



Уеб програмиране

Курсова работа

Версия 2.0.0

Въведение

По време на обучението в Компютърна Академия IT Step, студентите, които се обучават в Софтуерно Разработване, не само минават изпити, но и подготвят курсови работи по дадена тема.

Нека помислим, защо учебната програма съдържа курсови работи? Има няколко отговора на този въпрос. Първия отговор е: Важно е студентите да работят по сложен проект, съдържащ множество технологии. Например, ако си говорим за Java, голям набор от Java технологии трябва да се използват. Втория отговор е: При завършек на обучението, студентите трябва да имат портфолио, а курсовите работи са точно това, което търсим.

Проекта за курса по програмиране се състои от 2 части: клиент и сървър. За да се имплементира клиентската среда, HTML /CSS , JavaScript , jQuery/ AngularJS/ReactJS и тн. трябва да се използват. За да се имплементира сървърната среда, PHP или ASP.NET MVC трябва да се използват. DBMS трябва да се използват за складиране на данни.

Избиране на Тема

Академията и учителят, който води определен курс не са задължени да възлагат определена курсова тема на студента. Бихме могли да предложим тема, но не сме задължени да го правим. Много е важно да вдъхновим студента да избере тема сам. Защо?

Ако човек избере проект, в който е заинтересован поради някаква причина, мотивацията той да работи по проекта е много по-висока. Ако избрания проект е работа, свързана с истинска реклама, имаме перфектния курсов проект. Разбира се, избраната тема за курсовия проект трябва да отговаря на целите в него.

Одобрение на Курсовата Работа

Темата на курсовата работа на студента трябва да бъде одобрена от Главния Отдел един или два месеца преди защитата и.

Студента описва тема на курсовата работа и създава списък от възможности на курсова работа в кандидатурата, състоящ се не по-малко от 2 страници. През първия етап, издадената кандидатура се одобрява от учителя,

отговорен за имплементация на курсови работи. Например, ако говорим за курсова работа по уеб програмиране, тогава това е учителя, който преподава PHP или ASP.NET. През втория етап, темата се одобрява от главния отдел на софтуерно разработване на бранша. През третия етап, приложението на студента се получава от Обучаващия Мениджър на специфичния отдел.

Кандидатурата се издава в две копия. Едно копие за студентите и едно такова за Преподаващия Мениджър. Може да бъде и в електронен вариант. Например, Преподаващия Мениджър държи сканираните кандидатури подписани.

Основната цел на целия този одобряващ процес е да се потвърди, че предложеният проект отговаря на курсовата работа, що се отнася до технологиите, сложност и компетентност. Подписите на учителя и главния отдел потвърждават, че са анализирали темата и са уверени, че курсовият проект следва горната критерия. Ако бранша няма главен отдел, тогава само учителят подписва кандидатурата.

Моля, имайте предвид, че курсовите работи могат да се представят от един или единствен студент или група от студенти. В случай на второто, обхвата на работата на всеки студент от групата трябва да бъде равен на обхвата на курсовата работа за един студент.

Начин на работа

Студента работи по курсовата работа у дома, т.е. извън академията. При възникване на въпроси, студента задава въпроса към учителя, в рамките на изученото.

В учебната програма за редовно обучение и ECO, времето е разпределено за консултации и защита. По-добре да се разпредели предназначеният. Хубаво е да се избягва изучаването на много неща накуп, напротив - по-добре е да се разпредели материала, предназначен за две седмици на няколко такива, с цел по-добра подготовка за студента. Даването на допълнително време за консултации и двойни обучения са отговорност на обучаващия мениджър.

За частични обучения, предназначеният време е отделено за защита.

Защита на курсовата работа

Датата на защитата на курсовата работа се определя от преподаващия мениджър след консултация с учителите и главния отдел.

Обучаващия мениджър (ако е приложим), учителят отговорен за курсовата работа на студента и главния отдел трябва да присъстват по време на защитата. Главният отдел предоставя кандидатурите на студентите, които са били задържани от обучаващия мениджър до защитата на проекта.

