

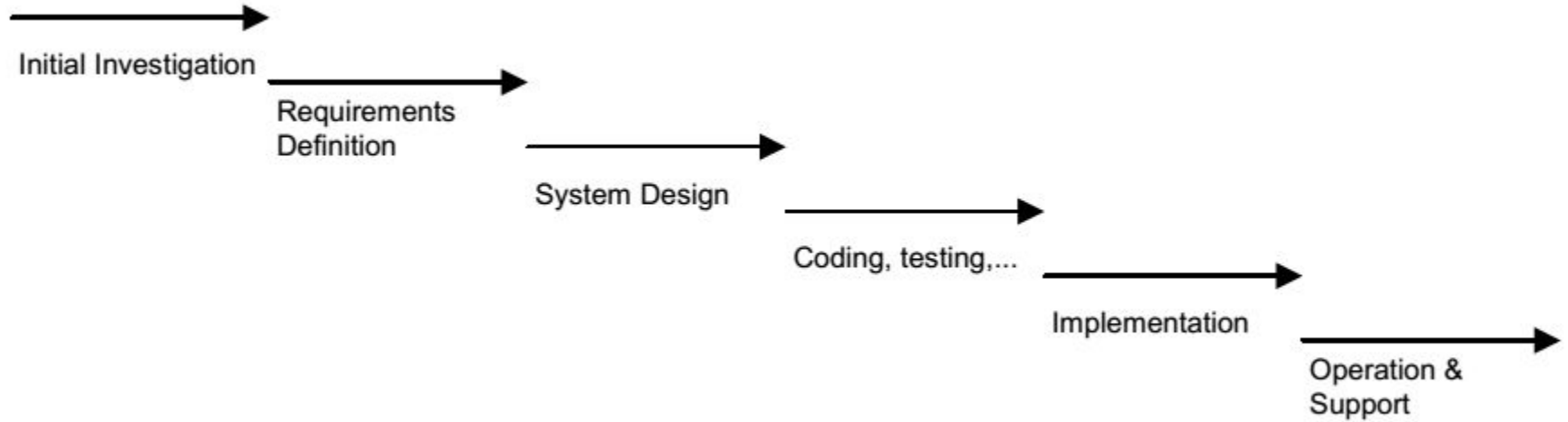
Další strategie vývoje SW

Prototypování není jediná možnost.

Další alternativy k vývoji interaktivních systémů:

- Waterfall
- Inkrementální vývoj
- Spirála
- Rapidní vývoj (RAD)
- Scrum
- ...

Waterfall



Waterfall

- lineární průběh (možné překrývání jednotlivých fází, ale nelze se vracet)
- důraz na kontrolu během celého vývoje (rozsáhlá dokumentace),
- použití při komplikovaných a finančně náročných řešeních s pevně daným cílem, nutností jsou jasně vymezené milníky a přesně definovaná cílová skupina se znalostí problému, je také nutné mít přesně zmapované její potřeby

