МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информатики

и вычислительной техники

Кафедра информационной безопасности

Отчёт к курсовому проекту

по дисциплине “Безопасность систем баз данных”

**Разработка базы данных для библиотеки**

Выполнили: студенты группы БИ-31

Новоселов Ф. М.

Николаев Р. В.

Лукашов Е. П.

Проверил: доцент кафедры

ИБ Сучков Д.С.

Йошкар-Ола

2020 г.

**СОДЕРЖАНИЕ**

**Введение**3

**1. Техническое задание4**

1.1 Требования к курсовой работе4

1.2 Требования к базе данных4

1.3 Требования к API (минимальное количество реализованных методов)4

**2. Порядок выполнения работы5**

2.1 Этапы разработки базы данных5-9

2.2 Этапы разработки API10-13

**3. Приложения14**

3.1 ER-диаграмма14

3.2 Ссылка на github.com14

**4. Вывод14**

**Введение**

В курсовой работе рассматривается создание базы данных, предназначенной для автоматизации работы служб библиотеки. База данных позволяет клиентам просматривать ассортимент, информацию. Также реализована автоматизация оформления заказа, которая позволяет клиентам формировать заказ.

## **Техническое задание**

## ***1.1 Требования к курсовой работе:***

* Получить структуру данных из файла, согласно варианта. Привести к 3й нормальной форме. Добавить недостающие таблицы.
* Составить ER-диаграмму
* Разработать API для базы данных на любом языке, выполняющемся на стороне сервера (php, ASP.NET, Java, python, node.js, etc)
* Взаимодействие должно осуществляться по клиент-серверной архитектуре, подключение с клиентской программы недопустимо
* Провести настройку пользователей базы данных для разграничения прав доступа, привести пример конфигурации
* Все документы и исходные коды для курсовой работы должны храниться под контролем системы контроля версий — git или mercurial (https://github.com/, https://bitbucket.org/)
* Во время сдачи курсового проекта необходимо предоставить отчет о проделанной работе в печатном виде (отчет)

## ***1.2 Требования к базе данных***

* Наличие не менее 7 таблиц, в том числе таблицы сессий и пользователей
* Структура таблицы должна содержать не менее 3-х полей, одно из которых ключевое
* Правомерное использование типов данных
* Обязательно использование триггеров и/или хранимых процедур
* Форма нормализации не менее 3NF
* Индексирование по полям поиска

## **1.3 Требование к API (минимальное количество реализованных методов)**

* аутентификация пользователя (создание сессии);
* добавление/удаление/изменение данных в таблицах;
* выборка данных их ключевых таблиц по запросам;
* выборка данных из таблиц с объединением результатов.

1. **Порядок выполнения работы**

База данных находится в первой нормальной форме, т.к. в любом допустимом значении отношения каждый его кортеж содержит только одно значение для каждого из атрибутов. Т.к. база данных находится в 1НФ, то она также находится во второй нормальной форме, потому что каждый неключевой атрибут неприводимо (функционально полно) зависит от её потенциального ключа. Наличие 2НФ и отсутствие зависимости неключевых атрибутов от ключевых доказывает, что база данных находится в третьей нормальной форме.

* 1. Этапы разработки базы данных

Разработана база данных, содержащая 8 таблиц. В том числе таблицы сессий (*lib\_session*) и пользователей (*lib\_user*). Для авторизации пользователя используется таблица - *lib\_accounts,* содержащая информацию об аккаунте пользователя. Для хранения информации об ассортименте используются следующие таблицы:

*lib\_book* – название и цена книги;

*lib\_list –* название книг и информация о них;

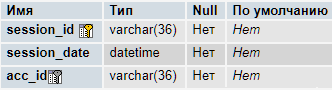
Для автоматизации оформления заказа, были созданы следующие таблицы:

*lib\_availability –* баланс и платежные данные покупателя;

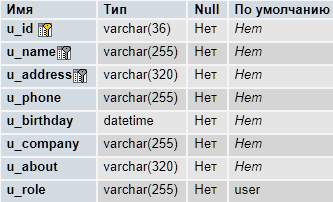
*lib\_cart* – информация о книгах, добавленных в корзину;

