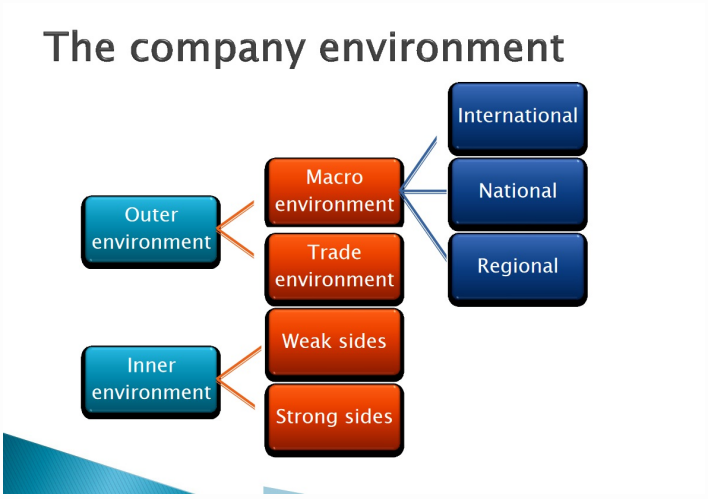
05 Základy managementu

Základy analýzy řízení

Management

- Proces dosažení souboru daných cílů pomocí jiných lidí
- 5 fází:
  - Vytváření cílů
  - Plánování
  - Realizace
  - Kontrola
  - Koordinace

Analýza prostředí



prostredi

Makroprostředí

Část vnějšího prostředí - faktory založené na mezinárodním, národním a regionálním prostředí.

PESTE

Framework používaný k analýze a monitoringu faktorů makroprostředí - Political, Economical, Social, Technological, Ecological

Mezinárodní prostředí

Globalizace podnikání

Národní prostředí

Národní kultura. Stakeholders

6C

Obchodní podmínky v Národním prostředí: - Country, Corporations, Customers, Costs, Competitors, Currency

Regionální prostředí

Důležité pro malé firmy.

Obchodní (oborové, odvětvové prostředí)

Obor představuje skupina firem produkující stejné, nebo navzájem zaměnitelné produkty. 3C - Consumers, Collaborators, Competitors

Vnitřní odvětví

Vnitřním prostředím rozumíme množinu prvků a jejich vzájemných vztahů existujících uvnitř organizace. Pro jeho pochopení je účelné organizaci definovat jako sociálně-technický systém, jehož prvky jsou lidé a věcné prostředky propojené vzájemnými komunikačními a řídícími vazbami. Silné a slabé části.

SWOT

SWOT analýza je metoda, jejíž pomocí je možno identifikovat silné a slabé stránky, příležitosti a hrozby spojené s určitým projektem, typem podnikání, …

SWOT strategie

Na základě SWOT analýzy vypracujeme jednotlivé strategie. - S-O (maxi-maxi) - S-T (maxi-mini) - W-O (mini-maxi) - W-T (mini-mini)

Zájmové skupiny a jejich zájmy

Shareholders (akcionáři) **Stakeholder** (zúčastněné strany)

Management podle kompetencí, teorie vitality

Vitalita - schopnost firmy soustavně dosahovat úspěchu, jediný kritický faktor jsou lidé. **Svět požadavků** (to, co chceme) vs. **Svět možností** (to, co můžeme).

Konečným cílem MbC je dosáhnout vitality!

Vitalita = možnosti + požadavky

Kompetence

Kompetence osoby je součtem **pracovních výkonů** a **potenciálu**.

