

Testík k predmetu Tvorba informačných systémov, pondelok, 17.januára 2011, čas max. 120 minút  
V otázkach, kde je výber možností, zakrúžkujte všetky správne odpovede. [spolu max. 109 bodov]

1. Krátko opíšte funkcionality nástrojov CVS a SVN, alebo podobných, prípadne ako sa líšia. [5]

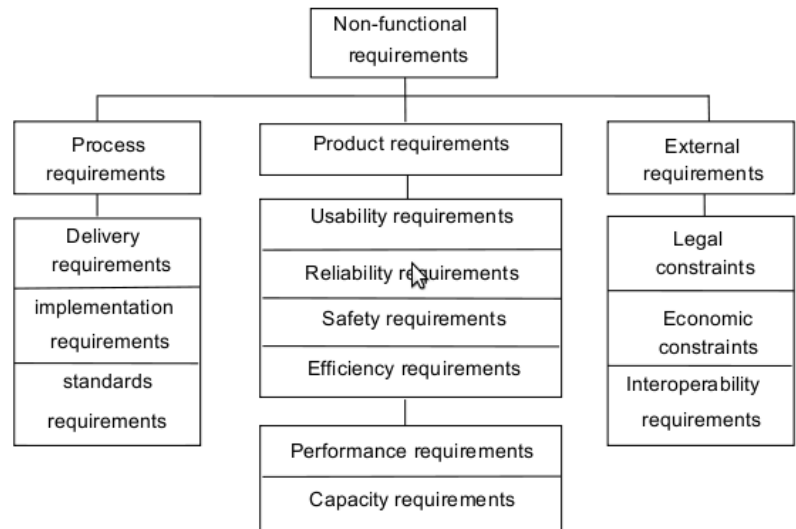
2. Uved'te tri základné druhy atribútov kvality softvéru a vysvetlite rozdiel medzi nimi, uved'te a vysvetlite niekoľko príkladov atribútov kvality softvéru. [7]

3. Uved'te rôzne druhy architektonických pohľadov na systém a na čo slúžia. [7]

4. Počas tvorby katalógu požiadaviek (Requirements Engineering) robia vývojári-analytici viaceré interview. Zakrúžkujte výroky, s ktorými súhlasíte: [7]

- a) Interview môže byť "uzavreté" alebo "otvorené", v otvorenom interview hľadá analytik odpovede na vopred určené otázky.
- b) V uzavretom interview nie je vopred určená agenda.
- c) Analytici robia interview výlučne so zadávateľmi, nie však s budúcimi používateľmi a ich manažérmi.
- d) Pred každým typom interview si treba vytvoriť presnú predstavu, čo budeme požadovať a pevne sa jej pridržiavať.
- e) Zaznamenané požiadavky z rôznych interview by nemali byť vo vzájomnom konflikte.
- f) Všetky reálne požiadavky musia byť prediskutované, politické súvislosti pri katalogizovaní požiadaviek nemôžu byť obmedzujúce.
- g) Vývojári softvéru, budúci používatelia, ich manažéri, externí regulátori, ktorí sú zodpovední za to, že systém neporušuje rôzne pravidlá, doménoví experti s poznatkami a informáciami z aplikačnej domény sú všetci zaangažovaní do tvorby katalógu požiadaviek a označujú sa ako 'stakeholders'.

