Оглавление

[1. ИССЛЕДОВАНИЕ И АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc518908929)

[1.1 Общие сведения 6](#_Toc518908930)

[1.2 Обзор аналогов 9](#_Toc518908931)

[1.3 Общие критерии относительно поставленной задачи 15](#_Toc518908932)

[**2. ПРОЕКТИРОВАНИЕ СТРУКТУРЫ СИСТЕМЫ** 17](#_Toc518908933)

[2.1. Выбор методов и средств реализации, его обоснование 17](#_Toc518908934)

[2.2. Архитектура, структура программного продукта 21](#_Toc518908935)

[2.5 Функциональная схема программного продукта 38](#_Toc518908936)

[СПИСОК ЛИТЕРАТУРЫ 40](#_Toc518908937)

ВВЕДЕНИЕ

Основой повышения качества и эффективности системы подготовки высококвалифицированных специалистов является применение современных образовательных технологий, интегрирующих в себе как новые информационные, так и традиционные образовательные [1].   
 В последние десятилетия произошли изменения в сфере использования информационно-коммуникативных технологий (ИКТ) в учебном процессе вузов. Заметное распространение поучили репозитории электронных учебно-методических материалов (ЭУММ) [2].

Но все же что нужно для повышения знаний студента – это иметь постоянный доступ к достоверным и структурированным учебным материалам. Однако обладателями достоверных сведений, как правило, являются преподаватели, которым необходимо преподнести информацию студенту в понятной для него форме. И все же доступность материалов, это залог того что потенциальный ученик или же студент, получит и возможно усвоит максимальный объем знаний из предоставляемого учебного материала.

Таким образом, *целью* данной выпускной квалификационной работы является создание *Web*-приложение для организации доступа к учебным материалам вуза.

Объектом исследования является организация доступа к учебным материалам в филиале Приднестровского государственного университета   
им. Т.Г.Шевченко в г. Рыбница.

Предметом исследования стала возможность организации хранения и свободного доступа к материалам вуза в сети интернет.

В соответствии с указанной выше целью поставлены следующие основные задачи:

* изучить теоретические аспекты исследуемой предметной области;
* реализовать модель *web*-сервиса с учетом основных функций участников приложения;
* реализовать основную фундаментальную программную площадь для дальнейшей разработки и модификации приложения;
* организовать древовидную систему доступа к сайту, систему которая позволит настраивать доступ настолько тонко и гибко, вплоть до отображения каких-либо отдельных визуальных элементов конкретному конечному пользователю;
* реализовать систему для назначения, создания, редактирования - ролей, разрешений, правил;
* реализовать функциональные возможности системы рекомендаций материалов для определенных групп студентов, пользователей, кафедр и направлений.
* реализовать наиболее выгодное хранение данных на сервере приложения;

Цель и задачи выпускной квалификационной работы определили её структуру, которая состоит из введение, трех глав, заключения, списка литературы, приложения.

Во введении определены актуальность, цель, задачи, объект и предмет исследования.

В первой главе рассматриваются сведения о предметной области, производиться обзор существующих программных продуктов по организации учебного процесса с использованием *web*-приложения.

Во второй главе рассматриваются методы и средства проектирования структуры и архитектуры программного продукта.

В третьей главе описываются методы реализации программного продукта.

Заключение содержит результаты проделанной работы, а также недостатки, преимущества, и перспективы дальнейшего развития программного продукта.

В приложении указывается перечень условных обозначений, символов, единиц и терминов.

# ИССЛЕДОВАНИЕ И АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

# Общие сведения

Происходящее в последнее время стремительное обновление основных массивов профессиональной информации в большинстве предметных областей всё более затрудняет использование в образовательном процессе классических форм и методов ведения учебного процесса. Это неизбежно сказывается на качестве подготовки специалистов. Только новые информационные технологии в состоянии в должной степени обеспечить растущие потребности обучаемых не только в области получения актуальных на настоящий момент знаний в большинстве отраслей современных культуры, искусства, науки, техники, но и помочь им в проведении научно-исследовательской работы студентов (НИРС).  
 Репозиторий – это хранилище, место, где хранятся и из которого распространяются по запросам какие-либо данные. В последние годы средством доставки данных из репозиториев являются компьютерные сети, в первую очередь, глобальная сеть Интернет. Существуют репозитории для хранения программ, написанных на одном языке или предназначенных для одной операционной системы. Современные компьютерные операционные системы имеют официальные репозитории совместимого с ними программного обеспечения, но также позволяют устанавливать пакеты из других мест. Большинство репозиториев бесплатны, однако некоторые компании предоставляют доступ к собственным репозиториям по платной подписке. Однако при использовании в обеспечении учебного процесса понятие «репозиторий» приобрело значение, несколько отличное от вышеизложенного. По сути, так теперь называют полнотекстовые базы данных учебного или справочного назначения – базы знаний, применительно к учебному процессу вузов ранее именовавшиеся электронным учебно-методическим комплексом (ЭУМК) вуза либо интегрированной информационной системой вуза (ИИС-вуз) в случае объединения ЭУМК вуза и его электронной библиотечной системы в едином интерфейсе с дополнительными возможностями электронного деканата.

В процессе разработки различные варианты репозитория нами были технологически созданы и внедрены в учебный процесс Санкт-Петербургского гуманитарного университета профсоюзов, Санкт-Петербургского филиала Института бизнеса и политики, филиала в г. Санкт-Петербурге Института государственного управления, права и инновационных технологий и частично в Санкт-Петербургском филиале Финансового университета при Правительстве РФ. При этом идеология репозитория для учебного процесса в рамках одной учебной дисциплины приводила в завершённом варианте к созданию мультимедийного учебника по этой дисциплине. Во многом такой подход описывается терминологией электронной библиотеки.

Электронная библиотека ­­ - упорядоченная коллекция разнородных электронных документов (в том числе книг, журналов), снабжённых средствами навигации и поиска. Может быть веб-сайтом, где постепенно накапливаются различные тексты (чаще литературные, но также научные и любые другие, вплоть до компьютерных программ) и медиафайлы, каждый из которых самодостаточен и в любой момент может быть востребован читателем. Электронные библиотеки могут быть универсальными, стремящимися к наиболее широкому выбору материала (как Библиотека Максима Мошкова или Либрусек), и более специализированными, как Фундаментальная электронная библиотека или проект Сетевая Словесность, нацеленный на собирание авторов и типов текста, наиболее ярко заявляющих о себе именно в Интернете.

Электронные библиотеки следует отличать от смежных структурных типов сайта, особенно литературного. В отличие от литературного журнала, родившегося как тип печатного издания, но успешно и без принципиальных изменений структуры перебравшегося в Интернет, электронная библиотека не подразделяется на выпуски и обновляется перманентно по мере появления новых материалов. В отличие от сайта со свободной публикацией, электронная библиотека, как правило, подбирается координатором проекта по своему усмотрению и, что гораздо более важно, не предусматривает создания вокруг публикуемых текстов коммуникативной среды. При этом в практике отдельных Интернет-проектов могут возникать и гибридные формы, и промежуточные решения: так, открытие в электронной библиотеке Сетевая Словесность гостевых книг для каждого публикуемого автора в известной степени вносит в проект элемент формирования коммуникативной среды, состоящей из авторов и читателей, что для электронных библиотек вообще нехарактерно.

