

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра математической экономики

**ПРОГРАММА ОБРАБОТКИ СТАТИСТИЧЕСКОЙ ИНФОРМАЦИИ В
СИСТЕМЕ УПРАВЛЕНИЯ ПРОЦЕССАМИ ТИПОГРАФИИ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 5 курса 561 группы

направления 09.03.03 Прикладная информатика

механико-математического факультета

Кузнецовой Людмилы Павловны

Научный руководитель:

доцент, к.э.н.

А. Р. Файзлиев

Заведующий кафедрой:

д.ф.-м.н., профессор

С.И. Дудов

Саратов 2018

Введение. В современном мире человек каждый день сталкивается с новыми данными, которые ему необходимо грамотно обработать, сохранить и при необходимости применить в нужное время. Чтобы сотрудники предприятий не держали все данные по работе лишь в памяти, чаще всего компании приобретают программное обеспечение для компьютеров, которое позволяет записывать информацию и долгое время хранить её до востребования. Это позволяет создавать для работников более комфортную среду, в которой им не нужно помнить все данные вплоть до цифр договоров. Кроме того, руководство производств, где внедрены автоматизированные информационные системы, может быть уверено в том, что если кто-то из персонала уволится или не сможет по каким-то причинам выйти на работу, процессы производства не будут остановлены. Нужные данные будут известны не только одному-двум людям в частности, а будут храниться в месте, доступ к которому должны иметь все имеющие на это права, работники.

Однако, на данный момент программ для хранения данных огромное множество, больший интерес сейчас вызывают те системы, в которых люди могут видеть статистику о работе производства. Это даёт компаниям ту самую необходимую информацию, которая позволяет настраивать свои процессы таким образом, чтобы прибыль приумножалась. Современный мир трактует бизнесу условия, при которых без непрерывного мониторинга невозможно стать достаточно конкурентно способным участником рынка. Именно поэтому каждое предприятие старается использовать помимо общепринятых и досупных средств автоматизации, свои собственные. Это позволяет компаниям направлять свою работу в то русло, которое им более выгодно и рентабельно. Для создания подобных программ бизнесмены нанимают группы людей, в которые входят программисты, бизнес – аналитики и тестировщики. Нынешние предприниматели понимают, какую пользу и выгоду для них может принести

команда квалифицированных ит-работников, которые смогут создать продукт, отвечающим всем нуждам бизнеса.

Целью выпускной работы является написание программы, с помощью которой пользователь сможет вносить необходимые данные и получать статистические сведения о работе предприятия, в нашем случае, типографии.

Соответствующими задачами являются:

- анализ проблемной области и имеющихся решений на основе литературных источников;
- разработка структуры программы;
- разработка алгоритма решения задачи;
- разработка информационной подсистемы планирования и управления;
- разработать схему базы данных;
- рассмотреть доступные способы сбора и отображения статистических данных;
- разработать руководство пользователя по программе;
- проверить программу на соответствии заявленным требованиям.

Предметом исследования является процесс работы на типографии.

Объектом исследования является существующая система по управлению производственными процессами.

Основное содержание работы. В первой части выпускной работы «Программа обработки статистической информации в системе управления процессами типографии» была описана цель создания подобной программы и её структура.

Любая фабрика по производству продукции должна быть в состоянии выполнять минимум необходимых для её работы функций:

- принять заказы от клиентов;
- занести данные заказа в свою систему;

- хранить данные о старых и новых заказах;
- иметь возможность дополнять список изготавливаемых продуктов/услуг;
- иметь возможность проанализировать свою работу, то есть предоставить доступ к статистике по выполняемым заказам.

Продуктивность выполняемой работы тесно связана с используемыми инструментами. Проводить статобработку данных вполне возможно, имея только лишь карандаш и бумагу, но намного быстрее и эффективнее делать это при помощи специальных инструментов.

Несмотря на имеющийся выбор подобных приложений, рациональней всего использовать собственный продукт. Это сэкономит деньги на закупках инструментов и их обновлений, а кроме того, можно создать удобный для работников интерфейс и только необходимый для производства набор функций.

Так как в данной работе рассматривается создание программы для производства, то лучше всего сразу рассчитывать на то, что это будет большой проект, который в дальнейшем нужно будет поддерживать и дополнять новыми идеями от руководства. Чаще всего подобные системы с бизнес-процессами разрабатываются при помощи паттерна проектирования – MVC. Этот паттерн позволяет минимизировать исправления кода и его обслуживание.

