МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

«Тольяттинский государственный университет»

Кафедра Прикладная математика и информатика
(наименование)

09.03.03 Прикладная информатика
(код и наименование направления подготовки, специальности)

Разработка программного обеспечения
(направленность (профиль) / специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему «Разработка программного обеспечения цифровизации муниципальных услуг».

Обучающийся Н.С. Шумкова
(Инициалы Фамилия) (личная подпись)

Руководитель к.т.н., доцент, Н.В. Хрипунов
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2024

Аннотация

Тема выпускной квалификационной работы «Разработка программного обеспечения цифровизации муниципальных услуг».

Ключевые слова: библиотека, библиотечная деятельность, IDEF0, UML, MS SQL Server, Visual C#.

Объектом работы является процесс оказания библиотечного обслуживания пользователей в общественных городских библиотеках, обследованный на базе ООО «Каскад-ПРО» в рамках оказания услуг по описанию и оптимизации процессов общественных библиотек.

Предметом работы является процесс разработки программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек города Тюмени на базе ООО «Каскад-ПРО».

Целью работы является разработка программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек города Тюмени на базе ООО «Каскад-ПРО» и по заказу ООО «Каскад-ПРО».

Выпускная квалификационная работа состоит из введения, трех глав, заключения, списка используемой литературы и приложения.

В первой главе данной работы приводятся теоретическое обоснование разработки программного обеспечения.

Во второй главе работы приведены функциональные требования к программному обеспечению, выбраны инструменты проектирования и разработки, проводятся концептуальное, логическое и физическое проектирование базы данных программного обеспечения.

В третьей главе описывается процесс реализации программного обеспечения цифровизации муниципальных услуг.

Разработанное программное обеспечение позволит существенно ускорить обработку информации в библиотечных учреждениях.

Оглавление

Введение 5
Глава 1 Анализ предметной области и постановка задачи на разработку программного обеспечения цифровизации муниципальных услуг 8
1.1 Краткая характеристика библиотечного учреждения 8
1.2 Краткий обзор автоматизированных библиотечных информационных систем (АБИС) 15
1.3 Моделирование деятельности библиотечного учреждения 21
Глава 2 Проектирование программного обеспечения цифровизации муниципальных услуг 32
2.1 Функциональные требования программного обеспечения 32
2.2 Выбор инструментов проектирования и разработки 33
2.2.1 Операционная система 33
2.2.2 Среда программирования 34
2.2.3 Система управления базами данных (СУБД) 37
2.3 Концептуальное проектирование базы данных 38
2.4 Логическое моделирование базы данных 41
2.5 Физическое моделирование базы данных 42
2.6 Сценарий создания базы данных 46
Глава 3 Реализация и тестирование программного обеспечения цифровизации муниципальных услуг 48
3.1 Разработка программного обеспечения цифровизации муниципальных услуг 48
3.2 Схема взаимосвязи программных модулей 52
3.3 Реализованный сценарий базы данных 53
3.4 Проектирование и разработка интерфейса 62
3.5 Тестирование программного обеспечения цифровизации муниципальных услуг 63

Заключение 68
Список используемой литературы и используемых источников 70
Приложение А Формы интерфейса программного обеспечения цифровизации муниципальных услуг 73

Введение

В современное время автоматизация рабочих процессов оказания услуг и их перевод в цифровой вид становится неотъемлемой частью эффективного функционирования не только в органах государственной власти, но и в органах местного самоуправления (далее – ОМСУ) и подведомственных им учреждениях. Общественные городские библиотеки, как учреждения социально-культурного и образовательного досуга, также нуждаются в автоматизации процессов библиотечного обслуживания и цифровизации библиотечных услуг в целях повышения качества обслуживания читателей и оптимизации рабочих процессов. В связи с этим разработка программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек является актуальной.

Библиотечное обслуживание населения относится к государственным и муниципальным услугам с кодом «07 00 000» и наименованием услуги «Библиотечные услуги/работы», согласно Номенклатуре государственных и муниципальных услуг/работ, выполняемых организациями культурно-досугового типа Российской Федерации (утв. Распоряжением Минкультуры России от 18.09.2009 № Р-6 «Об утверждении номенклатуры государственных и муниципальных услуг/работ, выполняемых организациями культурно-досугового типа Российской Федерации»).

Согласно пп. 16 п. 1 ст.16 Федерального закона от 6 октября 2003 года № 131-ФЗ «Об общих принципах организации местного самоуправления в Российской Федерации», библиотечное обслуживание пользователей в муниципальных библиотеках находится в зоне ответственности ОМСУ.

В соответствии с пп.2.3.7. Положения об организации библиотечного обслуживания населения, комплектовании и обеспечении сохранности библиотечных фондов библиотек города Тюмени, утвержденного Постановлением Администрации г. Тюмени от 13.12.2007 № 42-пк

«Об утверждении Положения об организации библиотечного обслуживания населения, комплектовании и обеспечении сохранности библиотечных фондов библиотек города Тюмени», в рамках библиотечного обслуживания пользователей должен быть обеспечен доступ к информационным ресурсам библиотек в различных форматах (книги, периодические издания, аудио-, видео-, электронные документы, базы данных сети Интернет и прочее).

Таким образом, в рамках процесса библиотечного обслуживания пользователей осуществляется предоставление инфоресурсов из библиотечных фондов, в том числе на руки пользователю.

При этом, как учет выдачи информационных ресурсов из библиотечных фондов, так и учет пользователей/читателей, зачастую ведется на бумажных носителях, что существенно снижает качество и оперативность оказания библиотечных услуг за счет увеличения времени обслуживания и за счет влияния человеческого фактора.

Объект работы – процесс оказания библиотечного обслуживания пользователей в общественных городских библиотеках, обследованный на базе ООО «Каскад-ПРО» в рамках оказания услуг по описанию и оптимизации процессов общественных библиотек на примере города Тюмени.

Предмет работы – процесс разработки программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек города Тюмени на базе ООО «Каскад-ПРО», включая разработку функциональных требований к программному решению, выбор инструментов проектирования и разработки, проектирование базы данных и разработку самого функционала.

Цель работы – разработка программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек города Тюмени на базе ООО «Каскад-ПРО» и по заказу ООО «Каскад-ПРО».

Для достижения поставленной цели необходимо решить следующие задачи:

– представить краткую характеристику библиотечного учреждения города Тюмени;
– провести обзор автоматизированных библиотечных информационных систем (АБИС);
– провести моделирование деятельности библиотечного учреждения города Тюмени «Как есть»;
– провести реинжиниринг внутренних процессов в библиотечном учреждении города Тюмени в целях определения потребности в разработке программного обеспечения, предназначенного для автоматизации процессов библиотечного обслуживания и цифровизации оказываемых муниципальных услуг;
– реализовать на базе ООО «Каскад-ПРО» программное обеспечение цифровизации муниципальных услуг для общественных городских библиотек города Тюмени;
– провести функциональное тестирование программного обеспечения на предмет соответствия функциональным требованиям и тестирование SQL-скрипта (сценария) базы данных на предмет отсутствия ошибок в синтаксисе.

В настоящее время существующие в библиотечных учреждениях программные решения возможно усовершенствовать, разработав программное обеспечение, позволяющую улучшить эффективность выполнения процессов библиотечного обслуживания пользователей и позволяющее осуществить перевод библиотечных услуг в цифровой вид, что делает работу актуальной.

Глава 1 Анализ предметной области и постановка задачи на разработку программного обеспечения цифровизации муниципальных услуг

1.1 Краткая характеристика библиотечного учреждения

Муниципальное автономное учреждение культуры города Тюмени «Централизованная городская библиотечная система» (МАУК «ЦГБС») включает в себя центральную городскую библиотеку и 27 филиалов МАУК «ЦГБС» (из них: 25 филиалов – действующие, 2 филиала – временно не функционируют, 1 филиал – для внестационарных форм обслуживания):

– центральная городская библиотека (г. Тюмень, ул. Луначарского, д. 51/3);
– центральная детская библиотека имени В.П. Крапивина (г. Тюмень, ул. Харьковская, д. 48/2);
– литературно-краеведческий центр (г. Тюмень, ул. Первомайская, д. 14);
– центр семейного чтения имени А.С. Пушкина (г. Тюмень, ул. Газовиков, д. 30/1);
– информационно-библиотечный центр для молодёжи (г. Тюмень, ул. Червишевский тракт, д. 15/1);
– библиотека истории города имени А.И. Текутьева (г. Тюмень, ул. Щербакова, д. 11);
– индустриальный парк «Литера» (г. Тюмень, ул. Тальниковая, д. 5);
– библиотека № 1 (г. Тюмень, ул. Матросова, д. 27);
– библиотека № 2 имени А.А. Гришина (г. Тюмень, ул. Мира, д. 31/2);
– детская библиотека № 3 имени С.В. Мальцева (г. Тюмень, ул. Ямская, д. 110/4);

– библиотека № 4 имени М.М. Пришвина (модельная) (г. Тюмень, ул. Ватутина, д. 14/1);
– детская библиотека № 5 (г. Тюмень, ул. Карла Маркса, д. 108);
– библиотека № 6 имени Зота Корниловича Тоболкина (г. Тюмень, ул. Одесская, д. 39/1);
– библиотека № 7 имени И.М. Ермакова (г. Тюмень, ул. Камчатская, д. 1а);
– библиотека № 8 имени А.П. Чехова (г. Тюмень, ул. Северная, д. 2/1);
– библиотека № 9 (г. Тюмень, ул. Волгоградская, д. 119/2);
– библиотека № 10 (г. Тюмень, ул. Александра Пушкина, д. 10);
– детская библиотека № 11 (г. Тюмень, ул. Жуковского, д 28) – временно не функционирует;
– библиотека № 12 (г. Тюмень, ул. Маршала Захарова, д. 5/3);
– библиотека № 13 (г. Тюмень, ул. Мельничная, д. 10/1) – для внестационарных форм обслуживания;
– библиотека журналистики имени Р.С. Гольдберга (г. Тюмень, проезд Геологоразведчиков, д. 38/4);
– библиотека № 15 имени П.П. Ершова (г. Тюмень, ул. Минская, д. 98/3);
– библиотека «Олимп» (г. Тюмень, ул. Олимпийская, д. 20а);
– библиотека № 17 (г. Тюмень, ул. Игримская, д. 33).
– библиотека № 18 (г. Тюмень, ул. Пражская, 41) – временно не функционирует;
– библиотека № 19 (г. Тюмень, ул. Станционная, д. 18а/1).;
– библиотека № 20 (г. Тюмень, ул. Пермякова, д. 6а/1);
– библиотека № 21 (г. Тюмень, ул. 70 лет Октября, д. 5-в).

В центральной городской библиотеке МАУК «ЦГБС» (ЦГБ МАУК «ЦГБС») и в филиалах МАУК «ЦГБС» применяются одни и те же процессы организации библиотечного обслуживания пользователей. В связи

с этим в рамках услуг по описанию и оптимизации процессов общественных библиотек, оказанных на базе ООО «Каскад-ПРО», обследован процесс организации библиотечного обслуживания пользователей на примере ЦГБ МАУК «ЦГБС».

Организационная структура и штатная численность подразделения ЦГБ МАУК «ЦГБС» включает в себя:

а) дирекция:

1) директор (1 шт.ед.);
2) заместитель директора по организации библиотечного обслуживания (1 шт.ед.);
3) заместитель директора по информационным технологиям (1 шт.ед.);
4) заместитель директора по административно-хозяйственной деятельности (1 шт.ед.);
5) главный бухгалтер (1 шт.ед.);
6) помощник директора (1 шт.ед.);
7) ведущий юрисконсульт (1 шт.ед.);
8) специалист по закупкам (1 шт.ед.).

б) отдел кадров:

1) начальник отдела (1 шт.ед.);
2) документовед 1 категории (1 шт.ед.).

в) хозяйственный отдел:

1) начальник отдела (1 шт.ед.);
2) водитель (1 шт.ед.);
3) уборщик территорий (1 шт.ед.);
4) электромонтер по ремонту и обслуживанию электрооборудования (1 шт.ед.).

г) отдел развития:

1) заведующий отделом (1 шт.ед.);

2) главный библиотекарь (1 шт.ед.).

д) отдел автоматизации:

– заведующий отделом (1 шт.ед.);

– инженер-электроник 1 категории (1 шт.ед.).

е) редакционно-издательский отдел:

1) заведующий отделом (1 шт.ед.);

2) художник-дизайнер (1 шт.ед.);

3) специалист по печати и постпечатной обработке документа (1 шт.ед.).

ж) отдел комплектования и технической обработки документов:

1) заведующий отделом (1 шт.ед.);

2) главный библиотекарь (1 шт.ед.);

3) библиотекарь 1 категории (1 шт.ед.);

4) помощник библиотекаря (1 шт.ед.).

и) сектор обменно-резервного фонда:

1) заведующий отделом (1 шт.ед.);

2) главный библиотекарь (1 шт.ед.).

к) отдел библиографической деятельности:

1) заведующий отделом (1 шт.ед.);

2) главный библиограф (1 шт.ед.);

3) ведущий библиограф (1 шт.ед.).

л) отдел социокультурной и досуговой деятельности:

1) заведующий отделом (1 шт.ед.);

2) фотограф (1 шт.ед.);

3) организатор социокультурной и досуговой деятельности (1 шт.ед.).

м) отдел обслуживания:

1) заведующий сектором (1 шт.ед.);

2) главный библиотекарь (1 шт.ед.).

н) сектор читальных залов:

1) заведующий сектором (1 шт.ед.);

2) ведущий библиотекарь (1 шт.ед.).

п) сектор обслуживания детей и подростков:

1) заведующий сектором (1 шт.ед.);

2) ведущий библиотекарь (1 шт.ед.).

р) сектор обслуживания взрослых пользователей.

1) заведующий сектором (1 шт.ед.);

2) ведущий библиотекарь (1 шт.ед.).

с) отдел электронных ресурсов:

1) заведующий сектором (1 шт.ед.);

2) главный библиотекарь (1 шт.ед.);

3) ведущий библиотекарь (1 шт.ед.);

4) библиотекарь 1 категории (1 шт.ед.).

т) сектор по формированию корпоративного сайта.

1) заведующий сектором (1 шт.ед.);

2) специалист по технической поддержке сайтов (1 шт.ед.);

3) менеджер по информационным ресурсам (1 шт.ед.);

4) контент-редактор (1 шт.ед.).

Организационная структура ЦГБ МАУК «ЦГБС» представлена на рисунке 1.

Функциональная структура ЦГБ МАУК «ЦГБС» представлена на рисунке 2.

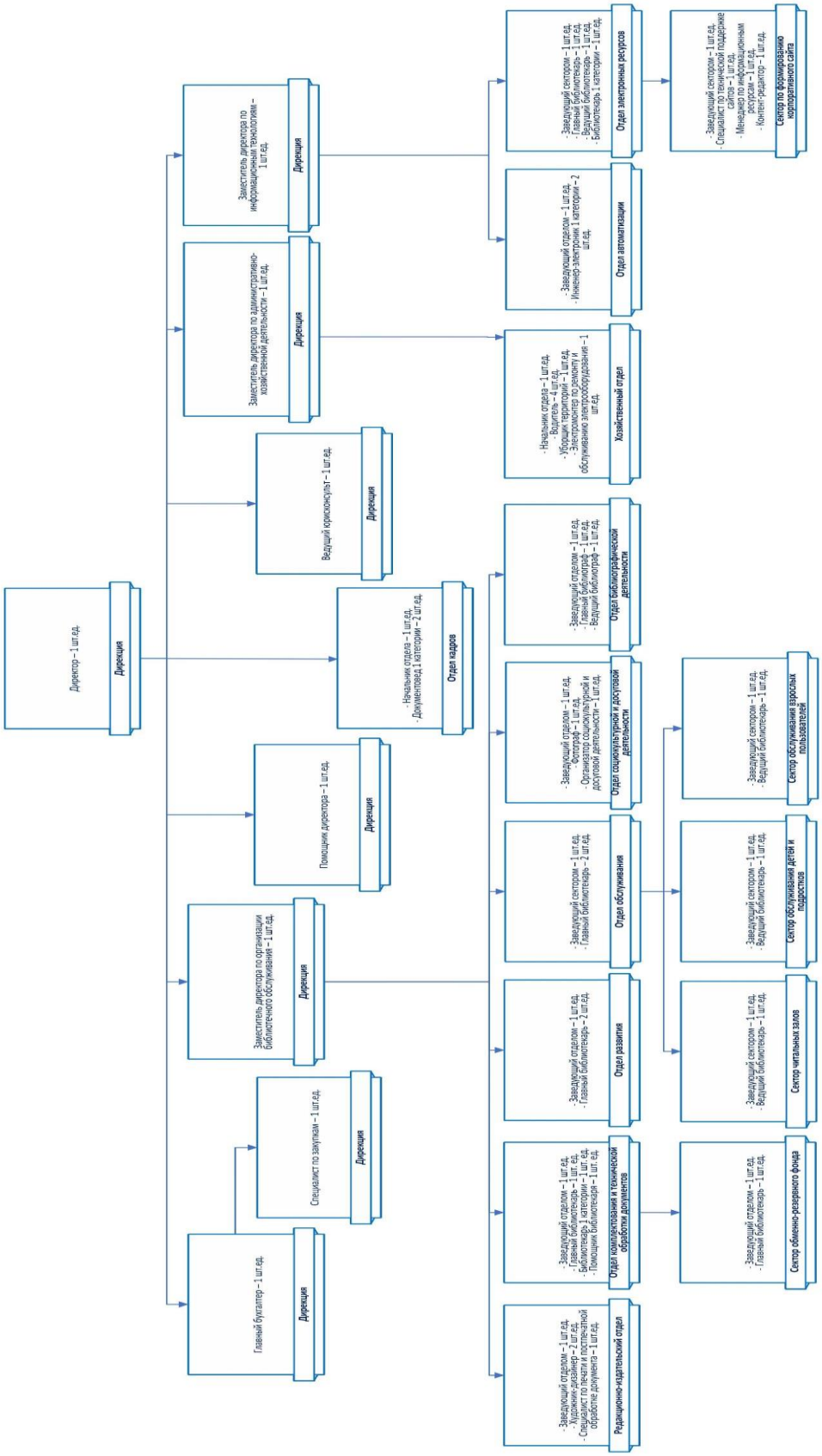


Рисунок 1 – Организационная структура ЦГБ МАУК «ЦГБС»

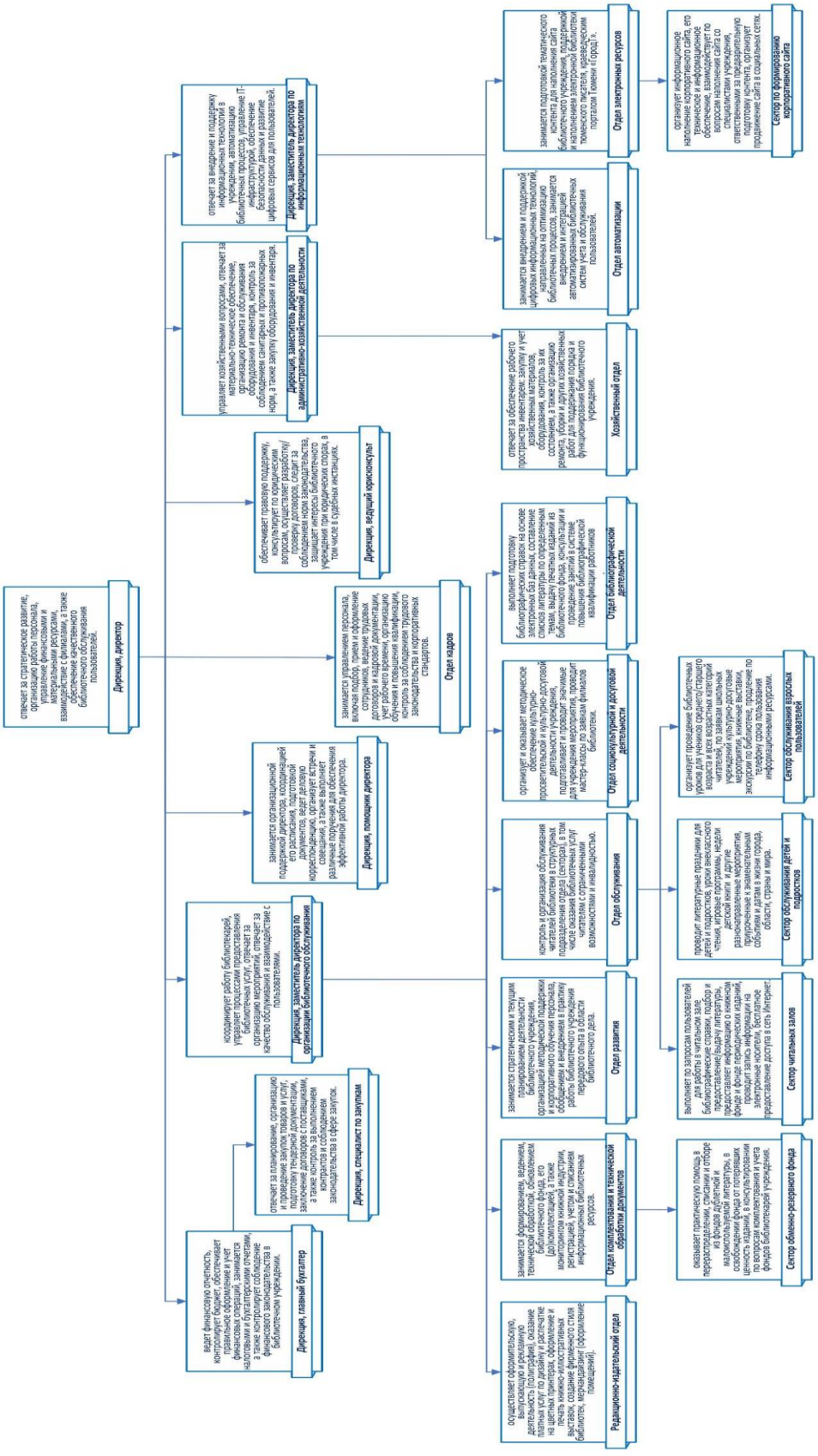


Рисунок 2 – Функциональная схема ЦГБ МАУК «ЦГБС»

В МАУК «ЦГБС» активно используются цифровые информационные технологии в процессе библиотечного обслуживания. В МАУК «ЦГБС» пользователи имеют возможность получения бесплатного удаленного доступа к лицензированным материалам таких электронно-библиотечных систем, как IPRbooks, НЭБ и Литрес. Кроме того, во в ЦГБ МАУК «ЦГБС» и во всех остальных филиалах МАУК «ЦГБС» пользователи могут получить доступ к актуальной правовой информации, содержащейся в справочной правовой системе «КонсультантПлюс».

