

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра математической теории упругости и биомеханики

**Разработка официального сайта вуза как информационной системы с
веб-интерфейсом. Аспект безопасности и распределения прав.**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 441 группы

направления 09.03.03 – Прикладная информатика

механико-математического факультета

Алексеева Дмитрия Андреевича

Научный руководитель
старший преподаватель

Л.В. Бессонов

Зав. кафедрой
д.ф.-м.н., профессор

Л.Ю. Коссович

Саратов 2017

1 Введение

Передовые технологии стали незаменимым помощником для учреждений научно-образовательной сферы. В настоящее время высшие учебные заведения активно позиционируют себя в сети Интернет, уделяя особое внимание развитию своих электронных представительств. Одна из приоритетных целей современного научно-образовательного сообщества – повышение качества обучения.

Веб-сайт вуза может нести на себе множество различных функций. Прежде всего это представительские функции. Образовательное учреждение в современных реалиях РФ вынуждено конкурировать, поэтому вопрос маркетинговой стратегии вуза чрезвычайно важен, а сайт является одним из наиболее мощных инструментов реализации этой стратегии. Кроме того, сайт вуза также может нести образовательную функцию. Под образовательным веб-сайтом понимается совокупность веб-страниц с повторяющимся дизайном, несущих в себе целенаправленный процесс обучения и воспитания в интересах личности, общества, государства, объединенных по смыслу, навигационно и физически находящихся на одном сервере. При этом направленность может быть на поддержку учебного процесса, на поддержку самообучения студентов или собственно на обучение. Такое разнообразие целей и задач порождает множество различных требований, которым должен удовлетворять сайт образовательного учреждения. При этом надо понимать, что сайт также не должен превращаться в «набитый до отказа шкаф», сочетающий в себе чрезмерно большое количество ресурсов, расположенных на малой площади.

В условиях характерного для нашего времени широкомасштабного внедрения в сферу образования информационно-коммуникационных технологий и создания единого информационно-образовательного пространства создание сайтов занимает все большее место в информатизации вузов. Разработка сайтов по праву рассматривается вузами как весьма перспективное направление деятельности. Разработка представляет собой большие возможности для оптимизации их развития в условиях жесткой конкурентной ситуации и открывающей качественно новые перспективы для широкого позиционирования достижений вуза в едином информационно-образовательном пространстве. Обилие требований к сайту наведено внешними вызовами общества к

системе образования в целом. Эти требования перерастают в ещё более комплексные и сложные при наложении на уже сложенную инфраструктуру образовательного учреждения.

Неподдельный интерес к информатизации вузов проявляет государство, что подтверждает Федеральный закон «Об образовании в Российской Федерации», с одной стороны, регламентирует особенности деятельности вузов в информационной экономике при экспоненциально нарастающем объеме информации и глобальном внедрении Интернет, а с другой стороны, определяет изменение условий образовательной деятельности вуза. Профессиональная информация, отражающая знания в среде Интернет, превращается в важнейший нематериальный фактор образовательной деятельности вуза. Возникла и развивается виртуальная среда управления вузом – виртуальная кафедра, виртуальный факультет, виртуальный университет. В соответствии со «Стратегией развития информационного общества в Российской Федерации» сектор виртуальных организаций является одним из приоритетов Правительства РФ. Повышение доли и значимости виртуального образовательного контента обуславливает необходимость пересмотра приоритетов, ролей, структур, содержания, средств и технологий разработки и поддержания функционирования сайтов университетов.

При реализации сайта СГУ были выбраны универсальные решения, в рамках которых уже можно создавать наполнение сайта. В данном случае, сайт – это информационная система, разработанная на базе вуза. Она подразумевает внутри себя поддержание административных процессов, протекающих внутри вуза. На сайте присутствует много одноранговых подразделений, каждое из которых должно плодить один и тот же вид контента, но при этом по правам доступа их нужно разграничивать между собой. Чтобы не возникало ошибок при редактировании контента, таких как, например один ответственный за заполнение изменяет контент другого ответственного.

