

Департамент образования города Москвы

**Государственное бюджетное образовательное учреждение
высшего профессионального образования города Москвы
«Московский городской педагогический университет»**

**Институт математики, информатики и естественных наук
Кафедра прикладной информатики**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

**Тема: Проектирование и реализация электронного
образовательного ресурса с веб доступом по дисциплине
«веб технологии»**

**Выполнила:
студентка группы О-Б-06
Лапаева Мария Александровна
Руководитель:
Иванова К.Н., доцент, к.т.н.**

Москва 2014

- Корреспондентское обучение
- Обучающие радио передачи
- Обучающие теле передачи
- Обучающие программы
- Использование электронной почты
- Обучающие сайты
- Онлайн семинары(вебинары)

**История развития
дистанционного обучения**

- Библиотеки
- Словари
- Энциклопедии
- Ресурсы для абитуриентов
- Ресурсы дополнительной деятельности
- Ресурсы дистанционного образования
- Ресурсы для учителей, методистов

Виды образовательных ресурсов

- Анализ и проектирование сайта
 - Разработка структуры сайта
 - Разработка функционала сайта
 - Выбор способа хранения данных
 - Выбор языка программирования
- Написание кода
- Наполнение сайта
- Тестирование
- Публикация
- Раскрутка
- Поддержка

Этапы создания ресурса

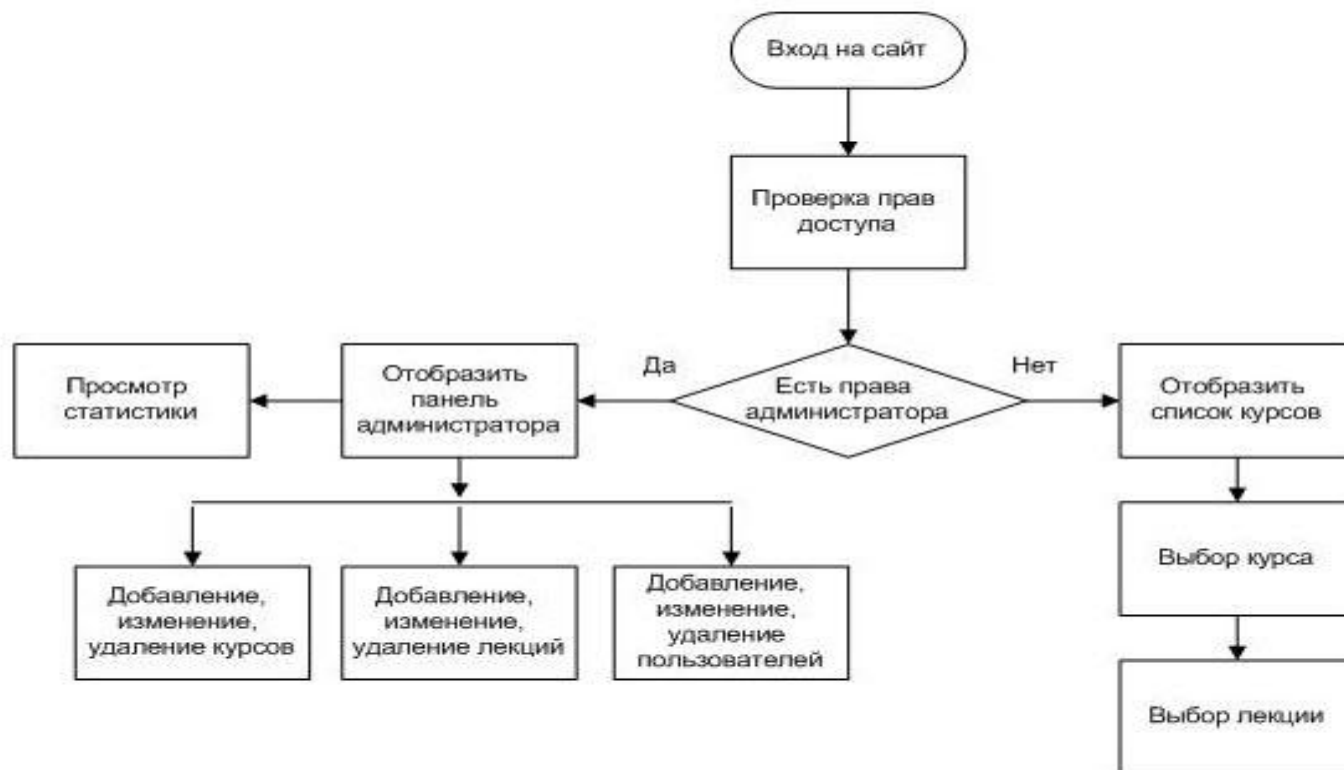
| | |
|--|------------------------------|
| Здесь, обычно, располагается логотип, меню. | |
| Здесь навигация | Основное содержимое страницы |
| Здесь контактная информация, ссылки на сайты партнеров | |

