

СОФИЙСКИ УНИВЕРСИТЕТ "СВ. КЛИМЕНТ ОХРИДСКИ"

Факултет по Математика и Информатика

ДИПЛОМНА РАБОТА

на тема:

Преглед на понятието електронно обучение и примерна система за електронно обучение

на

Горка Живкова Иванова
Факултетен номер 42155

Специалност "Информатика",
Специализация "Информационни и комуникационни технологии",
Катедра "Информационни технологии"

дипломен ръководител доц.Красен Стефанов

гр. София
2005 г.

СЪДЪРЖАНИЕ

УВОД

- 1. ДЕФИНИЦИЯ НА ПОНЯТИЕТО "ЕЛЕКТРОННО ОБУЧЕНИЕ"**
- 2. ХАРАКТЕРНИ ОСОБЕНОСТИ НА ЕЛЕКТРОННОТО ОБУЧЕНИЕ**
- 3. ОСНОВНИ ЧЕРТИ НА WEB-БАЗИРАНОТО ОБУЧЕНИЕ КАТО ВИД ЕЛЕКТРОННО ОБУЧЕНИЕ**
- 4. ИЗИСКВАНИЯ КЪМ ПРОЦЕСА НА СЪЗДАВАНЕ НА СИСТЕМИ ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ**
- 5. ОРГАНИЗАЦИЯ И УПРАВЛЕНИЕ НА ЕЛЕКТРОННОТО ОБУЧЕНИЕ**
- 6. СТАНДАРТИ ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ**
- 7. ПРОГРАМИ ЗА СЪЗДАВАНЕ НА СИСТЕМИ ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ**
- 8. ОПИСАНИЕ НА ПРИМЕРНА СИСТЕМА ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ**
 - 8.1. ЦЕЛИ**
 - 8.2. РЕАЛИЗАЦИЯ**

ЗАКЛЮЧЕНИЕ

ЛИТЕРАТУРА

ПРИЛОЖЕНИЕ

УВОД

Днес младите хора са по-различни от всички други преди тях – те са в пряк достъп с дигиталните технологии във всеки един аспект на живота си и е съвсем ясно, че този факт оказва огромно влияние върху личностите им, поведението и начина им на мислене. Що се отнася до образователната система, новите технологии предоставят много повече възможности и в същото време – много повече предизвикателства към способностите на преподавателите.

Деца, на възраст между 1 и 12 години са първото поколение, което израства изцяло обградено с дигитална медиа. Компютрите са навсякъде – в дома, училището, офиса, производството, дори в ресторантите и магазините. А камерите за видео-наблюдение, CD-ROM-а, видео игрите, банкоматите, картите за достъп и мобилните устройства вече не учудват никого. За младите хора днес дигиталните устройства са нещо толкова обикновено колкото радиото и аналоговите телефони за техните родители.

Технологичната революция и комуникациите формират светогледа на младите, нещо, на което сме били свидетели и преди. Когато днешните родители бяха тийнейджъри, телевизията се превърна в най-мощния източник на информация познат на човечеството. Но днес, за потъналите във виртуалната реалност деца и младежи, телевизията изглежда тромава и старомодна. Нещо повече – тя може да бъде единствено едностранен източник на информация, докато дигиталните устройства предлагат нещо много по-изкусително – двустранна комуникация, преодоляла ограниченията на пространството. И ето тук се намесва феноменът Интернет – след появата му животът на обществото вече никога няма да е същия.

Дефиницията на думата "Интернет", която ще намерите в кой да е речник, гласи приблизително следното:

"Огромна колекция от вътрешно-свързани мрежи, които използват TCP/IP протокол и която произхожда от ARPANET в края на 60-те и началото на 70-те. Днес Интернет включва повече от 100 000 независими мрежи".

Световната компютърна мрежа Интернет е най-вълнуващата и революционна форма на комуникация в историята на човечеството. Въпреки, че в буквален превод Интернет означава "мрежа от мрежи", това е твърде ограничено възприемане на понятието, много по-точно е да мислим за Интернет като световно информационно пространство, в което непрекъснато се обменят данни между свързаните устройства – компютри, клетъчни телефони и т.н. Това, което прави Интернет толкова специален е факта,

че всеки компютър може да бъде едновременно източник и потребител на информация, сървър и клиент. Всяко свързано с Интернет устройство може да предоставя и черпи неограничени по количество и съдържание данни, което като всяка технология за масово ползване си има своите лоши и добри страни. Най-добра аналогия, макар и не съвсем прецизна, може да се направи между глобалната виртуална мрежа и една огромна, побрала цялото хилядолетно познание на човечеството и за съжаление несъществуваща библиотека. Интернет е нещо много повече от компютри, свързани в мрежа – той е начин на живот. Връзката с мрежата стана неотменна част от ежедневието, особено за младите хора. Затова никак не е изненадваща и появата на нов термин – понятието “електронно обучение”.

Още в зората на информационните технологии стана видно, че компютрите са идеално средство за подпомагане на обучението. Чрез средствата на мултимедията информацията може да бъде представена по най-подходящия и лесен за усвояване начин, като се използва звук, видео, анимация, графика и текст. Ако към средствата на мултимедията обаче се добавят невероятните възможности, предлагани от Интернет, ясно е, че е настъпила нова ера в развитието на системите за обучение.

Цел на дипломната работа

Да се направи преглед на понятието електронно обучение и да се създаде примерна система за електронно обучение.

Задачи произтичащи от целта:

1. Да се направи обзор на понятието “електронно обучение” и на възможностите, които то предоставя.
2. Да се направи преглед на основните характеристики, на които трябва да отговарят системите за електронно обучение.
3. Да се разработи примерно приложение, което да илюстрира характерните особености на системите за електронно обучение и да проследи в детайли създаването на такъв един специфичен вид софтуер.

За постигане на поставените задачи, е необходимо:

1. Да се даде дефиниция на понятието “електронно обучение”, на основните стандарти в областта и на техните производители.

2. Да се анализират основните характеристики на системите за електронно обучение и необходимите условия за създаването на такива системи.

3. Да се направи преглед на приложенията, които спомагат за създаването на системи за електронно обучение и на техните производители.

4. Да се проектира и реализира система за електронно обучение, която не само да илюстрира характерните особености на понятието, но и да проследи основните стъпки в процеса на разработка на системи за електронно обучение.

1. ДЕФИНИЦИЯ НА ПОНЯТИЕТО "ЕЛЕКТРОННО ОБУЧЕНИЕ"

Още от най-ранната история на човечеството образованието е един от основните негови приоритети. Стремежа да се прилагат в областта на обучението най-новите идеи и методи е разбираем – целта е по-бързо и по-качествено обучение. В тази връзка използването на последните най-модерни технологии на времето си е съвсем естествено. Най-съвременния такъв метод е така нареченото електронно обучение (e-learning) което включва употребата на компютър или някакъв друг вид електронно устройство и чрез него предоставя информация в подходящ за заучаване вид.

Има известни спорове относно това кой е точния термин за дейността, която на български наричаме "електронно" или "електронно" обучение. Някои специалисти настояват да се използва термина "e-learning", други – "online learning", трети – "online delivery" – но каквото и да изберем, целта е една – да означим революционната промяна в образователните системи настъпила с необратимото навлизането на дигиталните технологии в ежедневието.

Нищо не се променя толкова бързо, както термините, свързани с понятието "електронно обучение". Употребата на термина "електронно обучение" се сменя по-бързо дори и от съдържанието и начините за осъществяването му. Днес терминът често се използва като обща дума за:

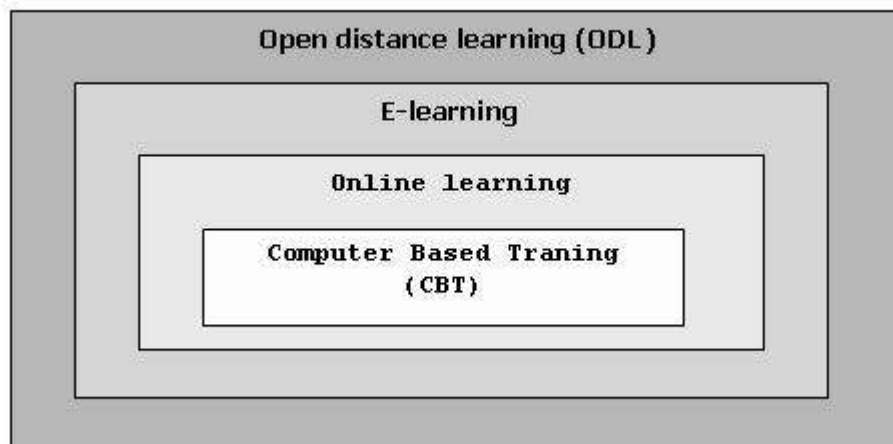
- open distance learning
- web based training (WBT)
- computer based training (CBT)
- technology based learning
- online learning

И все пак, ако трябва да формулираме точна дефиниция за термина електронно обучение тя би била следната:

“Предоставянето на формални и неформални дейности, процеси, общности и събития, с цел обучение в дадена област, посредством употребата на всички електронни медии като Internet, intranet, extranet, CD-ROM, видео, телевизия, DVD, мобилни устройства и т.н.”

Електронното обучение и online-обучението не са синоними. Online-обучението се използва единствено за Internet, intranet, LAN, WAN обучение и изключва употребата на CD-ROM.

Графично най-често използваните термини в областта на електронното обучение могат да се представят така:



Дистанционното обучение, включващо електронното, онлайн и компютърното обучение, става все по-широко разпространено. Причините за това са няколко:

- По ниската цена на компютърния хардуер, софтуер и телекомуникационните услуги.
- Младите хора, все по-добре запознати с компютрите и Интернет
- Подобрени умения за работа с клавиатура и по-малък страх от новите технологии.
- По-широкото разпространение на компютърните технологии във всички слоеве на обществото и всички сфери на дейност.
- По-удобен, лесен за ползване и приятен потребителски интерфейс.

2. ХАРАКТЕРНИ ОСОБЕНОСТИ НА ЕЛЕКТРОННОТО ОБУЧЕНИЕ

Живеем във века на космическите технологии, Internet и безжичните комуникации. Получаването на всякакъв вид информация никога не е било толкова лесно и широкодостъпно. Това, наред с големите предимства, които носи, има и някои недостатъци – голяма част от информацията, която получаваме, е неточна, непълна или изцяло погрешна. Отсяването на достоверната от недостоверна информация е един от големите проблеми пред потребителите на Internet и това в пълна сила важи и за областта на електронното обучение. Множеството термини, категории и характеристики, които се използват когато се говори за електронно обучение, могат да се систематизират в следните три основни категории:

- Синхронизация
- Комуникация
- Формат на медията

3.1. СИНХРОНИЗАЦИЯ

Електронното обучение бива два вида – синхронно и асинхронно. Най-лесно можем да ги разграничим отговряйки на въпросите кога и как се извършва обучението.

Синхронно обучение

При този тип обучение преподаването и заучаването се извършват по едно и също време, но двете страни се намират физически на различни места.

Примери за синхронно обучение:

- слушане на радио предаване на живо
- гледане на телевизионно предаване на живо
- аудио / видео конференция
- интернет телефония
- онлайн лекции
- онлайн семинари
- chat-rooms

Асинхронно обучение

При този тип обучение преподаването и заучаването не се извършват по едно и също време, а със известно отместване. Тук, както и при синхронното обучение, двете страни са разделени физически.

Примери за асинхронно обучение:

- уроци за самоподготовка в Internet или на CD-ROM
- записани аудио или видео презентации и семинари
- e-mail
- fax
- участие във форуми

т.е. основната характеристика на асинхронното обучение е тази, че преподавателя подготвя материала за курса преди провеждането на самия курс, а обучаващия се е свободен да реши кога иска да усвои новите знания.

3.2. КОМУНИКАЦИЯ

Комуникацията е процес на обмяна на информация между две страни, но това не означава непременно две физически лица. Съобразно броя на лицата, участващи в процеса, можем да разграничим четири основни вида комуникация:

- един към един
- един към много
- много към един
- много към много

Примери:

- private chat; изпращане на e-mail до преподавател; използване на програми за споделяне на екрана (screen shering)
- видео конференция; online семинар
- отговор на зададен въпрос във форум
- виртуални класни стаи, телефонни конференции

3.3. ФОРМАТ НА МЕДИАТА

Има множество видове медия, които могат да бъдат използвани за целите на електронното обучение. При избора на някой от тях, или на комбинация помежду им, задължително трябва да се има предвид, че избраната форма на представяне на информацията има огромен ефект върху крайния резултат, т.е. количеството усвоен материал от обучаващия се. Неправилно избраната форма на представяне на материала (поради пазарни или други причини) води до нищожен ефект от обучението. От

друга страна подходящата форма на представяне подобрява качеството на обучение. Например нека имаме текст, показван на екран, придружен от обяснения в аудио формат. Когато текста и звука са добре съгласувани, това би довело до огромен обучаващ ефект. И напротив, когато текста на екрана и звука не са така добре съгласувани или подбрани (например има разминаване във времето) ефекта от курса ще бъде минимален.

Избора на форма за представяне на информацията е от огромно значение и затова не бива да се пренебрегват и ограниченията на средата. Дори перфектно подбрана и поднесена видео лекция би нищожен, ако не и отрицателен ефект при лоша internet връзка. В подобна ситуация избора на текстов формат би бил много по-уместен.

Имаме пет основни вида форми за представяне на информацията:

- текст
- картина
- анимация
- аудио
- видео

ТЕКСТ

Тестът се състои от думи, изречения и параграфи, групирани в страници. Най-важните аспекти на текста са неговите размер, шрифт и цвят. Малките по размер или с блед цвят букви се четат трудно и водят до бърза умора и съответно до ниска продуктивност от обучението, а използването на твърде "екзотичен" шрифт, липсващ на клиентската машина може да доведе до нулев ефект от обучението.

Основните правила при подбор на размер, шрифт и цвят на текста са:

- лесен за възпроизвеждане
- широкоразпространен
- обучаващия се може да чете дълго време
- подходящ за печат

Този вид представяне на информация има два основни недостатъка – четенето на голямо количество информация на компютърния екран е тежко и уморително. Също така описанието на визуални концепции е трудно и неефективно.

КАРТИНА

Картината е двуизмерно представяне на данни. Дигиталната картина (тази, която виждаме на екрана на компютъра) се състои от известен брой пиксели, подредени в правоъгълен масив с определена широчина и височина. Всеки пиксел може да съдържа един или няколко бита информация, представяща яркостта и цвета на картината в тази точка.

Този начин за представяне на информация е много атрактивен и компактен.

От друга страна обаче за използването му скоростта на Internet трябва да е много по-висока, отколкото е нужда за представянето на тестова информация.

АНИМАЦИЯ

Анимацията е симулация на движение посредством серия от отделни картини или кадри. Този начин за представяне на информация е един от основните компоненти на мултимедийното представяне. Анимация може да се създаде посредством специално предназначени за целта програми като Flash или чрез създаване на картинни приложение чрез програми като Photoshop и Fireworks и конвертирането му в GIF.

Ползата от този вид представяне на информацията се състои в лесното представяне на сложни системи в движение, например начина на работа на мотора на колата. Освен това анимацията е достатъчно атрактивна, за да задържи вниманието на обучаващия се дълго време.

Този формат има два основни недостатъка. Първият е по-високите технологични изисквания, в сравнение с представянето на текстова информация. Вторият е, че създаването на анимация отнема доста повече време, в сравнение с останалите видове медия.

АУДИО

Аудио е звук в определен звуков диапазон, който човешкото ухо може да долови.