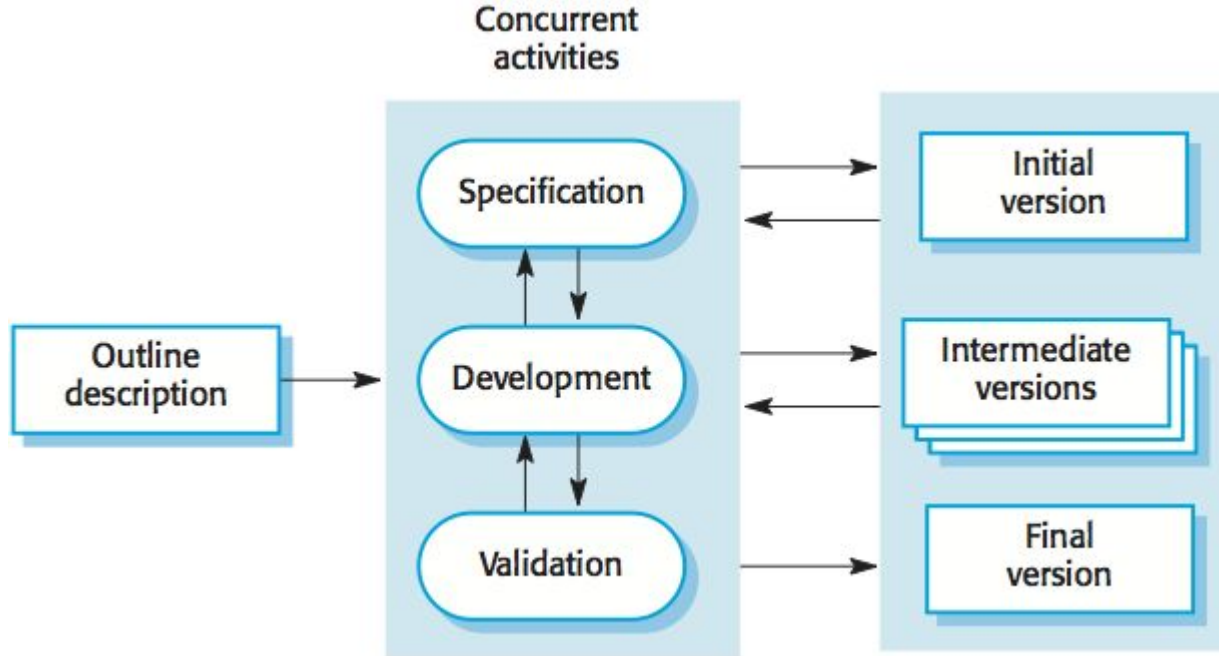
Klady

- měřitelnost výkonu v průběhu vývoje, přehlednost vývoje v jednotlivých fázích, lineárnost je dobrá pro začátečníky, je možné udržet náklady ve stanovené výši

Zápory

- pomalý průběh, téměř žádný prostor pro iteraci a případné změny, častá identifikace problémů až ve fázi vývoje rozhraní nebo testování, nízká flexibilita, časově náročný proces tvorby dokumentace a specifikace pro uživatele

Inkrementální vývoj



Inkrementální vývoj

- kombinace iterativního a lineárního vývoje pro větší flexibilitu a toleranci změn, např.:
 - 1) mini-waterfall pro každou část lineárního vývoje;
 - 2) koncept, požadavky a architektura jsou vyvinuté pomocí strategie waterfall, zbytek se prototypuje až do finálního produktu
- použití u velkých projektů, které jsou vystaveny vnějším i vnitřním změnám, u webových aplikací nebo výzvy s velkým potenciálem

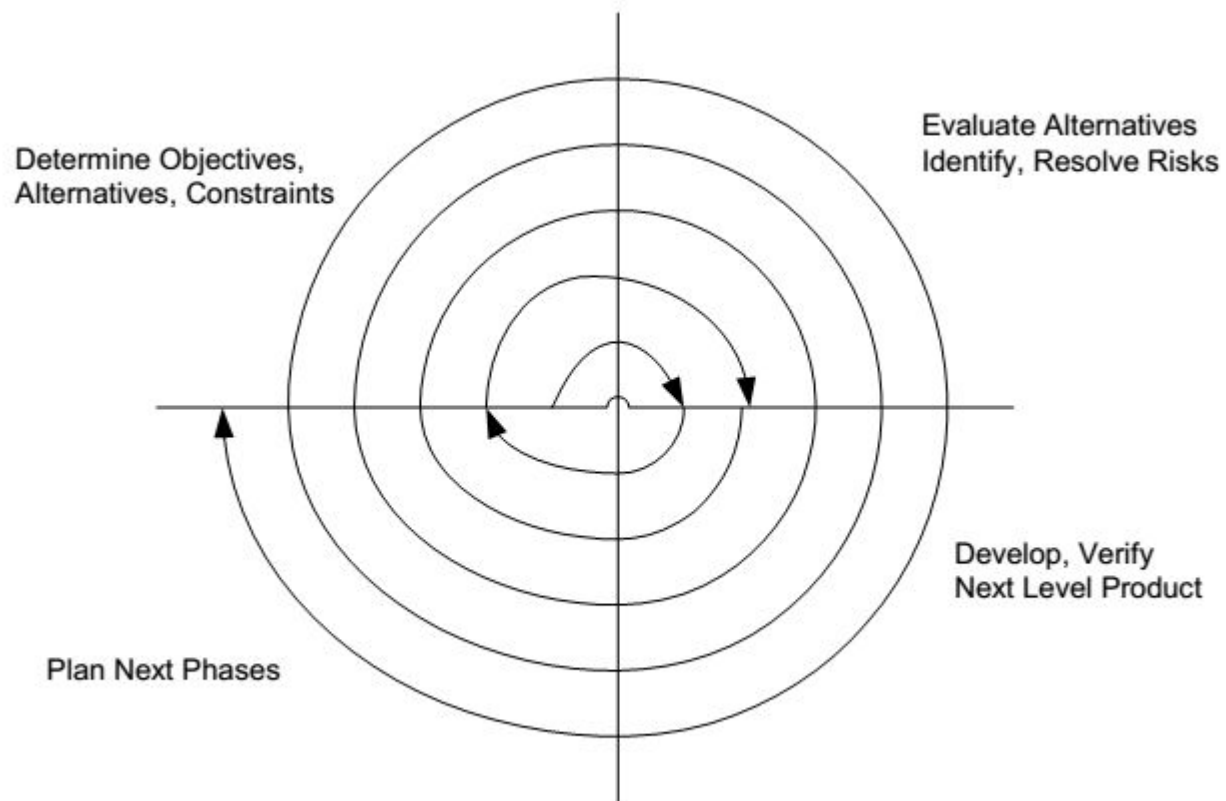
Klady

- snížení rizik selhání produktu, možnost získat informace o produktu již na počátku vývoje (následuje porovnání), zadavatelé mohou vidět přibližnou podobu výstupů již v průběhu vývoje