1.2 Краткий обзор автоматизированных библиотечных информационных систем (АБИС)

В целях автоматизации процесса библиотечного обслуживания в настоящее время существует различные программные решения – автоматизированные библиотечно-информационные системы (АБИС), сравнительный анализ которых, в том числе АБИС ИРБИС64, используемой в МАУК «ЦГБС», представлен ниже.

АБИС OPAC-Global – разработка компании ООО «ДИТ-М».

АБИС OPAC-Global имеет 3 версии (OPAC-Global, OPAC-midi и OPAC-mini), построенные на основе стандартной архитектуры web-серверов и клиентов и используемая в целях автоматизации работы сети/филиалов библиотечных учреждений с корпоративной системой с организацией централизованного сервера (OPAC-Global), распределенного(ых) сервера(ов) (OPAC-midi) и с единым протоколом обмена данными HTTP.

В версиях OPAC-Global и OPAC-midi пользователи получают доступ к библиотечной системе и выполняют функции в web-браузере [9].

В упрощенной версии OPAC-mini, на базе формата RUSMARC-Holding, реализован модуль для учета/регистрации посещаемости. При регистрации читателей используются современные

технологии штрих-кодирования и радиочастотной идентификации (RFID). Также в OPAC-midi реализован модуль создания отчетов и статистических данных о выданных книгах, составе пользователей и посещаемости библиотеки.

Преимущество OPAC-mini заключается в автоматизации процесса генерации статистики и использование RFID-технологий. Однако данная система не учитывает все услуги, предоставляемые читателям, такие как доступ к электронным изданиям и предоставление справок, что является существенном недостатком OPAC-mini, поскольку не позволяет удовлетворять все потребности в рамках процесса библиотечного обслуживания.

АБИС НЕВА – разработка компании ООО «Балтик софт».

В АБИС НЕВА реализована автоматизация основных рабочих процессов библиотечных учреждениях с гибкими настройками, позволяющими адаптировать систему под индивидуальные особенности библиотек различного профиля. Для автоматизации отдельного библиотечного процесса в структуре АБИС НЕВА реализованы конкретные АРМ (автоматизированные рабочие места). Возможно внедрение АБИС НЕВА как в библиотечные учреждения с минимальным уровнем автоматизации, так и с возможностью перехода на нее с устаревших библиотечных-информационных систем.

В АБИС НЕВА предусмотрена автоматизация процессов отчетности, включая как генерацию статистических, так и формирование отчетных документов в разрезе:

– информации о деятельности работников библиотеки;
– информации о составе библиотечного фонда с детализированными данными;
– информации о категориях читателей/пользователей;
– информацией о выдаче книг.

Преимущество АБИС НЕВА заключается в возможности контролировать выдачу книг и состав библиотечных фондов с помощью генерации отчетной документации. Дополнительно к этому, с помощью генерации отчетов по категориям читателей имеется возможность планировать и организовывать различные мероприятия с учетом состава читателей. Однако в АБИС НЕВА не учитываются все услуги, предоставляемые читателям, что является недостатком системы. Кроме того, к недостаткам системы относится отсутствие возможности задавать в АБИС НЕВА отчетный период [10].

АБИС ИРБИС64 – разработка специалистов Государственной публичной научно-технической библиотеки (ГПНТБ России) и Международной ассоциации пользователей и разработчиков электронных библиотек и новых информационных технологий (ЭБНИТ).

АБИС ИРБИС64 является модульной системой в составе TCP/IP сервера баз данных с 7 базовыми АРМ(ами), предназначенными для работы в Windows 2000/XP (и выше) в архитектуре клиент-сервера, как для средних, так и для крупных иблиотек и филиалов.

ИРБИС64+ предназначена для формирования/ведения/каталогирования электронной библиотеки.

ИРБИС128 предназначена для работы в Windows, *nix, MacOS и других ОС, основывается на Web-технологиях и распределенных БД и имеет в своей структуре три звена – СУБД, сервер приложений и клиент.

Web-ИРБИС64 предназначена для работы в Windows 2000/XP (и выше) и HTTP-серверов IIS, Apache, WebSite – предоставляет доступ к БД ИРБИС посредством Web-технологий.

J-ИРБИС, как дополнительный модуль, предназначен для создания/ведения контента библиотечных порталов и электронных библиотек на основе CMS Joomla.

Преимуществом линейки «Ирбис» является возможность интеграции с другими элементами ЭИОС образовательных учреждений и бесшовная авторизация в электронных библиотечных системах. Как к преимуществам, так и к недостаткам относятся низкие системные требования к библиотечной технике.

АИБС МегаПро – разработка компании ООО «Дата Экспресс».

АИБС МегаПро основана на современных веб-технологиях и реляционных базах данных, также доступна через веб-интерфейс. АИБС МегаПро поддерживает интеграцию с другими библиотечными и информационными системами, а также позволяет подключать электронные ресурсы.

К недостаткам АИБС МегаПро относятся первоначальные затраты на лицензирование, установку системы и совеременное оборудование для ее поддержания, которые могут быть неприемлемыми для библиотек с минимальным бюджетом.

АБИС «Руслан» – разработка компании ООО «Открытые библиотечные системы» (ранее – центр «Открытые библиотечные системы» Санкт-Петербургского государственного технического университета).

АБИС «Руслан» – модульная система, предназначенная для автоматизации в библиотеках различного профиля всех основных этапов обработки единиц хранения библиотечных фондов и этапов библиотечного обслуживания читателей.

В составе АБИС «Руслан» реализованы взаимосовместимые АРМ(ы), вследствие добавления которых становится доступным для работы функционал, необходимый для организации библиотечного обслуживания.

К преимуществам системы относится возможность миграции всей библиографической информации с других АБИС в АБИС «Руслан», возможность создания/ведения электронных библиотек и полнотекстовых

ресурсов как элемента библиотечного научно-образовательного пространства образовательного учреждения.

Все группы пользователей (с ролями читатели, библиотекари, администраторы) в АБИС «Руслан» имеют возможность удаленной работы с помощью web-браузеров через web-интерфейс. В целях сохранения данных используется база компании Oracle, что, в свою очередь, отменяет ограничения на количество одновременно работающих в АБИС «Руслан» пользователей, на количество баз и на количество записей в них.

К недостаткам АБИС «Руслан» относятся сложности в ее настройке, связанные с архитектурой построения компьютерных сетей, наличием знаний/опыта работы в среде Linux/UNIX, в связи требуется либо высококвалифицированный специалист в структуре библиотеке для ее настройки, либо использование сервиса технической поддержки АБИС «Руслан», что является финансовозатратным.

Таким образом, к недостаткам АБИС «Руслан» относится ценовая политика – затраты на её внедрение и техническую поддержку, а также необходимость наличия современного компьютерного оборудования для стабильной работы системы.

АБИС «МАРК-SQL» – разработка компании НПО «Информ-система».

АБИС «МАРК-SQL» основана на реляционной базе данных SQL, предназначена для комплексной автоматизации библиотечных процессов через продвинутый веб-интерфейс с учетом современных информационных технологий, включающих в себя многофункциональный спектр инструментов для библиотеки (создание/ведение/учет и работа с электронными ресурсами, мониторинг библиотечного фонда, печать выходных форм и др.).

По аналогии с другими АБИС возможно расширение функционала/модификация АБИС «МАРК-SQL» посредством существующих АРМ (модулей).

К преимуществам АБИС «МАРК-SQL» относится наличие возможности подключения клиента для хранения записей к любым свободным и проприетарным решениям, что демонстрирует независимость от существующей в библиотечном учреждении базы данных и, тем самым, отсутствие привязки библиотечных процессов к какому-либо одному программному решению. АБИС «МАРК-SQL» использует технологию MARC (Machine-Readable Cataloging) для хранения и обмена метаданными, которая позволяет проводить интеграцию с другими библиотечными системами и базами данных, а также производить упрощенную миграцию на АБИС «МАРК-SQL» с других библиотечных систем.

К недостаткам АБИС «МАРК-SQL» относится значительные затраты на ее внедрение, поддержку и резервное копирование, для стабильной работы предъявляются достаточно высокие системные требования к оборудованию: мощный сервер для управления ОС Windows Server 2008, большой объем оперативной памяти и большой объем дискового пространства.

Имеющиеся достоинства и недостатки перечисленных АБИС представлены в таблице 1.

Таблица 1 – Сравнительная характеристика анализируемых АБИС

АБИС	Мультиплатформенность	Сетевой режим работы	Модульная архитектура	Возможность доработки
АБИС Орас-Global	-	+	-	-
АБИС Нева	-	+	-	-
АБИС Ирис 64	+	+	-	-
АБИС МегаПро	+	-	+	-
АБИС Руслан	+	+	+	-
Mark-SQL	-	+	+	-
Разрабатываемое программное обеспечение	-	+	+	+

Каждая из указанных АБИС имеет свои особенности, достоинства и недостатки. У одних АБИС есть привязка к используемым операционным системам и соблюдение требований к дорогостоящим аппаратным средствам для стабильной работы АБИС, для других не требуется значительных затрат на техническое оснащение [24], но требуется техническое обслуживание высококвалифицированными специалистами для устойчивости работы баз данных АБИС. Так или иначе, в любом случае выбор программного решения осуществляется с учетом бюджетного лимита библиотченого учреждения и с учетом характеристик самих АБИС, которые далеко не всегда в полном объеме адаптированы под нужды конткретного учреждения. Зачастую плюсы и минусы подобных выбранных решений обнаруживаются только после закупки в процессе эксплуатации. Дороботки готовых пакетных решений под текущие нужды либо не возможны, либо являются дорогостоящими и трудозатратными. В связи с этим в качестве альтернативы подходит вариант разработки собственного программного обеспечения.

1.3 Моделирование деятельности библиотечного учреждения

Контекстная диаграмма деятельности ЦГБ МАУК «ЦГБС» включает в себя следующие элементы [25]:

– входные данные – данные пользователей, данные об информационных ресурсах и данные о брони;
– выходные данные – забронированные читателями информационные ресурсы;
– механизмы – пользователи (читатель и библиотекарь);
– управляющие воздействия – ЦГБ МАУК «ЦГБС».

Контекстная диаграмма деятельности ЦГБ МАУК «ЦГБС» приведена ниже на рисунке 3.

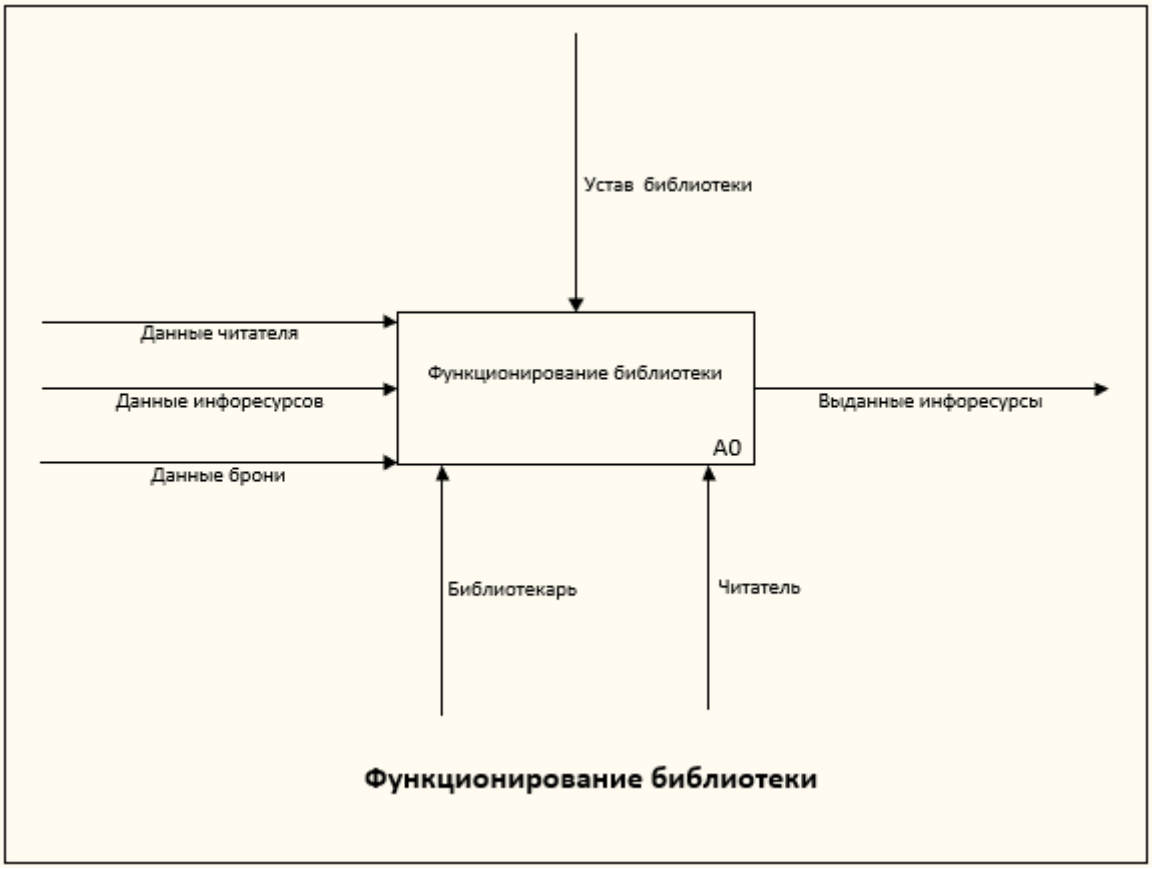


Рисунок 3 – Контекстная диаграмма функционирования ЦГБ МАУК «ЦГБС»

Декомпозиция контекстной диаграммы функционирования ЦГБ МАУК «ЦГБС» приведена на рисунке 4.

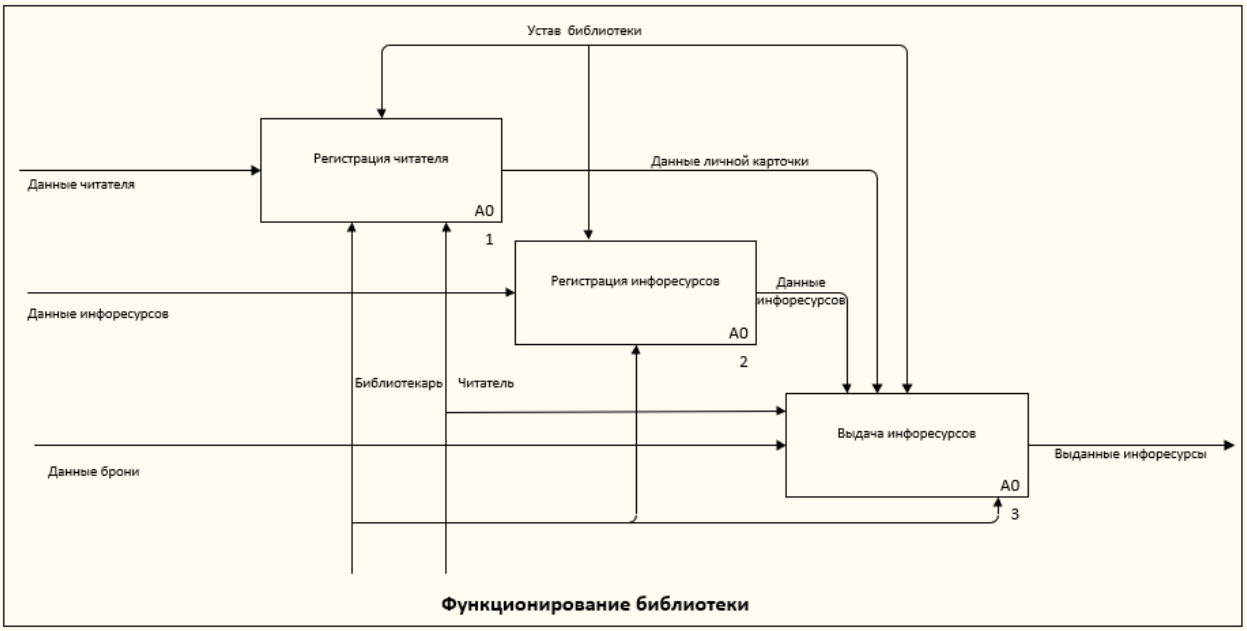


Рисунок 4 – Декомпозиция контекстной диаграммы функционирования ЦГБ МАУК «ЦГБС»

В декомпозиции контекстной диаграммы функционирования ЦГБ МАУК «ЦГБС» отображено, что в ЦГБ МАУК «ЦГБС» проводится регистрация читателей, регистрация информационных ресурсов и выдача информационных ресурсов.

Декомпозиция процесса «Регистрация читателя» в ЦГБ МАУК «ЦГБС приведена на рисунке 5.

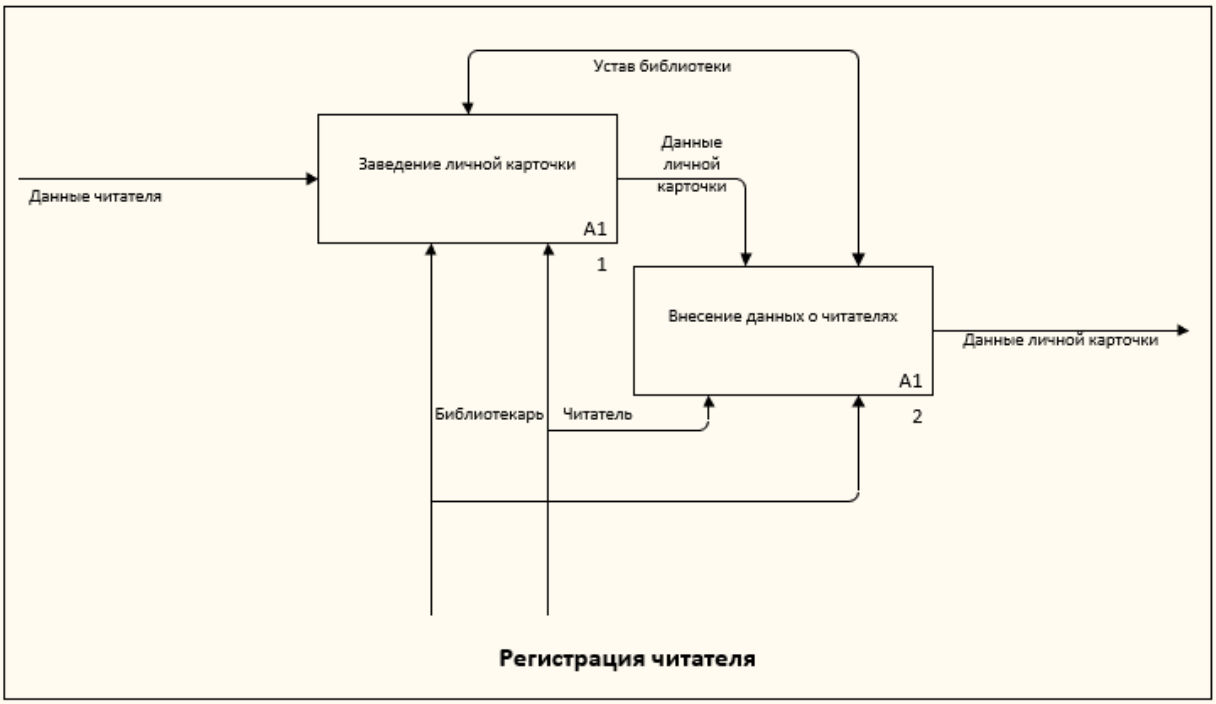


Рисунок 5 – Декомпозиция процесса «Регистрация читателя»

Декомпозиция процесса «Регистрация информационных ресурсов» в ЦГБ МАУК «ЦГБС» приведена на рисунке 6.

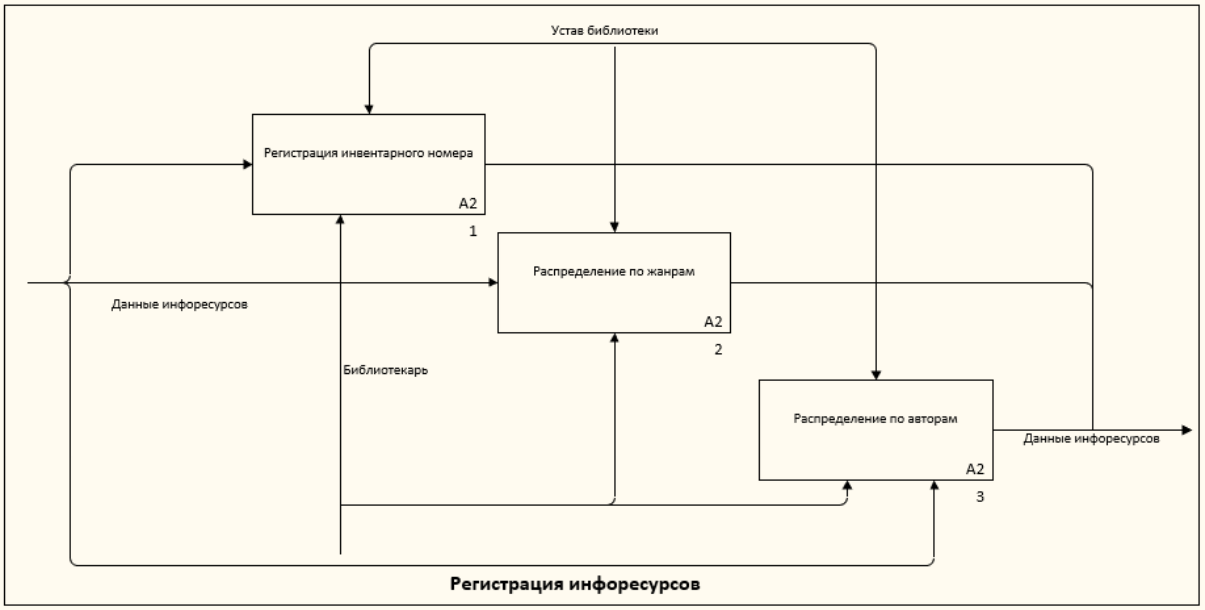


Рисунок 6 – Декомпозиция процесса «Регистрация информационных ресурсов»

Проведенное моделирование процессов позволяет наглядно продемонстрировать недостатки [21], выявленные в процессе библиотечного обслуживания в ЦГБ МАУК «ЦГБС», связанные с отсутствием автоматизации деятельности библиотекарей:

– наличие со стороны библиотекарей технических ошибок/опечаток при заполнении документов;
– наличие со стороны библиотекарей периодически неразборчивого почерка;
– наличие со стороны библиотекарей влияния человеческого фактора.

В целях устранения/исключения перечисленных факторов/недостатков требуется модернизация библиотечных процессов в ЦГБ МАУК «ЦГБС», в связи с чем в ЦГБ МАУК «ЦГБС» проведено реинжинирование библиотечных процессов и IDEF0-моделирование системы «Как должно быть» [26].