Аудио файловете обикновено са компресирани за да имат по-малък обем и по-бърза трансмисия. За да се получи желания мултимедиен ефект е важно звукът да бъде предаван като поток (streaming sound), а не първо да бъде съхранен на клиентската машина. Добре известни формати за звукови файлове са wave (.wav) , midi (.mid), MPEG, audio layer 3 (.mp3).

Предимството на този вид представяне е, че е много атрактивен за обучаващия се и ако е съчетан с подходяща анимация може да увеличи многократно резултата от обучението. За съжаление представянето на аудио информация има много по-високи технологични изисквания, отколкото представянето на текстова или графична информация (висока скорост на Internet, звукова карта, високоговорител и т.н.)

ВИДЕО

Видеото представлява всъщност последователност от статични картини (кадри) в много бърза последователност. Времевия интервала между всеки два кадъра е толкова малък, че мозъка интерпретира последователността от кадри като движение. Видео форматът е много близък с анимацията, но основната разлика е в много по-големия брой кадри в секунда. Положителните страни на този вид представяне са голямата изразителна сила, атрактивността, възможността за невербална комуникация. Но и тук, както и при аудио формата технологичните изисквания са много по-високи в сравнение с тези на текстовия и графичния формат. Освен това е относително трудно и скъпо за използване.

3. ОСНОВНИ ЧЕРТИ НА WEB-БАЗИРАНОТО ОБУЧЕНИЕ КАТО ВИД ЕЛЕКТРОННО ОБУЧЕНИЕ

Web-базираното обучение може да се опише като обучение, при което WWW (World Wide Web) се използва като виртуална среда за представяне на учебните материали и осъществяване на учебния процес.

Най-големия плюс на Web-базираното обучение, когато бива сравнено с традиционния стил на обучение (в класна стая) включва всички онези удобства, предоставяни от новите технологии, като например това, че всеки обучаващ се сам определя темпото си на учене, обучението е високо интерактивно и в същото време, въпреки по-високите първоначални инвестиции, като цяло по-евтино и продуктивно.

Когато Web-обучението бива сравнено с компютърното обучение, напр. посредством CD-ROM, отново се виждат предимствата на първото – достъпът до съдържанието на курса е лесен и не изисква разпространение на физически носител. Освен това Web-базираното обучение има допълнителни предимства, като често от тях са:

- Достъпът е възможен по всяко време, от всяко място, по целия свят – студентите винаги имат достъп до потенциално огромна библиотека от учебни материали и информация докато работят от къщи, офиса или хотелската стая. А с популяризирането на клетъчните модеми, студентите ще бъдат в състояние да участват в курс дори на място, където я няма традиционната телефонна линия или мрежова връзка.

- Усигоряването на учебни материали и учебна среда за всеки студент струва определена сума, която (съвсем логично) нараства с нарастването броя на студентите. От друга страна цената за web базирано обучение остава една и съща, без значение дали курса ще бъде посещаван от един, двама или хиляда студента

- Веднъж създаден, курсът може да бъде достъпван от неограничен брой обучаващи се, като единствените изисквания са компютър и достъп до Интернет или Интранет.

- Оценяването на напредъка на всеки студент е лесно и бързо. Тъй като студентите се обучават и в същото време са свързани в определена мрежа, оценяването на напредъка може да стане лесно и бързо чрез изпращане на данни на сървърската част на програмата за обучение. За разлика от CD-ROM обучението, което изисква от студентите да разпечатват своите работи и да записват резултатите си на диск, web-базираното обучение е много по-бързо, по-евтино и не по-малко ефективно. Данните, които се изпращат могат да варират от най-обща информация за това, кой и по кое време е достъпвал курса, до най-подробна информация включваща как определен студент е отговорил на индивидуални въпроси и колко време му е отнело усвояването на определен учебен материал.

- Обучението е строго индивидуално – всеки сам определя темпото си на обучение и обема учебен материал, който може да усвои. При добре проектираните системи за електронно обучение съдържанието на всяка "лекция" е разделена на логически обособени и не особено големи части, като този подход на представяне дава голяма свобода на потребителя – той може да следва праволинейно предварително дефинираната логичка последователност или, използвайки например search engine като google да намери нужната информация по даден въпрос.

- Съдържанието се обновява изключително лесно и може би това е най-голямото предимство на web-базираното сравнено със стандартния метод на обучение.
- Предоставя се възможност за комуникация между учащите и преподавателите от една страна и между самите учащи от друга. Това може да стане по различни начини – електронна поща, дискуссионни групи, телефонна и видео конференция, чат, форум.

Въпреки очевидните предимства на web-базираното обучение, като всяка система за обучение и то има не само положителни черти. Всъщност web-базираното обучение има само 3 сериозни недостатъка, но по всичко изглежда, че и трите ще бъдат преодоляни в следващите 5-10 години, когато мрежовата връзка стане също толкова широко разпространена и достъпна колкото е телефонната в момента.

Първият недостатък, сравнен със стандартния метод на обучение, е липсата на жив контакт между преподавателя и студента.

Разбира се, има много и разнообразни средства за обмяна на информация между участниците в курса – студентите могат да ползват своята web връзка за да изпратят e-mail на преподавателя, да влязат в т.нар. chat-roomя или да публикуват съобщение в определен форум. Възможно е да се организира и телефонна конференция, което е много полезно за обсъждане на даден задача в реално време. И въпреки наличието на всички тези чудесни възможности, липсата на жив контакт и контрол от страна на преподавателя не може да бъде компенсирана и продължава да указва своето негативно въздействие върху качеството на обучение. Но решаването на този проблем е в обозримото бъдеще – момента, когато видео-конференцията ще бъде обичайна част от повечето web-базирани системи за обучение не е никак далеч, тъй като трябва да се осигурят само две неща – интернет връзка с висока скорост и качествен софтуер за видео-комуникация.

Вторият сериозен недостатък на електронното обучение е липсата на мултимедийност в много web-базирани системи. Съществуват предметни области, в които употребата на аудио и видео са критични за създаването на реалистични симулации на определен проблем. Една графика може да бъде десетки пъти по-изразителна, лесна за разбирана и заучаване отколкото няколко екрана текст и цифри. По аналогичен начин един кратък видео материал може да представи информацията от множество графики по много по-достъпен и въздействащ начин. За съжаление в

повечето курсове мултимедийност липсва, тъй като дори студентите да разполагат с високоскоростна интернет връзка, използването на големи мултимедийни файлове намалява скоростта на цялата мрежа. Поради тази причина в голямата си част web-базираните системи за електронно обучение се състоят единствено от текст и графики. Но и този проблем изглежда ще бъде преодолян в близко бъдеще с напредъка на в областта на мрежовите стандарти и протоколи и най-вече с усъвършенстването на методите за видео- и аудио-компресия.

Третият недостатък на web-базираното обучение са високите му софтуерни и хардуерни изисквания сравнени с компютърното обучение като цяло. Така например web-базираното обучение има следните изисквания:

Software

- Web browser (специално приложение, даващо възможност на потребителя да разглежда и взаимодейства с HTML документи, намиращи се на определени web сървъри. Най-известните такива програми са Microsoft Internet Explorer, Mozilla Firefox, Opera, Netscape, Safari and Konqueror).
- Програми за възпроизвеждане на звук и за показване на картини и анимации
- Интернет връзка

Hardware

- Web server
- Application server
- Database server
- Компютри с мрежови устройства за връзка с Интернет
- Мултимедийни периферни устройства за възпроизвеждане на звук, картина, анимация (звукова и видео карти, микрофон, слушалки, тонколони)

4. ИЗИСКВАНИЯ КЪМ ПРОЦЕСА НА СЪЗДАВАНЕ НА СИСТЕМИ ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ

СПЕЦИАЛИСТИ В ПРЕДМЕТНАТА ОБЛАСТ

Задълбочените инженерни курсове трябва да бъдат създавани от инженерни специалисти със задълбочено познаване на предметната област. По същия начин бизнес експертите трябва да участват в създаването на курсове за бизнес професионалисти. Или с други думи материалите за всеки сериозен курс трябва да бъдат създавани от доказани специалисти в съответната област. Твърде често обаче курсовите материали са подготвени от студенти, стажанти или дори още по-лошо – от създателя на приложението за e-Learning. Всички тези хора могат да имат прекрасни преподавателски умения, но да не познават самия материал в нужната дълбочина. В крайна сметка резултатът е интересно, удобно за ползване приложение, където учащият се е мотивиран и ангажиран, но от което получава малко ценна информация тъй като съдържанието на курса просто не е достатъчно.

СПЕЦИАЛИСТИ В ОБУЧЕНИЕТО

Използването само на специалисти в предметната област води до някои проблеми, които са видими в предложенията на много съществуващи фирми за електронно обучение. Хората, които са експерти в тяхната област не винаги представят материала по най-ефективния начин. Често те не са наясно с най-добрите методи и принципите на обучение. А дори и тези експерти, които водят семинари и университетските професори, които се чувстват комфортно в класната стая, понякога срещат големи трудности в адаптирането на техния материал към един асинхронен курс, където студента се обучава в избрано от него време и определя голяма част от темпото на обучение. Следователно нуждата от специалист в обучението е голяма – някой, който разбира принципите в проектирането на курсове за електронно обучение.

За да стане възможно сътрудничеството на експертите в предметната област с тези в обучението е необходима специална методология. Въпреки че методологията трябва да бъде достатъчно детайлна, няма нужда проектантите на един курс по електронно обучение да губат часове в планиране на графиката, събиране на материали и т.н. Четири са основните области, на които трябва да бъде обърнато специално внимание в

проектирането на курс за електронно обучение, за да се постигне желаното високо ниво на качество:

- Структура на курса
- Потребителски интерфейс
- Интерактивност
- Контрол от страна на учащия се

СТРУКТУРА НА КУРСА

Необходимо е всички курсове да имат сходна структура. Това е нужно поради няколко причини. Първо, обучаващите се могат да се запишат в няколко курса и да открият една комфортна цялостност в представянето на материала. Добре дефинирана вътрешна структура на курсовете означава, че е необходимо студента да научи структурата само веднъж. Важно е учащият се да отделя повече време в заучаването на самия материал, а не в заучаване на механизма на представяне на този материал.

Предлагам описание на структурата и правилата за създаването и използването ѝ, която се наблюдава в повечето web-базирани курсове, разработени в Катедра "Информационни технологии".

Като пример може да се посочи курса "[Internet Technologies](#)", разработен в рамките на проект "[Обществени компютърни и комуникационни центрове](#)" и създаден от преподаватели и студенти от [Катедра "Информационни технологии"](#) при Факултета по Математика и Информатика на СУ "Св. Климент Охридски", както и сътрудници на [Центъра по технологии на информационното общество \(CIST\)](#) или курса "[Web-based Interface to Databases](#)").

УЧЕБНА ЕДИНИЦА

Учебна единица наричаме условно учебен обект, който може да съдържа един или няколко екрана, текст, под формата на страници (web страници), мултимедийни компоненти и други.

Основната структура на всеки курс е формирана от една или повече учебни единици.

- Всеки курс трябва да бъде разделен на отделни уроци от 3 до 15 учебни единици всеки. По този начин учащият се не е затруднен с огромно количество материал, не се чувства преуморен и обезкуражен, още повече че в днешно време често се налага уроците да бъдат заучавани в почивките между други задачи.

- Всеки курс съдържа от 5 до 15 урока и целия не е по-дълъг от 100 учебни единици. Курсовете в електронното обучение често са разделени на определен брой малки модули, които учащият се може да достъпва в произволна последователност. Дали да бъде възприет този подход или не зависи от материала и от изискванията на курса. Ако това е обучение за получаване на някакъв вид сертификат не би трябвало да дават възможност на учащия се да избира последователността и съдържанието на материала. Ако обучението има за цел опресняване или разширяване на знанията, то малките модули са за предпочитане.
- Всяка учебна единица на курса съдържа максимум 200 думи. Голямо количество текст на една страница може да бъде отегчително и това прави курса да изглежда повече като традиционен учебник. Късите учебни единици, които не се налага да бъдат превъртани са идеални, като предоставят възможност на студента да поеме материала наведнъж. На често целия материал не може да бъде събран на един екран. В такъв случай не е нужно заради превъртането на страницата материала изкуствено да бъде разделен на две по-малки части.
- Всяка учебна единица на курса съдържа минимум 20 думи. Малкото думи не носят почти никаква информация. Много онлайн курсове използват страници с заглавие и план с точки за ключовите понятия в курса, но това е погрешна практика.
- Всяка учебна единица показва текущ номер на учебната единица и брой учебни единици в курса. Виждайки текущия номер на учебната единица и учебни единици, студентите работят целенасочено. Много от учащите се са професионалисти с ограничено време и те биха искали да знаят колко още материал от курса трябва да усвоят. Без тази информация студентите биха се чувствали сякаш прекосяват дълъг и тъмен тунел, чийто край не се вижда.
- Първата учебна единица въвежда учащия се в основните принципи на потребителския интерфейс. Това дава възможност на студента да добие опит в навигацията преди още самия курс да е започнал. В някои курсове навигацията е толкова сложна и неинтуитивна, че риска да се пропусне материал, докато се правят експерименти с движението в рамките на курса е огромен.

- Всеки курс има уводна учебна единица, която съдържа кратко резюме на курса, целите му, списък със заглавията на уроците, включени в него, снимка и кратка биография на преподавателя. По този начин студентите знаят какво да очакват от този курс. Освен това учащите се информират за квалификацията на специалиста в предметната област, подготвил материала за курса.
- Всеки урок започва с описание на темите, които ще бъдат покрити в него. Този добре известен подход в представянето на материала дава на студента виждане за урока преди още да е започнал, а също и успокоителното чувство, че на въпросите възникнали в предходната част от курса, ще бъде отговорено в следващата.
- В края на всеки дълъг или сложен урок трябва да има обобщаваща учебна единица, която припомня ключовите точки в урока. Обобщението в края представя информацията в изчистен вид и затвърдява полученото знание. За кратки уроци обобщението не е необходимо, дори би могло да бъде отегчително.
- Всеки урок трябва да завършва с кратък тест. Тестът е чудесен начин да се задържи вниманието на студента и да му даде възможност да упражни току-що полученото знание. Отговарянето на въпроси затвърждава знанието, особено ако преминаването на следващите уроци изисква някакво постигнато ниво в изучаването на материала.
- Всяка учебна единица съдържа връзка към помощна учебна единица с описание на навигационните бутони. Трябва да бъде сигурно, че информацията за основните средства за навигация е достъпна от всяка една учебна единица на курса.
- Всеки курс съдържа речник на специализираните термини. По време на курса техническите термини трябва да бъдат свързани с речника. Тази полезна опция дава възможност на учащия се мигновено да опреснява паметта си за термини или факти без да прекъсва естествената последователност на курса.

ПОТРЕБИТЕЛСКИ ИНТЕРФЕЙС

Алберт Айнщайн казва "Всичко трябва да бъде направено толкова просто, колкото е възможно, но не и по-просто" (**Quotation #2927 from [Laura Moncur's Motivational Quotations](#)**).

Когато мислим за проектирането на потребителския интерфейс, този принцип трябва да бъде основно правило. Всички необходими функции трябва да бъдат включени в семпли, лесни за разбиране бутони, и нищо повече. За съжаление съществуват онлайн курсове с чудесен, гъвкав, зашеметяващ със своята красота потребителски интерфейс, но с толкова сложен дизайн, че е невъзможно да се открие бутона за следваща страница.

НАВИГАЦИЯ

Навигационните бутони би трябвало да използват проста графика, която да показва тяхната функция и само необходимите функции трябва да бъдат включени. Функцията на всеки бутон трябва да бъде очевидна. Всички бутони трябва да изглеждат по сходен начин и да имат една и съща функция във всеки курс. Студентите не би трябвало да си губят времето опитвайки се да разберат как да се предвижват в курса. Това отклонява времето и енергията на учащия се от заучаването на материала в съвсем нежелана посока.

