

Испит траје 2.5 сата. Није дозвољена употреба литературе.

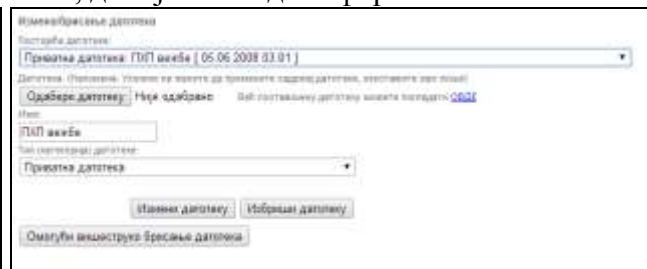
- [6] Фаза транзиције у RUP-у: кључне активности и прекретнице.
- [6] Навести сва задужења модератора у процесу формалне инспекције (пре, за време и након састанка).
- [8] Користећи SVN (*Subversion*) систем, навести редослед свих потребних команди, које је потребно извршити и објаснити ситуацију која доводи до конфликта, а затим написати команде и објаснити како се конфликт може разрешити. Претпоставити да постоје два корисника који користе исти SVN репозиторијум, да у репозиторијуму постоје бар два фајла и да на почетку корисници немају радне - локалне копије фајлова из репозиторијума. У сваком тренутку навести и која је верзија фајла (ревизија) са којом корисник ради.
- [40] Корисници факултетског система су администратор, наставници и студенти. Администратор има своје корисничко име, име, презиме, контакт телефон и адресу е-поште. Наставник има своје корисничко име, име, презиме, контакт телефон, адресу е-поште, звање (сарадник, асистент, доцент, ванредни проф., редовни проф.) и титулу (дипл.инж., мастер, магистар, доктор наука). Студент има корисничко име, име, презиме, контакт телефон, адресу е-поште, одсек, и годину студија. Када се корисник региструје, администратор треба да му додели корисничку ролу (наставничку или студентску), након чега се лозинка аутоматски генерише и шаље на адресу е-поште коју је корисник пријавио.

Наставник може да дода вест, измени постојећу или да је обрише. Свака вест има наслов, садржај вести, предмет на који се вест односи (при чему наставник мора да предаје тај предмет), датум до када је вест актуелна и опционо за вест може да се прикачи датотека.

Датотеке се уносе у посебном делу за рад са датотекама, преко стандардне форме за *upload*, као што је приказано на сл. 1. Када се одабере датотека, треба јој доделити име и категорију (предмет коме припада датотека или приватна датотека). Датотеке је могуће изменити и брисати. Када се одабере измена/брисање датотеке, добија се следећа форма на сл. 2.



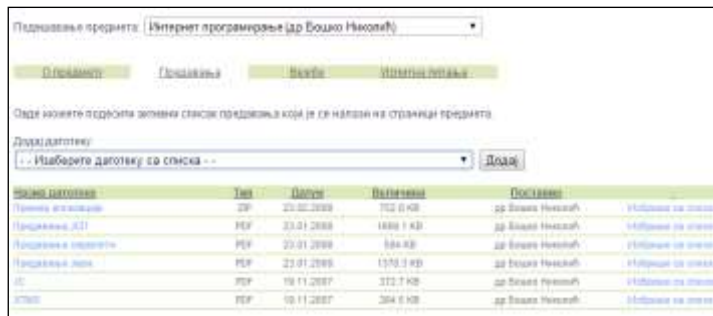
Слика 1



Слика 2

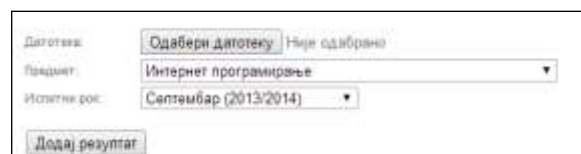
Сваки наставник држи један или више предмета, који су му додељени од стране администратора. Наставник може да управља садржајем тих предмета. Од основних информација сваки предмет има шифру предмета, статус (обавезни/изборни), ЕСПБ, циљ предмета, садржај по темама (недељама), списак литературе и структуру поена на предмету (активност у току предавања, практична настава, домаћи задатак, колоквијум, писмени испит, усмени испит, испит за рачунаром). За сваки свој предмет наставник може да ажурира дати садржај предмета. Осим тога наставник може да придружи неку датотеку једној од следећих секција: предавањима, вежбама или испитним задацима. Датотеке које су послате на сервер могу да се повежу са неком секцијом тог предмета, само уколико су приликом слања датотека на сервер, оне већ придружене том предмету.

На предмету може бити ангажовано више од једног наставника, па се приликом додавања датотеке у одређену секцију памти ко је ту датотеку поставио и ког датума. Наставник који је поставио одређену датотеку, може да је избрише из секције (али датотека остаје на серверу). Датотека која је повезана за неку вест не може да се дода у једну од ове три секције.



Слика 3

Наставник има могућност постављања резултата, за одређени предмет и одређени испитни рок, као што је приказано на следећој форми на слици 4.



Слика 4

а) [7] Нацртати изглед корисничког екрана методом *storyboard*, који служи за резервисање сала од стране наставника, тако што он изабере датум из *JavaScript* компоненте календар, салу коју жели одабере из падајуће листе постојећих сала на факултету, одабере почетак термина и крај термина (такође из падајуће листе) и уписује сврху резервације (као текстуални податак). Претпоставити да сала може да се резервише минимално на интервале од по 15 минута (00, 15, 30 или 45 мин.). Обавезно за сваку салу оставити могућност приказа тренутних резервација, да би наставник који резервише видео да ли је та сала у термину који жели слободна.

б) [8] Нацртати *IE* модел релационе базе података, са свим ентитетима и везама, које су потребне за реализацију ове апликације коју пројектујете, према описаној спецификацији (обухватити и функционалност описану у тачки а)).

в) [9] Коришћењем *PHP CodeIgniter* фрејмворка и *MVC* пројектног узорка имплементирати део кода који врши претрагу да ли је термин који је наставник одабрао приликом резервације у тачки а) слободан потпуно или делимично, и уколико је слободан извршити упис резервације у базу података, а уколико је делимично термин слободан, треба понудити тај нови термин наставнику. Није потребно реализовати корисничку форму за резервацију, довољно је написати само називе поља у форми који се прослеђују функцији.

г) [8] Приказати путем *WAE* дијаграма све релевантне делове апликације који остварују функционалности слања датотеке на сервер и придруживање датотеке новој вести, као што је описано у спецификацији задатка. Стереотипима назначити тип елемената дијаграма (<<server page>> и сл.). Не користити графичке симболе за представу елемената. Јасно назначити податке који се шаљу између појединих елемената на дијаграму.

д) [8] Написати сценарио случаја употребе регистрације наставника на систем и одобравање од стране администратора. Разматрати успешан сценарио и могуће алтернативне токове. Нацртати дијаграм секвенце који одговара случају коришћења „Резерација сала“ од стране наставника, описаном у тачки а), а након тога шаље наставнику шаље потврду резервације на адресу е-поште. Сматрати да у систему постоји класа која извршава слање поруке електронском поштом са задатом адресом примаоца, насловом (string) и телом поруке (string): `Mailer::send(recipientEmail, subject, body)`.