- `<html><body>`
- `<p>`
- `<table border="1" width="690">`
- `<tr> <td>Здесь, обычно, располагается логотип, меню/<td>`
`</tr>`
- `<tr> <td>`
- `<table width="690" border="1">`
`<tr>`
- `<td width="190"> Здесь`
навигация
`

`
`</td>`
- `<td> Основное содержимое`
сайта
`

`
`</td>`
`</tr>`
- `</table> </td>`
- `</tr><tr>`
- `<td>Здесь контактная`
информация, ссылки на сайты
партнеров
`</td>`
- `</tr></table>`
- `</body></html>`

Структура сайта



Функционал сайта

- Текстовые файлы
- Типизированные файлы
- Статические страницы
- Базы данных

Способ хранения данных

- Реляционная база данных — база данных, основанная на реляционной модели данных. Слово «реляционный» происходит от англ. *relation* («отношение», «зависимость», «связь»).
- Таблицы в реляционных базах данных обладают рядом свойств :
 - В таблице не допускается двух одинаковых строк. В математике такие таблицы называются отношениями (по-английски *relation*, отсюда и название - реляционные).
 - Столбцы создаются вместе с таблицей и имеют определенный порядок. В таблице должен быть хотя бы один столбец.
 - Все значения столбца – однотипны.
 - Столбец имеет имя, уникальное в пределах таблицы.
 - На пересечении столбца и строки находится только одно значение.
- Реляционные базы данных разработаны для быстрого сохранения и получения больших объемов информации.
- Для этого выполняются нормализация баз данных.

Реляционные базы данных

- Нормализация - это механизм создания логической модели базы данных.
- Для ее выполнения необходимо решить следующие задачи :
 - - сгруппировать атрибуты в отношения предметной области;
 - - сгруппировать атрибуты в отношения баз данных.
- Данные задачи имеют большое множество решений и схем отношений. Для выбора наиболее рационального из них необходимо соблюдать ряд правил:
 - Первичные ключи должны быть минимальными (минимальность ключа).
 - Число отношений должно давать наименьшую избыточность данных (надежность данных).
 - Число отношений не должно влиять на производительность системы (производительность системы).
 - Данные не должны быть противоречивыми (непротиворечивость данных).
 - Схема отношений должна быть устойчивой, способной адаптироваться к изменениям при ее расширении дополнительными атрибутами (гибкость структуры).
 - Скорость выполнения различных запросов должна быть одинакова (производительность).
 - Данные должны отображать актуальную картину предметной области (актуальность).

Нормализация баз данных

- 1) Отношение находится в первой нормальной форме тогда и только тогда, когда все атрибуты отношения являются простыми, то есть не имеют компонентов. Иными словами атрибут должен состоять из неделимых значений и не может включать в себя более мелкие атрибуты.
- 2) Отношение находится во второй нормальной форме тогда и только тогда, когда оно находится в первой нормальной форме и каждый не ключевой атрибут неприводимо зависит от ключа.
- 3) Отношение находится в третьей нормальной форме тогда и только тогда, когда оно находится во второй нормальной форме и в нем отсутствуют транзитивные связи

Нормальные формы баз данных

- Курс :

- - название курса;
- - краткое описание курса;
- - полное описание курса.