Вследствие чего входные данные в контекстной диаграмме функционирования ЦГБ МАУК «ЦГБС» остаются без каких-либо изменений, но в текущие механизмы включается программное обеспечение, по результатам работы которого читателю предоставляется/выдается информационный ресурс.

Процесс функционирования ПО ЦГБ МАУК «ЦГБС» с учетом наличия правил и регламентов выглядит следующим образом на рисунке 7.

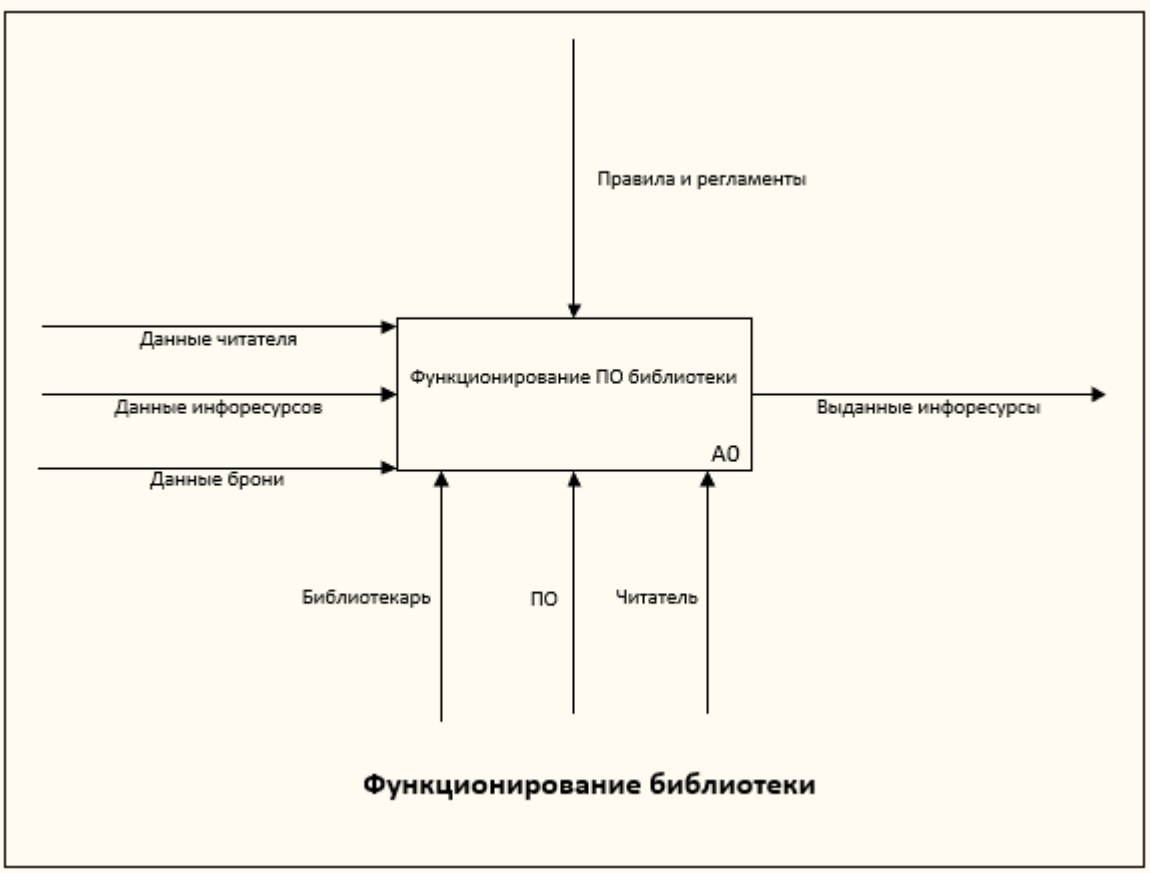


Рисунок 7 – Контекстная диаграмма функционирования ПО в ЦГБ МАУК «ЦГБС»

В декомпозиции контекстной диаграммы функционирования ПО в ЦГБ МАУК «ЦГБС» на рисунке 8 отображено, что процесс работы ПО в ЦГБ МАУК «ЦГБС» состоит из следующих подпроцессов:

– ввод первичных данных;

– бронирование информационных ресурсов;

– прием и выдача информационных ресурсов.

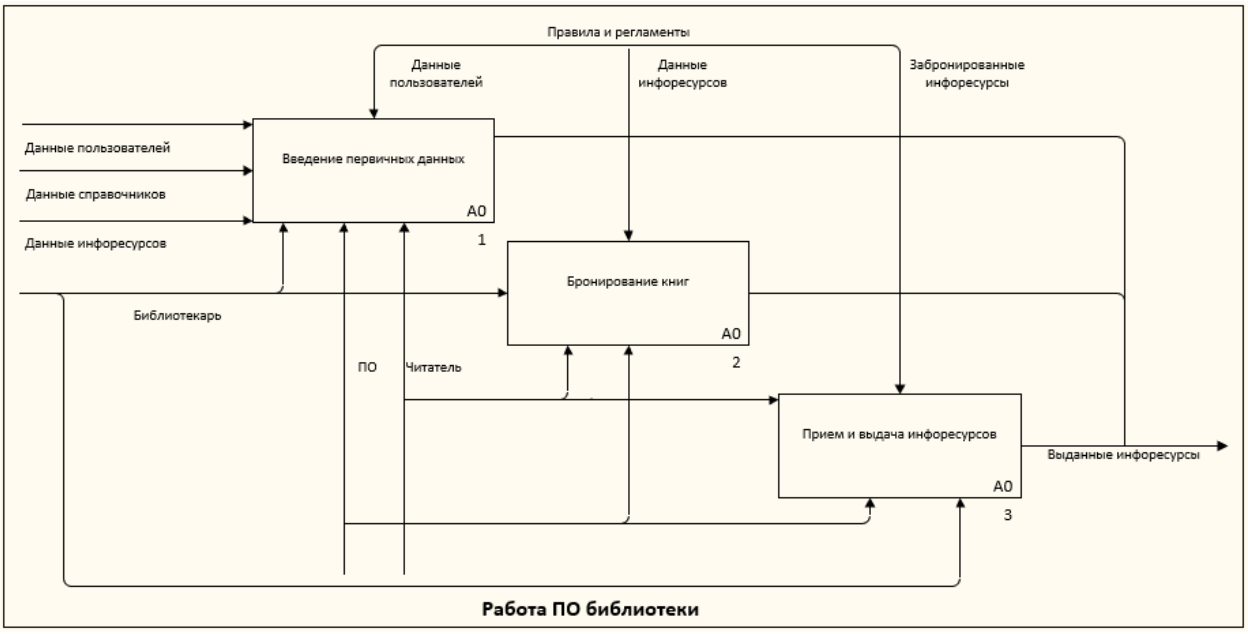


Рисунок 8 – Декомпозиция контекстной диаграммы работы ПО ЦГБ МАУК «ЦГБС»

На рисунке 9 приведена декомпозиция первого подпроцесса «Ввод первичных данных».

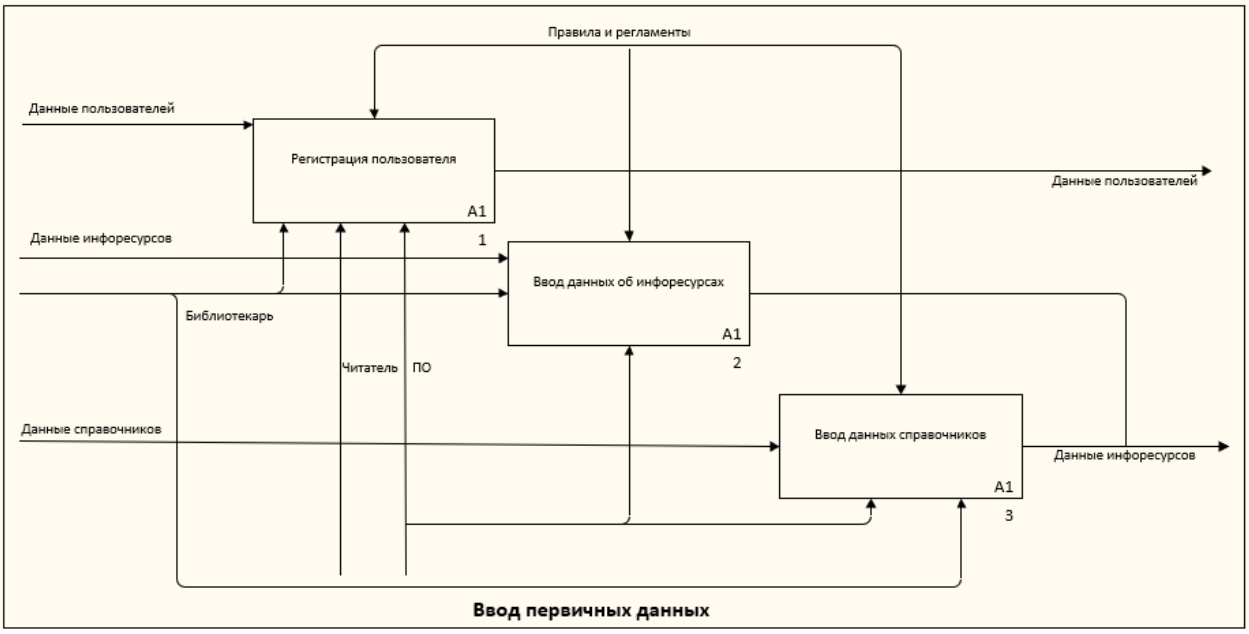


Рисунок 9 – Декомпозиция процесса «Ввод первичных данных»

Декомпозиция второго подпроцесса «Бронирование информационных ресурсов» приведена на рисунке 10.

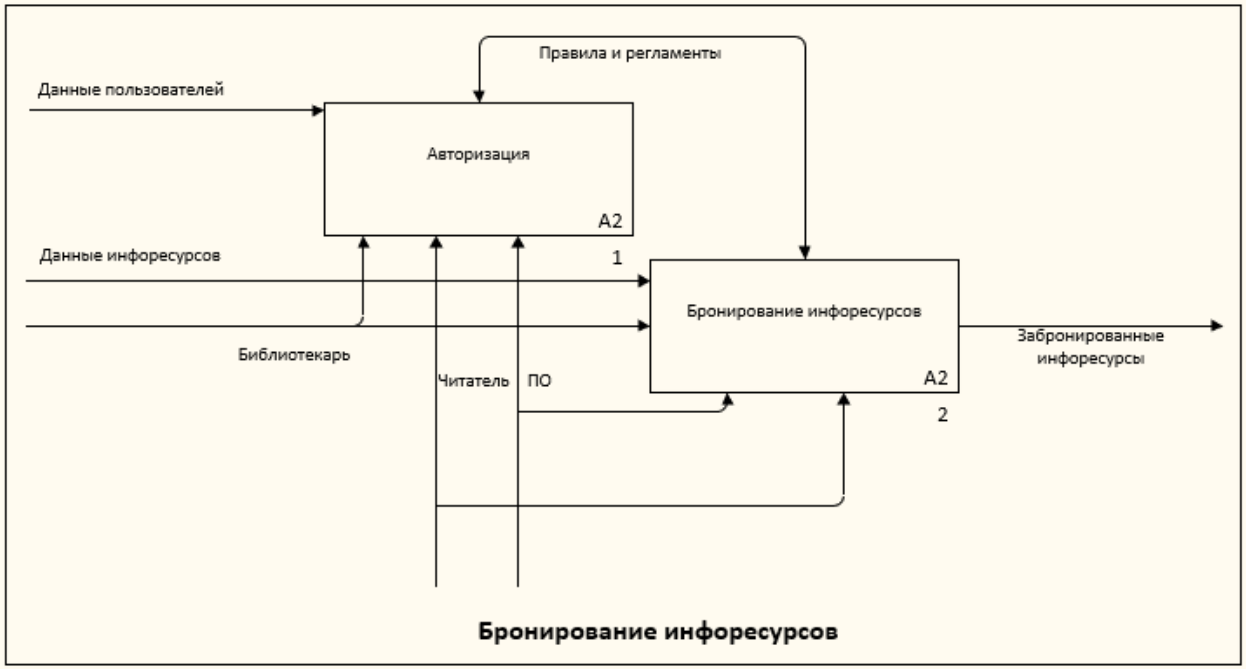


Рисунок 10 – Декомпозиция процесса «Бронирование информационных ресурсов»

Декомпозиция первого подпроцесса «Прием и выдача информационных ресурсов» приведена на рисунке 11.

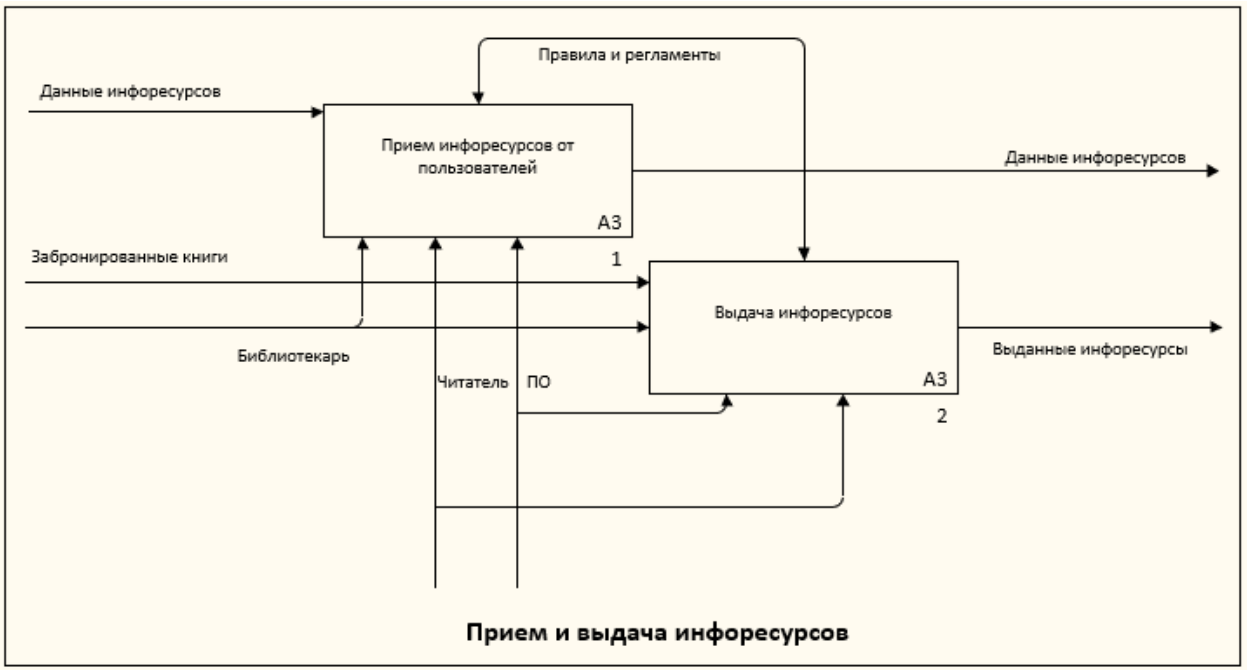


Рисунок 11 – Декомпозиция процесса «Прием и выдача информационных ресурсов»

В результате реинжиниринга внутренних процессов в ЦГБ МАУК «ЦГБС» в процесс библиотечного обслуживания пользователей добавлено онлайн-бронирование информационных ресурсов, исключено хранение персональных данных пользователей на бумажных носителях и организовано их хранение непосредственно в программном обеспечении. За счет подобного нововведения пользователи смогут от своего имени производить вход в библиотечную систему и выполнять необходимые им операции.

В соответствии с нотацией UML составлена Use Case диаграмма функционирования программного обеспечения на рисунке 12.

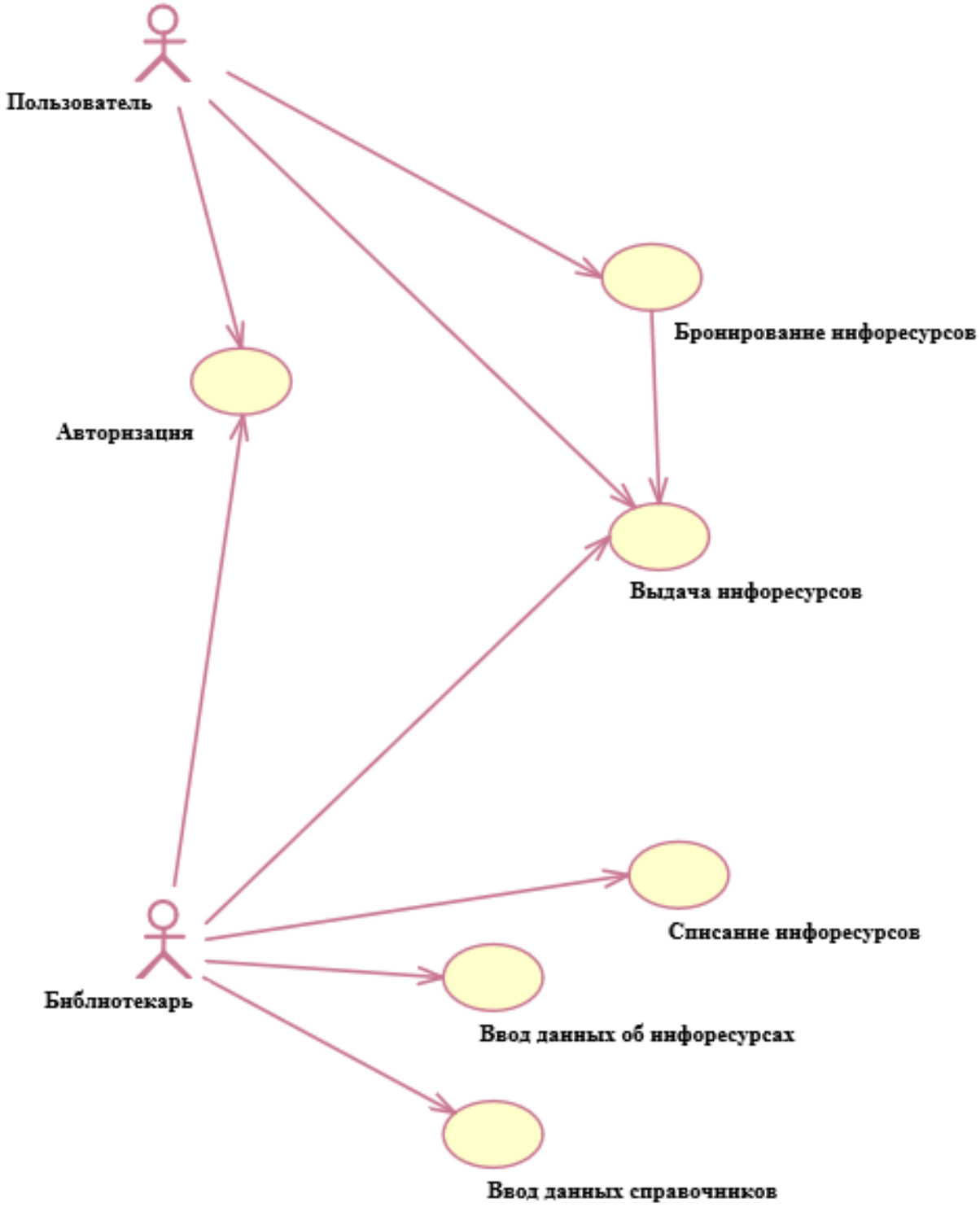


Рисунок 12 – Use Case диаграмма

На диаграмме Use case отображено, что в системе пользователь читатель и библиотекарь являются актерами.

UML (англ. Unified Modeling Language – унифицированный язык моделирования) – стандартизированный язык визуального моделирования [15], который используется в целях проектирования и в целях документирования программного обеспечения. UML предлагает целый набор различных графических диаграмм, с помощью которых разработчики могут визуализировать структуру и архитектуру систем [23].

В Use Case диаграмме функционирования программного обеспечения отражено, что читатель принимает непосредственное участие в следующих операциях:

– бронирование информационных ресурсов;

– выдача информационных ресурсов.

В Use Case диаграмме функционирования программного обеспечения отражено, что библиотекарь принимает непосредственное участие в следующих операциях:

– выдача информационных ресурсов;

– списание информационных ресурсов;

– ввод данных о пользователях;

– ввод данных об информационных ресурса;

– ввод данных справочников (виды инфоресурсов, авторы, жанры, издательства, причины списания).

Выводы по главе 1

В данной главе проведен анализ предметной области и постановка задачи на разработку программного обеспечения цифровизации муниципальных услуг, в том числе приведены кратная характеристика библиотечного учреждения, краткий обзор автоматизированных библиотечных информационных систем, осуществлено моделирование деятельности данного учреждения.

Глава 2 Проектирование программного обеспечения цифровизации муниципальных услуг

2.1 Функциональные требования программного обеспечения

К функционалу разрабатываемого программного обеспечения сформированы следующие требования:

– разграничение ролей пользователей с соответствующими правами доступа (на библиотекарей и на читателей);
– ввод данных об информационных ресурсах;
– ввод данных о пользователях;
– ввод данных в справочник «Виды инфоресурсов»;
– ввод данных в справочник «Издательства» (в случае наличия);
– ввод данных в справочник «Авторы» (в случае наличия);
– ввод данных в справочник «Жанры» (в случае наличия);
– ввод данных в справочник «Причины списания» (в случае наличия);
– выдача информационных ресурсов;
– списание информационных ресурсов.

Для читателей:

– наличие возможности оформления брони на информационный ресурс;
– наличие возможности просмотра справочника «Инфоресурсы»;
– наличие возможности просмотра справочника «Авторы»;
– наличие возможности просмотра справочника «Жанры»;
– наличие возможности просмотра справочника «Издательства»;
– наличие возможности просмотра справочника «Виды инфоресурсов».

Для библиотекарей:

– ввод и редактирование данных пользователей системы с указанием прав доступа;

– формирование справочника «Пользователи»;

– формирование справочника «Виды инфоресурсов»;

– формирование справочника «Авторы»;

– формирование справочника «Издательства»;

– формирование справочника «Жанры»;

– формирование справочника «Списание»;

– формирование справочника «Причины списания»;

– формирование справочника «Инфоресурсы»;

– закрытие брони инфоресурсов;

– выдача инфоресурсов;

– списание инфоресурсов.

2.2 Выбор инструментов проектирования и разработки

2.2.1 Операционная система

Реализуемое программное обеспечение ориентировано для использования в операционных системах Windows, что обусловлено большей распространенностью ОС Windows среди пользователей, нежели ОС Linux и другие ОС.

ОС Windows не является уникальной, но у нее имеется некоторые существенные преимущества перед другими операционными системами [1].