По време на защитата, студентите трябва да имат хартиен документ, който включва:

1. Заглавие с името на темата, име на курсовия проект, пълното име на студента;
2. Копие от кандидатурата на студента;
3. UML-ите (UMLs) на проекта. Диаграма на класът и диаграма за използваем случай са задължителни. Други диаграми са по желание;
4. Схема от база данни (ако е приложимо);

5. Централния код на проекта (source code; ако студента не е готов да демонстрира такъв на негови проект поради някаква причина, тогава се показват само ключовите части на кода)

Защитата протича под формата на презентация. Студента разказва за своята част/работа пред комисията. Презентации от типа на PowerPoint са пример в този случай. В рамките на презентацията, студента трябва да обясни за своя проект: защо беше избрана темата, да спомене бизнес съперници, да обясни използваните технологии и тн. След като студента приключи с разговора за проекта си, трябва да демонстрира как работи той на практика.

Комисията може да зададе всякакви въпроси, касаещи проекта по време на защитата. Ако комисията има въпрос, засягащ централния (source) код, студентът трябва да демонстрира частта на кода чрез дискусия.

Максималното време за говорене е 15 минути, препоръчителното време за въпроси, зададени от комисията е 15 минути. Общото време предназначено за работа с един студент по време на защитата е 30 минути.

След като всички студенти са завършили защитата си, комисията обсъжда оценките на колегиална основа (без студентите). Всеки студент е оценен чак след като комисията или мнозинството стигнат до решение. След като са обявени всички оценки, главния отдел ги казва преди студентите веднага след официалното решение. Ако студента повдигне под въпрос оценката си, главния отдел трябва да обясни всичко за оценката индивидуално, след като са обявени всички оценки.

Ако студента се провали или желае по-висока оценка, защитата се прави наново. Датата се определя от обучаващия мениджър след консултация с учителите и главния отдел.

Идеи за курсови работи по Уеб Програмиране

1. Уеб Чат

- *Кратко описание:* студента трябва да имплементира уеб чат. Чата трябва да съдържа история на чата, списък от контакти, частно и групово разменен е на съобщения, черен списък, смяна на състоянията на потребител и те. Имплементацията на уеб чата трябва да включва администратори, които могат да забраняват достъпа на потребители нарушаващи правилата за комуникация.

2. Уеб Портал за Тестове

- *Кратко описание:* проект за уеб тестове. Тестовите случаи и потребителски списък се складира в базата данни. Системата държи записи за всеки един взет тест. Портала трябва да включва разни типове въпроси и прикачване на изображения. Софтуера се състои от 2 части: тестов редактор и тестова система. Тестовите се създават в тестов редактор и се складира в базата данни. Всеки тест се състои от няколко въпроса. Колкото се изисква отговора могат да се добавят към даден въпрос. Има 2 вида въпроси: с единичен отговор и с няколко отговора. Всеки въпрос има своята тежест. Изображение може да се прикачва към въпрос. Тестовата система повтаря тестовите за потребителите. Потребителя се вписва в своя акаунт. Имат ограничен брой опити за тестване. Резултатите се поставят в базата данни.

3. Уеб Проект за Гласуване

- *Кратко описание:* уеб проект, който предоставя средства за гласуване. Може да се гласува за Президент на Класът или нещо подобно. Потребителите могат да гласуват за техния кандидат. Системата се състои от 2 части: клиентска страна за гласуване и администраторски интерфейс. Потребителя на гласуващата система се регистрира през клиентската страна. След това, той може да гласува. Един потребител може да гласува само веднъж. Системният администратор може да създава разни гласове, да добавя, изтрива и редактира кандидати, да задава стартово и крайно време за гласуване и тн.

4. Уеб Организатор

- *Кратко описание:* уеб организатор, който позволява на потребителите да покажат своите задачи. Задачите могат да бъдат персонализирани или да имат списък с под-задачи. Една задача съдържа приоритет, крайна дата, описание, хаштаг, коментар и прикачен файл. Потребителя може да добавя, редактира, изтрива и търси всяка задача. Трябва също да може да вижда списък от задачи за ден, седмица, месец и година.

5. Уеб Система за Създаване на Прости Профили

Клиентската страна и администраторския интерфейс са задължителни.

6. Уеб Система за Къщи под Наем

Клиентската страна и администраторския интерфейс са задължителни.