Особое место в ряду электронных библиотек занимают библиотеки научно-образовательной тематики, в которых собраны издания, необходимые для осуществления образовательного процесса. Использование электронных изданий в высших учебных заведениях зачастую ведётся на не совсем законных основаниях. Однако, этот нелицеприятный факт свидетельствует об интересе нового поколения к электронной форме представления информации. Мировой опыт показывает, что образовательные и научные электронные ресурсы формируются как правило самими правообладателями или с их согласия. Большинство из них действуют на коммерческой основе [4]. Самыми яркими примерами данного подхода являются крупнейшие издательские дома «*Elsevier*», «*Springer*».

В последнее время стали появляться электронные научные библиотеки открытого доступа (*open* *access*), одними из первых представителей которых стали *arXiv.org*и *PubMed Central* [5,6].

В России же образовательные электронные библиотеки ознаменовали специальным термином — электронно-библиотечная система, внедрить которую обязали в вузовские библиотеки новые стандарты ФГОС ВПО[6]. Электронно-библиотечную систему ВУЗ может разработать как сам на основе своей электронной библиотеки, так и воспользоваться внешним поставщиком услуг. Вузовская электронная библиотека как основной электронный образовательный ресурс, выполняющий возложенные на него функции по работе с полными текстами, позволит разгрузить АБИС библиотеки и наладить эффективную работу с издательской литературой [7].

# 1.2 Обзор аналогов

В связи с ростом ценности знаний в сети присутствует много сервисов, которые могут предложить свою платформу создания онлайн курсов для обучения. Рассмотрим некоторые их примеры.

*DSPACE*

*DSpace* — открытое, свободное (лицензия BSD) кроссплатформенное J2EE приложение, платформа для институционального репозитория (для долгосрочного хранения цифровых материалов, используемых в академических исследованиях [8].

Для сборки и функционирования *DSpace* требуется J2EE контейнер. Для версии 1.8.х желательно использовать *Oracle* *Java* 6(обычного JDK достаточно, J2EE использовать не обязательно). С работоспособностью под *Oracle* *Java* 7 имеются проблемы[3]. Для версии 3.х возможно использование как *Oracle* *Java* 6, так и *Oracle* *Java* 7 (для сборки потребуется maven 2.2 и выше)[4], версии 4.х и 5.х требуют *Java* 7[5].

Для хранения метаданных и другой информации можно использовать *Oracle*, *PostgreSQL*. Визуальное оформление(рис.1.1).

Платформа электронных библиотек *DSpace* разрабатывалась совместно компанией *Hewlett*- *Packard* и библиотеками *MIT* (*Massachusetts* *Institute* of *Technology*).

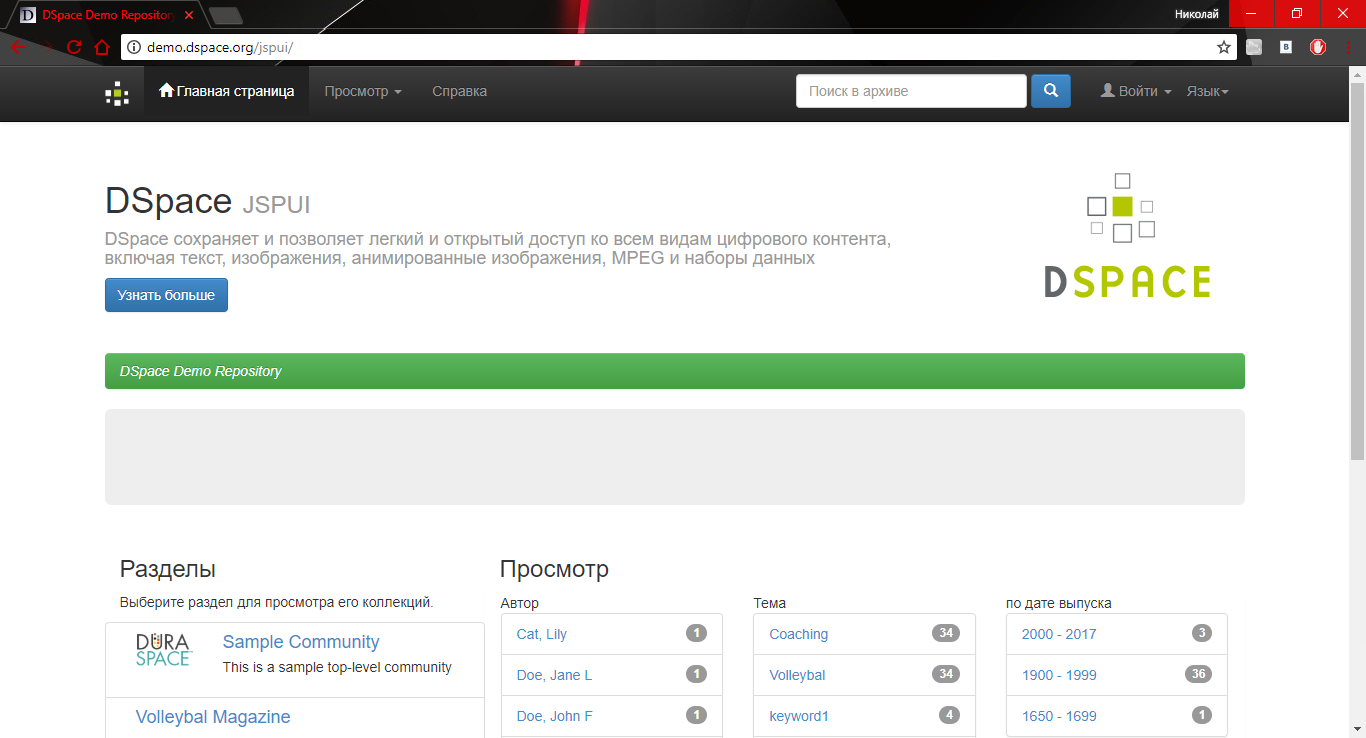


Рис. 1.1. Демо-страница сайта построенного на системе «*DSpace*»

4-го ноября 2002 года система была запущена как действующая служба, поддерживаемая библиотеками *MIT*. *DSpace* формировалась под влиянием научных исследований в области систем электронных библиотек. Платформа *EPrints* произошла из движения *Scholarly* *Communication* (“Общение в мире науки ”), как следствием конфигурация по умолчанию направлена на поддержку научных статей. *EPrints* имеет много сходства с *DSpace*, но оптимизирована так, чтобы обеспечить доступ к самостоятельному депонированию (внесению) автором материалов (статей, отчетов, книг и пр.), в то время как *DSpace* обеспечивает платформу, для долгосрочного хранения цифровых материалов, используемых в академических исследованиях Функциональные возможности *DSpace*:

* Для базовой организации данных зафиксирована определенная модель данных;
* Система хранит и индексирует метаданные в разнообразных форматах;
* Система хранит информацию о пользователях системы;
* Авторизация пользователей с целью разделения уровней доступа к репозиторию;
* Система должна быть способна принимать приходящие материалы, такой процесс называется поглощением;

