

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
НАБЕРЕЖНОЧЕЛНИНСКИЙ ИНСТИТУТ

СОЗДАНИЕ WEB-ПРИЛОЖЕНИЯ

Учебно-методическое пособие
по дисциплине
«WEB-ПРОГРАММИРОВАНИЕ»

Набережные Челны
2018

Галиуллин Л.А. Создание web-приложения: учебно-методическое пособие по дисциплине «Веб-программирование» [Электронный ресурс] / Казанский федеральный университет, Электронный архив, 2018.

Рассматриваются проблемы проектирования и реализации современных web-приложений. Представлены методики формирования интерфейса web-приложений, рассмотрена работа с СУБД. Приведены контрольные вопросы. Для студентов направлений подготовки «Информатика и вычислительная техника», «Программная инженерия».

Введение

Веб-программирование – раздел программирования, ориентированный на разработку веб-приложений (программ, обеспечивающих функционирование динамических сайтов Всемирной паутины).

Языки веб-программирования — это языки, которые в основном предназначены для работы с веб-технологиями. Языки веб-программирования можно условно разделить на две пересекающиеся группы: клиентские и серверные.

Как следует из названия, программы на клиентских языках обрабатываются на стороне пользователя, как правило, их выполняет браузер. Это и создает главную проблему клиентских языков – результат выполнения программы (скрипта) зависит от браузера пользователя. То есть, если пользователь запретил выполнять клиентские программы, то они исполняться не будут, как бы ни желал этого программист. Кроме того, может произойти такое, что в разных браузерах или в разных версиях одного и того же браузера один и тот же скрипт будет выполняться по-разному. С другой стороны, если программист возлагает надежды на серверные программы, то он может упростить их работу и снизить нагрузку на сервер за счет программ, исполняемых на стороне клиента, поскольку они не всегда требуют перезагрузки (генерацию) страницы. Самыми распространенными клиентскими языками программирования являются:

- HTML
- CSS
- JavaScript
- VBScript
- ActionScript
- Java
- CoffeeScript

Когда пользователь дает запрос на какую-либо страницу (переходит на неё по ссылке или вводит адрес в адресной строке своего браузера), то вызванная страница сначала обрабатывается на сервере, то есть выполняются все программы, связанные со

страницей, и только потом возвращается к посетителю по сети в виде файла. Этот файл может иметь расширения: HTML, PHP, ASP, ASPX, Perl, SSI, XML, DHTML, XHTML.

Работа программ уже полностью зависима от сервера, на котором расположен сайт, и от того, какая версия того или иного языка поддерживается. К серверным языкам программирования можно отнести: PHP, Perl, Python, Ruby, любой .NET язык программирования (технология ASP.NET), Java, Groovy.

Важной стороной работы серверных языков является возможность организации непосредственного взаимодействия с системой управления базами данных (или СУБД) – сервером, на котором упорядоченно хранится информация, которая может быть вызвана в любой момент. Популярными среди систем управления базами данных являются:

- Firebird
- IBM DB2
- IBM DB2 Express-C
- Microsoft SQL Server
- Microsoft SQL Server Express
- mSQL
- MySQL
- Oracle
- PostgreSQL
- SQLite
- Sybase Adaptive Server Enterprise
- ЛИНТЕР
- MongoDB

Задание на курсовую работу/проект

Создать информационную систему «Интернет-магазин» (на заданную тему) с использованием web-технологий. Должны выполняться следующие условия:

- 1) База данных товаров должна включать не менее 3-х таблиц (пользователи и заказы не считаются)
- 2) Используется СУБД MySQL 5.0
- 3) Исполняемые скрипты написаны на языке PHP (не ниже

- 5.0)
- 4) В качестве веб-сервера используется Apache 1.3.x (register_globals = off)
 - 5) Вход в систему должен быть авторизованным
 - 6) При последующих обращениях к системе пользователь должен автоматически идентифицироваться
 - 7) Список товаров для продажи – не менее 20 наименований
 - 8) Количество записей во вспомогательных таблицах – не менее 5
 - 9) Система должна позволять заказывать товары (помещать в корзину заказов) и редактировать существующий заказ
 - 10) Система должна предлагать варианты оплаты и доставки товара. Соответственно меняется и итоговая сумма
 - 11) Для администраторов должна быть предусмотрена возможность добавления/удаления товаров, пользователей
 - 12) Менеджеры должны иметь возможность просматривать существующие заказы, а также изменять их статус (выполнен/выполняется/отменен/ и пр.)

