

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

РАБОТА ПРОВЕРЕНА

Рецензент

_____ З.А. Фельк

«___»_____ 2019 г.

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой ЭВМ

_____ Г.И. Радченко

«___»_____ 2019 г.

РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ ПОДДЕРЖКИ
БИЗНЕС-ДЕЯТЕЛЬНОСТИ УЧЕБНОГО ЦЕНТРА

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУрГУ-09.03.01.2019.114/167.ВКР

Руководитель работы,
к.т.н., доцент каф. ЭВМ

_____ И.Л. Надточий

«___»_____ 2019 г.

Автор работы,
студент группы КЭ-452

_____ В.Д. Белокопный

«___»_____ 2019 г.

Нормоконтролёр,
ст. преп. каф. ЭВМ

_____ С.В. Сяськов

«___»_____ 2019 г.

Челябинск-2019

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ
Заведующий кафедрой ЭВМ
_____ Г.И. Радченко
«___» _____ 2019 г.

ЗАДАНИЕ

на выпускную квалификационную работу бакалавра
студенту группы КЭ-452
Белоконный Владислав Дмитриевич
обучающемуся по направлению
09.03.01 «Информатика и вычислительная техника»

- 1. Тема работы:** «Разработка программного комплекса для поддержки бизнес-деятельности учебного центра» утверждена приказом по университету от 25 апреля 2019 г. № 899
- 2. Срок сдачи студентом законченной работы:** 19 июня 2019 г.
- 3. Исходные данные к работе:**
 - 3.1. Веллинг Л., Томсон Е. PHP и MySQL Веб-разработка. – М.: Вильямс, 2010. – 848 с.
 - 3.2. Michele E. Davis, Jon A Phillips. Learning PHP and MySQL. – USA: O'reilly, 2008. – 448 с.
 - 3.3. Руководство по языку программирования PHP.
[Электронный ресурс] URL: <http://www.php.net>
- 4. Перечень подлежащих разработке вопросов:**

- 4.1. Ознакомиться с регламентом работы Автономной некоммерческой организации дополнительного профессионального образования «Сибирский учебный центр СПАС».
- 4.2. Выполнить обзор существующих решений по автоматизации деятельности учебных центров, занимающихся дополнительным профессиональным образованием.
- 4.3. Спроектировать и реализовать систему автоматизации деятельности центра дополнительного профессионального образования «Сибирский учебный центр СПАС».
- 4.4. Протестировать работу системы.
5. **Дата выдачи задания:** 1 декабря 2018 г.

Руководитель работы _____ /И.Л. Надточий/

Студент _____ /В.Д. Белоконный /

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	01.03.2019	
Разработка модели, проектирование	01.04.2019	
Реализация системы	01.05.2019	
Тестирование, отладка, эксперименты	15.05.2019	
Компоновка текста работы и сдача на нормоконтроль	24.05.2019	
Подготовка презентации и доклада	30.05.2019	

Руководитель работы _____ /И.Л. Надточий/

Студент _____ /В.Д. Белоконный/

АННОТАЦИЯ

В.Д. Белоконный. Разработка программного комплекса для поддержки бизнес-деятельности учебного центра. – Челябинск: ФГАОУ ВО «ЮУрГУ (НИУ)», ВШЭКН; 2019, 87 с., 31 ил., 6 таб. библиогр. список – 16 наим.

В рамках представленной квалификационной работы производилось проектирование и разработка программного комплекса для поддержки бизнес-процессов учебного центра. На этапе проектирования производились оценка аналогов проектируемого решения, выделение и изучение бизнес-процессов предприятия, разработка проекта базы данных для хранения информации, анализ и разработка процессов взаимодействия пользователя с приложением, подготовка визуальных макетов. На этапе разработки рассматривались и анализировались доступные технологии для реализации проекта. Непосредственно реализуется разработка программного комплекса с использованием выбранных технологий. Доказывается способность реализованного решения к обеспечению успешной реализации задач, поставленных перед программным комплексом.

В ходе выполнения выпускной квалификационной работы разработана система со следующим функционалом:

- 1 Автоматизация процесса обучения;
- 2 Управление информацией о клиентах;
- 3 Электронный документооборот;
- 4 Менеджер задач.

Используемая инструментальная база:

- среды разработки: NetBeans 8.2;
- языки разработки: Php, CSS;
- веб-сервер: Apache;
- СУБД: MySQL;

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	7
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	11
1.1 ОБЗОР АНАЛОГОВ.....	11
1.2 АНАЛИЗ ОСНОВНЫХ ТЕХНОЛОГИЧЕСКИХ РЕШЕНИЙ.....	12
1.3 ХАРАКТЕРИСТИКА ОБЪЕКТА АВТОМАТИЗАЦИИ	18
1.3.1 Задачи учебного центра.....	18
1.3.2 Задействованные работники и их роли	19
1.3.3 Документы учебного центра	20
1.3.4 Описание учебного процесса	22
2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ.....	28
2.1 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ	28
2.2 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ	31
3 ПРОЕКТИРОВАНИЕ	33
3.1 АРХИТЕКТУРА ПРЕДЛАГАЕМОГО РЕШЕНИЯ	33
3.2 КОМПОНЕНТЫ СИСТЕМЫ.....	34
3.3 ОПИСАНИЕ ДАННЫХ	36
4 РЕАЛИЗАЦИЯ ПРОГРАММНОГО КОМПЛЕКСА.....	38
4.1 РЕАЛИЗАЦИЯ ИНТЕРФЕЙСОВ	40
5 ТЕСТИРОВАНИЕ	53
5.1 МЕТОДОЛОГИЯ ТЕСТИРОВАНИЯ.....	53
5.2 ПРОВЕДЕНИЕ ФУНКЦИОНАЛЬНОГО ТЕСТИРОВАНИЯ	53
5.3 ПРОВЕДЕНИЕ ЮЗАБИЛИТИ ТЕСТИРОВАНИЯ	56
5.4 ПРОВЕДЕНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ТЕСТИРОВАНИЯ.....	56
ЗАКЛЮЧЕНИЕ.....	58
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	59
ПРИЛОЖЕНИЕ А	61
ПРИЛОЖЕНИЕ Б.....	87

ВВЕДЕНИЕ

Для эффективного управления бизнес процессами предприятия все чаще используются специализированные решения, например, ERP, MES, SCM, CRM-системы, позволяющие автоматизировать деятельность предприятия на всех уровнях, установить контроль над процессами, вести оперативный учет на основе показателей, делегировать задачи, управлять ресурсами. Потребность в подобных системах с каждым годом растет, поскольку удобный и эффективный инструмент управления позволит оценивать текущую ситуацию и более оперативно реагировать на все изменения [1].

Автоматизация бизнес-процессов позволяет значительно сократить временные издержки, предотвратить ошибки, связанные с человеческим фактором, контролировать ход выполнения каждого процесса. Большую значимость приобрели сайты. Это объясняется тем, что заинтересованные лица чаще обращают на удобство и простоту использования сайта.

Большую значимость приобрели сервисы совместной работы. Это объясняется тем, что заинтересованные лица зачастую географически распределены, растет доля сотрудников, работающих удаленно, различные компании открывают свои филиалы в нескольких регионах. Современные веб-технологии позволяют решить проблему географической удаленности сотрудников и предоставляют различные инструменты для эффективной коллективной работы.

К современным информационным системам уровня предприятия выдвигается все большее число требований, это ведет к тому, что функциональность современного корпоративного программного обеспечения со временем значительно увеличивается. Одним из важнейших требований к

такому роду программного обеспечения является интеграция с другими программными инструментами, такими как, например, офисными пакетами для работы с различными видами документов, различными отраслевыми решениями, например, системами проектирования, бухгалтерскими системами, системами документооборота [2].

Реализация программного комплекса для предприятия «Сибирский учебный центр СПАС» даст положительную имиджевую составляющую, позволит значительно сократить временные издержки и решит проблему географической удаленности сотрудников. Именно поэтому руководством АНО ДПО «Сибирский учебный центр СПАС» было принято решение разработать и внедрить программный комплекс.

ЦЕЛЬ И ЗАДАЧИ

В рамках данной работы необходимо реализовать автоматизированную систему управления деятельности учебного центра. Такого рода система призвана значительно облегчить рабочий процесс и повысить эффективность деятельности, организовать централизованное хранение проектной и экономической документации, исключить рутинные операции путем автоматизации процесса, предоставлять отчетность как по каждому заказу, так и по всем интересующим показателям в целом.

Для достижения цели необходимо выполнить следующие задачи:

- выделить и изучить бизнес-процессы предприятия, для которого разрабатывается система;
- выработать требования к разрабатываемой системе;
- изучить существующие подходы к реализации корпоративных систем и выявить наиболее подходящие варианты архитектуры системы;
- реализовать возможность совместной работы пользователей в рамках проекта с учетом их ролей в системе;

- протестировать разработанную систему;
- внедрить систему в эксплуатацию на предприятии.

СТРУКТУРА И ОБЪЕМ РАБОТЫ

Работа состоит из введения, пяти разделов, заключения и библиографии. Объем работы составляет 87 страниц, объем библиографии – 16 источников.

Введение содержит краткое описание состояния рассматриваемой предметной области, показана мотивация в разработке подобных систем для нужд предприятий.

Глава «Анализ предметной области» посвящена обзору существующих решений в области реализации веб-решений уровня предприятия. Рассмотрены подходы и лучшие практики в области построения корпоративных систем. Приводятся примеры устоявшихся архитектурных шаблонов, их сравнительный анализ, а также обоснован выбор конкретной платформы для реализации системы.

Глава «Определение требований» содержит информацию об основных функциях системы, приводятся варианты использования и типовые сценарии использования системы, приводятся требования к интеграции с уже существующей инфраструктурой предприятия.

В главе «Проектирование» уделено особое внимание проектированию архитектуры системы, ее компонентам, дается обоснование выбранной архитектуры.

В главе «Реализация» приводится описание этапов разработки, реализация основных компонентов системы.

Глава «Тестирование» содержит информацию о тестовом запуске системы, отчет модульного тестирования, тестирования производительности, а также нагрузочного тестирования.

В заключении приводятся основные выводы о проделанной работе, подводятся итоги этапов проектирования, реализации, тестирования и внедрения программного продукта, а также рассмотрены перспективы дальнейшего развития созданной системы.

1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

На сегодняшний день существует ряд решений, предлагающих автоматизацию деятельности как предприятия. В этом разделе рассмотрены некоторые из них.

1.1 ОБЗОР АНАЛОГОВ

1С: Предприятие. Система программ «1С: Предприятие» включает в себя платформу и прикладные решения, разработанные на ее основе, для автоматизации деятельности организаций и частных лиц [3]. Для использования в конкретном учреждении необходимо разработать конфигурацию, учитывающую специфику работы и бизнес-процессов в этом учреждении. Помимо этого, имеет высокую цену. Окно редактирование конфигурации в программе 1С: Предприятие приведено на рис. 1.

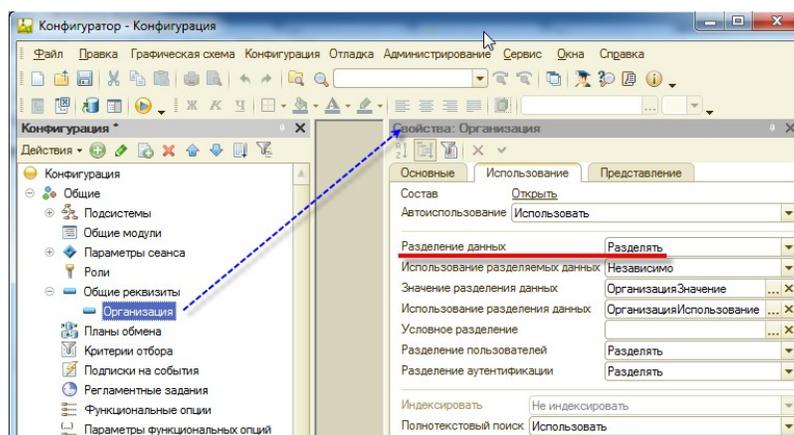


Рисунок 1 – Окно программы 1С: Предприятие

Платформа ВРМ-online в своей основе содержит наиболее прогрессивные технологии. Программные продукты на платформе ВРМonline позволяют моделировать бизнес-процессы в графическом редакторе для последующего

выполнения в системе. Созданные в редакторе сложные разветвленные процессы построены по нотации BPMN, поэтому наглядны и для аналитиков, и для пользователей [4].

StorVerk CRM – решение, которое работает на платформе 1С и используется для взаимодействия с клиентами, а также содержит много полезных отчетов по маркетингу и продажам. В CRM системе настроен обмен со всеми типовыми конфигурациями 1С. Даже если платформа дописана, можно доработать программу под нужды клиента. Продукт сертифицирован и имеет статус 1С: Совместимо [5].

vTiger CRM – система управления взаимоотношениями с клиентами (CRM) с открытым кодом (Open Source), написанная на PHP. Является форком SugarCRM, с 2008 года сама стала основой для множества форков. В отличие от классических CRM, позволяет автоматизировать не только продажи и поддержку, но и смежные аспекты работы — оплаты, ведение проектов, сервисные контракты, склад (ограниченно). Подходит для малых и средних предприятий, вовлеченных в бизнес типа B2B, особенно с длительным циклом продаж и B2C, особенно занимающихся оказанием услуг. vTiger CRM на связке технологий PHP+MySQL. Чаще всего разворачивается на серверах под управлением Linux на стеке LAMP [6].

1.2 АНАЛИЗ ОСНОВНЫХ ТЕХНОЛОГИЧЕСКИХ РЕШЕНИЙ

Обзор платформ для построения веб-приложений

NET Framework – это программная платформа, выпущенная компанией Microsoft, которая подходит для разных языков программирования. ASP.NET – технология создания веб-приложений и веб-сервисов от компании Microsoft. Она является составной частью платформы Microsoft .NET и развитием более старой технологии Microsoft ASP.

Считается, что платформа .NET Framework явилась ответом компании Microsoft на набравшую к тому времени большую популярность платформу Java. ASP.NET основывается на Common Language Runtime: разработчики могут писать код для ASP.NET, используя практически любые языки программирования, некоторые из которых входят в комплект .NET Framework (C#, Visual Basic.NET и JScript.NET), а другие могут быть установлены дополнительно (IronRuby, IronPython, PHP, Perl, Smalltalk, Haskell и др.). Тем не менее, нередко используется не вполне корректный термин «NET-программист», который, видимо, характеризует разработчика на любом языке программирования, который может быть использован в NET-проекте.

Некоторые особенности ASP.NET:

- компилируемый код выполняется быстрее, а большинство ошибок отлавливается ещё на стадии разработки;
- расширяемый набор элементов управления и библиотек классов, ускоряющий разработку;
- возможность кэширования всей страницы, её частей или данных, используемых на странице;
- возможность разделения визуальной части и бизнес-логики по разным файлам, есть возможность выделять часто используемые шаблоны пользовательских элементов управления, таких как меню сайта, наличие master-страниц для задания шаблонов оформления, поддержка AJAX (расширение ASP.NET AJAX);
- расширяемые модели событий, обработки запросов и серверных элементов управления;
- поддержка CRUD-операций при работе с таблицами через GridView;

- возможно создание веб-приложений, которые реализуют шаблон Model-View-Controller (ASP.NET MVC Framework).

NET достаточно широко распространён в сфере разработки внутрикорпоративных программных продуктов, но в веб-разработке всё же встречается относительно редко, как и другие программные продукты корпорации Microsoft. Поэтому найти разработчиков для веб-проекта бывает достаточно непросто. Использование NET «тянет» за собой покупку и иного ПО от корпорации Microsoft (серверной ОС, СУБД и т.п.) [7].

Node.js. Программная платформа, основанная на движке V8 (транслирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения. Node.js добавляет возможность JavaScript взаимодействовать с устройствами ввода вывода через свой API (написанный на C++), подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript-кода. Node.js применяется преимущественно на сервере, выполняя роль веб-сервера. В состав Node.js входит собственный установщик пакетов npm [8].

Django – фреймворк для веб-приложений на языке Python. Один из основных принципов фреймворка – DRY (don't repeat yourself). Веб-системы на Django строятся из одного или нескольких приложений, которые рекомендуется делать отчуждаемыми и подключаемыми. Это одно из заметных архитектурных отличий этого фреймворка от некоторых других (например, Ruby on Rails). Также, в отличие от многих других фреймворков, обработчики URL в Django конфигурируются явно (при помощи регулярных выражений), а не автоматически задаются из структуры контроллеров.

Django проектировался для работы под управлением Apache (с модулем mod_python) и с использованием PostgreSQL в качестве базы данных. В настоящее время, помимо PostgreSQL, Django может работать с другими

СУБД: MySQL (MariaDB), SQLite, Microsoft SQL Server, DB2, Firebird, SQL Anywhere и Oracle. Для работы с базой данных Django использует собственный ORM, в котором модель данных описывается классами Python, и по ней генерируется схема базы данных.

Архитектура Django похожа на «Модель-Представление-Контроллер» (MVC). Контроллер классической модели MVC примерно соответствует уровню, который в Django называется Представление (View), а презентационная логика Представления реализуется в Django уровне Шаблонов (Templates). Из-за этого уровневую архитектуру Django часто называют «Модель-Шаблон-Представление» (MTV).