1. В некоторых случаях требуется, чтобы материалы или связанные с ними метаданные, вносимые в архив, были проверены или дополнены назначенными пользователями. Эта последовательность действий называется рабочим процессом (*work*-*flow*);
2. Материалы в архиве доступны по ссылкам, приведенным в описании конкретного элемента. Поэтому же описанию можно делать библиографические ссылки на данный материал;
3. Конечные пользователи должны иметь возможность просматривать и находить содержимое репозитория. В связи с этим система должна обеспечивать функции поиска и просмотра (навигации);
4. Для предоставления возможности интегрированного поиска документов поддерживается протокол сбора метаданных *OAI*-*PMH*;
5. Должна существовать возможность уведомления конечных пользователей системы об интересующих свежих поступлениях репозитория, чтобы избавить их от необходимости регулярно посещать репозиторий для выполнения такой проверки. Система предоставляет автоматическую рассылку уведомлений по электронной почте через службу подписки;
6. Предоставляется возможность обрабатывать данные произвольных форматов, от простых текстовых документов до наборов данных и цифрового видео;
7. Доступ к перечисленным функциональным возможностям предоставляется посредством веб-интерфейса.

С помощью *EPrints* можно так же обеспечить подобную функциональность репозитория. Основные отличия *EPrints* и *DSpace* лежат в их подходе к организации структуры данных. В *DSpace* данные организуются таким образом, чтобы отобразить структуру организации, использующую данную систему. *DSpace* платформа на которой возможно построение электронной библиотеки, за короткие сроки, но с учетом открытости кода,

*ELIBRARY.RU*

*elibrary.ru* - крупнейшая в России электронная библиотека научных публикаций, обладающая богатыми возможностями поиска и анализа научной информации. Библиотека интегрирована с Российским индексом научного цитирования (РИНЦ) - созданным по заказу Минобороны науки РФ бесплатным общедоступным инструментом измерения публикационной активности ученых и организаций. *elibrary.ru* и РИНЦ разработаны и поддерживаются компанией "Научная электронная библиотека".

Платформа *elibrary.ru* была создана в 1999 году по инициативе Российского фонда фундаментальных исследований для обеспечения российским ученым электронного доступа к ведущим иностранным научным изданиям. С 2005 года *elibrary.ru* начала работу с русскоязычными публикациями и ныне является ведущей электронной библиотекой научной периодики на русском языке в мире.

На сегодня посетителям elibrary.ru доступны рефераты и полные тексты более 26 млн научных статей и публикаций, в том числе электронные версии более 5300 российских научно-технических журналов. Общее число зарегистрированных институциональных пользователей (организаций) - более 2800. В системе зарегистрированы 1,7 миллиона индивидуальных пользователей из 125 стран мира. Ежегодно читатели получают из библиотеки более 12 миллионов полнотекстовых статей и просматривают более 90 миллионов аннотаций. Свыше 4500 российских научных журналов размещены в бесплатном открытом доступе. Для доступа к остальным изданиям предлагается возможность подписаться или заказать отдельные публикации. Главная страница сайта (рис 1.2)



Рис. 1.2. Страница сайта eLIBARARY.RU

Российский индекс научного цитирования (РИНЦ) - это национальная информационно-аналитическая система, аккумулирующая более 11 миллионов публикаций российских авторов, а также информацию о цитировании этих публикаций из более 6000 российских журналов. Она предназначена не только для оперативного обеспечения научных исследований актуальной справочно-библиографической информацией, но является также и мощным инструментом, позволяющим осуществлять оценку результативности и эффективности деятельности научно-исследовательских организаций, ученых, уровень научных журналов и т.д.

RGB

Российская Государственная Библиотека.

Данная библиотека это некий сборник сайтов разных библиотек разных классификаций Российской Федерации.

Эта коллекция создана для расширения доступа к цифровым фондам РГБ. Ее формирование позволяет сделать еще один шаг в урегулировании отношений библиотек с правообладателями.

В коллекцию включены: монографии, в том числе переводные, сборники научных трудов;

учебники и учебные пособия для высших и средних учебных заведений, выпущенные за последние 10 - 12 лет и отражающие современное состояние науки и знания;

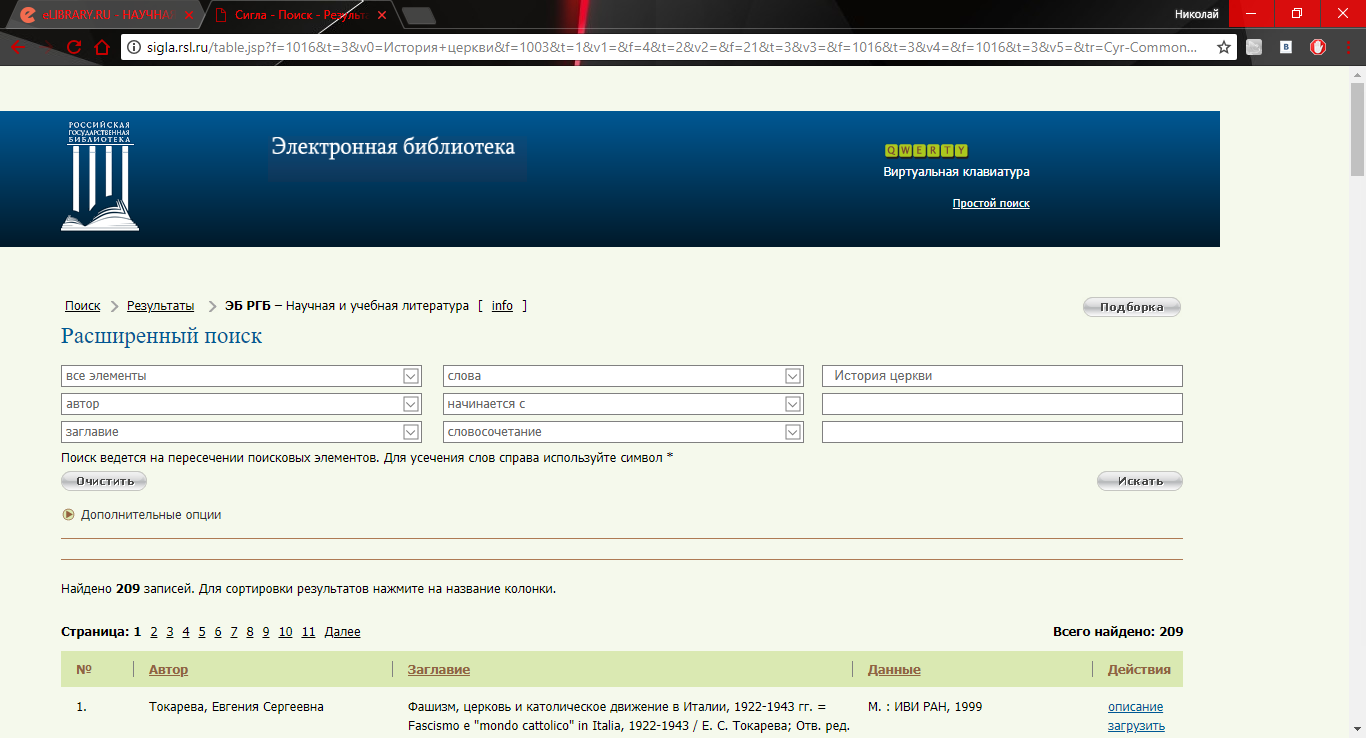
классические научные труды, имеющие непреходящее значение.

