**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«Рязанский государственный радиотехнический университет»**

**(ФГБОУ ВО «РГРТУ», РГРТУ)**

«Утверждаю»

Заведующий кафедрой

ВПМ Пылькин А.Н.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  *(подпись)*

«\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2017 г.

ОТЧЕТ

О ПРЕДДИПЛОМНОЙ ПРАКТИКЕ СТУДЕНТА

**4 курса, 343 группы**

*(Степуро Е.Н.)*

Рязань 2017

Направление подготовки: 09.03.04 «Программная инженерия»

*(шифр и наименование)*

Тема ВКР: «Разработка клиентского приложения для системы контроля качества обучения в IT-компании»

|  |  |  |
| --- | --- | --- |
| № | Результаты научной работы студента | Результаты |
| 1 | Рассмотрение сферы исследования и обоснование её актуальности | Составлен отчет об актуальности системы контроля качества обучения |
| 2 | Обоснование выбора темы | Составлен отчет о выборе темы |
| 3 | Изучение и анализ | Проведён анализ LMS (Learning Management System) – систем управления обучением |
| 4 | Отчет о полученных результатах | Составлен отчет по результатам работы |

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(Степуро Е.Н.)

*(Ф.И.О. студента, подпись)*

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2017 г.

Научный руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(Каширин И.Ю.)

*(Ф.И.О., подпись)*

*Студент четвертого курса Степуро Е.Н.*

*Направление 09.03.04. «Программная инженерия»*

**Отчет об актуальности сферы исследования**

В настоящее время существует огромное множество различных обучающих программ. Все они имеют разное назначение, но есть то, что объединяет их все – наличие данных, которые необходимо как-то хранить и в последствие использовать. В связи с этим в процесс обучения все более активно внедряются программные технологии на базе персональных ЭВМ.

Целью данной выпускной квалификационной работы является разработка клиентской части системы контроля качества обучения в *IT*-компании. Программа должна предоставить пользователю всю необходимую информацию, а именно:

1. направления подготовки, которые проводит компания;
2. персоны, которые проходят собеседование и становятся кандидатами на какие-либо курсы;
3. кандидаты, которые были рассмотрены на конкурсной основе для поступления на курсы;
4. реквизиции – наборы студентов в группы;
5. персоны, которые в последствие рассматриваются в качестве кандидатов на курсы;
6. *rdm – Resource Development Manager –* сотрудники, управляющие направлениями подготовки;
7. анализ успеваемости и посещаемости студентов, на примере одного из направлений подготовки.

Большинство компаний при разработке любой системы делает акцент на серверную часть, приковывая все внимание к наличию данных, которые должны реализовывать идею проекта, а также безопасности этих данных и построению бизнес логики. Недостатком такого подхода является отсутствие полноценной *front-end* части, которая отвечает за вывод данных и представление их пользователю. В таком подходе *front*-*end* разработчик выступает, как человек, который просто делает сайт привлекательным. В каком-то смысле, так и есть, но в целом, это стереотип. Пользователи, кроме содержания, хотят видеть приятный, хорошо работающий сайт с интуитивно понятным интерфейсом. Продуманный дизайн производит впечатление профессионального сайта. К тому же, человек, работающий с серверной частью, затрачивает больше времени на написание всей программы. Гораздо экономичнее и быстрее будет найти человека, который займется созданием «внешнего вида» сайта. Впечатление и удобство системы, а также время разработки имеют экономическую ценность для заказчика. Именно поэтому данная выпускная квалификационная работа имеет большую значимость.

Данная выпускная квалификационная работа подразумевает разработку клиентской части веб-приложения. Но существует БД, из которой берутся данные и подаются на сервер. Эти данные и отображаются на разрабатываемой в данной работе клиентской части приложения.

Так как разрабатывается именно клиентская часть веб-приложения, то интерфейс в данной работе играет наиболее важную роль. Интерфейс должен быть интуитивно-понятным пользователю, приятным на вид и выводящим конкретные, нужные данные. Яркость интерфейса данного проекта делает дизайн сайта не только не скучным, но и соответствующим современным тенденциям. Интерфейс не загроможден огромным количеством различных ссылок, по которым необходимо переходить, чтобы найти нужную информацию. На каждой отдельной странице располагается информация, соответствующая одной или нескольким таблицам базы данных.