Model-View-Controller (MVC, «Модель-Представление-Контроллер», «Модель-Вид-Контроллер») – схема разделения данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо соответствии с рисунком 1

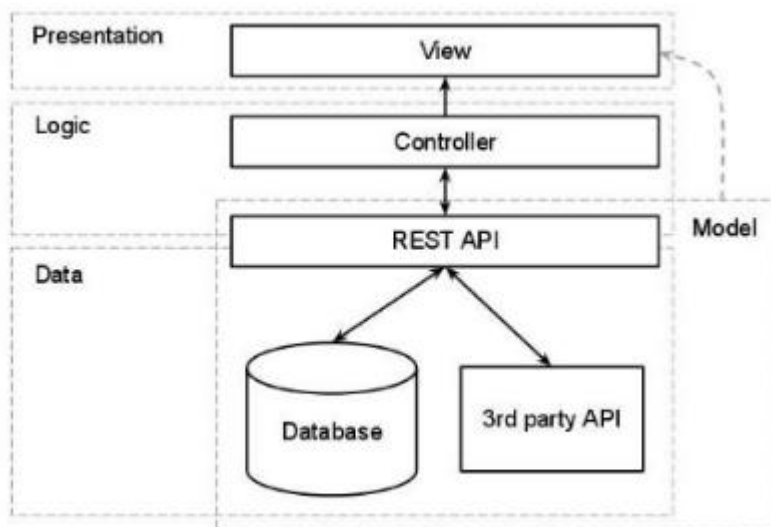


Рисунок 1– Шаблон проектирования «MVC»

Шаблон проектирования приложения MVC подразумевает разделение приложения на три глобальных компонента — Модель, Вид и Контроллер.

Каждый из этих компонентов решает свою определенную задачу:

- модель работает непосредственно с базой данных;
- вид отвечает за отображение данных пользователю на экране;
- контроллер получает запросы от пользователя на различные действия (actions) и вызывает для каждого действия свою модель (или определенный метод в модели).

MVC является основой для многих современных PHP фреймворков.

Лучше всего для системы обработки процессов производства подойдет веб-приложение, доступ к которому может получить любой участник производства с удалённого компьютера, подключенного к интернету. Поэтому для написания такой программы был выбран язык программирования – php.

PHP - 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.

Для хранения данных и работы с ними понадобится база данных, а для отображения – гипертекстовый язык разметки, а также дополнительные библиотеки для передачи данных в базу и выгрузку информации на страницы приложения. Таким образом, стек технологий будет состоять из следующих компонентов:

- php
- mysql
- html, css
- bootstrap
- jQuery

Во второй части работы была описана реализация проекта и приведены части кода, который использовался в том или ином месте. Был реализован функционал по добавлению продуктов, заказов и преобразования этих данных в визуальные.

В ходе работы было создано веб-приложение, которое может авторизовать пользователя, если он содержится в базе, а также производить редирект на форму авторизации, если неавторизованный работник пытается пройти по страницам, доступ к которым должен быть закрыт. Также, если авторизация уже была произведена, то это проверялось с помощью сессии, чтобы пользователю не нужно это было делать каждый раз при переходе по страницам.

Для этого необходимо запустить сессию и на каждой из страниц проверять заполненность в ней поля “user” подобным образом:

```
if(!$_SESSION['user']) {  
    header("Location: /");  
}
```

В зависимости от того вернётся “true” или “false” происходит редирект на ту или иную страницу.

При переходе на главную страницу кабинета пользователь должен будет видеть минимальную статистику по данным производства, а то есть количество заказов, продуктов и клиентов типографии. А помимо этого график с данными по месяцам.

Для начала необходимо реализовать вывод необходимых данных. Для этого после создания шаблона страницы нужно создать контроллер и модель главной страницы кабинета. В контроллере объявляется приватное свойство — шаблон для отрисовки, а также метод по умолчанию (`index`), который будет отрисовывать страницу с некоторыми данными, которые хранятся в массиве `pageData`. Первым этапом нужно вывести небольшую общую статистику на главную страницу — количество заказов, товаров и заказчиков.

Так как производство всегда должно производить только актуальную продукцию и быть в курсе того, что в данный момент интересно большому числу потребителей, то вид производимого продукта может быть часто заменён более новым или же дополнен тем, что пользуется спросом, например в какие-то праздники или сезоны. Для того, чтобы менеджеры могли занести заказ и указать любой из доступных для производства продукт, необходимо добавить в программу возможность работы с продуктами. А то есть страницу на которой продукты можно будет добавить в систему.