Целью моей бакалаврской работы является разработка дополнительного модуля, реализующего механизм разграничения прав доступа на официальном сайте СГУ.

Данная цель порождает следующие задачи:

1. Анализ предметной области.

2. Рассмотрение типовых сценариев использования информационной системы.
3. Проектирование механизма разграничения прав доступа.
4. Реализация некоторых принципиальных аспектов.

Структура бакалаврской работы состоит из трех глав. В первой главе "Анализ предметной области" содержится описание инструмента, на котором был реализован сайт, были выдвинуты предположения, почему выбран данный инструмент и приведены типовые сценарии взаимодействия пользователя с системой. Вторая глава "Проектирование информационной системы веб-сайта" содержит UML-диаграммы, которые отражают принципы и механизмы работы системы. Также внутри данной главы приведена инфологическая модель, на основе которой и будет происходить реализация системы. В третьей, завершающей главе, "Реализация некоторых принципиальных компонентов" проводится разработка системы безопасности, показан момент интеграции уже готового решения, а также приведены некоторые принципиально важные участки кода с объяснениями.

2 Основное содержание работы

Для того чтобы разработать программную систему, приносящую реальные выгоды определенным пользователям, необходимо сначала выяснить, какие же задачи она должна решать для этих людей и какими свойствами обладать.

Официальный сайт СГУ реализован на базе системы управления контентом Drupal. Drupal – это открытая система управления содержимым, которая используется для создания веб-сайтов с богатым информационным наполнением. Создание веб-сайта на основе Drupal заключается в комбинировании различных строительных блоков, которые позволяют точно настроить функциональность конкретного веб-сайта.

На вопрос, почему сайт реализован при помощи системы управления контентом можно дать вполне понятный ответ, сайт вуза представляет собой огромную систему, содержащую огромное количество информации. Такую систему нереально реализовать при помощи конструкторов сайтов, а для реализации сайта «с нуля» требуется немало ресурсов. При этом вузу средних размеров (особенно техническому), скорее всего, хватит ресурсов на разработку сайта. Однако после внедрения в эксплуатацию потребуется как поддержка внедрённого решения, так и его развитие. Особняком в вопросах поддержки стоят вопросы безопасности. Как правило, образовательное учреждение не обладает техническим потенциалом ИТ-компаний. Потому подход с написанием сайта «с нуля» в итоге толкает образовательную организацию на один из следующих путей: выведение сайта на аутсорсинг, организация мощного ИТ-отдела, специализирующегося на веб (по сути, открытие внутренней веб-студии), либо использование системы управления контентом (CMS). Самым оптимальным вариантом для СГУ был признан вариант построения сайта на системе управления контентом. При этом в критерии оптимальности были учтены: потенциальная гибкость итогового решения, требуемые трудозатраты, стоимость решения.

Базовой системы обеспечения безопасности в выбранном инструменте по мере роста сайта оказалось недостаточно. Поэтому было решено дополнить базовую систему обеспечения безопасности.

В рамках работы было приведено два типовых сценария взаимодействия пользователей с системой. Данные сценарии помогают понять общий смысл работы системы, а также более конкретные детали реализации, описанные алгоритмически.

Для того, чтобы разобраться во взаимосвязях между элементами в нашей сложной системе, а также компактно и понятно рассказать об исследуемом объекте, его составляющих и функциях используем UML диаграммы. В работе было использовано лишь два вида UML диаграмм: диаграмма вариантов использования и диаграмма взаимодействия.

В качестве графического представления основных функций системы применяется диаграмма вариантов использования. Она дает представление о том, что должна делать система. С помощью данной вида диаграмм было показано откуда возникла необходимость во введении в систему еще одного действующего лица - ответственный.

