

# Intro to Software Engineering

Exam May 28 2020 - Lukáš Bastián

---

## Q1

Aktivity:

1. specifikace
    - část, kdy se zjišťuje, co by mělo být vyvinuto - zjišťují se požadavky zákazníků, jaké funkce má následné dílo mít, hledají se omezení mimo jiné plynoucí ze zákona nebo z podstaty věci (v závislosti na doméně, ve které jsme)
  2. vývoj/development
    - v této části se snažíme splnit to, co se specifikovalo a co chce zákazník
  3. validace
    - jde o kontrolu, zda jsme skutečně vyvinuli to, co zákazník chtěl
  4. další vývoj/evoluce
    - jde o aktivní údržbu a vylepšování díla, kdy se přidávají a upravují funkce podle požadavků trhu nebo i nově vzniklých omezení
- 

## Q2

user requirements (1) VS system requirements (2)

1. user requirements
    - většinou pouze nějaká tvrzení/věty v přirozeném jazyce o tom, jaké služby by měl systém poskytovat; většinou psané pro management, zákazníka nebo end-usery
  2. system requirements
    - strukturovaný dokument s detailním popisem: služeb, funkcí a omezení systému; definuje, co se bude implementovat psané hlavně pro soft. inženýry, vývojáře a i pro end-user
- 

## Q3

Návrh softwarové architektury

- má sloužit jako most/propojení mezi softwarovým designem jako takovým a (SW) požadavky
  - slouží k tomu, aby bylo možné pochopit, jak má systém být strukturovaný v závislosti na požadavcích
  - jde ovšem pouze o nějaký skeleton našeho systému ==> jsou vynechány některé detaily, které by přinesly moc komplexity do návrhu
  - má sloužit jako základ k tomu, aby se dalo o systému nějak bavit diskutovat o jeho složení a funkcích jednotlivých elementů
-

## Q4

### Unit testing

- prováděno vývojáři v průběhu vývoje
  - jde o testování jednotlivých částí/komponent (unit ~ jednotka) v izolaci a mělo by pokrýt co největší část (ideálně všechny) stavů, ve kterých se může nějaká ta jednotka/komponenta nacházet a to včetně přechodů mezi nimi
  - úplnost lze zajistit tím, že se zavedení nějaké nové části podmíní i zavedením odpovídajících testů, které musí projít, než se bude vyvíjet další jednotka
- 

## Q5

Ano, lze.

### Veřejná licence

- umožňuje sdílení za určitých podmínek popsanych v licenci
    - většinou jde minimálně o uvedení zdroje, ze kterého dílo pochází
  - jakmile je něco pod veřejnou licenci, nelze toto licencování odvolat
  - licencování je ukončeno ihned po porušení podmínek
  - jde o snahu vynutit otevřenost obsahu pomocí nějaké smlouvy/licence
  - vztahuje se k ní úzce princip pěti "R" ~ 5R
    - retain
    - reuse
    - revise
    - remix
    - redistribute
- 

## Q6

### Odhadování ceny

1. pomocí plánu projektu
    - odhad na základě vlastností (velikost, složitost) částí, které se mají dodat, dále na základě milestoneů (velikost, časový úsek mezi nimi) a počtem jednotlivých úkonů (tasků), které jsou v plánu
  2. COCOMO
    - empirická metoda
    - založena na počtu řádků
      - kolik tisíc řádků
    - do empiricky získaného vzorce se dá velikost kódu a vrátí počet pracovních měsíců - doba vývoje
- 

## Q7

### 1. Incremental development

- software se vyvíjí po částech, ale k jeho dodání dojde až po dokončení díla

### 2. Incremental delivery

- software se vyvíjí po částech, ty se ovšem co nejdříve dodávají zákazníkovi a to umožňuje rychlé nasazení kritických funkcí a postupně se dostává i na další méně prioritní funkce (umožňuje prioritizaci), zatím ale už běží fungující systém

---

## Q8

### Funkční požadavek

Přesný je dost na to, aby byla funkcionality umožněna, ale ptal bych se dál na omezující podmínky jako:

- Je tato změna časově omezena - jak dlouho před začátkem filmu smí proběhnout poslední změna?
- Jak se zákazník dozví o změně?

### Scénář:

- Film, na který si zákazník rezervoval místa se dostatečně neprodává a došlo ke změně sálu a ten má jiné rozložení sedadel - obsluha se tedy zákazníka může zeptat, zda chce svá místa v novém rozložení změnit, nebo nechat nově přiřazená/namapovaná místa.

---

## Q9

(papír)

---

## Q10

(papír)

---

## Q11

Riziko - Konkurenční systém(y)

Ovlivní - firmu/business

Popis - nový systém se na trh neuchytí, pokud nebude mít dostatečnou přidanou hodnotu oproti už používaným systémům (pokud kina nebudou mít důvod systém změnit) (ano, jde o práci pro zákazníka, pokud se už s někým konkrétně bavím, ale předpoklad je, že nějaký systém už používají)

Pravděpodobnost - high (nemám dost detailů ze zadání, ale jde o invazivní změnu kritické části pro fungování kina)

Dopad - katastrofický

---

# Q12

## Akceptační testování

- došlo by na předvedení funkcionality systému (například ukážeme flow, jak si zákazník může místo rezervovat po prohlédnutí detailu nějaké projekce filmu, na kterou chce jít) a zákazník rozhodne, zda je to pro něj dost a přijme systém v tom stavu a s těmi funkcemi, co vidí, nebo ne.