ИНТЕРАКТИВНОСТ

Друг ключов аспект на електронното обучение е нуждата от взаимодействие между учащия се от една страна и приложението от друга. Съществуват много онлайн курсове, които се движат "сами" от урок на урок, изливайки огромно количество информация и не искайки от потребителя нищо друго освен да гледа втрещено екрана. Ситуацията е подобна на тази в училище – слушането на монотонната реч на учителя в класната стая често води до сладка следобедна дрямка. Но при онлайн курсовете е дори още по-лошо. Проучванията сочат, че много от обучаващите се губят концентрация по време на такъв вид "обучение". За да научи необходимия материал задължително условие е студента да бъде ангажиран и активен в процеса на обучение. Дори простото действие от натискането на бутона за преминаване на следващата страница заангажира вниманието на учащия се. Допълнителни материали като аудио и видео сегменти, анимации, аплети и рор-уп страници трябва да бъдат активирани единствено при определено действие от страна на студента. Например за да се пусне видео материал е необходимо студента да натисне бутон, за да се пусне анимация студента трябва да постави курсора на мишката над графика и т.н.

Кратките изпити и тестове са добър начин за припомняне и затвърждаване на наученото и в същото време ангажират вниманието и дават възможност за взаимодействие между студента и курса.

Много фирми, занимаващи се с разработването на онлайн курсове често създават тестове базирани на обща схема – въпрос, последван от множество HTML радио бутони с един верен отговор. Този вид тестове освен че нямат много естетичен вид, след известно време стават отегчителни за попълване. Добра практика е да се използват графики в тестовете. По този начин тестовете стават не само по-интересни и красиви на външен вид, но също така изискват и повече мислене. Тестовете могат да включват избирането на няколко верни отговора, избирането на диаграма или схема за отговор на въпрос, попълването на таблица или въвеждането на дума или цифра. Тестовете изискващи повече мислене решават проблема с произволното избиране на отговор от студента, само защото знае, че верния отговор може да е точно един.

КОНТРОЛ ОТ СТРАНА НА УЧАЩИЯ СЕ

Контрол от страна на учащия се означава, че студента може да получи такова количество и вид информация от курса, каквото самият той желае. Контрола превръща стриктния линеен курс в сбор от разклонения за по нататъшно изследване. Всеки студент преминава един и същ базов курс и научава един и същ основен материал, но имайки контрол може да игнорира или не допълнителните материали към курса, които трябва да са достъпни под различни форми. Например някои студенти биха предпочели да не гледат видео материали с говорещ преподавател, а да прочетат съдържанието курса, преминавайки през целия материал колкото е възможно по-бързо. Други пък избрали да гледат все пак видео материали с преподавателя за да получат някаква допълнителна обяснения. Един студент може лесно да разбере идеята за въртенето на електрона и магнитното поле, докато друг се нуждае от анимация, която да я илюстрира; един може да е съвсем наясно с икономиката на "свободния пазар" докато друг може да се нуждае от допълнителни материали, един вид мини-курс в основния курс, за да научи повече. Контрол от страна на учащия се означава, че студентите могат да избират дали да активират или не допълнителните опции, докато преминават един и същ курс.

Контрола от страна на учащия се има две основни цели:

- Да предостави избран допълнителен материал, който да илюстрира ключови моменти от курса, който студентите, интересувани се от определена тема биха могли да видят.
- Да разнообрази дългите, пълни само с текст страници, поддържайки интереса на студента и предоставяйки интерактивен начин за получаване на допълнителен опит.

Примери за контрол от страна на учащия се:

- Анимации – графика или фигура, която се движи, илюстрирайки ключов момент от теорията, в отговор на заявка от страна на студента. Например точка, която се върти, за да илюстрира принципите на магнитния усуквач момент.
- Аплети – малка програма, вградена в HTML кода, която студентите могат да стартират, за да получат допълнителна информация за дадена тема. Например калкулатор, който учащия се може да използва, за да изчисли каква такса би могъл да взема като независим консултант.
- Pop-up страници – страници, съдържащи допълнителна информация, видими само при изрична заявка от страна на учащия се, които съдържат полезна за някои, но не и за по-голямата част от аудиторията, информация.
- Препратки – хипервръзки (links) към допълнителни страници или WEB сайтове, които могат да бъдат добър източник на по-подробна, но не и задължителна за покриването на курса, информация.
- Аудио/Видео материали – кратки клипове, илюстриращи идеята или целта на определена част от курса или на курса като цяло.

ОБОБЩЕНИЕ НА ПРИНЦИПИТЕ

Всеки висококачествен, полезен Web-базиран курс за електронното обучение изисква спазването на основни правила и ръководни принципи, уникални за тази нова медиа. Посочените по-горе критерии, базирани на опита и изследванията на множество високи професионалисти в областта на дизайна и електронното обучение могат да бъдат обобщени по следния начин:

- Съдържанието на курсовете задължително трябва да е базирано на материали, предоставени от специалисти в съответната предметна област.
- Курсовете трябва да бъдат подготвени от преподаватели с опит в този вид обучение.
- Структурата на курса е добре обмислена, единна и последователна за всеки курс.
- Потребителският интерфейс е удобен за ползване, интуитивен и еднакъв за всеки курс в приложението.
- Всеки курс изисква определени действия от страна на студента, за да продължи процеса на обучение, т.е. всеки курс е интерактивен.
- Курсовете предоставят свобода на учащия се да определя колко и какви допълнителни материали към курса да ползва.

С всички тези важни критерии би трябвало да се съобразяват не само фирмите, разработващи продукти за електронно обучение, но и тези, които избират определен продукт за обучение на своите служители или клиенти.

5. ОРГАНИЗАЦИЯ И УПРАВЛЕНИЕ НА ЕЛЕКТРОННОТО ОБУЧЕНИЕ

Има един въпрос, който хората, отговорни за обучението на служителите на фирмата или нейните клиенти, задават на консултантите и производителите на системи за електронно обучение повече от всеки друг. Консултантите чуват този въпрос навсякъде – по телефона, след почти всяка презентация, на фирмени събрания и изобщо навсякъде, където съответната система за електронно обучение е демонстрирана по един или друг начин. Този привидно прост въпрос е “Колко струва това?”

Консултантите изпитват смущение, когато ги питат колко струва един проект за обучение. Това е нещо като да се обадиш на строителна компания питайки “Искам да построя къща, колко ще струва?” и очевидният отговор винаги е “Зависи”. Ясно е, че цената на една къща зависи от множество фактори – като се започне от подготовката на парцела за строеж, мине се през размера на жилищната площ, броя стаи, стила, качеството и срока на строителството и се завърши с детайли като подовата настилка, облицовката на стените и обзавеждането. По абсолютно същия начин цената на един проект за обучение зависи от много различни фактори.

Когато се изработва бюджет за система за електронно обучение, изработена по поръчка, трябва да се отговори на следните важни въпроси:

- Как електронното обучение ще бъде доставено на клиента – CD-ROM или Web?
- Ако е Web-базирано, каква ще бъде средната скорост на интернет връзката – висока или ниска?
- Каква е предметната област
- Колко време би отнело на един обучаващ се да завърши курса или колко екрана ще съдържа той? (по този начин се получава една приблизителна представа за размера на проекта)
- Трябва ли програмата да използва аудио, видео, анимации, графики, сложни инструкции?
- Кой ще предостави съдържанието на курса? Има ли съществуващи модули за обучение? Има ли специалист в предметната област, който може да се включи в проекта?
- Ще трябва ли програмата да проследява прогреса на студента?
- Кога трябва да започне проекта? Кога трябва да завърши?

С тази информация един опитен производител на софтуер би трябвало да бъде в състояние да направи една груба оценка на цената на изработване на нужния продукт. С повече детайли дори може да се стигне до уточняване на крайната цена.

6. СТАНДАРТИ ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ

Преди един ръководен принцип да се превърне в "стандарт", той бива наричан "спецификация". Спецификациите са разглеждани от една официално призната организация например IEEE. По-долу са описани накратко ключовите организации в областта на електронното обучение.

Advanced Distributed Learning (ADL)
www.adlnet.org

ADL е американска правителствена организация, която разследва и разработва спецификации за утвърждаване и поощряване на електронното обучение. Целта на ADL е да осигури достъп до висококачествено обучение и учебни материали, които могат да бъдат приспособени към индивидуалните нужди и направени

леснодостъпни за широката общественост. Комбинацията от изследвания и препоръки на ADL помага на превръщането на спецификациите в стандарти.

Най-широко приетата ADL публикация е т.н. the ADL Shareable Content Object Reference Model (SCORM). SCORM спецификацията комбинира елементи от IEEE, AICC и IMS спецификации в единен утвърден от тези организации документ, който може да бъде лесно имплементиран. Съществен принос на ADL към съществуващите стандарти представляват множеството примери, най-добрите практики и обяснения, които помагат на разработчиците да имплементират eLearning спецификациите по един последователен и съвместим между отделните продукти начин.

Aviation Industry CBT Committee (AICC)
www.aicc.org

Създадена през 1988, AICC е международна група състояща се от професионалисти в областта на технологично-базираното обучение. Те създават CBT ръководствата за въздухоплавателната индустрия. Тяхната цел е по-стойностно и ефективно обучение. Въпреки че те публикуват множество препоръки – включително софтуерни и хардуерни конфигурации – техните инструкции за управление на компютърни системи (CMI) оказват най-голямо въздействие.

Institute for Electrical and Electronic Engineers Learning
Technology
Standards Committee (IEEE LTSC)
www.ltsc.ieee.org

IEEE е международна организация, която разработва технически стандарти и препоръки за електрически, електронни, компютърни и комуникационни системи.

IEEE спецификациите са вече широко разпространени и са се превърнали в международни стандарти.

Вътре в самата организация има комисия наречена Learning Technology Standards Committee (LTSC), която предоставя спецификация съдържаща най-добрите практики в разработването на Learning Management Systems. Най-широкопризнатата такава спецификация е Learning Object Metadata (LOM) спецификацията, която дефинира групи и елементи, които описват необходимите ресурси. И двете организации IMS и ADL използват LOM елементи и структури в техните спецификации.

IMS Global Consortium (IMS)
www.imsproject.org

IMS е консорциум от доставчици и разработчици, които се фокусират върху разработката на XML-базирани спецификации. Тези спецификации описват ключовите характеристики на курсовете, уроците, оценяването, обучаващите се и групите. В добавка, XML спецификацията и Best Practices Guidelines предоставят структура за представяне на eLearning meta-data (определят се като данни за данните). IMS предлага последователен и внимателно структуриран подход за описание на множеството ресурси и предлага общо множество от елементи, които могат да бъдат обменяни между множество системи и продукти. Описването на eLearning ресурсите помага в търсенето за съществуващи ресурси както, дава възможност за обмяна на ресурси и данни с други разработчици и улеснява управлението и поддръжката на тези ресурси по време на техния живот. Едни от най-популярните спецификации на IMS консорциума са: IMS Meta-data, IMS Content Packaging and IMS QTI (Question and Test Interchange).

Три са основните теми, предмет на обсъждане в и между организациите, отговорни за създаване на спецификации и стандарти:

- Интерфейс за комуникация (API) – как ресурсите комуникират с други системи
- Мета-данни – как да се опишат eLearning ресурсите по логичен и стандартизиран начин
- Пакетиране – как да се групират eLearning ресурсите така че да бъдат максимално удобни за използване от различни eLearning Management Systems

7. ПРОГРАМИ ЗА СЪЗДАВАНЕ НА СИСТЕМИ ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ

BLOGGING TOOLS (БЛОГОВЕ)

Блог-а представлява личен "дневник", често осъвременяван и предназначен за обща публична употреба. Блог съобщенията обикновено изразяват личното мнение на автора или служат за целите на Web site-а, на който се публикува блог-а. За създаване, попълване и редактиране съдържанието на един такъв блог има специални програми наречени blogging tools (виж приложението).

COLLABORATION TOOLS – APPLICATION SHARING

Приложения, даващи възможност на водещия презентацията да споделя програми, прозорци или цял екран с останалите участници (виж приложението).

COLLABORATION TOOLS – AUDIO CONFERENCING

Програми, предоставящи възможност на участниците в аудио конференцията да говорят един с друг (виж приложението).

COLLABORATION TOOLS – CHAT AND INSTANT MESSAGING

Програми, предоставящи възможност на участниците водят текст-базирани разговори в реално време (виж приложението).

COLLABORATION TOOLS – E-MAIL

E-mail клиентите дават възможност за изпращане и получаване на съобщения посредством Internet (виж приложението).

COLLABORATION TOOLS – NEWS READER

Клиентска програма, която дава възможност достъп, четене и участие в newsgroups (виж приложението).

COLLABORATION TOOLS – ONLINE DISCUSSION

Програми, предоставящи възможност на участниците в дискусията да четат и изпращат съобщения на определено място, където другите участници могат да ги прочетат и да им отговорят (виж приложението).

COLLABORATION TOOLS – VIDEO CONFERENCING

Програми, даващи възможност на участниците във видео конференцията да се виждат и чуват един друг (виж приложението).

COLLABORATION TOOLS – VOTING

Програми, с помощта на които участниците могат да изразяват мнението си чрез гласуване (виж приложението).

COLLABORATION TOOLS – WEB TOUR

Програми, даващи възможност на участниците да разглеждат web страници заедно и едновременно (виж приложението).

COLLABORATION TOOLS – WHITEBOARD

Програми, симулиращи комуникацията между преподавател, пишещ на дъска и после даващ възможност на студент да допълни написаното (виж приложението).

COLLABORATION TOOLS – FILE FORMAT

Програми с чиято помощ е възможно преобразуването на файлове от един формат в друг (виж приложението).

CONTENT CONVERTERS – GENERAL

Програми, преобразуващи файлове от разнообразни формати в удобни за публикуване в web файлове (виж приложението).

CONTENT CONVERTERS – POWERPOINT

Преобразуват Microsoft PowerPoint файлове в web файлове (виж приложението).

CONTENT CONVERTERS – WORD

Програми, преобразуващи Microsoft Word в HTML файлове (виж приложението).

CONTENT MANAGEMENT SYSTEMS

Програми, опростяващи задачата за управление и повторна употреба на всички видове информационно съдържание (виж приложението).

COURSE AUTHORIZING TOOLS

Специализирани приложения за създаване на e-learning курсове (виж приложението).

HELP AUTHORIZING TOOLS

Специализирани приложения за създаване на online Help файлове (виж приложението).

LEARNING CONTENT MANAGEMENT SYSTEMS

Програми, които улесняват и опростяват задачата по създаване, управление и повторна употреба на учебното съдържание, т.е. аудио и видео медиа, страници с графики, тестове, уроци и всички други компоненти, които биха могли да бъдат част от едно eLearning приложение (виж приложението).

LEARNING MANAGEMENT SYSTEMS

Програми, опростояващи процеса по администриране на учебния процес и обучение. Една от основните им функции е да предоставят колекция от курсове и да проследяват кои курсове обучаващите са посетили/усвоили (виж приложението).

MEDIA EDITORS - ANIMATION

Програми за създаване на анимация (виж приложението).

MEDIA EDITORS - AUDIO

Аудио програми предназначени за записване, създаване и редактиране на звук (виж приложението).

MEDIA EDITORS - GRAPHICS

Програми за създаване на графики, диаграми, схеми, снимки, картини, икони (виж приложението).