Тематика коллекции литература по юриспруденции ведущих издательств, в том числе комментарии к законодательным материалам, учебники по законодательству РФ и зарубежных стран и т. д.;

* литература по всему спектру психологических наук;
* издания по широкому ряду экономических дисциплин;
* монографии и учебники по социологии, политологии, истории, культурологии, филологии;
* учебники и учебные пособия по математическим и физическим дисциплинам;
* издания по технике, логистике, строительству, геологии, экологии, океанологии, безопасности жизнедеятельности, дизайну и др.

Читателю доступны книги таких издательств, как «Просвещение», «Юнити», «Когито-Центр», «Питер», «Лань», «Финансы и статистика», «Книжный мир», «Алетейя», издательств Института психологии РАН и РАГСа и др. Коллекция создается на базе договоров с авторами и правообладателями на оцифровку произведений и включение их в состав электронной библиотеки РГБ. В коллекцию входят также документы, оцифрованные в ходе работы по созданию Национального библиотечного ресурса.

В связи с ограничениями, накладываемыми законодательством об авторском праве, большинство материалов коллекции находятся в ограниченном доступе. С ними можно познакомиться в одном из читальных залов РГБ. Коллекция постоянно пополняется. Визуально, сайт выглядит рис(1.3).

Рис. 1.3. Страница одного из сайта каталогов RGB

# 1.3 Общие критерии относительно поставленной задачи

Так как РФ ПГУ им Шевченко, книжный фонд относительно других университетов и библиотек, является не большим, то организация доступа к материалам не должна содержать древовидную структуру поиска, так как банально по причине сложности его создания, и просто очень огромный массив индексов библиотечного дела - усложнит поиск материалов для нуждающихся в них студентов, а не упростит. К тому же не все аналоги используют такую систему хранения. Собственно, потому что это быстро для поиска, но не удобно для пользователя. Так как уровень такого дерева (бесконечно глубокого) естественно не поддается мысленным соображениям.

Так же по опросам библиотекарей учебного учреждения РФ ПГУ им Шевченко, было выявлено что для студентов данные индексы не представляют преимуществ, то есть это означает что смысл хранения и сильно структуризации файлов отсутствует, так как конечному пользователю цитирую, нужно: «зайти на сайт, вбить в поиск и скачать».

Так как разрабатываемый программный продукт предназначается в основном для студентов учебного учреждения РФ ПГУ им Шевченко, то был проведен не гласный опрос, студентов с разных кафедр и разных направлений.

Опрос показал, что есть нуждаемость студентов в простоте доступа к материалам, и максимизации поиска по нужным им критериям.

**Выводы по первой главе**. В ходе изучения предметной области были проанализированы:

* специфика организации доступа к файлам в РФ ПГУ им. Т.Г. Шевченко;
* аналогичные платформы организации доступа к учебным материалам и не только;
* недостатки готовых продуктов типа CMS;

**2. ПРОЕКТИРОВАНИЕ СТРУКТУРЫ СИСТЕМЫ**

# 2.1. Выбор методов и средств реализации, его обоснование

Не смотря на то что целью данной работы является расширение функционала разработанного ранее *web*-приложения. Было решено разрабатывать *web*-приложения заново. Так как

* данные приложения писались на языках программирования ветки поддержки которых в скорейшем времени обещают прекратить. То есть актуальность их дальнейшего использования в доработки и разработки каких-либо определенных веток будет под сильным затруднением;
* нуждаемость в функционале предшественников, в своей разработке, присутствовала в минимальном количестве;
* несовместимость платформ *web*-разработки, цели данной выпускной квалификационной работы затрагивают некоторые важные возможности которые было бы проще выполнить на новой платформе *web*-разработки;
* отсутствие жестких критериев по поводу взятия на вооружения в своей разработки предшественников;

Затруднения читаемости кода предшественников. Отсутствие комментариев, и сжатые сроки разработки вынудили использовать несколько новые и другие методы разработки *web*-приложений.

В данной выпускной квалификационной работе были использованы продукты и языки программирования предыдущей разработки, однако, обновлены на более новые версии, а именно: фреймворк *Yii*(2.0.15), фреймворк *Yii*(1.1.14), язык программирования *PHP* (5.6.32), вместо *PHP* (5.4.19), база данных *MySQL*(5.7.32), *jQuery*(1.8.3).

*PHP*

Главным фактором, влияющим на выбор языка *РНР*, является его практичность. *РНР* предоставляет программисту средства для быстрого и эффективного решения поставленных задач. Практический характер *РНР* обусловлен пятью важными характеристиками:

* традиционностью;
* простотой;
* эффективностью;
* безопасностью;
* гибкостью.

Существует еще одна «характеристика», которая делает *РНР* особенно привлекательным: он распространяется бесплатно с открытыми исходными кодами (*Open Source*) [19].

*jQuery*

Зачастую для добавления в проекты динамичности используют *JavaScript*, *html*-страницы становятся более отзывчивыми и привлекательными по сравнению с чистым *HTML*. Для того чтобы в каждом новом проекте заново не писать уже реализованных алгоритм используют фреймворк, т.е. *jQuery*.

Что дает использование *jQuery*:

1. Скорость. Сокращение количества набираемого кода.
2. Понятность. Код и функции *jQuery* гораздо более удобны для чтения и понимания.
3. Кроссбраузерность. Большинство современных браузеров уже поддерживают *jQuery*.
4. Поддержка. *JQuery* разрабатывается уже давно, разные версии *JQuery* совместимы друг с другом, над этим продуктом трудятся тысячи программистов. Позвольте им писать за вас.
5. Конкурентоспособность. Программист, использующий и знающий *JQuery* гораздо более высоко ценится на рынке труда.

*Bootstrap*

*Bootstrap* (от англ. сущ. "шнуровка" (ботинок) или гл. "добиваться самостоятельно") – бесплатный фреймворк (набор инструментов) для быстрой вёрстки сайтов и прототипирования интерфейсов веб-приложений. Создан изначально для нужд популярного микроблогингового сервиса *Twitter*, однако в 2011 году открыт для свободного скачивания всем желающим.

*Bootstrap* включает в себя готовые шаблоны блоков различной конфигурации, кнопок, таблиц, форм, которые добавляются в Ваш проект простым прописыванием специальных классов. Кроме готовых *CSS* и *HTML*-сниппетов, имеется и ряд *JavaScript*-дополнений, которые реализуют тот или иной специфический функционал.

Всё это в совокупности позволяет довольно быстро создавать прототипы различных пользовательских интерфейсов. Поэтому, если нужно быстро сверстать современный адаптивный сайт, можно воспользоваться средствами *Bootstrap*. В свою очередь является самым популярным фреймворком, у его ближайшего конкурента в 3-5 раз меньше сообщество. Это не только *css*, но и *js*-фреймворк. То есть в *Bootstrap* написаны готовые стили и скрипты, для применения которых вам достаточно всего лишь прописать необходимые стилевые классы и атрибуты *html*-элементам.