К преимуществам ОС Windows относятся:

– её аппаратная и программная совместимость;

– стандартный для пользователей графический интерфейс;

– возможности многозадачности;

– возможности интеграции и возможности взаимодействия с различными программными решениями;

– возможности обмена данными между приложениями;

– наличие современных сетевых технологий.

2.2.2 Среда программирования

Объектно-ориентированный язык программирования C# (произносится как «си-шарп») – современный, объектно-ориентированный язык программирования, разработанный в 2000 году компанией Microsoft как часть платформы .NET [8]. Простота синтаксиса языка C# и наличие мощных возможностей его использования в программировании подходят как для новичков, так и для опытных разработчиков [6].

К основным особенностям языка C# относятся [22]:

– наличие поддержки основных принципов объектно-ориентированного программирования (далее – ООП), благодаря чему имеется возможность создания программных решений, моделирующих реальные объекты с помощью классов, объектов, наследования, инкапсуляции и полиморфизма;
– наличие строгой типизации языка C#, что исключает в коде большое количество ошибок за счет проверки типов на этапе компиляции;
– наличие автоматического управление памятью, включающего сборку мусора (Garbage Collection) для выделения и освобождения памяти, занимаемую больше не использующимися объектами;
– наличие межъязыковой интеграции, позволяющей в рамках платформы .NET комбинировать код на C# с другими языками и использовать компоненты и библиотеки, написанные на других языках, таких как F#, VB.NET и C++/CLI, что в совокупности позволяет проводить интеграцию с существующими решениями;
– C# имеет такие встроенные механизмы для обеспечения безопасности, как проверка границ массива, проверка нулевых ссылок, исключение и защита от переполнения;
– наличие возможностей применения языка C# для разработки программного обеспечения, включая разработку web-приложений (с использованием ASP.NET Core), игр (интеграция

с движком Unity), мобильных приложений (с помощью Xamarin и MAUI), настольных приложений (с использованием Windows Forms, WPF), а также для разработки микросервисов и облачных приложений (Azure).

C# не накладывает ограничений на разработку различных типов и направленностей программного решений. C# может использоваться для таких разнообразных проектов, как создание текстовых процессоров, графики, электронных таблиц и компиляторов для других языков [3].

C# удобен для использования, сочетает в себе простоту, объектную ориентацию, гибкость, модульность, производительность и мощные возможности для создания современных программных решений, что делает его популярным среди разработчиков.

Объектно-ориентированный язык программирования Java – язык программирования, разработанный компанией Sun Microsystems (в настоящий момент – Oracle) в 1995 году [2].

К основным особенностям Java относятся:

– отсутствие таких сложных элементов, как указатели, множественное наследование, ручное управление памятью (Garbage Collection автоматически очищает неиспользуемую память);
– наличие большой экосистемы библиотек и фреймворков и инструментов, таких как Spring, Hibernate, Apache Maven, значительно упрощающих и ускоряющих процесс разработки программного решения;
– наличие автоматического управление памятью, включающего сборку мусора (Garbage Collection) для выделения и освобождения памяти, занимаемую больше не использующимися объектами;
– наличие широкого спектра применения Java для разработки программного обеспечения, включая разработку web-приложений (благодаря фреймворкам, таким как Spring и Hibernate),

масштабируемых серверных приложений, банковских систем, CRM-систем, ERP-систем, кроссплатформенных приложений (Java запускается на различных операционных системах без изменения исходного кода – благодаря JVM) [5].

Java является одним из самых востребованных языков программирования благодаря своим преимуществам, устойчивости и способности адаптироваться к новым требованиям в процессе разработки программного обеспечения.

Интегрированная среда разработки Visual Studio (Visual C# в среде Visual Studio) – мощная и популярная интегрированная среда разработки (IDE) для создания приложений на языке C#, разработанная компанией Microsoft.

К основным особенностям Visual C# в среде Visual Studio относятся:

– наличие поддержки автозаполнения кода (IntelliSense), удобных инструментов отладки с возможностью просмотра переменных, стеков вызовов и пошагового выполнения кода, возможности интеграции с системами контроля версий, такими как Git;
– наличие поддержки широкого набора библиотек и инструментов .NET Framework и .NET Core (включая .NET 6+) для решения различных задач;
– наличие поддержки различных типов программных решений таких, как десктопные приложения (WPF, WinForms), web-приложения (ASP.NET Core), мобильные приложения (через Xamarin), облачные приложения (интеграция с Azure), игровая разработка (с помощью Unity);
– наличие кроссплатформенности, что позволяет программам, созданным в среде .NET Core и .NET 5/6, работать на Windows, macOS и Linux;

– наличие высокой производительности, которые обеспечивают C# и среда .NET. JIT-компиляция (Just-in-Time) и другие механизмы оптимизации повышают скорость выполнения программных решений [7];
– наличие возможности интеграции с другими продуктами Microsoft – такими как SQL Server, Azure DevOps, Office.

Совместимость с другими программными продуктами Microsoft и наличие кроссплатформенности дает возможность выбрать Visual C# в среде Visual Studio для разработки программного обеспечения в виду перечисленных преимуществ.

2.2.3 Система управления базами данных (СУБД)

В целях проектирования базы данных выбран Microsoft SQL Server 2019 Express.

Microsoft SQL Server 2019 Express – бесплатная, легковесная версия системы управления базами данных (СУБД) Microsoft SQL Server 2019, предназначенная для разработки небольших и средних программных решений [16]. Microsoft SQL Server 2019 Express предоставляет основные функции, необходимые для управления данными, включая поддержку реляционных баз данных, T-SQL (Transact-SQL) запросов и хранение данных [12].

К основным особенностям SQL Server 2019 Express относятся:

– наличие бесплатного распространения с возможностью создания приложений с базами данных без необходимости покупки лицензии;
– наличие ограничений по объему базы данных (до 10 ГБ), по количеству подключений, по поддержке таких продвинутых функций, не ограничивающих возможности обеспечивать достаточно высокую производительность для небольших и средних программных решений;

– наличие упрощенной установки и минимальных требований к настройке;
– наличие поддержки ОС Windows, включая специальные сервисы, такие как Microsoft Reporting Services;
– наличие возможности масштабируемости и перехода на более мощные версии SQL Server (Standard, Enterprise) без значительных изменений в коде или структуре данных [11];
– наличие поддержки Transact-SQL (T-SQL) – расширенного диалекта SQL, разработанного Microsoft;
– наличие обеспечения высокого уровня безопасности за счет поддержки шифрования данных, аутентификации и возможность управления доступом на уровне пользователя или роли;
– наличие поддержки резервного копирования и восстановления данных;
– ориентированность на работу в среде ОС Windows.

SQL Server 2019 Express – отличный выбор для разработки небольших и средних программных решений, где важны простота настройки, бесплатность и интегрированность с продуктами Microsoft с возможностью масштабироваться, в случае необходимости, до более мощных решений на базе SQL Server.

2.3 Концептуальное проектирование базы данных

Описание предметной области в целях формирования концептуальной модели базы данных приведено ниже.

Информационные ресурсы ЦГБ МАУК «ЦГБС» для работы используют пользователи, имеющие роли – библиотекарь или читатель.

Пользователи с ролью «Библиотекарь» осуществляет ввод данных в справочники «Пользователи», «Инфоресурсы», «Вид инфоресурсов»,

«Авторы», «Жанры», «Издательства», «Причины списания», закрывают бронь на инфоресурсы в программном обеспечении, осуществляют выдачу и списание инфоресурсов.

Справочник «Инфоресурсы» включает в себя: инвентарный номер инфоресурса (в случае наличия), вид инфоресурса, автор (в случае наличия), издательство (в случае наличия), жанр (в случае наличия), название, год выпуска, на руках.

Пользователи с ролью «Читатель» оформляют бронь на информационный ресурс в программном обеспечении, после чего информация о брони появляется в журнале бронирования.

После выдачи информационного ресурса в программном решении формируется журнал выдачи.

При списании информационного ресурса формируется журнал списания с указанием причины списания.

После описания предметной области сформирована концептуальная модель базы данных, которая представлена на рисунке 13.

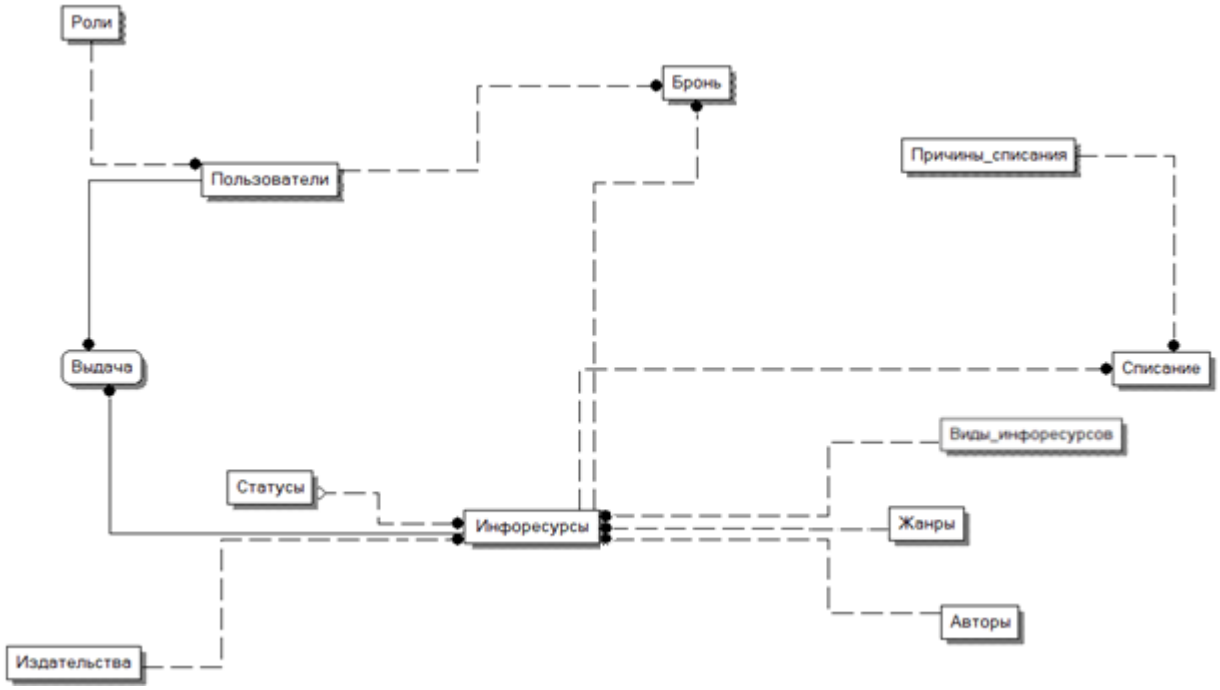


Рисунок 13 – Концептуальная модель базы данных

В концептуальной модели данных, таким образом, представлены следующие сущности:

– пользователи;

– роли (библиотекарь и читатель);

– инфоресурсы;

– виды инфоресурсов;

– авторы;

– издательства;

– жанры;

– причины списания;

– статусы;

– журнал бронирования;

– журнал списания;

– журнал выдачи.

2.4 Логическое моделирование базы данных

После проведения концептуального моделирования базы данных программного обеспечения требуется логическое моделирование его базы данных.

В процессе логического моделирования базы данных программного обеспечения добавляются новые атрибуты к ранее смоделированным сущностям.

Логическая модель базы данных программного обеспечения представлена на рисунке 14.

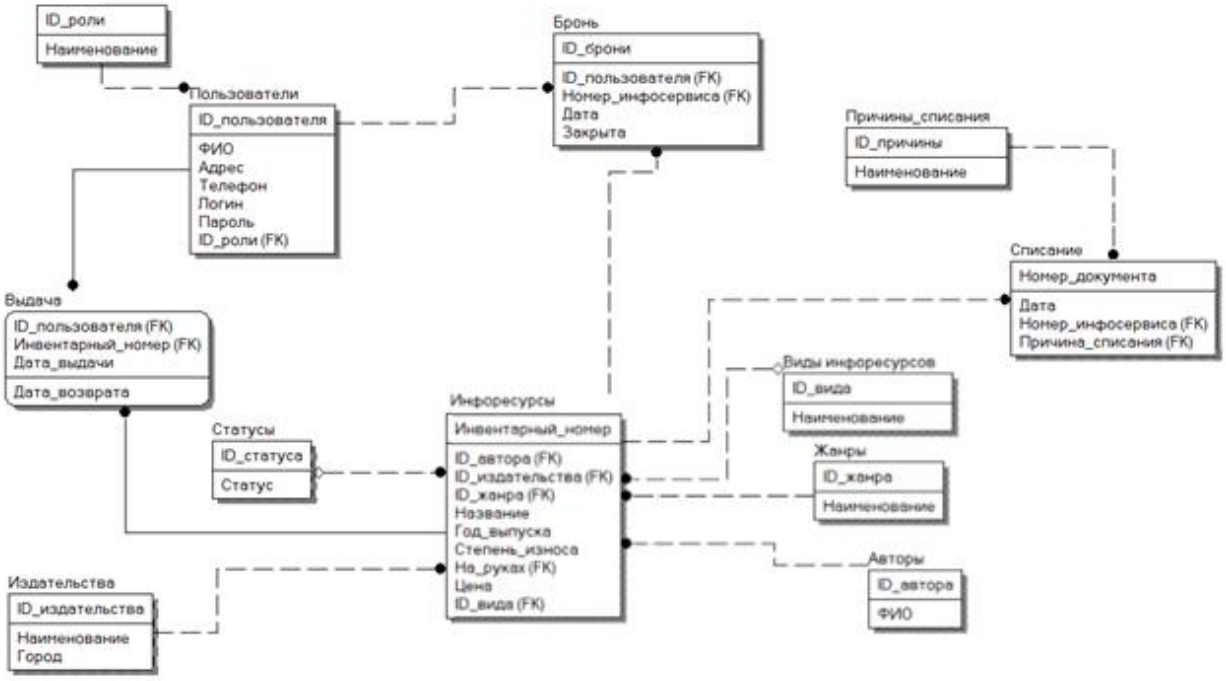


Рисунок 14 – Логическая модель базы данных

Таким образом, логическая модель базы данных включает в себя все необходимые сущности, атрибуты, ключи и взаимосвязи.

2.5 Физическое моделирование базы данных

После логического моделирования базы данных программного обеспечения требуется физическое моделирование его базы данных.

В процессе физического моделирования базы данных программного обеспечения добавляются типы данных к ранее смоделированным атрибутам сущностей.

Физическая модель базы данных программного обеспечения представлена на рисунке 15.

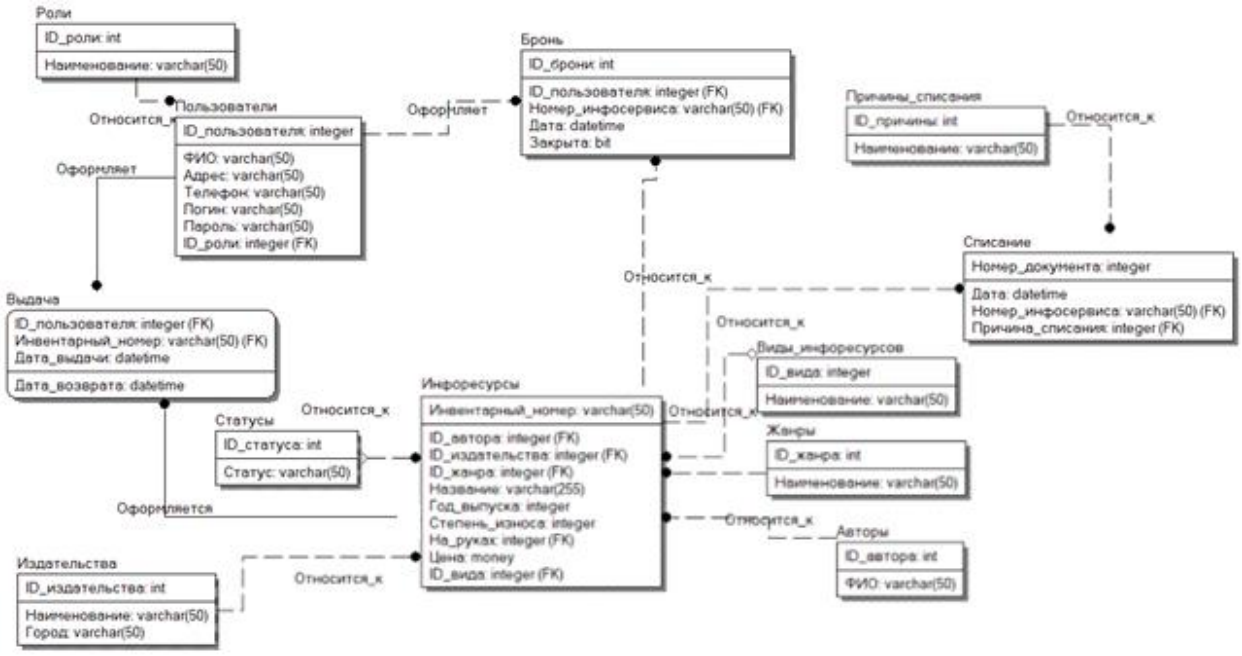
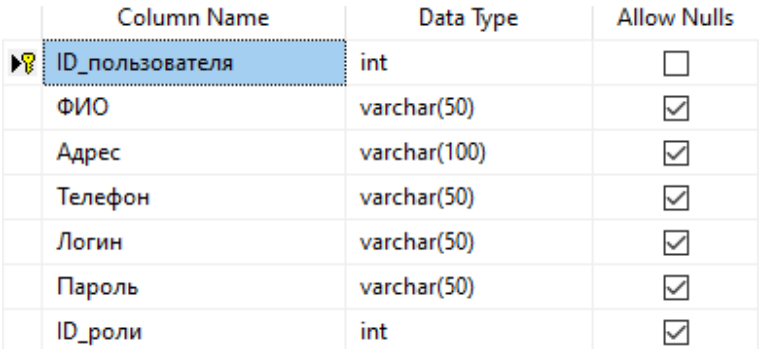


Рисунок 15 – Физическая модель базы данных

Подробный состав таблиц базы данных программного обеспечения, с указанием для каждого атрибута его названия, типа данных, длины полей и параметров ключа, приводится ниже.

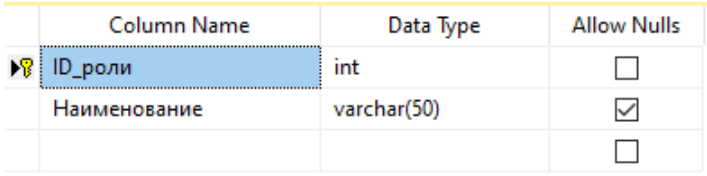
Состав таблицы «Пользователи» приведен на рисунке 16.



Column Name	Data Type	Allow Nulls
ID_пользователя	int	☐
ФИО	varchar(50)	☑
Адрес	varchar(100)	☑
Телефон	varchar(50)	☑
Логин	varchar(50)	☑
Пароль	varchar(50)	☑
ID_роли	int	☑

Рисунок 16 – Состав таблицы «Пользователи»

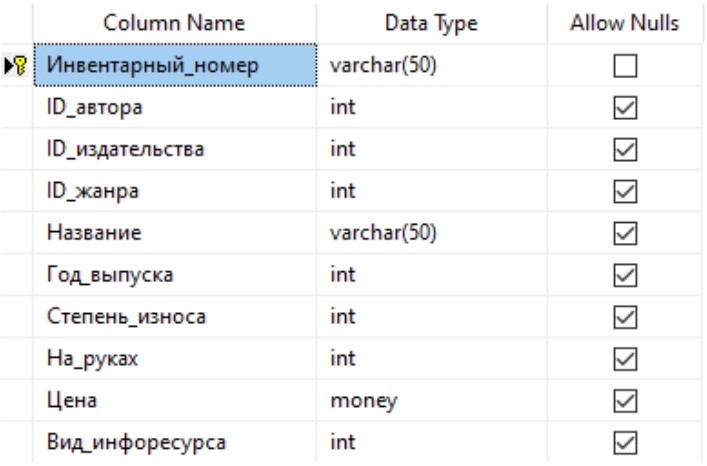
Состав таблицы «Роли» приведен на рисунке 17.



Column Name	Data Type	Allow Nulls
ID_роли	int	☐
Наименование	varchar(50)	☑
		☐

Рисунок 17 – Состав таблицы «Роли»

Состав таблицы «Инфоресурсы» приведен на рисунке 18.



Column Name	Data Type	Allow Nulls
Инвентарный_номер	varchar(50)	☐
ID_автора	int	☑
ID_издательства	int	☑
ID_жанра	int	☑
Название	varchar(50)	☑
Год_выпуска	int	☑
Степень_износа	int	☑
На_руках	int	☑
Цена	money	☑
Вид_инфоресурса	int	☑

Рисунок 18 – Состав таблицы «Инфоресурсы»

Состав таблицы «Издательства» приведен на рисунке 19.

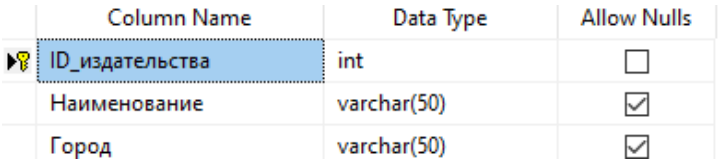


Рисунок 19 – Состав таблицы «Издательства»

Состав таблицы «Жанры» приведен на рисунке 20.

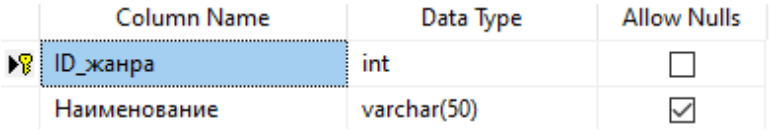


Рисунок 20 – Состав таблицы «Жанры»

Состав таблицы «Статусы» приведен на рисунке 21.

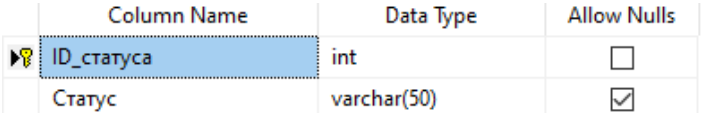


Рисунок 21 – Состав таблицы «Статусы»

Состав таблицы «Авторы» приведен на рисунке 22.

