

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
Факультет компьютерных систем и сетей
Кафедра информатики

Курсовой проект по дисциплине:
«Программирование»

Пояснительная записка к курсовой работе

Тема работы:
«Система электронного заказа товаров»

Исполнитель
студент гр. 653501

(подпись)

Качановский Е.Д.

Руководитель

(подпись)

Козуб В.Н.

(оценка)

Минск
2017 год

Оглавление

Введение	3
1. Описание используемых технологий	5
1.1 Описание HTML.....	5
1.2 Описание CSS	6
1.3 JavaScript.....	7
1.3.1 Описание языка	7
1.3.2 Библиотека JQuery.....	8
1.3.3 Технология AJAX	8
1.4 Описание PHP.....	9
1.5 MySQL	10
1.5.1 Описание MySQL.....	10
1.5.2 PhpMyAdmin.....	10
2. Логика программы	11
2.1 Взаимодействие с пользователем	11
2.1.1 Взаимодействие с клиентом	11
2.1.2 Взаимодействие с продавцом.....	13
2.2 Структура хранимых данных	14
2.3 Структура программы	16
2.3.1 Папка Templates.....	16
2.3.2 Папка requires	17
2.3.3 Папка serverScripts.....	17
2.3.4 Папка src	18
2.3.5 Папка Styles	19
2.3.6 Основная папка.....	19
Заключение	20
Литература	21

Введение

Одно из проявлений закона подлости — блины пекутся дольше, чем поедаются.

Именно такой цитатой можно охарактеризовать ситуацию, сложившуюся в столовых БГУИРа. Блины готовятся долго, а желающих их попробовать совсем немало: 82% студентов БГУИРа (Рисунок 1), прошедших подготовленный мной опрос, покупают блины в столовых нашего учебного заведения.

Как часто вы едите блины, приготовленные в столовых БГУИР?

150 ответов

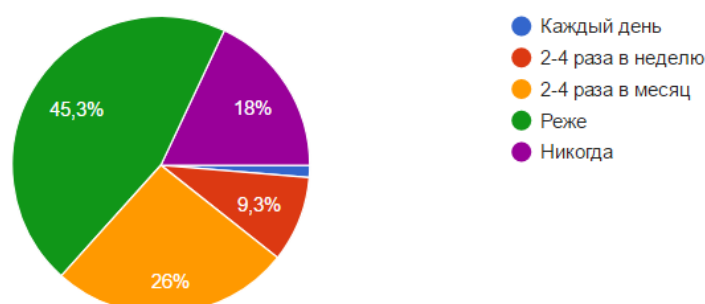


Рисунок 1

Предположим, из 150 опрошенных 123 человек (эти 82%) стоят в очередях за блинами. В среднем, у студентов БГУИРа 4 пары в день, следовательно, 3 перемены. Каждая перемена длится в среднем 15 минут. В очередях за блинами на каждой перемене, по моим подсчетам, стоит 10-15 человек, иногда даже больше. Проведя нехитрые математические вычисления, подсчитаем: каждому студенту, чтобы оплатить заказ, дождаться его выполнения и съесть блин необходима 1 минута. Но один блин готовится в течение 2-ух минут (без учета времени на оплату). В таком случае, за одну перемену столовая может обслужить только 8 студентов. Учтем, что каждому студенту также необходимо съесть блин (на что уходит приблизительно 5 минут) и не опоздать на пару, что сокращает вероятность обслуживания большего количества студентов. Значит, из 15 желающих попробовать блины на одной перемене только 8 смогут это сделать, что составляет всего 53%. Согласитесь, для лучшего IT-вуза страны – это неприемлемый показатель.

Также приведем некоторую статистику:

Нравится ли вам стоять в длинных очередях за блинами?

150 ответов

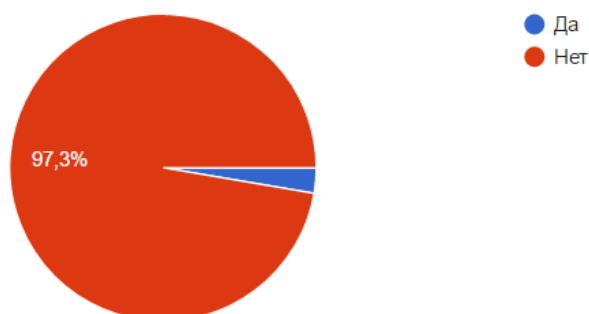


Рисунок 2

Считаете ли вы нужным создание возможности заказа блинов онлайн, что позволит не стоять в очередях?