Основные инструменты *Bootstrap*:

* Сетки – заранее заданные размеры колонок, которые можно сразу же использовать, например ширина колонки 140 px относится к классу .span2 (.col-md-2 в третьей версии фреймворка), который можно использовать в CSS-описании документа.
* Шаблоны – фиксированный или резиновый шаблон документа.
* Типографика – описания шрифтов, определение некоторых классов для шрифтов, таких как код, цитаты и т. п.
* Медиа – представляет некоторое управление изображениями и видео.
* Таблицы – средства оформления таблиц, вплоть до добавления функциональности сортировки.
* Формы – классы для оформления форм и некоторых событий, происходящих с ними.
* Навигация – классы оформления для табов, вкладок, страничности, меню и панели инструментов.
* Алерты – оформление диалоговых окон, подсказок и всплывающих окон.

Эта библиотека начала разрабатываться как внутренняя библиотека компании *Twitter* под названием *Twitter Blueprint.* После нескольких месяцев разработки он был открыт под названием *Bootstrap* 19 августа 2011 года, и по сейчас она разрабатывается и модифицируется квалифицированным персоналом компании *Twitter.* [12]

*Yii*

*Yii* – это высокоэффективный основанный на компонентной структуре *PHP*-фреймворк для разработки масштабных *web*-приложений. Он позволяет максимально применить концепцию повторного использования кода и может существенно ускорить процесс веб-разработки. Название *Yii* означает простой эффективный и расширяемый.

В *Yii* вложено много возможностей которые выделяют этот фреймворк на фоне остальных и делают разработку с его помощью очень быстрой и приятной.

*Архитектура Model-View-Controller (MVC).* Архитектура программного обеспечения, в которой модель данных приложения, пользовательский интерфейс и управляющая логика разделены на три отдельных компонента, так, что модификация одного из компонентов оказывает минимальное воздействие на другие компоненты. Благодаря отличной гибкости *Yii* вместо *MVC* вы можете использовать любую другую архитектуру (к примеру модульность в *HMVC*)

*Генератор кода.* *Yii* предоставляет отличный встроенный генератор исходного кода. Указав лишь основные параметры - *Yii* генерирует для вас общую структуру приложения которая будет содержать все необходимые модели/контроллеры/отображения для старта. Так же в новой версии фреймворка присутствует возможность писать свой генератор кода, что собственно было и реализовано для ускорения разработки.

*Расширения*. *Yii* позволяет подключить классы других фреймворков в качестве расширений и использовать их в своем приложении! Расширения подключаются буквально двумя строчками кода через встроенное приложение «*composer*» и позволяют использовать полностью весь функционал подключенных классов! На форуме фреймворка можно прочитать много статей про то как подключать классы от *Zend Framework, Code Igniter, Kohana* и тп.

*Защита*. Все стандартные классы *Yii* позволяет обеспечить высокий уровень безопасности, что при умелом обращении помогает реализовать систему защиты созданного сайта от *Sql-Inj, XSS, CSRF* и других атак.

# 2.2. Архитектура, структура программного продукта

Рассматриваемый программный продукт использует схему проектирования *MVC* (*Model-View-Controller*), которое представляет собой взаимодействие трех компонентов.

*Model-view-controller* (*MVC*, «модель – представление – контроллер») – схема использования нескольких шаблонов проектирования, с помощью которых модель приложения, пользовательский интерфейс и взаимодействие с пользователем разделены на три отдельных компонента таким образом, чтобы модификация одного из компонентов оказывала минимальное воздействие на остальные. Данная схема проектирования часто используется для построения архитектурного каркаса, когда происходит переход от теории к реализации в конкретной предметной области.

Концепция *MVC* позволяет разделить данные, представление и обработку действий пользователя на три отдельных компонента:

1. Модель (англ. *Model*). Модель предоставляет знания: данные и методы работы с этими данными, реагирует на запросы, изменяя свое состояние. Не содержит информации, как эти знания можно визуализировать.
2. Представление, вид (англ. *View*). Отвечает за отображение информации (визуализацию). Часто в качестве представления выступает форма (окно) с графическими элементами.
3. Контроллер (англ. *Controller*). Обеспечивает связь между пользователем и системой: контролирует ввод данных пользователем и использует модель и представление для реализации необходимой реакции [11]. Визуальная структура на рис 2.1

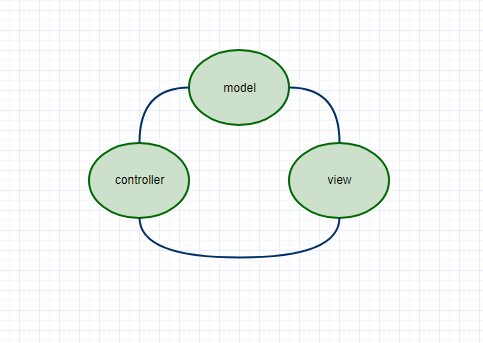


Рис. 2.2. Концепция *MVC*

Для организации доступа к материалам, собственно изначально этот доступ нужно контролировать. Логично что его нужно контролировать, то есть давать определенные разрешения, права, роли пользователям. Реализованы функциональные возможности для пользователей:

* администратор;
* модератор;
* библиотекарь;
* куратор;
* преподаватель;
* студент;
* староста;
* пользователь.

Используя в отличии от предшественников систему *RBAC (Role Base System Controll).* Была выведена структура которая показана на рис 2.2

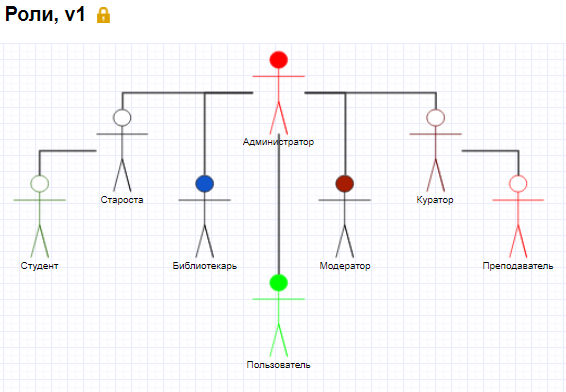


Рис. 2.2. Структура ролей, генерированная с помощью RBAC

Так, для пользователя «Администратор» представлены следующие основные функции (рис 2.4):

* регистрация пользователей (через стандартный интерфейс)
* создание направления, удаление направления, редактирование данных направления, просмотр данных направления;
* создание кафедры, удаление кафедры, редактирование данных кафедры, просмотр данных кафедры;
* создание группы, удаление группы, редактирование данных группы, просмотр данных группы;
* удаление пользователей;
* редактирование (частичных) данных, пользователей;
* просмотр (частичных) данных, пользователей;
* создание файлов, удаление файлов, редактирование данных файлов, просмотр данных файлов, скачивание файлов, открытие;
* создание типов файлов, удаление типов файлов, редактирование данных типов файлов, просмотр данных типов файлов;
* добавление новой роли;
* добавление разрешения;
* добавление правил для разрешений;
* удаление роли;
* редактирование разрешений роли;
* просмотр данных информации о ролях;
* назначение ролям разрешений;
* назначения ролям правил;
* назначение ролям роли;
* назначения пользователям роли:
* назначения пользователям разрешения;
* назначения разрешениям разрешения;
* создание разделов, удаление разделов, редактирование данных разделов, просмотр данных разделов;
* создание подразделов, удаление подразделов, редактирование данных подразделов, просмотр данных подразделов;
* назначения и удаления рекомендации группам;
* назначения и удаления рекомендации кафедрам;
* назначения и удаления рекомендации пользователям;
* назначения и удаления рекомендации направлениям.