Диаграмма деятельности позволяет нам ответить на вопрос, как система будет делать определенные в ней действия. Данный вид диаграмм применяется для моделирования динамических аспектов системы. На этих диаграммах была отражена последовательность действий пользователя, а также ответных действий системы. На основе диаграмм, было выяснено, что при совершении каких-либо действий администратором, система не производит каких-либо вспомогательных проверок. Но в случае с ответственным, система перед тем как дать разрешение на выполнение выбранной операции над узлом проводит ряд проверок, удостоверяющих, что данный пользователь имеет доступ к узлу и может выполнять над данным узлом, выбранную операцию.

Самым принципиальным моментом в проектирование информационной системы является проектирование структуры данных, которая позволяет в нормализованном виде хранить информацию. Хранить информацию с минимальной логической избыточностью, которая потенциально может привести к логически ошибочным результатам выборки или изменения данных.

При установке базовой комплектации Drupal, система создает перечень таблиц, которые необходимы для отображения контента на сайте и управления доступом к различным его аспектам. При рассмотрении работы старого метода обеспечения безопасности было выяснено, что данный метод не

справляется с нагрузкой из-за недостатков в организации структуры данных, которая разрабатывалась с целью хранения небольших по вложенности и по количеству элементов меню. В базе данных хранение данной структуры было завязано только на таблице `menu_links`. В данной таблице для каждого меню, хранилось рN его предков, тем самым с одной стороны это хорошо, можно достать сразу необходимого из предков, но данный подход таит в себе потенциальные проблемы, такие как огромные накладные расходы памяти и проблемы с противоречивостью. Проблемы с противоречивостью подразумевают, что если где-то необходимо поменять предка, то грамотно выполнить это получится, если только все его упоминания разом поменять, что невозможно сделать для огромного количества контента.

Поэтому для замены предыдущего механизма, нужно было привести изменения в структуру хранения данных. Для этого была создана таблица `nodehierarchy_menu_links`, которая связывает элементы меню и узлы между собой. Принцип работы доработанного механизма выглядит следующим образом. Первым делом у нас есть номер узла, с которого пришел запрос. Затем на основе этого узла, в таблице `nodehierarchy_menu_links`, находится соответствующий ему идентификатор, который является ссылкой на меню. Затем на основе данной ссылки, в таблице `menu_links` находится идентификатор родительской ссылки меню, и затем процесс повторяется, пока не найдется родитель, либо в теории не достигается корень сайта.

На этапе реализации необходимого решения, было выяснено, что практически все в системе Drupal вращается вокруг понятия модуля, который является файлом с программным кодом на языке PHP, реализующим функциональные возможности. С которыми система Drupal умеет работать. Все функциональные возможности Drupal, доступные администратору или конечному пользователю, от фундаментальных особенностей, таких как возможность выполнять вход в систему, добавлять новое содержимое в динамические фотогалереи и сложные системы голосования, – все они реализованы в виде модулей. Примерами модулей могут служить: модуль Contact, который реализует форму с контактной информацией. Модуль User, который реализует процедуру аутентификации пользователя и проверку прав доступа. В

других системах управления содержимым модули также называются дополнениями или расширениями.

Для реализации механизма разграничения прав, который лежит в основе обеспечения безопасности информационной системы сайта вуза, необходимо было доработать модуль *menu*. В доработку входит изменение структуры данных, а также интерфейса работы с данным модулем. Для того, чтобы сэкономить ресурсы, на создание механизма, было решено использовать уже готовый модуль - модуль Node Hierarchy, разработанный огромным сообществом Drupal. Модуль является обособленным, что отражается его независимостью от других модулей.

Затем в процессе разработки стало понятно, что при работе с базой данных используется множество одинаковых запросов, либо одинаковых последовательностей запросов. Поэтому было решено создать хранимые процедуры. Хранимые процедуры дают возможность соединить вместе последовательность запросов и сохранить их на сервере базы данных. Хранимые процедуры позволяют повысить производительность, расширяют возможности программирования и поддерживают функции безопасности данных. При вызове хранимой процедуры её содержимое сразу же обрабатывается сервером. Кроме собственно выполнения запроса, хранимые процедуры позволяют также производить вычисления и манипуляцию данными — изменение, удаление и вызывать другие хранимые процедуры, выполнять сложную транзакционную логику.