Первоначально разработка Django велась для обеспечения более удобной работы с новостными ресурсами, что достаточно сильно отразилось на архитектуре: фреймворк предоставляет ряд средств, которые помогают в быстрой разработке веб-сайтов информационного характера. Например, разработчику не требуется создавать контроллеры и страницы для административной части сайта, в Django есть встроенное приложение для управления содержимым, которое можно включить в любой сайт, сделанный на Django, и которое может управлять сразу несколькими сайтами на одном сервере. Административное приложение позволяет создавать, изменять и удалять любые объекты наполнения сайта, протоколируя все совершённые действия, и предоставляет интерфейс для управления пользователями и группами (с пообъектным назначением прав) [9].

Ruby on Rails (RoR или) – это многоуровневый MVC-фреймворк для построения веб-приложений, использующих реляционные и NoSQL базы данных (например, MySQL, MariaDB, PostgreSQL, MongoDB). Фреймворк написан на языке программирования Ruby. Rails подходит как для разработки обычных сайтов, которые должны быть реально быстрыми, отказоустойчивыми

и работающими под высокой нагрузкой, так и для веб-приложений со сложной бизнес-логикой и динамичными web-интерфейсами. Ruby on Rails является открытым программным обеспечением и распространяется под лицензией MIT.

Основными принципами разработки на Rails являются:

- принцип DRY (Don't repeat yourself) – фреймворк предоставляет механизмы повторного использования программного кода. Это позволяет не только минимизировать дублирование кода, но и повысить скорость разработки;
- принцип Convention over configuration – по умолчанию во фреймворке используются многочисленные соглашения по конфигурации, типичные для большинства приложений. Это очень упрощает создание приложений, так как явная спецификация конфигурации требуется только в нестандартных случаях;
- автоматизированное тестирование – в составе RoR поставляются средства для проведения полностью автоматического модульного, интеграционного и функционального тестирования, а идеология Ruby on Rails предполагает использование методов разработки через тестирование (TDD – Test Driven Development). Всё это делает разработанные приложения реально надёжными [10].

СУБД для информационных систем

Выбор системы управления баз данных (СУБД) является одним из важных этапов при разработке приложений, обеспечивающих работу со структурированными данными. Выбранный программный продукт должен удовлетворять как текущим, так и будущим потребностям предприятия, при этом следует учитывать финансовые затраты на приобретение необходимого оборудования, самой системы, разработку необходимого программного обеспечения на ее основе, а также обучение персонала. На сегодняшний день

существует большое количество вариаций систем управления базами данных, различающихся архитектурой, размером, применимостью, количеством пользователей и возможностью одновременной работы с БД. Среди корпоративных СУБД наиболее популярны, следующие, применяющиеся в крупных компаниях с достаточно большим количеством пользователей: Oracle, MS SQL Server, DB2, MySQL, PostgreSQL.

Oracle Database. Это объектно-реляционная система, поддерживающая некоторые технологии, реализующие объектно-ориентированный подход, то есть обеспечивающие управление созданием и использованием баз данных [11]. Ключевыми возможностями являются: Real Application Cluster (обеспечивает работу одного экземпляра базы данных на нескольких узлах grid, позволяя управлять нагрузкой и гибко масштабировать систему в случае необходимости), Automatic Storage Management (позволяет автоматически распределять данные между имеющимися ресурсами систем хранения данных), Real Application Testing (позволяет значительно снизить затраты на проведение плановых изменений в конфигурации) и другие. Хорошо подходит для больших систем.

PostgreSQL. Это объектно-реляционная система управления базами данных (СУБД), которая имеет традиционные возможности коммерческих СУБД с расширениями, которые есть в СУБД нового поколения. PostgreSQL – это свободное и полностью открытое программное обеспечение. Сильными сторонами PostgreSQL считаются: высокопроизводительные и надежные механизмы транзакций и репликации, расширяемая система встроенных языков программирования. В стандартной поставке поддерживаются PL/pgSQL, PL/Perl, PL/Python и PL/Tcl, наследование, легкая расширяемость.

MySQL. 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 и веб-приложений, 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 такого языка, как PHP [12].

1.3 ХАРАКТЕРИСТИКА ОБЪЕКТА АВТОМАТИЗАЦИИ

1.3.1 Задачи учебного центра

АНО ДПО «Сибирский УЦ СПАС» создан 28 декабря 2010 года в целях дополнительного образования взрослых, профессиональной переподготовки (подготовки). Преподавание в учебном центре ведется на русском языке. Занятия проводятся по очной, индивидуальной и очно-заочной (с применением дистанционных технологий) формам обучения. Согласно лицензии №0081 от 27 мая 2014 года серия 72 Л 01 № 0000592, выданной Департаментом по лицензированию, государственной аккредитации, надзору и контролю в сфере образования тюменской области и уведомления о внесении в реестр аккредитованных организаций, оказывающих услуги в области охраны труда.

1.3.2 Задействованные работники и их роли

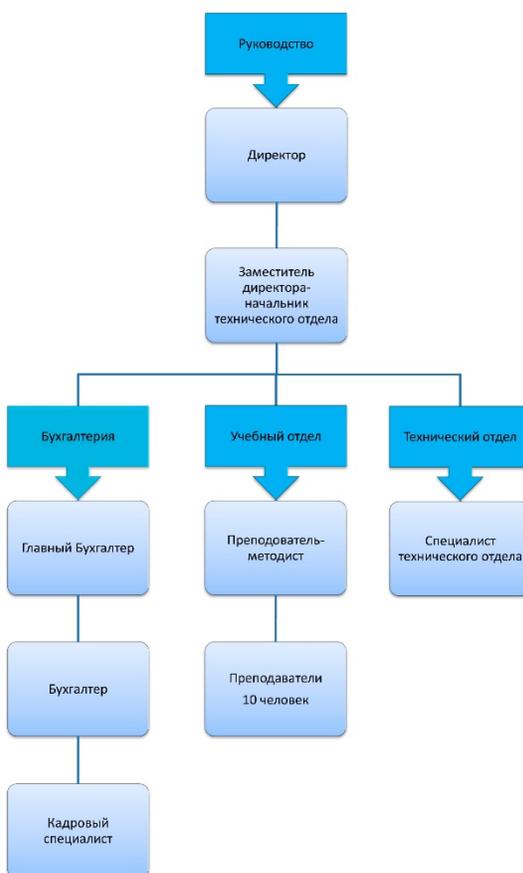


Рисунок 2 – Структура АНО ДПО «Сибирский УЦ СПАС»

Директор учебного центра в соответствии с Конституцией Российской Федерации, действующими законодательными и иными нормативными правовыми актами Российской Федерации, указами и распоряжениями Президента Российской Федерации, постановлениями и распоряжениями Правительства Российской Федерации, Уставом учебного центра осуществляет свои полномочия и организует работу учебного центра.

Организация делопроизводства осуществляется штатными преподавателями-методистами. Данные требования распространяются на организацию работы с документами (включая документы, содержащую служебную информацию ограниченного распространения – для служебного

пользования), представленными на электронных носителях информации, в том числе на организацию работы с электронными документами, включая их подготовку, регистрацию и учет.

Канцелярия обеспечивает прием корреспонденции, учет и регистрацию документов, передачу документов на исполнение работникам в соответствии с резолюциями. Прием письменных обращений граждан также осуществляет канцелярия. Все обращения граждан из почтовых отделений, посредством факсимильной и электронной связи или на личном приеме, обрабатываются и рассматриваются в соответствии с требованиями Федерального закона от 02.05.2006 № 59-ФЗ «О порядке рассмотрения обращения граждан Российской Федерации» (Собрание законодательства Российской Федерации 2006, N 19, ст. 2060).

1.3.3 Документы учебного центра

В учебном центре действуют следующие распорядительные документы: приказы и распоряжения, решения педагогического совета, протоколы служебных совещаний, проводимых директором учебного центра. Приказы по учебному центру подписывает директор учебного центра либо лицо, на которое приказом возложено исполнение обязанностей директора учебного центра в период его отсутствия (командировка, отпуск, болезнь). Направление в командировку работников учебного центра осуществляется приказами директора учебного центра на основании положения о служебных командировках. Решения педагогического совета учебного центра (соответствующие протоколы этих заседаний) подписываются в порядке, определенном положением о педагогическом совете учебного центра. Протоколы служебных совещаний подписываются должностными лицами, проводившими эти совещания. Порядок подготовки и представления на

подпись проектов протоколов совещаний, проводимых руководством учебного центра, определяется руководящими документами учебного центра.

Поступающие на обучение в Учебный центр предоставляют в обязательном порядке следующие документы:

- юридические лица – заявку с приложением списка сотрудников, направляемых на обучение, физические лица - заявление о приеме на обучение, с указанием образовательной программы, выбранной для изучения
- договор об оказании платных образовательных услуг;
- копию документа, удостоверяющего личность и гражданство (паспорт);
- соглашение об обработке персональных данных;
- копию документа об образовании или справку из образовательного учреждения об обучении.

При необходимости, от поступающих на обучение в Учебный центр могут быть затребованы дополнительные документы:

- справка по результатам медицинского освидетельствования;
- фото размером 3x4;
- копия документа о смене фамилии;
- копия трудовой книжки;
- копия платежного поручения, подтверждающего факт оплаты образовательных услуг.

Основанием для оказания платных образовательных услуг слушателям является письменный договор между Учебным центром и заказчиком. Договор регламентирует условия и сроки получения образовательных услуг, порядок оплаты, права, обязанности и ответственность сторон.

Таблица 1 – Основные документы учебного центра

Документ \ Вид	Приказ	Протокол	Удостоверение
Тип	Зачисление/Отчисление /Окончание	-	-
На основании приказа	-	+	-
На основании протокола	-	-	+

1.3.4 Описание учебного процесса

Учебный процесс в учебном центре организуется в соответствии с Федеральным законом от 29.12.2012 № 273 РФ «Об образовании в Российской Федерации» (Собрание законодательства Российской Федерации 2012, N 53 (ч. 1), ст. 7598, 2015, N 1 (часть I), ст. 53), Методическими рекомендациями по организации учебного процесса в государственных образовательных учреждениях дополнительного профессионального образования учебных центрах (учебных пунктах), а также Уставом учебного центра. Основные усилия при организации и проведении учебной работы в центре направлены на качественное выполнение требований руководящих документов, учебных программ, квалификационных требований к профессиональной подготовке, неукоснительное соблюдение расписания занятий и распорядка дня.

Учебный центр реализует различные по срокам и форме обучения дополнительные профессиональные программы; программы профессионального обучения; краткосрочные образовательные программы (не заканчивающиеся итоговой аттестацией и (или) срок освоения которых менее 16 часов). Содержание образовательного процесса определяется Учебным центром на основе учебных планов и программ, утверждаемых директором Учебного центра. Обучение в Учебном центре ведется на русском языке –

государственном языке Российской Федерации. Основными видами учебных занятий являются: уроки, практические занятия, консультации, занятия на автоматизированных средствах и системах управления, на средствах связи и вычислительной технике, самостоятельная работа слушателей, практика, стажировка, аттестационные и выпускные квалификационные работы, деловые игры, индивидуальные собеседования, учения. Все виды учебно-методических материалов хранятся у преподавателей в течение всего периода обучения слушателей по действующим учебным планам и программам. Самостоятельная работа слушателей направлена на закрепление и углубление полученных знаний и навыков, на поиск и приобретение новых знаний, в том числе с использованием автоматизированных обучающих курсов (систем), а также выполнения учебных заданий, подготовку к предстоящим занятиям, зачетам и экзаменам. Преподаватели учебных групп ежедневно организуют самостоятельную работу слушателей (согласно расписанию занятий) с использованием дистанционных технологий, а технический отдел еженедельно осуществляет контроль за организацией этой работы. Консультации по предметам проводятся согласно графику, индивидуальные – при необходимости, в часы самостоятельной подготовки. Текущий контроль успеваемости проводится в ходе всех видов занятий в форме, избранной преподавателем или предусмотренной тематическим планом. Результаты контроля успеваемости отражаются в «Журнале учета учебных занятий» и используются для оперативного управления образовательным процессом. Образовательные программы могут реализовываться: полностью или частично в форме стажировки; с использованием модульного принципа представления содержания программ.

По каждой учебной группе приказом директора Учебного центра назначается куратор из числа преподавателей. В функции куратора группы

входят: персональный учет посещения слушателями аудиторных занятий; контроль за результатами промежуточной и итоговой аттестации.

Прием на обучение проводится в течение всего календарного года в сроки, устанавливаемые графиком обучения слушателей в Учебном центре, а также по согласованию с корпоративными заказчиками. Комплектование учебных групп осуществляется преподавателями методистами на основании поступивших заявок/заявлений. По программам профессионального обучения: – лица, ранее не имевшие профессии рабочего или должности служащего (по программам профессиональной подготовки рабочих и служащих); – лица, имеющие рабочую профессию или должность служащего, в целях совершенствования профессиональных знаний и навыков в рамках имеющейся профессии рабочего или должности служащего (по программам повышения квалификации рабочих и служащих); – лица, имеющие рабочую профессию или должность служащего, в целях получения новой профессии рабочего или новой должности служащего (по программам профессиональной переподготовки рабочих и служащих). Вступительные испытания (входной контроль) проводятся, если они включены в образовательную программу.

Зачисление граждан на обучение производится приказом директора Учебного центра, при условии предоставления обязательных документов, прохождения вступительных испытаний (при наличии в образовательной программе) и их результатам, оплаты образовательных услуг. Лицо, зачисленное в Учебный центр приказом директора для обучения по образовательным программам, приобретает статус «слушатель». Освоение образовательных программ завершается: дополнительных профессиональных программ - итоговой аттестацией слушателей в форме, определяемой Учебным центром самостоятельно; программ профессионального обучения - итоговой

аттестацией слушателей в форме квалификационного экзамена, который включает в себя проверку теоретических знаний и практическую квалификационную работу. Право на получение дополнительного профессионального образования лицами без гражданства регламентируется Федеральным законом от 24 мая 1999 № 99-ФЗ «О государственной политике Российской Федерации в отношении соотечественников за рубежом». Прием иностранных граждан на обучение по образовательным программам проводится с учетом признания в Российской Федерации документов об образовании и (или) квалификации, полученных в иностранном государстве. Признание и установление в Российской Федерации эквивалентности документов об образовании осуществляет Федеральная служба по надзору в сфере образования и науки (Рос. Обр. надзор).

1.3.5 Документы, выдаваемые при завершении обучения

Лицам, успешно прошедшим краткосрочные курсы и семинары, выдаются сертификаты (справки).

Лицам, успешно прошедшим повышение квалификации, итоговую аттестацию, выдается удостоверение повышения квалификации.

Лицам, успешно прошедшим профессиональную переподготовку, итоговую аттестацию, выдается диплом профессиональной переподготовки с присвоением дополнительной специальности.

Лицам, успешно прошедшим профессиональное обучение, итоговую аттестацию, присваиваются квалификационные разряды, классы, категории по соответствующим профессиям рабочих и выдаются свидетельства о профессии рабочего.

Лицам, не прошедшим итоговой аттестации или получившим на итоговой аттестации неудовлетворительные результаты, а также лицам, освоившим часть

образовательной программы и (или) отчисленным из Учебного центра выдается справка об обучении или о периоде обучения установленного образца [6].

1.4 ВЫВОД

Из материала главы следует, что

1 После анализа существующих проектов и деятельности учебного центра было принято решение реализовывать собственную систему по следующим причинам:

- специфичный алгоритм деятельности предприятия «Сибирский учебный центр СПАС»;
- неполный функционал существующих решений, их высокая стоимость и при этом необходимость доработки.

Таблица 2 - Обзор аналогов

	1С: Предприятие	BPM-Online	StorVerk CRM
Стоимость	Внедрение от 58 т.р.	Внедрение от 162 т.р.	Внедрение от 97 т.р.
Требуемый функционал	-	-	-

2 Для реализации веб-приложения была выбрана CRM платформа с открытым кодом (Open Source) vTiger в качестве основы для приложения. С использованием LAMP – комплекс серверного программного обеспечения, широко используемый во всемирной паутине. LAMP назван по первым буквам входящих в его состав компонентов:

Linux – операционная система Linux; Apache – веб-сервер;

MariaDB / MySQL – СУБД; PHP;

Данный набор ПО устанавливают на сервер для отображения динамических веб-сайтов и веб-приложений. Эта платформа позволяет установить большинство типов веб-сайтов и сетевого программного обеспечения на вашем сервере.

3 В качестве хранилища базы данных MySQL, поскольку данный подход базируется на свободно-распространяемых продуктах и хорошо подходит для реализации веб-систем с небольшим объемом хранимых данных.

2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ

Автоматизация процесса обучения см. п. п. 1.3.4 внедрение электронного документооборота см. п. п. 1.3.3 и 1.3.5, систематизирование работы сотрудников между собой и работы с клиентами.

2.1 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

Автоматизация процесса обучения - контроль на какой стадии находится студент от подачи документов на обучение и до выдачи сертификата см. п. п. 1.3.4 База данных об учебном процессе: студенты распределены по группам с информацией о них, преподаватели, расписание занятий, дисциплины.

Менеджер задач – указываются ответственные и участники задач по процессу. Система должна напоминать сотрудникам о дедлайнах по задачам, сообщать о событиях.

Управление клиентским сервисом – обработка данных физических/юридических лиц (занесение их в базу клиентов/организаций).