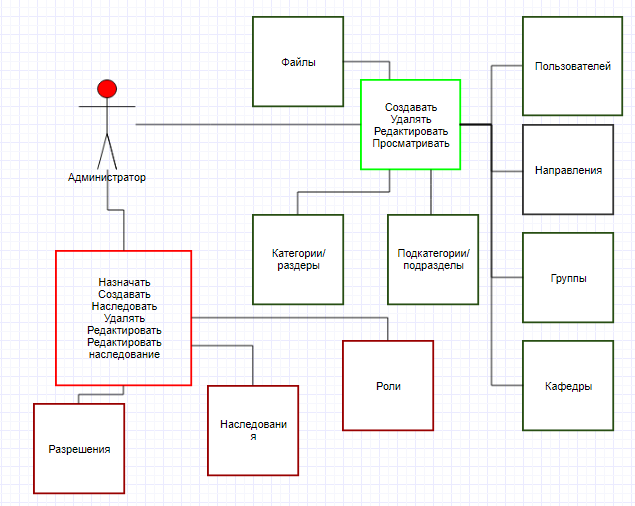
****

Рис. 2.5 Структура возможностей роли «Администратор»

Администратор это такая сущность, которая по сути может все в данном приложении, а так-же гибко его настраивать, дополнять нужной информацией, давать пользователям нужные права, на нужные ресурсы, или их ограничивать.

Так, для пользователя «Библиотекарь» представлены следующие основные функции (рис 2.4):

* регистрация пользователей (через стандартный интерфейс)
* добавление файлов, удаление файлов, редактирование данных файлов, просмотр данных файлов, скачивание файлов, открытие;
* создание типов файлов, удаление типов файлов, редактирование данных типов файлов, просмотр данных типов файлов;
* создание разделов, удаление разделов, редактирование данных разделов, просмотр данных разделов;
* создание подразделов, удаление подразделов, редактирование данных подразделов, просмотр данных подразделов;
* назначения и удаления рекомендации группам;
* назначения и удаления рекомендации кафедрам;
* назначения и удаления рекомендации пользователям;
* назначения и удаления рекомендации направлениям;

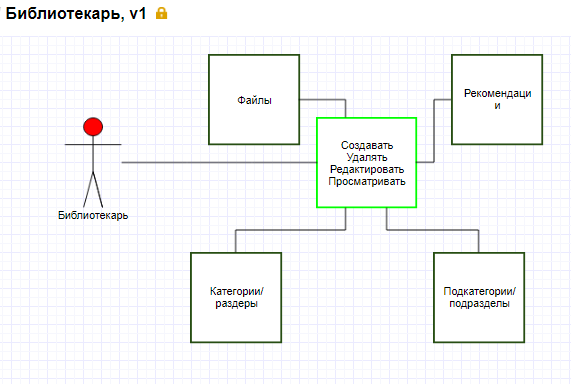
****

Рис. 2.6 Структура возможностей роли «Библиотекарь»

Роль «Библиотекарь» создавалась с одной целью – наполнять контентом приложения, «библиотекарь» может полностью взаимодействовать с файлами и все что связанно с ними. Его главная задача наполнять, редактировать и удалять за надобностью файлы из системы.

Для пользователя «Студент» представлены следующие основные функции рис 2.7

* просмотр данных файлов, скачивание файлов, открытие;
* просмотр рекомендуемых;
* просмотр рекомендуемых для кафедры;
* просмотр рекомендуемых для направления;
* просмотр рекомендуемых для группы;
* редактирование профиля;
* добавление фотографии на профиль,

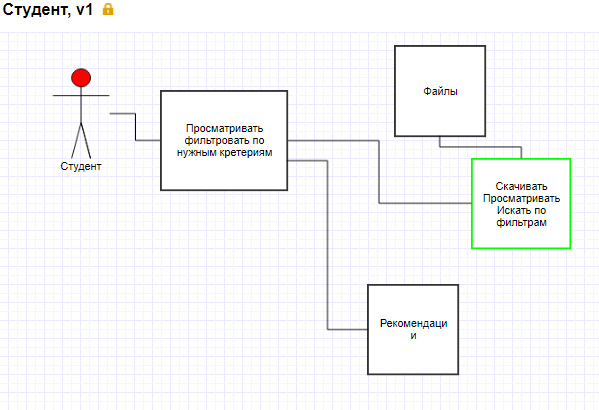
****

Рис. 2.7 Структура возможностей роли «Студент»

Роль «Студент» предназначалась основному классу пользователей, которые имеют участие в университете, так же данная роль была разработана для дальнейшей модификации приложения. В данный момент пользователи с ролью «Студент» имеют право, просматривать именно те рекомендации которые ему были назначены, другими пользователями системы по направлению, группе или кафедре.

Для пользователя «Староста» представлены следующие основные функции рис 2.8

* просмотр данных файлов, скачивание файлов, открытие;
* просмотр рекомендуемых;
* просмотр рекомендуемых для кафедры;
* просмотр рекомендуемых для направления;
* просмотр рекомендуемых для группы;
* добавление рекомендуемых для группы;
* удаление рекомендуемых для группы;
* редактирование профиля;
* добавление фотографии на профиль.

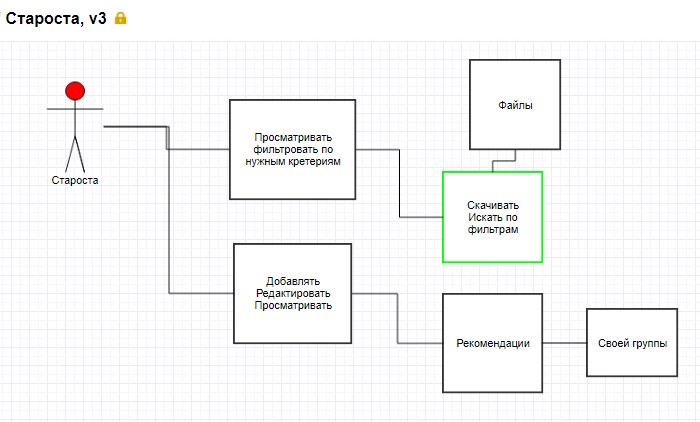
****

Рис. 2.8 Структура возможностей роли «Староста»

Роль «Староста» предназначалась основному классу пользователей, которые имеют участие в университете, так же данная роль была разработана для дальнейшей модификации приложения. В данный момент пользователи с ролью «Староста» имеют право, просматривать именно те рекомендации которые ему были назначены, другими пользователями системы по направлению, группе или кафедре. Но так же они могут рекомендовать файлы пользователям из своей группы.