Firemní kultura

Soubor vztahů mezi klíčovými faktory nezbytnými pro společnost. - Prvky kultury - symboly, hodnoty, rituály, hrdinové

Teorie vitality

Teorie vitality popisuje strategii budování vitální firmy. - Usefulness - Víme jak a pro koho jsme užiteční - Effectivity - Děláme věci efektivně - procesy/produkty - Stability - Dokážeme reagovat na změny - Dynamics - Víme, jak se naše odvětví vyvíjí

Teorie omezení (Theory of constraints (TOC))

Teorie omezení je proces zlepšování jakékoliv metodologie tím, že identifikujeme "system constrain" (bottleneck). Idea je taková, že každý systém má limitující faktor a zaměřením se na tento prvek je běžně nejefektivnější cesta, jak zvýšit profitabilitu.

Pyramida kultury

Pyramida kultury nám pomáhá ve vedení zaměstnanců ve firmě. - **Definice** - Definice firemních myšlenek - **Orientace** - Seznámení lidí s definicemi firemních myšlenek -

Motivace - Dosažení souladu firemních myšlenek s potřebami jednotlivců = akceptace korporátních idejí - **Habilitace** - Vytvoření rovnováhy mezi požadavky a schopnostmi lidí jejich rozvojem (vzděláváním), po habilitaci mají lidi potřebné schopnosti - **Synergetizace** - Budování spolupráce k dosažení synergetického efektu (výsledek spolupráce je hodnotnější, než pouhý součet práce jednotlivců) - **Integrace** - Integrace se zaměřuje na lidské vlastnosti. Nejde o jejich změnu, ale snahu umístit je ve firmě tak, aby synergickému efektu nebránily.

Strategická orientace společnosti a plánování

Strategický rámec

Definuje společnost a její podnikání. Je zárodkem, ze kterého vycházejí myšlenky společnosti. Skládá se z 5 částí: - **Business hypotéza** - Identifikuje obchodní příležitosti - **Vize** - Odpoví na otázku: "Jak bude vypadat podnik a společnost ve vzdálené budoucnosti?" - **Mise** - Označuje výhody pro zákazníky, dodavatele, zaměstnance a měla by přitahovat pozornost - **Values and Rules** - Definuje způsoby a hranice, v nichž se společnost bude pohybovat, aby dosáhla svých vizí. - **Strategie** - Definuje konkrétní obchodní aktivity, které umožňují udržet úspěch společnosti v současnosti i v budoucnosti

Strategické kontinuum

Strategické kontinuum znázorňuje, jak by firma měla řídit své konkurenční výhody v čase. Pomáhá firmě orientovat se v současné strategii a připravovat si budoucí strategii s budoucími konkurenčními výhodami.

Hodnocení lidí a motivace

Krátkodobé - rozhovor zaměřený na kvalitu provedených úkolů v posledním období. Dlouhobé - analýza výkonu za účelem nalezení optimalizace s ohledem na podnikové požadavky a možnosti pracovníka.

Motivace

Motivace je o dosažení harmonie toho, co cítí člověk jako své vnitřní potřeby.

Stimulace

Činnost, která je pro lidi vyžadována, se provádí v souvislosti s obecnými pracovními stimuly (např. finanční pobídky).

Strategické řízení, synergie a její role v řízení a vedení

Strategické řízení

Oblast managementu zaměřená na dlouhodobé plánování a řízení firmy. Celý proces managementu probíhá ve 4 primárních, opakujících se fázích (tzv. strategický cyklus): 1. **Formulace strategie** - mise, vize, cíl 2. **Strategické plánování** - vymyšlení strategického plánu a plánování implementace 3. **Implementace strategie** - alokace zdrojů, aktivity, měření pro dosažení cílů 4. **Monitoring a hodnocení strategie**

Synergie a její role v řízení a vedení

Synergie ve společností řízené lidmi: - Řízení ovlivňují názory nadřízených

Synergie ve společnosti poháněné nápady: - Rozhodují vize, cíle a úkoly firmy

Role v řízení a vedení

Role se nesmí zaměňovat s osobami!