150 ответов

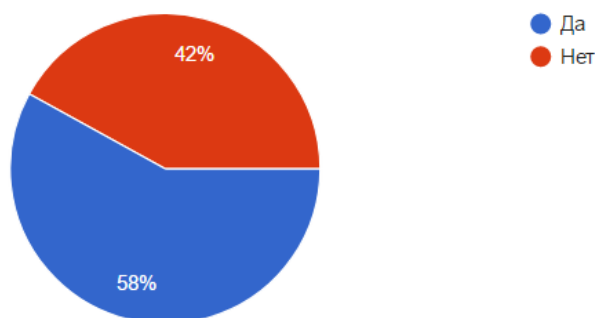


Рисунок 3

Отсюда видно (Рисунки 2, 3), что большинство студентов БГУИРа желают провести информатизацию столовых, а именно, введение системы электронного заказа товаров (блинов).

Цель: создать и разработать удобную систему электронного заказа товаров (блинов) в столовых БГУИРа.

Для достижения сформулированной цели я решил использовать ряд web-технологий: *html, css, javascript, php, MySQL*, а также смежных с ними *JQuery, Ajax, phpMyAdmin*. Мой выбор пал именно на web-технологии, так как практически все студенты пользуются интернетом и имеют к нему свободный доступ. На мой взгляд, это является самым подходящим, простым и лаконичным решением.

1. Описание используемых технологий

1.1 Описание HTML

HTML^[1]— стандартизированный язык разметки документов во Всемирной паутине. Большинство веб-страниц содержат описание разметки на языке HTML. Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства.

Язык HTML является приложением SGML (стандартного обобщённого языка разметки) и соответствует международному стандарту ISO 8879.

Во всемирной паутине HTML-страницы, как правило, передаются браузерам от сервера по протоколам HTTP или HTTPS, в виде простого текста или с использованием шифрования.

Язык HTML был разработан британским учёным Тимом Бернерсом-Ли приблизительно в 1986—1991 годах в стенах ЦЕРНа в Женеве в Швейцарии. HTML создавался как язык для обмена научной и технической документацией, пригодный для использования людьми, не являющимися специалистами в области вёрстки. HTML успешно справлялся с проблемой сложности SGML путём определения небольшого набора структурных и семантических элементов — дескрипторов. Дескрипторы также часто называют «тегами». С помощью HTML можно легко создать относительно простой, но красиво оформленный документ. Помимо упрощения структуры документа, в HTML внесена поддержка гипертекста. Мультимедийные возможности были добавлены позже.

Изначально язык HTML был задуман и создан как средство структурирования и форматирования документов без их привязки к средствам воспроизведения (отображения). В идеале, текст с разметкой HTML должен был без стилистических и структурных искажений воспроизводиться на оборудовании с различной технической оснащённостью (цветной экран современного компьютера, монохромный экран органайзера, ограниченный по размерам экран мобильного телефона или устройства и программы голосового воспроизведения текстов). Однако современное применение HTML очень далеко от его изначальной задачи. С течением времени основная идея платформонезависимости языка HTML была принесена в жертву современным потребностям в мультимедийном и графическом оформлении.

1.2 Описание CSS

CSS^[2] — формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Преимущественно используется как средство описания, оформления внешнего вида веб-страниц, написанных с помощью языков разметки HTML и XHTML, но может также применяться к любым XML-документам, например, к SVG или XUL.

CSS используется создателями веб-страниц для задания цветов, шрифтов, расположения отдельных блоков и других аспектов представления внешнего вида этих веб-страниц. Основной целью разработки CSS являлось разделение описания логической структуры веб-страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой веб-страницы (которое теперь производится с помощью формального языка CSS). Такое разделение может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом. Кроме того, CSS позволяет представлять один и тот же документ в различных стилях или методах вывода, таких как экранное представление, печатное представление, чтение голосом (специальным голосовым браузером или программой чтения с экрана), или при выводе устройствами, использующими шрифт Брайля.

До появления CSS оформление веб-страниц осуществлялось исключительно средствами HTML, непосредственно внутри содержимого документа. Однако с появлением CSS стало возможным принципиальное разделение содержания и представления документа. За счёт этого нововведения стало возможным лёгкое применение единого стиля оформления для массы схожих документов, а также быстрое изменение этого оформления.

