

Министерство науки и высшего образования  
Российской Федерации

Федеральное Государственное  
Автономное Образовательное Учреждение  
Высшего Образования  
Национальный ядерный университет «МИФИ»

Кафедра: «Финансовый мониторинг»

Техническое задание

Редакция 5

Студент Монастырский М. О.

Группа С21-703

Москва 2023г.

## Содержание

<b>1. Введение.....</b>	<b>5</b>
<b>1.1 Название программы.....</b>	<b>5</b>
<b>1.2 Краткая характеристика области применения. ....</b>	<b>5</b>
<b>2. Основание для разработки.....</b>	<b>5</b>
<b>3. Назначение разработки.....</b>	<b>5</b>
<b>3.1 Функциональное назначение.....</b>	<b>5</b>
<b>3.2 Эксплуатационное назначение.....</b>	<b>6</b>
<b>4. Технические требования к программе. ....</b>	<b>6</b>
<b>4.1 Требования к функциональным характеристикам.....</b>	<b>6</b>
<b>4.1.1 Функция «выдать книгу читателю в зал или на руки». ....</b>	<b>6</b>
<b>4.1.2 Функция «оформить поступление новой книги». ....</b>	<b>7</b>
<b>4.1.3 Функция «получить и подтвердить предварительный заказ книги».....</b>	<b>8</b>
<b>4.1.4 Функция «напомнить о возврате книги» .....</b>	<b>8</b>
<b>4.1.5 Функция «Наложить/снять штраф».....</b>	<b>8</b>
<b>4.1.6 Функция «заблокировать читателя».....</b>	<b>9</b>
<b>4.1.7 Функция «Оформить потерю или порчу книги».....</b>	<b>10</b>
<b>4.1.8 Функция «Редактировать книгу (страницу книги в системе учета)».....</b>	<b>10</b>
<b>4.1.9 Функция «Найти книгу в хранилище или читальных залах».</b>	<b>11</b>
<b>4.1.10 Функция «Списать книгу».....</b>	<b>11</b>

4.1.11	Функция «Регистрировать студента, сотрудника, как читателя».	11
4.1.12	Функция «Формировать и распечатать формуляры студентов».	12
4.1.13	Функция «Формировать и распечатать читательские билеты для читателей».	12
4.1.14	Функция «Формировать и распечатать наклейки для книг».	13
4.1.15	Функция «Работа со сканером штрих-кодов».	13
4.1.16	Функция « по каталогу книг в т. ч. по каталогу выданных на данный момент».	14
4.1.17	Функция «Вернуть книгу в библиотеку».	14
4.1.18	Функция «Редактировать читателя».	15
4.2.1	Требования к обеспечению надежного (устойчивого) функционирования программы.	16
4.2.2	Восстановление после отказа.	16
4.2.3	Отказы из-за некорректных действий оператора.	16
4.3	Условия эксплуатации.	16
4.4	Требования к составу и параметрам технических средств.	17
4.4.1	Серверная часть.	17
4.4.2	Конечное устройство пользователя.	17
4.5	Требования к информационной и программной совместимости.	17
4.7	Требования к транспортированию и хранению.	17
5	Требования к программной документации	18
6	Технико-экономические показатели	18
5	Стадии и этапы разработки	18

7.1 Стадии разработки, этапы и содержание работ .....	18
7.1 Сроки разработки .....	20
7.2 Исполнители .....	20
6 Порядок контроля и приемки .....	20
8.1 Виды испытаний .....	20
8.2 Общие требования к приемке работ .....	20
9. Приложения .....	21

## **1. Введение.**

### **1.1 Название программы.**

Название программы: «StudLIB»

### **1.2 Краткая характеристика области применения.**

Данная программа предназначена для автоматизации учета книг и других материалов в библиотеке ВУЗа. Она позволяет быстро и удобно осуществлять поиск, выдачу, возврат и учет книг, а также контролировать их наличие и состояние. Программа также предоставляет возможность формирования отчетов о наличии книг, задолженностях студентов и другой статистической информации, что позволяет библиотекарям эффективно управлять библиотечным фондом и обеспечивать быстрый доступ к необходимым материалам для студентов и преподавателей.

## **2. Основание для разработки.**

Основанием для разработки данного продукта является задание, выданное в рамках курса «Стандартизация, сертификация и управление качеством программного обеспечения».