Дополнительные требования:

- 1) Таблицы с товарами должны выводиться с помощью PHP
- 2) Значения полей форм должны проверяться на соответствие типов и на наличие тегов HTML как на стороне сервера, так и на стороне клиента (JavaScript)

Требования к оформлению КР (14pt, полуторный интервал, 45-50 стр.)

- 1) Титульный лист
- 2) Содержание
- 3) Описание задачи (то, что должна делать Ваша программа)
- 4) Варианты использования системы (то, что можно сделать в Вашей системе)
- 5) Описание баз данных
 - a) структура таблиц (имена полей, типы)
 - b) описание таблиц (для чего нужна каждая таблица, ее назначение)

- c) схема баз данных (рисунок с указанием связей между таблицами)
- d) SQL запросы на создание таблиц
- 6) Назначение скриптов (описание каждого скрипта, то, что он делает)
- 7) Описание интерфейса
 - a) описание основных экранов
 - b) описание основных форм в соответствии со списком вариантов использования системы
- 8) Тесты для проверки работоспособности системы
 - a) название и содержание тестов
 - b) список выполнения тестов (выполнено, не выполнено)

Исходные тексты программ с комментариями.

Выбор темы

Тема курсовой работы (проекта) выбирается преподавателем или самостоятельно с одобрения преподавателя. При самостоятельном выборе курсовой работы необходимо сразу оценить сложность, чтобы удовлетворить всем требованиям (количество таблиц, типы и количество связей и т.п.). Из этого следует, что необходимо нарисовать примерную схему базы данных, которая будет использоваться в вашей работе и показать преподавателю.

В качестве темы курсовой работы (проекта) может быть, например: разработка систем учета, контроля, обработки заказов, мониторинга состояния, системы поддержки принятия решений и т.п. В отдельных случаях возможна работа над созданием систем для автоматизированного ввода и обработки данных.

После выбора темы каждому проекту назначается имя, которое нельзя изменять. Это имя используется для названия каталога с проектом и для базы данных.

Установка программного обеспечения

Для выполнения курсовой работы (проекта) необходимо наличие следующего программного обеспечения: web-сервер Apache, СУБД MySQL, интерпретатор PHP.

Каждый из этих программных продуктов можно установить и настроить по отдельности, но можно воспользоваться пакетом «всё-в-одном» DenWer, который самостоятельно установит и настроит все три компонента.

Установка базового пакета DenWer

Установка базового пакета DenWer обычно не вызывает трудностей, если в качестве ответов на вопросы выбирать значения по умолчанию. После успешной установки на рабочем столе появятся 3 ярлыка: «Start servers», «Restart Servers» и «Stop servers». Для проверки работоспособности нужно запустить ярлык «Start servers», далее загрузить браузер (например, Internet Explorer) и в адресной строке набрать «http://localhost». Если установка прошла успешно, то появится страница «Ура, заработало!», на которой объясняется как пользоваться пакетом DenWer и даны ссылки на некоторые дополнительные утилиты (например, phpMyAdmin).

Формулирование требований к системе

Начальный этап разработки – это анализ предметной области, потребностей потенциальных пользователей и формулирование требований к системе. Результат анализа представляется в виде небольшого обзора (объем 1 - 2 страницы А4). Например, результат анализа предметной области при разработке АИС «Преподаватель»:

«Большинство преподавателей в настоящее время тратят в среднем от 1 до 2 астрономических часов в неделю на анализ работы со студентами в течение семестра. Сюда входит контроль посещаемости, учет сдачи лабораторных работ, контроль хода работ над курсовым заданием. Приходится

работать с большим количеством документов и не всегда удастся обойтись без ошибок. Особенно это сказывается при проведении промежуточных аттестаций. А чтобы оценить общее состояние по всем группам, по всем предметам нужно затратить не менее 2-х астрономических часов.

Преподаватели хотят иметь удобный и понятный инструмент для контроля учебного процесса, который избавит их от рутинной работы и позволит в любое время получать подробную информацию

Обо всех показателях учебного процесса. Подобное нововведение сэкономит значительную часть времени за счет того, что вся нужная для преподавателя информация находится в одном месте. Ни одна «бумажка» теперь не потеряется, а проставление аттестации будет производиться в полуавтоматическом режиме».

