

Semestrálne zadanie č.1

Cieľom zadania je vypracovať dokument, ktorý bude obsahovať špecifikáciu softvérového systému, ktorý vám bol pridelený. Samotný dokument bude prehľadný, vhodne štruktúrovaný a formátovaný, s obsahom a prednou stranou. Jazyk dokumentu je písaný odborným štýlom. Použité skratky vysvetlite, prípadne môžete zaviesť aj malý slovník použitých pojmov. Dokument odovzdáte do AIS do **piatka 27.11.2020 23:59 (vrátane)** vo formáte PDF a tiež aj so zdrojom (napr. .doc, .docx, .odt .tex, ...) z ktorého to PDF vzniklo. V dokumente použite normálne riadkovanie (1.0), Times font veľkosti 12pt a okraj 2,5cm (1"). Vymyslíte aj svoj vlastný, marketingovo pôsobivý, komerčne úspešný a jedinečný názov pre vami vyvíjaný softvérový systém. Odovzdaný dokument obsahuje nasledovné časti:

1. Používateľská špecifikácia (cca. 2-3 strany textu) [2b]

Vžite sa do role majiteľa softvérovej firmy, ktorého zákazník požiadal o vytvorenie systému a vytvorte používateľskú špecifikáciu, ktorá bude slúžiť ako súčasť zmluvy.

- 1.1. *Stručný úvod do problematiky.* Tu treba popísať, čo sa v danej oblasti robí, aké sú tam pravidlá, ciele, postupy, aká je business logika (doménová logika) atď. Heslovite je táto informácia zhrnutá v zadaní, ktoré ste dostali, avšak treba ju rozvinúť a doplniť nespomenuté časti a súvislosti. Použijete vlastnú inteligenciu, tvorivosť, externé zdroje a diskusiu s inými ľuďmi, aby ste zistili, ako funguje daná doménová oblasť.
- 1.2. *Používateľské požiadavky.* Definujte zákazníkove ciele a prepíšte ich na merateľné požiadavky. Identifikujte a popíšte funkcionálne, nefunkcionálne a doménové požiadavky.

2. Systémová špecifikácia

V diagramoch použite notáciu UML verzie 2.x

- 2.1. *Diagramy prípadov použitia.* Nakreslite diagram(y) prípadov použitia pre daný softvérový systém. Diagram (minimálne jeden, prípadne viacej ak sa to hodí), bude pomocou prípadov použitia obsahovať *všetku* funkcionálnu kapacitu systému. Každý prípad použitia by mal, v rámci svojej realizácie, poskytovať svojmu hráčovi (alebo hráčom) niečo hodnotné, nejakú užitočnú funkcionálnu kapacitu, nejaký pozorovateľný výsledok alebo zmenu. [4b]
- 2.2. *Popis rolí.* Stručne popíšte role jednotlivých hráčov.
- 2.3. *Use-case tabuľky.* K trom najzložitejším prípadom použitia vytvorte tabuľku, ktorá bude obsahovať [2b]:
 - identifikátor - ako identifikátor môžete použiť svoje vlastné číslovanie, ktoré bude spájať jednotlivé prípady použitia z diagramu s ich popisom
 - názov
 - vstupné podmienky
 - hráčov
 - postupnosť udalostí
 - alternatívnu postupnosť (ak je)
 - poznámky (ak sú)

VZOR: <http://pmblog.accompa.com/2009/10/08/use-case-template-example-requirements-management-basics/>

- 2.4 *Diagramy aktivít a sekvenčné diagramy.* K vybraným netriviálnym prípadom použitia

nakreslite diagramy graficky popisujúce tieto prípady použitia. Nakreslite 2 sekvenčné diagramy a 2 diagramy aktivít. [4b+4b]

2.5 *Stavový diagram*. Nakreslite stavový diagram pre vami vyvíjaný systém a v tabuľkách popíšte jednotlivé stavy a prechody. [3b]

2.6 *Diagram tried*. Vytvorte jeden detailný dátový model pre celý váš systém, ktorý bude zahŕňať všetky atribúty, vzťahy, násobnosti a aspoň niektoré metódy. Zobrazte ho ako jeden UML 2.x diagram tried vo vašej výslednej dokumentácii. Ak je systém príliš komplexný, môžete rozčleniť diagram na viacero menších diagramov, ktoré budú reprezentovať len príslušný podsystém. [5b]

PRÍKLAD:

<http://www.museumsandtheweb.com/mw2001/papers/breiteneder/Breiteneder2.fig2.gif>

3. Akceptačné testy [2b]

Vytvorte testy, na základe ktorých sa rozhodne o tom, či vytvorený systém spĺňa alebo nespĺňa požiadavky – teda či ho zákazník akceptuje alebo nie. Každý test by mal v tabuľke obsahovať minimálne tieto časti:

- identifikátor
- prípad použitia, ku ktorému test prislúcha
- cieľ testu (čo overujeme – nanajvýš stručne)
- vstupné podmienky
- výstupné podmienky
- jednotlivé kroky testu

Kroky testu reprezentujú sekvenciu testovania a ku každému kroku prislúcha a je v teste popísaná určitá akcia (podnet od aktéra) a určitá reakcia systému na tento podnet. Aby bol výsledný systém zákazníkom akceptovaný, musí splniť všetky testy.

Do dokumentácie doplňte aspoň 5 akceptačných testov

- štyri, ktoré súvisia s funkcionálnymi požiadavkami a
- jeden, ktorý overuje nefunkcionálne požiadavky.

PRÍKLAD: <https://uim.fei.stuba.sk/wp-content/uploads/2019/10/AkceptacneTestyPrıklad.pdf>

4. Projektové plánovanie [5b]

Vytvorte plán tvorby vášho systému.

1. Rozdeľte prácu na aspoň 10 úloh a rozdeľte úlohy pre aspoň 4 ľudí tvoriacich váš tím.
2. Odhadnite časovú náročnosť úloh, naplánujte postupnosť úloh do kalendára.
3. V dokumente v kapitole 4.1 zobrazte Ganttov graf aj s tabuľku závislostí a postupnosti vykonávania úloh, míľniky (WBS-work breakdown schedule, dependencies, milestones).
4. V dokumente v kapitole 4.2. zobrazte sieťový graf pre postupnosti vykonávania úloh.

Na túto úlohu použite vami zvolený systém na projektový manažment (či už offline, lokálny program alebo ľubovoľný/dostupný online produkt). Zoznam je napr. na:

http://en.wikipedia.org/wiki/Comparison_of_project-management_software). Úlohou je oboznámiť sa

so systémom na projektový manažment.

Príklad: Microsoft Project, Project Libre alebo google: alternatives to ms project

Zaznamenávanie zmien [1b]

Na začiatku dokumentu, ešte pred obsahom, v tabuľke zaznamenajte dátumy vykonania zmien a stručne popíšte, aké zmeny boli vykonané, tak **aby bolo jasné, kedy nastala zmena v dokumente a aká zmena to bola.**

PRÍKLAD: <https://www.cdu.edu.au/governance/doclibrary/pro-054.pdf>

(napr. tabuľka na poslednej 6. strane) alebo google: document change control