Klíčové role: - Lídři - Definují strategický rámec a přesvědčují ostatní o jeho významu - Manažeři - Mají pochopit a přijmout strategický rámec a definovat následné požadavky - Dělníci - Dosahují cílů a plní úkoly

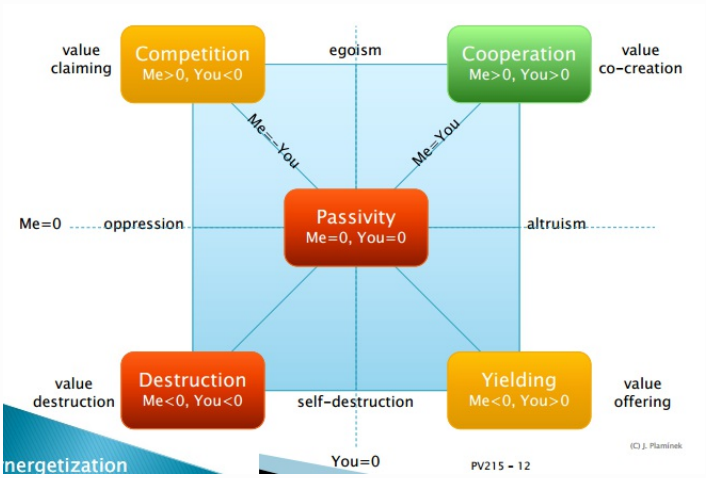
Management společnosti

Management společnosti vyjadřuje synergii mezi vedoucími, manažery a pracovníky. Dobře vést znamená dělat správné věci. Dobře řídit znamená dělat věci správným způsobem.

Synergie

Při synergii výsledek spolupráce převyšuje výsledek prací jednotlivců.

Vztahy



relationships

Řízení rizik a jeho principy

Řízení rizik

Řízení rizik je soubor činností a opatření umožnující snížit ztráty a případně jiné následky vyvolané rizikovými událostmi. Riziko je pojem, který označuje nejistý výsledek s možným nežádoucím stavem. - 5 fází - Identifikace, analýza, zhodnocení, ošetření/zvládnutí, monitoring rizik.

---

06 Projektové řízení

Set metod, technik, nástrojů, kompetencí, které se aplikují na projekty, aby splnili projektové požadavky, naplnili svůj cíl a pomáhají udržet rovnováhu mezi COST-TIME-SCOPE

Projekt

Časové ohraničené úsilí (*temporary*), které má unikátní vlastnosti (*unique*), dochází během něho ke změnám (*change driving*), nachází se v něm určitá míra nejistoty (*uncertain*) a ultimátně směřuje k dosáhnutí cíle. - Vizualizovatelné pomocí Gantt chartu.

Proces

Sled samostatných činností, které na sebe navazují a na základě vstupních požadavků vytvářejí výstup. Procesy jsou opakovatelné, lehce monitorovatelné a měřitelné, odladěné a vyzkoušené. - Vizualizovatelné pomocí Flow chartu.

PPP - Portfolio, Program, Project

Portfolio - Probíhající set projektů a programů, které spolu směřují ke strategickému cíli. **Přidává hodnotu businessu**. **Program** - Set dočasných projektů, které mají společné vlastnosti a směřují ke společnému cíli. **Přináší hodnotu stakeholderům**. **Project** - Dočasné úsilí vynaložené na vytvoření jedinečného produktu, služby nebo výsledku.

Standard projektového řízení

Využívají se, abychom dosáhli lepších výsledků, pracovali efektivně, zvýšili transparenci a "nevynalézali znovu kolo."