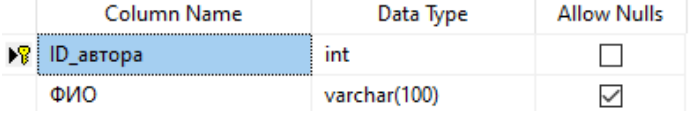
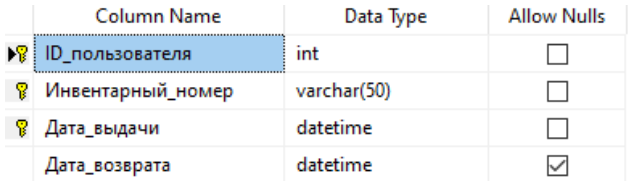


Рисунок 22 – Состав таблицы «Авторы»

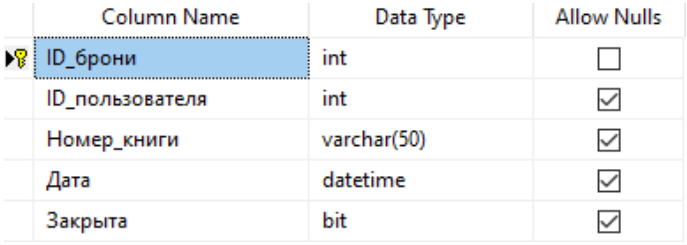
Состав таблицы «Выдача» приведен на рисунке 23.



Column Name	Data Type	Allow Nulls
ID_пользователя	int	☐
Инвентарный_номер	varchar(50)	☐
Дата_выдачи	datetime	☐
Дата_возврата	datetime	☑

Рисунок 23 – Состав таблицы «Выдача»

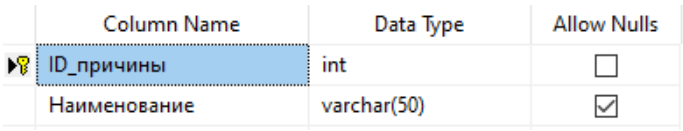
Состав таблицы «Бронь» приведен на рисунке 24.



Column Name	Data Type	Allow Nulls
ID_брони	int	☐
ID_пользователя	int	☑
Номер_книги	varchar(50)	☑
Дата	datetime	☑
Закрыта	bit	☑

Рисунок 24 – Состав таблицы «Бронь»

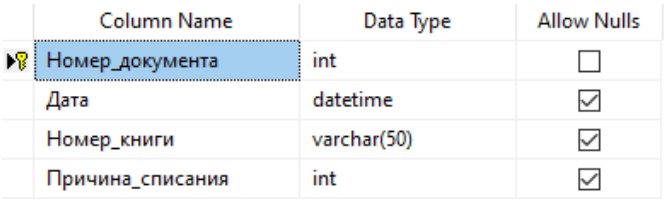
Состав таблицы «Причины списания» приведен на рисунке 25.



Column Name	Data Type	Allow Nulls
ID_причины	int	☐
Наименование	varchar(50)	☑

Рисунок 25 – Состав таблицы «Причины списания»

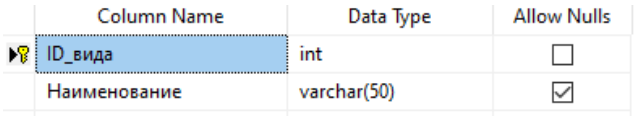
Состав таблицы «Списание» приведен на рисунке 26.



Column Name	Data Type	Allow Nulls
Номер_документа	int	☐
Дата	datetime	☑
Номер_книги	varchar(50)	☑
Причина_списания	int	☑

Рисунок 26 – Состав таблицы «Списание»

Состав таблицы «Виды инфоресурсов» приведен на рисунке 27.



Column Name	Data Type	Allow Nulls
ID_вида	int	☐
Наименование	varchar(50)	☑

Рисунок 27 – Состав таблицы «Виды инфоресурсов»

Подробный состав таблиц базы данных программного обеспечения позволяет продемонстрировать, что каждая таблица базы данных соответствует определенной сущности предметной области.

2.6 Сценарий создания базы данных

В инструментальной среде для проектирования и документирования баз данных ErWin Data Modeler реализован сценарий создания базы данных [17], для чего в ErWin Data Modeler использовалась функция Forward Engeneer/Schema Generation [19].

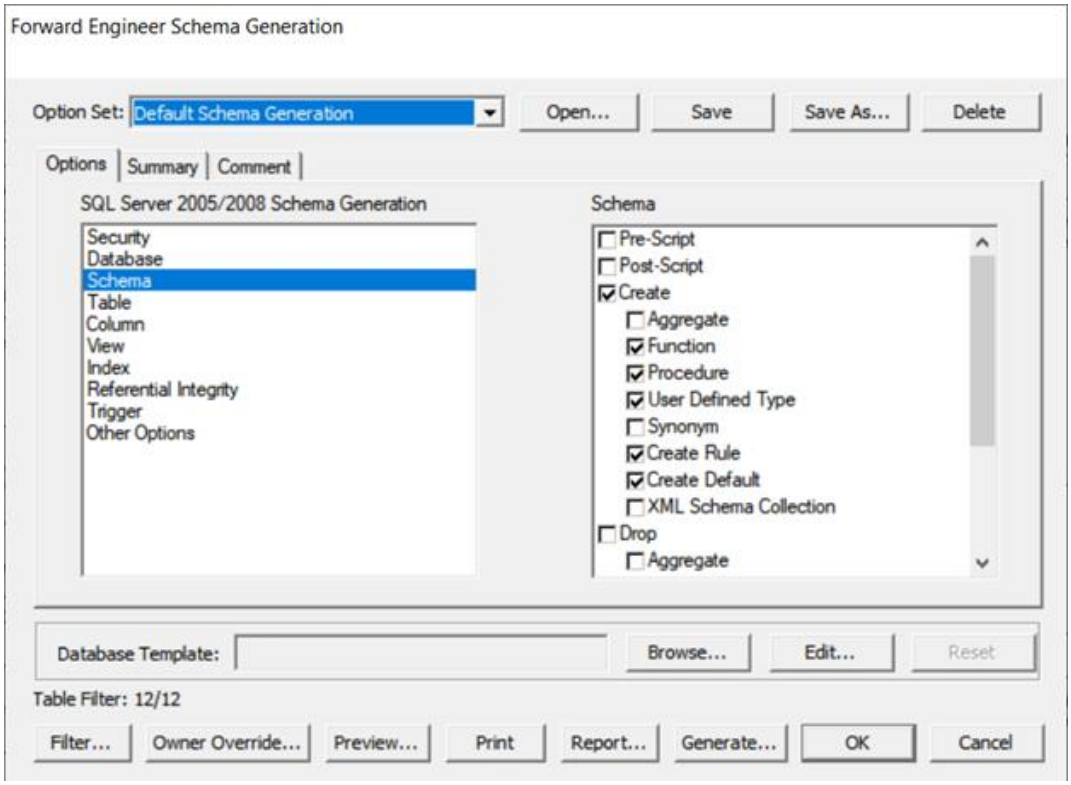


Рисунок 28 – Функция Forward Engeneer/Schema Generation

Сценарий создания базы данных реализован на основе созданной модели базы данных программного обеспечения [18].

Выводы по главе 2

В данной главе определены функциональные требования к программному обеспечению, выбраны инструменты проектирования и разработки, проведено концептуальное, логическое, физическое проектирование базы данных и реализована база данных.

Глава 3 Реализация и тестирование программного обеспечения цифровизации муниципальных услуг

3.1 Разработка программного обеспечения цифровизации муниципальных услуг

Основной функционал программного обеспечения реализован с помощью визуальных компонентов. Формирование справочников в программном решении требует динамического формирования запросов к базе данных на основании данных, введенных пользователем [14], в связи с чем для их функционирования используется программный код.

В программном обеспечения реализован механизм взаимосвязи с базой данных.

Для удобства пользователей, в компонентах типа DataGridView реализована автоматическая подстановка значений из различных таблиц базы данных на основании их содержимого пользователя – вместо ключевых значений пользователь видит в таблице соответствующие значения из БД [3].

Ниже описана реализация данного механизма.

После размещения компонента типа DataGridView на форме необходимо задать для него источник данных, нажав на иконку стрелки в правой части формы и указав источник данных.

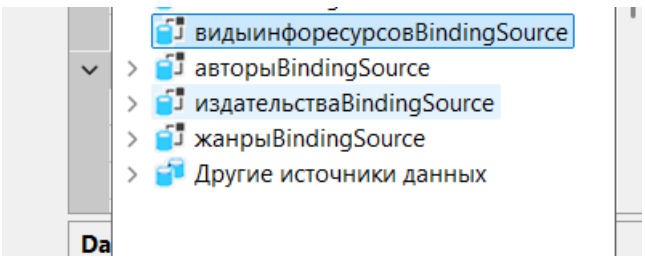


Рисунок 29 – Указание источника данных для компонента

После указания источника данных на форме «Правка столбцов» появится таблица «Свойства привязанного столбца» [20]. Для настройки отображения данных в столбце необходимо нажать правой кнопкой мыши на компонент и выбрать «Правка столбцов», после чего откроется форма настройки столбцов.

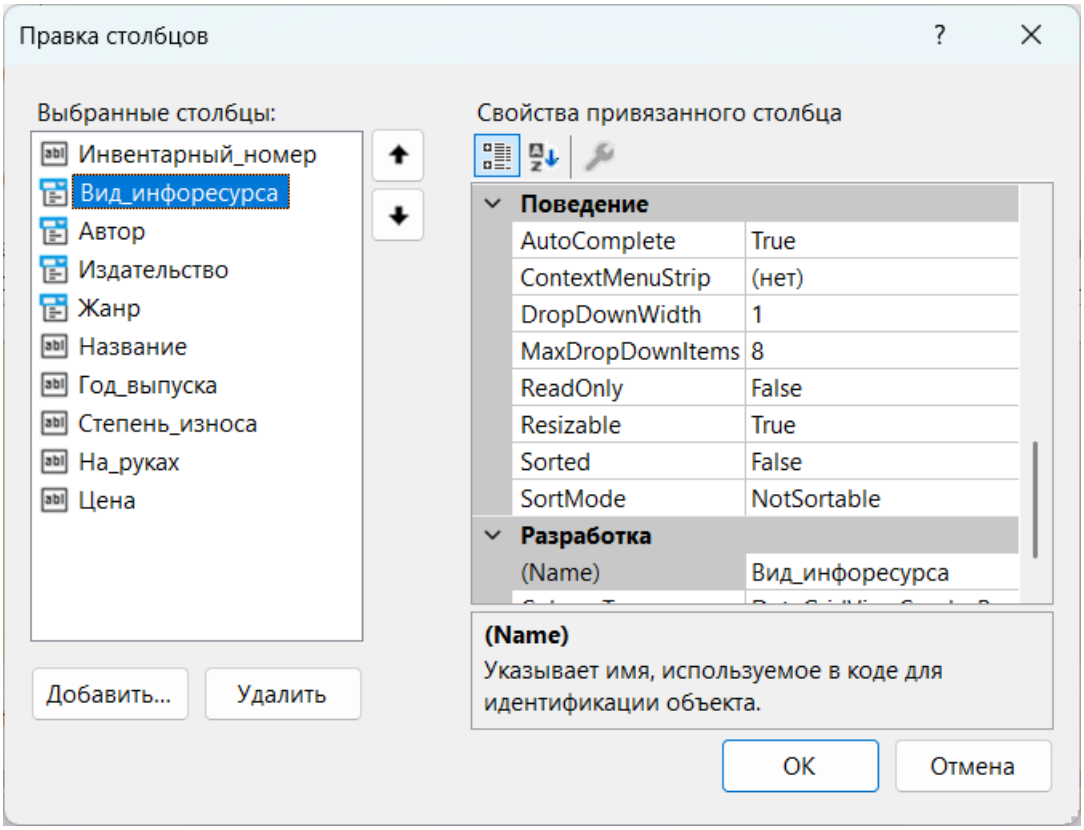


Рисунок 30 – Форма настройки столбцов

В форме настройки столбцов необходимо выбрать столбец, требующий внесения изменений, например, столбец «ID_автора», после чего в правой части формы появятся настройки столбца.

Для изменения типа столбца, из выпадающего списка в поле свойства ColumnType необходимо выбрать тип столбца DataGridViewComboBoxColumn [4].

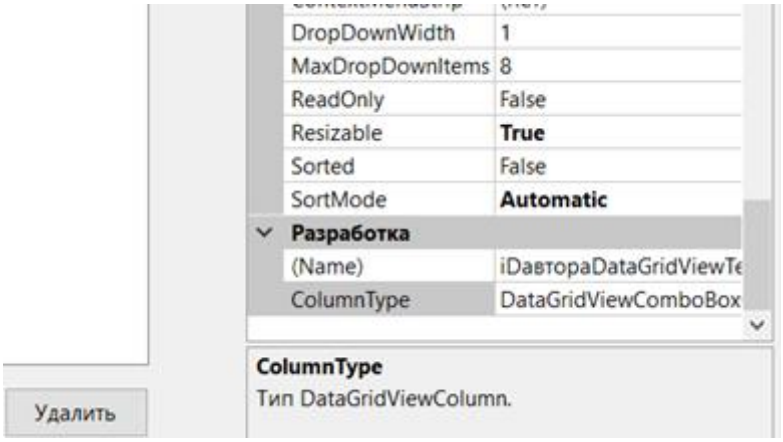


Рисунок 31 – Выбор типа столбца

После изменения типа столбца вид формы свойств привязанного столбца изменится – в ней появятся новые свойства.

Для указания связанной таблицы необходимо перейти в форме «Свойства привязанного столбца» к группе свойств «Данные».

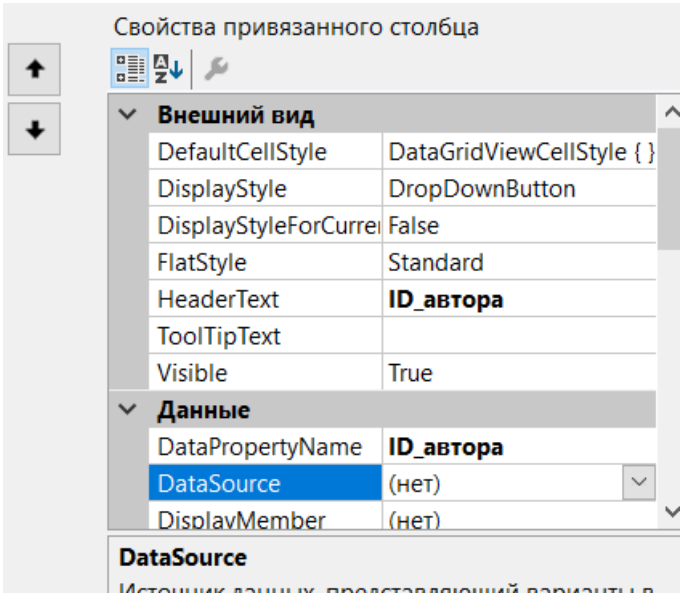


Рисунок 32 – Группа свойств «Данные»

Источник данных необходимо указать в свойстве DataSource.

Столбце таблицы, который будет отображаться, необходимо указать в DisplayMember.

Значения из столбца, которые должны быть сопоставлены с таблицей, необходимо указать в свойстве ValueMember.

В результате редактирования привязанного столбца получается следующий результат:

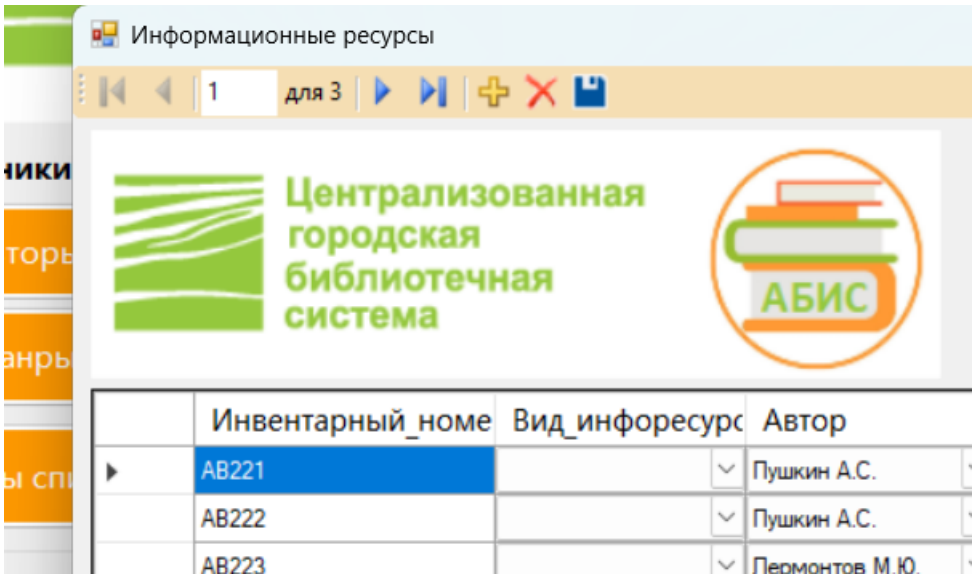


Рисунок 33 – Результат редактирования привязанного столбца

Таким образом, выбор автора станет доступен из выпадающего списка.

Остальные столбца настраиваются в форме аналогичным образом.

Далее необходимо настроить визуальные компоненты редактирования, размещенные на форме.

Для того, чтобы настроить компонент ComboBox для отображения данных, необходимо нажать на иконку стрелку справа от него, после чего поставить галочку «Использовать компоненты, привязанные к данным» [5].

В окне появятся новые свойства.

В строке «Источник данных» необходимо указать источник данных для компонента.

В строке «Отобразить члена» необходимо указать столбец таблицы, который будет отображаться в компоненте.

В строке «Член значения» необходимо указать столбец для автоподстановки.

В строке «Выбранное значение» указывается столбец, в котором необходимо сохранять значения.

3.2 Схема взаимосвязи программных модулей

Схема взаимосвязи программных модулей приведена на рисунке 34.

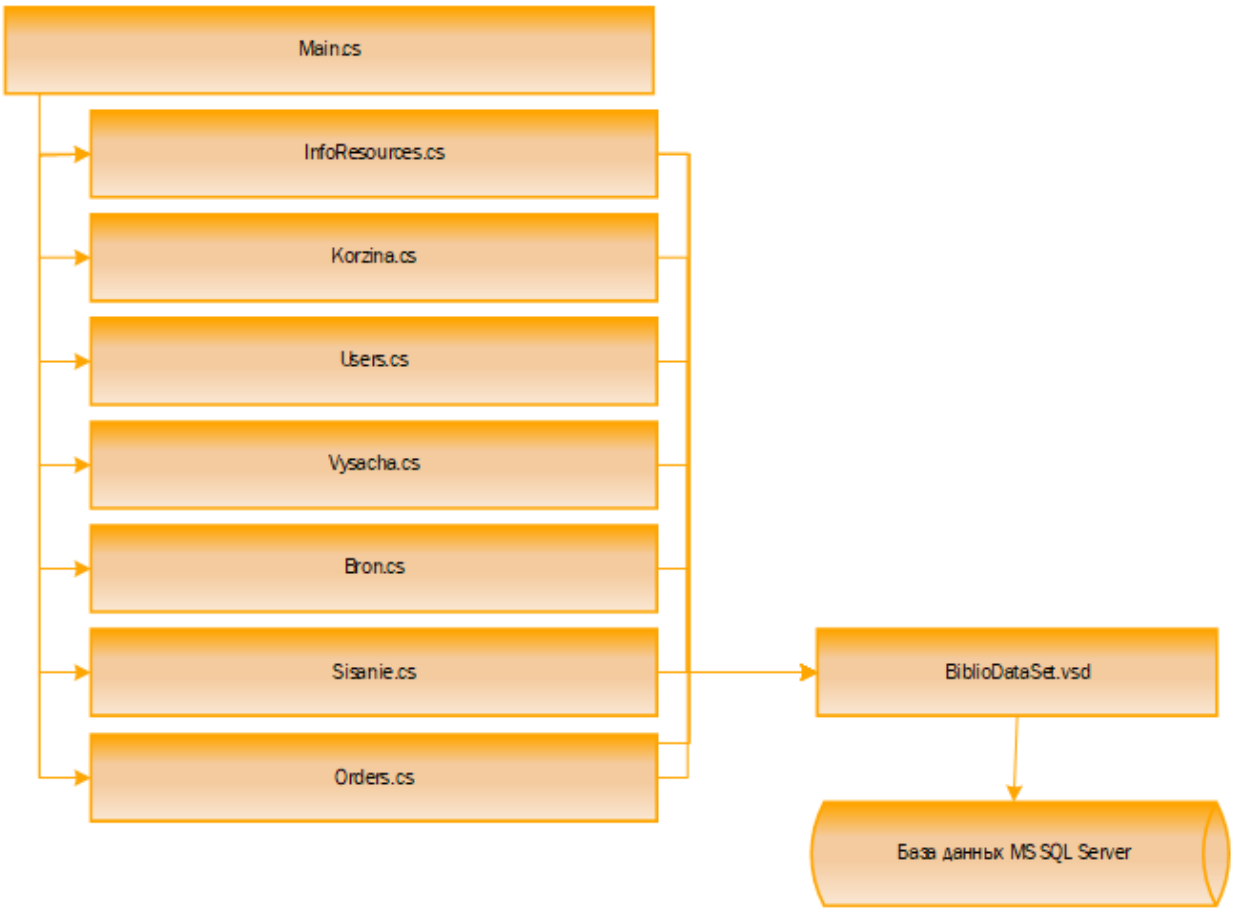


Рисунок 34 – Схема взаимосвязи программных модулей

Схема взаимосвязи программных модулей отображает структуру программного обеспечения, его модули (составные части) и связь между ними.

3.3 Реализованный сценарий базы данных

Ниже представлен реализованный сценарий создания базы данных программного обеспечения для среды MS SQL Server.

