

Е. И. Сафонов, Д. В. Шиленков, И. Е. Кузнецова

**АВТОМАТИЗАЦИЯ СОСТАВЛЕНИЯ СПРАВКИ ОБ ОБЕСПЕЧЕННОСТИ
УЧЕБНО-МЕТОДИЧЕСКОЙ ДОКУМЕНТАЦИЕЙ
ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ ПО НАПРАВЛЕНИЮ ОБУЧЕНИЯ**

В статье приводится описание автоматизации процесса составления справок об обеспеченности учебно-методической документацией образовательной программы преподавателями Югорского государственного университета. В настоящее время преподаватели заполняют справку вручную в соответствии с рабочей программой дисциплин и учебным планом. При ручном заполнении имеется большая вероятность совершения ошибок, указания недопустимой литературы, недостаточного количества изданий и прочих недочетов. Автоматизация процесса составления справки об обеспеченности учебно-методической документацией образовательной программы позволяет заполнять большинство динамических полей за преподавателя автоматически. Для автоматизации описанного процесса было принято решение разработать веб-приложение. Авторами приводится обзор аналогов. Проведено исследование системы автоматизации библиотек ИРБИС и баз данных электронного каталога Научной библиотеки Югорского государственного университета. Приводится описание используемых при разработке веб-приложения технологий и языков.

Ключевые слова: веб-приложение, MVC, PHP, Laravel Framework, ИРБИС, книгообеспеченность, система автоматизации библиотек.

E. I. Safonov, D. V. Shilenkov, I. E. Kuznetsova

**AUTOMATION OF THE PROCESS TO PREPARE A REFERENCE ON TEACHING-
METHODOLOGICAL DOCUMENTATION OF AN EDUCATIONAL PROGRAM**

The article describes the automation process of preparation references on educational and methodical documentation of educational program elaborated by Yugra State University teaching staff. Currently, all teachers fill in the reference manually in accordance with the working program of a discipline and a study plan. In such case there is a high probability of making mistakes, indicating unacceptable literature sources, insufficient number of publications and other shortcomings. The automation process of preparation references on educational and methodical documentation of educational program allows to fill in automatically more dynamic fields for a teacher. To automate the described process, it was decided to develop a web application. The authors provide an overview of analogues. The study of the automation system of IRBIS libraries and databases of the electronic catalog of Yugra State University library was carried out. The description of the technologies and languages used in the elaboration of the web application is given.

Key words: web application, MVC, PHP, Laravel Framework, IRBIS, Book Supply, Library Automation System.

Введение

Справка об обеспеченности учебно-методической документацией образовательной программы – нормативный документ, устанавливающий наличие и местоположение запрашиваемого документа в фонде библиотеки Югорского государственного университета.

Цель справки – обеспечение преподавателей и обучающихся необходимой в достаточном количестве литературой по дисциплинам учебного плана, а также реализация требований федеральных государственных образовательных стандартов.

В настоящее время в образовательном процессе университета для подсчета книгообеспеченности дисциплин используется утвержденный документ «Справка об обеспеченности учебно-методической документацией образовательной программы», эта справка является неотъемлемой частью основной профессиональной образовательной программы. Справка заполняется вручную в соответствии с дисциплинами учебного плана. Во время выполнения этого процесса в ручном режиме имеется большая вероятность совершения ошибок, указания недоступной литературы и прочих недочетов.

Как следствие, совершение ошибок в процессе составления справок часто приводит к их отклонению при проверке сотрудниками библиотеки. Это в свою очередь ведет к дополнительным трудозатратам и повторным согласованиям справок.

При этом не исключается вероятность повторного отклонения уже исправленной справки.

Автоматизация процесса составления справки об обеспеченности учебно-методической документацией позволяет заполнять большинство динамически изменяющихся полей автоматически. При этом в приложении будет проверяться корректное заполнение справок, такое как выбор обязательного количества основной и дополнительной литературы, возможность библиотеки обеспечить направление обучения необходимой литературой и прочие частые причины отклонения.

Обзор аналогов

Для автоматизации работы библиотеки существуют системы автоматизации библиотек (САБ).