*lib\_orders* – оформленные заказы на доставку книг;

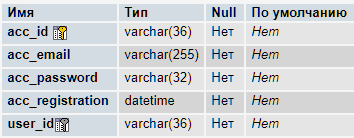
Структуры, реализованных таблиц:



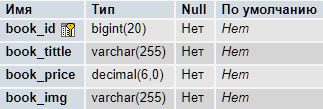
*Рис 1. lib\_session*

**

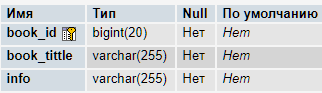
*Рис 2. lib\_user*

**

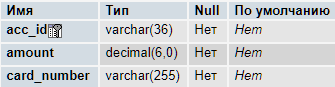
*Рис 3. lib\_accounts*

**

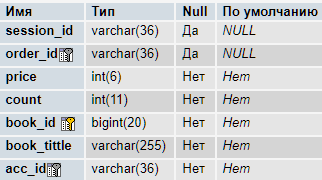
*Рис 4. lib\_book*

**

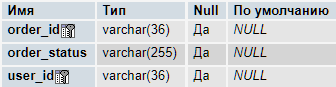
*Рис 5. lib\_list*

**

*Рис 6. lib\_availability*

**

*Рис 7. lib\_cart*

**

*Рис 8. lib\_orders*

Используемые триггеры:

* *drop\_order –* триггер отвечающий за удаление товаров из корзины;

*DROP TRIGGER IF EXISTS `drop\_order`;*

*DELIMITER ;;*

*CREATE TRIGGER `drop\_order`*

*BEFORE DELETE ON `lib\_orders` FOR EACH ROW*

*BEGIN*

*DELETE FROM ` lib\_cart`*

*WHERE ` lib\_cart`.order\_id = OLD.order\_id;*

*END*

*;;*

*DELIMITER;*

* *generate\_id –* триггер отвечающий за генерацию id аккаунта;

DROP TRIGGER IF EXISTS `*generate\_id*`;

DELIMITER ;;

CREATE TRIGGER `*generate\_id*`

BEFORE INSERT ON `*lib\_accounts*` FOR EACH ROW

BEGIN

IF NEW.acc\_id IS NULL THEN

SET NEW.acc\_id = UUID();

END IF;

END

;;

DELIMITER;

Созданные индексы:

* в таблице lib\_accounts



* в таблице lib\_availability



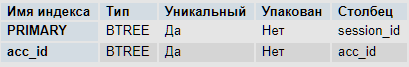
* в таблице lib\_cart



* в таблице lib\_orders



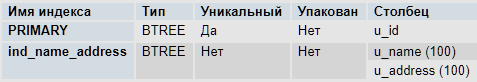
* в таблице lib\_session



* в таблице lib\_list



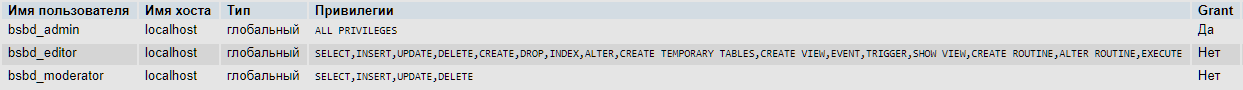
* в таблице lib\_user



* в таблице lib\_book



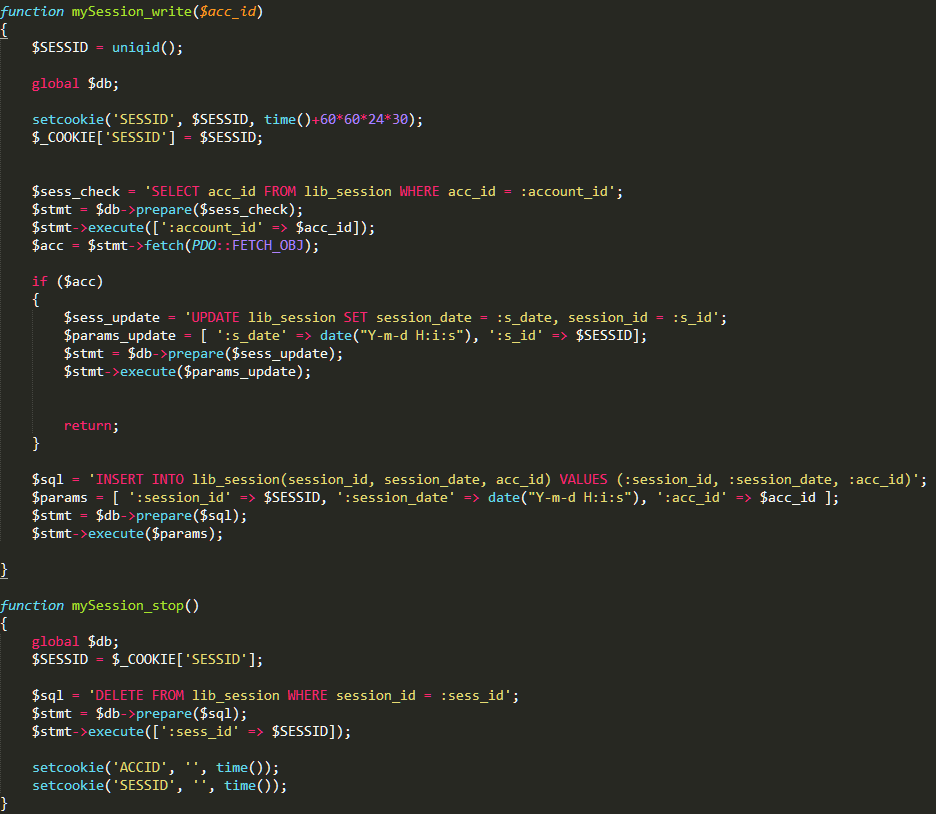
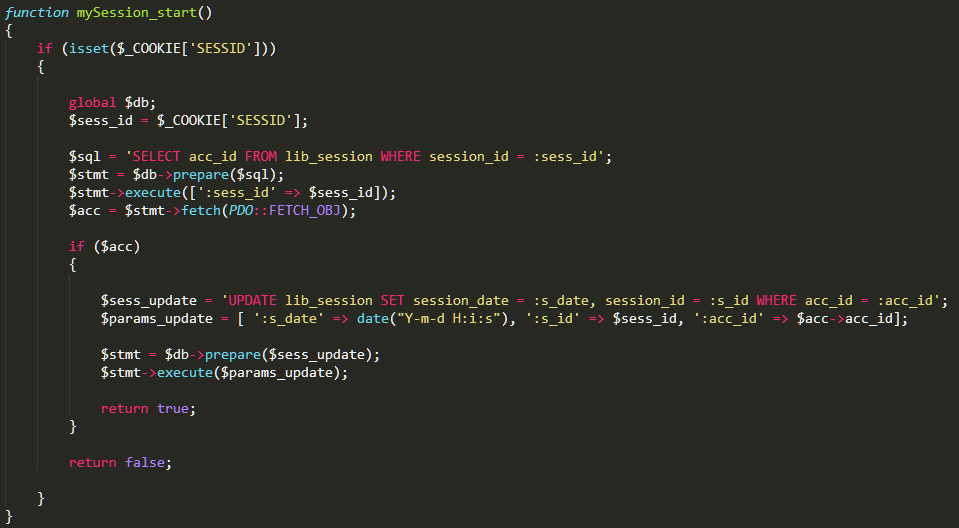
Проведена настройка пользователей базы данных для разграничения прав доступа:



*Рис 9. Пользователи базы данных*

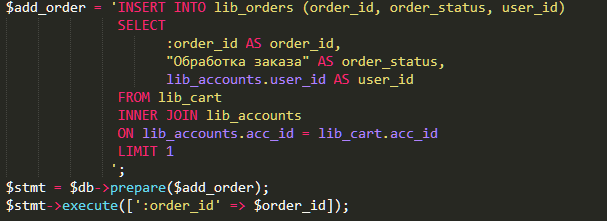
2.2 Этапы разработки API

Было разработано API для аутентификации пользователя, состоящий из 3 функций, написанных на языке PHP.