## **3. Назначение разработки.**

### **3.1 Функциональное назначение.**

Программа представляет собой инструмент для управления библиотечным фондом с позиции библиотекаря, она позволяет управлять выдачей\получением книг, списком выданной литературы, списком читателей и позволяет выполнять некоторые заранее определенные функции с профилями читателей и книгами, такие как списание книг, регистрация новых, блокировка читателей, наложения и снятие штрафов, а также возможность вывода всей информации на бумажный носитель.

### **3.2 Эксплуатационное назначение.**

Программное обеспечение, созданная в соответствии с настоящим ТЗ, позволит библиотеке планировать списание и поступление новой литературы, контролируя ее доступность для читателей. А также упростит связь с ними за счет плотной интеграции ПО с менеджерами, которые позволят оперативно следить за выполнением заказов, снизят задержки выдачи и сдачи литературы.

## **4. Технические требования к программе.**

### **4.1 Требования к функциональным характеристикам.**

#### **4.1.1 Функция «выдать книгу читателю в зал или на руки».**

Для использования данной функции библиотекарю необходимо отсканировать штрих код читательского билета/ ввести его вручную/ в поле поиска при помощи оборудования, предусмотренного на рабочем месте библиотекаря.

При этом программное обеспечение открывает профиль читателя и проверяет его доступ к услугам библиотеки. Если доступ есть, то на странице читателя появится возможность добавления новой литературы с помощью графического интерфейса. Иначе, появится соответствующая надпись о том, что доступ читателю запрещен.

После вызова функции добавления библиотекарь заполнит форму поиска нужной книги и поля дополнительных настроек (тип выдачи: на дом или в читальный зал; дату возврата книги в библиотеку).

Необходимо предусмотреть автоматическую установку даты на конец текущего учебного года (31.08.xx)).

Дата возврата должна быть в будущем относительно текущей даты и валидироваться по принципу dd.mm.yy

Поиск книги должен осуществляться согласно функционалу, описанному в п (4.1.9) Технического Задания. Для завершения операции выдачи книги библиотекарь должен отсканировать ее штрих код или вручную подтвердить выдачу. Последним этапом настоящая функция должна вызвать функцию Печати формуляра студента (см. 4.1.12), предварительно внося в него изменения.

По результатам выполнения функции на экран выводится сообщение о статусе выполненной операции, которое затем должно исчезнуть без участия пользователя, если операция была успешна. Иначе, в сообщении должна быть отражена суть ошибки.

#### **4.1.2 Функция «оформить поступление новой книги».**

На основном экране программы графическим интерфейсом должна быть предусмотрена быстрая возможность вызова этой функции, по факту вызова появляется форма заполнения информации о новой книге (см прил. 2). Во время ее заполнения библиотекарю предлагаются данные для автозаполнения полей автор и издательство, а поле ISBN должно быть валидировано по общепринятому формату. Поле штрих-код создается автоматически и является уникальным неизменяемым значением. После заполнения функция должна сделать книгу доступной для выдачи (если во время оформления не был установлен иной статус книги), распечатать штрих-код, который можно вклеить в форзац книги.

По результатам выполнения функции на экран выводится сообщение о статусе выполненной операции, которое затем должно исчезнуть без участия пользователя, в случае если операция была успешна, иначе в сообщении должна быть отражена суть ошибки.

### **4.1.3 Функция «получить и подтвердить предварительный заказ книги»**

Заявки хранятся в таблице заявок (см. прил. 2).

Текущие заявки отображаются в соответствующем разделе графического интерфейса основной страницы программы.

После получения заявки, библиотекарь должен ее подтвердить. Когда книга будет доступна для заказа, графический интерфейс страницы заявки должен позволить библиотекарю забронировать книгу и отправить читателю уведомление о готовности выдать ее, при этом распечатать в книгу вкладыш с номером заявки, таким образом статус заявки изменится на «готова к выдаче», по факту выдачи библиотекарь должен повторно подтвердить заявку, тем самым изменив ее статус на «выдана»

### **4.1.4 Функция «напомнить о возврате книги»**

Функция выполняется автоматически за определенное количество дней, установленное библиотекарем в настройках программы, а также за 1 день до конца срока сдачи.

Уведомление может быть отправлено через мессенджер Telegram по номеру телефона в профиле читателя.