Выделим основные требования к САБ:

1. Система не должна зависеть от операционной системы Windows.
2. Система не должна требовать установки дополнительных компонентов в среду.
3. Интерфейс системы должен быть интуитивно понятен всем пользователям.
4. Должна поддерживаться работа с базой данных SQL-Oracle.
5. Стоимость лицензии должна быть как можно меньше.
6. Система должна быть расширяемой.
7. Система должна иметь достаточно подробную документацию.
8. Система должна иметь электронный каталог.
9. Система должна уметь формировать справку об обеспеченности учебно-методической документацией.

Для сравнения были выбраны следующие системы:

1. САБ «Ирбис».
2. САБ «Руслан».
3. САБ «МегаПро».
4. САБ «Марк-SQL».

После проведения анализа этих систем была построена сравнительная таблица (таблица 1) (сравнение проводилось на основе технических характеристик и субъективного мнения).

Таблица 1 – Сравнение систем автоматизации библиотек

| САБ | «Ирбис» | «Руслан» | «МегаПро» | «Марк-SQL» | Наш сервис |
|-----------------------|------------------------|----------|---------------------------|---------------------------|---------------|
| Платформа | Windows 2000/XP и выше | Linux | WindowsServer 2008 и выше | WindowsServer 2008 и выше | Linux/Windows |
| Требования к ресурсам | Низкие | Средние | Средние | Высокие | Низкие |

| | | | | | |
|--|---|---|---|---|---------------------------------------|
| Модульность, расширяемость | Ограниченно да (дополнительные модули за отдельную плату) | Ограниченно да (дополнительные модули за отдельную плату) | Ограниченно да (дополнительные модули за отдельную плату) | Ограниченно да (дополнительные модули за отдельную плату) | Да |
| Поддержка облачных технологий | Нет | Есть | Есть | Есть | Есть |
| Работа с базами данных | Встроенная | Oracle-SQL | MS SQL 2008 и выше | MS SQL 2008 и выше, Oracle 9 и выше, MS Access | Oracle-SQL, MS SQL, MySQL, PostgreSQL |
| Требуется установка дополнительных компонентов в архитектуру имеющейся системы | Да | Да | Да | Да | Нет |
| Дружелюбный интерфейс | Нет | Нет | Да | Нет | Да |
| Полнота документации | Неполная | Полная | Полная | Полная | Полная |
| Наличие электронного каталога | Есть | Есть | Есть | Есть | Есть |
| Стоимость лицензии для университетов | 0-165000 | 0-237000 | - | 0-360000 | - |
| Анализ обеспеченности учебного процесса | Да | Да | Да | Да | Да |

Сравнив все САБ по необходимым нам характеристикам, мы выяснили, что ни одно из имеющихся готовых решений не удовлетворяет всем выдвинутым требованиям по составлению справок об обеспеченности учебно-методической документацией. В связи с этим возникла необходимость в разработке веб-приложения.

Данное веб-приложение позволит сократить время на составление справки, существенно сократить количество различных отклонений и ошибок, заполняя большинство динамических полей документа автоматически, контролируя корректный выбор литературы, основываясь на данных, полученных из САБ ИРБИС.

Исследование САБ ИРБИС

В процессе исследования структуры САБ ИРБИС были выделены сущности, в которых хранится необходимая литература, доступная для использования в составляемых справках. К таким сущностям относятся:

- FOND – база данных научной библиотеки ЮГУ;
- LAN – база данных ЭБС «Лань»;
- ZNANIUM – база данных ЭБС Znanium.com;
- URAIT – база данных ЭБС «Юрайт».

Также было обнаружено, что САБ ИРБИС имеет свою собственную нереляционную СУБД – IBIS. Этот фактор существенно осложнил процесс разработки веб-приложения, так как эта СУБД не позволяет осуществить прямой доступ к данным литературы.

При исследовании метода обмена информацией с сервером библиотеки была изучена документация к программному обеспечению данного сервера, а именно IRBIS Server. Во время изучения документации было выяснено, что САБ ИРБИС так же, как и разрабатываемое веб-приложение, имеет клиент-серверную архитектуру и выполняет передачу данных между сервером и клиентом, используя сетевой протокол TCP_IP [1].

Разработка началась непосредственно с попытки установить соединение с сервером. С данным этапом проблем не возникло. Следующим шагом стала авторизация на сервере библиотеки. Написанный в соответствии с документацией [2] программный код для формирования и отправки пакетов на сервер долгое время не давал результатов. Информация от пользователей официального форума САБ ИРБИС никак не помогла прояснить ситуацию.