5. Vysvetlite nasledujúcu schému 5-10 zmysluplnými vetami: [7]



6. Pokúste sa jednou vetou vysvetliť nasledujúce ciele, ktoré sledujeme pri návrhu softvéru: [10]

*Sufficiency:*

*Understandability:*

*Modularity:*

*Cohesion:*

*Coupling:*

*Robustness:*

*Flexibility:*

*Reusability:*

*Information hiding:*

*Efficiency:*

*Reliability:*

7. Vyberte si jeden návrhový vzor a pokúste sa ho vysvetliť. [6]

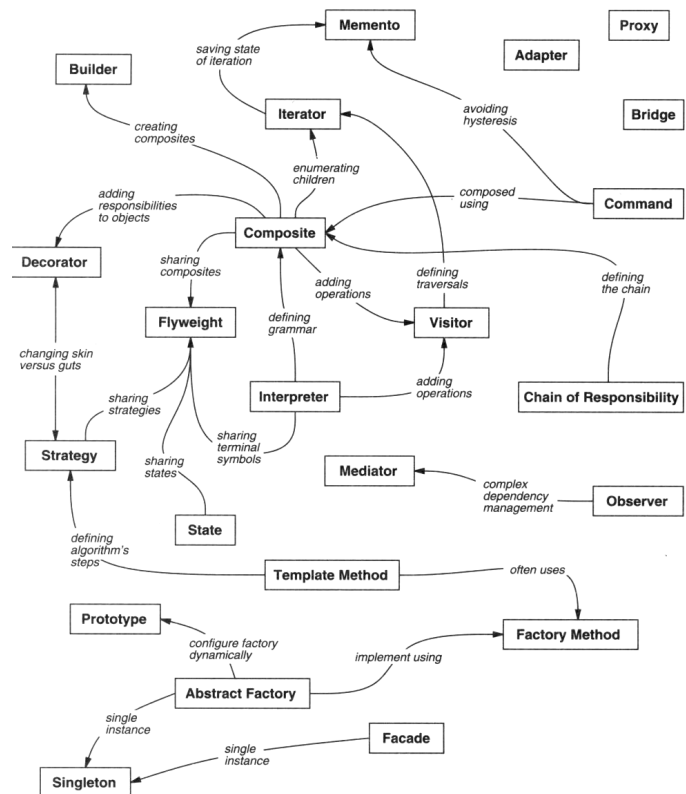


Figure 1.1: Design pattern relationships

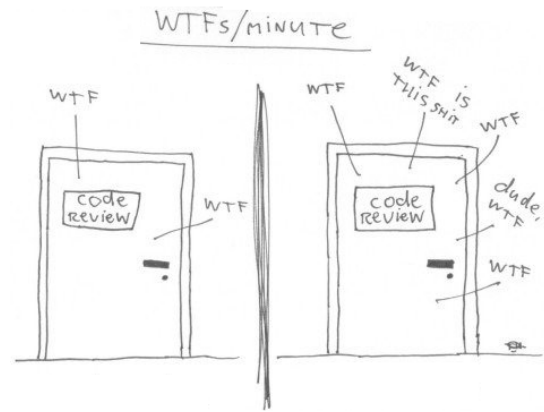
8. Opíšte tri rôzne druhy integrácie existujúcich aplikácií z hľadiska poskytovanej funkčnosti. [7]

9. Najlepšie riešenia integrácie existujúcich aplikácií komunikujú na úrovni (zakrúžkujte): [3]

- a) používateľského rozhrania
- b) aplikačnej logiky
- c) perzistentných údajov

10. Charakterizujte dva základné spôsoby riadenia softvérovej spoločnosti, ich výhody/nevýhody? [5]

11. O čo ide na nasledujúcom obrázku, uveďte a vysvetlite 5 spôsobov, ktoré situáciu vpravo zmenia na situáciu vľavo. [8]



12. Vymyslite si príklad akéhokoľvek informačného systému (iný, ako ten, ktorý ste navrhovali doteraz). Činnosť navrhovaného systému v krátkosti opíšte prirodzeným jazykom (max 10 riadkov). Predstavte si návrh Vášho systému a z tohto návrhu nakreslite: [30]

- diagram tried (*class diagram*)
- diagram používateľských scenárov (*use case diagram*)
- sekvenčný diagram (*sequence diagram*)
- stavový diagram (*state diagram*) alebo diagram činností (*activity diagram*)
- diagram komponentov (*component diagram*)
- diagram rozmiestnenia elementov (*deployment diagram*)

(stačí vždy jeden vybraný príklad pre nejakú časť systému, ku každému diagramu pridajte krátky komentár; celkový návrh nemusí byť kompletný, ale uvedené by malo byť navzájom konzistentné)

*Prémia:* Pripomína Vám niečo nasledujúci obrázok? Ak áno, pokúste sa to charakterizovať. [extra 7]