Электронный документооборот – совокупность автоматизированных процессов по работе с документами, представленными в электронном виде, с реализацией концепции «безбумажного делопроизводства».

Требования к пользователям:

Каждый пользователь, работающий в системе, обязан:

- иметь логин и пароль для входа, сгенерированный специалистом технического отдела в среде дистанционного обучения Moodle.

В системе должны быть выделены следующие типы пользователей:

- администратор;

- руководитель;
- преподаватель/сотрудник;
- слушатель.

Администратор должен иметь возможность пользоваться и управлять менеджером задач, клиентским сервисом, электронным документооборотом и учебным процессом имеет возможность редактировать любую информацию в модулях/пользователях.

Руководитель должен иметь доступ ко всему функционалу и иметь привилегии по указанию ответственных и участников процесса в менеджере задач.

Сотрудник должен иметь доступ к менеджеру задач, клиентскому сервису, процессу обучения, электронному документообороту.

Студент должен иметь возможность просматривать свою успеваемость и расписание занятий.

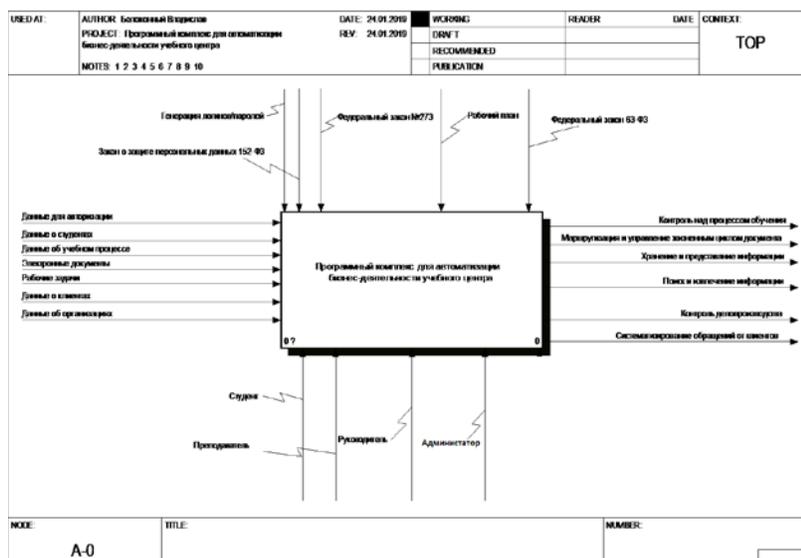


Рисунок 3 – Бизнес-процесс верхний уровень

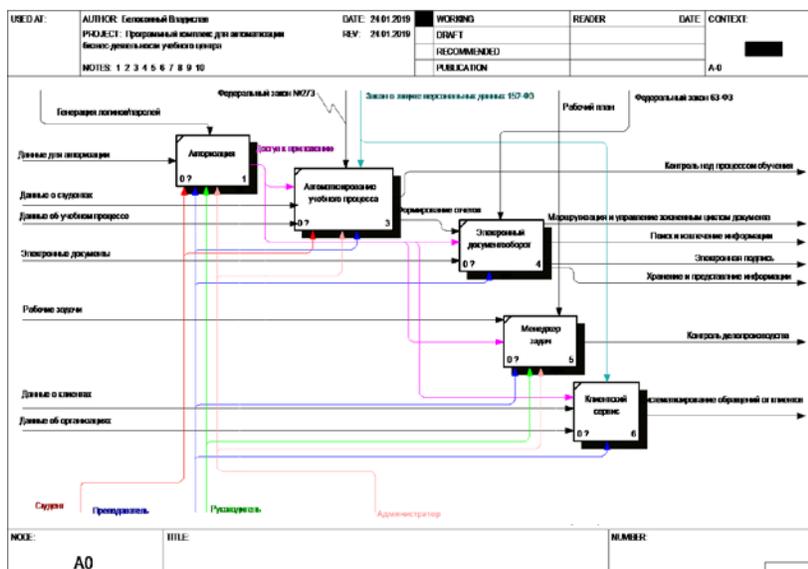


Рисунок 4 – Бизнес-процесс функциональный уровень

Требования к менеджеру задач:

Менеджер задач должен включать в себя следующий функционал:

- интерактивная доска для сотрудников и руководителя, на которой можно добавлять задачи;
- список событий, которые нужно ввести в работу (вывести на интерактивную доску и добавить ответственных и участников) и которые уже выведены из работы;
- напоминания для сотрудников об их важных событиях;

У процесса есть следующие атрибуты:

- ответственный;
- дедлайн;
- время в работе;

Требования к автоматизации процесса обучения

Автоматизация процесса обучения должна включать в себя следующий функционал для студента:

- просмотр расписания занятий;

- просмотр информации о своей успеваемости;
- просмотр на какой стадии находится процесс обучения.

Автоматизация процесса обучения должна включать в себя следующий функционал для сотрудника:

- поиск по студентам и просмотр их успеваемости;
- изменение информации о студенте и его успеваемости;
- добавление и редактирование учебных планов.

Требования к клиентскому сервису:

Клиентский сервис должен включать в себя следующий функционал:

– работа с базой клиентов/организаций (название, ФИО, телефон, причины обращений):

- добавление клиентов/организаций;
- удаление клиентов/организаций;
- поиск клиентов/организаций, просмотр их обращений (обращения клиентов заносятся с веб-сайта или вручную).

Требования к лингвистическому обеспечению:

Приложение должно быть доступно на русском языке.

2.2 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

Система должна быть реализована в виде веб-приложения.

Веб-приложение должно корректно работать и отображаться в браузерах Chrome версии 45 и выше, Mozilla Firefox версии 40 и выше, Microsoft Edge версии 20 и выше, Opera версии 12 и выше.

Требования к визуальному оформлению:

Приложение должно иметь интуитивно понятный интерфейс.

В левой части должен быть выдвигающийся список с основными компонентами программы (менеджер задач, документооборот и т.д.), под ним должны отображаться функции этого компонента.

По центру должна отображаться визуальная часть программы (документы, планирование и т.д.)

В правой части вспомогательные кнопки для работы с функциями (сохранить, чекбоксы, текстбоксы).

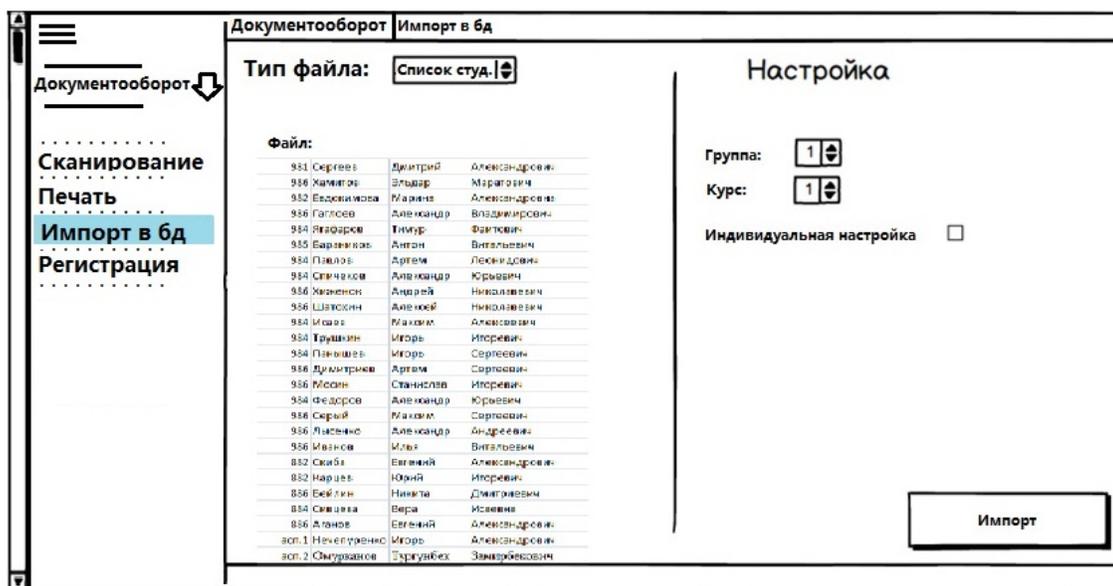


Рисунок 5 – Пример пользовательского интерфейса

3 ПРОЕКТИРОВАНИЕ

3.1 АРХИТЕКТУРА ПРЕДЛАГАЕМОГО РЕШЕНИЯ

Выбор архитектуры сетевого приложения был между двухзвенной и трехзвенной архитектурой [11].

Двухзвенная архитектура проще, так как все запросы обслуживаются одним сервером, но именно из-за этого она менее надежна и предъявляет повышенные требования к производительности сервера.

Трехзвенная архитектура сложнее, но благодаря тому, что функции распределены между серверами второго и третьего уровня, эта архитектура представляет:

1. Высокую степень гибкости и масштабируемости.
2. Высокую безопасность (т.к. защиту можно определить для каждого сервиса или уровня).
3. Высокую производительность (т.к. задачи распределены между серверами).

Как весомое преимущество LAMP сборки:

1. Она замечательно подходит для быстрого развертывания приложения, из-за простой конфигурации;

2. Гибкость при подборе баз данных, web серверов и языков сценариев. Актуальной заменой для MySQL служат PostgreSQL и SQLite. Python, Perl и Ruby можно заменить PHP. А Nginx, Cherokee и Lighttpd альтернатива Apache [12].

Поэтому архитектура проектируемого решения, изображенная на рисунке 6, представляет собой классическую трехзвенную систему на сборке LAMP. Первым звеном является клиент, который представлен веб-страницей в браузере пользователя. Вторым – сервер, расположенный на виртуальной машине; он выполняет обработку запросов, взаимодействие с базой данных и выдачу результата. Третьим звеном является непосредственно СУБД.

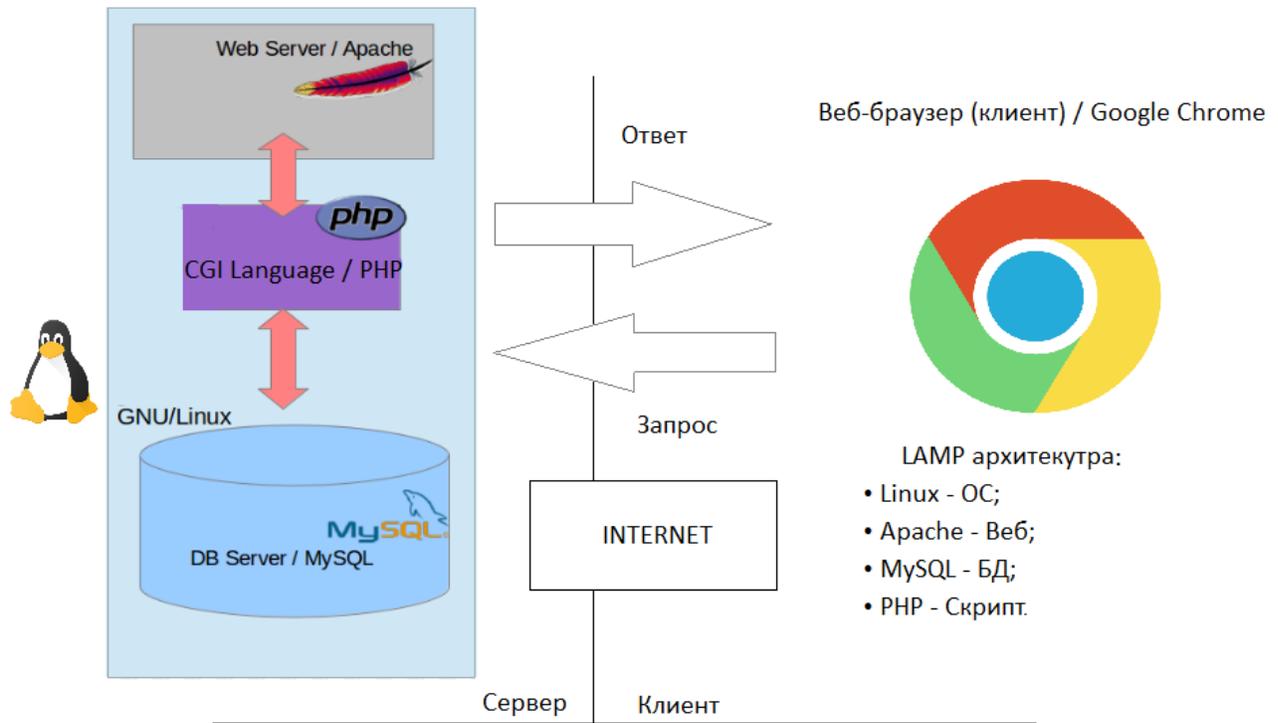


Рисунок 6 – Архитектура системы

3.2 КОМПОНЕНТЫ СИСТЕМЫ

Для пояснения действий, выполняемых сотрудниками, на рис. 7 приводится диаграмма прецедентов.

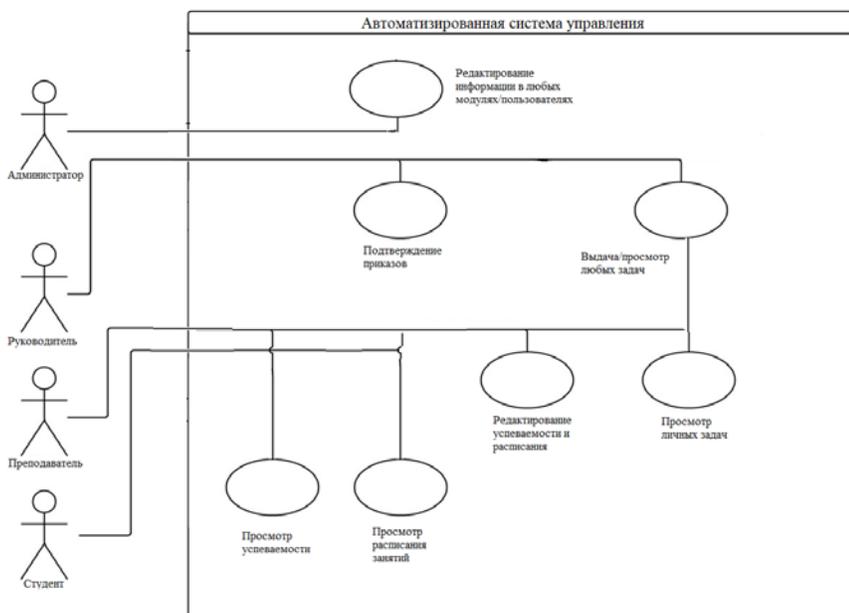


Рисунок 7 – Диаграмма прецедентов

Система состоит из менеджера задач и 9 модулей для реализации документооборота и автоматизации процесса обучения:

Модуль «Контрагенты» – номер, ответственный (ссылка на пользователя)

Модуль «Группы» – связанный список контактов-студентов, номер, куратор (ссылка на преподавателя)

Модуль «Студенты» – ФИО, контрагент, ответственный.

Модуль «Приказы» – номер, группа, контакт. Типы: Допуск, Зачисление, отчисление, окончание. Возможность вывести в PDF. (3 шаблона PDF)

Модуль «Протоколы» – множество пользователей-преподавателей, один председатель. Возможность вывести в PDF. (1 шаблона PDF)

Модуль «Удостоверения» – номер, кому выдано, ссылка на протокол

Модуль «Курсы» – преподаватель, дисциплина, группа Возможность вывести в PDF. (1 шаблона PDF “Журнал”)

Модуль «Занятие» – дата, курс, список студентов, которые отсутствуют.

Модуль «Зачёт» – контакт, курс, номер билета, оценка.

3.3 ОПИСАНИЕ ДАННЫХ

Входные данные для веб-сервиса будут определяться пользовательским взаимодействием со страницей сайта. Пользователем вводятся данные в различных форматах (текст, даты, числовые данные), а также инициируется отправка данных на сервер.

Выходные данные представлены, прежде всего, отрисованными HTML страницами, приходящими в браузер от сервера. Кроме того, выходными данными являются email сообщения и уведомления (текстовый формат), получаемые пользователем.

Для хранения данных на сервере используется база данных, работающая под управлением СУБД MySQL.

Концептуальная схема базы данных (универсальное представление структуры данных в рамках предметной области, независимое от конечной реализации базы данных и аппаратной платформы), представлена на рис. 8.

4 РЕАЛИЗАЦИЯ ПРОГРАММНОГО КОМПЛЕКСА

Так как для реализации проекта был выбран комплекс LAMP, потребовался компьютер на ОС Linux и его настройка[13].

Была выбрана виртуальная машина как ресурс Яндекс Облака.