### **4.1.5 Функция «Наложить/снять штраф».**

Функция вызывается автоматически в случае, если книга не была возвращена вовремя, при этом читателю направляется уведомление с указанием причины и суммы штрафа аналогично п. 4.1.4.

Штраф может быть рассчитан в соответствии с правилами библиотеки, на основании стоимости книги (см. прил. 2) и должен быть зарегистрирован в базе данных. Если книга была возвращена и штраф был оплачен, функция



предоставляет возможность библиотекарю снять штраф через графический интерфейс профиля читателя, либо уменьшить его сумму в случае неполной оплаты.

Кроме автоматического наложения штрафов необходимо предусмотреть возможность библиотекарю наложить штраф с помощью выбора книги в профиле читателя и установки ей статуса (утеряна/испорчена). Функция должна автоматически наложить штраф, уведомить читателя, затем вызвать функцию 4.1.7 настоящего Технического Задания.

По результатам выполнения функции на экран выводится сообщение о статусе выполненной операции, которое затем должно исчезнуть без участия пользователя, если операция прошла успешно, иначе в сообщении должна быть отражена суть ошибки.

#### **4.1.6 Функция «заблокировать читателя»**

Для использования функции библиотекарь должен на странице читателя установить его статус как заблокированного пользователя средствами графического интерфейса. После чего в специальной форме указать причину. При блокировке доступа к услугам библиотеки, функция должна уведомить читателя об этом.

После истечения срока блокировки, функция автоматически разблокирует доступ к услугам библиотеки и уведомит читателя о возобновлении доступа.

По результатам выполнения функции на экран выводится сообщение о статусе выполненной операции, которое исчезает без участия пользователя, если операция была успешна, иначе в сообщении должна быть отражена суть ошибки.

#### **4.1.7 Функция «Оформить потерю или порчу книги».**

Функция автоматическая, без графического представления.

После регистрации потери или повреждения книги в базе данных функция должна обновлять информацию о количестве доступных экземпляров каждой книги и убрать поврежденный экземпляр из выдачи.

По результатам выполнения функции на экран должно выводиться сообщение о статусе выполненной операции, которое должно исчезнуть без участия пользователя, если операция была успешна, иначе в сообщении должна быть отражена суть ошибки.

#### **4.1.8 Функция «Редактировать книгу (страницу книги в системе учета)».**

Функция должна предоставлять возможность изменять информацию о книге средствами графического интерфейса на персональной странице данной книги, это название, автор, издательство, год издания, ISBN и т. д. (см. таб.1 Прил. 2).

После редактирования информации о книге в базе данных, функция должна обновлять информацию.

Для ее использования библиотекарь необходимо открыть страницу книги, выбрав ее через поиск и нажав на ней кнопку редактирования. Ограничения наследуются из 4.1.2.

По результатам выполнения функции на экран должно выводиться сообщение о статусе выполненной операции, которое затем должно исчезнуть без участия пользователя, если операция была успешна, иначе в сообщении должна быть отражена суть ошибки.

#### **4.1.9 Функция «Найти книгу в хранилище или читальных залах».**

Функция должна предоставлять возможность поиска книги по различным параметрам средствами функции п 4.1.16.

После нахождения нужной книги в базе данных, функция должна предоставлять информацию о том, где находится книга: в библиотеке, в хранилище или на полке в читальном зале, местоположение в зале должно выводиться в формате (зал, полка).

В качестве графического представления результатов данной функции следует использовать поле «статус» на персональной странице экземпляра.

#### **4.1.10 Функция «Списать книгу».**

Вызов функции удаления вызывается с личной страницы экземпляра, средствами графического интерфейса. После вызова функции появляется форма, в которой указывается причина списания.

После удаления информации о книге из базы данных, функция должна обновлять информацию о количестве доступных экземпляров каждой книги.

По результатам выполнения функции на экран должно выводиться сообщение о статусе выполненной операции, которое затем должно исчезнуть без участия пользователя, если операция была успешна, иначе в сообщении должна быть отражена суть ошибки.