Цели проекта и критерии успеха

Для повышения успешности проекта нужно четко представлять, чего нужно достигнуть, т.е. сформулировать цели проекта. Основное требование к цели – конкретность. Например, цели для АИС «Преподаватель»:

Бизнес-цель 1. Уменьшить количество затрачиваемого за неделю времени на контроль учебного процесса на 50% (до 1 астрономического часа).

Бизнес-цель 2. Уменьшить количество ошибок при аттестации студентов в 2 раза.

Бизнес-цель 3. Уменьшить время на получение сводной информации по всем группам, по всем предметам до минимума в 2 минуты.

После определения целей проекта, необходимо сформулировать критерии успешности проекта. Критерии должны поддаваться измерению, т.е. нужно задать правило, которое позволит четко определить, является проект успешным или нет.

Критерий успеха 1. От работы с системой не откажутся 50% преподавателей в течение первых 2-х месяцев работы.

Критерий успеха 2. 75% преподавателей станут

использовать систему в следующем семестре.

Группы пользователей

Следующий этап – это определение групп пользователей, которых так или иначе будут взаимодействовать с проектируемой системой.

| Заинтересованные лица | Понимание основной ценности проекта | Отношение | Основные интересы | Ограничение |
|-----------------------|--|--|-----------------------------|---|
| Преподаватели | Увеличение точности учета, сокращение затрат личного времени | Большой энтузиазм, но могут использовать систему меньше, чем ожидается | Сокращение расходов времени | Возможно потребуются обучение работе с системой |

Функциональные возможности

Функциональные возможности системы – это то, что может делать система, т.е. то, что нужно будет реализовать в программном коде. Функциональные возможности разрабатываемой системы должны быть определены еще до начала написания какого-либо программного кода.

Основные функции 1. Создание, редактирование, удаление группы студентов.

Основные функции 2. Создание, редактирование, удаление студента.

Основные функции 3. Создание, редактирование, удаление предмета и его рабочей программы.

Основные функции 4. Ввод информации по посещаемости занятий.

Бизнес-правила

Бизнес-правила – это формализованные или неформализованные правила, которые используются при работе. Обычно бизнес-правила задают ограничения для программ, из которых состоит система. Основные трудности вызывают неформализованные бизнес-правила, которые используются не очень часто, т.е. многие сотрудники используют такие правила, даже не задумываясь о том, что кому-то это может быть неизвестно. Задача разработчика – выявить и формализовать такие правила.

Примеры бизнес-правил для АИС «Деканат»:

Бизнес-правило 1. Количество студентов в группе может быть от 1 до 35.

Бизнес-правило 2. Название группы в общем случае имеет следующий формат: XXXX[-A], где X – цифры, A – буквы, квадратными скобками обозначена необязательная часть.

Варианты использования системы

Варианты использования системы – это сценарии работы пользователей в системе. Сценарий реализует одну или несколько функциональных возможностей. Разработка вариантов использования производится по шаблону (см. пример).

Пример. Вариант использования из АИС «Преподаватель»:

| | | | |
|-------------------------|--|-----------------------------------|----------------|
| Название | Создание группы | | |
| Автор | Галиуллин Л.А. | Последнее обновление | Галиуллин Л.А. |
| Дата создания | | Дата последнего обновления | |
| Действующие лица | Преподаватель | | |
| Описание | Преподаватель создает в системе новую группу студентов, у которых он ведет занятия | | |

| | |
|-----------------------------------|--|
| Предварительные условия | 1. Преподаватель загрузил систему «Преподаватель» |
| Выходные условия | 1. В системе появляется новая группа 2. Обновляется список доступных групп |
| Нормальное направление | 1.0. Создание одной группы 1. Преподаватель делает запрос на создание группы (нажимает кнопку) 2. Система просит пользователя ввести имя группы 3. Пользователь вводит имя группы 4. Пользователь подтверждает создание группы 5. Система записывает данные о новой группе в свою базу 6. Система выводит сообщение о результате операции создания группы 7. Пользователь нажимает ОК |
| Альтернативные направления | 1.1. Создание нескольких групп (ответвление после пункта б) 1. Пользователь делает запрос на создание еще одной группы 2. Возврат к пункту 2. |
| Исключения | 1.0.И.1 Недопустимое количество символов в имени группы (в пункте 3) 1. Система извещает пользователя, что имя группы слишком длинное 2а. Пользователь отменяет ввод 2б. Система завершает вариант использования 3а. Пользователь решает исправить ввод 3б. Система начинает вариант использования сначала |
| Включает | - |
| Приоритет | высокий |
| Частота использования | приблизительно 10-15 раз в начале семестра |
| Бизнес-правила | Бизнес-правило 2 |

| | |
|----------------------------|---|
| Особые требования | Нет |
| Допущения | Предполагается, что в 30% случаев группы будут создаваться по несколько сразу |
| Замечания и вопросы | Нет |