MEDIA EDITORS - MULTIMEDIA

Програми, обединяващи възможностите на аудио и графичните програми, предназначени за запис, създаване и показване на мултимедийна информация (виж приложението).

VIRTUAL-SCHOOL SYSTEMS

Приложенията, наречени "Virtual-school systems" дават възможност за създаване на курсове, в които обучението се доближава максимално до класическото обучение в училище, водено от учител (виж приложението).

Докато специализираните приложения за създаване на мултимедия (картина, звук, анимация и т.н.) са малко или много познати на

масовия потребител, доста по-интересно би било да се разгледат по-подробно системите за виртуално обучение и възможностите, които те предлагат.

Blackboard Learning System например е проектирана за институции, чиято дейност е пряко свързана с обучението. Blackboard Learning System е технология, предлагаща ефективни решения за университети, колежи, гимназии и виртуални курсове. Голямото ѝ предимство е в комбинирането на множество разнообразни възможности за генериране и редактиране съдържанието на един учебен курс и в същото време за управление на студентите и тяхното усвояване на учебния материал.

Три са темите, върху които е акцентирано при генерирането и управлението на учебния курс и това са:

- Факти и обяснения
- Комуникация
- Оценяване

Blackboard Learning System разполага голям набор от средства за управление на курсовете и пригаждането на обясненията към индивидуалните нужди на обучаващия се. Тези средства включват Course Management tools (приложения, фокусирани върху бързо и ефективно създаване на курсове, семестър по семестър миграция и архивиране); Content Authoring tools (приложения за редактиране на текст подобни на Word); Learning Units (функционалност, даваща възможност на преподавателя да създава последователност своя собствена последователност от уроци, специфична за нуждите на текущия курс и да контролира навигацията на студентите през тези уроци); Teaching and Learning Tools (дават възможност на преподавателя да създава речници); Personal Information Management (дават възможност на студентите и преподавателите за по-добро управление на тяхната работа чрез използване на календар и Blackboard messages – специален e-mail).

8. ОПИСАНИЕ НА ПРИМЕРНА СИСТЕМА ЗА ЕЛЕКТРОННО ОБУЧЕНИЕ

8.1. ЦЕЛИ

Изработването на качествена система за електронно обучение е нелека програмна задача тъй като в процеса на разработка трябва да се посрещнат специфични за този вид софтуерен продукт изисквания и да се включат експерти от различни области – софтуерни специалисти, специалисти от конкретната предметна област и специалисти в областта на

обучението. Приложенията от типа "Learning Management System" са дори още по-сложни – те предоставят възможност за добавяне на съдържание, за наблюдаване качеството на обучението на студентите, както и степента на справянето им с поставените задачи. Една подобна система може да предоставя възможности за интерактивно обучение като видео конференции, дискуссионни форуми и др.

Целта, преследвана с разработеното в тази дипломна работа приложение е да се проследят основните стъпки при разработването на системите за електронно обучение и да се илюстрират техните основни характеристики. Системата "E-School" е вид "Learning Management System" т.е. интерактивна среда за предоставяне на услугата електронно обучение и в същото време за динамична промяна съдържанието на системата. Целта е да се създаде система за електронно обучение, която да дава възможност за създаване на курсове и чрез която съдържанието на курсовете може да бъде разширявано (чрез добавяне на нови курсове), редактирано (чрез добавяне / изтриване / редактиране на съществуващия учебен материал) или просто да се разглежда като обикновена система за електронно обучение. Съобразно изискванията, поставени пред дипломната работа, системата трябва да поддържа два вида потребители – преподаватели и студенти, като потребителския интерфейс и предоставените възможности за тях са различни.

8.2. РЕАЛИЗАЦИЯ

Основните фази, през които минава разработването на всеки софтуерен продукт, са следните:

1. Специфициране
2. Дизайн
3. Имплементиране
4. Тестване
5. Документиране

През всяка от тези фази преминава и приложената към тази дипломна работа информационна система "E-School".

СПЕЦИФИЦИРАНЕ

Фазата специфициране се занимава с анализ на потребителските изисквания и на възможностите за удовлетворяването им. Ключовата дума тук е "какво", т.е. какви данни се подават на приложението и какъв изходен резултат се очаква от тяхното обработване.

Спецификацията на приложената информационна система е съобразена с целите на дипломната работа, а именно – се покрие основната функционалност на система за електронно обучение.

Изискванията към системата могат да се групират в следните категории:

User Management
Roles Management
Content Management
Security

Основни изисквания към системата са:

- възможност за регистрация на нов потребител
- режим на работа на два вида потребители – преподаватели и студенти
- управление и промяна на личните данни на потребител
- въвеждане на нов курс или изтриване на курс – за потребителите от група преподаватели
- управление и промяна съдържанието на курс – за потребителите от група преподаватели
- възможност за въвеждане на тестове към курс – за потребителите от група преподаватели
- възможност за разглеждане на информация за преподавателя, водещ даден курс – за потребителите от група студенти
- възможност за записване в курс и отписване от курс – за потребителите от група студенти
- възможност за разглеждане съдържанието на курс и съдържанието на тест – за потребителите от група студенти

ДИЗАЙН

Фазата дизайн или проектиране се занимава с детайлно анализиране на потребителските изисквания, изброени в спецификацията на проекта и средствата за постигане на нужния резултат. Тук ключовата дума е "как", т.е. какви средства (софтуер, хардуер) са нужни за реализиране на проекта, каква

последователност от действия трябва да се следва, за да се постигне търсената цел, какви са основните компоненти и как те взаимодействат помежду си. Дизайнът може да бъде както по-обобщен, акцентиращ само върху основните компоненти, така и съвсем детайлен включващ различни видове диаграми (State Diagrams, Sequence Diagrams, Entity Relationship Diagrams (ERD) и т.н.) и описание на use cases.

Проектирането е тясно свързано с потребителските изисквания и затова, в конкретния случай, за целите на примерната система "E-School", беше решено да се използва:

Software

- Web Browser - Microsoft Internet Explorer, версия 5.0 и по-висока
- Web Server - JBoss, мощен, сигурен, бърз и удобен за употреба open source J2EE базиран сървър, изцяло имплементиран на Java, версия 3.2.1
- Application Server - Отново JBoss, версия 3.2.1
- Development Tool - Eclipse, open source Java базирано IDE (Integrated Development Environment), с възможност за лесно интегриране с JBoss сървър, версия 3.0 и по-висока
- Database tool - MySQL Server

Hardware

- Няма специални изисквания

ОСНОВНИ КОМПОНЕНТИ И ВРЪЗКА МЕЖДУ ТЯХ

Системата за електронно обучение "E-School" е реализирана съобразно т.н. Model-View-Controller pattern (шаблон), затова преди да навлезем в детайли за дизайна на самата система е необходимо да се запознаем накратко с този вид архитектура.

Model-View-Controller (MVC) Architecture

Софтуерът, както и всяко друго нещо, трябва да взаимодейства с нещо друго, за да бъде полезен. Понякога той взаимодейства с друг софтуер или хардуер, а много често - с хора. В последния случай се появява нуждата от интерфейси, за

програмирането и поддържането на които понякога отиват много повече усилия отколкото за цялата останала част от програмата. Особеното при интерфейсите обаче е, че понякога е необходимо да има два, три или повече различни интерфейса за една и съща програма. Освен това след известен период от време почти всеки един интерфейс има нужда от осъвременяване. В същото време останалата част от приложението може да остане непроменена. Разработчиците на софтуер постепенно стигнали до идеята, че тясната интеграция между интерфейса и останалата част от приложението е в ущърб и на двете - от една страна прави приложението по-малко гъвкаво и трудно за миграция, от друга прави интерфейса по-труден за използване от други приложения. И така, през 70-те Smalltalk дефинират архитектура, чиято цел е да се справи с този проблем наречена Model-View-Controller архитектура. Оттогава MVC дизайна става нещо почти задължително, особено в обект-ориентираните системи.

Целта на MVC design pattern е да раздели приложението (model), начина, по който то се представя на потребителя (view) и начина, по който потребителя го контролира (controller).

Model-View-Controller

Model (модел) наричаме непроменящата се същност на едно приложение или в обектно-ориентирана терминология - множеството от класове, които моделират и поддържат разрешавания проблем и следователно чиято продължителност на "живот" е толкова, колкото и на самия проблем. Каква информация би трябвало да има модела (класовете) за обкръжаващия го свят? Никаква.

Моделът има информация за всички данни, които трябва да бъдат показани и за операциите, които могат да бъдат прилагани върху тях. В същото време не знае нищо за вида на потребителския интерфейс, начина, по който данните ще бъдат показвани, възможностите за манипулация на данните, които потребителския интерфейс предлага и т.н.

Прост пример за модел може да бъде обект, представляващ часовник.

Съобразно вътрешната си логика този обект измерва времето, обновявайки някаква своя вътрешна променлива за време всяка секунда. Обекта може (но не е задължително) да предоставя методи за проверка на текущото време (които ще бъдат ползвани

от view обекта) и методи за промяна на текущото време (които ще бъдат ползвани от controller обекта).

Model-View-Controller

За дадена ситуация и версия е възможно да има един или повече интерфейса за даден модел, които наричаме views.

В обектно-ориентирана терминология - това е множеството от класове, които ни дават "поглед" върху модела, т.е.

- GUI (graphical user interface) view
- CLI (command line interface) view
- API (application program interface) view

Въпреки че view-та са много често графични, това не е задължително.

Какво знаят те за модела? Само това, че го има и съвсем малко за неговата природа.

View обекта използва методите за достъп до данните, предоставени от model обекта и след това показва тази информация.

Показването на данните може да бъде под всякаква форма, или ако използваме предишния в горния пример с часовника, един view обект може да покаже текущото време като електронен, друг view обект - като механичен, а трети view обект - като пясъчен часовник. Но както и да бъде показано времето, това няма каквото и да е отношение към вътрешното поведение на часовниковия механизъм, т.е. model обекта.

Model-View-Controller

Controller наричаме обект, който ни позволява да манипулираме интерфейсите, т.е. view-та.

В сравнение с view-та, controller-ът има много повече знание за платформата и операционната система, докато

View-та са сравнително независими от това дали събитието, което предизвиква промяна на показваните данни, идва от Microsoft Windows, X Windows или някъде другаде.

И точно както view-та знаят за съществуването на модела, но модела за view-та - не,

така и контролера знае кои са неговите view-та, но view-та не знаят кой е техния controller.

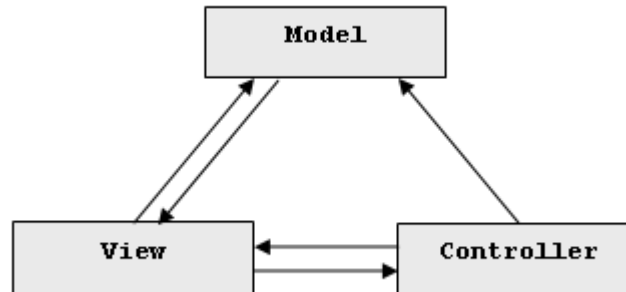
Controller обектът получава известие, когато потребителя желае да манипулира данните, пазени в модела и съответно прави

възможна тази манипулация чрез извикването на предоставените от модела методи.

В примера с часовника потребителя може да иска да промени текущото време чрез въвеждане на новото време в дадено поле на потребителския интерфейс (view обекта). При натискане на бутон например controller обекта, асоцииран с този view обект получава известие за желаната промяна и вика предоставения от model обекта метод "setTime()".

В някои ситуации controller обектът може да комуникира с view обекта без да се налага да се минава през model обекта.

Например, нека view-то показва списък с данни. Чрез обръщение към controller-а данните могат да бъдат подредени в азбучен или някакъв друг ред, което е проста операция и не налага данните да бъдат осъвременявани от model обекта.



Предимства на Model-View-Controller архитектурата

Използването на Model-View-Controller архитектурата има следните предимства:

1. Множество view обекти, ползващи един и същ model обект, а това от своя страна води до по-лесно имплементиране, тестване и управление на компонентите на модела.
2. По-лесна включване на нови потребители на системата, което на практика се свежда просто до добавяне на ново вид view.
3. Чистота на дизайна - при една добре проектирана система само един преглед на списъка методите с "public" достъп, предоставени от модела, дава ясна представа как и доколко данните могат да бъдат манипулирани.
4. Модулност - дизайна допуска лесна подмяна на едни компоненти с други (един view обект може да замени друг view

обект; асоциираният към view-то controller обект може да бъде заменен с друг controller обект, дори достъпвания от controller-a model обект- да бъде заменен с друг model обект), което води до лесно модифициране и поддръжка на цялото приложение.

5. Разширяемост - всеки един от компонентите може да бъде заменен с подобрена и осъвременена версия без да бъде засегнат никой от останалите компоненти.

6. Разпределимост - само чрез двойка proxy обекти (обекти, преобразуващи входно-изходните данни на приложението) всеки един софтуер може да комуникира с всеки друг, стига вторият да е проектиран съобразно изискванията на Model-View-Controller архитектурата.

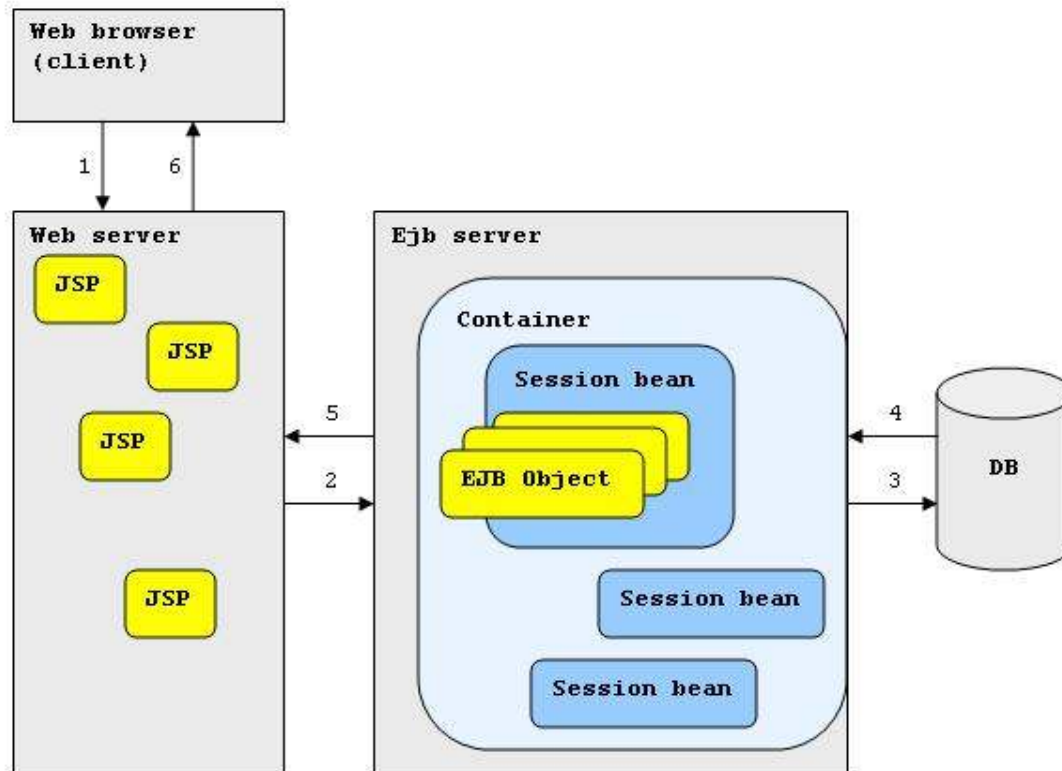
Архитектурата на системата "E-learning" следва Model-View-Controller архитектурата и следователно притежава всички предимства на този вид дизайн - гъвкавост, лесна модифицируемост, простота на използване. И всички тези характеристики взети заедно превръщат прилагането на този вид pattern в проектирането и реализирането на web-базирани приложения почти стандарт. Но ако се замислим по-внимателно над Model-View-Controller архитектурата ще установим, че тя улеснява програмирането и усъвършенстването на приложенията, но не прави нищо в посока бързодействие и оптимизация на работа от страна на клиента. Или с други думи този вид шаблон улеснява живота на програмиста, но с нищо не допринася за качеството на продукта, гледан от страна на крайния потребител. За постигане на по-висок коефициент на полезно действие в web-базирани приложения е утвърдило използването на редица други шаблони, като *The Façade Pattern*, *The Mediator Pattern*, *The Singleton Pattern*, *The Proxy Pattern* и т.н. Последните два са използвани и в системата "E-learning".