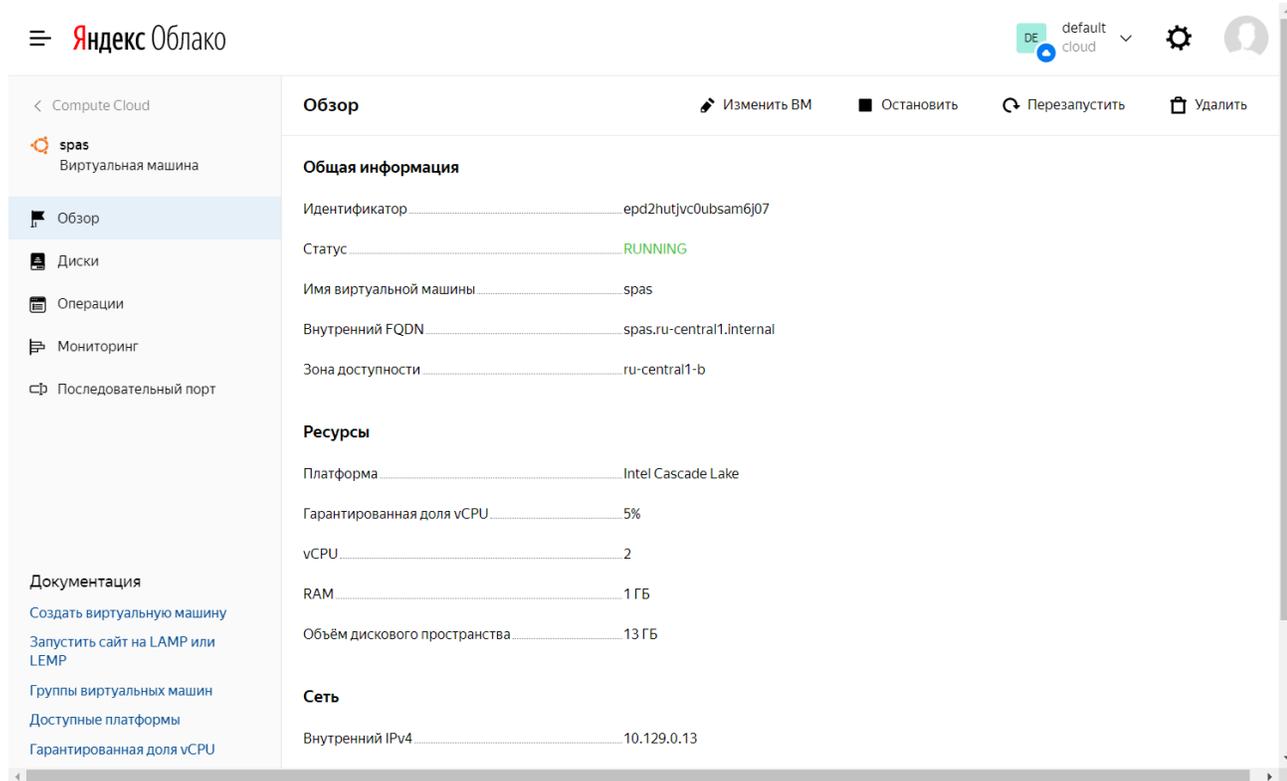


Рисунок 9 – Виртуальная машина на ОС Linux

На виртуальной машине установлен веб сервер Apache.

```
sudo apt-get install apache2 libapache2-mod-auth-mysql
```

Интерпретатор php с дополнительными библиотеками.

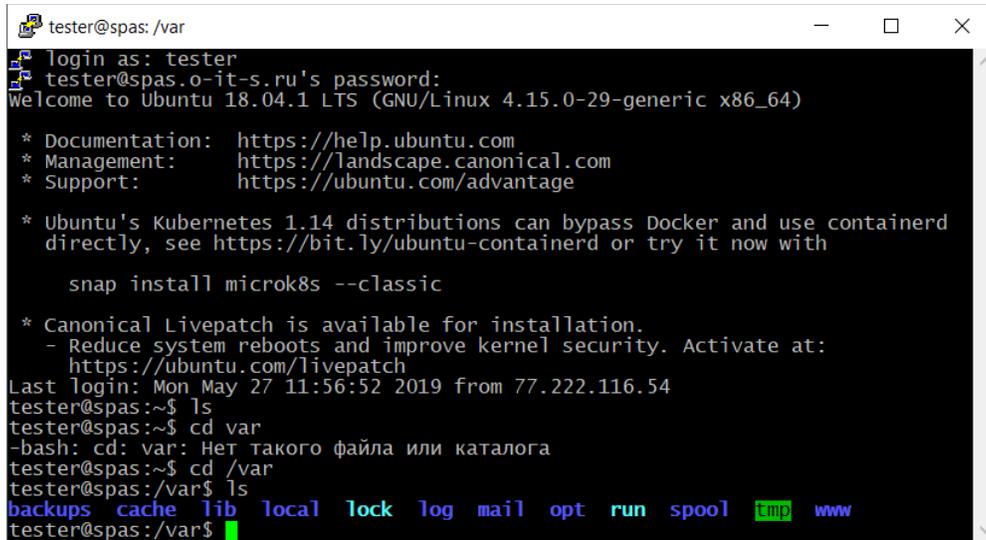
```
sudo apt-get install php5-common php5 libapache2-mod-php5 php5-cli php5-cgi php5-mysql
```

```
sudo /etc/init.d/apache2 reload
```

Установлена СУБД MySQL

sudo apt-get install mysql-server mysql-client mysql-admin

На рисунке 10 и 11 представлено подключение к серверу через SSH и FTP
соответственно.



```
tester@spas: /var
login as: tester
tester@spas.o-it-s.ru's password:
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-29-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Ubuntu's Kubernetes 1.14 distributions can bypass Docker and use containerd
  directly, see https://bit.ly/ubuntu-containerd or try it now with

snap install microk8s --classic

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
Last login: Mon May 27 11:56:52 2019 from 77.222.116.54
tester@spas:~$ ls
tester@spas:~$ cd var
-bash: cd: var: Нет такого файла или каталога
tester@spas:~$ cd /var
tester@spas:/var$ ls
backups cache lib local lock log mail opt run spool tmp www
tester@spas:/var$
```

Рисунок 10 – Подключение к ВМ (PuTTY)

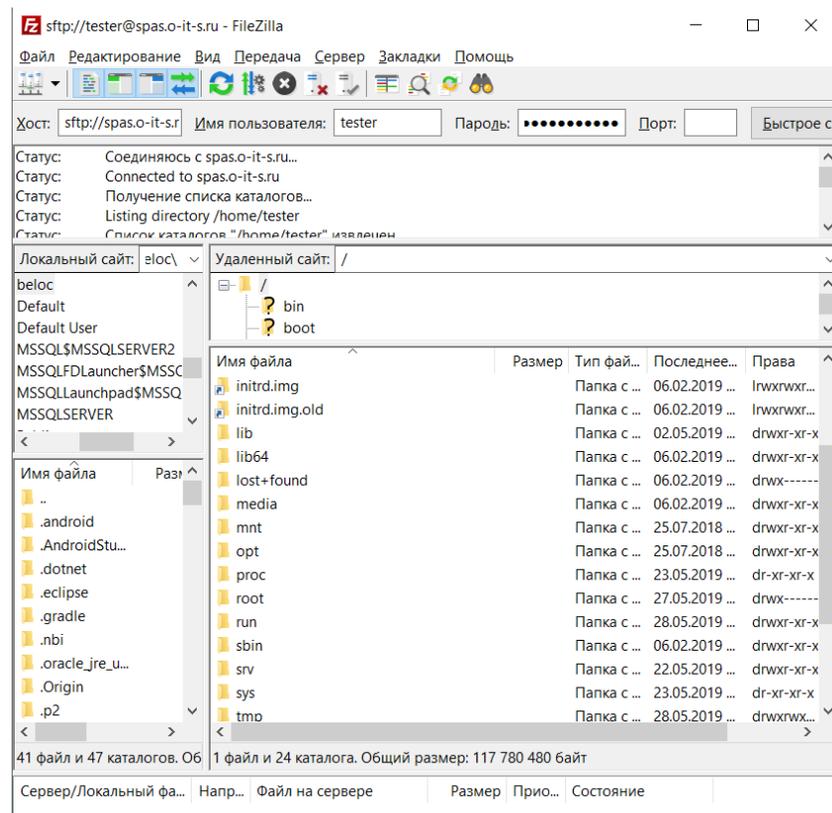


Рисунок 11 – Подключение к ВМ (FileZilla)

4.1 РЕАЛИЗАЦИЯ ИНТЕРФЕЙСОВ

Основным интерфейсом приложения является шаблон навигации, представлен в левой части рисунка 13. Он представляет собой описание навигационного меню, которое повторяется на каждой странице сайта. С помощью использования шаблона навигации упрощается написание проекта и уменьшается дублирование кода.

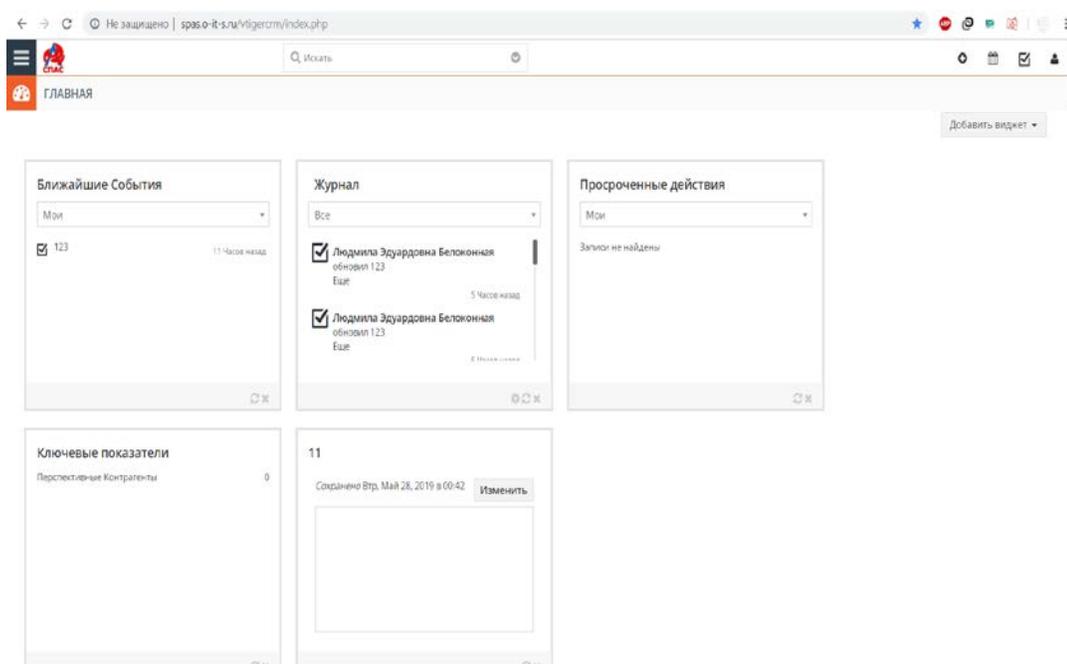


Рисунок 12 – Свернутый интерфейс навигации

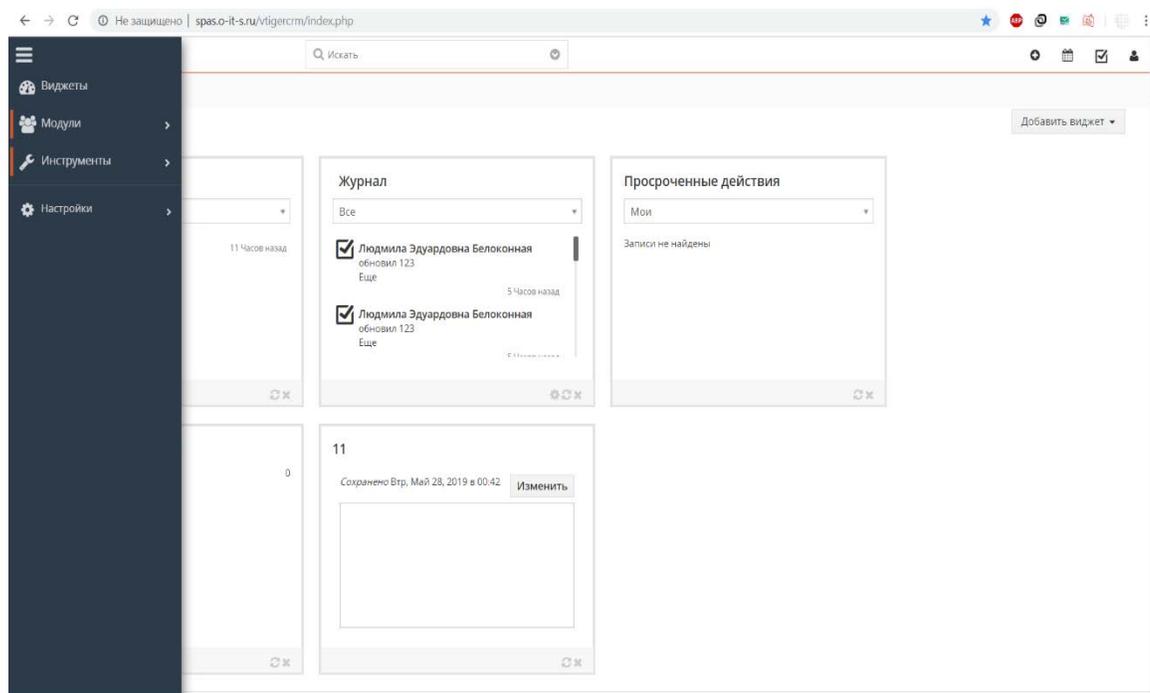


Рисунок 13 – Развернутый интерфейс навигации

Для отображения основных модулей приложения в интерфейсе навигации есть кнопка «Модули», здесь находятся 9 модулей необходимые для документооборота и автоматизации процесса обучения.

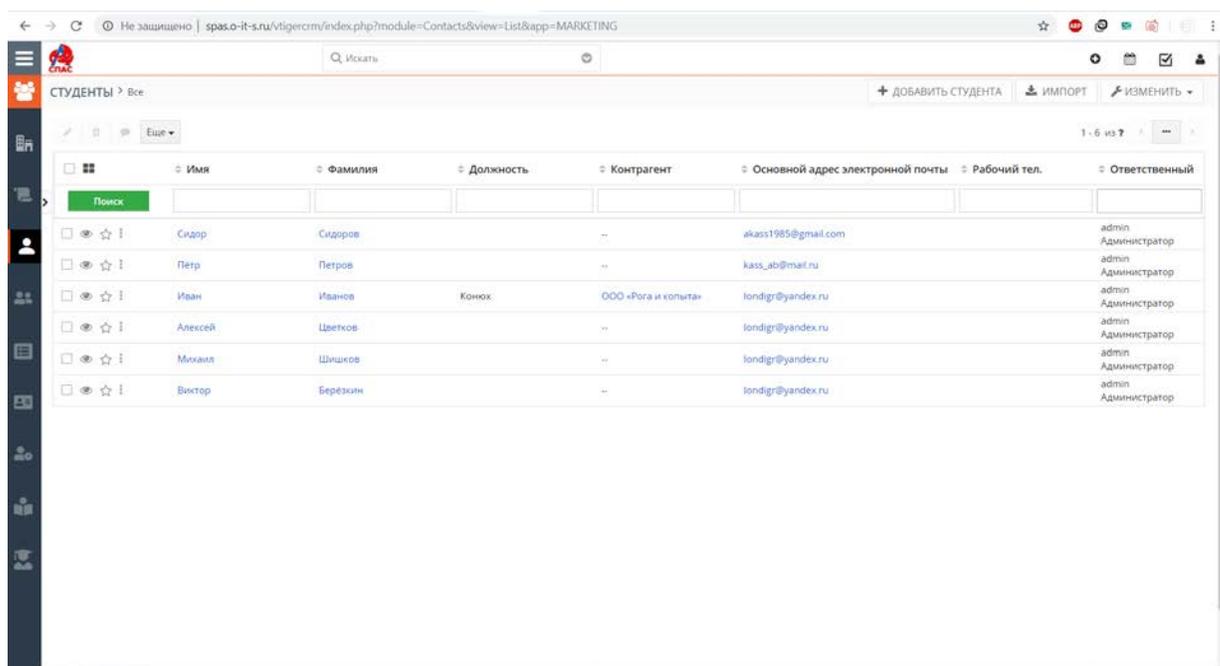


Рисунок 14 – Развернутый модуль «студенты»

Процедура создания модуля состоит из шагов:

- а) реализовать код-скелета модуля см. в Приложении А;
- б) запуск сценария создания модуля, заносит в БД MySQL информацию о модуле см. в Приложении А.

Всё ядро vTiger сплошь использует концепцию инверсии управления (IoC), поэтому при дальнейшей разработке модуля есть возможность переопределить внутри нашего модуля разные классы: модель записи, модель поля и т.п. Это позволяет в модуле сделать своё нестандартное поведение, практически любое. При этом сохраняются базовые узловые свойства предка класса Entity см. в Приложении А, от которого наследуются все модули. В нём есть инфо о том, кто создал запись, кто ответственный за неё, флаг «удалено» и прочие подобные атрибуты[14].

На рисунке 15 продемонстрирована возможность экспорта в pdf-файл: приказов 3-ех типов окончание учебы/зачисление/допуск, протоколов и курсов (журнал посещаемости и успеваемости). Все документы соответствуют действующим предприятия «Сибирский учебный центр СПАС». Приказы и протоколы можно экспортировать с факсимиле.

