

OOT

Objektově orientované technologie

Požadavky na systém

Daniela Szturcová

Co jsou to požadavky?

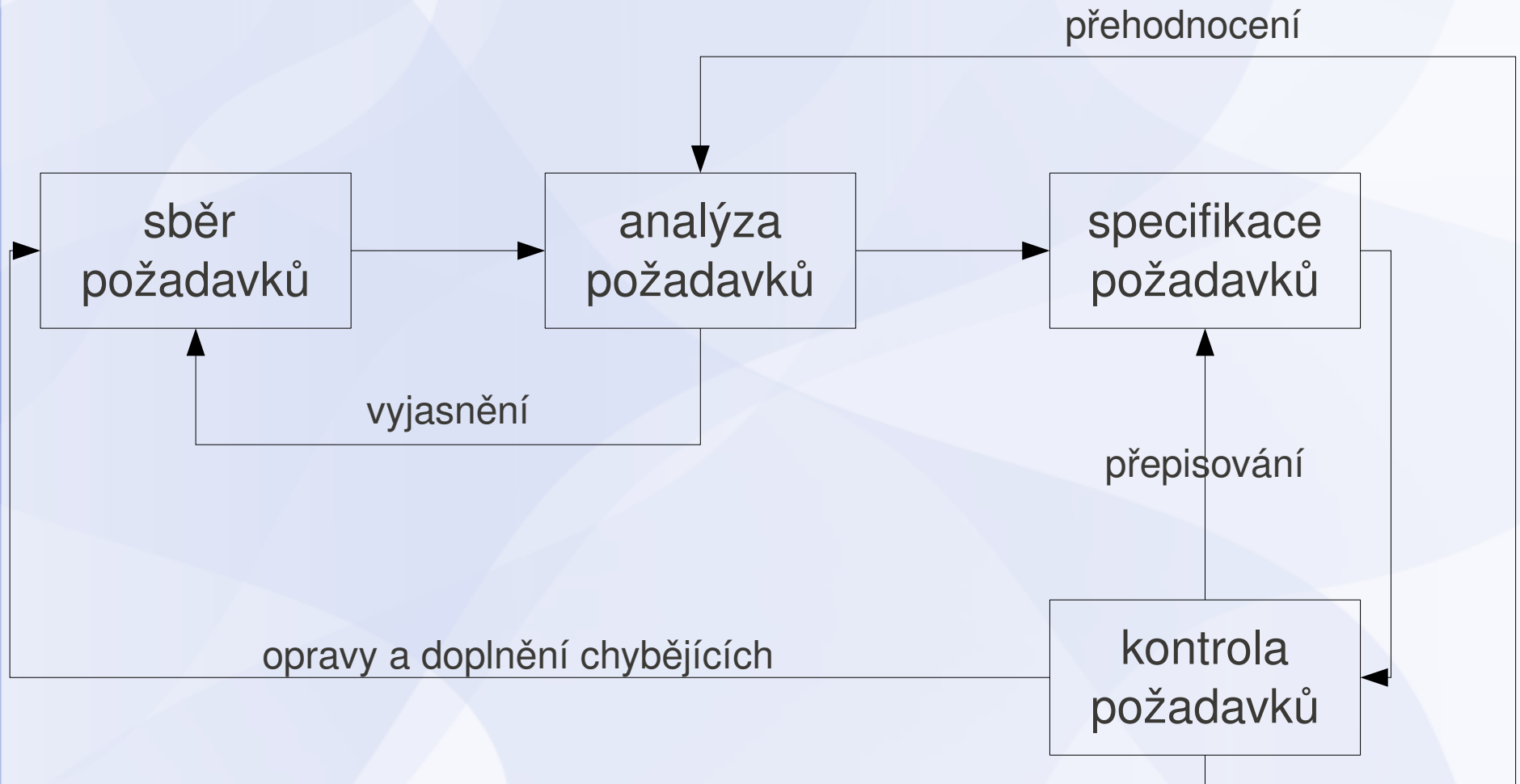
Účelem požadavků je popsat, **CO** by mělo být implementováno.