После большого количества безуспешных попыток авторизации на сервере библиотеки было принято решение установить аналогичный сервер в домашней локальной сети. При первой же попытке выполнить авторизацию на локальном сервере был получен положительный результат. Сразу после этого была выполнена повторная попытка авторизации уже на сервере библиотеки. Но, как и все предыдущие попытки, она не увенчалась успехом.

В результате появились предположения о том, что версия сервера, используемая библиотекой ЮГУ, не имеет возможности осуществлять обмен информацией, используя сетевое соединение, или же информация из документации к серверу не соответствует действительности. Для проверки данных теории нужно было определить, каким же образом используемая версия сервера выполняет передачу информации между клиентами. Для этого необходимо было отследить весь трафик, который передается между клиентом и сервером библиотеки от момента запуска клиента до его закрытия. Таким образом был получен формат пакетов, которые отправляет клиент серверу для авторизации.

После проведения анализа данных пакетов была подтверждена теория о том, что информация, описывающая формат отправляемых на сервер пакетов, не соответствует версии сервера, используемого библиотекой, по причине того, что формат пакета, посылаемого клиентом серверу, отличается от формата пакета, указанного в документации. Соответственно, был исправлен код, формирующий отправляемые серверу пакеты.

Разницу между фактическими форматами передаваемых пакетов клиентом и форматом, описанным в документации, можно увидеть на рисунке 1.

| | Формат пакета из документации | | Фактический формат пакета | |
|----|-------------------------------|--------|---------------------------|----------|
| | | | | |
| 1 | Код команды | K | Код команды | K |
| 2 | АРМ | R | АРМ | C |
| 3 | Код команды | K | Код команды | K |
| 4 | Идентификатор | 240954 | Идентификатор | 240954 |
| 5 | Номер команды | 3 | Номер команды | 3 |
| 6 | | | Пароль пользователя | Password |
| 7 | | | Логин пользователя | Username |
| 8 | | | | |
| 9 | | | | |
| 10 | | | | |
| 11 | БАЗА | FOND | БАЗА | FOND |
| 12 | Запрос | QUERY | Запрос | QUERY |
| 13 | Количество записей | 1 | Количество записей | 1 |
| 14 | Начиная с записи | 1 | Начиная с записи | 1 |
| 15 | Формат ответа | @brief | Формат ответа | @ |

Рисунок 1 – Разница между форматами передаваемых пакетов

Интересно то, что сервер отклоняет любые попытки авторизации пользователей и администраторов САБ ИРБИС, используя данные для авторизации, принимаемые клиентами, кроме логина и пароля системного суперпользователя.

Решив проблему с авторизацией, можно было приступить к попыткам поиска литературы. Этот этап начался с изучения языка поисковых запросов САБ ИРБИС, имеющего одноименное название с собственной СУБД IBIS [2]. В процессе изучения конструкции поисковых запросов выяснилось, что ИРБИС способен осуществлять только полнотекстовый поиск литературы и только используя крайне ограниченный список параметров. Среди этих параметров присутствуют:

- автор;
- заглавие;
- ключевые слова.

Для упрощения составления поисковых запросов был спроектирован и разработан генератор поисковых запросов IBIS на языке PHP. Данный генератор безошибочно составлял поисковые запросы по полученным от пользователя данным для поиска. При получении информации о литературе ошибок не возникало.

После того, как был налажен метод получения информации о литературе из системы библиотеки, стало ясно, что необходимо дополнительно разработать синтаксический анализатор результатов поиска. Для этого требовалось изучить формат, в котором сервер возвращает литературу, соответствующую сгенерированному поисковому запросу результата поиска. В первую очередь были определены элементы описания литературы, которые необходимо получить и учитывать при составлении справки об обеспеченности учебно-методической документацией. К таким элементам относятся:

- автор издания;
- заглавие издания;
- вид издания;
- год издания;
- город издания;
- краткое описание;
- количество экземпляров, находящихся на хранении в библиотеке;
- ссылка на издание, если литература относится к электронным ресурсам.