Zápory

- při tříštění do jednotlivých částí může dojít ke ztrátě přehledu o celku, problémy s dodržováním času a odsouváním rizik někam do budoucna („Ono se to udělá.“)

Spirála



Spirála

- hlavním cílem je minimalizace risků
- výstupem z každé fáze (pomyslného obkroužení spirály) je produkt, jehož komplexita postupně narůstá a jsou odstraňovány možné problémy
- každý běh čtyřmi kvadranty začíná identifikací stakeholderů a končí reflexí procesu
- použití při potřebě co nejvíce minimalizovat rizika, za přítomnosti vyškoleného týmu a dokumentace procesů, když je přesnost prioritní stejně jako implementace

Klady

- omezení risků, integrace ostatních strategií a z toho plynoucí výhody a flexibilita

Zápory

- složitost při propojení s ostatními strategiemi, závislost na konkrétním projektu - nelze použít znovu, nutnost nastavení kontroly a milníků pro přechod mezi fázemi, inklinace k waterfall

Rapid Application Development (RAD)

- iterativní charakter
- hlavní myšlenka: kvalitní řešení rychle a za málo peněz vyvíjené s uživateli
- rozdělení projektu na části, použití prototypování v kombinaci s nástroji vývoj produktu za běhu (Photoshop a jiné grafické nástroje, objektové modelování, UML, programovací jazyky, databázové systémy)
- důraz na zapojení uživatelů a časové hledisko: deadline se nepřizpůsobuje reálnému stavu vývoje, ale naopak
- použití při vývoji malých až středních projektů v limitu max. 6 měsíců s jasně vymezeným účelem a ideálně vyškolenými členy týmu, kteří v projektu působí kontinuálně a jsou schopni evaluovat produkt společně s uživateli

Rapid Application Development (RAD)

Klady

- rapidní strategie s levnými výsledky, za které se platí energií a vůlí vývojářů a stakeholderů oddat se procesu vývoje, podstatnou je dále schopnost adaptovat produkt na různé změny a přítomnost uživatelské perspektivy

Zápory

- kvůli důrazu na rychlost a nízkou cenu logicky klesá kvalita, riziko zahcení funkcionalitou, riziko nekonzistence kódu a modulů, budoucí nemožnost do systému zasáhnout, složitosti při implementaci formálních prvků (audity, hodnocení)

SCRUM

The Agile: Scrum Framework at a glance

Inputs from Executives,
Team, Stakeholders,
Customers, Users



Sprint Planning Meeting



Sprint Backlog



Sprint end date and team deliverable do not change



Scrum

- agilní rámec pro vývoj, hlavním cílem je rapidní vyvinutí prototypu nebo použitelného produktu, který se následně iteruje
- na začátku zadavatelé vytyčí cíle, z nichž vedoucí týmu (neboli tzv. Scrum-master) vybírá ty, které se splní, následuje sprint (max. 4 týdny), během něhož všichni členové týmu intenzivně pracují na projektu a každý den probíhají malé evaluace
- kontrola je nejdůležitějším prvkem (pozorování, vytváření backlogs neboli záznamů)

Klady

- transparentnost, rychlost, komplexní přístup, otevřenost pro zapojení ostatních metod a technik

Zápory

- náročnost na vedení týmu a dodržování stanovených cílů, riziko nepřesnosti a nízké kvality způsobené spěchem



Diagramy jinak: Software development explained with cars toggl.com/developer-methods-infographic

Díky za pozornost!
Dotazy?

415437@mail.muni.cz

@petrkalisek

Literatura

Selecting a development approach. In: *Centers for Medicare & Medicaid Services* [online]. USA: Centers for Medicare & Medicaid Services, 2005 [cit. 2016-02-20]. Dostupné z: <https://www.cms.gov/research-statistics-data-and-systems/cms-information-technology/xlc/downloads/selectingdevelopmentapproach.pdf>

SCHWABER, Ken a Jeff SUTHERLAND. The Scrum Guide™: The Definitive Guide to Scrum: The Rules of the Game. In: *Scrum Guides* [online]. 2013 [cit. 2016-02-20]. Dostupné z: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100>