PRINCE2 = Project in Controlled Environments

Procesně orientovaná metodika řízení projektu. Má preskriptivní charakter. Jinými slovy je to *step-by-step formula* pro úspěšný projekt. - 7 základních principů - Learn from experience, Focus on products, … - 7 témat - Business case - WHY, Organization - WHO, Quality - WHAT, … - 7 hlavních procesů - Starting up a project (SU), Directing, Initiating Project, …

PMI PMBOK = Project Management Institute (PMI) Project Management Body of Knowledge

Procesně orientovaná metodika (podobně jako PRINCE2), ale dává větší volnost. Cíle je dosahováno pomocí definovaných procesů, každý proces má určeny své vstupy a výstupy a techniky a návody, jak by měl být prováděn - 10 Knowledge areas - Cost, Quality, Scope, Resource, Communication, … - 5 Process groups - Initiating, Planning, Executing, Closing, Monitoring & Controlling - 49 Processes - Develop a Project charter, Collect requirements, …

IPMA ICB = International Project Management Association (IPMA) Individual Competence Baseline.

IPMA ICB je **Competence-based** přístup. Její gró je takové, že popisuje jednotlivé kompetence, které by měl projektový manažer splňovat. Má 3 oblasti kompetencí a pro každou jsou popsané požadované dovednosti i způsoby, jak je změřit. - 5 Perspective competencies - Strategy, culture and values, compliance standard, … - 10 "People" competencies - Leadership, teamwork, … - 13 Practice competencies - Finance, stakeholders, project desing, …

Procesní skupiny v projektu

V PMBOK jde o logické seskupení do inicializace, plánováni, provedení, monitoring a kontrola a Closing (ukončení). - **Initiating phase** - **Planning phase** - Vytvoření budgetu + risk assessment + project schedule - **Execution phase** - **Monitoring & Controlling phase** - **Closing phase**

Životní cyklus IT projektů

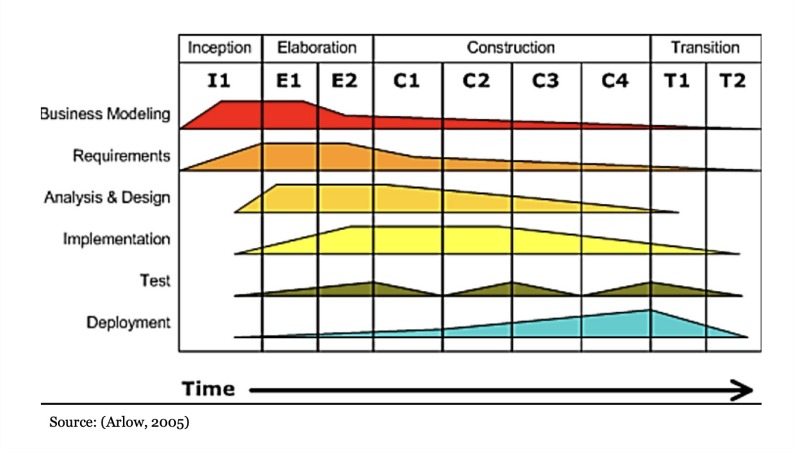
Životní cyklus se skládá z jednotlivých etap řízení IT projektů. IT projekty se typicky realizují **iterativně-inkrementálním** způsobem. Výsledek projektu je součástí služeb a produktů, které firma nabízí a vyžaduje další management. Např. vodopád, spirála, etc.

Přístupy k SW developmentu

Prediktivní přístup

Rigidní, zaměřuje se na procesy a má fixní požadavky.

Metodika Unified Process (UP)

- **Prediktivní vývojový framework, kde víme, jaký bude náš konečný výsledek. Je iterativní a inkrementální.**
- 4 fáze, 6 workflows
- 

Source: (Arlow, 2005)

Agilní

Flexibilní, zaměřuje se na lidi, požadavky jsou pravidelně aktualizované, neplánuje moc dopředu.