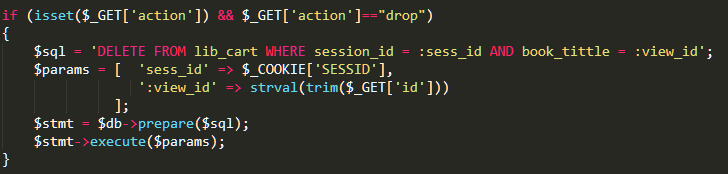
*Рис 10. Аутентификация пользователя*

Разработана функция добавления, удаления и изменения данных.



Добавлена функция вывода и объединения данных из ключевых таблиц.

Рис.11 Добавление и изменение данных



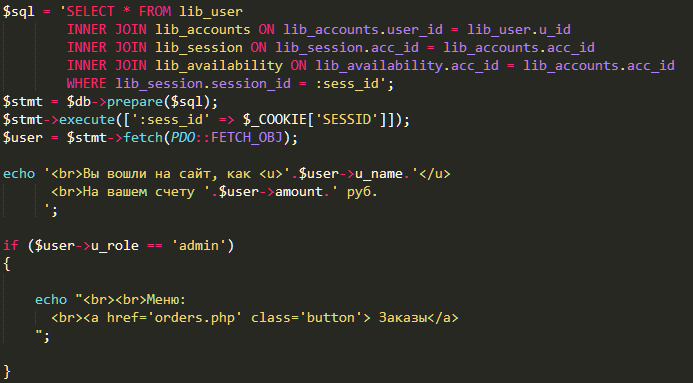


Рис.13 Вывод данных по заказам

Рис.12 Удаление данных

Пользователи обладают разными ролями на сайте. В роли админа есть функция просмотра/изменения статуса/удаления заказов для всех пользователей. Для пользователя доступны только функции авторизации на сайте и добавление товаров в корзину.

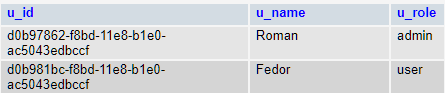
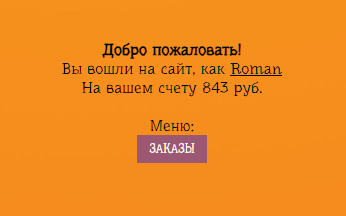
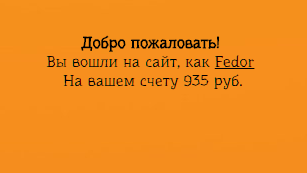




Рис.14 Корзина покупателя



*Рис.15 Личный кабинет администратора*

*  
Рис.16 Личный кабинет пользователя*

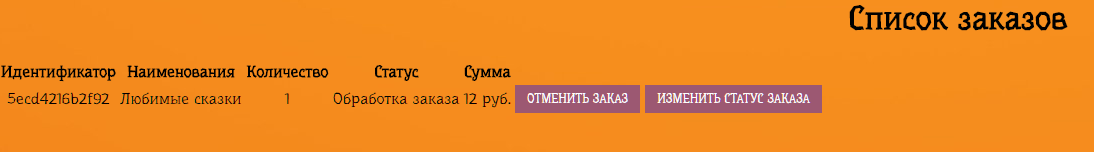


Рис.17 Список заказов (доступен только администратору)