1.3 JavaScript

1.3.1 Описание языка

JavaScript^[3] — прототипно-ориентированный сценарный язык программирования. Является реализацией языка ECMAScript (стандарт ECMA-262).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

На JavaScript оказали влияние многие языки, при разработке была цель сделать язык похожим на Java, но при этом лёгким для использования непрограммистами. Языком JavaScript не владеет какая-либо компания или организация, что отличает его от ряда языков программирования, используемых в веб-разработке.

JavaScript является объектно-ориентированным языком, но используемое в языке прототипирование обуславливает отличия в работе с объектами по сравнению с традиционными класс-ориентированными языками. Кроме того, JavaScript имеет ряд свойств, присущих функциональным языкам — функции как объекты первого класса, объекты как списки, карринг, анонимные функции, замыкания — что придаёт языку дополнительную гибкость.

JavaScript используется в клиентской части веб-приложений: клиент-серверных программ, в котором клиентом является браузер, а сервером — веб-сервер, имеющих распределённую между сервером и клиентом логику. Обмен информацией в веб-приложениях происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются кроссплатформенными сервисами.

На сегодняшний день поддержку JavaScript обеспечивают современные версии всех наиболее часто используемых браузеров. В Internet Explorer, Opera, Mozilla Firefox, Safari, Google Chrome имеется полная поддержка третьей редакции ECMA-262. При этом в Mozilla Firefox предпринята попытка осуществления поддержки четвёртой редакции спецификации, а первым браузером, в котором появилась неполная поддержка спецификации 3.1, явился Internet Explorer 8.

1.3.2 Библиотека JQuery

jQuery^[4] — библиотека JavaScript, фокусирующаяся на взаимодействии JavaScript и HTML. Библиотека jQuery помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими. Также библиотека jQuery предоставляет удобный API для работы с AJAX. Сейчас разработка jQuery ведётся командой jQuery во главе с Джоном Резигом.

Точно так же, как CSS отделяет визуализацию от структуры HTML, JQuery отделяет поведение от структуры HTML. Например, вместо прямого указания на обработчик события нажатия кнопки, управление передаётся JQuery, которая идентифицирует кнопки и затем преобразует его в обработчик события клика. Такое разделение поведения и структуры также называется принципом ненавязчивого JavaScript.

Библиотека jQuery содержит функциональность, полезную для максимально широкого круга задач. Тем не менее, разработчиками библиотеки не ставилась задача совмещения в jQuery функций, которые подошли бы всюду, поскольку это привело бы к большому коду, большая часть которого не востребована. Поэтому была реализована архитектура компактного универсального ядра библиотеки и плагинов. Это позволяет собрать для ресурса именно ту JavaScript-функциональность, которая на нём была бы востребована.

1.3.3 Технология AJAX

AJAX^[5] — подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером. В результате, при обновлении данных веб-страница не перезагружается полностью, и веб-приложения становятся быстрее и удобнее.

AJAX — не самостоятельная технология, а концепция использования нескольких смежных технологий.

Действия с интерфейсом преобразуются в операции с элементами DOM, с помощью которых обрабатываются данные, доступные пользователю, в результате чего представление их изменяется. Здесь же производится обработка перемещений и щелчков мышью, а также нажатий клавиш. Каскадные таблицы стилей, или CSS, обеспечивают согласованный внешний вид элементов приложения и упрощают обращение к DOM-объектам. Объект XMLHttpRequest (или подобные механизмы) используется для асинхронного взаимодействия с сервером, обработки запросов пользователя и загрузки в процессе работы необходимых данных.

В качестве формата передачи данных могут использоваться фрагменты простого текста, HTML-кода, JSON или XML.

1.4 Описание PHP

PHP^[6] — скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов.

В области веб-программирования, в частности серверной части, PHP — один из популярных сценарных языков (наряду с JSP, Perl и языками, используемыми в ASP.NET).

Популярность в области построения веб-сайтов определяется наличием большого набора встроенных средств для разработки веб-приложений. Основные из них:

- автоматическое извлечение POST и GET-параметров, а также переменных окружения веб-сервера в предопределённые массивы;
- взаимодействие с большим количеством различных систем управления базами данных;
- автоматизированная отправка HTTP-заголовков;
- работа с HTTP-авторизацией;
- работа с cookies и сессиями;
- работа с локальными и удалёнными файлами, сокетами;
- обработка файлов, загружаемых на сервер;

PHP отличается от JavaScript тем, что PHP-скрипты выполняются на сервере и генерируют HTML, который посылается клиенту^[7].