Был написан синтаксический анализатор, способный обнаруживать необходимые элементы в результатах поиска. Однако и здесь не обошлось без подводных камней. А именно время от времени при выполнении поиска на этапе синтаксического анализа выполнение кода программы останавливалось с сообщением об ошибке при попытке получить какой-либо из элементов. При выявлении причин данных ошибок были обнаружены некорректные записи в базе данных. Среди них были:

- отсутствие информации об авторе или заглавии литературы;
- слитые воедино годы издания и переиздания литературы;
- заголовком описания может быть как автор, так и название издания;
- неработающие ссылки на электронные издания.

Для решения сложившихся проблем были приняты меры по усовершенствованию синтаксического анализатора и увеличению его отказоустойчивости. Результатом принятых мер стало повышение точности получения необходимых элементов из результатов поиска с 63 % до 96 %. Причиной появления ошибок в оставшихся 4 % является неверное заполнение полей в базе данных. Также при разработке синтаксического анализатора было выявлено изменение форматов ответа сервера в каждой из последующих версий. Данный факт может негативно сказаться на работоспособности разрабатываемого веб-приложения, лишив его возможности синтаксического анализа, а как следствие, и корректного поиска литературы и отображения пользователям результатов.

Последними этапами реализации функционала разрабатываемого веб-приложения, позволяющего выполнять обмен информацией с сервером библиотеки, стало разлогирование пользователя и разрыв соединения с сервером.

Исследование базы данных ЮГУ

В ходе исследования базы данных были выявлены сущности, которые необходимо задействовать для получения информации о пользователях, направлениях обучения, дисциплинах и годах набора.

Таблица с рабочими программами дисциплин – V_RPD_DISC содержит в себе всю достаточную и необходимую информацию о годах набора, формах обучения, направлениях обучения, кодах направлений обучения, дисциплинах и кодах дисциплин.

Данные о планах набора абитуриентов заимствуются из таблицы V_LR_PlanNabora.

Данные о пользователях берутся из таблицы Appointments. Данная таблица содержит в себе информацию о должностях сотрудников и их ФИО. На момент разработки веб-приложения этих данных достаточно для осуществления авторизации пользователей.

Разработка веб-приложения

Основным языком программирования серверной части разрабатываемого веб-приложения было решено использовать PHP 7.3. Для экономии времени и уменьшения трудозатрат в процессе разработки был выбран фреймворк Laravel.

Во время разработки пользовательского интерфейса (клиентской части) использовались таблица стилей CSS 3.0, язык гипертекстовой разметки HTML 5.0 и один из лидеров среди фреймворков, применяемых для разработки пользовательских интерфейсов, – Bootstrap 4.

Для программирования клиентской части разрабатываемого приложения использовался язык JavaScript.

Беря во внимание то, что в будущем имеется возможность внедрения разрабатываемого веб-приложения в ИС ЮГУ, в частности, как модуль системы ЮГУ «ELIOS», веб-приложение разрабатывается на клиент-серверной архитектуре, повторяя архитектуру ИС «ELIOS», позволяющую пользователям осуществлять доступ к электронным ресурсам серверов ЮГУ. В разрабатываемом приложении используется шаблон проектирования Model-View-Controller (MVC) [3].

Разработанное веб-приложение содержит в себе 7 моделей, 9 контроллеров и 14 представлений.

Опишем основные модели и контроллеры.

Модель рабочих программ – используется для получения из базы данных информации о годах набора, кодах и названиях направлений (в зависимости от полученного от пользователя года набора), кодах и названиях дисциплин (в зависимости от полученных от пользователя года набора и направления обучения). Эти данные необходимы для начала составления справки об обеспеченности учебно-методической документацией по направлению обучения.

Модель справок об обеспеченности учебно-методической документацией – инициализирует подключение к базе данных ЮГУ. Это подключение используется в методах этой же модели на протяжении всей работы с ней. В модели присутствуют методы, принимающие данные от контроллеров для их обработки, составления и выполнения запросов записи в БД или возврата полученной из БД информации. Также эта модель возвращает перечень уже составленных справок и сохраняет промежуточные данные о составляемой или редактируемой справке от момента начала ее составления до момента сохранения в БД при помощи механизма сессии. Фреймворк Laravel имеет встроенный конструктор запросов, который позволяет составлять запросы значительно быстрее в удобной для чтения и восприятия форме.