SCRUM

Agilní, iterativně inkrementální metoda. - 5 Eventů - 3 Artifact - 3 Roles

Plánování projektu

Project charter

Dokument, který formálně autorizuje existenci projektu a poskytuje projektovému manažerovi autoritu použít organizační zdroje na aktivity projektu. Poskytuje ho IPMA - Business case = WHY - Outcome = What - Stakeholder = Who - Approach = How - Schedule = When

Výpočet nákladů na projekt

- Pomocí WBS a PERTu zjistíme cenu za jednotlivé work package = *Project estimate*
- K *Project estimate* přidáme *Contingency reserve*, kterou jsme získali z kvantitativní analýzy rizik, tím vzniká *Cost baseline*
- Ke *Cost baseline* přidáme *Management reserve* (5-10%)
- Dohromady tvoří *Cost budget*

PERT

Pravděpodobnostní technika pro odhadování potřebného času ke splnění tasku. - 3 typy časů - optimistický, pesimistický, nejpravděpodobnější - $te = (o + 4m + p)/6$

Síťová analýza

Síťová analýza je nástrojem pro analýzu a řízení projektů. Modelem projektu je síťový graf. **AOA** - Activity on arrow - Činnosti jsou reprezentovány hranami grafu, zatímco uzly grafu představují stavy projektu - používá se častěji. **AON** - Activity on node - Uzly představují činnosti a hrany grafu reprezentují návaznosti činností.

Gantt chart

Ganttův diagram je nástroj pro projektové plánování. Osa X reprezentuje čas a osa Y plánované aktivity. - Finish-to-start - Aktivita B nemůže začít, dokud neskončí aktivita A. - Finish-to-finish - Aktivita B nemůže skončit, dokud neskončí aktivita A. - Start-to-start - Aktivita B nemůže začít, dokud nezačně aktivita A. - Start-to-finish - Aktivita B nemůže skončit, dokud nezačně aktivita A.

Critical Path Method (CPM) - Kritická cesta

Metoda projektové řízení, která pomáhá s plánováním projektu. Používá se ke spočítání kritické cesty, což je sekvence tasků, které musí být splněny včas, jelikož jakékoliv jejich zpoždění by mělo za následek zpoždění celého projektu. - Dopředný a zpětný přechod

WBS (Work breakdown structure)

Hierarchická dekompozice práce, která se má vykonat na celém projektu. Celý projekt dekomponuje na části (work packages), což jsou nejmenší dekomponovatelné jednotky.

Zajištění kvality projektu, testy, přezkoumání, měření a standardy

Měření

Měření projektu: - AC (Actual Costs) - celkové náklady, které byly dosud spotřebovány. - PV (Planned Value) - plánované náklady na vytvoření produktu k datu kontroly. - EV (Earned Value) - poměrová hodnota z nákladů plánovaných na úkol odpovídající procentu dokončenosti úkolu. - SPI (Schedule performance index) - poměr množství hotové práce / množství plánované práce. SPI < 1 - projekt se zpožďuje, SPI > 1 - projekt je v předstihu. Počítá se SPI = EV / PV - CPI (Cost performance index) - kolik budgetu se zatím utratilo v porovnání s hodnotou, co zatím vznikla. CPI = EV / AC. CPI > 1 - úspora nákladů.

Zajištění kvality

Se zajištěním kvality souvisí hlavně hrozba rizik, které mohou projekt značně poškodit. Řešíme kvalitativní a kvantitativní část. Kvalitativní - Risk matice - poměr pravděpodobnosti a dopadu Kvantitativní - číselné vyjádření rizik - vytvářím z něj *Contingency reserve*.

---

07 Řízení IT služeb

ITSM

IT service management řeší jak nejlépe poskytnout IT služby zákazníkovi. Má za cíl zajistit, aby při poskytování IT služeb byly použity správné procesy, lidé a technologie způsobem, který umožní dané společnosti dosáhnout cílů.

Služba

Nehmotný prostředek doručení určitého výstupu zákazníkovi, aniž by na něj přešlo vlastnictví rizika a s ním spojené náklady.

Proces

Soubor akcí, které jsou vykonávány za účelem dosažení konkrétního výsledku, který má zákazníkovi poskytnout hodnotu. Má vstupy a výstupy.