АРХИТЕКТУРА НА СИСТЕМАТА "E-School"

Архитектурата на системата за електронно обучение "E-School" е реализирана съобразно Model-View-Controller design pattern и съответно е разделена на 3 компонента:

- Model - Database, Java classes
- View - Web interface
- Controller - Enterprise Java Beans

Графиката по-долу представя в опростен вид сценария на взаимодействие между model-a, view-то и controller-a в приложението:



Както се вижда приложението е разделено на три основни слоя - web, application и база.

Web частта се състои от Java Server Pages (JSPs) като back-end и стандартен web browser като клиент на back-end-a. За да могат да се използват по предназначение, JSP файловете се нуждаят от определена среда, предоставяна от т.нар. web servers. На пазара има голямо разнообразие от web servers, като част от тях са специализирани единствено за поддръжка на JSP страници, а други изпълняват и много други функции. Такива Web servers са например JBoss, Apache Tomcat, Macromedia JRun Application Server, InQMy Application Server, SAP J2EE Engine и много други. В случая за контейнер на JSP файловете на приложението е използван JBoss 3.2.1.

Application частта, или иначе казано частта, която съдържа основната логиката на приложението, е капсулирана в Enterprise Java Beans (EJBs) обекти. Те, както и JSP обектите, се нуждаят от специален контейнер - application server. Има голямо разнообразие от application servers, продукти на различни производители. Често даден application server бива

едноверменно и web server. Такъв е случая и с използвания от приложението application server, който отново е JBoss 3.2.1.

Чрез използване на един и същ сървър за web и application частта се решават множество проблеми, които биха могли да възникнат между различни продукти от различни производители.

За база е използван MySQL сървър поради удобството, простотата на конфигуриране и стабилността, които предлага.

ИМПЛЕМЕНТИРАНЕ

Фазата имплементиране се занимава с разработването на класове и методи, реализиращи конкретизирания в предишната фаза дизайн.

В случая на системата "E-School" трябваше да се създадат класове и методи, реализиращи основните компоненти на дизайна и връзките между тях, а именно – създаване на таблици с определена схема в базата данни; създаване на класове и методи за работа с базата данни; създаване на потребителски итерфейс; създаване на класове и методи, обработващи данните от потребителския интерфейс.

Model

Използвани технологии: MySQL

Сървър: MySQL

Данните, показвани във view-то се пазят в model обекта, в случая – база данни и java classes.

Базата данни съдържа 2 вида таблици – едните пазят данните, които могат да се гледат и манипулират посредством приложението, а другите – данните, необходими за вътрешната организация и работа на програмата. Основните критерии, на които трябва да отговорят всички таблици са простота, логическа обособеност, бърз достъп и нормализация.

Достъпът от програмата до базата данни се осъществява посредством JDBC код (JAVA технология за работа с бази данни), вграден в Stateless Session Beans. За удобство и оптимизация на работата с базата (т.е. избягване на ненужни remote calls) за всяка от таблиците има съответен java class, в който се пазят извлечените данни или тези, които предстои да бъдат записани. По този начин се намалява броя на remote calls между View-то и

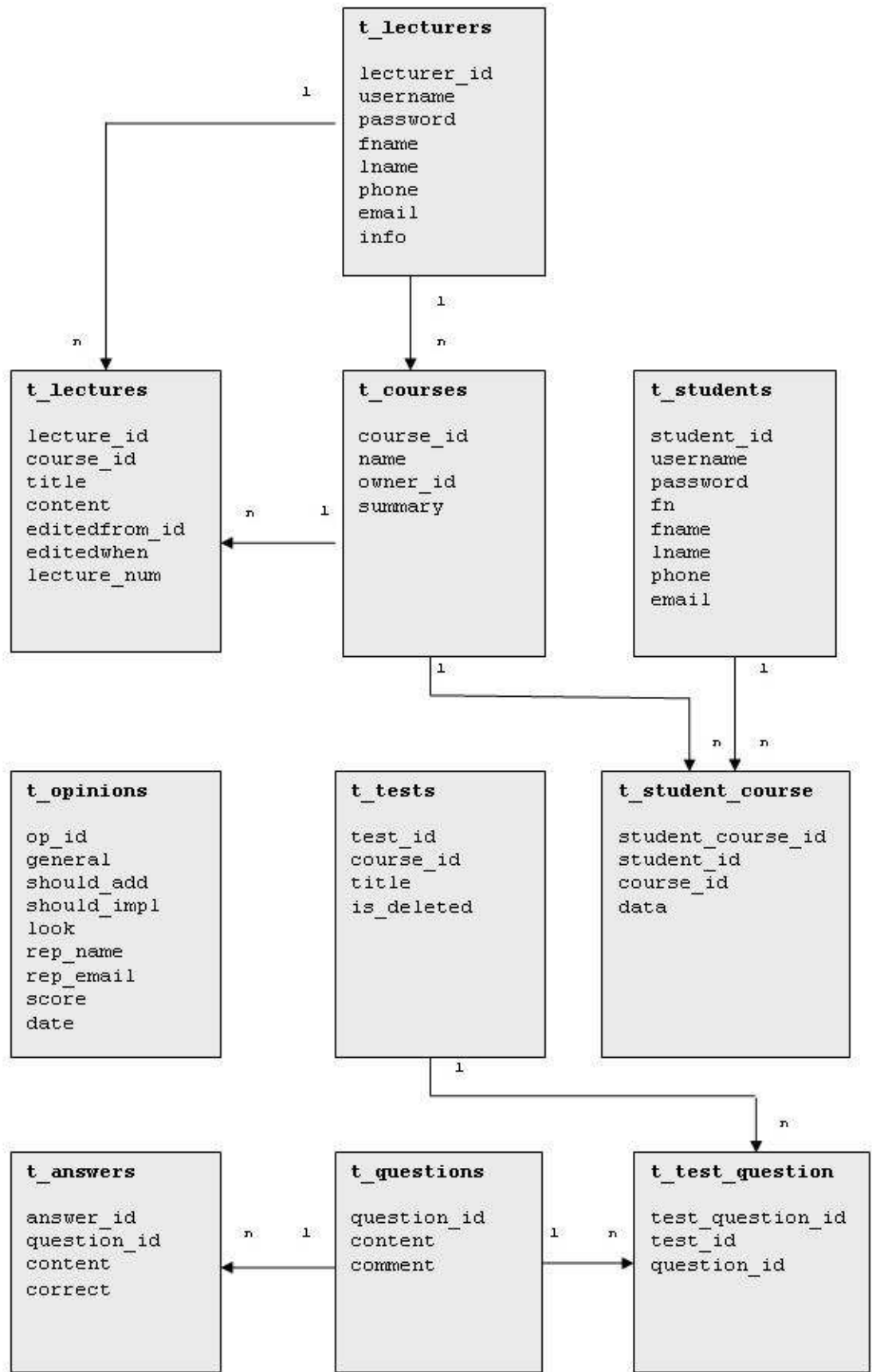
Controller-a от една страна и между Controller-a и Model-a от друга.

Тези класове-контейнери на данни, се намират в пакета *eschool.data* и имената им съответсват на имената на таблиците:

- *Answer.java*
- *Course.java*
- *Lecture.java*
- *Lecturer.java*
- *Question.java*
- *QuestionAnswer.java*
- *Rights.java*
- *Student.java*
- *StudentResult.java*
- *Test.java*
- *User.java*

Всяка инстанция от даден клас (т.е. всеки обект на класа) съдържа данните за един ред в съответната таблица. Този вид оптимизация реализира идеята, залегнала в същността на шаблона *Proxy Pattern*. Така при всеки remote call от view-то например данните за един логически обособен обект се изпращат заедно, което макар и да коства малко повече процесорно време за конструиране и попълване на обекта, в крайна сметка многократно увеличава бързодействието на програмата. Например ако от страна на view-то са необходими данните за един студент, за да се покаже страница с лична информация, то тези данни се пакетират от controller-a в инстанция на класа *Student*, и се изпращат всички заедно. Така няма да има нужда да се правят отделни remote calls за първо име, за презиме, за фамилия, за факултетен номер и т.н.

Схемата по-долу илюстрира таблиците в базата данни и връзките между тях:



View

Използвани технологии: JSP, HTML, JavaScript
Сървър: JBoss Application Server

Както стана ясно view обектите могат бъдат твърде разнообразни по своя вид и природа – едни и същи данни могат да бъдат показвани и модифицирани от command line view, desktop приложение или web базирано view.

За целите на системата "E-learning" view-то е реализирано чрез web interface.

Въпреки, че всички web страници, формиращи визуалния интерфейс на приложението, наричаме най-общо "view", те би трябвало да бъдат разделени в 3 категории, или subviews.

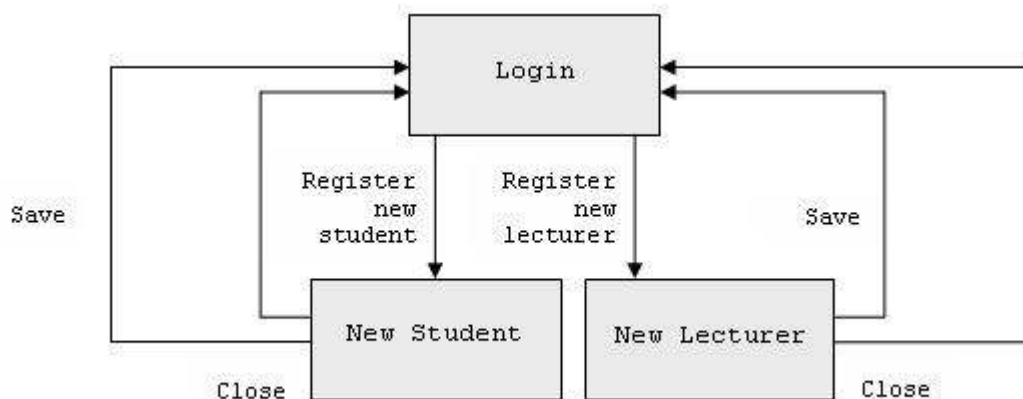
Всяка от категориите представлява различен поглед към данните, пазени в модела, и съответно предоставя различни възможности за манипулация на тези данни. За всяко от тези subview-та има съответно и определена потребителска група, като принадлежността към дадена група определя правата на потребителя за преглед и манипулация на данните. Потребителските групи са следните:

- преподаватели
- студенти
- администратори

При регистрация всеки потребител трябва да избере групата, към която принадлежи.

Последователността от стъпки за регистрация на потребител от коя да е от потребителските групи е следната:

РЕГИСТРАЦИЯ НА НОВ ПОТРЕБИТЕЛ



Както се вижда от схемата по-горе, регистрацията като потребител от група "студенти" става, като от Login екрана се избере хипер-връзката "Register a new student", попълнят се всички задължителни полета и се натисне бутона "Save". Веднага след това създадения акаунт е активен и потребителя може да влезе в системата с новото си потребителско име и парола.

Аналогично регистрацията като потребител от група "преподаватели" става, като от Login екрана се избере хипер-връзката "Register a new lecturer", попълнят се всички задължителни полета и се натисне бутона "Save". Новосъздадения акаунт се активира веднага.

Екрана за регистрация на нов потребител от група преподаватели:

E-school 1.0 - Register a new lecturer

All fields marked with * are mandatory

[User Manual](#)

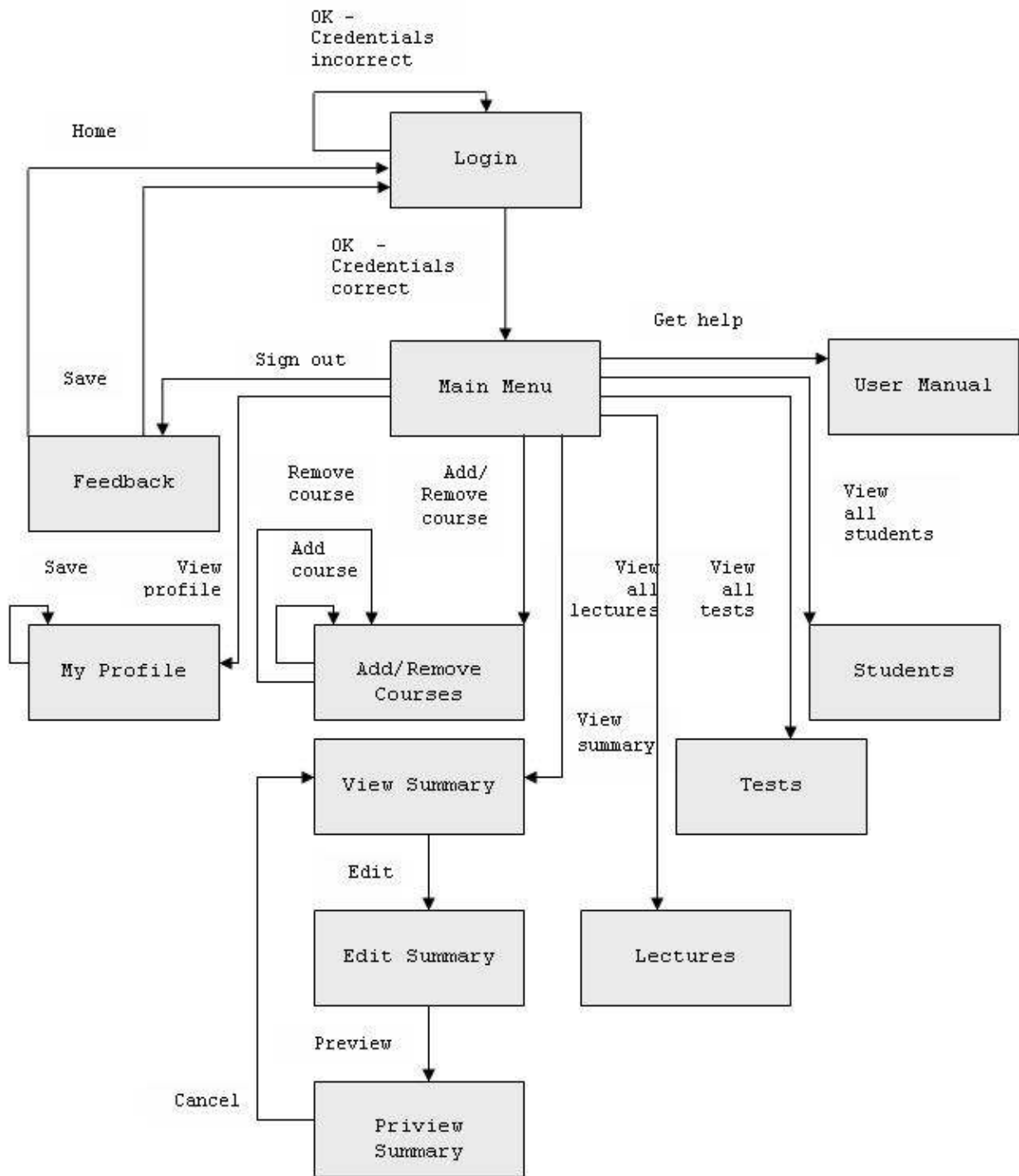
Login information	
* Username	<input type="text" value="gorka"/>
* Password	<input type="password" value="••••••"/>
* Confirm Password	<input type="password" value="••••••"/>
Personal Information	
* First Name	<input type="text" value="Gorka"/>
* Last Name	<input type="text" value="Ivanova"/>
Contact info	
* E-mail	<input type="text" value="gorka@mail.bg"/>
Phone	<input type="text" value="0888 288 687"/>
Additional Information	<input type="text" value="Some additional information about lecturer..."/>

Екранът за регистрацията на нов потребител от група студенти изглежда по подобен начин с изключение на полето "Additional Information", което липсва.