Данное приложение имеет гибкую структуру, что также можно отнести к практической ценности проекта.

В итоге, в рассматриваемой выпускной квалификационной работе разрабатывается приложение с данными результатами:

1. интерфейс, соответствующий всем стандартам разработки, но при этом обладающий уникальностью;
2. клиентская часть, которая напрямую не зависит от базы данных, а только берет необходимые данные с сервера;
3. представление необходимых пользователю данных для работы в системе контроля качества обучения;
4. авторизация в системе под своим уникальным логином и паролем.

*Студент четвертого курса Степуро Е.Н.*

*Направление 09.03.04. «Программная инженерия»*

**Обоснование выбора темы**

При проведении каких-либо курсов важно понимать, как лучше построить учебную программу. Это является одним из решающих факторов успешного освоения материала. Правильная последовательность тем позволяется качественнее понять суть представленного материала и запомнить его. Анализируя предоставленное на каждую из тем время и сопоставляя его с общим доступным на курсы, можно корректно построить учебный план.

Рассмотрим основные причины, по которым студенты могут плохо понять программу курсов, и которые реализует представленное приложение.

*Выбор способа разделения и предоставления информации ресурсному менеджеру*. Правильное разделение актуальной и необходимой информации является важным фактором быстрого построения учебного плана и разграничения доступного функционала. Также это помогает избежать ошибок среди данных, обусловленных человеческим фактором.

*Отсутствие ссылочной целостности*. К этому недостатку можно отнести *unfriendly user-interface* («недружелюбный пользовательский интерфейс») и долгий переход между страницами приложения. В случае удобного доступа к ссылкам на другие вкладки происходит экономия временных затрат и отсутствие нагромождения данных.

*Корректное использование предоставляемой информации*. Правильно используемые данные – залог успешной работы. Вся получаемая информация должна быть верно проанализирована. Такие данные обладают большей информативностью и ценностью.

*Сильная зависимость клиентской части от базы данных*. Данный недостаток позволяет развернуть приложение целиком только на одном компьютере. В случае переполнения памяти подтягивание данных из базы и отображение их на клиенте значительно увеличивает временные затраты на загрузку и корректное отображение данных, а также на их анализ.

*Анализ данных в режиме реального времени*. Тщательный анализ предоставляемой информации не имеет особой ценности, если такая информация не актуальна. Своевременный обзор материала несет большое значение для любой системы.

Разрабатываемая программа является клиентской частью веб-приложения. Поэтому основной выбор программных средств направлен на языки программирования и утилиты, предназначенные для верстки и прорисовки пользовательского интерфейса, а также для обработки данных на клиенте.

Средой программирования был выбран *Eclipse IDE for Java EE Developers* версии *Neon* 1.0. Данная среда является одной из последних версий *Eclipse*, разработанных компанией *ORACLE*. Для работы на серверном уровне был выбран контейнер сервлетов *Apache Tomcat v7.0*. Веб-сервер *Apache* выбран в проектируемом ВКР из-за его гибкости, мощности и широкой распространенности. Он может быть расширен с помощью системы динамически загружаемых модулей и исполнять программы на большом количестве интерпретируемых языков программирования без использования внешнего программного обеспечения. При разработке рассматриваемого приложения основной акцент был обращен на прорисовку пользовательского интерфейса.

В качестве языка разметки был выбран *HTML*, а для его отображения – язык стилей *CSS*. *HTML* – это самый распространенный инструмент создания сайтов. Он был выбран по следующим причинам:

* оптимально поддерживается и интерпретируется интернет-проводниками;
* читается на различных платформах;
* считается открытым стандартом.

Использование *CSS* обусловлено тем, что стили являются удобным и практичным инструментом при верстке веб-страниц и оформления различных элементов.

*JQuery* представляет собой кроссплатформенную библиотеку. В представленной ВКР она использует для связи *JavaScript* и *HTML*. При помощи *JQuery* код пишется намного легче и быстрее, чем просто на *JavaScript*. Это эффективно с точки зрения экономии времени. К тому же, проект, разработанный в *JQuery*, будет работать в любых современных браузерах.