Когда в распоряжении менеджера есть набор продуктов, он может приступить к тому, чтобы заполнить данные заказа, который был сделан для типографии. Для это нужно составить план шагов по добавлению данных в базу.

1. пользователь должен иметь доступ к странице с данными о старых заказах;
2. на этой странице должна быть кнопка или ссылка для создания нового заказа;

3. при нажатии на неё нужно отобразить форму с возможностью заполнить данные заказчика:
4. так как в одном заказе может быть много продуктов, необходима кнопка или ссылка, которая будет отображать форму для заполнения данных по одному из них, а при нажатии на неё повторно, выдавать снова форму для заполнения;
5. после внесения данных необходима кнопка для отправки их на сервер, а также после обработки запроса, пользователь должен видеть актуальное состояние заказов.

Для того, чтобы визуально отобразить данные о работе нашего предприятия, понадобится добавить в программу обработку информации подобным образом.

В качестве данных для вывода статистики будут использованы месяцы года, за которые были сделаны заказы и их количество. Для этого в подключаемом javascript файле “chartStatistic.js” с помощью ajax будет осуществляться вызов функции getStatistic, которая в свою очередь обращается к базе с вызовом getOrderSplitMonth метода. Вид графика для главной страницы соответствует рисунку 2.

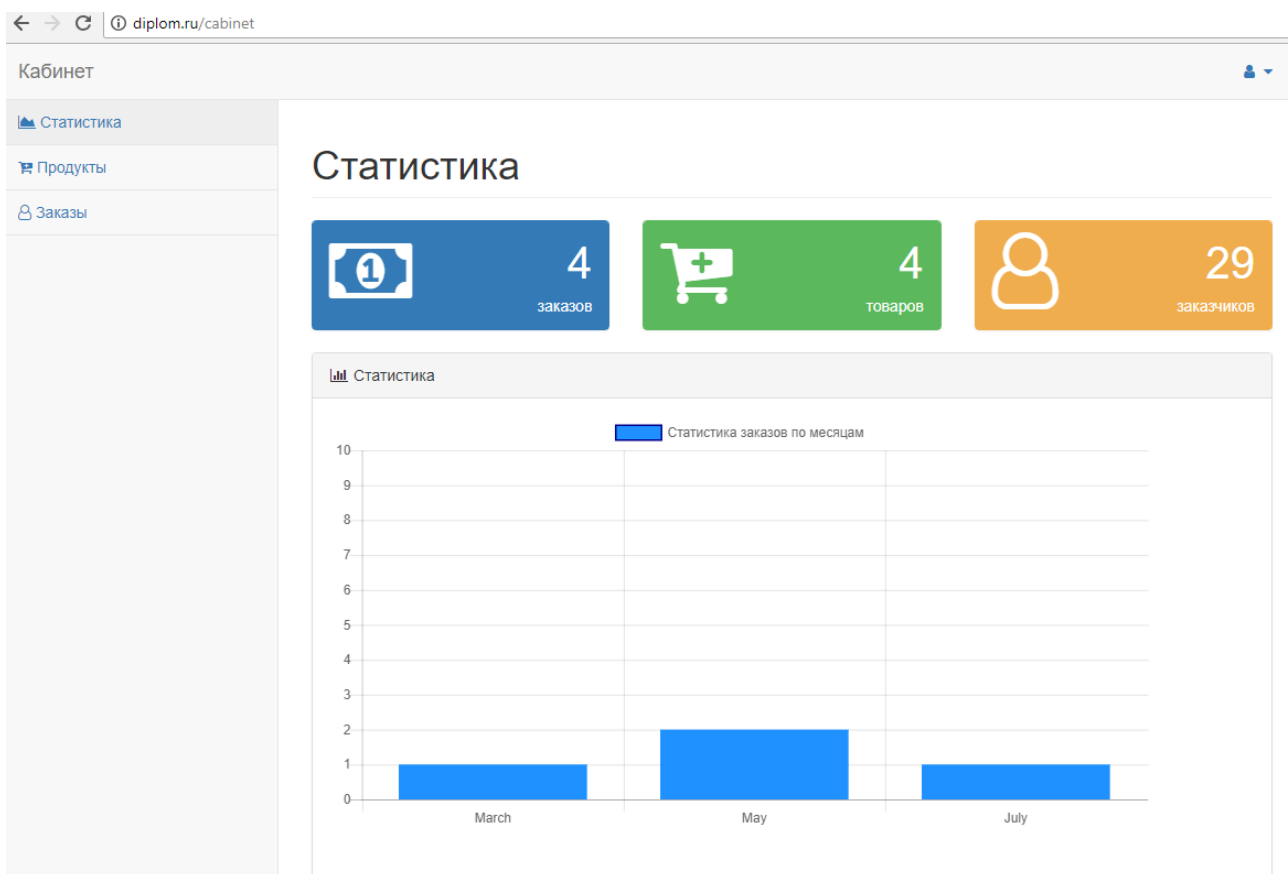


Рисунок 2 – Главная страница приложения

В третьем разделе вся добавленная в программу функциональность была протестирована на соответствие её первоначальным заданным функциям по отношению к тем, которые были реализованы к готовом программном обеспечении.

В данном случае техническим заданием, можно считать ранее описанные в предыдущих разделах функции приложения.

Тестирование программного обеспечения можно разделить на два вида: ручное и автоматизированное. Так как данное приложение не требует большого количества тестов и покрытия сценариев его работы, то можно ограничиться ручным.

Для начала тестирования необходимо составить тест план.

Удобно делить прецеденты на логические группы, подобные этим:

1. Функция авторизации пользователя(вход и выход из системы)
2. Возможность использовать навигацию в приложении
3. Функция по добавлению продуктов
4. Функция по добавлению заказов

Этот минимум – это план для приёмочных тестов данного приложения. Так как здесь перечислены его основные функции, которые обязательно должны быть доступны при передаче программного продукта заказчику.

В четвёртом разделе «Программа обработки статистической информации в системе управления процессами типографии» была рассчитана экономическая эффективность созданного приложения.

Создание и заказ программного продукта для предприятия – это впервые очередь – оптимизация процессов работы и получение прибыли. В данном случае прибыль будет получаться в результате того, что никакие данные не будут потеряны, а менеджеры смогут с выгодой использовать данные для того, чтобы можно было направлять производство в том направлении, которое принесёт наибольшую выгоду.