Разработка архитектуры системы

Декомпозиция системы на модули

Специфика web-технологии заключается в том, что программный проект не содержит исполняемых файлов, которые подключает библиотеки функций, а состоит из множества скриптовых файлов, которые выполняют вполне конкретные задачи. Разделение программного кода на модули производится по функциональному признаку. В большинстве случаев для каждого варианта использования будет отдельный модуль – файл с программой на языке PHP. Однако в некоторых случаях бывает полезно выделить дополнительные модули в системе.

Во-первых, желательно написать собственный модуль для работы с базой данных. Все обращения к БД на запись или чтение производить только через функции этого модуля и подключать его всегда, когда требуется обращение к базе данных. Такой подход имеет ряд преимуществ:

- При переходе на другую СУБД, необходимо будет переписать только несколько функций
- Конфигурирование соединения с БД можно производить, изменяя 1 файл
- Облегчается тестирование SQL-запросов
- Снижение временных затрат на доработку и сопровождение программного кода системы

К недостаткам относится – увеличенное время на первоначальную разработку.

Во-вторых, желательно все функции, которые реализуют бизнес-логику системы выделить в отдельный модуль, который также подключать по мере необходимости. Преимущество этого

подхода – облегчается тестирование всех значимых функций системы, облегчается доработка и сопровождение кода.

Проектирование базы данных

С точки зрения программиста, база данных – это набор таблиц, содержащих информацию. Проектирование базы данных сводится к определению количества таблиц, определению типа и количества полей в таблицах, а также к определению связей между ними. Проектирование базы данных – достаточно трудоемкий и ответственный процесс. В случае допущения ошибки на этом этапе или при неправильном проектировании, может потребоваться изменить значительную часть кода проекта.

Проектирование баз данных подробно рассматривается в курсе «Базы данных». В данном проекте от студента требуется применить на практике полученные знания.

Результатом работы на этом этапе является схема базы данных.

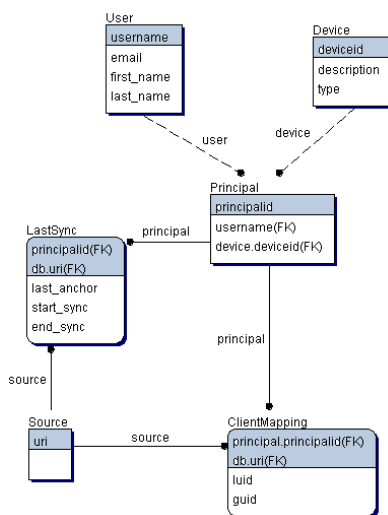


Рис. 1. Пример схемы базы данных

Контрольные вопросы

1. Какие серверные языки веб-программирования Вы знаете?
2. Какие СУБД Вы знаете?
3. Как на HTML создать объект «кнопка»?
4. Как на HTML создать объект «текстовое поле»?
5. Как на HTML создать объект «флажок»?
6. Как на HTML создать таблицу 3x3?
7. Что делает SQL-команда «SELECT»?
8. Что делает SQL-команда «INSERT»?
9. Что делает SQL-команда «DELETE»?
10. Что делает SQL-команда «UPDATE»?

Рекомендуемые источники

1. Колдаев В.Д. Основы алгоритмизации и программирования: Учебное пособие / В.Д. Колдаев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2017. - 416 с. [Электронный ресурс]. <http://znanium.com/bookread.php?book=336649>.
2. Гагарина Л.Г. Технология разработки программного обеспечения: Учеб. пос. / Л.Г.Гагарина, Е.В.Кокорева, Б.Д.Виснадул; Под ред. проф. Л.Г.Гагариной - М.: ИД ФОРУМ: НИЦ Инфра-М, 2015. - 400 с. [Электронный ресурс]. <http://znanium.com/bookread.php?book=389963>.
3. Голицына О. Л. Программирование на языках высокого уровня: Учебное пособие / О.Л. Голицына, И.И. Попов. - М.: Форум, 2016. - 496 с. [Электронный ресурс]. <http://znanium.com/bookread.php?book=139428>.