- Лекция :

- - название курса;
- - название лекции;
- - краткое описание;
- - полный текст лекции;
- - просмотры.

- Пользователь:

- - ФИО;
- - возраст;
- - логин;
- - пароль;
- - права пользователя.

Хранимые данные

● Курсы(Subject)

| Поле | Описание | Тип данных | Ограничения |
|--------|-------------------------------|------------|-------------------------|
| ids | Идентификационный номер курса | Int | РК, счетчик, не пустой |
| name | Название курса | Char | 255 символов, не пустой |
| meta_d | Краткое описание | Char | 255 символов |
| text | Полное описание | Text | |

Атрибуты сущностей Курсы

● Лекции (Lesson)

| Поле | Описание | Тип данных | Ограничения |
|-------------|--------------------------------|------------|-------------------------|
| idl | Идентификационный номер лекции | Int | PK, счетчик, не пустой |
| title | Название лекции | Char | 255 символов, не пустой |
| ids | Идентификационный номер курса | Int | не пустой, FK |
| description | Краткое описание | Char | 255 символов |
| text | Полное описание | Text | не пустой |
| views | Счетчик просмотров | Int | не пустой |

Атрибуты сущностей Лекции

● Пользователи(User)

| Поле | Описание | Тип данных | Ограничения |
|-------|--------------------------------------|------------|-------------------------|
| idu | Идентификационный номер пользователя | Int | PK, счетчик, не пустой |
| fam | Фамилия | Char | 18 символов, не пустой |
| name | Имя | Char | 18 символов |
| ot | Отчество | Char | 18 символов |
| age | Возраст | Int | не пустой |
| pol | Счетчик просмотров | Int | не пустой |
| login | Логин пользователя | Char | 255 символов, не пустой |
| pass | Пароль пользователя | Char | 255 символов, не пустой |
| idg | Права пользователя | Int | не пустой |

Атрибуты сущностей
Пользователь

- SELECT
select_выражение1,
select_выражение2,
- ...
- [FROM
table_references
- [WHERE
where_definition]
- [ORDER BY
{число | имя_столбца
|
- формула}
- [ASC | DESC],
- ...]]

- INSERT
[LOW_PRIORITY |
DELAYED] [IGNORE]
- [INTO]
имя_таблицы
[(имя_столбца,...)]
- VALUES
(выражение,...),(...),...

Язык программирования SQL
Извлечение и вставка данных

- UPDATE
[LOW_PRIORITY]
[IGNORE]
имя_таблицы
- SET
имя_столбца1=выражение1
- [,
имя_столбца2=выражение2, ...]
- [WHERE
where_definition]
- [LIMIT число]

- DELETE
[LOW_PRIORITY]
FROM имя_таблицы
- [WHERE
where_definition]
- [LIMIT rows]

Язык программирования SQL
Изменение и удаление данных

- ресурс
mysql_connect (
[строка server [,
строка username [,
строка password [,
логическое
new_link [, целое
client_flags]]]])

- логическое
mysql_select_db (
строка
database_name [,
ресурс
link_identifier])