Для пользователя «Преподаватель» представлены следующие основные функции рис 2.9

* просмотр данных файлов, скачивание файлов, открытие;
* просмотр рекомендуемых
* просмотр рекомендуемых для кафедры
* просмотр рекомендуемых для направления
* просмотр рекомендуемых для кафедры
* добавление рекомендуемых для пользователей
* редактирование профиля
* добавление фотографии на профиль

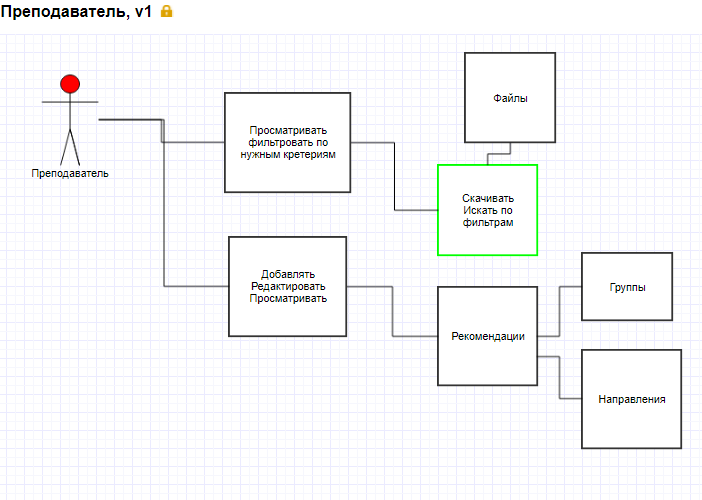


Рис. 2.9 Структура возможностей роли «Преподаватель»

Роль «Преподаватель» в основном была создана для дальнейшей разработки данного программного продукта. Тем не менее преподаватель, может назначать рекомендации по группам и направлениям.Так, для пользователя «Модератор» представлены следующие основные функции (рис 2.10):

* регистрация пользователей (через стандартный интерфейс)
* создание направления, удаление направления, редактирование данных направления, просмотр данных направления;
* создание кафедры, удаление кафедры, редактирование данных кафедры, просмотр данных кафедры;
* создание группы, удаление группы, редактирование данных группы, просмотр данных группы;
* удаление пользователей;
* редактирование (частичных) данных, пользователей;
* просмотр (частичных) данных, пользователей;
* создание файлов, удаление файлов, редактирование данных файлов, просмотр данных файлов, скачивание файлов, открытие;
* создание типов файлов, удаление типов файлов, редактирование данных типов файлов, просмотр данных типов файлов;
* создание разделов, удаление разделов, редактирование данных разделов, просмотр данных разделов;
* создание подразделов, удаление подразделов, редактирование данных подразделов, просмотр данных подразделов;
* назначения и удаления рекомендации группам;
* назначения и удаления рекомендации кафедрам;
* назначения и удаления рекомендации пользователям;
* назначения и удаления рекомендации направлениям.

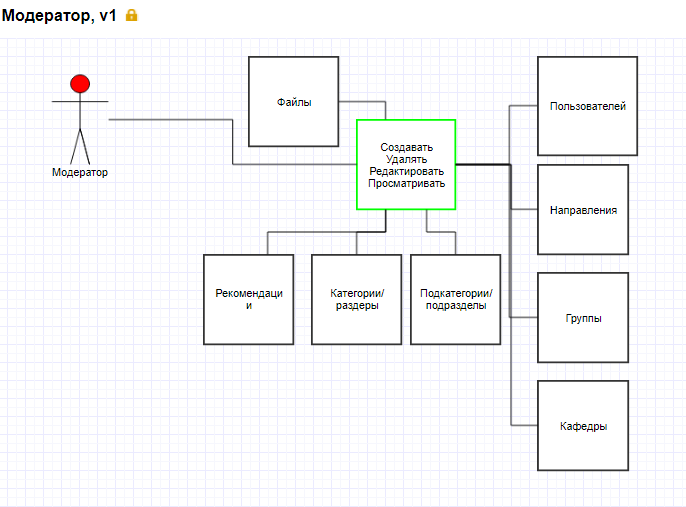
****

Рис. 2.10. Структура возможностей роли «Модератор»

Модератор может делать все тоже самое что и администратор, но не может назначать пользователям роли и не имеет доступ в систему *RBAC*. Предназначен для помощи администратору в корректировке данных на сайте и наполнение сайта.

|  |  |
| --- | --- |
| C:\Users\Westprophet\Desktop\BD.PNG | Рис 2.11 ER-диаграмма базы данных *web*-приложения |

Хранение всей необходимой информации организовано с помощью базы данных, имеющей структуру, представленную на рис 2.11. Она включает 10 таблиц, в которых хранятся сведения о пользователях, направлениях, кафедрах, группах, разделах, подразделах, файлах, типах файлов и 4 таблицы дерева RBAC, в них хранятся данные о системе доступа к сайту. (табл.2.1).

Особого внимания из указанных таблиц заслуживают те, которые хранят основные сведения о личных данных пользователей, о системных данных пользователя, файлах, и связанные с ними таблицах.

Для хранения данных, таких как файлы используются таблица *Files.* Где содержатся данные о файлах и все что связанно с файлами, под понятием файл имеется ввиду электронные книги, методички, пособия, научных работ и т.д. Конкретно о каком типе файла идет речь, указано в смежной таблице *type*\_*file* тип файла. Там идет перечень всех типов файлов используемых в системе.

Таблица 2.1

**Таблицы, используемые в базе данных**

|  |  |
| --- | --- |
| **Название таблицы** | **Описание содержимого таблицы** |
| *Files* | содержит список файлов |
| *Type\_file* | содержит список типов файлов |
| *Departmetns* | содержит список кафедр |
| *Groups* | содержит список групп |
| *Specialty* | содержит список направлений |
| *Categories* | содержит список категорий |
| *SubCategories* | содержит список подкатегорий |
| *User* | содержит список пользователей |
| *migrations* | содержит список миграций в базе данных |
| *auth\_assignment* | содержит список назначений ролей и разрешений |
| *auth\_item* | содержит список ролей и разрешений |
| *auth\_item\_child* | содержит список наследований |
| *auth\_rule* | содержит список правил для разрешения |
| *menu* | содержит список генерированных меню |

Таблица *Files* хранит в себе файлы и содержит следующие поля:

*id* - уникальный идентификатор записи (он же первичный ключ);

*name* – имя файла;

*description*- описание;

*authors*- авторы;

*content*-содержание ;

*type\_file\_id* - тип файла (тип пособия);

*date\_relise -* дата релиза*;*

*update\_time –* время загрузки*;*

*create\_time –* время редактирования*;*

*count\_download –* количество скачиваний*;*

*extension -* разширение*;*

*postfix –* постфикс индефикатор*;*

*dir\_link –* ссылка на папку*;*

*file\_link –* ссылка на файл*;*

*title\_link –* ссылка на титульник*;*

*back\_title\_link –* ссылка на обратную сторону (оборот) титульника;

*avatar –* ссылка на аватар*;*

*category\_id -* категория*;*

*subcategory\_id –*подкатегория*;*

*book\_index –* книжный индекс*;*

*publisher –* издание*;*

Таблица *type\_file* хранит в себе типы файлов:

*id –* уникальныйидентификатор записи (первичный ключ);

*name* – имя типа;

*description*- описание;

Таблица *Groups* хранит в себе группы:

*id –* уникальныйидентификатор записи (первичный ключ);

*name* – имя типа;

*description*- описание;

*department\_id -* уникальныйидентификатор записи кафедры;

*specialty\_id -* уникальныйидентификатор записи направления;

*count\_users –*количество пользователей в этой группе;

*recommended\_files\_ids -* в этой графе содержаться идентификатор рекомендуемых файлов.