За момента регистрацията като потребител от група "администратори" не е възможна.

РАБОТА СЪС СИСТЕМАТА НА ПОТРЕБИТЕЛИ ОТ ГРУПА "ПРЕПОДАВАТЕЛИ"

Схемата по-долу дава най-обща представа за възможностите за манипулация на данните на потребителите от група "преподаватели" и последователността от стъпки, за постигане на определена промяна на данните.



Както се вижда от схемата потребителите от група "преподаватели" имат следните възможности за работа със системата:

1. Преглед и промяна на въведената лична информация, както и курсовете, които преподавателя води: Login – Main Menu – My Profile.

Екранът, който реализира тази функционалност, изглежда по следния начин:

E-school 1.0 / Lecturer View	
My Profile	My Profile Save
User Manual	Personal Information
Add / Remove Courses	
Sign Out	* Username <input type="text" value="gorka"/>
Analiz 1	* First Name <input type="text" value="Gorka"/>
Analitichna Geometria	* Last Name <input type="text" value="Ivanova"/>
Java	* E-mail <input type="text" value="gorka@mail.bg"/>
Pascal	Phone <input type="text" value="0888 288 687"/>
Scheme	Additional Information <input type="text" value="Some additional information about lecturer..."/>
Prolog	
	Owned Courses
	Java
	Pascal
	Scheme
	Prolog

2. Преглед на потребителската документация "User Manual" ("Наръчник на потребителя") – избира се хипервръзката "User Manual" от менюто вляво.

3. Добавяне и изтриване на курс, воден от този преподавател (автоматично се добавя в "Owned Courses") – промяната се записва само при валидна администраторска парола: Login – Main Menu – Add/Remove Courses.

За добавяне на нов курс е необходимо единствено да се въведат име на курса, кратката информация за съдържанието му и администраторската парола за добавяне на курс.

За изтриване на съществуващ курс е необходимо единствено да се селектира курса, който трябва да се изтрие и да се въведе администраторската парола за изтриване на курс.

Екранът, който реализира тази функционалност, изглежда по следния начин:

E-school 1.0 / Lecturer View	
My Profile	Add Course Save
User Manual	
Add / Remove Courses	
Sign Out	
Analiz 1	Administrator password <input type="password" value="....."/>
Analitichna Geometria	Course Name <input type="text" value="Prolog"/>
Java	Lecturer <input type="text" value="Gorka Ivanova"/>
Pascal	Summary <div style="border: 1px solid black; height: 80px;"></div>
Scheme	
	Remove Course Remove
	Administrator password <input type="password"/>
	<input type="checkbox"/> Java
	<input type="checkbox"/> Pascal
	<input type="checkbox"/> Scheme

4. Преглед и редактиране съдържанието на резюме на курс - промяната е възможна само ако текущия потребител е също и собственик на курса: Login - Main Menu - Summary - Edit Summary.

Екранът, който реализира тази функционалност, изглежда по следния начин:

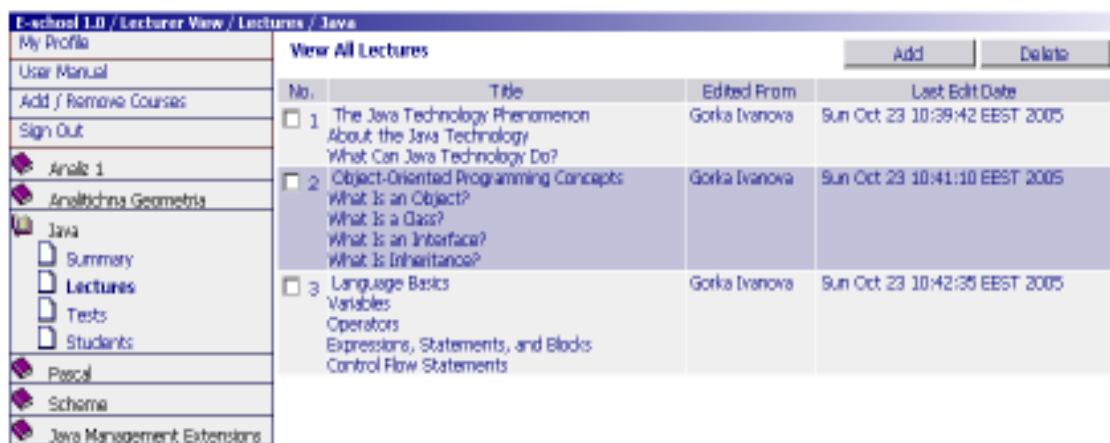
E-school 1.0 / Lecturer View / Summary / Java	
My Profile	View Summary Edit
User Manual	
Add / Remove Courses	
Sign Out	
Analiz 1	
Analitichna Geometria	
Java	
Summary	
Lectures	
Tests	
Students	
Pascal	
Scheme	
	<p style="text-align: center;">JAVA Course</p> <p>The lessons in this course give a quick introduction to Java programming. They tell you what Java is and provide you with an opportunity to compile and run some simple Java programs. Finally, they give you the background knowledge you need to understand how the programs work.</p> <p>In the course will be discussed the following topics:</p> <ul style="list-style-type: none"> - The Java Technology Phenomenon - Object-Oriented Programming Concepts - Language Basics - Classes and Inheritance - Handling Errors with Exceptions - Threads: Doing Two or More Tasks At Once - I/O: Reading and Writing - Accessing System Resources - Creating a GUI with JFC/Swing

5. Преглед и редактиране на всички лекции за даден курс: Login - Main Menu - Lectures.

Ако текущия потребител не е преподавателя, собственик на съответния курс, лекциите могат да бъдат само разглеждани (бутоните "Add" и "Delete" липсват).

Ако текущия потребител е преподавателя, собственик на курса, към списъка с лекции могат да бъдат добавяни нови, а съществуващите лекции могат да бъдат редактирани и изтритвани.

Екранът, който реализира тази функционалност, изглежда по следния начин:



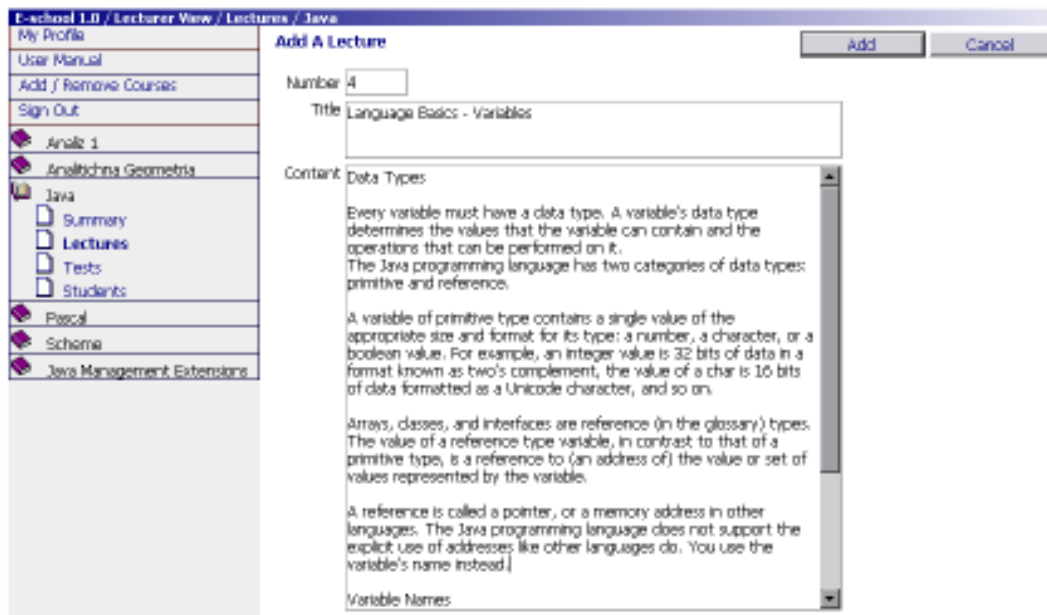
The screenshot shows a web application interface for a lecturer. On the left is a navigation menu with items like 'My Profile', 'User Manual', 'Add / Remove Courses', 'Sign Out', and a tree view for 'Java' containing 'Summary', 'Lectures', 'Tests', and 'Students'. The main area is titled 'View All Lectures' and contains a table with columns for 'No.', 'Title', 'Edited From', and 'Last Edit Date'. There are 'Add' and 'Delete' buttons at the top right of the table.

No.	Title	Edited From	Last Edit Date
<input type="checkbox"/> 1	The Java Technology Phenomenon About the Java Technology What Can Java Technology Do?	Gorka Ivanova	Sun Oct 23 10:39:42 EEST 2005
<input type="checkbox"/> 2	Object-Oriented Programming Concepts What is an Object? What is a Class? What is an Interface? What is Inheritance?	Gorka Ivanova	Sun Oct 23 10:41:10 EEST 2005
<input type="checkbox"/> 3	Language Basics Variables Operators Expressions, Statements, and Blocks Control Flow Statements	Gorka Ivanova	Sun Oct 23 10:42:35 EEST 2005

6. Преглед на съдържанието на конкретна лекция от курса.

7. Добавяне на нова лекция - промяната е възможна само ако текущия потребител е също и собственик на курса: Login - Main Menu - Lectures.

Екранът, който реализира тази функционалност, изглежда по следния начин:



8. Изтриване на лекция – промяната е възможна само ако текущия потребител е също и собственик на курса: Login – Main Menu – Lectures.

9. Редактиране на лекция – промяната е възможна само ако текущия потребител е също и собственик на курса: Login – Main Menu – Lectures.

10. Преглед на всички тестове в курса: Login – Main Menu – Tests.

11. Преглед на съдържанието на конкретен тест от курса: Login – Main Menu – Tests.

12. Добавяне на нов тест към курса – промяната е възможна само ако текущия потребител е също и собственик на курса: Login – Main Menu – Tests.

Екранът, който реализира тази функционалност, изглежда по следния начин:

E-school 1.0 / Lecturer View / Tests / Java

My Profile
 User Manual
 Add / Remove Courses
 Sign Out
 Analiz 1
 Analitichna Geometria
 Java
 Summary
 Lectures
Tests
 Students
 Pascal
 Scheme

Add A Test Save Cancel

Title: Java - Test 1

Question: A program written in the Java programming language can run on any platform because...

Comment: More than one answer can be correct

Answer 1: Java programming is derived from C++. Correct Incorrect

Answer 2: The Java Virtual Machine(JVM) interprets the program for the native operating system. Correct Incorrect

Answer 3: The compiler is identical to a C++ compiler. Correct Incorrect

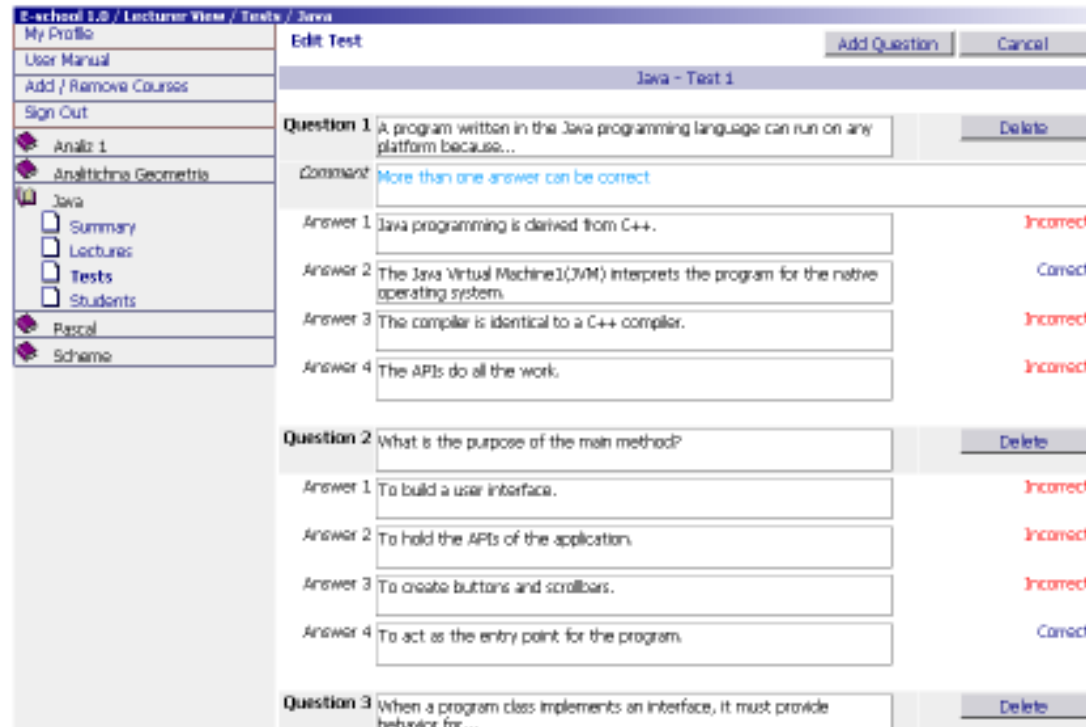
Answer 4: The APIs do all the work. Correct Incorrect

Answer 5: Correct Incorrect

При създаване на нов тест е задължително попълването на полето "Title". При добавяне на нови въпроси към съществуващ курс това поле не се попълва. Попълването на полето "Comment" не е задължително, като идеята е да се попълва само при необходимост за добавяне на допълнителна информация към текущия въпрос. Всеки въпрос може да има не повече от пет възможни отговора, като за всеки от тях преподавателя трябва да посочи дали е верен или грешен.

13. Редактиране и изтриване на тест - промяната е възможна само ако текущия потребител е също и собственик на курса: Login - Main Menu - Tests.

Екранът, който реализира тази функционалност, изглежда по следния начин:



14. Преглед на всички студенти, участващи в курса: Login - Main Menu - Students.

15. Добавяне на нови студенти към курса - промяната е възможна само ако текущия потребител е също и собственик на курса: Login - Main Menu - Students.

16. Преглед на текущите резултати на всеки студент: Login - Main Menu - Students.

17. Оценка на продукта от потребителите на системата: Login - Feedback.