#### **4.1.11 Функция «Регистрировать студента, сотрудника, как читателя».**

Для добавления нового читателя на основной странице программы должна быть предусмотрена кнопка, открывающая форму добавления нового читателя, в которой необходим уникальный номер человека (номер зачетной книги) и номер телефона, на который будут отправляться уведомления, остальная информация будет внесена путем использования внешнего

источника в виде БД ВУЗа приложение 1, при этом контактная информация должна обязательно валидироваться и иметь общепринятый вид:

**+7(900)111-11-11**

После регистрации нового читателя в базе данных, функция должна распечатать заполненный читательский билет и внести новую запись в базу данных читателей, открыв доступ последнему к услугам библиотеки.

По результатам выполнения функции на экран должно выводиться сообщение о статусе выполненной операции, которое должно исчезнуть без участия пользователя, если операция была успешна, иначе в сообщении должна быть отражена суть ошибки.

#### **4.1.12 Функция «Формировать и распечатать формуляры студентов».**

Для вызова данной функции имеется два сценария:

1. Функция вызывается изнутри других функций, как это описано в Техническом задании.
2. Функция может быть вызвана по запросу библиотекаря средствами графического интерфейса автоматизированной системы со страницы профиля читателя.

После создания формуляра, функция должна предоставлять возможность распечатать его и выдать студенту вместе с книгами. Формуляр должен содержать информацию о выданных книгах, датах выдачи и возврата, а также о штрафах за просроченный срок возврата.

#### **4.1.13 Функция «Формировать и распечатать читательские билеты для читателей».**

Для вызова данной функции имеется два сценария:

1. Функция вызывается изнутри других функций, как это описано в Техническом задании.
2. Функция может быть вызвана по запросу библиотекаря средствами графического интерфейса автоматизированной системы со страницы профиля читателя.

После создания читательского билета, функция должна предоставлять возможность распечатать его и выдать студенту. Читательский билет должен содержать: ФИО студента, дату рождения, контактную информацию и другие данные, необходимые для идентификации читателя, а также штрих-код для быстрого доступа через АС библиотеки.

#### **4.1.14 Функция «Формировать и распечатать наклейки для книг».**

5. Функция вызывается изнутри других функций, как это описано в Техническом задании.
6. Функция может быть вызвана по запросу библиотекаря средствами графического интерфейса автоматизированной системы со страницы книги.

После создания наклейки, функция должна предоставлять возможность распечатать ее и приклеить на книгу. Наклейка должна содержать буквенно-цифровое обозначение книги и его аналог в виде штрих-кода для быстрого доступа через АС библиотеки.

#### **4.1.15 Функция «Работа со сканером штрих-кодов».**

Функция должна предоставлять возможность сканировать штрих-коды на книгах и читательских билетах, чтобы получить информацию о книге или читателе. Сценарии использования, совместные с другими функциями, были описаны выше в п 4.1 Технического Задания.

После сканирования штрих-кода, функция должна предоставлять доступ к персональным страницам или читателей, или книг.

#### **4.1.16 Функция «Поиск по каталогу книг в т. ч. по каталогу выданных на данный момент».**

Функция «поиск по каталогу книг» позволяет библиотекарям быстро находить нужную книгу в базе данных библиотеки. Для использования функции необходимо иметь доступ к каталогу книг, который содержит информацию о названии, авторе, издательстве, годе издания и других характеристиках книг, указанные в таблице в приложении 2, а также их совокупности, полученной с использованием логического И. Также следует учесть возможность фильтрации книг по статусу и возможность упорядочивать их по различным параметрам.

Книги, выданные на момент поиска, должны отображаться в специальном разделе поиска, в случае, если они должны быть сданы в ближайшие N<sup>1</sup> дней.

#### **4.1.17 Функция «Вернуть книгу в библиотеку».**

Функция «Возврат книги в библиотеку» позволяет пользователям вернуть книги, которые они брали из библиотеки, обратно на полки. Эта функция включает в себя проверку состояния книги библиотекарем, после чего посредством граф интерфейса он может отметить книгу как поврежденную, если это имело место быть, в ответ функция выдает место, куда следует поместить книгу и ставит отметку в системе о том, что книга была возвращена. По результатам работы функции должно возникать уведомление со статусом операции, аналогично пунктам выше.

---

<sup>1</sup> Число N – число дней установленное в настройках, как количество дней, за которое программа автоматически уведомляет читателя о необходимости сдать книгу.

Помимо этого, в Программе должен предусматриваться вариант вызова функции автоматически, при условии, что был отсканирован штрих-код книги, которая была на данный момент выдана.