Сценарий создания в базе данных таблицы «Авторы» и ключа «ID_автора» представлен на рисунке 35.

```
CREATE TABLE Авторы
(
    ID_автора            int IDENTITY ,
    ФИО                  varchar(50)  NULL
)
go


ALTER TABLE Авторы
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (ID_автора ASC)
go
```

Рисунок 35 – Сценарий создания в базе данных таблицы «Авторы» и ключа «ID_автора»

Сценарий создания в базе данных таблицы «Бронь» и ключа «ID_брони» представлен на рисунке 36.

```
CREATE TABLE Бронь
(
    ID_брони              int IDENTITY ,
    ID_пользователя       integer  NOT NULL ,
    Номер_Инфоресурсы           varchar(50)  NOT NULL ,
    Дата                  datetime  NULL ,
    Закрыта               bit  NULL
    CONSTRAINT DEFAULT_1_No_2023925958
         DEFAULT  No
)
go


ALTER TABLE Бронь
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (ID_брони ASC)
go
```

Рисунок 36 – Сценарий создания в базе данных таблицы «Бронь» и ключа «ID_брони»

Сценарий создания в базе данных таблицы «Виды инфоресурсов» и ключа «ID_вида» представлен на рисунке 37.

```
CREATE TABLE Виды_инфоресурсов
(
    ID_вида               integer  NOT NULL ,
    Наименование          varchar(50)  NULL
)
go


ALTER TABLE Виды_инфоресурсов
    ADD CONSTRAINT XPKВиды_инфоресурсов PRIMARY KEY  CLUSTERED (ID_вида ASC)
go
```

Рисунок 37 – Сценарий создания в базе данных таблицы «Виды инфоресурсов» и ключа «ID_вида»

Сценарий создания в базе данных таблицы «Выдача», ключа «Дата_выдачи» и взаимосвязи с ключами других таблиц представлен на рисунке 38.

```
CREATE TABLE Выдача
(
    ID_пользователя      integer  NOT NULL ,
    Инвентарный_номер    varchar(50)  NOT NULL ,
    Дата_выдачи          datetime  NOT NULL ,
    Дата_возврата        datetime  NULL
)
go


ALTER TABLE Выдача
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (ID_пользователя ASC,Инвентарный_номер ASC,Дата_выдачи ASC)
go
```

Рисунок 38 – Сценарий создания в базе данных таблицы «Выдача», ключа «Дата_выдачи» и взаимосвязи с ключами других таблиц

Сценарий создания в базе данных таблицы «Жанры» и ключа «ID_жанра» представлен на рисунке 39.

```
CREATE TABLE Жанры
(
    ID_жанра             int IDENTITY ,
    Наименование         varchar(50)  NULL
)
go


ALTER TABLE Жанры
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (ID_жанра ASC)
go
```

Рисунок 39 – Сценарий создания в базе данных таблицы «Жанры» и ключа «ID_жанра»

Сценарий создания в базе данных таблицы «Издательства» и ключа «ID_издательства» представлен на рисунке 40.

```
CREATE TABLE Издательства
(
    ID_издательства      int IDENTITY ,
    Наименование         varchar(50)  NULL ,
    Город                varchar(50)  NULL
)
go


ALTER TABLE Издательства
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (ID_издательства ASC)
go
```

Рисунок 40 – Сценарий создания в базе данных таблицы «Жанры» и ключа «ID_жанра»

Сценарий создания в базе данных таблицы «Инфоресурсы» и ключа «Инвентарный номер» представлен на рисунке 41.

```
CREATE TABLE Инфоресурсы
(
    Инвентарный_номер    varchar(50)  NOT NULL ,
    ID_автора            integer  NOT NULL ,
    ID_издательства      integer  NOT NULL ,
    ID_жанра             integer  NOT NULL ,
    Название             varchar(50)  NULL ,
    Год_выпуска          integer  NULL ,
    Степень_износа       integer  NULL ,
    На_руках             integer  NULL ,
    Цена                 money  NULL ,
    ID_вида              integer  NULL
)
go


ALTER TABLE Инфоресурсы
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (Инвентарный_номер ASC)
go
```

Рисунок 41 – Сценарий создания в базе данных таблицы «Инфоресурсы» и ключа «Инвентарный номер»

Сценарий создания в базе данных таблицы «Пользователи» и ключа «ID_пользователя» представлен на рисунке 42.

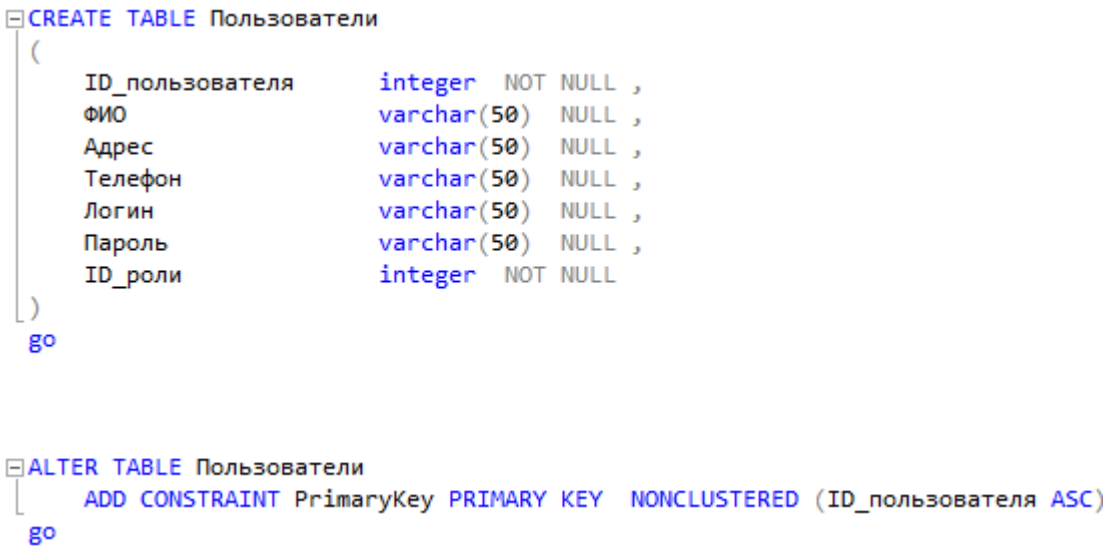


Рисунок 42 – Сценарий создания в базе данных таблицы «Пользователи» и ключа «ID_пользователя»

Сценарий создания в базе данных таблицы «Причины_списания» и ключа «ID_причины» представлен на рисунке 43.

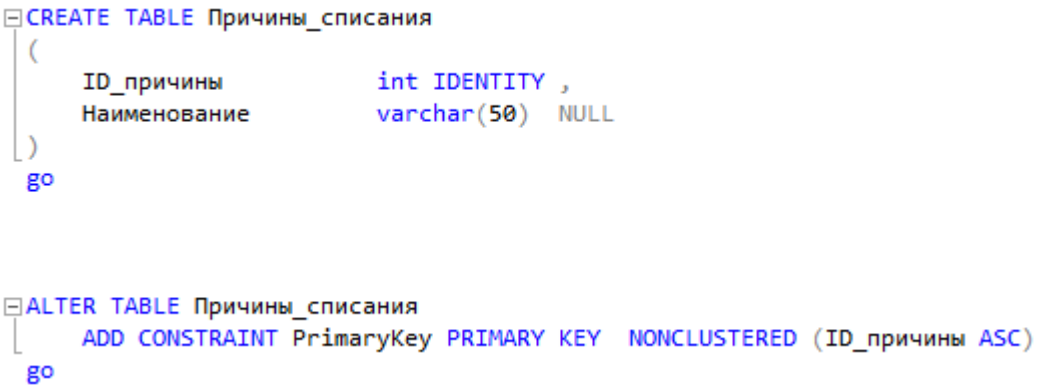


Рисунок 43 – Сценарий создания в базе данных таблицы «Причины_списания» и ключа «ID_причины»

Сценарий создания в базе данных таблицы «Роли» и ключа «ID_роли» представлен на рисунке 44.

```
CREATE TABLE Роли
(
    ID_роли              int IDENTITY ,
    Наименование         varchar(50)  NULL
)
go


ALTER TABLE Роли
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (ID_роли ASC)
go
```

Рисунок 44 – Сценарий создания в базе данных таблицы «Роли» и ключа «ID_роли»

Сценарий создания в базе данных таблицы «Списание» и ключа «Номер документа» представлен на рисунке 45.

```
CREATE TABLE Списание
(
    Номер_документа      integer  NOT NULL ,
    Дата                 datetime  NULL ,
    Номер_Инфоресурсы          varchar(50)  NOT NULL ,
    Причина_списания     integer  NOT NULL
)
go


ALTER TABLE Списание
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (Номер_документа ASC)
go
```

Рисунок 45 – Сценарий создания в базе данных таблицы «Списание» и ключа «Номер документа»

Сценарий создания в базе данных таблицы «Статусы» и ключа «ID_статуса» представлен на рисунке 46.

```
CREATE TABLE Статусы
(
    ID_статуса          int IDENTITY ,
    Статус              varchar(50)  NULL
)
go


ALTER TABLE Статусы
    ADD CONSTRAINT PrimaryKey PRIMARY KEY  NONCLUSTERED (ID_статуса ASC)
go
```

Рисунок 46 – Сценарий создания в базе данных таблицы «Статусы» и ключа «ID_статуса»

Сценарий создания в базе данных таблицы «Бронь» внешних ключей других таблиц представлен на рисунке 47.

```
ALTER TABLE Бронь
    ADD CONSTRAINT Оформляет FOREIGN KEY (ID_пользователя) REFERENCES Пользователи(ID_пользователя)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go


ALTER TABLE Бронь
    ADD CONSTRAINT Относится_к FOREIGN KEY (Номер_Инфоресурсы) REFERENCES Инфоресурсы(Инвентарный_номер)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
```

Рисунок 47 – Сценарий создания в базе данных таблицы «Бронь» внешних ключей других таблиц

Сценарий создания в базе данных таблицы «Выдача» внешних ключей других таблиц представлен на рисунке 48.

```
ALTER TABLE Выдача
    ADD CONSTRAINT Оформляет FOREIGN KEY (ID_пользователя) REFERENCES Пользователи(ID_пользователя)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
```

```
ALTER TABLE Выдача
    ADD CONSTRAINT Оформляется FOREIGN KEY (Инвентарный_номер) REFERENCES Инфоресурсы(Инвентарный_номер)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
```

Рисунок 48 – Сценарий создания в базе данных таблицы «Выдача» внешних ключей других таблиц

Сценарий создания в базе данных таблицы «Инфоресурсы» внешних ключей других таблиц представлен на рисунке 49.

```
ALTER TABLE Инфоресурсы
    ADD CONSTRAINT Относится_к FOREIGN KEY (ID_автора) REFERENCES Авторы(ID_автора)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go

ALTER TABLE Инфоресурсы
    ADD CONSTRAINT Относится_к FOREIGN KEY (ID_жанра) REFERENCES Жанры(ID_жанра)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go

ALTER TABLE Инфоресурсы
    ADD CONSTRAINT Относится_к FOREIGN KEY (ID_издательства) REFERENCES Издательства(ID_издательства)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go

ALTER TABLE Инфоресурсы
    ADD CONSTRAINT Относится_к FOREIGN KEY (На_руках) REFERENCES Статусы(ID_статуса)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go

ALTER TABLE Инфоресурсы
    ADD CONSTRAINT Относится_к FOREIGN KEY (ID_вида) REFERENCES Виды_инфоресурсов(ID_вида)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
```

Рисунок 49 – Сценарий создания в базе данных таблицы «Инфоресурсы» внешних ключей других таблиц

Сценарий создания в базе данных таблицы «Пользователи» внешнего ключа другой таблицы представлен на рисунке 50.

```
ALTER TABLE Пользователи
    ADD CONSTRAINT Относится_к FOREIGN KEY (ID_роли) REFERENCES Роли(ID_роли)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
```

Рисунок 50 – Сценарий создания в базе данных таблицы ««Пользователи» внешнего ключа другой таблицы

Сценарий создания в базе данных таблицы «Списание» внешних ключей других таблиц представлен на рисунке 51.

```
ALTER TABLE Списание
    ADD CONSTRAINT Относится_к FOREIGN KEY (Причина_списания) REFERENCES Причины_списания(ID_причины)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go

ALTER TABLE Списание
    ADD CONSTRAINT Оформляется FOREIGN KEY (Номер_Инфоресурсы) REFERENCES Инфоресурсы(Инвентарный_номер)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
```

Рисунок 51 – Сценарий создания в базе данных таблицы «Списание» внешних ключей других таблиц

Пользователи имеют возможность взаимодействовать с программнымобеспечением через его интерфейс. Проектирование интерфейса включает в себя не только разработку визуально привлекательного дизайна, но и включает в себя учет потребностей пользователей и учет их ожиданий, реализацию интуитивно понятного интерфейса для всех пользователей [13].

3.4 Проектирование и разработка интерфейса

В целях проектирования макетов интерфейса использован векторный графический редактор Microsoft Visio.

При создании дизайна пользовательского интерфейса учитывалась его интуитивная понятность, а также его соответствие функциональным требованиям и ожиданиям потенциальных пользователей. Макет главной

формы при авторизации пользователя с ролью «Библиотекарь» представлен на рисунке 52.

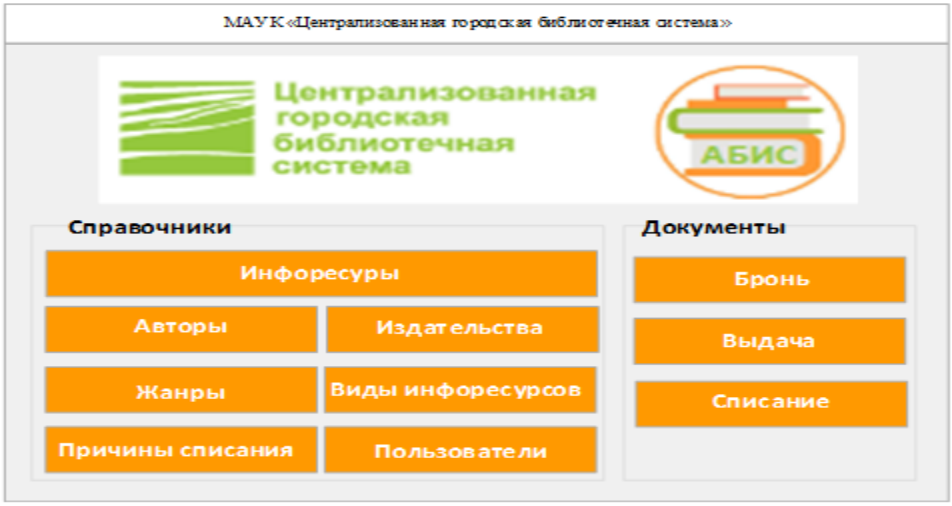


Рисунок 52 – Макет главной формы при авторизации пользователя с ролью «Библиотекарь»

Пользовательский интерфейс выступает в качестве связующего звена между пользователями и функционалом программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек, позволяя комфортно и эффективно взаимодействовать с ним.

Все реализованные формы интерфейса программного обеспечения цифровизации муниципальных услуг приведены в Приложении А.

3.5 Тестирование программного обеспечения цифровизации муниципальных услуг

Проведено успешное функциональное тестирование разработанного программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек. Сценарий, описание и результаты тестирования приведены в таблице 2.

Таблица 2 – Сценарий тестирования

Сценарий тестирования	Описание тестирования	Результат тестирования
Шаг 1. Проверка формы авторизации	Ввод в форме авторизации: – некорректных логина и пароля пользователем с ролью «Библиотекарь»; – корректных логина и пароля пользователем с ролью «Библиотекарь»; – некорректных логина и пароля пользователем с ролью «Читатель»; – корректных логина и пароля пользователем с ролью «Читатель»	Ошибок не выявлено
Шаг 2. Проверка формы «Пользователи»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – ФИО; – адрес; – телефон; – логин; – пароль; – ID роли (1 – библиотекарь; 2 – читатель)	Ошибок не выявлено
Шаг 3. Проверка формы «Виды инфоресурсов»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – наименование	Ошибок не выявлено
Шаг 4. Проверка формы «Виды инфоресурсов»	Просмотр пользователем с ролью «Читатель» следующих данных в форме: – наименование	Ошибок не выявлено
Шаг 5. Проверка формы «Авторы»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – ФИО	Ошибок не выявлено
Шаг 6. Проверка формы «Авторы»	Просмотр пользователем с ролью «Читатель» следующих данных в форме: – ФИО	Ошибок не выявлено
Шаг 7. Проверка формы «Жанры»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – наименование	Ошибок не выявлено
Шаг 8. Проверка формы «Жанры»	Просмотр пользователем с ролью «Читатель» следующих данных в форме: – наименование	Ошибок не выявлено
Шаг 9. Проверка формы «Издательства»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – наименование – город	Ошибок не выявлено

Продолжение Таблицы 2

Сценарий тестирования	Описание тестирования	Результат тестирования
Шаг 10.Проверка формы «Издательства»	Просмотр пользователем с ролью «Читатель» следующих данных в форме: – наименование город	Ошибок не выявлено
Шаг 11.Проверка формы «Причины списания»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – наименование	Ошибок не выявлено
Шаг 12.Проверка формы «Инфоресурсы»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – инвентарный номер – вид инфоресурса – автор – издательство – жанр – название – год выпуска – степень износа – на руках – цена	Ошибок не выявлено
Шаг 13.Проверка формы «Инфоресурсы»	Просмотр пользователем с ролью «Читатель» следующих данных в форме: – инвентарный номер – вид инфоресурса – автор – издательство – жанр – название – год выпуска – степень износа – на руках – цена	Ошибок не выявлено
Шаг 14. Проверка формы «Бронь»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – наименование инфоресурса – дата (брони) – закрыто (наличие/отсутствие отметки)	Ошибок не выявлено
Шаг 15. Проверка формы «Бронь»	Просмотр пользователем с ролью «Читатель» следующих данных в форме: – наименование инфоресурса – дата (брони) – закрыто (наличие/отсутствие отметки)	Ошибок не выявлено

Продолжение Таблицы 2

Сценарий тестирования	Описание тестирования	Результат тестирования
Шаг 16.Проверка формы «Выдача»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – пользователь – инфоресурс – дата выдачи дата возврата	Ошибок не выявлено
Шаг 17. Проверка формы «Списание»	Ввод/удаление/сохранение пользователем с ролью «Библиотекарь» следующих данных в форме: – дата – инфоресурс – причина списания	Ошибок не выявлено

Пример проверки функции шага 1 сценария тестирования (ввод некорректных логина и пароля читателя в форму авторизации) приведен на рисунке 53.

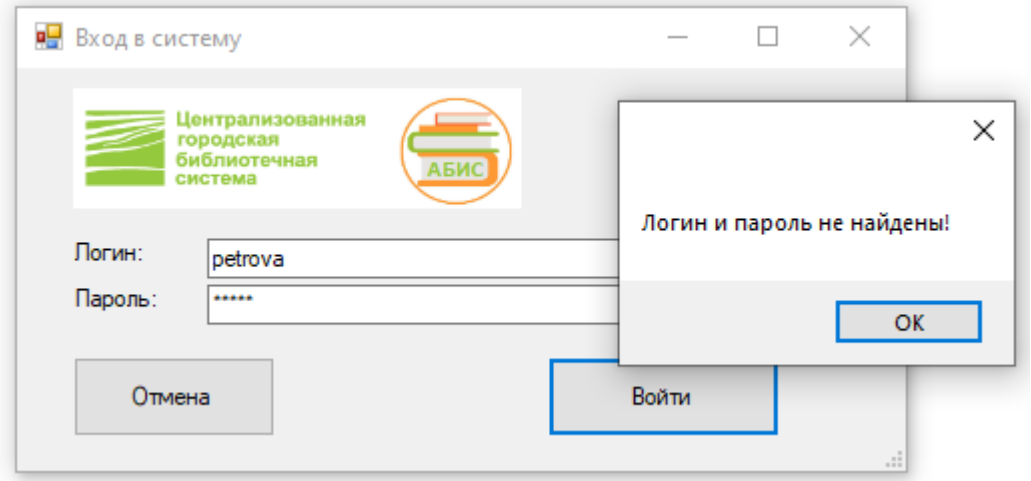


Рисунок 53 – Пример проверки функции шага 1 сценария тестирования (ввод некорректных логина и пароля читателя в форму авторизации)

Проведено успешное тестирование SQL-скрипта (сценария) базы данных на предмет отсутствия ошибок в синтаксисе. Тестирование

выполнено в инструменте Microsoft SQL Server Management Studio в окне New Query. Результаты данного тестирования приведены на рисунке 54.

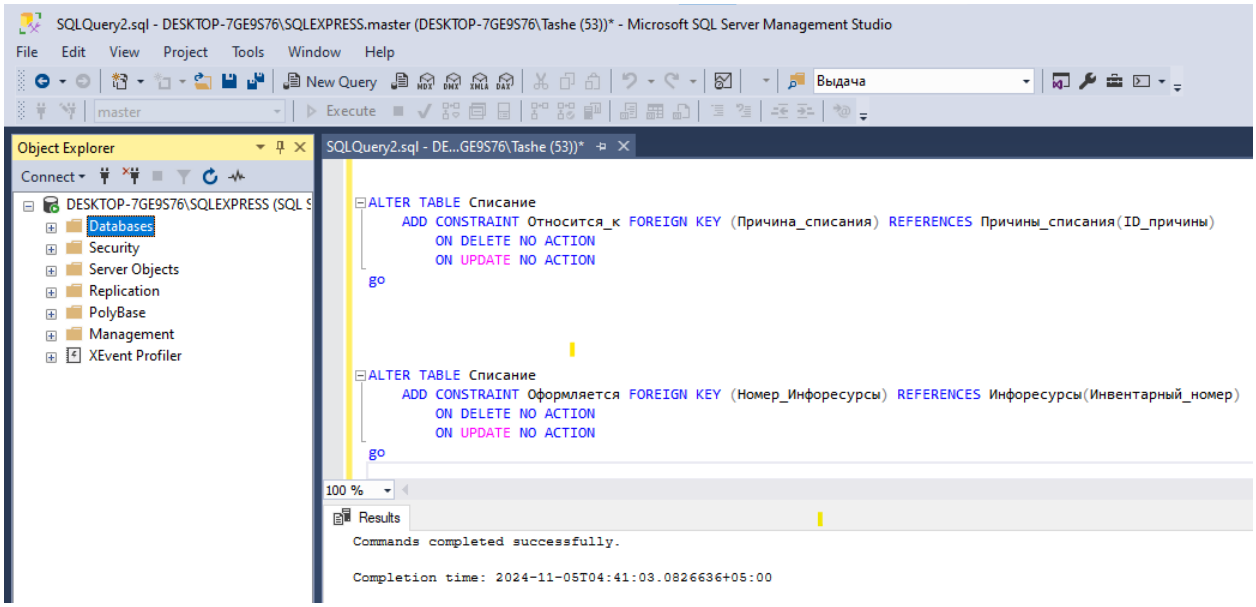


Рисунок 54 – Результат тестирования SQL-скрипта (сценария) базы данных

Выводы по главе 3

В данной главе описана реализация функционала программного обеспечения с помощью визуальных компонентов, описаны механизмы взаимосвязи с базой данных и процесс настройки данных в формах компонентов, приведен сценарий базы данных, осуществлены проектирование и разработка интерфейса программного обеспечения.

Далее проведены функциональное тестирование разработанного программного обеспечения цифровизации муниципальных услуг для общественных городских библиотек и тестирование SQL-скрипта (сценария) базы данных, по результатам которых выявлено, что программное обеспечение цифровизации муниципальных услуг выполняет свои функции в соответствии с функциональными требованиями, ошибки в синтаксисе SQL-скрипта (сценария) базы данных отсутствуют.

Заключение

В результате выполнения работы проанализированы возможности автоматизации процесса библиотечного обслуживания, изучены механизмы работы библиотеки, ее структура, организация, особенности и бизнес-процессы.

Полученные следующие результаты работы.

