

Princípy tvorby softvéru

Požiadavky

Robert Lukočka

lukotka@dcs.fmph.uniba.sk

www.dcs.fmph.uniba.sk/~lukotka

M-255

Na čo môžu byť užitočné požiadavky

- Vstup pre dizajn a implementáciu

Na čo môžu byť užitočné požiadavky

- Vstup pre dizajn a implementáciu
- Výber dodávateľa softvéru
- Kontrakt
- Tvorba estimácií, projekt management
- Validácia a verifikácia
- základ dokumentácie / orientácia v projekte
- Management rizík

S požiadavkami prichádzajú do styku takmer všetci stakeholderi.

Zdroje požiadaviek

Okrem iného

- Zákazník (najmä ciele projektu, + veľa iných vecí)
- Znalosti z domény (vnútorné predpisy, legislatíva)
- Ostatní stakeholderi (architektonické požiadanky, požiadavky na implementáciu, testovanie, ...)
- Všeobecné pravidlá a predpisy
- Podmienky nasadenia
- Organizačné prostredie zákazníka

Hierarchia požiadaviek

- Business požiadavky

Chceme znížiť spotrebu papiera ...

- Používateľské požiadavky

Po ukončení práce s dokumentami budú automaticky notifikovaní ďalší ...

- Systémové požiadavky

Tento typ dokumentu po podpise dostane ...

Traceability - stopovateľnosť - je dôležité poznať súvis medzi požiadavkami (a špeciálne súvis medzi požiadavkami z rôznych vrstiev).

- Od ktorých požiadaviek sa má odvíjať kontrakt?

Typy požiadaviek

- Procesné požiadavky
- Produktové požiadavky
 - Funkčné požiadavky
 - Nefunkčné požiadavky

Škatuľkovanie požiadaviek umožňuje kontrolovať, či sme na nič vážne nezabudli.

Typy požiadaviek

Je ich veľa

- Príklad delenia

Je ľahké na niečo pozabudnúť.

Vlastnosti dobrých požiadaviek

- Unitary (Cohesive)
- Úplné
- Konzistentné
- Non-Conjugated
- Stopovateľné
- Aktuálne
- Jednoznačné
- Majú mať špecifikovanú dôležitosť
- **Overiteľné**

Požiadavky hovoria ČO, nie AKO (niekedy ČO=AKO)

Vlastnosti dobrých požiadaviek

- **Úplné** (pre relevantnú časť systému)
 - od všetkých stakeholderov
 - bez šedých zón - zahŕňa aj NEpožiadavky!!!
 - Na zabezpečenie kompletnosti môžu pomôcť šablóny ([IEEE830](#)), delenia požiadaviek na **typy**, a iné nástroje.

Príklady požiadaniek a ich implikácie

Business požiadavka

Chceme znížiť spotrebu papiera ...

- Business požiadavka, zdrojom je zadávateľ projektu
- Je naozaj kvantifikovaná?

Príklady požiadaniek a ich implikácie

Business požiadavka

Chceme znížiť spotrebu papiera ...

- Business požiadavka, zdrojom je zadávateľ projektu
- Je naozaj kvantifikovaná?
- Doležitosť?

Príklady požiadaniek a ich implikácie

Business požiadavka

Chceme znížiť spotrebu papiera ...

- Business požiadavka, zdrojom je zadávateľ projektu
- Je naozaj kvantifikovaná?
- Doležitosť?
- Je merateľná?

Príklady požiadaniek a ich implikácie

Business požiadavka

Chceme znížiť spotrebu papiera ...

- Business požiadavka, zdrojom je zadávateľ projektu
- Je naozaj kvantifikovaná?
- Doležitosť?
- Je merateľná? Je nevyhnutné mať systém merania spotreby papiera.

Príklady požiadaniek a ich implikácie

Po ukončení práce s dokumentami budú notifikovaní ďalší . . .

- Používateľská požiadavka, okrem iného môže odkazovať na predchádzajúcu business požiadavku (ak pôvodne notifikácia prebiehala papierovo).
- Potreba slovníka, môže byť potrebné zavedenie nových pojmov (Typy dokumentov, pozície v procesoch, čo znamená ukončenie práce)
- Ako rýchlo a akou cestou majú byť notifikovaný?
- Aj na tejto úrovni je žiadúce (ale nie nevyhnutné) zachytávať alternatívne vetvy.

Príklady požiadaniek a ich implikácie

Dokumentu typu X po podpise Y dostane do Z.

- Čo sa stane ak dokument nebude podpísaný (nebude podpísaný do daného času).

Spôsoby zachytenia požiadaviek

Ako zachytiť požiadavky?

- “Victorian novel”
- Použitie niektorej zo šablón
- Spreadsheet / iná forma zoznamu
- Issue tracking systém
- Zápisky zo stretnutí so zákazníkom (takýto dokument má nevyhnutne iba krátkodobú platnosť) ...

Problémy a výzvy:

- Koľko dokumentácie chceme vytvoriť a **spravovať**?
- Ako zachovať stopovateľnosť?
- Ako zistiť ktorá požiadavka súvisí s ktorými inými požiadavkami?
- Aký je stav požiadavky?

Sposoby zachytenia funkčných požiadaviek

Jednotka pri funkčných požiadavkách je **use case - prípad použitia**.

Use case

Plný use case má:

- Aktor
- Cieľ
- Úplná postupnosť krokov vrátane alternatívnych ciest
- Kopec iných vecí

Use casey možno skladateľ, rozširovať a zovšeobecňovať (výsledkom).

Use casey - příklad

Příklad

Sposoby zachytenia funkčných požiadaviek

Funkčné požiadavky možno zachytiť rôzno detailne:

- Use case
- Scenár - Obsahuje iba jeden prechod use casom.
- User story - As a <role> I can <capability>, so that <receive benefit>.

Sposoby zachytenia funkčných požiadaviek

Čo ak je use cas príliš rozvetvený?

- UML activity diagram

Use case je dlhý, opakuje sa?

- Môžeme ho rozdeliť, a v use case používať iné use casy.

Use caseov a aktorov je veľa a strácam sa v tom / chem poskytnúť rýchly grafický prehľad use casov systému:

- Uml use case diagram

Proces tvorby požiadaviek

- Identifikácia stakeholderov
- Elicitation
- Analýza požiadaviek
- Špecifikácia
- Validácia

Proces je iteratívny.

Elicitation

- Rozhovory
- Focus Groups
- Dotazníky
- Priame pozorovanie
- Štúdium dokumentácie, legislatívy, ...
- Štúdium podobných projektov
- ...

Analýza a špecifikácia

- Identifikácia aktorov (vrátane vytvorenia potrebných nových abstraktných rolí)
- Prototypovanie
- Matica stopovateľnosti
- Vyhýbať sa Requirements smells
 - Subjektívny jazyk
 - Nejasné prídavné mená, superlatívy
 - Negatívne požiadavky - iné ako NEpožiadavky
 - ...
- Dôslednejšie spísanie často pomôže nájsť slabé miesta
 - Analýza use caseov, alternatívnych scenárov
 - Zovšeobecňovanie

Zdroje

- SWEBOK V3, Kapitola 1
- en.wikipedia.org/wiki/Requirement
- en.wikipedia.org/wiki/Requirements_analysis
- J. Kostičová: Requirements,
<http://www.dcs.fmph.uniba.sk/~lukotka/Requirements.pdf>
- J. Mifsud: Requirements Gathering: A Step By Step Approach For A Better User Experience,
<http://usabilitygeek.com/requirements-gathering-user-experience-pt1/>
- <http://www.uml-diagrams.org/>