Рис.18 Главная страница

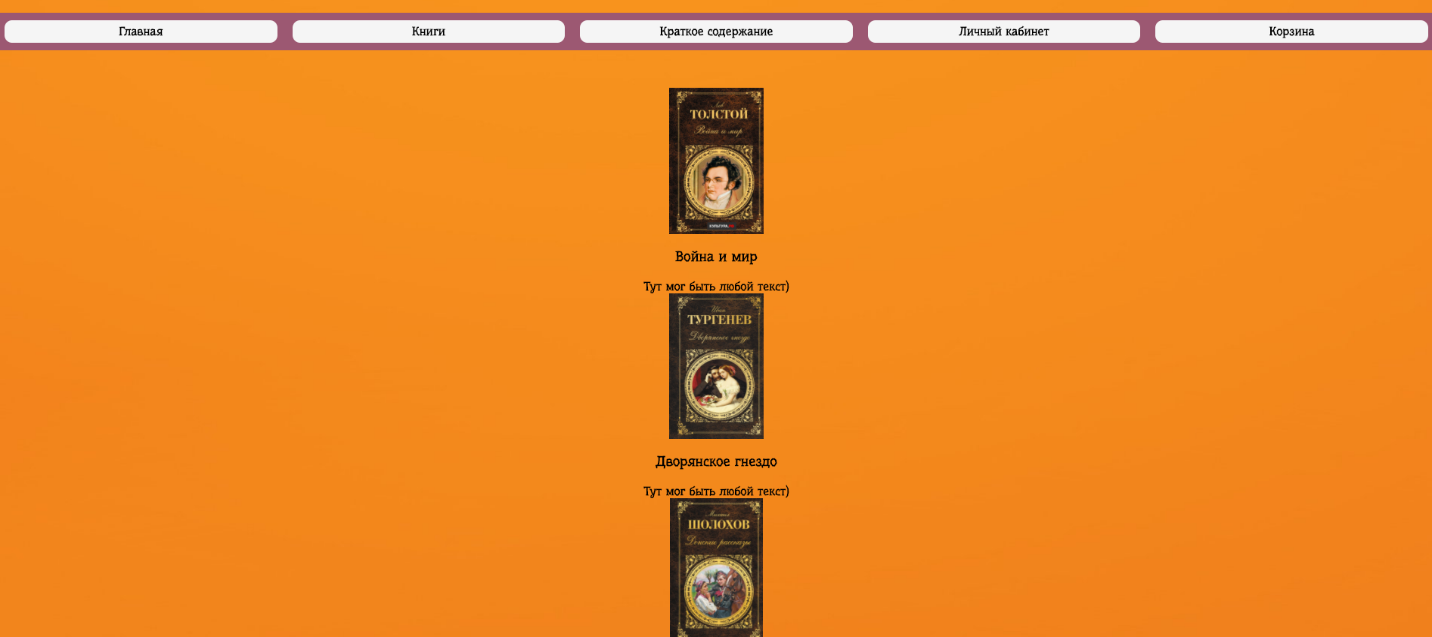
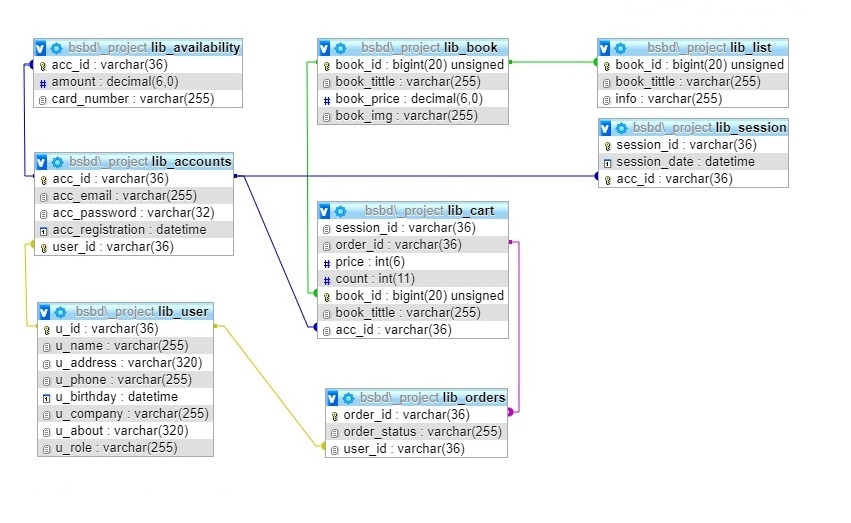


Рис.19 Краткое содержание

**Приложения**

1. ER-диаграмма



1. Исходные коды и документы:

<https://github.com/Batman2RED/bsbd>

**Вывод**

Во время выполнения курсового проекта были изучены методы работы с базами данных, способы управления. Таким образом, в ходе работы была разработана база данных, а также API, для автоматизации работы службы доставки.