В настоящее время PHP используется сотнями тысяч разработчиков. Согласно рейтингу корпорации TIOBE, базирующемуся на данных поисковых систем, в мае 2016 года PHP находился на 6 месте среди языков программирования. К крупнейшим сайтам, использующим PHP, относятся Facebook, Wikipedia и др.

1.5 MySQL

1.5.1 Описание MySQL

MySQL^[8] – это одна из самых популярных и самых распространенных СУБД (система управления базами данных) в интернете. Она не предназначена для работы с большими объемами информации, но ее применение идеально для интернет сайтов, как небольших, так и достаточно крупных.

MySQL отличается хорошей скоростью работы, надежностью, гибкостью. Работа с ней, как правило, не вызывает больших трудностей. Поддержка сервера MySQL автоматически включается в поставку PHP.^[9]

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц.

1.5.2 PhpMyAdmin

phpMyAdmin^[10] — веб-приложение с открытым кодом, написанное на языке PHP и представляющее собой веб-интерфейс для администрирования и управления СУБД MySQL. PHPMyAdmin позволяет через браузер и не только осуществлять администрирование сервера MySQL, запускать команды SQL и просматривать содержимое таблиц и баз данных. Приложение пользуется большой популярностью у веб-разработчиков, так как позволяет управлять СУБД MySQL без непосредственного ввода SQL команд, предоставляя дружелюбный интерфейс.

PHPMyAdmin широко распространён по всему миру и является одним из лидеров в работе СУБД. Интерфейс доступен более чем на 62 языках мира. Благодаря простому и понятному интерфейсу данным приложением могут пользоваться даже начинающие пользователи.^[11]

2. Логика программы

2.1 Взаимодействие с пользователем

2.1.1 Взаимодействие с клиентом

Взаимодействие с клиентом осуществляется посредством web-браузера. Потенциальный покупатель открывает главную страницу сайта со своего устройства и видит дружелюбный и интуитивно понятный интерфейс (Рисунок 4).

Закажите блин в 2 простых шага:

1. Выберите блин:

Название	Цена
С ветчиной и сыром	2.72 BYN
С сыром и грибами	2.25 BYN
С ветчиной, сыром и грибами	3.13 BYN

Цена: 2.72 BYN

Ваш заказ:

Общая стоимость: 3.47 BYN

Введите ваш пароль:

2. Выберите начинки:

Название	Цена
Майонез	0.14 BYN
Кетчуп	0.11 BYN
Огурец	0.53 BYN
Сыр	1.13 BYN

Цена: 0.67 BYN

Рисунок 4

Как можно видеть на рисунке 4, процесс заказа блина является простым и незамысловатым. Тем не менее, опишем основные шаги осуществления заказа:

1. Клиент выбирает блин в соответствующей таблице. При выборе, строка с заказываемым блином становится зеленой.
2. Аналогично, клиент, по желанию, выбирает дополнительные начинки для блина. Единственным отличием от предыдущего пункта является возможность выбрать несколько опций сразу.
3. Клиент вводит пароль, с помощью которого он в последствии заберет заказанный блин.
4. Последнее, что надо сделать – нажать кнопку «Заказать».

Хотелось бы отметить, что, если пользователь не выберет блин либо длина его пароля будет менее 5 символов, на странице отобразятся соответствующие ошибки. Также в каждой таблице и в окне заказа выводятся стоимости заказа, что значительно упрощает использование программы.

После нажатия клиентом кнопки «Заказать», открывается страница, оповещающая об успешном выполнении заказа (Рисунок 5). На этой же странице выводится номер заказа клиента.




Рисунок 5

Далее, нажав на ссылку «На главную», пользователь вновь переходит на главную страницу, где, если необходимо, может выполнить еще один заказ.

2.1.2 Взаимодействие с продавцом

Взаимодействие с продавцом происходит посредством страницы продавца (Рисунок 6).