Екранът, който реализира тази функционалност, изглежда по следния начин:

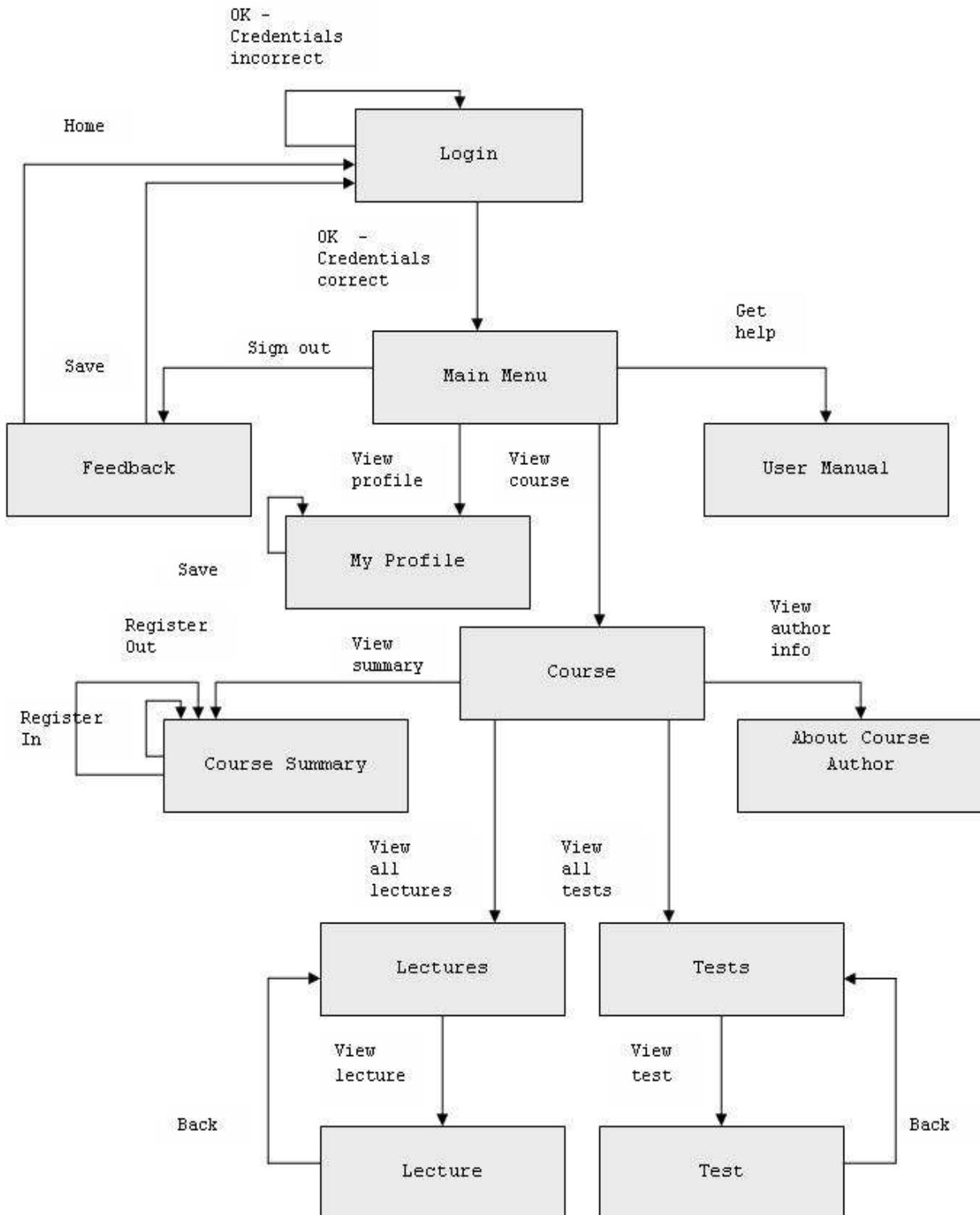
E-school 1.0

Thank you for using the E-school System. Your recommendations, suggestions and comments you send at gorka@abv.bg or you can fill the following form:

General opinion	<input type="text" value="Not bad learning management system but with some missing functionality."/>	
What should be added	<input type="text" value="Tools for editing, pictures, graphics, calendars..."/>	
What should be improved	<input type="text"/>	
Look and feel	<input type="text" value="Possibility for customization of look and feel of the views."/>	
Your name	<input type="text" value="Gorka Ivanova"/>	Your e-mail <input type="text" value="gorka@mail.bg"/>
Score the site	<input type="radio"/> no opinion <input type="radio"/> terrible <input type="radio"/> bad <input type="radio"/> middling <input checked="" type="radio"/> good <input type="radio"/> very good <input type="radio"/> excelent	

РАБОТА СЪС СИСТЕМАТА НА ПОТРЕБИТЕЛИ ОТ ГРУПА "СТУДЕНТИ"

Схемата по-долу илюстрира възможностите за манипулация на данните, които view-то предоставя на потребителя от група "студенти" и последователността от стъпки, за постигане на определена промяна на данните.



Както се вижда от схемата потребителите от група "студенти" имат следните възможности за работа със системата:

1. Преглед и редактиране на лични данни: Login - Main Menu - My Profile.

На този екран студентът може да види въведените при регистрацията лични данни и да редактира стойността на всяко едно поле с изключение на полетата "Username" и "FN".

На същия екран може да се видят всички курсове, за които студентът се е записал, както и резултатите от тестовете, които е положил.

Екранът, който реализира тази функционалност, изглежда по следния начин:

Welcome **Bojidara Ivanova**

E-school 1.0 / Student View / My Profile

My Profile	My Profile	<input type="button" value="Save"/>
User Manual	Personal Information	
Sign Out		
Analiz 1	* Username	<input type="text" value="bobeto"/>
Analitichna Geometria	* FN	<input type="text" value="50103"/>
Java	* First Name	<input type="text" value="Bojidara"/>
Pascal	* Last Name	<input type="text" value="Ivanova"/>
Scheme	* E-mail	<input type="text" value="bobeto@mail.bg"/>
Java Management Extensions	Phone	<input type="text" value="0888 433 123"/>

Courses

Java	Gorka Ivanova	gorka@mail.bg	0888 288 687
Pascal	Gorka Ivanova	gorka@mail.bg	0888 288 687
Analitichna Geometria	gg gg	gg	mob. 0888 288 687 / home: 123
Java Management Extensions	Georgi Mihaylov	georgi_mihaylov@fmi.uni-sofia.bg	0899 324 019 / 02 66 71 12

Test Results

2. Преглед на "User Manual" ("Наръчник на потребителя").

3. Преглед на съществуващите курсове - менюто вляво съдържа всички създадени до момента курсове: Login - Main Menu.

4. Преглед на резюме на курс и включване в / изключване от курс: Login - Main Menu - Course - Course Summary.

При избиране на хипервръзката <курс>/Summary се вижда кратка информация за съдържанието на курса.

Включването в курса става чрез бутона "Register In". Ако студента вече е включен в текущия курс, бутона "Register In" е заменен от бутон "Register Out".

Екранът, който реализира тази функционалност, изглежда по следния начин:

Welcome **Bojidara Ivanova**

E-school 1.0 / Student View / Summary / Java Management Extensions	
<ul style="list-style-type: none"> My Profile User Manual Sign Out Analiz 1 Analitichna Geometria Java Pascal Scheme Java Management Extensions <ul style="list-style-type: none"> Summary About Author Lectures Tests Results 	<p>View Summary Register In</p> <p>You are still not registered in the course</p> <div style="border: 1px solid black; padding: 5px;"> <p>JavaTM Management Extensions Course</p> <p>The JavaTM Management extensions (also called the JMX specification) define an architecture, the design patterns, the APIs, and the services for application and network management and monitoring in the Java programming language. This chapter introduces all of these elements, presenting the broad scope of these extensions. The rest of this document, along with its corresponding Javadoc API, represents the first phase of the JMX specification.</p> </div>

5. Преглед на данните на преподавателя, автор на даден курс: Login - Main Menu - Course - About Course Author.

Екранът, който реализира тази функционалност, изглежда по следния начин:

Welcome **Bojidara Ivanova**

E-school 1.0 / Student View / Author / Java Management Extensions											
<ul style="list-style-type: none"> My Profile User Manual Sign Out Analiz 1 Analitichna Geometria Java Pascal Scheme Java Management Extensions <ul style="list-style-type: none"> Summary About Author Lectures Tests Results 	<p>About Course Author</p> <table border="1"> <tbody> <tr> <td>First Name</td> <td>Georgi</td> </tr> <tr> <td>Last Name</td> <td>Mihaylov</td> </tr> <tr> <td>Phone</td> <td>0899 324 019 / 02 66 71 12</td> </tr> <tr> <td>E-mail</td> <td>georgi_mihaylov@fmi.uni-sofia.bg</td> </tr> <tr> <td>Additional Information</td> <td></td> </tr> </tbody> </table>	First Name	Georgi	Last Name	Mihaylov	Phone	0899 324 019 / 02 66 71 12	E-mail	georgi_mihaylov@fmi.uni-sofia.bg	Additional Information	
First Name	Georgi										
Last Name	Mihaylov										
Phone	0899 324 019 / 02 66 71 12										
E-mail	georgi_mihaylov@fmi.uni-sofia.bg										
Additional Information											

6. Преглед на лекциите в даден курс: Login - Main Menu - Course - Lectures.

Екранът, който реализира тази функционалност, изглежда по следния начин:

E-school 1.0 / Student View / Lectures / Java									
My Profile	Lectures <table border="1"> <thead> <tr> <th>Number</th> <th>Title</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>The Java Technology Phenomenon About the Java Technology What Can Java Technology Do?</td> </tr> <tr> <td>2</td> <td>Object-Oriented Programming Concepts What Is an Object? What Is a Class? What Is an Interface? What Is Inheritance?</td> </tr> <tr> <td>3</td> <td>Language Basics Variables Operators Expressions, Statements, and Blocks Control Flow Statements</td> </tr> </tbody> </table>	Number	Title	1	The Java Technology Phenomenon About the Java Technology What Can Java Technology Do?	2	Object-Oriented Programming Concepts What Is an Object? What Is a Class? What Is an Interface? What Is Inheritance?	3	Language Basics Variables Operators Expressions, Statements, and Blocks Control Flow Statements
Number		Title							
1		The Java Technology Phenomenon About the Java Technology What Can Java Technology Do?							
2		Object-Oriented Programming Concepts What Is an Object? What Is a Class? What Is an Interface? What Is Inheritance?							
3		Language Basics Variables Operators Expressions, Statements, and Blocks Control Flow Statements							
User Manual									
Sign Out									
Analiz 1									
Analitichna Geometria									
Java									
Summary									
About Author									
Lectures									
Tests									
Results									
Pascal									
Scheme									
Java Management Extensions									

7. Преглед съдържанието на конкретна лекция в курса: Login – Main Menu – Course – Lectures – Lecture.

8. Преглед на всички тестове в даден курс: Login – Main Menu – Course – Tests.

9. Попълване на тест за даден курс: Login – Main Menu – Course – Tests – Test.

Екранът, който реализира тази функционалност, изглежда по следния начин:

E-school 1.0 / Student View / Tests / Java	
My Profile	Test yourself Save
User Manual	
Sign Out	
Analiz 1	
Analitichna Geometria	
Java	Java - Test 1
Summary	Question: A program written in the Java programming language can run on any platform because...
About Author	<i>More than one answer can be correct</i>
Lectures	Answers
Tests	1. Java programming is derived from C++. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
Results	2. The Java Virtual Machine(JVM) interprets the program for the native operating system. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
Pascal	3. The compiler is identical to a C++ compiler. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
Scheme	4. The APIs do all the work. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
Java Management Extensions	Question: What is the purpose of the main method?
	Answers
	1. To build a user interface. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
	2. To hold the APIs of the application. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
	3. To create buttons and scrollbars. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
	4. To act as the entry point for the program. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
	Question: When a program class implements an interface, it must provide behavior for...
	Answers
	1. Two methods defined in that interface. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect
	2. Only certain methods in an interface. <input type="radio"/> Correct <input checked="" type="radio"/> Incorrect

10. Преглед на резултатите на студента от попълнените тестове.
11. Оценка на продукта - възможност за обратна връзка с потребителите на системата: Login - Feedback.

РАБОТА НА СИСТЕМАТА НА ПОТРЕБИТЕЛИ ОТ ГРУПА "АДМИНИСТРАТОРИ"

- преглед и промяна съдържанието на всички таблици
- промяна на администраторската парола
- изтриване на таблици

Controller

Използвани технологии: Enterprise Java Beans
Сървър: JBoss Application Server

Controller обекта съдържа логиката за работа с данните. Той предоставя методи за преглед и промяна на данните, които се виждат в view-то. Понякога данните, които трябва да се предоставят на view-то трябва да се вземат от модел обекта, т.е. базата данни, но това не е задължително.

Логиката на controller-а се съдържа в класове, наречени Enterprise Java Beans и по-точно Stateless Session Beans.

Enterprise JavaBeans (EJB) технологията е "server-side component" архитектура за Java 2 Platform, Enterprise Edition (J2EE) platform. EJB технологията дава възможност за бърза и опростена разработка на разпределени, сигурни и преносими приложения базирани на Java технологията.

Всеки Enterprise Java Bean представлява компонент, предоставящ интерфейс за работа с него под формата на методи. Имплементира се посредством Java езика и главните му характеристики са:

- операциите, които може да извършва;
- възможността за многократна употреба;
- независимост от използвания хардуер, операционна система, сървър или мрежови протокол;

Поради тези причини логиката на приложението е изцяло капсулирана в имплементацията на 3 enterprise java beans, намиращи се в пакета "eschool.ejb" - _CoursesBean, _LecturesBean и _UsersBean.

Всеки от тях съдържа методи за работа в съответната област – методи за манипулация на курсовете, методи за манипулация на лекциите и методи за манипулация на потребителските данни.

ТЕСТВАНЕ

Фазата тестване е от изключително значение. Тя се занимава с проверка коректността на готовия продукт в няколко направления, като основните три са:

- проверка дали готовото приложение покрива всички потребителски изисквания, изброени в спецификацията на проекта;
- проверка дали реализираната функционалност работи по коректен начин;
- проверка дали реализацията отговаря на определени стандарти за оптималност и ефективност;

Тестовете могат да бъдат както ръчни така и автоматизирани с помощта на специализирани приложения за тестване.

В случая на реализираната примерна система "E-School" тестването е от първия вид тъй като автоматичното тестване е неприложимо за тестване коректността на потребителски интерфейс.

ДОКУМЕНТИРАНЕ

Документирането на един проект е последната фаза от разработването му. Документацията се дели на два основни вида – документация, предназначена за потребителите на приложението (user manuals, help pages, tutorials) и документация, предназначена за технически лица, които ще трябва да поддържат приложението и евентуално да го развият, тестват и стабилизират.

Потребителската документация може да бъде част от самото приложение, т.е. вградена в самата програма – като бутон или хеперлинк "help", даващ бърз достъп до основните функции на приложението, а може да бъде и под формата на отделен документ, който не е част от приложението.

Пример за вградена потребителска документация е бутона Help в основното меню на повечето Microsoft Office Tools. Пример за втория тип потребителска документация е "Наръчник на потребителя" на JBoss сървър, който не е част от самия сървър.

Техническата документация обикновено не е част от приложението. Тя включва спецификации, дизайн документи, документи описващи класовете и методите на приложението и т.н.