Контроллер IRBIS – выполняет все операции, связанные с обменом информацией с САБ библиотеки. Данный контроллер включает в себя такие методы, как:

- генерация поискового запроса на языке запросов IBIS;
- подключение к серверу библиотеки;
- авторизация на сервере библиотеки;
- конструктор пакета, отправляемого на сервер библиотеки;

- поиск литературы в базе данных библиотеки;
- отправка пакета серверу библиотеки;
- получение пакетов от сервера библиотеки;
- анализатор полученных от сервера пакетов;
- разлогирование на сервере библиотеки;
- разрыв соединения с сервером библиотеки.

Контроллер IRBIS позволяет осуществлять поиск литературы, необходимой для составления справки об обеспеченности учебно-методической документацией по направлению обучения.

Контроллер синтаксического анализа результатов поиска – занимается определением и получением обязательной информации о найденной литературе (автор, заглавие, вид издания, количество экземпляров в библиотеке, год издания, город издания, краткое описание) для составления справки.

Контроллер фильтрации результатов поиска – доводит до пользователя только те сведения, которые соответствуют всем указанным пользователем параметрам. Дополнительно для экономии времени составителя к поисковым параметрам был добавлен такой параметр, как «Стоп-слова». Этот параметр позволяет исключить из выдачи ту литературу, которая содержит в себе какое-либо из стоп-слов, например, хрестоматия, сборник трудов и т. д.).

К данному контроллеру также были добавлены такие правила фильтрации результатов поиска, как:

- искомая литература должна иметь возраст менее 20 лет;
- нельзя использовать издания, количество экземпляров которых в фонде библиотеке менее 3 экз.

Эти два правила применяются к результатам поиска исключительно в момент составления или редактирования справки.

Контроллер Word – реализует функционал сохранения принятых сотрудниками библиотеки справок об обеспеченности учебно-методической документацией образовательной программы в документ формата MS Word.

Для определения уровня доступа к разрабатываемому приложению выполняется определение роли пользователя путем получения структурного подразделения, к которому он принадлежит. После определения уровня доступа определяются доступные этому пользователю методы и контроллеры.

При помощи механизма сессии осуществляется авторизация пользователей в системе и промежуточное хранение информации о составляемых или редактируемых справках (общая информация о справке (год набора, код и название направления, код и название дисциплины, список литературы и история активности) до момента их записи в БД).

При каждом действии пользователя, связанном с заполнением информации о справке, которое требует получения сведений из базы данных, промежуточного сохранения составленной справки и сохранения справки в базе данных из представления, посылается AJAX-запрос на определенный URL-адрес.

Фреймворк Laravel позволяет сопоставить URL-адреса с конкретным контроллером и его методом, который будет заниматься его обработкой. Данный функционал может быть реализован при помощи встроенных в фреймворк методов маршрутизации [4].

Для того чтобы веб-приложение могло предоставить пользователям возможность загрузки на персональный компьютер составленных ими и принятых сотрудниками библиотеки справок в формате документа MS Word, к разрабатываемому приложению была подключена сторонняя PHP-библиотека – PhpWord, предназначенная для не прямой работы с документами формата MS Word с использованием языка программирования PHP [6]. Также был сформирован шаблон справки по утвержденной форме, который представлен на рисунке 2.

СПРАВКА об обеспеченности учебно-методической документацией программы направления подготовки $\{hr_special_special\}$, $\{hr_special_year\}$ год набора

| № ПП | Наименование дисциплины | Наименование печатных и (или) электронных учебных изданий, методические издания, периодические издания по всем входящим в реализуемую образовательную программу учебным предметам, курсам, дисциплинам (модулям) в соответствии с рабочими программами дисциплин, модулей, практик | | Количество экземпляров | студентов учебной литературой (экземпляров на одного студента) |
|-----------------------------|-------------------------|--|--|------------------------------|--|
| $\{d_discipline_number\}$ | $\{d_discipline\}$ | $\{one_literature\}$ | $\{one_books\} \{one_memory\}$ | $\{One_NumberOfCopies\}$ | $\{One_NumberOfCopiesForStud\}$ |
| | | | $\{one_books_2\} \{one_memory_2\}$ | $\{One_NumberOfCopies_2\}$ | $\{One_NumberOfCopiesForStud_2\}$ |
| | | $\{two_literature\}$ | $\{two_books\} \{two_memory\}$ | $\{Two_NumberOfCopies\}$ | $\{Two_NumberOfCopiesForStud\}$ |
| | | | $\{two_books_2\} \{two_memory_2\}$ | $\{Two_NumberOfCopies_2\}$ | $\{Two_NumberOfCopiesForStud_2\}$ |