 Pancake Order					
№	Блин	Начинки	Цена	Готово	Забрать
3	С ветчиной и сыром	Майонез+Огурец	3.47	<input type="button" value="Готово"/>	<input type="button" value="Забрать"/>
6	С ветчиной, сыром и грибами	Майонез	3.35	<input type="button" value="Готово"/>	<input type="button" value="Забрать"/>
7	С ветчиной, сыром и грибами	Майонез+Кетчуп	3.46	<input type="button" value="Готово"/>	<input type="button" value="Забрать"/>
17	С ветчиной и сыром	Майонез+Кетчуп	3.05	<input type="button" value="Готово"/>	<input type="button" value="Забрать"/>
18	С ветчиной и сыром	Майонез+Огурец	3.47	<input type="button" value="Готово"/>	<input type="button" value="Забрать"/>
20	С ветчиной и сыром	Огурец+Сыр	4.46	<input type="button" value="Готово"/>	<input type="button" value="Забрать"/>
22	С ветчиной и сыром	Майонез+Огурец	3.47	<input type="button" value="Готово"/>	<input type="button" value="Забрать"/>

©2017-2017 Частное предприятие "Pancake Order"

Рисунок 6

На данной странице присутствует таблица, каждая строка которой содержит номер заказа, заказанный блин, дополнительные начинки к блину, цена заказа, кнопка «готово» и кнопка «забрать». Рассмотрим эти кнопки подробнее.

При нажатии на кнопку «готово» заказа помечается как готовый и фон соответствующей строки становится зеленым. При нажатии на кнопку «забрать», появляется модальное окно, с просьбой ввести пароль заказа (Рисунок 7).

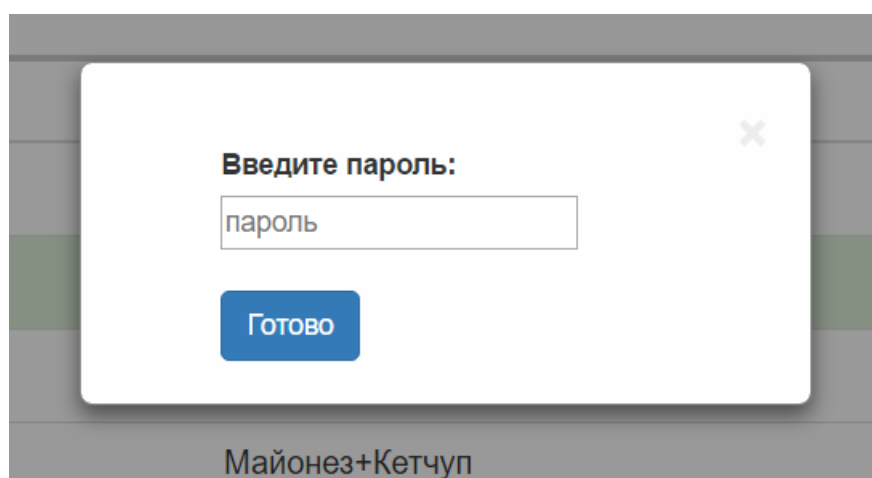


Рисунок 7

2.2 Структура хранимых данных

Для хранения данных о товарах и заказах в приложении используется СУБД MySQL, а также веб-приложение phpMyAdmin для удобного управления данными.

Рассмотрим базу данных создаваемого приложения (Рисунок 8).

Таблица	Действие	Строки	Тип	Сравнение	Размер	Фрагментировано
<input type="checkbox"/> fillings	Обзор Структура Поиск Вставить Очистить Удалить	4	InnoDB	utf8_general_ci	48 КБ	-
<input type="checkbox"/> orders	Обзор Структура Поиск Вставить Очистить Удалить	23	InnoDB	utf8_general_ci	16 КБ	-
<input type="checkbox"/> pancakes	Обзор Структура Поиск Вставить Очистить Удалить	3	InnoDB	utf8_general_ci	48 КБ	-
3 таблицы	Всего	30	InnoDB	utf8_general_ci	112 КБ	0 Байт

Рисунок 8

Как видно, база данных содержит всего 3 таблицы:

- Таблица pancakes – для хранения наименований и стоимостей блинов
- Таблица fillings – для хранения наименований и стоимостей начинок
- Таблица orders – для хранения сделанных пользователями заказов

В базе данных используется кодировка utf8_general_ci для корректного отображения и хранения кириллических символов.

Взглянем на структуру каждой из таблиц поближе.

Таблицы pancakes и fillings содержат идентичные поля (Рисунок 9).