- Jaké by mělo být chování systému.
- Vlastnosti systému.
- Omezení kladená na systém.

Typy požadavků

- Podnikatelské
 - Uživatelské
 - Funkční
 - Nefunkční
-
- Pochází z jiných zdrojů i fází projektu
 - Funkční by měl mít jasnou vazbu na uživatelský, podnikatelské by měly obsahovat všechny uživatelské

Vývoj požadavků



Sběr požadavků

- Popis vize a rozsahu projektu
- Hledání tříd uživatelů a jejich vlastností
- Výběr produktového šampióna z každé třídy
- Hledání případů užití
- Hledání systémových událostí a reakcí na ně
- Dílna požadavků – společně: zainteresovaní & analytici
- Sledovat uživatele při práci, zlepšovány starého systému, recyklace požadavků předchozího projektu

Analýza požadavků

- Upřesňování požadavků tak, aby jim rozuměli všichni zúčastnění
- Hledání chyb a mezer
- Rozpracování obecných požadavků do podrobností, prototypování
- Analýza proveditelnosti
- Jednat o prioritách požadavků

Analýza požadavků – postup 1/2

- Kresba kontextového diagramu
- Tvorba prototypu – technického, UI – pomůže vytvořit společnou představu o systému
- Proveditelnost požadavků – náklady a rizika implementace, přijatelný výkon, konflikty mezi požadavky
- Stanovit priority – dle UC, funkcí systému a požadavků – při žádaných změnách nutno vyhodnotit

Analýza požadavků – postup 2/2

- Vytvořit model požadavků – ERD, DFD, ClassD, StateD, SeqD, Mapa dialogů, Rozhodovací strom nebo tabulka
- Vytvořit datový slovník
- Rozdělit požadavky mezi podsystémy
- Quality Function Deployment(QFD) – analýza vztahů mezi funkcemi a vlastnostmi systému (očekávané&důležité, ale nezmíněné; běžné; bonusové)

Specifikace požadavků

- Zapsat požadavky:
 - podnikatelské – vize projektu
 - uživatelské – UC, tabulka událostí/reakcí
 - ne/funkční – nejlépe s použitím šablony
- Sledovat a zapsat zdroj požadavku
- Označit požadavek jednoznačným identifikátorem
- Zapsat podnikatelská pravidla
- Zapsat kvalitativní pravidla – nefunkční požadavky

Kontrola požadavků

- Provést revizi požadavků – z různých úhlů pohledu:
 - analytik, zákazník, tester, vývojář
 - ve všech vytvořených dokumentech
- Testovat požadavky
- Definovat kritéria kvality pro přijetí systému:
 - Co bude rozhodující pro splnění očekávání uživatelů?

Zápis požadavku

<id> <system> **bude** (musí umět) <funkce>

System TaxiS bude nabízet benefity VIP
zákazníkům.

Funkční požadavky – **co** bude systém dělat.

"TaxiS automaticky vypočte cenu za provedenou jízdu dle platných
tarifů."

Nefunkční požadavky – **jak** budou v systému
implementovány funkční požadavky či jak bude systém
ovlivněn nároky na kvalitu, rychlost ap.

"TaxiS akceptuje platební karty Visa a MasterCard."

Ne/Funkční požadavky – příklady

- Funkční – Co bude systém dělat:
 - Bankomat bude ověřovat validitu vložené karty.
 - Bankomat bude ověřovat validitu zákazníkem zadaného kódu PIN.
 - Bankomat vydá na každou kartu maximálně 10000Kč/24 hodin.
- Nefunkční – Jak to bude systém dělat:
 - Řídící systém bankomatu bude napsán v jazyce C++.
 - Řídící systém bankomatu bude s bankou komunikovat kanálem zabezpečeným 512bitovým zašifrováním.
 - Řídící systém bankomatu ověří validitu bankovní karty maximálně do tří sekund.
 - Řídící systém bankomatu ověří validitu kódu PIN maximálně do tří sekund.