Рисунок 2 – Шаблон справки об обеспеченности учебно-методической документацией

После того, как пользователь принимает решение скачать принятую библиотекой справку, на серверной стороне приложения создается переменная «объект типа `PhpWord`», в который загружается шаблон справки. Далее из базы данных загружается вся информация о справке, которую пользователь решил скачать. Из полученных данных формируется массив, включающий в себя следующие данные:

1. **Year** – год набора.
2. **SpecialityCode** – код направления обучения.
3. **Speciality** – название направления обучения.
4. **DisciplineCode** – код дисциплины.
5. **Discipline** – название дисциплины.
6. **Literature** – массив, содержащий в себе всю информацию о литературе.
 - 6.1. **AmountOfLiterature** – общее количество литературы.
 - 6.2. **AmountOfBaseLiterature** – количество основной литературы.
 - 6.3. **AmountOfDLCLiterature** – количество дополнительной литературы.
 - 6.4. **AmountOfBookLiterature** – количество печатной литературы.
 - 6.5. **AmountOfeBookLiterature** – количество электронной литературы.
 - 6.6. **TBook** – массив, содержащий в себе информацию обо всей печатной литературе.
 - 6.7. **EBook** – массив, содержащий в себе информацию обо всей электронной литературе.

Полученные из массива данные вставляются в шаблон в места, соответствующие этим данным. Если какие-либо данные являются пустыми, строка из шаблона, соответствующая этим данным, удаляется.

Если какие-либо переменные из этого массива также являются массивами, выполняется определение количества элементов в этом массиве. По результатам определения количества элементов в массиве выполняется перебор вложенного массива. При этом нужные строки в шаблоне дублируются необходимое количество раз. Таким образом, формируется динамический документ вне зависимости от количества выбранной для скачивания информации.

После того как файл был сформирован и все данные были записаны в соответствующие им места в шаблоне, выполняется сохранение файла на сервере. В случае если файл был успешно сохранен, пользователю возвращается ссылка на скачивание составленного документа.

Заключение

Разработано веб-приложение автоматизированного составления справок об обеспеченности учебно-методической документацией по направлению обучения.

Среднее время заполнения данных по дисциплине в справке с использованием веб-приложения – от 2 до 4 минут. При осуществлении этого процесса вручную среднее время составляло от 10 до 20 минут. Исходя из этого можно сделать вывод, что разработанное приложение позволяет сократить время на заполнение справки в 5 раз.

Также следует отметить, что меры, принятые для сокращения вероятности отклонения библиотекой справки об обеспеченности учебно-методической документацией, тоже продемонстрировали положительный результат и позволили значительно сократить весь процесс составления, проверки и утверждения справки.

Литература

1. Хант, К. TCP/IP. Сетевое администрирование / К. Хант. – 3-е издание. – Санкт-Петербург : Символ, 2007. – 816 с. – ISBN 5-93286-056-1. – Текст : непосредственный.
2. Справка ИРБИС64+ (Система автоматизации библиотек) : сайт. – URL: <http://sntnarciss.ru/irbis/spravka/irbis64.html> (дата обращения 20.01.2019). – Текст : электронный.
3. Реализация MVC паттерна на примере создания сайта-визитки на PHP – Хабр. – Текст : электронный // Хабр : сайт. – 2012. – 27 августа. – URL: <https://habr.com/ru/post/150267/> (дата обращения: 10.01.2019).
4. Документация 5.x. – Текст : электронный // Laravel по-русски : сайт. – URL: <https://laravel.ru/docs/v5> (дата обращения 01.01.2019).
5. GitHub – PHPOffice/PHPWord A pure PHP library for reading and writing word processing document. – Текст : электронный // GitHub : сайт. – URL: <https://github.com/PHPOffice/PHPWord> (дата обращения: 14.03.2019).