После этого функция вызывает «Печать формуляра» с внесенными в него изменениями

#### **4.1.18 Функция «Редактировать читателя».**

Функция должна предоставлять возможность изменять информацию о читателе средствами графического интерфейса на персональной странице данного читателя (см. таб.1 Прил. 2).

После редактирования информации о читателе в базе данных, функция должна обновлять информацию.

Для ее использования библиотекарю необходимо открыть страницу читателя, выбрав ее через поиск и нажав на ней кнопку редактирования. Ограничения наследуются из 4.1.11.

По результатам выполнения функции на экран должно выводиться сообщение о статусе выполненной операции, которое затем должно исчезнуть без участия пользователя, если операция была успешна, иначе в сообщении должна быть отражена суть ошибки.

## **4.2. Требования к надежности.**

### **4.2.1 Требования к обеспечению надежного (устойчивого) функционирования программы.**

Пользователю, работающему с программой, должен быть предоставлен непрерывный доступ к приложению и необходимым для его работы базам данных, находящимся во внутренней сети ВУЗа и недоступным извне. Сервис не должен непредвиденно прерывать свою работу.

### **4.2.2 Восстановление после отказа.**

В случае отказа работы приложения по любой из причин и последующей недоступности приложения, время восстановления не должно превышать одни рабочие сутки.

### **4.2.3 Отказы из-за некорректных действий оператора.**

После запуска программы отключение программы вследствие некорректных действий оператора должно быть исключено. В случае попытки отключения, программный продукт должен требовать от оператора подтверждения его действий средствами графического интерфейса.

## **4.3 Условия эксплуатации.**

Специальные требования не предъявляются.



#### **4.4 Требования к составу и параметрам технических средств.**

##### **4.4.1 Серверная часть.**

Минимальные аппаратные требования: Процессор Linux совместимый, с тактовой частотой не менее 2 GHz, оперативная память не менее 1024 МБ, и свободное дисковое пространство не менее 10 ГБ, обозначенная техника должна быть пригодна для работы приложения в многопоточном режиме

##### **4.4.2 Конечное устройство пользователя.**

Конечное устройство пользователя должно иметь разрешение экрана не менее 800x600 точек, иметь не менее 256 Мб оперативной памяти и любой процессор тактовой частотой не менее 500 MHz.

Дополнительно к вышеописанному, АРМ комплектуется сканером штрих кодов и принтером бумаги формата А4.

#### **4.5 Требования к информационной и программной совместимости.**

Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией операционной системы Astra Linux версии актуальной на момент согласования настоящего документа.

#### **4.6 Требования к маркировке и упаковке.**

Требования к упаковке и маркировке продукта не предъявляется.

#### **4.7 Требования к транспортированию и хранению.**

Требования к транспортировке продукта не предъявляется.

## **5 Требования к программной документации**

Состав программной документации включает в себя:

1. Спецификация, разработанная в соответствии с ГОСТом 19.202-78;
  2. Текст программы, разработанный в соответствии с ГОСТом 19.401-78;
- Также необходимо предоставить заказчику:
3. Техническое задание, разработанное в соответствии с ГОСТом 19.201-78;
  4. Описание программы, разработанное в соответствии с ГОСТом 19.402-78;
  5. Программа и методика испытаний, разработанные в соответствии с ГОСТом 19.201-78 19.301-79 ЕСПД;
  6. Руководство по техническому обслуживанию, разработанное в соответствии с ГОСТом 19.508-79;
  7. Руководство программиста, разработанное в соответствии с ГОСТом 19.504-79;
  8. Руководство оператора, разработанное в соответствии с ГОСТом 19.505-79;
  9. Протокол проведения испытаний;
  10. Акт приема-передачи программно-информационного обеспечения.

## **6 Технико-экономические показатели**

Технико-экономические показатели не рассчитываются

## **5 Стадии и этапы разработки**

### **7.1 Стадии разработки, этапы и содержание работ**

Согласно ГОСТ 19.102-77 ЕСПД. Стадии разработки, этапы и содержание работ должны соответствовать указанным в таблице.