Taxonomie požadavků

- Uspořádání požadavků do hierarchie podle typu požadavků.
- Užitečné pro případ mnoha požadavků či pro nástroj na řízení požadavků (vyhledávání, třídění dle typu).
- Úroveň rozlišení – hloubka hierarchie.

Taxonomie požadavků

- Funkční – Skupiny (Internetový obchod)
 - Produkty
 - Rozhraní
 - Objednávka
 - Platba
- Nefunkční – hledisko posuzování
 - Výkon
 - Kapacita
 - Dostupnost
 - Shoda se standardy
 - Zabezpečení

Atributy požadavků

Priorita (MoSCoW – dle UP) – ovlivní termín realizace

- Nezbytný (M) – tvoří jádro systému
- Nutný (S) – rozšiřuje jádro o nutné funkce
- Eventuální (C) – bylo by dobré je realizovat, ale systém bude schopen fungovat i bez něj
- Chceme mít (W) – lze chápat jako “třešinku na dortu”
- Význam požadavku – měl by určit doménový expert
- Upřednostnění požadavku – dohoda zadavatele a tvůrce, ovlivní termín implementace

Atributy požadavků (RUP)

- Status
 - Navrženo/Přijato/Odmítnuto/Začleněno
- Benefit(význam)
 - Kritický/Důležitý/Užitečný
- Effort(snaha) – ohodnocení normohodinami
- Risk(riziko) – vysoké/střední/nízké
- Stability(stabilita) – vysoká/střední/nízká
- TargetRelease(cílová verze, upřednostnění)

Hledání požadavků

- Konzultace – důležitá je volba osob
- Dotazníky – mohou chybět důležité informace, které vyplynou z osobního styku
- Dílna požadavků – doporučuje se menší skupina – zástupci obou stran, pouze zapsat nápady, zpřesňování a vyloučení požadavků nechat na fázi analýzy

Hledání požadavků

Noam Chomsky – tři procesy při výběru relevantnosti požadavku:

- **Odstranění** – informace je odfiltrována
- **Deformace** – informace je modifikována souvisejícími mechanismy tvorby a halucinace
- **Zobecnění** – tvorba pravidel, víry a zásad týkající se pravdy a klamu

Tyto filtry jsou mechanismem utvářejícím přirozené jazyky. V případě, že chceme zachytit požadavky a udělat jejich analýzu, je důležité o nich vědět.

Seznam požadavků

ID	Stručný popis	Skupina	Typ	Priorita
R1	TaxiS zobrazí přehledné informace o možnostech rezervace jízdy taxíkem, nabízené typy vozidel - kapacita,	Rezervace	Funkční	M
R2	TaxiS uspořádá seznam způsobu rezervací.	Rezervace	Funkční	M
R3	TaxiS zobrazí detailní popis způsobu, jak provést rezervaci přes vybraný komunikační kanál po výběru kanálu zákazníkem.	Rezervace	Funkční	M
R4	TaxiS zobrazí předpokládanou cenu jízdy dle položek v rezervačním formuláři	Rezervace	Funkční	C
R5	TaxiS akceptuje platbu za jízdu v hotovosti nebo platební kartou obvyklého typu.	Jízda	Funkční	M
R6	TaxiS potvrdí platbu provedenou platební kartou patřičné společnosti.	Jízda	Funkční	M
R7	TaxiS automaticky vypočte cenu za provedenou jízdu dle platných tarifů.	Jízda	Funkční	S
R8	TaxiS akceptuje platební karty Visa a MasterCard.	Jízda	Nefunkční	M

Zdroje, literatura

- Wiegers, K. E.: Software Requirements 2
- J. Arlow, I. Neustadt: UML a unifikovaný proces vývoje aplikací
- M. Fowler: Destilované UML