ITSM frameworky

COBIT, MOF (Microsoft Operations Framework), **ITIL** (IT Infrastructure Library)

Gartnerův I&O Maturity Model

Vyhodnocuje úroveň vyspělosti řízení služby ve firmě Zohledňuje lidi, procesy, technologie a obchodní management Obsahuje 6 úrovní: - Level 0 - Survival - Level 1 - Awareness - Level 2 - Committed - Level 3 - Proactive - Level 4 - Service-aligned - Level 5 - Business partnership

RACI matice

Nástroj pro přiřazení kompetencí k jednotlivým krokům procesu. * responsible, accountable, consulted, informed

4P

Faktory, které jsou klíčové pro marketing služby. Také označovány jako marketing mix. * Product * Place * Price * Promotion

ITIL

- Mezinárodně uznávaná sada postupů pro řízení IT služeb. ISO 20000 JE standard.
- ISO řeší *CO?* ITIL řeší *JAK?*
- **Výhody**:
  - Vyšší kvalita, nižší náklady – centralizace a standardizace služeb
  - Větší kontrola nad IT infrastrukturou
  - Flexibilita – procesy mohou být přizpůsobeny
  - Uspokojení zákazníka – zlepšuje organizační schopnosti
  - Škálovatelnost – pro firmu s 6 i 1000 lidmi.
- **Hlavní prvky:**
  - Procesy
  - Funkce (tým a zdroje na vykonávání činností)
  - Role (množina povinností pro tým nebo jednotlivce)
    - Vlastník služby (service owner)
    - Vlastník procesu (process owner)
    - Manažer procesu (process manager)
    - Vykonavatel procesu (process practitioner)
- **5 základních fází života služby:**
  1. **Strategie služby** – *CO? PROČ?* Co a komu nabídneme službou, jakou přinese hodnotu? :arrow_right: výsledek: dokumentace požadavků i s výsledky
  2. **Návrh služby** – *JAK?* Detailní parametry služby, rizika, bezpečnosti, dostupnost, … Zahrnuje technologie a postupy
  3. **Přechod služby** – vytvoření, otestování, nasazení služby. Patří sem např. change mgmt
  4. **Provoz služby** – aktivity k spolehlivému provozu služby. Patří sem např. service desk
  5. **Neustálé zlepšování služeb** – vylepšování a aktualizace podle potřeb trhu
     - Demingův cyklus:
       - PLAN – jaké metriky, *CO* a *PROČ* zlepšovat
       - DO – sběr dat pro metriky a nastavení KPI
       - CHECK – analýza dat, dasahujeme cíle? Jasné trendy a náklady?
       - ACT – implementace zlepšení

SLA - Service level agreement

- Dokument popisující úroveň služeb, které budou poskytovány.
- Objasňuje *CO?* (service elementy), *JAK?* (management elementy)

Outtasking

- Flexibilní a rychlé rozšíření pracovních zdrojů
- Zodpovědnost nad úkolem zůstává internímu týmu
- Nevhodné jako dlouhodobé řešení
- Např. specialisti na databáze, sítě, záloh (outtaskuju údržbu, která má malý přínos, interní tým se soustředí na jádro věci)
- Plná kontrola nad úlohou

Outsorcing

- Outside resource using
- Delegace úlohy včetně zdrojů na třetí stranu (např. telefonická podpora) za měsíční/roční poplatek
- Nižší náklady (mizivá režie), flexibilita, přesun rizika
- Definován několika SLAs

Modely Služeb

Popisuje strukturu (jednotlivé prvky potřebné k doručení služby a jejich konfigurace) a dynamiku (aktivity, tok zdrojů, koordinaci a interakci mezi zákazníkem a prvky ve struktuře). - SaaS (Software as a Service) - taxi - PaaS (Platform as a Service) - car renting - IaaS (Infrastructure as a Service) - car leasing