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Дополнительно
<input type="checkbox"/> 1	id	int(12)		UNSIGNED	Нет	Нет	AUTO_INCREMENT
<input type="checkbox"/> 2	name	varchar(128)	utf8_general_ci		Нет	Нет	
<input type="checkbox"/> 3	price	float			Нет	Нет	

Рисунок 9

- Поле id – служит для хранения уникального идентификационного номера каждого из товаров. Также имеет свойство AUTO_INCREMENT, что при добавлении очередного товара автоматически увеличивает это поле на единицу по сравнению с предыдущей строкой.
- Поле name – служит для хранения наименования товара, блина либо начинки соответственно.
- Поле price – служит для хранения стоимости каждого товара.

Таблица orders содержит 8 полей, каждое из которых выполняет необходимую для приложения функцию. Рассмотрим их (Рисунок 10).

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Дополнительно
1	id	bigint(32)		UNSIGNED	Нет	Нет	AUTO_INCREMENT
2	pancakeName	varchar(128)	utf8_general_ci		Нет	Нет	
3	fillings	varchar(256)	utf8_general_ci		Нет	Нет	
4	time	timestamp		on update CURRENT_TIMESTAMP	Нет	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP
5	code	varchar(32)	utf8_general_ci		Нет	Нет	
6	done	tinyint(1)			Нет	Нет	
7	show	tinyint(1)			Нет	Нет	
8	totalPrice	float			Нет	Нет	

Рисунок 10

- Поле **id** – служит для хранения уникального идентификационного номера сделанных заказов. Имеет свойство **AUTO_INCREMENT**.
- Поле **pancakeName** – служит для хранения наименования заказанного блина.
- Поле **fillings** – служит для хранения всех заказанных с блином дополнительных начинок.
- Поле **time** – служит для хранения даты и времени оформления заказа. Имеет свойство **CURRENT_TIMESTAMP**, что позволяет добавлять дату и время автоматически.
- Поле **code** – служит для хранения пароля, введенного пользователем при заказе. Однако, хранится не сам пароль, а его хэш в формате md5.
- Поле **done** – служит идентификатором готовности блина. Может принимать одно из двух значений. 1 – заказ готов, 0 – заказ не готов.
- Поле **show** – Может принимать одно из двух значений. 1 – заказ не был забран клиентом, 0 – заказ был забран.
- Поле **totalPrice** – служит для хранения полной стоимости заказа.

2.3 Структура программы

Программа логично разбита на некоторое число папок и модулей, что значительно упрощает работу по ее изменению и расширению (Рисунок 11).

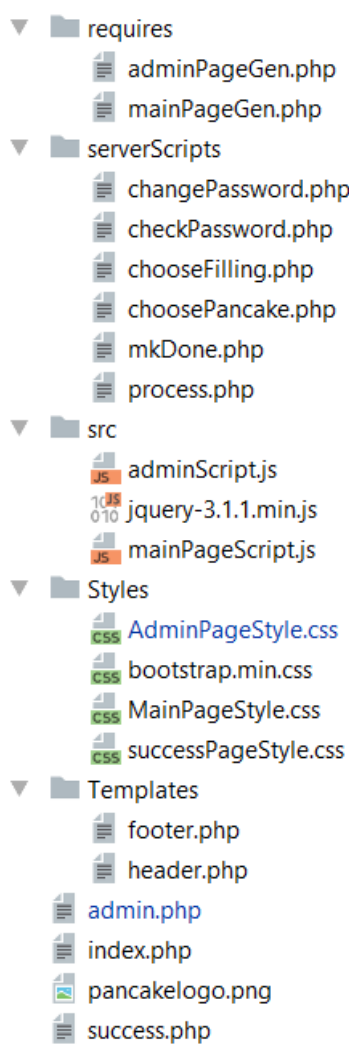


Рисунок 11

Далее подробнее рассмотрим все папки и модули, содержащиеся в них.

2.3.1 Папка *Templates*

В данной папке хранятся модули, отвечающие за вид footer'а и header'а (элементы, находящиеся в шапке и внизу страницы). Файлы названы соответствующим образом. Модули подключаются к каждой странице с помощью команд языка PHP и отображаются на всех страницах идентично.

Заметим, что templates с английского переводится как «шаблоны».

2.3.2 Папка *requires*

Данная папка содержит в себе 2 файла, которые служат для создания таблиц на страницах программы. Файл *adminPageGen.php* – для отображения таблицы заказов на странице продавца, файл *mainPageGen.php* – для отображения таблиц блинов и дополнительных начинок на странице клиента.