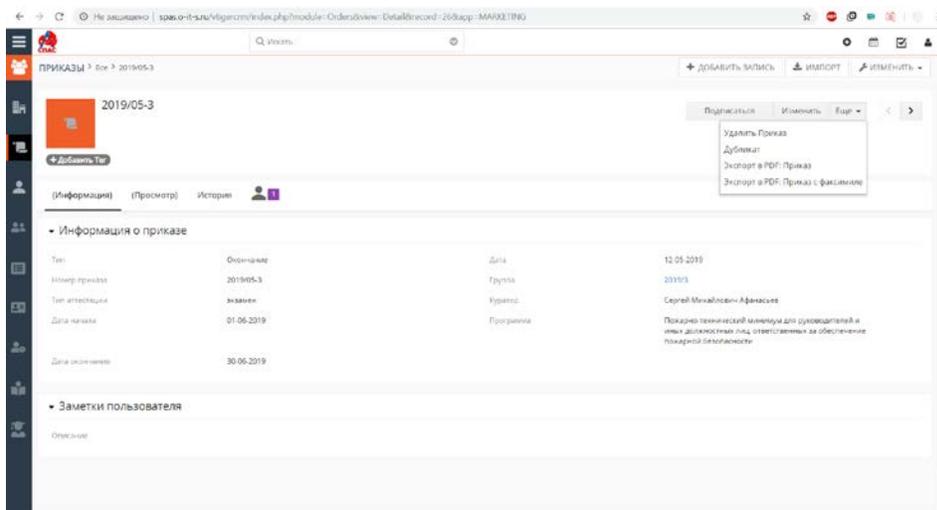


Рисунок 15 – Возможность экспорта в PDF «Приказ»

Примеры выходных документов представлены на рисунках 16, 17, 18, 19.

 Автономная некоммерческая организация
дополнительного профессионального образования
«Сибирский учебный центр СПАС»
адрес: 625048, Россия, Тюмень, г. ул. Широкая 138/4, ул. Широкая 138/4
телефакс: 8(3452)554058(3452)554058, e-mail: mail@spas72.com

ПРИКАЗ
г. Тюмень

12 Мая 2019 г. №2019/05-3

Об окончании обучения в АНО ДПО «Сибирский учебный центр СПАС»

В АНО ДПО «Сибирский учебный центр СПАС» состоялась итоговая аттестация (экзамен) по окончании обучения, по программе «Пожарно-технический минимум для руководителей и иных должностных лиц, ответственных за обеспечение пожарной безопасности». На основании вышеизложенного, п р и к а з ы в а ю :

1. Считать окончившимися с 30 Июня 2019 г. учебную группу №2019/3

№ п/п	ФИО слушателя	Подразделение
1	Иван Иванов	ООО «Рога и копыта»
2	Петр Петров	
3	Сидор Сидоров	

Директор Белокодная Л.Э.

Рисунок 16 – Приказ в PDF

Автономная некоммерческая организация
дополнительного профессионального образования
«Сибирский учебный центр СПАС»

ЖУРНАЛ
контроля учебных занятий

Начат _____
Окончен _____

Рисунок 17 – Журнал в PDF

ДИСЦИПЛИНА
Пожарно-технический минимум для руководителей и иных должностных лиц, ответственных за обеспечение пожарной безопасности

Оценка знаний и посещаемость

№ п/п	Дата Ф.И.О.	05	05	05	05	05
		09	08	10	11	12
1	Иванов Иван	н	н	н		
2	Петров Пётр	н				

ПРЕПОДАВАТЕЛЬ Афанасьев Сергей Михайлович

Рисунок 18 – Журнал в PDF

Дата проведения занятий	Кол-во часов, вид занятий	Номер, наименование темы	Фамилия и подпись преподавателя
2019-05-09	2 лекция	Организационные мероприятия по обеспечению пожарной безопасности детских учреждений, школ и дошкольных учреждений	
2019-05-08	3 лекция	Основные нормативные документы, регламентирующие требования пожарной безопасности	
2019-05-10	3 лекция	Требования пожарной безопасности к зданиям, помещениям и территориям в детских учреждениях, школах и дошкольных учреждениях	
2019-05-11	2 лекция	Меры пожарной безопасности в детских учреждениях, школ и дошкольных учреждений	
2019-05-12	2 лекция	Средства тушения пожаров и правила их применения для тушения пожаров, действия при пожаре и вызов пожарной охраны	
2019-05-13	4 зачёт		

Количество часов по программе: 16
Фактически проведено _____

Рисунок 19 – Журнал в PDF

Для формирования PDF: используется библиотека tcpdf. Библиотека позволяет формировать PDF из удобного HTML-разметочного кода шаблона документа [15]. Также каждый модуль может определять некоторые свои переменные для подстановки в PDF. Для примера есть файл OrdersPDFController.php см. в Приложении А. В нем можно проследить иерархию и посмотреть, как подменяются унаследованные свойства.

На рисунке 20 продемонстрирован менеджер задач с возможностью назначения ответственных, назначение сроков, выставления приоритета и напоминанием о крайних сроках.

Менеджер задач – стандартный модуль vTiger. Модуль подходит по требованиям заказчика, которые указаны в п. 2.1.

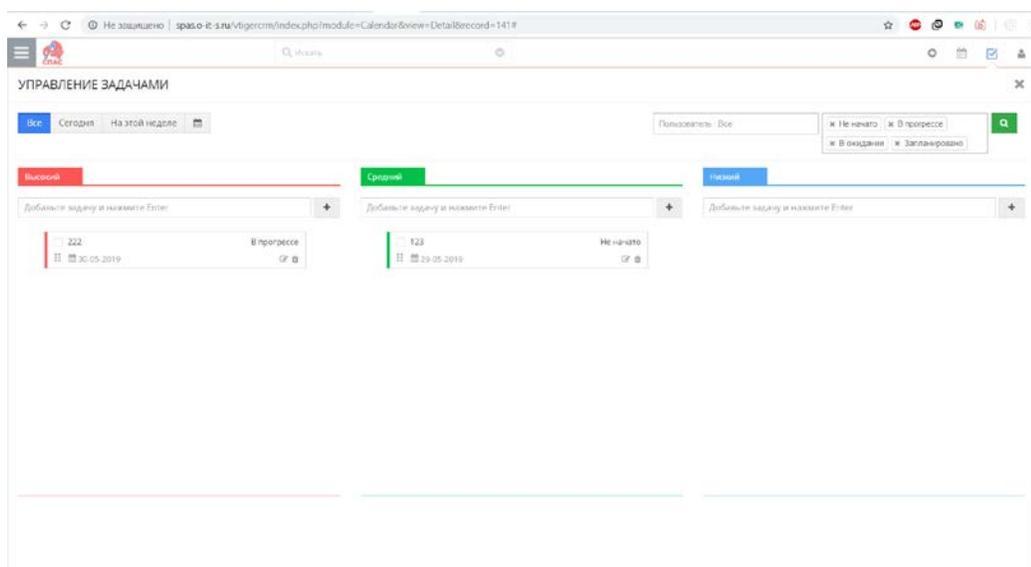


Рисунок 20 – Менеджер задач

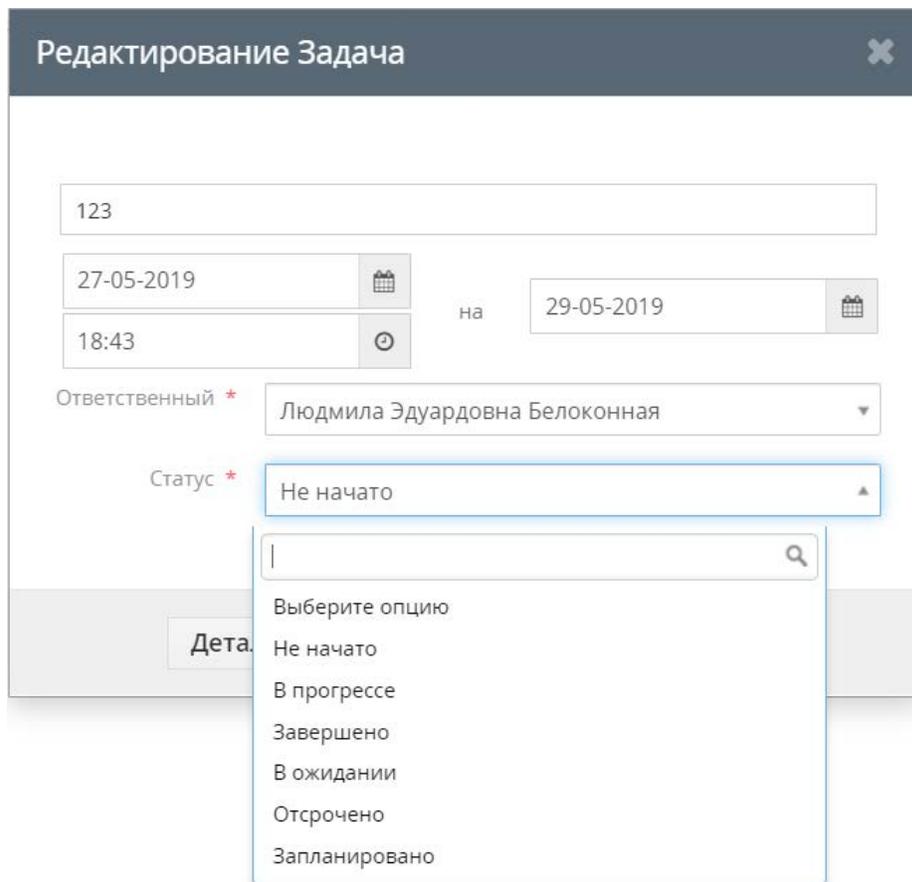


Рисунок 21 – Создание задач

ПРИМЕР РАБОТЫ ПРОГРАММЫ

Допустим, мы хотим добавить курс с новой группой и новыми студентами, у которых будут занятия. Также мы хотим вывести журнал в pdf-файл для этого курса.

Входные данные:

1) Студенты: Добавляем студентов.

Таблица 3 – Входные данные студентов

ФИО	Должность	Место работы
Константин Юрьевич Юрин	Начальник отделения	Тюменская таможня
Евгений Леонидович Попов	Начальник отдела	Тюменская таможня

2) Группа «3» : Создаем группу.

3) Курсы: Добавим курс «2019/1» с группой «3».

Таблица 4 – Входные данные курса

Дисциплина	Группа	Курс
Пожарно-технический минимум	3	2019/1

4) Занятия: Добавим к курсу «2019/1» занятия.

Таблица 5 – Входные данные занятий

Дата	Курс	Вид занятия	Кол-во часов	Тема занятия
10-05-2019	2019/1	Лекция	3	Требования Пожарной безопасности
13-05-2019	2019/1	Зачет	4	Зачет
12-05-2019	2019/1	Лекция	2	Средства тушения пожаров
11-05-2019	2019/1	Лекция	2	Меры пожарной безопасности
09-05-2019	2019/1	Лекция	2	Организационные мероприятия
08-05-2019	2019/1	Лекция	3	Основные нормативные документы

4) Группа: Зачислим в группу «3» – 2 студента из Тюменской таможни.

5) Журнал: Через модуль курсы выведем журнал в pdf-файл для нужного курса, в нашем случае «2019/1».

Выходные данные:

Имя	Фамилия	Должность	Контрагент	Основной адрес электронной почты	Рабочий тел.	Ответственный
Евгений	Белочкин	Администратор	---	evgeny@unibek.ru		Администратор Людмила Захаровна Белоносова
Константин	Курочкин	Педагог	Телекоммуникационная группа			Педагог Людмила Захаровна Белоносова
Евгений	Легендрович	Педагог	Телекоммуникационная группа			Педагог Людмила Захаровна Белоносова
Елена	Ивановна	МАОУ детский сад №112 города Томска				МАОУ детский сад №112 города Томска Людмила Захаровна Белоносова
Светлана	Юрьевна	Руджова	МАОУ детский сад №112 города Томска			МАОУ детский сад №112 города Томска Людмила Захаровна Белоносова

Рисунок 22 – Пример выходных данных для студентов

Номер группы
3
07
2

Рисунок 23 – Пример выходных данных для групп

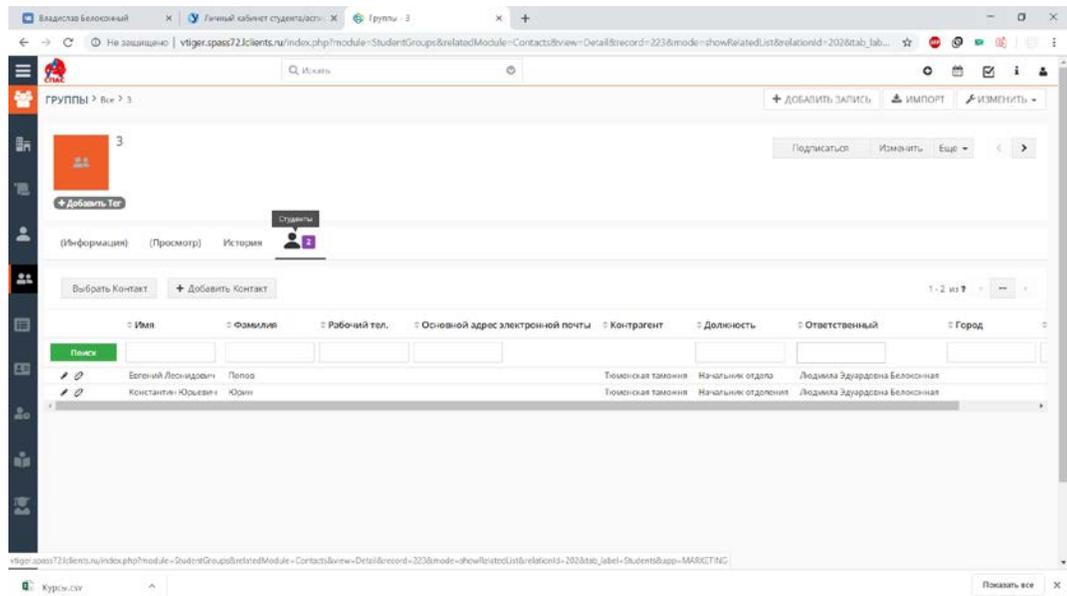


Рисунок 24 – Пример записи студента в группу

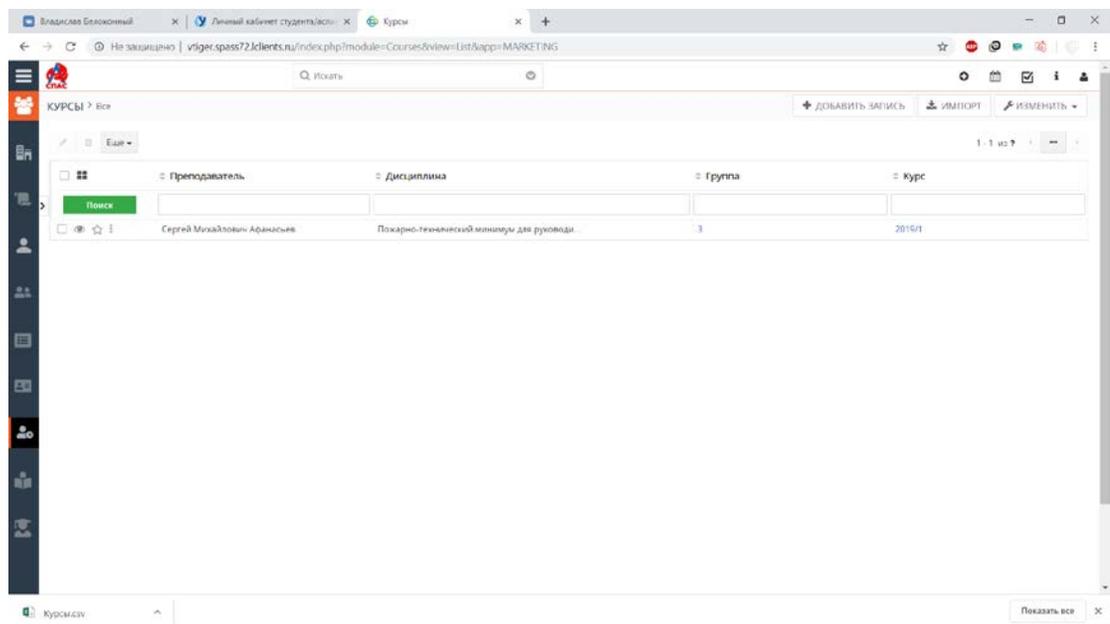


Рисунок 25 – Пример выходных данных для курсов

Дата	Курс	Вид занятия	Кол-во часов	Тема занятия
10-05-2019	2019/1	лекция	3	Требования пожарной безопасности к здани...
13-05-2019	2019/1	зачет	4	Зачет
12-05-2019	2019/1	лекция	2	Средства тушения пожаров и правила их пр...
11-05-2019	2019/1	лекция	2	Меры пожарной безопасности в детских учр...
09-05-2019	2019/1	лекция	2	Организационные мероприятия по обеспечен...
08-05-2019	2019/1	лекция	3	Основные нормативные документы, регламен...