*Студент четвертого курса Степуро Е.Н.*

*Направление 09.03.04. «Программная инженерия»*

**Анализ современных систем управления обучением**

*LMS* (*Learning Management System*) – это  программа или веб-технология, при помощи которой можно хранить, создавать и распространять учебные материалы, отслеживать успеваемость учащихся, проводить оценивание и администрировать процесс обучения. Создаются данные материалы в визуальной учебной среде с заданием последовательности изучения. В состав системы входят различного рода индивидуальные задания, проекты для работы в малых группах и учебные элементы для всех студентов, основанные как на содержательном компоненте, так и на коммуникативном. Рассмотрим их поподробнее.

*Blackboard* – одна из наиболее популярных LMS в мире, особенно в системе высшего образования. Данная платформа является платной. Здесь есть множество возможностей, и постоянно появляются новые функции, такие как возможность выставлять оценки анонимно, или делегировать выставление оценок ассистенту, возможность аудио или видео в качестве обратной связи и продвинутая аналитика. Blackboard стоит использовать, если действительно нужна продвинутая платформа, и если есть готовность инвестировать средства в покупку лицензии и обучение преподавателей.

*Moodle* – еще одна популярная система управления обучением. Ее использование подразумевает определенные расходы. Скачать набор файлов с программным кодом с официального сайта можно бесплатно, но необходимо будет оплатить хостинг и услуги специалиста, который будет устанавливать, обновлять и обслуживать платформу. *Moodle* предлагает огромное количество возможностей для созданий и хранения материалов, оценивания учащихся и взаимодействия между учащимися и преподавателем. Система очень гибкая, есть огромной количество настроек, которые позволяют адаптировать систему под потребности пользователя. Так как код открытый, то можно вносить в него любые изменения.

В 2015 году появился бесплатный хостинг для *Moodle*-сайтов *MoodleCloud*, который является хорошим решением для репетиторов и небольших образовательных организаций, которые не могут обеспечить необходимую техподдержку или не могут позволить оплату хостинга, а также для тех, кто хочет ознакомиться с основными функциями платформы. Достоинствами *Moodle* являются:

* подробная документация;
* наличие форумов и сообществ пользователей, которые могут помочь разобраться с особенностями работы платформы.

*Edmodo* часто характеризуется, как *Facebook* для школ. При работе с этой системой учителя и учащиеся создают бесплатные учетные записи, а затем учитель создает свой класс в виде группы. У одного преподавателя может быть несколько групп, и каждый учащийся может состоять в нескольких группах. Для регистрации учащихся электронная почта необязательна. Когда учащиеся входят в *Edmodo*, то видят ленту новостей — все объявления и записи  от групп, в которых они состоят. Преподаватели могут хранить в системе  учебные материалы, доступные для скачивания учащимися, принимать и оценивать задания, выдавать значки учащимся за успехи. *Edmodo* — удобная платформа для общения и взаимодействия. Платформа полностью бесплатная и не содержит рекламы.

*Google Класс* становится все более популярной *LMS*. Чтобы воспользоваться платформой, необходимо быть пользователем *Google* *Apps* для образования. *Google* *Класс* позволяет создавать учебные классы, создавать и оценивать задания, предоставлять учащимся обратную связь, публиковать объявления и распространять учебные материалы. У данной системы нет некоторых функций, которые можно найти у других популярных *LMS*. Например, возможности совместного обсуждения ограничены. Так как платформа интегрирована с другими инструментами *Google*, ее будет легко начать использовать тем, кто уже знаком с различными приложениями *Google*. Класс бесплатен для учебных заведений.

*Schoology* обладает всеми основными функциями *LMS*. Основная отличительная особенность платформы  — это большие  возможности для взаимодействия и совместной работы: учащихся между собой, преподавателей между собой, преподавателя индивидуально с  учащимся, в небольших группах, и так далее. Пользователь *Schoology* видит ленту новостей с записями всех групп, в которых он состоит. Он также может перейти на страницы этих групп, чтобы видеть только их сообщения. На странице курса  учителя могут публиковать задания, собирать работы и оценивать их, отмечать в календаре крайние сроки сдачи заданий, вести дискуссии, проводить тестирование, хранить материалы курса, которые учащиеся могут скачивать. Есть функция создания групп, в которых учителя могут объединяться как с учащимися, так и с другими учителями, в том числе и из других школ.  Учащиеся могут создавать собственные группы, при этом  школа полностью контролирует, как они функционируют. *Schoology* интегрирована с различными приложениями, такими как  *Vimeo*, *eduCanon*, *Google* *Drive*  и *Evernote*.