В первой главе приведены кратная характеристика библиотечного учреждения, краткий обзор автоматизированных библиотечных информационных систем, рассмотрены основные понятия в области моделирования бизнес-процессов.

Для реализации задач практики изучена работа в ЦГБ МАУК «ЦГБС» и построены схемы библиотечного процесса «Как есть».

После этого проведен реинжиниринг библиотечных процессов с целью разработки программного обеспечения, проведено IDEF0 и UML моделирование, построена схема библиотечного процесса «Как должно быть».

Во второй главе определены функциональные требования к программному обеспечению, выбраны инструменты проектирования и разработки, проведено концептуальное, логическое, физическое проектирование базы данных программного обеспечения и реализована база данных.

В третье главе описана реализация функционала программного обеспечения, приведены схема взаимосвязи программных модулей программного обеспечения и реализованный сценарий базы данных, спроектирован и разработан интерфейс программного обеспечения.

Далее проведены функциональное тестирование разработанного программного обеспечения и тестирование SQL-скрипта (сценария) базы данных, по результатам которых выявлено, что программное обеспечение

цифровизации муниципальных услуг выполняет свои функции в соответствии с функциональными требованиями, ошибки в синтаксисе SQL-скрипта (сценария) базы данных отсутствуют.

Реализованное программное обеспечение цифровизации муниципальных услуг для общественных городских библиотек позволяет читателям бронировать информационные ресурсы, а библиотекарям – заполнять справочники по авторам, жанрам, издательствам, информационным ресурсам, видам информационных ресурсов, причинам списания информационных ресурсов, проводить выдачу, прием и, собственно, списание информационных ресурсов с указанием причин списания.

Данное программное обеспечение позволит уменьшить количество ошибок за счет автоматизации процесса библиотечного обслуживания при оказании муниципальных услуг, что приведет к снижению влияния человеческого фактора, сокращению трудозатрат библиотечных работников и затрат на бумагу, повышению качества обслуживания читателей, прозрачности и эффективности управления информационными ресурсами.

Современные цифровые информационные технологии в библиотечных учреждениях приобретают все бóльшую и бóльшую значимость, поскольку в процессе библиотечного обслуживания пользователи нуждаются в оперативном доступе к информации/информационным ресурсам. В общественных библиотеках идет активный процесс адаптации к новым реалиям, в том числе оснащение передовыми техническими средствами (принтерами, сканерами, копировальными устройствами), обеспечение доступа к сети Интернет и цифровизация библиотечных/ муниципальных услуг. Автоматизация процессов библиотечного обслуживания и перевод библиотечных услуг в цифровой вид играют значительную роль в ускорении процессов оказания муниципальных услуг.

Список используемой литературы и используемых источников

1. Агальцов В.П. Базы данных. – М.: Мир, 2002. – 376 с., ил.

2. Арчибальд Р.Д. Управление высокотехнологичными программами и проектами. – М.: ДМК Пресс, 2009. – 463 с.

3. Бабань Ю. Методология планирования бизнес-процессов. «Финансовая газета», 31.05.2001.

4. Бажин И.И. Информационные системы менеджмента. – М.: ГУ-ВШЭ, 2000. – 688 с.

5. Бальжинов А.В., Михеева Е.В. Анализ и диагностика финансово-хозяйственной деятельности предприятия: учеб. Пособие. – Улан-Удэ: Изд. ВСГТУ, 2003.

6. Барановская Т.П., В.И. Лойко, М.И. Семенов, А.И. Трубилин. Информационные системы и технологии в экономике. Учебник. 2-е изд., доп и перераб. М.: Финансы и статистика, 2005. – 416 с., ил.

7. Ватсон, К. C#. / К. Ватсон, М. Беллиназо, О. Корнс, Д. Эспиноза. – М.: Лори, 2005. – 863 с.

8. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: Учебное пособие / Л.Г. Гагарина. – М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. – 384 с.: ил.

9. Гвоздева В.А. Введение в специальность программиста: Учебник / В.А. Гвоздева. – 2-е изд., испр. и доп. – М.: ИД ФОРУМ: ИНФРА-М, 2011. – 208 с.: ил.

10. Голицына О.Л. Попов И.И. Программирование на языках высокого уровня: Учебное пособие / О.Л. Голицына, И.И. Попов. – М.: Форум, 2008. – 496 с.: ил.

11. Дубейковский, В.И. Эффективное моделирование с AllFusion Process Modeler (BPWin) 4.1.4 / В.И. Дубейковский. – М.: Диалог-МИФИ, 2007 – 384 с.

12. Емельянова Н.З. Партыка Т.Л. Попов И.И. Проектирование информационных систем: учеб. пособие / Н.З. Емельянова, Т.Л. Партыка, И.И. Попов. – М.: Форум, 2009. – 432 с.: ил.

13. Заботина Н.Н. Проектирование информационных систем: Учебное пособие / Н.Н. Заботина. – М.: ИНФРА-М, 2011. – 331 с.

14. Колдаев В.Д. Структуры и алгоритмы обработки данных: Учебное пособие / В.Д. Колдаев. – М.: ИЦ РИОР: НИЦ ИНФРА-М, 2014. – 296 с.

15. Кондратьев В.В., Кузнецов М.Н. Показываем бизнес-процессы. – М.: Эксмо, 2007. – 73 с.

16. Култыгин О. П. Администрирование баз данных. СУБД MS SQL Server: учеб. пособие / О. П. Култыгин. – М.: МФПА, 2012. – 232 с.

17. Маклаков С.В. ERwin и Bpwin. CASE-средства разработки информационных систем. – М.: Фолио ,1999. – 178 с.

18. Маклаков С.В. Создание информационных систем с ALLFusion Modelling Suite. М., 2003. – 315 с.

19. Мокеев В.В. Методология моделирования данных в среде ERWin: Учебное пособие для лабораторных работ. – Челябинск: Изд. ЮУрГУ, 2003. – 45 с.

20. Нортроп, Т. Основы разработки приложений на платформе Microsoft.NET Framework. Учебный курс Microsoft. Перевод с англ./ Т. Нортроп, Ш. Уилдермьюс, Б. Райан. – М.: «Русская редакция», 2007. – 864 с.

21. Репин В. В., Елиферов В. Г. Процессный подход к управлению. Моделирование бизнес-процессов. – М.: Манн, Иванов и Фербер, 2012. – 51 с.

22. Робинсон, С. C# для профессионалов. Том 2. / С. Робинсон, О. Корнес, Д. Глин, Б. Харвей. – М.: Лори, 2003. – 998 с.

23. Самуйлов К.Е. Серебренникова Н.В – Основы формальных методов описания бизнес-процессов. – М.: Фолио, 2014. – 511 с.

24. Титоренко Г.А. Информационные системы в экономике. Учебник для ВУЗов. – М.: Юнити-ДАНА, 2008. – 463 с.

25. Федоров И.Г. Системный подход к выявлению бизнес-процессов методом «сверху вниз» // Прикладная информатика. – 2012. № 5 (41), С.5-13.

26. Харрингтон Дж., Эселинг Э., Харм Ван Нимвеген. Оптимизация бизнес-процессов. – М.: «Бизнес-микро», 2002. – 311 с.

Приложение А

Формы интерфейса программного обеспечения цифровизации муниципальных услуг

Скриншоты программного обеспечения цифровизации муниципальных услуг представлены ниже.

Форма авторизации пользователей (библиотекаря и читателей) выглядит следующим образом.

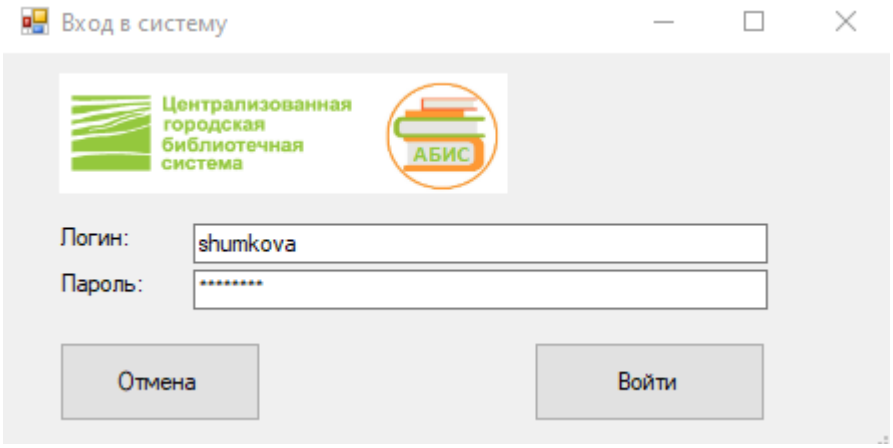


Рисунок А.1 – Форма авторизации

Главная форма программного обеспечения при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

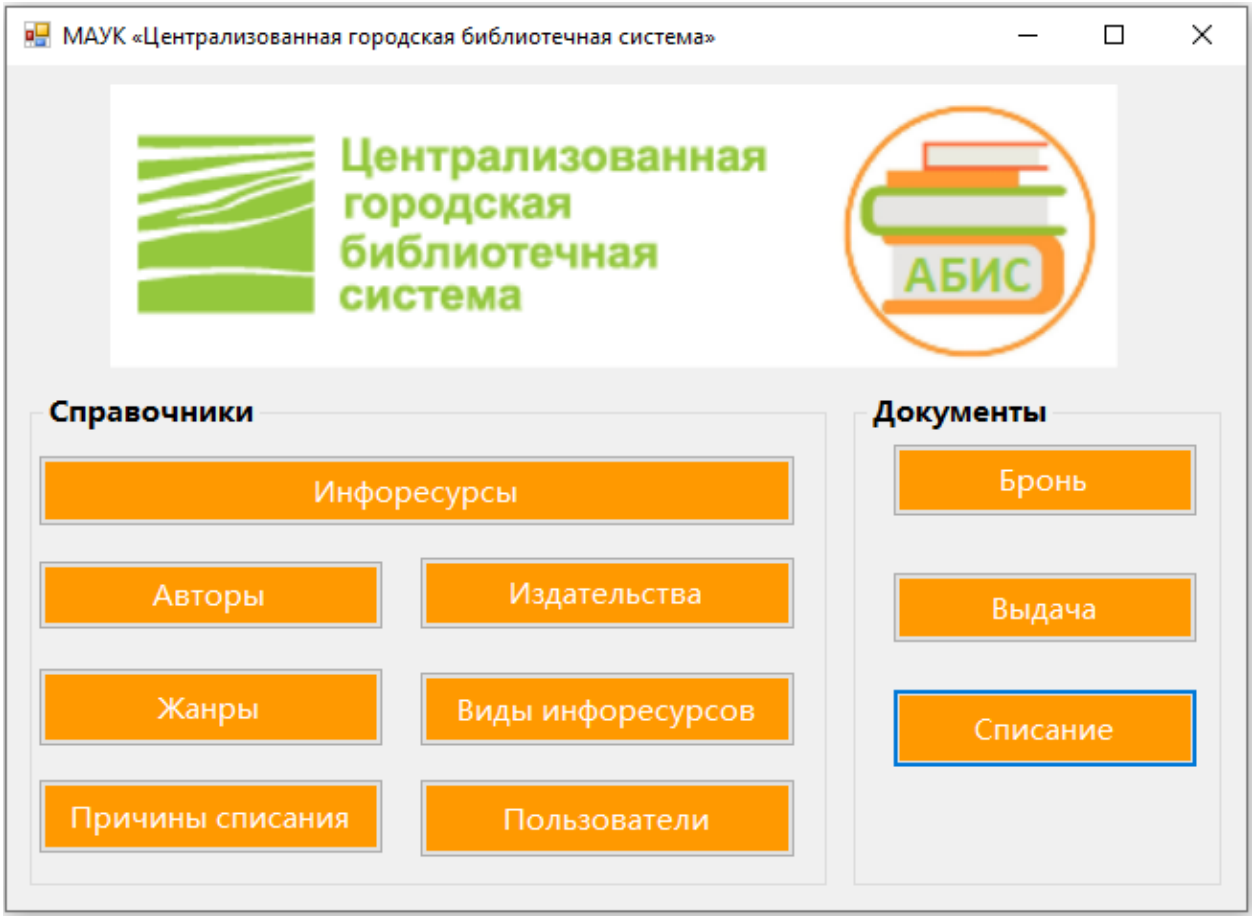


Рисунок А.2 – Главная форма при авторизации пользователя с ролью «Библиотекарь»

Главная форма программного обеспечения при авторизации пользователя с ролью «Читатель» выглядит следующим образом.

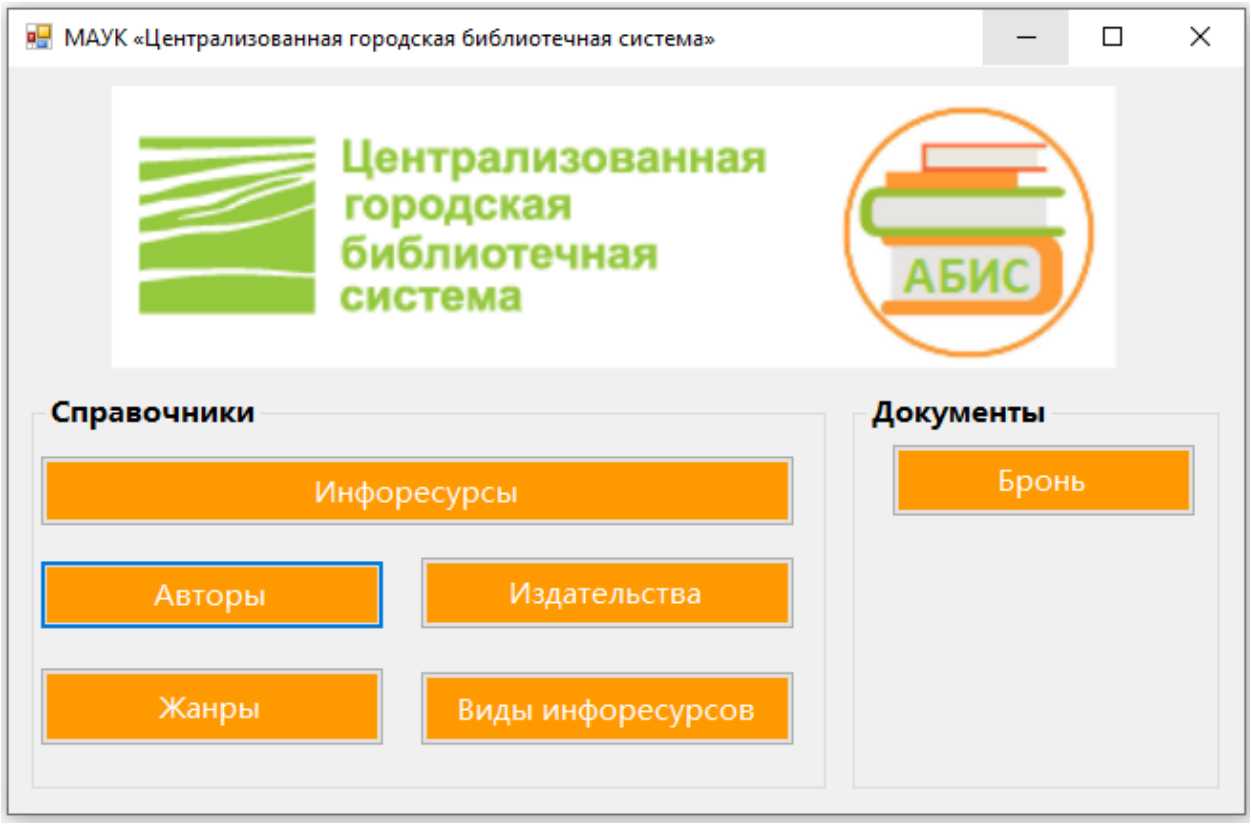


Рисунок А.3 – Главная форма при авторизации пользователя с ролью «Читатель»

Форма «Информационные ресурсы» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

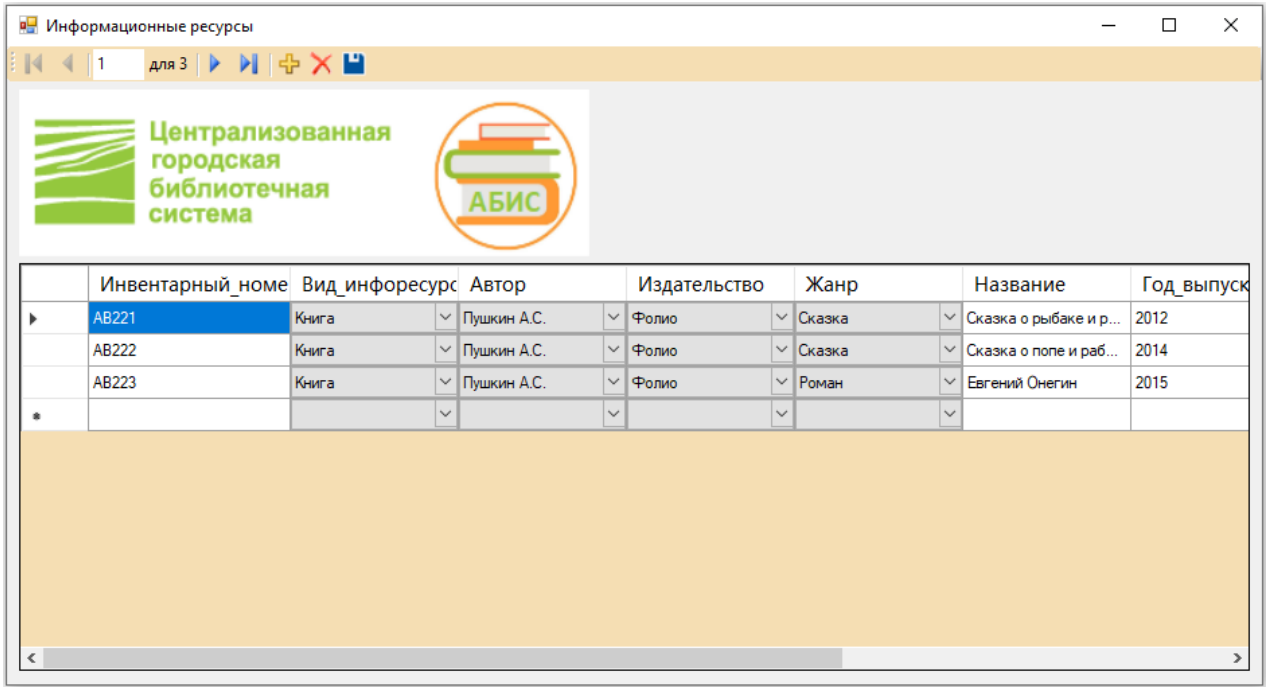


Рисунок А.4 – Форма «Инфоресурсы» при авторизации пользователя с ролью «Библиотекарь»

Форма «Информационные ресурсы» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

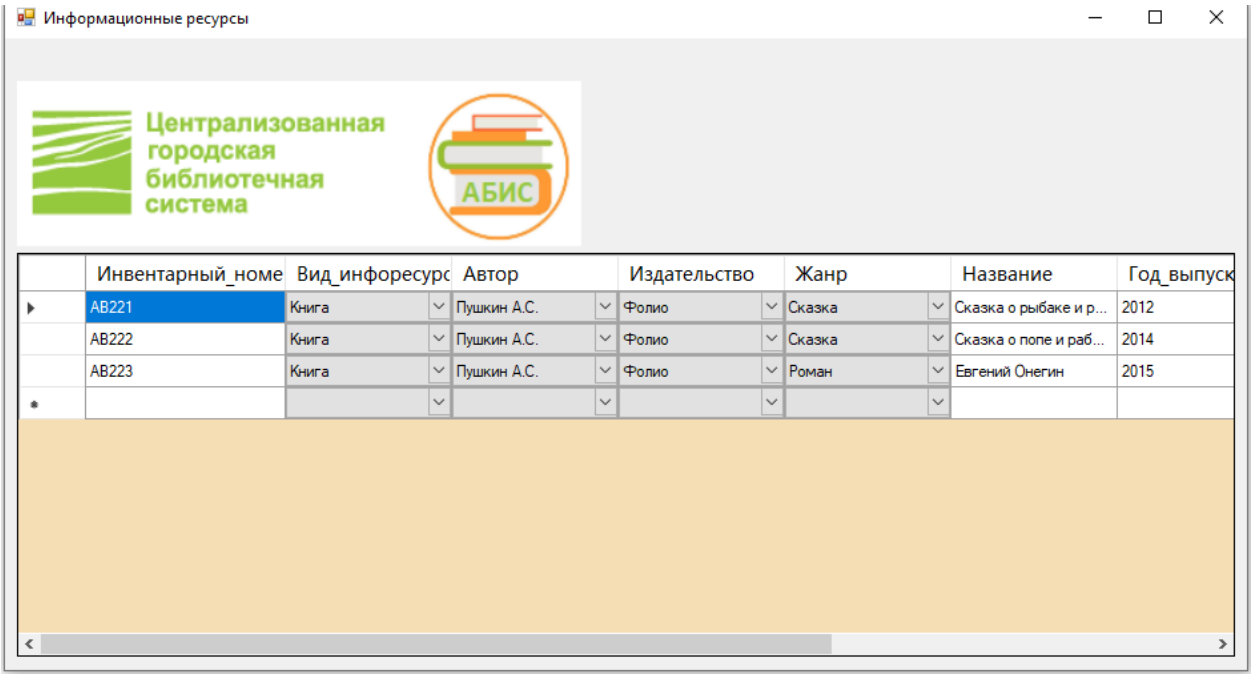


Рисунок А.5 – Форма «Инфоресурсы» при авторизации пользователя с ролью «Читатель»

Форма «Виды инфоресурсов» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

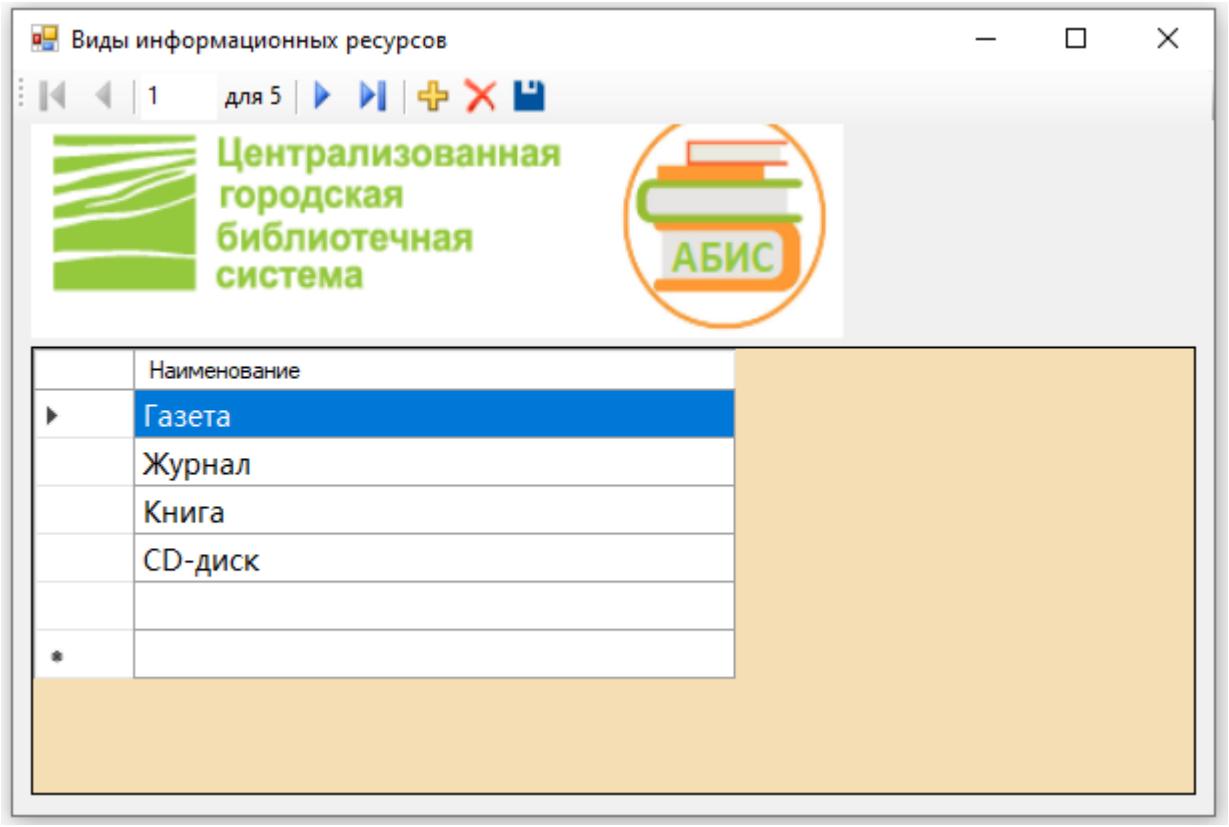


Рисунок А.6 – Справочник «Виды инфоресурсов» при авторизации пользователя с ролью «Библиотекарь»

Форма «Виды инфоресурсов» при авторизации пользователя с ролью «Читатель» выглядит следующим образом.

