

Испит траје 2.5 сата. Није дозвољена употреба литературе.

1. [7] Одредити број функцијских поена на основу следећих података: ILF=2, EIF=1, EI=5, EO=3, EQ=3. Сви су средње сложености, изузев EQ који су сви ниске сложености. Од 14 фактора, 5 има вредност 1, остали имају вредност 0.

Класификација екстерних улаза				Класификација екст. излаза / екст. упита			
FTR	DET			FTR	DET		
	1-4	5-15	> 15		1-5	6-19	> 19
0-1	Ниска (3)	Ниска (3)	Средња (4)	< 2	Ниска (4)	Ниска (4)	Средња (5)
2	Ниска (3)	Средња (4)	Велика (6)	2 или 3	Ниска (4)	Средња (5)	Велика (7)
> 2	Средња (4)	Велика (6)	Велика (6)	> 3	Средња (5)	Велика (7)	Велика (7)

Класификација интерних логичких фајлова				Класификација екстерних интерфејсних фајлова			
RET	DET			RET	DET		
	1-19	20-50	51 и више		1-19	20-50	51 и више
1	Ниска (7)	Ниска (7)	Средња(10)	1	Ниска (5)	Ниска (5)	Средња(7)
2-5	Ниска (7)	Средња(10)	Велика(15)	2-5	Ниска (5)	Средња(7)	Велика(10)
> 5	Средња(10)	Велика(15)	Велика(15)	> 5	Средња(7)	Велика(10)	Велика(10)

2. [7] Кратко објаснити следеће карактеристике рачунарских облака: еластичност, *multitenancy*, измерена услуга.

3. [16] На слици је дат приказ структуре *html* странице коју генерише скрипта **prijave.php**. Мени (слева) генерише скрипта **menu.php**, а централни део генерише скрипта **examsList.php**.



а) За случај употребе описан у наставку, дати одговарајући дијаграм секвенце, који описује акције које се предузимају у систему прављеном користећи неку MVC технологију. Претпоставити да су све провере успешно остварене. За интеркацију са базом података користити класу DBAccess. У систему постоје следеће компоненте: Exam, која представља испит, ExamRegistration која представља пријаву полагања испита, BillingAccount која представља виртуелни рачун студента. Сваки испит садржи цену полагања коју је могуће дохватити, док за виртуелни рачун може дохватити тренутно стање и поставити ново стање.



Напомене:

- На дијаграму јасно назначити улогу учесника (*Model, View, Controller*), путем стереотипа (<<model>>, <<view>>, <<controller>>);
- при позивима метода других учесника, јасно назначити који се аргументи прослеђују;
- могуће је читаву интеракцију приказати на једном дијаграму или на два дијаграма.

Пријава испита

Предуслови:

- Студент је пријављен на систем и приказана је страница са списком испита које може да пријави

Ток акција:

1. Притиском на дугме “Пријави” (1), студент покреће проверу могућности пријаве испита, након чега му се приказује прозор (*popup*) са информацијама о томе да ли је могуће да пријави одабрани испит, колико средстава би било искоришћено са рачуна, и колико средстава на рачуну преостаје.
2. Након притиска на дугме “Потврди” (2), испит је пријављен. Систем евидентира пријаву и умањује износ средстава на рачуну студента за потребан износ.
3. Приказује се прозор са поруком о успешно пријављеном испиту и преосталим средствима на рачуну.

б) Приказати горе описани систем путем WAE дијаграма. На дијаграму експлицитно назначити типове објеката, употребом стереотипа.

4. [30] Нека је дата веб апликација за замену дежурстава асистената на ЕТФ-у. Корисници апликације су администратор и асистенти. Администратор је корисник са посебним привилегијама. За сваког корисника памти се корисничко име (минимално 6 карактера), лозинка (минимално 12 карактера), име и презиме, телефон и адреса е-поште (у формату *ime.prezime@etf.bg.ac.rs*). На почетној страни корисник треба да има могућност уноса корисничког имена и лозинке и у случају исправно унетих података, треба омогућити рад са остатком система. Осим тога, на почетној страни постоје још два линка: регистрација новог корисника и промена лозинке. Нерегистровани корисник може да изврши регистрацију, тако што попуни форму са потребним подацима (сва поља су обавезна, и за поља која очекују одређени текстуални формат приликом уноса, постоји *JavaScript* функција која одмах укаже на грешку, ако у пољу није коректан формат). Регистрација мора да буде одобрена од стране администратора, да би корисник имао приступ систему. Регистровани корисник може да измени лозинку, тако што унесе у форми следеће податке: корисничко име, стару лозинку, нову лозинку и потврду нове лозинке.

Када се асистент улогује, може да види три листе: листу својих дежурстава, листу дежурстава која су обележена за замену и листу својих дуговања. На првој страни треба да има могућност да прегледа сва своја дежурства, која му се табеларно приказују, али уз могућност филтрирања података у тој табели, по називу предмета и по датуму. Поред сваког свог дежурства, асистент може обележити одређено дежурство и понудити га за замену. Асистент не може да понуди своја дежурства за замену, ако има 10 или више дуговања.

У листи дежурстава за замену, асистент може да види сва дежурства која су понудили други асистенти за замену и поред сваког дежурства постоји линк, тако да кликом на тај линк асистент преузима дежурство од другог асистента. Након што прихвати одређено туђе дежурство, то дежурство треба приказати у листи својих дежурстава, а листу дуговања треба инкрементирати/декрементирати (за оба асистента која учествују у размени треба да буду ажурирана дуговања). Дежурства која су већ започела у тренутку коришћења система, не треба приказивати корисницима ни у једној од ове две листе.

У листи дуговања, асистент види листу свих особа и сумарни број дежурстава, приказан у две колоне: број дежурстава које њему та особа дугује и број дежурстава које он дугује тој особи.

Дежурство описују назив предмета (испита), датум и време одржавања, испитни рок (у коме се држи) и сала у којој се одржава (један испит може да се одржава у више сала и може да има више дежурстава у бази). Приликом уношења новог дежурства, администратор уноси дежурне, а испитни рок и списак сала добија из падајуће листе (која се аутоматски генерише из базе из релевантних табела *ИспитниРок* и *Сала*). На једном дежурству може бити више дежурних асистената. Администратор има могућност ручног уношења новог корисника у систем, прихватања или одбијања регистрације новог корисника и уношења нових дежурстава за све асистенте.

а) [8] Нацртати IE модел базе података, са свим ентитетима и везама, које су потребне за реализацију ове апликације коју пројектујете, према описаној спецификацији.

б) [6] Методом *storyboard* приказати све функционалности које у систему има асистент. Написати главни ток догађаја и све алтернативне токове у ССУ документу, који дефинише промену лозинке на почетној страни система.

в) [8] Написати РНР функцију која служи за формирање листе дуговања. Дозвољено је користити све табеле које сте дефинисали у моделу базе података, под тачком а).

г) [8] Написати функцију која ће коришћењем AJAX технологије филтрирати листу својих дежурстава према називу.