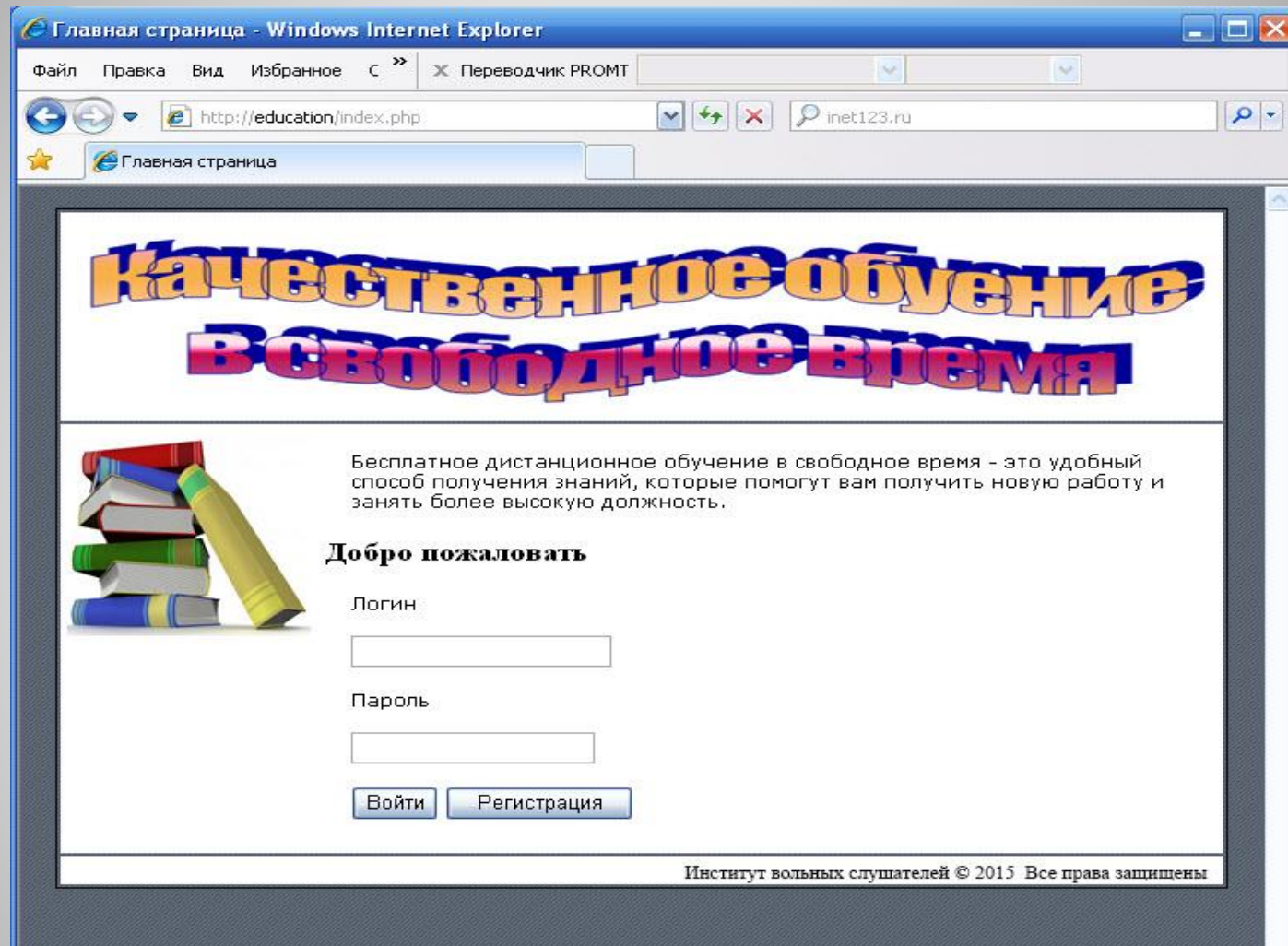
Язык программирования PHP
Соединение с базой данных

- ресурс
mysql_query (
строка query [,
ресурс
link_identifier])
- массив
mysql_fetch_array (
ресурс result [,
целое result_type])
- число
mysql_num_rows
(ресурс result)

Язык программирования PHP
Извлечение и работа с данными

-
- if (!\$result = mysql_query(Запрос к базе данных))
- {
- echo "<p>Вывести сообщение
Код ошибки:</p>";
- exit(mysql_error());
- }
- if (!mysql_num_rows(\$result))
- exit ("База пуста");
- while (\$myrow = mysql_fetch_array(\$result))
-

Программирование сайта



Заключение