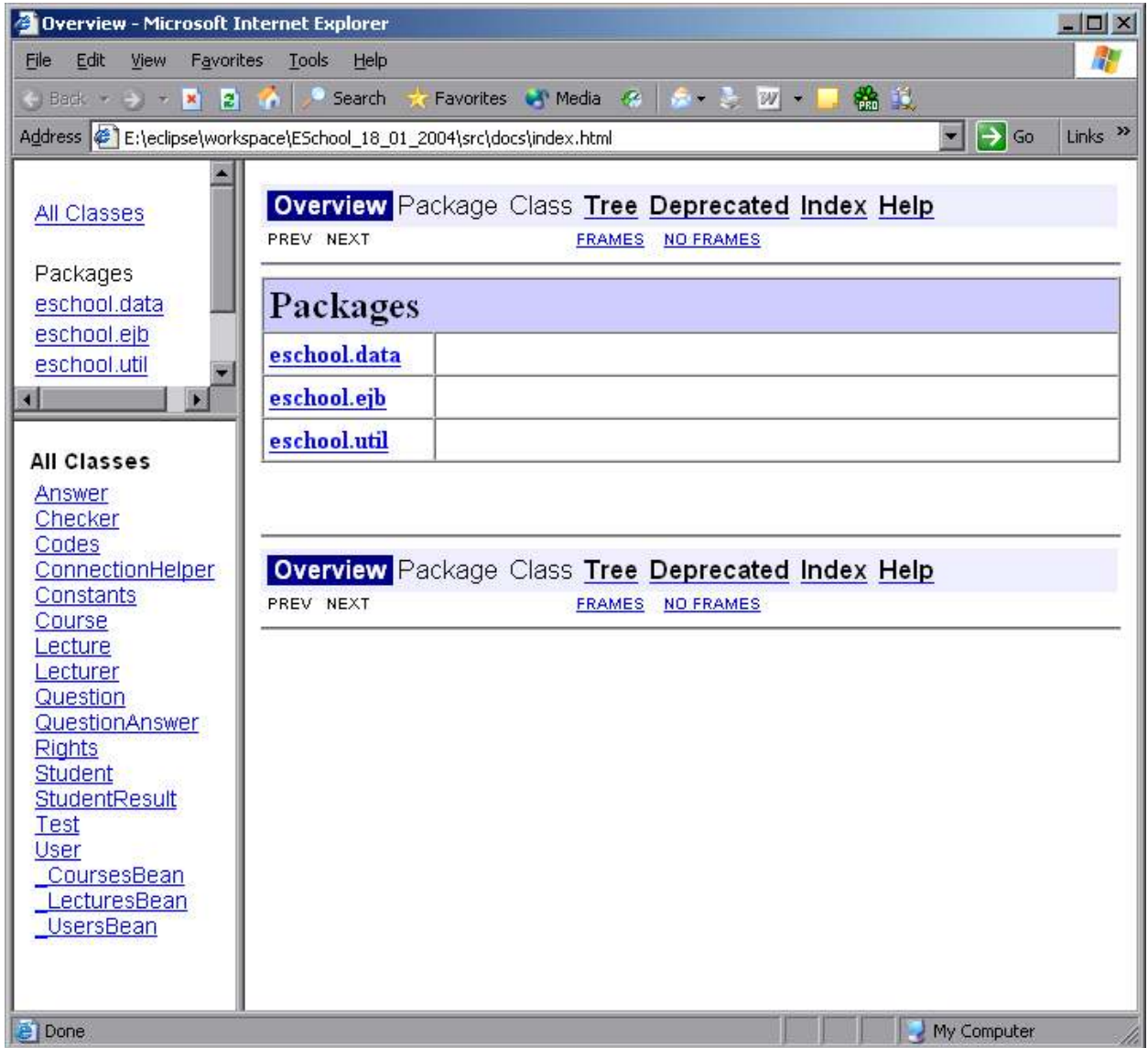
Системата "E-School" включва и двата основни вида документация – потребителска и техническа. Към дипломната работа е приложена приложението java documentation (техническа документация), това изложение (техническа документация) и "User Manual" страницата (потребителска документация).

Java documentation (javadoc) е кратко описание на действието, входните параметри и изходния резултат на всеки метод в Controller частта.

Java документацията е предназначена предимно за програмисти и други технически лица. Този вид документирание е широкоизползван поради множеството предимствата, които предоставя, като това че:

- прави програмния код по-лесен за разбиране – често програмиста губи много време, за да разбере какво прави даден метод, особено ако са минали месеци от създаването му. В този случай java документацията оказва безценна помощ;
- прави възможно и програмисти, които не са писали определен програмен код да се включат ефективно в развиването и коригирането му;
- прави кода много по-лесен за манипулиране и развиване;
- помага да се следи в детайли развитието на проекта дори и от не-програмисти;

Използвайки вградената в Java SDK библиотеката – генератор на javadoc, е лесно да се получи удобна за разглеждане и стандартизирана на външен вид документация в html формат. Заглавната страница на документацията изглежда по следния начин:



Менюто вляво предоставя бърз и удобен достъп към подробна информация за всеки от класовете в проекта. Цялата java документация представлява самостоятелен сайт, съставен от 38 .html файла и един .css файл. Главната страница на сайта с документацията може да се види при отваряне на файла `javadocs\index.html` в произволен web browser.

“User Manual” страницата е достъпна от всеки екран на приложението. Тя също е в html формат и е предназначена за потребителите на системата – преподаватели, студенти и администратори.

В “User Manual” страницата има кратко описание на всеки екран, достъпен за определена потребителска група и на

възможностите, които той предлага. Организирана е по следния начин:



и т.н.

Пълния текст на "User Manual" може да се види [ТУК](#).

ЗАКЛЮЧЕНИЕ

Целта на тази дипломна работа е да направи обзор на понятието "електронно обучение", на възможностите, които то предоставя на потребителите и на предизвикателствата, които поставя пред производителите на софтуер.

Направен е кратък обзор на по-известните стандарти, организации и спецификации в областта, както и на сайтове и фирми, специализирани в разработката на онлайн обучение.

С помощта на Java езика и няколко специализирани приложения за разработване на web базирани системи (database server, application server, web server) е разработената информационна система "E-School", илюстрира характерните особености на понятието, като в същото време показва процеса на разработка на един такъв специфичен вид приложения.

Примерното приложение е вид "Learning Management System" т.е. интерактивна среда чрез която съдържанието на системата може да бъде разширявано (чрез добавяне на нови курсове), редактирано (чрез добавяне / изтриване / редактиране на съществуващия учебен материал) или просто да се разглежда като обикновена система за електронно обучение. Системата поддържа два вида потребители - преподаватели и студенти, като потребителския интерфейс и предоставените възможности за тях са различни.

РАЗВИТИЕ

Приложението може да бъде развито и подоброено в няколко направления.

➤ Съдържание :

- лекциите, тестовете, както и списъка на студентите съдържат малко и есествено тестови данни, които могат да бъдат заменени от реални;

➤ Функционалност :

- във view-то биха могли да бъдат добавени екрани за администриране на приложението, т.е. за работа на потребителите от група "администратори". Тези екрани би трябвало да предоставят възможност за манипулация на таблиците в базата данни, но това не означава, че това са екрани, които просто заменят стандартен sql клиент. Администраторските екрани би трябвало да дават подробна информация за табилците в базата, връзките между тях и ефекта, които би имала промяната или изтриването на данни;
- добавяне на нови методи в controller-a - ясно е, че добавянето на администраторски екрани във view частта автоматично води до добавяне на нови методи за предоставяне на данните за тези view-та и манипулацията им;
- имплементацията на екраните за попълване на тест и съхраняване на резултата от страна на студента би могла да се разшири;
- би могла да се добави функционалност за запазване на вече изтрити данни в базата. Това би могло да е

изключително полезно например, когато студента е попълнил тест, който вече е изтрит или за получаване на стари резултати на студенти;

- добавяне на помощни инструменти за по-удобна работа на потребителите като вградени календари, текстови редактори, възможност за използване на html, css и javascript в изложението на лекциите, възможност за използване на аудио и видео материали към лекциите, чат, форум;

➤ **Look & feel:**

- външния вид на всяко от jsp-та формиращи view частта може да бъде променен и подобрен;
- би могло да се добави функционалност за промяна на цветовете настройки, размера или вида на шрифта съобразно предпочитанията на потребителя;

ЛИТЕРАТУРА

Belanger, F., & Jordan, D. H. (2000) Evaluation and Implementation of Distance Learning: Technologies, Tools and Techniques. Idea Group Publishing, Hershey USA, London UK.

Campione Mary, Walrath Kathy, Huml Alison (3rd Edition) The Java(TM) Tutorial: A Short Course on the Basics.

Интернет сайт: <http://www.e-learningguru.com>

Интернет сайт: <http://www.elearningguild.com/>

Интернет сайт: www.blackboard.com

Интернет сайт: <http://www.icus.net/elearning/elearnstandards.shtm>

Интернет сайт: <http://www.e-learningcentre.co.uk/>

ПРИЛОЖЕНИЕ

BLOGGING TOOLS

Tool: Big Blog Tool (Big Guy Media)

URL: www.bigblogtool

Tool: Blog Studio (Indigo Technology Partners)

URL: www.blogstudio.com

Tool: Blogger (Pyra Labs)
URL: www.blogger.com

COLLABORATION TOOLS - APPLICATION SHARING

Tool: Exchange Conferencing Server (Microsoft)
URL: www.microsoft.com/exchange

COLLABORATION TOOLS - AUDIO CONFERENCING

Tool: Exchange Conferencing Server (Microsoft)
URL: www.microsoft.com/exchange

Tool: Robust Audio Tool (University College London)
URL: www-mice.cs.ucl.ac.uk/multimedia/software/rat/

Tool: vat (Lawrence Berkeley National Laboratory)
URL: www-nrg.ee.lbl.gov/vat/

COLLABORATION TOOLS - CHAT AND INSTANT MESSAGING

Tool: Chat Blazer (Pendulab Pte Ltd)
URL: www.chatblazer.com

Tool: ChatSpace Community Server (Akiva)
URL: www.akiva.com

Tool: Dbabble (NetWin, LTD)
URL: www.netwinsite.com

Tool: Exchange Server (Microsoft)
URL: www.microsoft.com/exchange

COLLABORATION TOOLS - E-MAIL

Tool: Eudora Email (Qualcomm)
URL: www.eudora.com

Tool: MSN Hotmail (Microsoft)
URL: www.hotmail.com

Tool: Netscape Mail (Netscape)
URL: www.netscape.com

Tool: Outlook (Microsoft)
URL: www.microsoft.com/office/outlook/

COLLABORATION TOOLS - NEWS READER

Tool: Agent (Forté Internet Software)
URL: www.forteinc.com

Tool: CoffeeLink News Server (Burton Computer Corporation)
URL: www.burton-computer.com

Tool: DNews Server (NetWin, LTD)
URL: www.netwinsite.com

COLLABORATION TOOLS - ONLINE DISCUSSION

Tool: CoffeeLink News Server (Burton Computer Corporation)
URL: www.burton-computer.com

Tool: Dbabble (NetWin, LTD)
URL: www.netwinsite.com

Tool: DiscussionApp (Server Corporation)
URL: www.server.com

Tool: WebBoard (Akiva)
URL: www.akiva.com

COLLABORATION TOOLS - VIDEO CONFERENCING

Tool: Exchange Conferencing Server (Microsoft)
URL: www.microsoft.com/exchange

Tool: vic (Lawrence Berkeley National Laboratory)
URL: www-nrg.ee.lbl.gov/vic/

COLLABORATION TOOLS - VOTING

Tool: Poll Pro (AdComplete.com)
URL: www.pollpro.com

Tool: The Survey System (Creative Research Systems)
URL: www.surveysystem.com

COLLABORATION TOOLS - WEB TOUR

Tool: Multicity Value Package (Multicity)
URL: www.multicity.com

COLLABORATION TOOLS - WHITEBOARD

Tool: Groupboard (User Data Connections Limited)
URL: www.groupboard.com

Tool: wb (Lawrence Berkeley National Laboratory)
URL: www-nrg.ee.lbl.gov/wb/

COLLABORATION TOOLS - FILE FORMAT

Tool: Batch Converter (Sonic Foundry)
URL: www.sonicfoundry.com

Tool: Canvas (ACD Systems of America Inc.)
URL: www.deneba.com/canvas

CONTENT CONVERTERS - GENERAL

Tool: Acrobat (Adobe Systems)
URL: www.adobe.com

CONTENT CONVERTERS - POWERPOINT

Tool: Agility Presenter (Anystream, Inc.)
URL: www.agilitypresenter.com

Tool: Breeze (Macromedia)
URL: www.macromedia.com/software/breeze/

Tool: HotFoot (Interactive Video Technologies)
URL: www.digitallava.com

CONTENT CONVERTERS - WORD

Tool: HTML Transit (Avanstar)
URL: www.avantstar.com

Tool: WordToWeb (Solutionsoft)
URL: www.solutionsoft.com

CONTENT MANAGEMENT SYSTEMS

Tool: AuthorIT (AuthorIT)
URL: www.authorIT.com

Tool: Manila (UserLand Software)
URL: manila.userland.com

Tool: Microsoft Content Management server 2002 (Microsoft)
URL: www.microsoft.com/cmserver/

Tool: UserLand Frontier (UserLand Software)
URL: frontier.userland.com

COURSE AUTHORIZING TOOLS

Tool: Authorware (Macromedia)
URL: www.macromedia.com

Tool: Lectora Publisher (Trivantis)
URL: www.lectora.com

Tool: Quest (Allen Communication Learning Services)
URL: www.allencomm.com

Tool: Web Course Builder (ReadyGo)
URL: www.readygo.com

HELP AUTHORIZING TOOLS

Tool: AuthorIT (AuthorIT)
URL: www.authorIT.com

Tool: Doc-To-Help (ComponentOne)
URL: www.componentone.com

LEARNING CONTENT MANAGEMENT SYSTEMS

Tool: Aspen Learning Content Management Server (Click2learn)
URL: www.click2learn.com

Tool: CentreLearn (CentreLearn)
URL: www.centrelearn.com

Tool: Docent Enterprise Learning Content Management System (Docent)
URL: www.docent.com

Tool: e-Education Suite (WebsiteASP)
URL: www.omniupdate.com

Tool: ePath Learning ASAP (ePath Learning)
URL: www.ephathlearning.com

Tool: FlexTraining (Online Development)
URL: www.flextraining.com

Tool: iPerformance (Online Courseware Factory)
URL: www.courseware-factory.com

Tool: Knowledge Pathways (Global Knowledge)
URL: www.globalknowledge.com

LEARNING MANAGEMENT SYSTEMS

Tool: ABC Academy (Danish Probe)
URL: www.danishprobe.com

Tool: Active Learner (Resource Development Corporation)
URL: www.resourcedev.com

Tool: Boniva Learning Management (Boniva Software)
URL: www.boniva.com

Tool: Digital Think Learning Management (DigitalThink)
URL: www.digitalthink.com

Tool: eLearning Server 3000 (HyperMethod Company)
URL: www.hypermethod.com

Tool: Learning Management Services (Edcor)
URL: www.edcor.com

MEDIA EDITORS - ANIMATION

Tool: 3D Canvas Pro (Amabilis Software)
URL: www.amabilis.com

Tool: Flash MX (Macromedia)
URL: www.macromedia.com/flash

Tool: GIF Animator (Ulead Systems)
URL: www.ulead.com

MEDIA EDITORS - AUDIO

Tool: Acid Pro (Sonic Foundry)
URL: www.sonicfoundry.com

Tool: SONAR (Cakewalk)
URL: www.cakewalk.com

MEDIA EDITORS - GRAPHICS

Tool: AutoCAD (AutoDesk)
URL: www.autodesk.com

Tool: CorelDraw (CompuServe Incorporated)
URL: www.corel.com

Tool: Photoshop (Adobe Systems)
URL: www.adobe.com

Tool: Visio (Microsoft)
URL: www.microsoft.com/office/visio

MEDIA EDITORS - MULTIMEDIA

Tool: Director MX (Macromedia)
URL: www.macromedia.com

Tool: Flash MX (Macromedia)
URL: www.macromedia.com/flash

Tool: LiveMotion (Adobe Systems)
URL: www.adobe.com

VIRTUAL-SCHOOL SYSTEMS

Tool: Blackboard Learning System (Blackboard)
URL: www.blackboard.com

Tool: eCollege (eCollege)
URL: www.ecollege.com

Tool: Lotus Learning Space (IBM)
URL: www.lotus.com