Созданную хранимую процедуру можно вызвать в любой момент, что обеспечивает модульность и стимулирует повторное использование кода. Последнее облегчает сопровождение базы данных, так как она становится изолированной от меняющихся бизнес-правил. Модифицировать хранимую процедуру в соответствии с новыми правилами можно в любой момент. После этого все приложения, использующие её, автоматически придут в соответствие с новыми бизнес-правилами без непосредственной модификации.

Использование хранимых процедур позволяет ограничить или вообще исключить непосредственный доступ пользователей к таблицам базы данных, оставив пользователям только разрешения на выполнение хранимых процедур, обеспечивающих косвенный и строго регламентированный доступ к

данным. Эти функции безопасности позволяют изолировать от пользователя структуру базы данных, что обеспечивает целостность и надежность базы.

Следующим моментом в разработке стало создание прав для пользователей. Были рассмотрены особенности богатства и избирательности системы управления правами доступа, основанной на таких понятиях, как пользователи, роли и разрешения, в системе Drupal. Для реализации была переопределена встроенная функция `hook_permission()`. Этот хук может добавить разрешения, так что наш модуль определит какие из них выбраны на страницы разрешений пользователей. Также хук используется для полного или частичного доступа к действиям (actions) которые выполняет модуль. Данный хук является встроенным в ядро Drupal, он предоставляет очень гибкий и хорошо настраиваемый функционал, ведь в любом модуле, можно описать данный хук, и изменения появятся на странице администрирования.

После создания прав для пользователей, необходимо реализовать управление доступом к узлу на основе этих прав. Для управления доступом к узлу, есть хук `hook_node_access`. Данный хук на вход получает узел, с которым работает пользователь, тип выполняемой операции (например, вставка, обновление, просмотр, удаление) и учетную запись пользователя, выполняющего операцию над узлом. Тем самым при помощи данного хука, мы можем ограничивать пользователей в проводимых над узлами операциях, для поддержания безопасности системы. Результатом выполнения данного хука является значение одной из трех констант доступа.

Заключительным этапом разработки модуля стало создание интерфейса администрирования. Для удобного сопровождения модуля нужен интерфейс администрирования. Администратор не обязательно знает все подробности настройки всех своих сервисов, и у него не всегда есть время на поиск документации по этой теме. Drupal предоставляет удобный интерфейс администрирования контента, управления модулями и другие. Практически каждый разработанный сообществом модуль предоставляет интерфейс для собственной настройки, но для создания специфичных, необходимо прибегнуть к программированию.

3 Заключение

Целью бакалаврской работы являлось разработать обособленный модуль, реализующий механизм разграничения прав доступа на официальном сайте СГУ. Для указанной цели был составлен ряд задач и определены типовые сценарии взаимодействия пользователей с системой. При анализе предметной области удалось выяснить уязвимость выбранного инструмента для реализации информационной системы сайта вуза. На этапе проектирования была разработана дополненная инфологическая модель, которая впоследствии на этапе разработки легла в основу работы новой системы обеспечения безопасности. Этап разработки был разбит на несколько взаимосвязанных частей. Сначала было решено интегрировать в систему уже готовый модуль, который бы организовывал контент в дерево иерархии, так же как и модуль *тепи*, но поддерживающий свойство расширяемости. Затем был разработан обособленный модуль, который бы организовывал механизм разграничения прав на дереве иерархии контента. Таким образом, поставленные задачи решены в полном объеме, работа новой системы обеспечения безопасности соответствует типовым сценариям взаимодействия, а цель бакалаврской работы полностью достигнута.