Данные модули получают доступ к базе данных приложения, обращаются к необходимым таблицам и выводят их в виде удобном для пользователя. Так, например, модуль *adminPageGen.php* получает данные об идентификационных номерах, наименованиях блинов и начинок, стоимости и готовности заказов, значение поля *show* которых равняется единице. Далее он отображает полученные данные в виде таблицы, изображенной на рисунке 6. Также если блин уже готов на момент срабатывания модуля, соответствующая строка таблицы будет выделена зеленым цветом.

2.3.3 Папка *serverScripts*

Данная папка содержит PHP модули, срабатывающие на сервере, при выполнении некоторого действия на страницах приложения. Рассмотрим каждый из них:

- Модуль *changePassword.php* – записывает полученный со страницы пароль в сессию.
- Модуль *choosePancake.php* – как параметр принимает *id* блина. Далее обращается к таблице *pancakes* базы данных и получает цену блина с соответствующим *id*. Записывает цену блина в сессию и обновляет общую стоимость заказа, которая также хранится в сессии. Закрывает базу данных и возвращает цену блина, его *id* и общую стоимость всего заказа в формате *json*.
- Модуль *chooseFilling.php* – как параметр принимает *id* начинки. Далее обращается к таблице *fillings* базы данных и получает цену начинки с соответствующим *id*. Проверяет была ли заказана начинка ранее. Если была, вычитает стоимость начинки из общей стоимости всех начинок и из стоимости заказа. Иначе – прибавляет. Закрывает базу данных и возвращает общую стоимость всех заказанных начинок, *id* только что выбранной начинки и общую стоимость всего заказа в формате *json*.
- Модуль *process.php* – отвечает за обработку всех данных введенных пользователем при заказе. Данный модуль делает новую запись в таблицу *orders* базы данных. Записывает наименование заказанного блина, наименования всех заказанных начинок в формате «начинка+начинка+...+начинка», хэш-код пароля в формате *md5*, общую стоимость заказа. Присваивает 0 полю *done* и 1 полю *show*. *Id* и время заказа записываются автоматически.

- Модуль *mkDone.php* – принимает id заказа как параметр. Обращается к таблице orders базы данных и изменяет поле done строки с соответствующим id на 1. Возвращает id.
- Модуль *checkPassword.php* – принимает id заказа и пароль как параметры. Сравнивает полученный пароль и пароль находящийся в таблице orders базы данных с соответствующим id. Если они равны присваивает значение 0 полю show и 1 полю done в строке с соответствующим id, возвращает true. Иначе – возвращает false.

2.3.4 Папка src

Данная папка содержит JavaScript модули для управления элементами страниц приложения и осуществления связи с модулями, лежащими в папке serverScripts. В папке находятся 2 модуля и файл jquery-3.1.1.min.js. Данный файл является библиотекой JQuery, которая была описана в первом разделе. В приложении библиотека JQuery используется в основном для более удобного обращения и использования технологии AJAX (Раздел 1, глава 1.3.3).

Опишем функциональность этих двух модулей:

- Модуль *mainPageScript.js* – подключается к главной странице приложения для управления ее элементами. Содержит следующие функции:
 - Функция *choosePancake()* – срабатывает при нажатии на строку с блином (Рисунок 4). С помощью AJAX вызывает модуль *choosePancake.php* и передает как параметр id нажатой строки таблицы. При успешном выполнении модуля выделяет нажатую строку зеленым цветом и обновляет цену блина и общую стоимость заказа на странице.
 - Функция *chooseFilling()* – срабатывает при нажатии на строку с начинкой (Рисунок 4). С помощью AJAX вызывает модуль *chooseFilling.php* и передает как параметр id нажатой строки таблицы. При успешном выполнении модуля выделяет либо снимает выделение с нажатой строки в зависимости от того была ли она выделена ранее. Обновляет общую стоимость заказанных начинок и стоимость заказа на странице.
 - Функция *passwordChanged()* – срабатывает при вводе символов в поле пароля (Рисунок 4). При каждом нажатии проверяет было ли введено 5 символов. Если было, с помощью AJAX вызывает модуль *changePassword.php* и передает введенную строку как параметр. Иначе выводит предупреждение об ошибке на страницу пользователя.
 - Функция *makeOrder()* – срабатывает при нажатии на кнопку «Заказать» на главной странице (Рисунок 4). Проверяет был ли выбран блин и введен ли пароль длиной не менее 5-ти символов.