Таблица 1. Стадии и этапы разработки

Стадии разработки	Этапы разработки	Содержание работ по этапам
1. Техническое задание	Обоснование разработки	Постановка задачи
		Сбор исходных материалов
	Разработка утверждение технического задания	Определение требований программе
		Выбор языков программирования
		Согласование и утверждение технического задания
2. Эскизный проект	Разработка эскизного проекта	Предварительная разработка структуры входных и выходных данных
		Определение структуры информационной составляющей
		Разработка общего описания алгоритма решения задачи
	Утверждение эскизного проекта	Согласование и утверждение эскизного проекта
3. Технический проект	Разработка технического проекта	Уточнение структуры входных и выходных данных
		Разработка структуры информационной составляющей
		Разработка алгоритма решения задачи
		Определение формы представления входных и выходных данных
		Разработка структуры программы
		Окончательное определение конфигурации технических средств
	Утверждение технического проекта	Разработка плана мероприятий по разработке и внедрению программ
		Согласование и утверждение технического проекта
4. Рабочий проект	Разработка программы	Программирование и отладка программы
	Разработка программной документации	Разработка программных документов

	Испытание программы	Разработка, согласование и утверждение программы и методики испытаний
		Корректировка программы и программной документации по результатам испытаний
5. Внедрение	Подготовка и передача программы	Подготовка и передача программы и программной документации для сопровождения и (или) изготовления
		Оформление и утверждение акта о передаче программы на сопровождение и (или) изготовление

### **7.1 Сроки разработки**

Начало разработки: 27.09.2023

Окончание разработки: 27.12.2023

### **7.2 Исполнители**

Ответственным исполнителем со стороны Разработчика является студент группы С21-703  
Монастырский М. О.

## **6 Порядок контроля и приемки**

### **8.1 Виды испытаний**

Приемосдаточные испытания должны проводиться на объекте Заказчика в сроки не превышающие С \_\_\_\_\_ ПО \_\_\_\_\_.

Приемосдаточные работы должны проводиться согласно разработанной Исполнителем и согласованной с Заказчиком Программы методик испытаний.

### **8.2 Общие требования к приемке работ**

На основании Протокола проведения испытаний Исполнитель совместно с Заказчиком подписывает Акт приема-сдачи программы в эксплуатацию.

## 9. Приложения

### Приложение 1: «Структура таблицы в базе данных ВУЗа, содержащей основные данные»

Таблица: person

id - уникальный идентификатор (тип данных SERIAL)

first\_name - имя (тип данных VARCHAR(50))

last\_name - фамилия (тип данных VARCHAR(50))

date\_of\_birth - дата рождения (тип данных DATE)

email - адрес электронной почты (тип данных VARCHAR(255))

phone\_number - номер телефона (тип данных VARCHAR(20))

address - адрес проживания (тип данных VARCHAR(255))

created\_at - дата и время создания записи в базе данных (тип данных  
TIMESTAMP WITHOUT TIME ZONE )

### Приложение 2

**Таблица 2 «существующие таблицы в информационной системе  
библиотеки на момент написания технического задания»**

№	Объект	Атрибуты	Взаимосвязь
1	Book	1. название varchar (50) 2. автор varchar (50) 3. ISBN varchar (30) 4. описание varchar (1000)	зал

		5. издательство varchar (50) 6. год издания varchar (5) 7. оригинальное название, если книга – перевод varchar (50) 11. кол-во integer 12. читальный зал/хранилище boolean 13. место расположения (полка) integer 14. ключевые слова для поиска 15 стоимость integer 16. Тип книги varchar(10)	
--	--	--	--

**Таблица 3 «Структура сущностей создаваемых в процессе эксплуатации программного продукта»**

<b>№</b>	<b>Объект</b>	<b>Атрибуты</b>	<b>Взаимосвязь</b>
1	Reader	1. ИД в БД ВУЗа integer	Выданная книга

		2. номер читательского билета integer 3. номер телефона varchar (15)	
2	Lended_book	1. номер читательского билета integer 2. выдана в зал или на руки? boolean 3. что за книга штрих код integer 4. когда выдана Timestamp without time zone 5. срок сдачи date	
3	debtor	1. номер читательского integer 2. сумма fзадолженности float 3. книги которые не сданы integer	Читатель Выданная книга
4	room	- номер зала integer	

		- описание varchar (50)	
5	order	1. Читатель – Reader (integer) 2. Книга – Book (Integer) 3. Статус – Status(integer)	Читатель книга
6	barcode	code (integer) ISBN(varchar (50))	Книга