Рисунок 26 – Пример выходных данных для занятий



Рисунок 27 – Пример выходного журнала

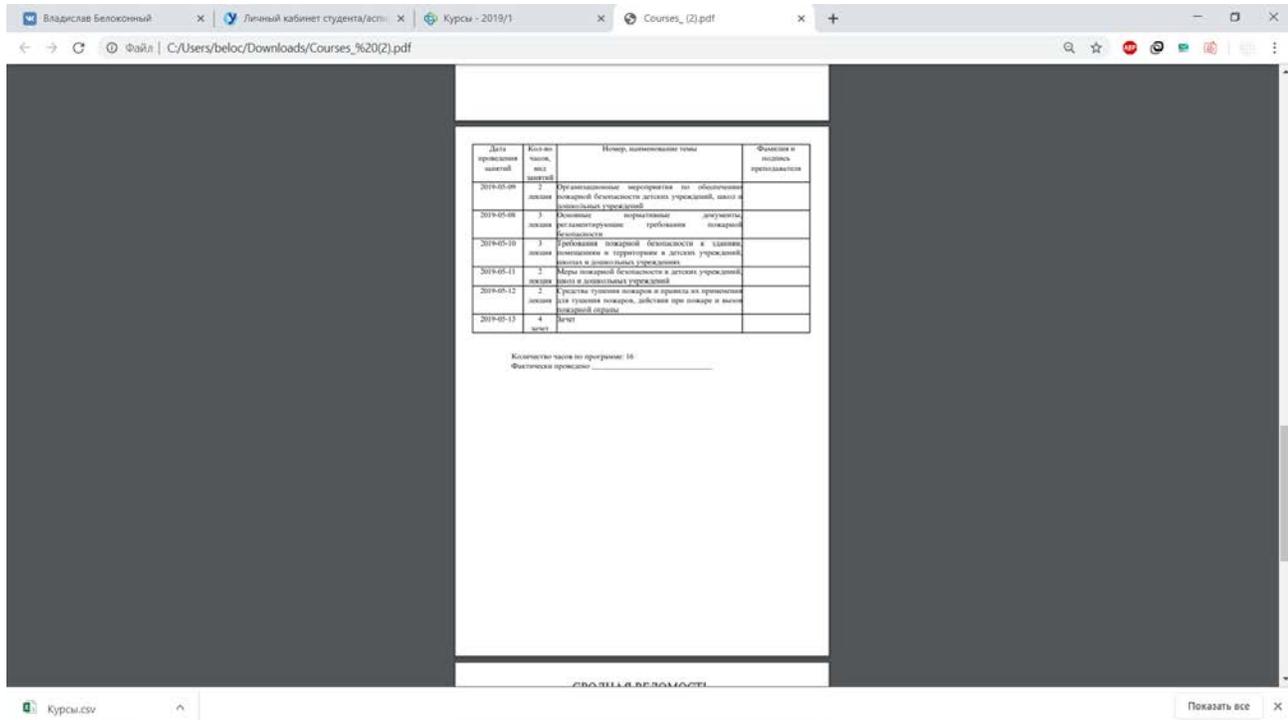


Рисунок 30 – Пример выходного журнала

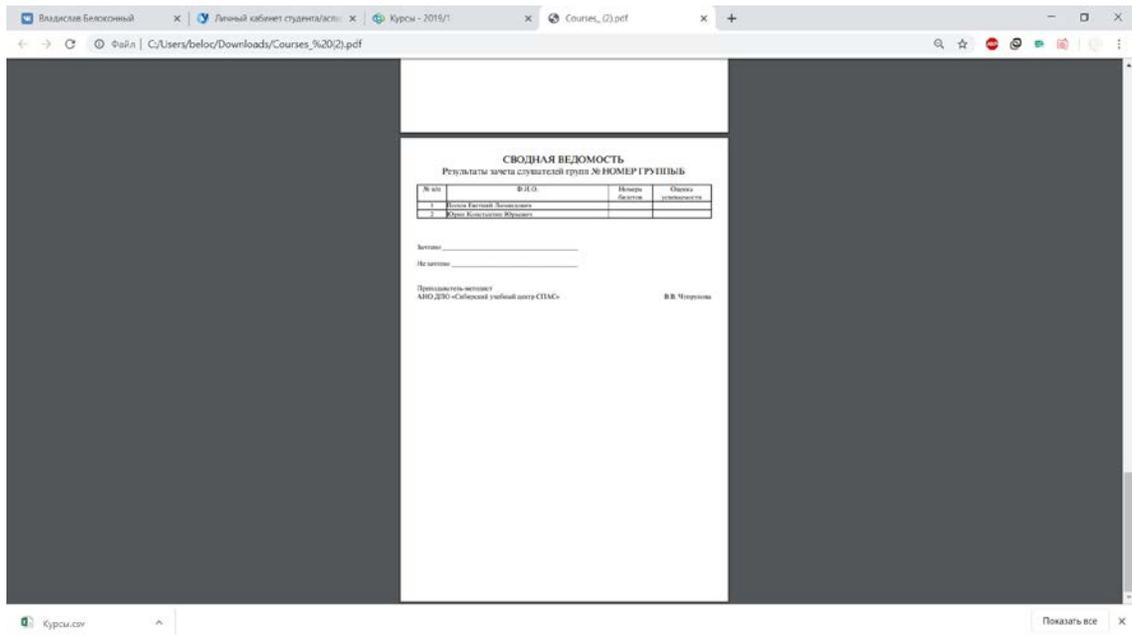


Рисунок 31 – Пример выходного журнала

5 ТЕСТИРОВАНИЕ

5.1 МЕТОДОЛОГИЯ ТЕСТИРОВАНИЯ

Для тестирования веб-приложения были использованы следующие методы:

- функциональное тестирование;
- юзабилити-тестирование;
- тестирование пользовательского интерфейса.

5.2 ПРОВЕДЕНИЕ ПРОЦЕДУРЫ ФУНКЦИОНАЛЬНОГО ТЕСТИРОВАНИЯ

Функциональное тестирование – один из ключевых видов тестирования, задача которого установить соответствие разработанного программного обеспечения исходным функциональным требованиям заказчика. То есть проведение функционального тестирования позволяет проверить способность информационной системы в определенных условиях решать задачи, нужные пользователям.

Методология проведения тестирования предполагает составление таблицы из следующих колонок:

- входные данные – точка входа для теста, обычно это описание действия, которые необходимо выполнить пользователю;
- ожидаемый результат – описание ожидаемого разработчиком внешнего поведения программы;
- полученный результат – фактическое поведение программы в ходе испытаний;
- отметка о прохождении теста – если полученный результат соответствует ожидаемому – тест считается пройденным.

Описание этапов функционального тестирования представлено на таблице

6.

Таблица 6 – Функциональное тестирование

№	Входные данные	Ожидаемый результат	Полученный результат	Тест пройден
1	Заполнение полей и регистрация пользователя	После заполнения полей и нажатия кнопки «Зарегистрировать» происходит регистрация и одновременный вход в систему. Пользователь попадает на главную страницу сайта, авторизация завершена.	После заполнения полей и нажатия кнопки переадресованы на главную страницу. Регистрация и авторизация успешны.	Да
2	Вход с верными учетными данными	После заполнения полей и нажатия кнопки «Подтвердить» происходит вход и переадресация на главную страницу.	После ввода данных и нажатия кнопки пользователь был переадресован на главную страницу. Авторизация успешна.	Да
3	Вход с неверными учетными данными	Происходит вывод сообщения о неверных данных и перезагрузка страницы входа.	Соответствует ожидаемому.	Да
4	Попытка зайти на страницу сайта без авторизации	Переадресация на страницу входа в приложение.	После перехода по ссылке на любую из страниц сайта без авторизации произошел переход на страницу ввода учетных данных.	Да
5	Изменение фамилии и email на странице настроек	После изменения данных в полях и нажатия кнопки «Сохранить» происходит перезагрузка страницы с новыми данными.	Соответствует ожиданию.	Да
6	Добавление новой записи в модули	При корректном вводе данных, запись отображается на странице.	Соответствует ожиданию.	Да

Окончание таблицы 6

№	Входные данные	Ожидаемый результат	Полученный результат	Тест пройден
7	Экспорт файла в pdf	После нажатия на кнопку «Экспорт» происходит открытие новой страницы с экспортированным pdf-файлом.	Соответствует ожиданию.	Да
8	Импорт через csv-файл	При корректном вводе данных, множество записей добавляется в базу данных и отображается на странице после её перезагрузки.	Соответствует ожиданию, после ввода данных и закрытия через доли секунды происходит перерисовка таблицы.	Да
9	Удаление записи	При нажатии на кнопку с корзиной в таблице запрашивается подтверждение удаления, после его получения производится отправка запроса и перерисовка таблицы.	Соответствует, подтверждение запрашивается, перерисовка таблицы выполняется без перерисовки остальной страницы, запись удаляется.	Да
10	Добавление задачи	После заполнения всей информации в окне добавления рецепта и нажатия на кнопку «Создать» происходит закрытие модального окна и перерисовка таблицы со новой задачей.	Соответствует ожидаемому, иногда происходит “дергание” формы с заполнением по неизвестной причине.	Да

5.3 ПРОВЕДЕНИЕ ПРОЦЕДУРЫ ЮЗАБИЛИТИ ТЕСТИРОВАНИЯ

Юзабилити-тестирование – тестирование, выполняемое с целью определения, удобна ли веб-страница или пользовательский интерфейс для его предполагаемого применения. Для этого теста были привлечены 4 сотрудника учебного центра. Тестировщикам было предложено решить следующие задачи:

- авторизоваться в системе;
- осуществить просмотр записей в модулях;
- добавить или изменить существующую запись;
- осуществить экспорт в pdf;
- осуществить поиск по пользователям;
- добавить или редактировать пользователя;
- создать и редактировать задачу;
- импортировать список студентов из csv-файла;

Все задачи были решены всеми тестировщиками без каких-либо затруднений. Недостатков не выявлено. Тест пройден успешно.

5.4 ПРОВЕДЕНИЕ ПРОЦЕДУРЫ ПОЛЬЗОВАТЕЛЬСКОГО ТЕСТИРОВАНИЯ

Интерфейс был протестирован в браузерах Microsoft Edge, Mozilla Firefox, Opera, Google Chrome. В ходе тестирования не возникло никаких проблем с версткой. Интерфейс во всех браузерах был идентичным. Тест пройден успешно.

5.5 ВЫВОД

Для тестирования системы автоматизации деятельности предприятия «Сибирский учебный центр СПАС» применялось функциональное тестирование, юзабилити-тестирование и тестирование пользовательского интерфейса. Для тестирования был разработан набор тестовых данных. Тестирование не выявило недостатков в системе.

ЗАКЛЮЧЕНИЕ

Данная работа посвящена разработке программного комплекса для автоматизации бизнес-деятельности учебного центра. В ходе проделанной работы мной были решены следующие задачи:

- изучены требования Регламента работы автономной некоммерческой организации дополнительного профессионального образования «Сибирский учебный центр СПАС».
- изучены существующие подходы, применяющиеся при построении информационных систем, а частности веб-приложений, был выбран наиболее оптимальный из них;
- на основе выбранной технологии реализована система автоматизации деятельности учебного центра. В процессе работы были изучены возможности работы с платформой vTiger, языком программирования Php, системой управления базами данных MySQL, ОС Linux;
- произведено тестирование системы с использованием тестового набора пользовательских данных. Система успешно прошла предложенные тесты.

Разработанное веб-приложение внедрено в эксплуатацию автономной некоммерческой организации дополнительного профессионального образования «Сибирский учебный центр СПАС» и используется для распределения задач между сотрудниками, быстрого составления и экспорта документов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Кинзябулатов, Р. CRM. Подробно и по делу.: справочник. / Р. Кинзябулатов. – М.: Издательские решения, 2009. – 430 с.
2. Ли, К. Создание клиентской базы. Пошаговое руководство.: учебное пособие / К. Ли. – М.: Недра, 2006. – 367 с.
3. Кудринов, А. CRM практика эффективного бизнеса.: справ. пособие / А. Кудринов, М. Сорокин, Е.Голышева. – М.: 1С-Публишинг, 2002. – 411 с.
4. Дише, Д. CRM-навигатор.: учебное пособие / Д. Дише. – М.: Издательство Алексея Капусты, 2006. – 534 с.
5. Трофимов, С. CRM для практиков.: справочник / С. Трофимов. – СПб.: АвтоКод, 2006. – 610 с.
6. Браун, П. Клиенты на всю жизнь.: учебное пособие / П. Браун, К. Сьюэлл. – М.: Манн, 2013. – 641с.
7. Шапошников, И. Web-сервисы Microsoft .NET.: справочник / И. Шапошников. – СПб.: БХВ-Петербург, 2002. – 199с.
8. Янг, А. Node.js в действии.: учебное пособие / А. Янг, М. Брэндли, М. Кантелеон. – СПб.: «Питер», 2009. – 519с.
9. Чан, У. Django. Разработка веб-приложений.: учебник / У. Чан, П. Биссекс, Д. Форсье. – СПб.: Символ-плюс, 2013. – 441с.
10. Heartl, M. Ruby on Rails Tutorial.: учебное пособие / M. Heartl. – New York.: Addison-Wesley, 2015. – 57 p.
11. Фейерштейн, С. Oracle PL/SQL. Для профессионалов.: руководство / С. Фейерштейн, Б. Прибыл. – СПб.: «Питер», 2015. – 1024 с.
12. Веллинг, Л. PHP и MySQL Веб-разработка: учебник / Л.

Веллинг, Е. Томсон. – СПб.: БХВ-Петербург, 2010. – 848 с.

13. Шкляр, Л. Архитектура веб-приложений: учебник / Л. Шкляр, Р. Рич. – М.: Эксмо, 2011. – 978 с.

14. Ташков, П. Веб-мастеринг на 100%: HTML, CSS, JavaScript, PHP, CMS, AJAX, графика, раскрутка.: справ. пособие / П. Ташков – СПб.: «Питер», 2010. – 512 с.

15. Beak, A. PHP 7 Zend Certification.: руководство / А. Beak – New York.: «Apress», 2017. – 294 р.

16. Белоконный, В.Д. Исходный код «Разработка программного комплекса для поддержки бизнес-деятельности учебного центра» / <https://github.com/Bvdldf/ABP> (дата обращения 05.06.2019)

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД НЕКОТОРЫХ МОДУЛЕЙ

```
//CRMentity
<?php
class vCRMEntity extends CRMEntity {
var $db, $log;
var $column_fields = Array();
var $IsCustomModule = true;
through initSortFields
var $sortby_fields = Array();
var $required_fields = Array ('assigned_user_id'=>1);
var $special_functions = Array('set_import_assigned_user');
function __construct() {
global $log;
$this->column_fields = getColumnFields(get_class($this));
$this->db = new PearDatabase();
$this->log = $log;
}
function save_module($module) {
}
function getQueryByModuleField($module, $fieldname, $srcrecord) {
}
/**
*/
function getListQuery($module, $where='') {
$query = "SELECT crmentity.*, $this->table_name.*";
if(!empty($this->customFieldTable)) $query .= ", " . $this-
>customFieldTable[0] . ".* ";
$query .= " FROM $this->table_name";
$query .= " INNER JOIN _crmentity ON _crmentity.crmid = $this-
>table_name.$this->table_index";
if(!empty($this->customFieldTable)) {
$query .= " INNER JOIN ".$this->customFieldTable[0]. " ON ".$this-
>customFieldTable[0].".$this->customFieldTable[1] .
" = $this->table_name.$this->table_index";
}
$query .= " LEFT JOIN _users ON _users.id = _crmentity.smownerid";
$query .= " LEFT JOIN _groups ON _groups.groupid = _crmentity.smownerid";
```

```

        $linkedModulesQuery = $this->db->pquery("SELECT distinct fieldname,
columnname, relmodule FROM _field" .
        " INNER JOIN _fieldmodulerelel ON _fieldmodulerelel.fieldid = _field.fieldid" .
        " WHERE uitype='10' AND _fieldmodulerelel.module=?", array($module));
        $linkedFieldsCount = $this->db->num_rows($linkedModulesQuery);
        for($i=0; $i<$linkedFieldsCount; $i++) {
            $related_module = $this->db->query_result($linkedModulesQuery, $i,
'relmodule');
            $fieldname = $this->db->query_result($linkedModulesQuery, $i, 'fieldname');
            $columnname = $this->db->query_result($linkedModulesQuery, $i, 'columnname');
                $other = CRMEntity::getInstance($related_module);
            vtlib_setup_modulevars($related_module, $other);
            $query .= " LEFT JOIN $other->table_name ON $other->table_name.$other-
>table_index =".
            "$this->table_name.$columnname";
        }
        global $current_user;
        $query .= $this->getNonAdminAccessControlQuery($module,$current_user);
        $query .= "WHERE _crmentity.deleted = 0 ".$where;
        return $query;
    }
    /**
    */
    function getListViewSecurityParameter($module) {
        global $current_user;
        require('user_privileges/user_privileges_'.$current_user->id.'.php');
        require('user_privileges/sharing_privileges_'.$current_user->id.'.php');
        $sec_query = '';
        $stabid = getTabid($module);
        if($is_admin==false && $profileGlobalPermission[1] == 1 &&
$profileGlobalPermission[2] == 1
        && $defaultOrgSharingPermission[$stabid] == 3) {
            $sec_query .= " AND (_crmentity.smownerid in($current_user->id) OR
_crmentity.smownerid IN
            (
            SELECT _user2role.userid FROM _user2role
            INNER JOIN _users ON _users.id= _user2role.userid
            INNER JOIN _role ON _role.roleid= _user2role.roleid
            WHERE _role.parentrole LIKE '".$current_user_parent_role_seq.':::%'
            )
        )