Таблица *Departments* хранит в себе кафедры:

*id –* уникальныйидентификатор записи (первичный ключ);

*name* – имя типа;

*description*- описание;

*code –* код кафедры;

*count\_users –*количество пользователей в этой группе;

*recommended\_files\_ids -* в этой графе содержаться идентификатор рекомендуемых файлов.

Таблица *Specialty* хранит в себе направления:

*id –* уникальныйидентификатор записи (первичный ключ);

*name* – имя типа;

*description*- описание;

*code –* код специальности;

*count\_users –*количество пользователей в этой группе;

*department\_id -* уникальныйидентификатор записи кафедры;

*recommended\_files\_ids -* в этой графе содержаться идентификатор рекомендуемых файлов

Таблица *Categories* хранит в себе разделы:

*id –* уникальныйидентификатор записи (первичный ключ);

*name* – имя типа;

*description*- описание.

*book\_index –* книжный индекс;

Таблица *SubCategories* хранит в себе подразделы:

*id –* уникальныйидентификатор записи (первичный ключ);

*name* – имя типа;

*description*- описание.

*category\_id -* уникальныйидентификатор записи Категории (вторичный ключ);

*book\_index –* книжный индекс;

Таблица *auth\_item* хранит в себе предметы индификации и предметы доступа, роли и разрешения:

*name –* уникальныйидентификатор записи (первичный ключ);

*type*- тип записи;

*decription –* описание*;*

*rule\_name –* имя правила*;*

*data –* тип данных для работы модуля;

*create\_at –* время создания*;*

*update\_at –* время редактирования*.*

Таблица *auth\_item\_child* хранит в себе списки наследования где есть предок и его потомок, где права предка наследуются потомку, то есть он может все что:

*name –* уникальныйидентификатор записи (первичный ключ);

*type*- тип записи;

*decription –* описание*;*

*rule\_name –* имя правила*;*

*data –* тип данных для работы модуля;

*create\_at –* время создания*;*

*update\_at –* время редактирования*.*

# 2.5 Функциональная схема программного продукта

Созданные разновидности пользователей *web*-приложение для организации доступа к учебным материалам вуза, основные объекты программной разработки, а также схемы взаимодействия участников с этими объектами позволили сформировать следующую функциональную схему программного продукта, представленную на рис. 2.8.

Каждый пользователь имеет определенные возможности по работе с различными объектами, созданными в рамках web-приложения. Иерархическая сеть этих объектов отражена на рис 2.3.

Каждый тип пользователя наделен своим набором функций, который обеспечивает его взаимодействие с системой.

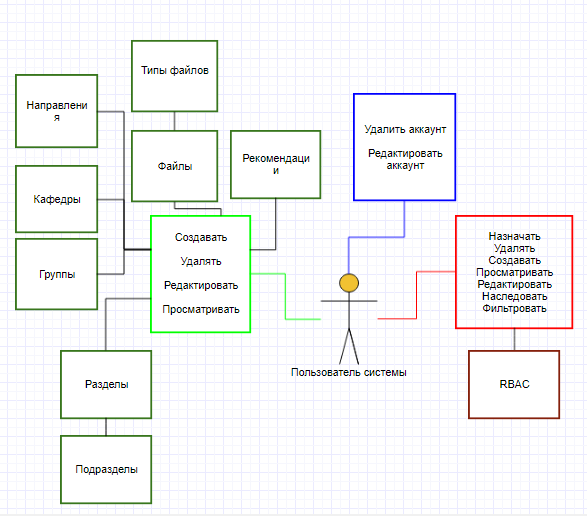


Рис. 2.3. Иерархическая сеть объектов *web*-приложения

Таким образом, в ходе работы над вторым разделом выпускного квалификационного проекта были описаны выбранные средства программной реализации, приведена архитектура и структура предложенного программного продукта как целостного явления, приведено функциональное назначение программного продукта путем рассмотрения диаграммы прецедентов и структуры разработанной базы данных.

# СПИСОК ЛИТЕРАТУРЫ

1. Махов А.М., Бондаренко Е.Ю. Электронный учебно-методический комплекс университета // Информационные ресурсы России. – 2004. – № 3. – С. 9-14.
2. Бондаренко Е.Ю., Махов А.М. Информационные системы в деятельности современного вуза / Сб. материалов II Общероссийской научно-практической конференции "Современные исследования социальных проблем". В 2 ч. - Ч.1. – Красноярск: ООО «Научно-инновационный центр», 2010. – С. 52-53.
3. А. Б. Антопольский, Т. С. Маркарова, Е. А. Данилина. Правовые и технологические проблемы создания и функционирования электронных библиотек. — М.: ИНИЦ «Патент», 2008. — С. 207. — ISBN 978-5-89513-119-0.
4. ПолянинА. Д., Журов А. И. Электронные публикации и основные физико-математические ресурсы Интернета. *EqWorld*. ИПМех РАН (2008). Проверено 9 февраля 2014.
5. ЛитвиноваН. Н. Научные публикации в Интернете: соотношение ограниченного (платного) и свободного доступов. *EqWorld*. ИПМех РАН (2005). Проверено 9 февраля 2014.
6. *Книжный рынок России. Состояние, тенденции и перспективы развития.2009 год: Отраслевой аналитический доклад / Под общ. ред. В. В. Григорьева. — М.: Федеральное агентство по печати и массовым коммуникациям, 2010. С.77* [Электронный ресурс] – режим доступа : [*http://www.fapmc.ru/magnoliaPublic/dms-static/5d876b2d-e3df-4ca1-927c-9dd0c532cbe2.pdf*](http://www.fapmc.ru/magnoliaPublic/dms-static/5d876b2d-e3df-4ca1-927c-9dd0c532cbe2.pdf)
7. Преимущества внедрения электронной библиотеки [*http://bibliostorm.ru*](http://bibliostorm.ru)
8. *Сравнение DSpace и EPrints* [Электронный ресурс] – режим доступа : [*http://rudspace.blogspot.com*](http://rudspace.blogspot.com)
9. *DSpace 1.8.x Installation* [Электронный ресурс] – режим доступа : *https://wiki.duraspace.org/display/DSDOC18/Installation#Installation-OracleJavaJDK6(standardSDKisfine,youdon'tneedJ2EE)*
10. *DSpace 3.x Installation* [Электронный ресурс] – режим доступа : *https://wiki.duraspace.org/display/DSDOC3x/Installation#Installation-OracleJavaJDK6or7%28standardSDKisfine,youdon%27tneedJ2EE%29orOpenJDK6or7.*
11. *PHP*. [Электронный ресурс]. – Режим доступа: [*https://ru.wikipedia.org/wiki/PHP*](https://ru.wikipedia.org/wiki/PHP).
12. *Bootstrap*. [Электронный ресурс]. – Режим доступа: *https://ru.wikipedia.org/wiki/Bootstrap*