Если хотя бы одно из условий не выполнено, возвращает соответствующее предупреждение об ошибке. Иначе вызывает модуль *process.php*. При успешном выполнении модуля, перенаправляет пользователь на страницу с номером его заказа (Рисунок 5).

- Модуль *adminScript.js* – подключается к странице продавца для управления ее элементами и осуществления связи с сервером. Содержит следующие функции:
 - Функция *mkDone()* – срабатывает при нажатии на кнопку «Готово» в одной из строк в таблице заказов (Рисунок 6). Получает id заказа и передает его как параметр в модуль *mkDone.php*. При успешном выполнении модуль выделяет данную строку зеленым цветом.
 - Функция *showModalPass()* – открывает модальное окно для ввода пароля (Рисунок 7). Срабатывает при нажатии на кнопку «Забрать».
 - Функция *hideModalPass()* – скрывает модальное окно при нажатии на крестик в правом верхнем углу самого окна.
 - Функция *mkAllReady()* – срабатывает при нажатии на кнопку «Готово» внутри модального окна (Рисунок 7). Вызывает модуль *checkPassword.php* и передает введенный пароль и id заказа как параметры. При успешном выполнении модуля проверяет полученные данные. Если значение равно false выводит сообщение о неверно введенном пароле. Иначе закрывает модальное окно и удаляет строку выбранного заказа, что оповещает об успешном выполнении заказа.

2.3.5 Папка Styles

Данная папка содержит файлы, отвечающие за отображение страниц на устройстве пользователя. Все файлы имеют расширение *.css*, описанное ранее в Разделе 1. Также папка содержит *bootstrap.min.css*. Это дополнительная библиотека Bootstrap добавлена в проект для управления уже готовыми стилями, часто используемыми в приложении.

2.3.6 Основная папка

В данной папке находятся основные файлы, к которым подключаются все остальные модули. Также данные файлы содержат разметку всех страниц приложения, описанную на языке HTML, также описанном в Разделе 1.

Заключение

В процессе написания курсовой работы была создана удобная и практичная система электронного заказа товаров (блинов) в столовых БГУИРа, а также изучены некоторые web-технологии, такие как HTML, CSS, PHP, JavaScript, AJAX, MySQL, PhpMyAdmin. Также вся программа удобно разбита на модули, которые логично объединены по папкам, что в будущем позволит с легкостью добавить новый функционал, если будет необходимо.

Данная система, при введении в производство, по подсчетам, сделанным во введении, увеличила бы количество обслуживаемых клиентов вдвое, соответственно увеличив прибыль столовых в пропорциональном размере.

Разработанная система хорошо отлажена и протестирована и уже готова быть внедрена в производство. Однако, несмотря на это, она все еще имеет ряд недостатков и недочетов, которые планируется исправить в будущих версиях.

К таким недостаткам, например, относится отсутствие возможности отслеживать готовности блина клиентом онлайн. Все еще можно подойти и спросить у продавца о готовности блина, однако с такой возможностью система выглядела бы намного более продуманной. Также в текущей версии отсутствуют такие возможности как заказ нескольких блинов сразу или отметка о просьбе разрезать блин пополам и т.д. Однако напомним, что, несмотря на такое количество недостатков (которые будут исправлены в будущей версии), система уже работает и готова ко внедрению в производство.

Также кратко опишу, что аналогов такой системы мной найдено не было (во всяком случае в столовых БГУИР таковых точно нет). Наиболее близкими являются системы заказа, работающие в заведениях KFC и Domino's pizza, однако они имеют ряд значительных отличий, что не позволяет ставить их в один ряд с разработанной в ходе написания курсовой работы программой. Данный факт позволяет называть разработанную систему абсолютно уникальной, что является несомненным плюсом.

Литература

1. <https://ru.wikipedia.org/wiki/HTML>
2. <https://ru.wikipedia.org/wiki/CSS>
3. <https://ru.wikipedia.org/wiki/JavaScript>
4. <https://ru.wikipedia.org/wiki/JQuery>
5. <https://ru.wikipedia.org/wiki/AJAX>
6. <https://ru.wikipedia.org/wiki/PHP>
7. <http://php.net/manual/ru/intro-what-is.php>
8. <https://ru.wikipedia.org/wiki/MySQL>
9. http://www.softtime.ru/bookphp/gl12_1.php
10. <https://ru.wikipedia.org/wiki/PhpMyAdmin>
11. <http://tradebenefit.ru/chto-takoe-phpmyadmin>