```

```

OR _crmentity.smownerid IN
(
SELECT shareduserid FROM _tmp_read_user_sharing_per
WHERE userid=".$current_user->id." AND tabid=".$tabid."
)
OR
(";
if (sizeof($current_user_groups) > 0) {
$sec_query .= " _crmentity.smownerid IN (". implode(",", $current_user_groups)
.) OR ";
}
$sec_query .= " _crmentity.smownerid IN
(
SELECT _tmp_read_group_sharing_per.sharedgroupid
FROM _tmp_read_group_sharing_per
WHERE userid=".$current_user->id." and tabid=".$tabid."
)";
$sec_query .= " )
)";
}
return $sec_query;
}
/**
*/
function create_export_query($where)
{
global $current_user,$currentModule;
include("include/utils/ExportUtils.php");
$sql = getPermittedFieldsQuery('ServiceContracts', "detail_view");

$fields_list = getFieldsListFromQuery($sql);
$query = "SELECT $fields_list, _users.user_name AS user_name
FROM _crmentity INNER JOIN $this->table_name ON _crmentity.crmid=$this-
>table_name.$this->table_index";
if(!empty($this->customFieldTable)) {
$query .= " INNER JOIN ".$this->customFieldTable[0]." ON ".$this-
>customFieldTable[0].".$this->customFieldTable[1] .
" = $this->table_name.$this->table_index";
}
$query .= " LEFT JOIN _groups ON _groups.groupid = _crmentity.smownerid";

```

```

$query .= " LEFT JOIN _users ON _crmentity.smownerid = _users.id and ".
" _users.status='Active'";
$linkedExceptionQuery = $this->db->pquery("SELECT distinct fieldname,
columnname, relmodule FROM _field" .
" INNER JOIN _fieldmodulerelel ON _fieldmodulerelel.fieldid = _field.fieldid" .
" WHERE uitype='10' AND _fieldmodulerelel.module=?", array($thismodule));
$linkedExceptionCount = $this->db->num_rows($linkedExceptionQuery);
for($i=0; $i<$linkedExceptionCount; $i++) {
    $related_module = $this->db->query_result($linkedExceptionQuery, $i,
'relmodule');
    $fieldname = $this->db->query_result($linkedExceptionQuery, $i, 'fieldname');
    $columnname = $this->db->query_result($linkedExceptionQuery, $i, 'columnname');
    $other = CRMEntity::getInstance($related_module);
    vtlib_setup_modulevars($related_module, $other);
    $query .= " LEFT JOIN $other->table_name ON $other->table_name.$other-
>table_index = ".
"$this->table_name.$columnname";
}
$query .= $this->getNonAdminAccessControlQuery($thismodule,$current_user);
$where_auto = " _crmentity.deleted=0";
if($where != '') $query .= " WHERE ($where) AND $where_auto";
else $query .= " WHERE $where_auto";
return $query;
}
/**
*/
function
getDuplicatesQuery($module,$table_cols,$field_values,$ui_type_arr,$select_cols='') {
    $select_clause = "SELECT ". $this->table_name .".".$this->table_index ." AS
recordid, _users_last_import.deleted,".$table_cols;
    if(isset($this->customFieldTable)) $query .= ", " . $this->customFieldTable[0]
. ".* ";
    $from_clause = " FROM $this->table_name";
    from_clause .= " INNER JOIN _crmentity ON _crmentity.crmid = $this-
>table_name.$this->table_index";

    if(isset($this->customFieldTable)) {
        $from_clause .= " INNER JOIN ".$this->customFieldTable[0]." ON ".$this-
>customFieldTable[0].".$this->customFieldTable[1] .
" = $this->table_name.$this->table_index";
    }
}

```

```

}
$from_clause .= " LEFT JOIN _users ON _users.id = _crmentity.smownerid
LEFT JOIN _groups ON _groups.groupid = _crmentity.smownerid";
$where_clause = " WHERE _crmentity.deleted = 0";
$where_clause .= $this->getListViewSecurityParameter($module);
if (isset($select_cols) && trim($select_cols) != '') {
$sub_query = "SELECT $select_cols FROM $this->table_name AS t " .
" INNER JOIN _crmentity AS crm ON crm.crmid = t.".$this->table_index;
if(isset($this->customFieldTable)) {
$sub_query .= " INNER JOIN ".$this->customFieldTable[0]." tcf ON tcf.".$this->
>customFieldTable[1]." = t.$this->table_index";
}
$sub_query .= " WHERE crm.deleted=0 GROUP BY $select_cols HAVING COUNT(*)>1";
} else {
$sub_query = "SELECT $table_cols $from_clause $where_clause GROUP BY
$table_cols HAVING COUNT(*)>1";
}
$query = $select_clause . $from_clause .
" LEFT JOIN _users_last_import ON _users_last_import.bean_id=" . $this->
>table_name .".".$this->table_index .
" INNER JOIN ( " . $sub_query . " ) AS temp ON
".get_on_clause($field_values,$ui_type_arr,$module) .
$where_clause .
" ORDER BY $table_cols, ". $this->table_name .".".$this->table_index ." ASC";
return $query;
}
}
//orders (сkeleton)
include_once 'modules/CRMEntity.php';
class Orders extends CRMEntity {
var $table_name = _orders';
var $table_index= 'ordersid';
var $customFieldTable = Array(_orderscf', 'ordersid');
var $tab_name = Array(_crmentity', _orders', _orderscf');
var $tab_name_index = Array(
_crmentity' => 'crmid',
'_orders' => 'ordersid',
'_orderscf'=>'ordersid');
var $entity_table = " _crmentity";
var $list_fields = Array (

```

```

'Order type' => Array('_orders', 'order_type'),
'Order date' => Array('_orders', 'order_date'),
'Order number' => Array('_orders', 'order_no'),
//      'Group number' => Array('_orders', 'studentgroup_id'),
'Curator' => Array('_orders', 'curator_id'),
'Certification type' => Array('_orders', 'certification_type'),
'Program' => Array('_orders', 'program'),
'Start date' => Array('_orders', 'start_date'),
'End date' => Array('_orders', 'end_date'),
);
var $list_fields_name = Array (
'Order type' => 'order_type',
'Order date' => 'order_date',
'Order number' => 'order_no',
//      'Group number' => 'studentgroup_id',
'Curator' => 'curator_id',
'Certification type' => 'certification_type',
'Program' => 'program',
'Start date' => 'start_date',
'End date' => 'end_date',
);

var $list_link_field = 'ordertype';
var $search_fields = Array(
'Order type' => Array('orders', 'order_type'),
'Order date' => Array('orders', 'order_date'),
'Order number' => Array('orders', 'order_no'),
//      'Group number' => Array('orders', 'studentgroup_id'),
'Curator' => Array('orders', 'curator_id'),
'Certification type' => Array('orders', 'certification_type'),
'Program' => Array('orders', 'program'),
'Start date' => Array('orders', 'start_date'),
'End date' => Array('orders', 'end_date'),
);
var $search_fields_name = Array (
'Order type' => 'order_type',
'Order date' => 'order_date',
'Order number' => 'order_no',
//      'Group number' => 'studentgroup_id',

```

```

'Curator' => 'curator_id',
'Certification type' => 'certification_type',
'Program' => 'program',
'Start date' => 'start_date',
'End date' => 'end_date',
);
var $popup_fields = Array ('order_type');
var $def_basicsearch_col = 'order_type';
var $def_detailview_recname = 'order_type';
var $mandatory_fields = Array('order_type');
var $default_order_by = 'order_type';
var $default_sort_order='ASC';

//orders (занесение)
<?php
include_once 'include/Webservices/Relation.php';
include_once 'includes/main/WebUI.php';
include_once 'vtlib/Module.php';
include_once 'vtlib/Block.php';
include_once 'vtlib/Field.php';
$_Utils_Log = true;
$MODULENAME = 'Orders';
$moduleInstance = _Module::getInstance($MODULENAME);
if ($moduleInstance) {
    echo "Module $MODULENAME already present - choose a different name.";
    $moduleInstance->delete();
} else {
    $moduleInstance = new _Module();
    $moduleInstance->name = $MODULENAME;
    $moduleInstance->parent= 'Marketing';
    $moduleInstance->save();
    $moduleInstance->initTables();
    $block = new _Block();
    $block->label = 'LBL_' . strtoupper($moduleInstance->name) . '_INFORMATION';
    $moduleInstance->addBlock($block);
    $blockcf = new _Block();
    $blockcf->label = 'LBL_CUSTOM_INFORMATION';
    $moduleInstance->addBlock($blockcf);
    $field0 = new _Field();
    $field0->name = 'order_type';

```

```

$field0->label = 'Order type';
$field0->uitype = 16;
$field0->column = $field0->name;
$field0->columntype = 'VARCHAR(255)';
$field0->typeofdata = 'V~M';
$field0->displaytype = 1;
$field0->setPicklistValues(Array());
$block->addField($field0);
echo "Created $field0->name \n";
    $field1 = new _Field();
    $field1->name = 'order_date';
    $field1->label = 'Order date';
    $field1->uitype = 5;
    $field1->column = $field1->name;
    $field1->columntype = 'date';
    $field1->typeofdata = 'D~M';
    $block->addField($field1);
echo "Created $field1->name \n";
    $field2 = new _Field();
    $field2->name = 'order_no';
    $field2->label= 'Order number';
    $field2->uitype= 2;
    $field2->column = $field2->name;
    $field2->columntype = 'VARCHAR(255)';
    $field2->typeofdata = 'V~M';
    $field2->displaytype = 1;
    $block->addField($field2);
echo "Created $field2->name \n";
        $moduleInstance->setEntityIdentifier($field2);
        $field4 = new _Field();
    $field4->name = 'curator_id';
    $field4->label= 'Curator';
    $field4->uitype= 10;
    $field4->column = $field4->name;
    $field4->columntype = 'int(19)';
    $field4->typeofdata = 'I~M';
    $field4->displaytype = 1;
    $block->addField($field4);
    $field4->setRelatedModules(Array('Users'));
echo "Created $field4->name \n";

```

```

        $field5 = new _Field();
$field5->name = 'certification_type';
$field5->label = 'Certification type';
$field5->uitype = 16;
$field5->column = $field5->name;
$field5->coluntype = 'VARCHAR(255)';
$field5->typeofdata = 'V~O';
$field5->displaytype = 1;
$field5->setPicklistValues(Array(''));
$block->addField($field5);
echo "Created $field5->name \n";

        $field6 = new _Field();
$field6->name = 'program';
$field6->label = 'Program';
$field6->uitype = 1;
$field6->column = $field6->name;
$field6->columnntype = 'VARCHAR(255)';
$field6->typeofdata = 'V~O';
$block->addField($field6);
echo "Created $field6->name \n";

        $field7 = new _Field();
$field7->name = 'start_date';
$field7->label = 'Start date';
$field7->uitype = 5;
$field7->column = $field7->name;
$field7->columnntype = 'date';
$field7->typeofdata = 'D~O';
$block->addField($field7);
echo "Created $field7->name \n";

        $field8 = new _Field();
$field8->name = 'end_date';
$field8->label = 'End date';
$field8->uitype = 5;
$field8->column = $field8->name;
$field8->columnntype = 'date';
$field8->typeofdata = 'D~O';
$block->addField($field8);
echo "Created $field8->name \n";

        $field9 = new _Field();
$field9->name = 'studentgroup_id';

```

```

$field9->label= 'Group number';
$field9->uitype= 10;
$field9->column = $field9->name;
$field9->columnType = 'int(19)';
$field9->typeofdata = 'I~M';
$field9->displaytype = 1;
$block->addField($field9);
$field9->setRelatedModules(Array('StudentGroups'));
echo "Created $field9->name \n";
    $field99 = new _Field();
    $field99->name = 'description';
    $field99->label= 'Description';
    $field99->uitype = 19;
    $field99->column = 'description';
    $field99->table = '_crmentity';
    $blockcf->addField($field99);
echo "Created $field99->name \n";
    $filter1 = new _Filter();
    $filter1->name = 'All';
    $filter1->isdefault = true;
    $moduleInstance->addFilter($filter1);
// $filter1->addField($field0)->addField($field1, 1)->addField($field2, 2)-
>addField($field3, 3)->addField($field4, 4)->addField($field5, 5);
    $filter1->addField($field0)->addField($field1, 1)->addField($field2, 2)-
>addField($field4, 3)->addField($field5, 4);
    echo "Created $filter1->name \n";
    $moduleInstance->setDefaultSharing();
    $moduleInstance->initWebservice();
    Settings_MenuEditor_Module_Model::addModuleToApp($MODULENAME, 'MARKETING');
    echo "OK\n";
}
global $adb;
$result = $adb->pquery("update _tab set source=null where name=?",
Array($MODULENAME));
$moduleInstance = Module::getInstance($MODULENAME);
if ($moduleInstance) {
    $relModuleName = 'Contacts';
    $relModuleInstance = Module::getInstance($relModuleName);
    if ($relModuleInstance){

```

```

        $moduleInstance->setRelatedList($relModuleInstance,      'Sended      down',
Array('ADD', 'SELECT'));
    }

    echo "OK\n";
}

// PDFController
include_once 'include/InventoryPDFController.php';

class OrdersPDFController extends InventoryPDFController{
function buildHeaderModelTitle() {
    $singularModuleNameKey = 'SINGLE_'. $this->moduleName;
    $translatedSingularModuleLabel = getTranslatedString($singularModuleNameKey,
$this->moduleName);
    if($translatedSingularModuleLabel == $singularModuleNameKey) {
        $translatedSingularModuleLabel = getTranslatedString($this->moduleName, $this-
>moduleName);
    }
    return sprintf("%s: %s", $translatedSingularModuleLabel, $this-
>focusColumnValue('invoice_no'));
}
function buildHeaderModelColumnCenter() {
    $customerName = $this->resolveReferenceLabel($this-
>focusColumnValue('account_id'), 'Accounts');
    $contactName = $this->resolveReferenceLabel($this-
>focusColumnValue('contact_id'), 'Contacts');
    $purchaseOrder = $this->focusColumnValue('_purchaseorder');
    $salesOrder = $this->resolveReferenceLabel($this-
>focusColumnValue('salesorder_id'));
    $customerNameLabel = getTranslatedString('Customer Name', $this->moduleName);
    $contactNameLabel = getTranslatedString('Contact Name', $this->moduleName);
    $purchaseOrderLabel = getTranslatedString('Purchase Order', $this-
>moduleName);
    $salesOrderLabel = getTranslatedString('Sales Order', $this->moduleName);
    $modelColumnCenter = array(
    $customerNameLabel =>    $customerName,
    $purchaseOrderLabel =>    $purchaseOrder,
    $contactNameLabel =>    $contactName,
    $salesOrderLabel =>    $salesOrder
    );
}
}

```

```

return $modelColumnCenter;
}

function buildHeaderModelColumnRight() {
    $issueDateLabel = getTranslatedString('Issued Date', $this->moduleName);
    $validDateLabel = getTranslatedString('Due Date', $this->moduleName);
    $billingAddressLabel = getTranslatedString('Billing Address', $this->
>moduleName);
    $shippingAddressLabel = getTranslatedString('Shipping Address', $this->
>moduleName);

    $modelColumnRight = array(
        'dates' => array(
            $issueDateLabel => $this->formatDate(date("Y-m-d")),
            $validDateLabel => $this->formatDate($this->focusColumnValue('duedate')),
        ),
        $billingAddressLabel => $this->buildHeaderBillingAddress(),
        $shippingAddressLabel => $this->buildHeaderShippingAddress()
    );
    return $modelColumnRight;
}

function getWatermarkContent() {
    return $this->focusColumnValue('invoicestatus');
}
}

include_once 'includes/PDF/ProductListPDFController.php';
require_once 'modules/Order.php';
require_once 'modules/Accounts/Accounts.php';
require_once 'modules/Contacts/Contacts.php';

class OrdersPDFController extends PDF_ProductListDocumentPDFController{

    function buildCuratorsFIO($curator_id){
        $curatorInstance = Users_Record_Model::getInstanceById($curator_id, 'Users');
        return $curatorInstance->get('last_name') . ' ' . $curatorInstance->
>get('first_name');
    }
}

```

```

function buildAudience($order_id) {
    $TEMPLATE_FILE = "modules/Orders/templates/audience.tpl";
    $smarty = new Smarty;
    $items = $this->getAudience($order_id);
    $smarty->assign("items", $items);
    return $smarty->fetch($TEMPLATE_FILE);
}

function getAudience($order_id){
    global $adb;

    $query = 'select cd.contactid from _crmentityrel cer inner join
    _contactdetails cd on cd.contactid=cer.relcrmid where crmid in (select
    studentgroupsid from _studentgroups where order_id=?) and module=? and relmodule=?';

    $query = 'select cd.contactid from _contactdetails cd where cd.contactid in
    (select relcrmid from _crmentityrel where module=? and relmodule=? and crmid=(select
    studentgroup_id from _orders where ordersid=?))';

    $result = $adb->pquery($query, Array('StudentGroups', 'Contacts', $order_id));

    $items = Array();

    while ($row = $adb->fetchByAssoc($result)) {
        $items[] = _Record_Model::getInstanceById($row['contactid'], 'Contacts');
    }

    return $items;
}

function getSendedDown($order_id){
    global $adb;

    $query = 'select cd.contactid from _crmentityrel cer inner join
    _contactdetails cd on cd.contactid=cer.relcrmid where crmid=? and module=? and
    relmodule=? limit 1';

    $result = $adb->pquery($query, Array($order_id, 'Orders', 'Contacts'));

    $items = Array();