Срок окупаемости:

$$T(\text{ок}) = C(p) / \text{Э}(\text{ф})$$

где $C(p)$ – цена разработки;

$\text{Э}(\text{ф})$ – экономический эффект от внедрения;

Экономический эффект определим как экономию затрат.

Так как, на покупку программного обеспечения, которое не создано непосредственно под нужды фабрики и имеет множество ненужных, но входящих в стоимость, функций, не будет потрачен бюджет, то это одна из частей экономии. Ориентировочная стоимость подобного приложения с распространением лицензии на несколько рабочих компьютеров составляет 170 400.

Кроме того, данное программное обеспечение потребует обучения персонала работе с ним, что повлечёт за собой потери в рабочем времени. Если предположить, что обучение займёт около 3 рабочих часов в рамках представления продукта и также около 2 дней в момент эксплуатации, то можно отнести 2 рабочих дня менеджера в потери. Если взять среднюю заработную плату менеджера по процессам производства в 3000 рублей за 22 рабочих дня, то потеря составит около 3000 рублей.

Итого экономический эффект от внедрения равен $170400 + 3000 = 173400$ рублей.

Таким образом срок окупаемости = $104000 / 173400 = 0,6$ (лет) = 7 месяцев.

Это довольно большой показатель окупаемости для такого масштабного производства, как типография. Кроме того, стоит упомянуть, что разработанное приложение, можно усовершенствовать и настроить под новые или дополнительные нужды производства. Можно заказать разработчику новые реализации или же заключить договор на поддержку программного продукта по доступной цене и быть уверенным, что в экстренных случаях не будет простоев связанных с несвоевременным ответом технической поддержки.

Заключение. В данном дипломном проекте была сформулирована цель использования программного обеспечения, созданного непосредственно для нужд типографии в соответствии с заданным техническим заданием. Это были такие основные функции, как работа с продуктами и заказами, а также вывод статистики по ним. Для достижения данной цели были решены следующие задачи: проанализирована актуальность данного приложения и предприятие с точки зрения имеющихся необходимых для него процессов.

Чтобы реализовать проект по контролю заказов производства, была создана веб-система с удобным и интуитивно понятным интерфейсом. В ней менеджер

может посмотреть существующие заказы, продукты, статистику по ним и создать новые при необходимости.

В ходе написания проекта были перечислены все задействованные технологии и произведено тестирование готового программного обеспечения с отображением результатов.

Проделанная работа показала преимущества создания продукта под нужды заказчика и возможности разнообразной реализации поставленной задачи.