```

```

while ($row = $adb->fetchByAssoc($result)) {
    $items[] = _Record_Model::getInstanceById($row['contactid'], 'Contacts');
}

return ($items ? $items[0] : null);
}

function getGroupNo($order_id){
    global $adb;

    $query = 'select group_no from _studentgroups where order_id=? limit 1';

    $result = $adb->pquery($query, Array($order_id));

    $items = Array();

    while ($row = $adb->fetchByAssoc($result)) {
        $items[] = $row['group_no'];
    }

    return $items[0];
}

function buildBody(){
    $TEMPLATE_FILE = "modules/Orders/templates/body.tpl";
    $smarty = new Smarty;
    return $smarty->fetch($TEMPLATE_FILE);
}

function buildDocumentModel() {
    global $app_strings, $current_user;
    try {
        $model = parent::buildDocumentModel();
        $this->generateEntityModel($this->focus, 'Orders', 'orders_', $model);
        $entity = new Orders();
        $entity->retrieve_entity_info($this->focus->id, 'Orders');
        if(!empty($entity)) {
            $TEMPLATE_BODY = "modules/Orders/templates/body.tpl";
            $TEMPLATE_PART1 = "modules/Orders/templates/part1.tpl";
            $TEMPLATE_PART2 = "modules/Orders/templates/part2.tpl";

```

```

$TEMPLATE_AUDIENCE = "modules/Orders/templates/audience.tpl";
$smarty = new Smarty;
$smarty->assign('orgName', $model->get('orgName'));
$smarty->assign('orgAddress', $model->get('orgAddress'));
$smarty->assign('orgLogoPath', $model->get('orgLogoPath'));
$smarty->assign('orgCity', $model->get('orgCity'));
$smarty->assign('order_date', $this->russianDate($entity-
>column_fields['order_date']));
    $smarty->assign('start_date', $this->russianDate($entity-
>column_fields['start_date']));
    $smarty->assign('end_date', $this->russianDate($entity-
>column_fields['end_date']));
    $smarty->assign('order_no', $entity->column_fields['order_no']);
    $smarty->assign('studentgroup_id', _Record_Model::getInstanceById($entity-
>column_fields['studentgroup_id']), 'StudentGroups');
    $smarty->assign('order_type', $entity->column_fields['order_type']);
    $smarty->assign('certification_type', $entity-
>column_fields['certification_type']);
    $smarty->assign('program', $entity->column_fields['program']);
    $smarty->assign('curator', Users_Record_Model::getInstanceById($entity-
>column_fields['curator_id'], 'Users'));
    $smarty->assign('orgDirector', $model->get('orgDirector'));
    $smarty->assign('items', $this->getAudience($this->focus->id));
    $smarty->assign('sended_down', $this->getSendedDown($this->focus->id));
    $model->set('director_id', 18);
    $model->set('part1', $smarty->fetch($TEMPLATE_PART1));
    $model->set('part2', $smarty->fetch($TEMPLATE_PART2));
}
return $model;
} catch (Exception $e) {
echo '<meta charset="utf-8" />';
if($e->getMessage() == $app_strings['LBL_RECORD_DELETE']) {
echo $app_strings['LBL_RECORD_INCORRECT'];
echo '<br><br>';
} else {
echo $e->getMessage();
echo '<br><br>';
}
return null;    }
$_Utils_Log = true;

```

```

$MODULENAME = 'Lessons';

$moduleInstance = _Module::getInstance($MODULENAME);
if ($moduleInstance) {
    echo "Module $MODULENAME already present - choose a different name.";
    $moduleInstance->delete();
} else {
    $moduleInstance = new _Module();
    $moduleInstance->name = $MODULENAME;
    $moduleInstance->parent= 'Marketing';
    $moduleInstance->save();

    $moduleInstance->initTables();

    $block = new _Block();
    $block->label = 'LBL_' . strtoupper($moduleInstance->name) .
'_INFORMATION' ;
    $moduleInstance->addBlock($block);

    $blockcf = new _Block();
    $blockcf->label = 'LBL_CUSTOM_INFORMATION' ;
    $moduleInstance->addBlock($blockcf);

    $field1 = new _Field();
    $field1->name = 'lesson_date';
    $field1->label = 'Lesson date';
    $field1->uitype = 5;
    $field1->column = $field1->name;
    $field1->columntype = 'date';
    $field1->typeofdata = 'D-M';
    $block->addField($field1);
    echo "Created $field1->name \n";

    $moduleInstance->setEntityIdentifier($field1);

    $field2 = new _Field();
    $field2->name = 'course_id';

```

```

$field2->label= 'Course';
$field2->uitype= 10;
$field2->column = $field2->name;
$field2->columntype = 'int(19)';
$field2->typeofdata = 'I~M';
$field2->displaytype = 1;
$block->addField($field2);
$field2->setRelatedModules(Array('Courses'));
echo "Created $field2->name \n";

$field3 = new _Field();
$field3->name = 'lesson_type';
$field3->label = 'Lesson type';
$field3->uitype = 16;
$field3->column = $field3->name;
$field3->columntype = 'VARCHAR(255)';
$field3->typeofdata = 'V~M';
$field3->displaytype = 1;
$field3->setPicklistValues(Array('P»PuPeC+PëC¶', 'PiC¶P°PeC, PëPeP°'));
$block->addField($field3);
echo "Created $field3->name \n";

$field4 = new _Field();
$field4->name = 'lesson_hours';
$field4->label = 'Lesson hours';
$field4->uitype = 7;
$field4->column = $field4->name;
$field4->columntype = 'int(4)';
$field4->typeofdata = 'N~M';
$field4->displaytype = 1;
$block->addField($field4);
echo "Created $field4->name \n";

$field5 = new _Field();
$field5->name = 'lesson_topic';
$field5->label = 'Lesson topic';
$field5->uitype = 2;
$field5->column = $field5->name;
$field5->columntype = 'VARCHAR(255)';
$field5->typeofdata = 'V~M';

```

```

$field5->displaytype = 1;
$block->addField($field5);
echo "Created $field5->name \n";

$field99 = new _Field();
$field99->name = 'description';
$field99->label= 'Description';
$field99->uitype= 19;
$field99->column = 'description';
$field99->table = '_crmentity';
$blockcf->addField($field99);
echo "Created $field99->name \n";

$filter1 = new _Filter();
$filter1->name = 'All';
$filter1->isdefault = true;
$moduleInstance->addFilter($filter1);
$filter1->addField($field1)->addField($field2, 1);
echo "Created $filter1->name \n";

$moduleInstance->setDefaultSharing();

$moduleInstance->initWebservice();

Settings_MenuEditor_Module_Model::addModuleToApp($MODULENAME,
'MARKETING');

echo "OK\n";
}

global $adb;

$result = $adb->pquery("update _tab set source=null where name=?",
Array($MODULENAME));

$focus= _Module::getInstance($MODULENAME);

```

```

if (!$focus) {
    echo "Module '".MODULENAME.'" not found. <br />\n";
    return;
}
include('version.php');

$HELPDESK_SUPPORT_EMAIL_ID = 'admin@admin.ru';
$HELPDESK_SUPPORT_NAME = 'your-support name';
$HELPDESK_SUPPORT_EMAIL_REPLY_ID = $HELPDESK_SUPPORT_EMAIL_ID;

/* database configuration
    db_server
    db_port
    db_hostname
    db_username
    db_password
    db_name
*/

$dbconfig['db_server'] = '127.0.0.1';
$dbconfig['db_port'] = ':3306';
$dbconfig['db_username'] = 'spas';
$dbconfig['db_password'] = 'spas';
$dbconfig['db_name'] = 'spas';
$dbconfig['db_type'] = 'mysqli';
$dbconfig['db_status'] = 'true';
$dbconfig['db_hostname'] = $dbconfig['db_server'].$dbconfig['db_port'];

// log_sql default value = false
$dbconfig['log_sql'] = false;

// persistent default value = true
$dbconfigoption['persistent'] = true;

// autofree default value = false
$dbconfigoption['autofree'] = false;

// debug default value = 0
$dbconfigoption['debug'] = 0;

```

```

// seqname_format default value = '%s_seq'
$dbconfigoption['seqname_format'] = '%s_seq';

// portability default value = 0
$dbconfigoption['portability'] = 0;

// ssl default value = false
$dbconfigoption['ssl'] = false;

$host_name = $dbconfig['db_hostname'];

$site_URL = 'http://spas.o-it-s.ru/crm/';
#$site_URL = 'http://spass72/spas/';

// url for customer portal (Example: http://.com/portal)
$PORTAL_URL = $site_URL.'/customerportal';
// root directory path
$root_directory = '/var/www/html/crm';
#$root_directory = '/var/www/html/spas/';

// cache direcorey path
$cache_dir = 'cache/';

// tmp_dir default value prepended by cache_dir = images/
$tmp_dir = 'cache/images/';

// import_dir default value prepended by cache_dir = import/
$import_dir = 'cache/import/';

// upload_dir default value prepended by cache_dir = upload/
$upload_dir = 'cache/upload/';

// maximum file size for uploaded files in bytes also used when uploading
import files
// upload_maxsize default value = 3000000
$upload_maxsize = 3145728;//3MB

// flag to allow export functionality
// 'all' to allow anyone to use exports
// 'admin' to only allow admins to export

```

```

// 'none' to block exports completely
// allow_exports default value = all
$allow_exports = 'all';

// files with one of these extensions will have '.txt' appended to their
filename on upload
// upload_badext default value = php, php3, php4, php5, pl, cgi, py, asp, cfm,
js, vbs, html, htm
$upload_badext = array('php', 'php3', 'php4', 'php5', 'pl', 'cgi', 'py',
'asp', 'cfm', 'js', 'vbs', 'html', 'htm', 'exe', 'bin', 'bat', 'sh', 'dll', 'phps',
'phtml', 'xhtml', 'rb', 'msi', 'jsp', 'shtml', 'sth', 'shtm');

// list_max_entries_per_page default value = 20
$list_max_entries_per_page = '20';

// history_max_viewed default value = 5
$history_max_viewed = '5';

// default_module default value = Home
$default_module = 'Home';

// default_action default value = index
$default_action = 'index';

// set default theme
// default_theme default value = blue
$default_theme = 'softed';

// default text that is placed initially in the login form for user name
// no default_user_name default value
$default_user_name = '';

// default text that is placed initially in the login form for password
// no default_password default value
$default_password = '';

// create user with default username and password
// create_default_user default value = false
$create_default_user = false;

```

```

//Master currency name
$currency_name = 'Russia, Rubles';

// default charset
// default charset default value = 'UTF-8' or 'ISO-8859-1'
$default_charset = 'UTF-8';

// default language
// default_language default value = en_us
$default_language = 'ru_ru';

//Option to hide empty home blocks if no entries.
$display_empty_home_blocks = false;

//Disable Stat Tracking of CRM instance
$disable_stats_tracking = false;

// Generating Unique Application Key
$application_unique_key = 'dc66d1a8b837fd5771a6ce148834c0a6';

// trim descriptions, titles in listviews to this value
$listview_max_textlength = 40;

// Maximum time limit for PHP script execution (in seconds)
$php_max_execution_time = 0;

$default_timezone = 'Europe/Moscow';
/** If timezone is configured, try to set it */
if(isset($default_timezone) && function_exists('date_default_timezone_set')) {
    @date_default_timezone_set($default_timezone);
}
include('config.inc.php');

$THIS_DIR = dirname(__FILE__);

if (file_exists($THIS_DIR.'/config_override.php')) {
    include_once $THIS_DIR.'/config_override.php';
}

class Config {

```

```

static function get($key, $defvalue='') {
    if (self::has($key)) {
        global $$key;
        return $$key;
    }
    return $defvalue;
}

static function has($key) {
    global $$key;
    return (isset($$key));
}

static function getOD($key, $defvalue='') {
    return '';
}

static function hasOD($key) {
    return false;
}
}

//Set the default layout
$default_layout = 'v7';

include_once 'config.security.php';

$relModuleName='Contacts';
$relModule= _Module::getInstance($relModuleName);
if (!$relModule) {
    echo "Module '". $relModuleName. "' not found. <br />\n";
    return;
}

$focus->setRelatedList($relModule, $relModuleName, Array('ADD', 'SELECT'));

echo "Relation $MODULENAME => $relModuleName was established\n";
$ _Utils_Log = true;

```

```

$MODULENAME = 'Exams';

$moduleInstance = _Module::getInstance($MODULENAME);
if ($moduleInstance) {
    echo "Module $MODULENAME already present - choose a different name.";
    $moduleInstance->delete();
} else {
    $moduleInstance = new _Module();
    $moduleInstance->name = $MODULENAME;
    $moduleInstance->parent= 'Marketing';
    $moduleInstance->save();

    $moduleInstance->initTables();

    $block = new _Block();
    $block->label = 'LBL_' . strtoupper($moduleInstance->name) .
'_INFORMATION' ;
    $moduleInstance->addBlock($block);

    $blockcf = new _Block();
    $blockcf->label = 'LBL_CUSTOM_INFORMATION' ;
    $moduleInstance->addBlock($blockcf);

    $field1 = new _Field();
    $field1->name = 'contact_id';
    $field1->label= 'Contact';
    $field1->uitype= 10;
    $field1->column = $field1->name;
    $field1->columnType = 'int(19)';
    $field1->typeofdata = 'I~M';
    $field1->displaytype = 1;
    $block->addField($field1);
    $field1->setRelatedModules(Array('Contacts'));
    echo "Created $field1->name \n";

    $moduleInstance->setEntityIdentifier($field1);

```

```

$field2 = new _Field();
$field2->name = 'course_id';
$field2->label= 'Course';
$field2->uitype= 10;
$field2->column = $field2->name;
$field2->columnntype = 'int(19)';
$field2->typeofdata = 'I~M';
$field2->displaytype = 1;
$block->addField($field2);
$field2->setRelatedModules(Array('Courses'));
echo "Created $field2->name \n";

```

```

$field3 = new _Field();
$field3->name = 'ticket_no';
$field3->label= 'Ticket number';
$field3->uitype= 2;
$field3->column = $field3->name;
$field3->columnntype = 'VARCHAR(255)';
$field3->typeofdata = 'V~M';
$field3->displaytype = 1;
$block->addField($field3);
echo "Created $field3->name \n";

```

```

$field4 = new _Field();
$field4->name = 'grade';
$field4->label= 'Grade';
$field4->uitype= 2;
$field4->column = $field4->name;
$field4->columnntype = 'VARCHAR(255)';
$field4->typeofdata = 'V~M';
$field4->displaytype = 1;
$block->addField($field4);
echo "Created $field4->name \n";

```

```

$field99 = new _Field();
$field99->name = 'description';
$field99->label= 'Description';
$field99->uitype= 19;
$field99->column = 'description';

```

```

        $field99->table = '_crmentity';
        $blockcf->addField($field99);
        echo "Created $field99->name \n";

        $filter1 = new _Filter();
        $filter1->name = 'All';
        $filter1->isdefault = true;
        $moduleInstance->addFilter($filter1);
        $filter1->addField($field1)->addField($field2, 1)->addField($field3,
2)->addField($field4, 3);
        echo "Created $filter1->name \n";

        $moduleInstance->setDefaultSharing();

        $moduleInstance->initWebservice();

        Settings_MenuEditor_Module_Model::addModuleToApp($MODULENAME,
'MARKETING');

        echo "OK\n";
    }

    global $adb;

    $result = $adb->pquery("update _tab set source=null where name=?",
Array($MODULENAME));
    //Update
    $result = $adb->query("SELECT id FROM _users");
    $num_rows=$adb->num_rows($result);
    for($i=0;$i<$num_rows;$i++)
    {
        echo 'Generating for ID=' . $id;
        $id=$adb->query_result($result,$i,'id');
        createUserPrivilegesfile($id);
        createUserSharingPrivilegesfile($id);
    }

```

ПРИЛОЖЕНИЕ Б

АКТ О ВНЕДРЕНИИ В ЭКСПЛУАТАЦИЮ



Автономная некоммерческая организация
дополнительного профессионального образования
«Сибирский учебный центр СПАС»

адрес: 625000, Россия, Тюменская область, г. Тюмень, ул. Широтная д. 138/4.
тел/факс: 8(3452) 56-54-06, e-mail: mail@spas72.com, www.spas72.com

Акт

о внедрении научно-технической продукции

Данный акт удостоверяет, что в Автономной некоммерческой организации дополнительного профессионального образования «Сибирский учебный центр СПАС» внедрен в эксплуатацию программный комплекс для поддержки бизнес-деятельности учебного центра (CRM-система для учебного центра на базе vTiger), разработанный в Южно-Уральском государственном университете студентом группы КЭ-452 «Информатика и вычислительная техника» Белоконым Владиславом Дмитриевичем, научный руководитель: доцент кафедры «Электронные вычислительные машины», кандидат технических наук Надточий Ирина Львовна.

Разработанный программный комплекс внедрен в эксплуатацию АНО ДПО «Сибирский УЦ СПАС» и используется сотрудниками учебного центра, для автоматизации, учета и формирования всей документации в соответствии с Федеральным законом "Об образовании в Российской Федерации" N 273-ФЗ от 29 декабря 2012 года с изменениями 2019 года.

Заместитель директора
по общим вопросам - Начальник
технического отдела



Дьяков А.А.

Рисунок Б.1 – Акт о внедрении в эксплуатацию