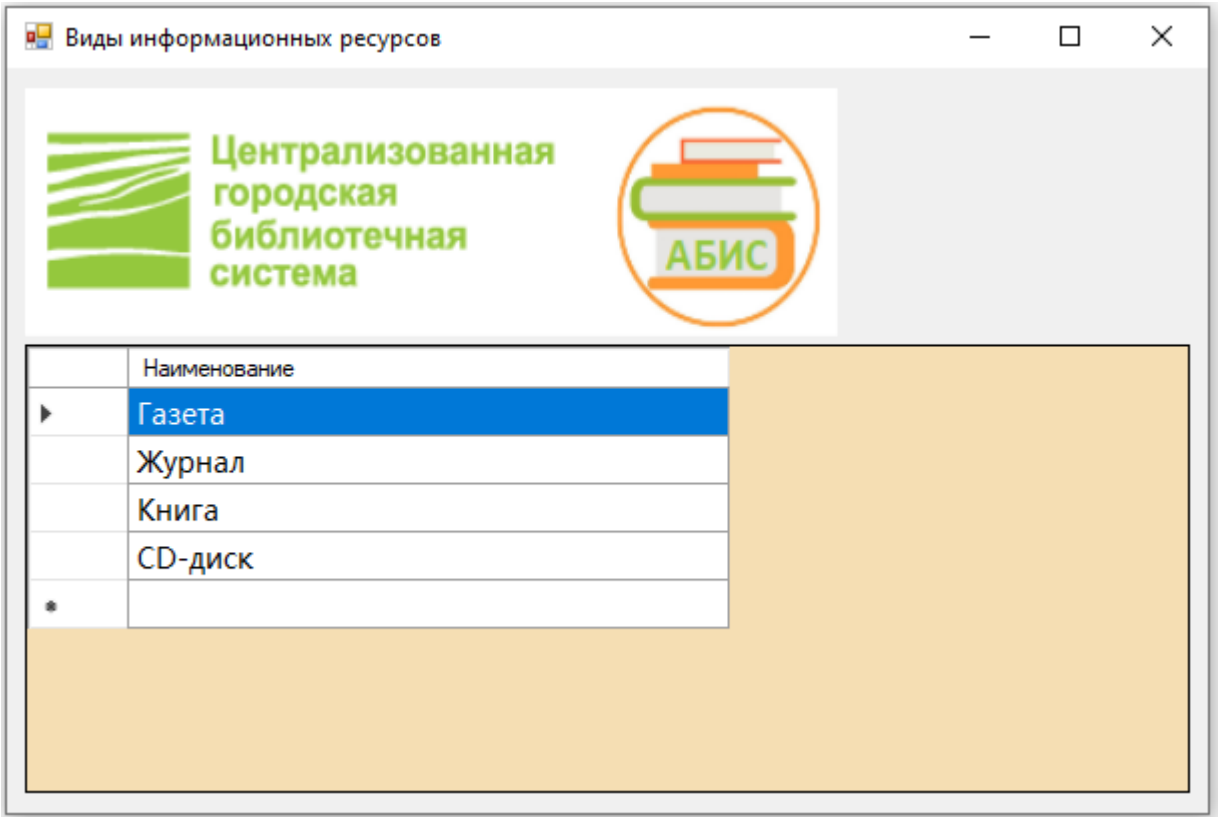


Рисунок А.7 – Справочник «Виды инфоресурсов» при авторизации пользователя с ролью «Читатель»

Форма «Авторы» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

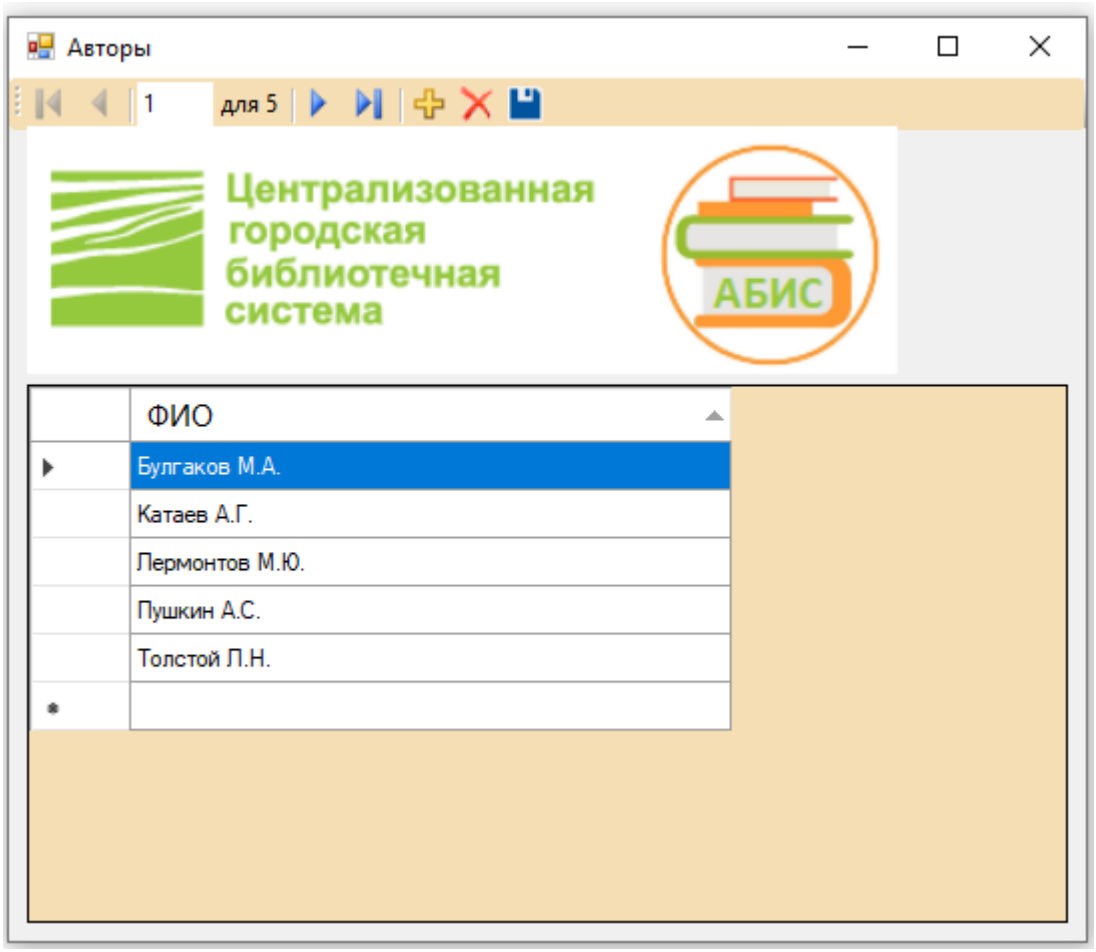


Рисунок А.8 – Форма «Авторы» при авторизации пользователя с ролью «Библиотекарь»

Форма «Авторы» при авторизации пользователя с ролью «Читатель» выглядит следующим образом.

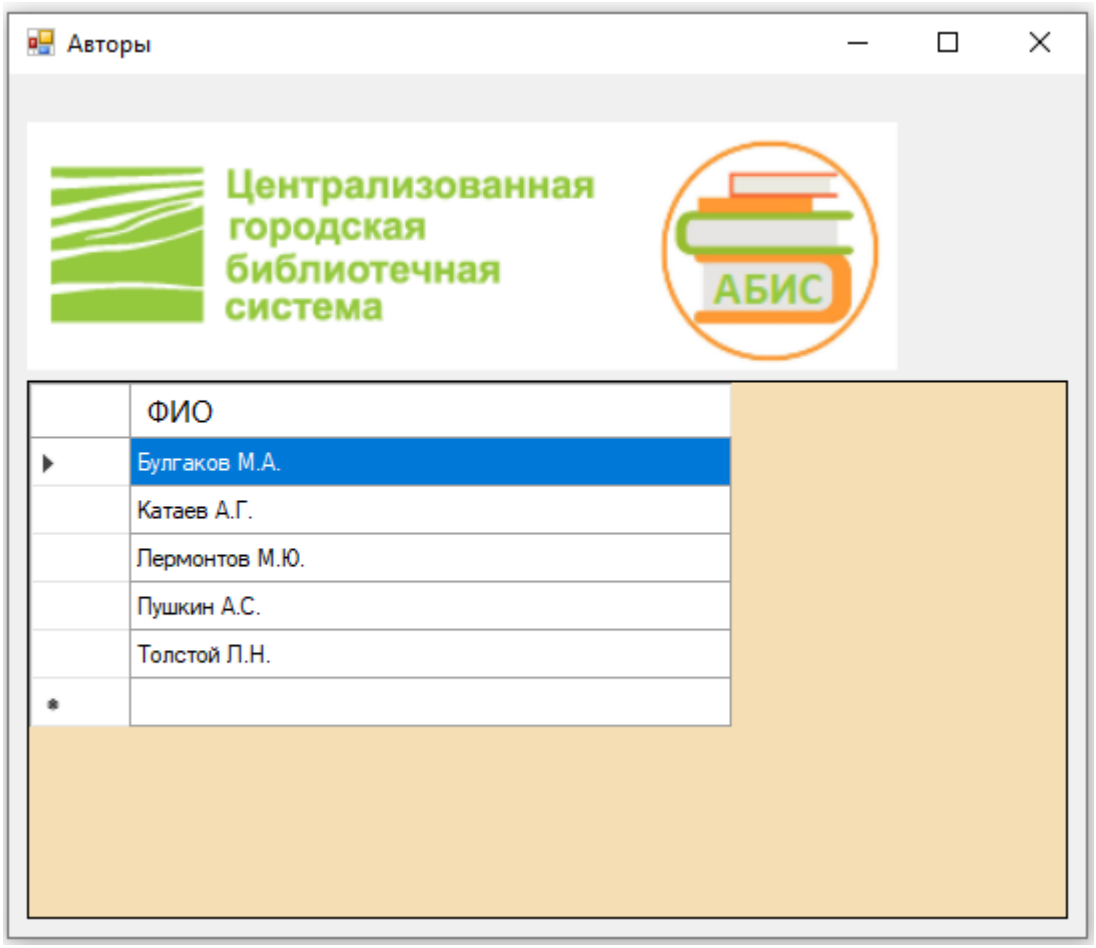


Рисунок А.9 – Форма «Авторы» при авторизации пользователя с ролью «Читатель»

Форма «Издательства» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

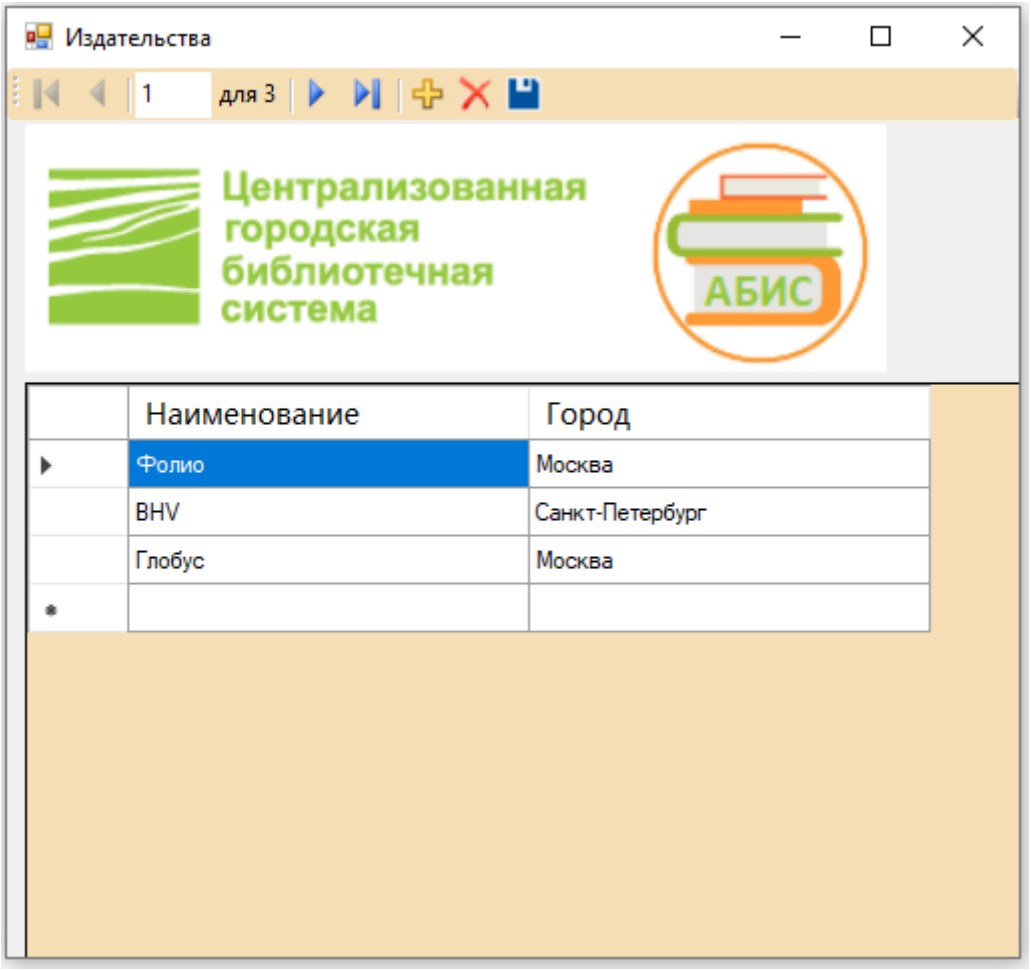


Рисунок А.10 – Форма «Издательства» при авторизации пользователя с ролью «Библиотекарь»

Форма «Издательства» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

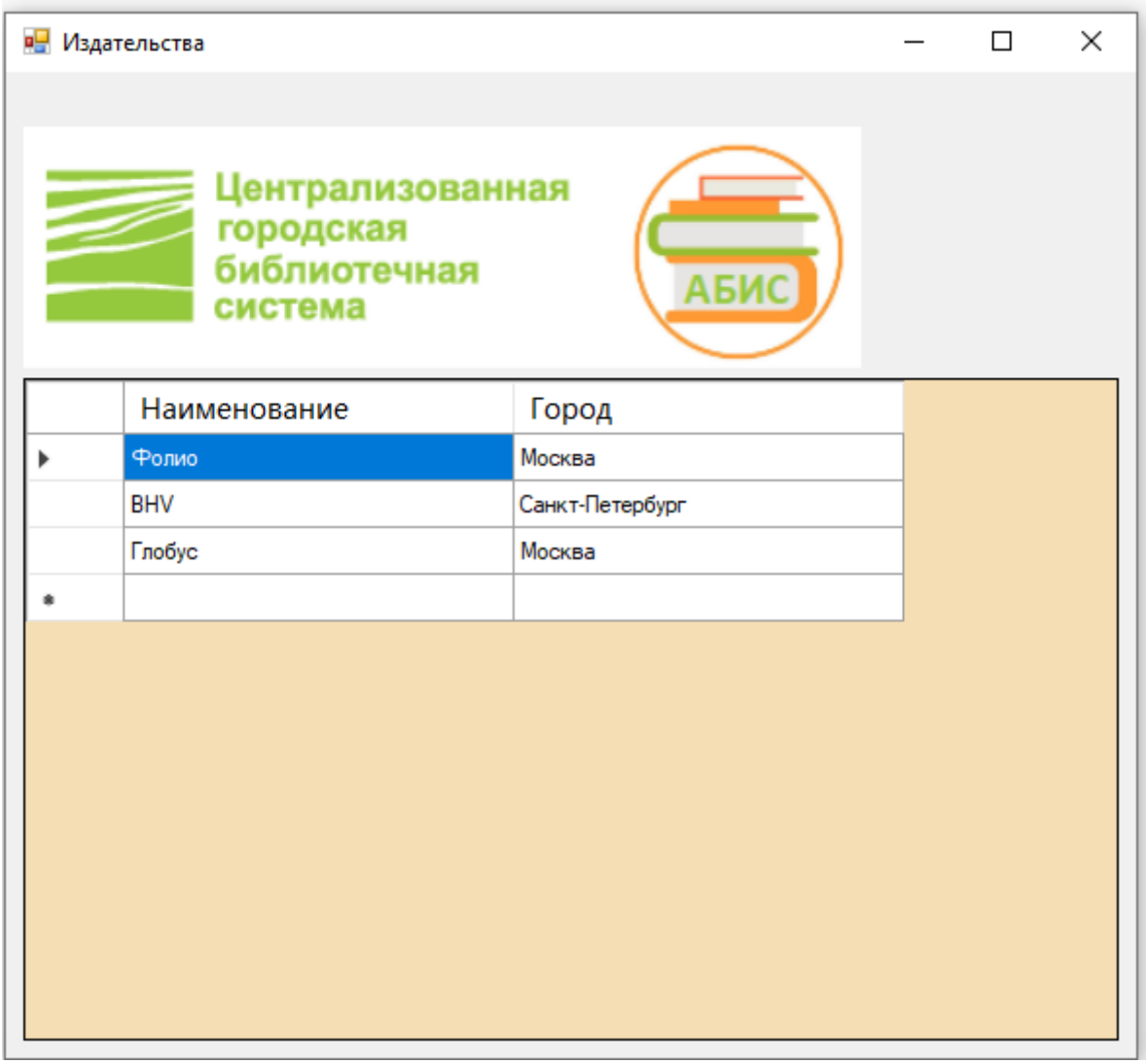


Рисунок А.11 – Форма «Издательства» при авторизации пользователя с ролью «Читатель»

Форма «Жанры» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

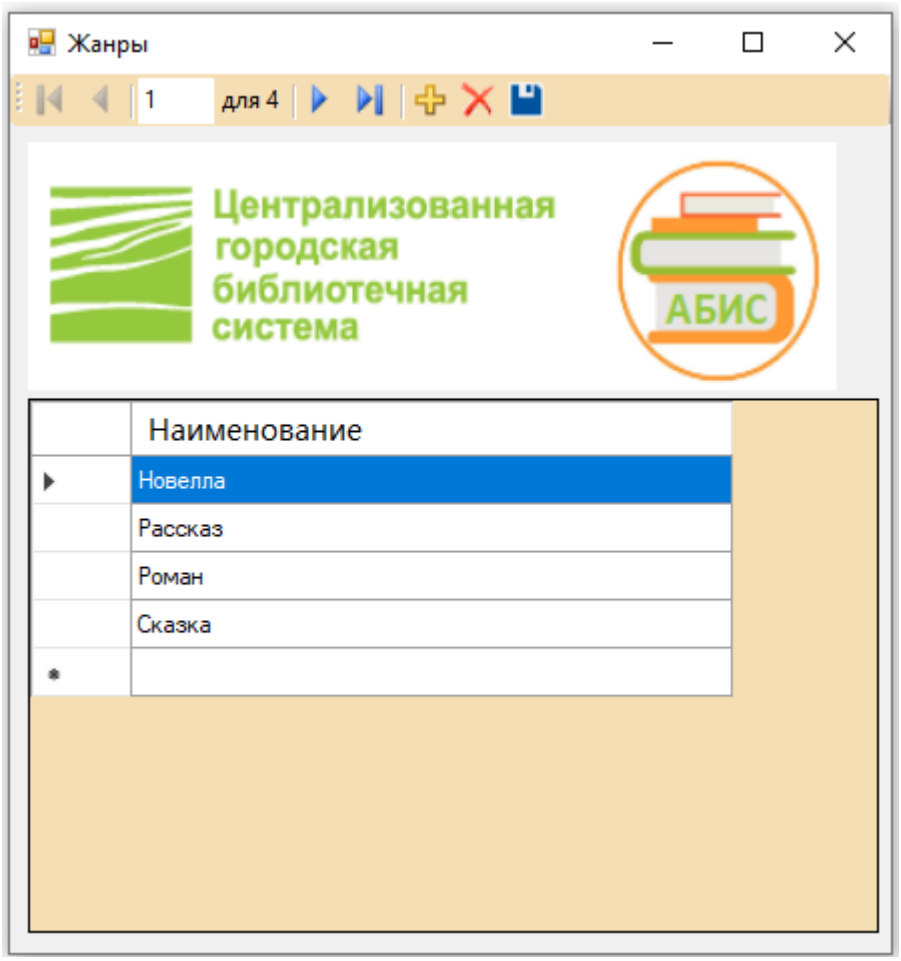


Рисунок А.12 – Форма «Жанры» при авторизации пользователя с ролью «Библиотекарь»

Форма «Жанры» при авторизации пользователя с ролью «Читатель» выглядит следующим образом.

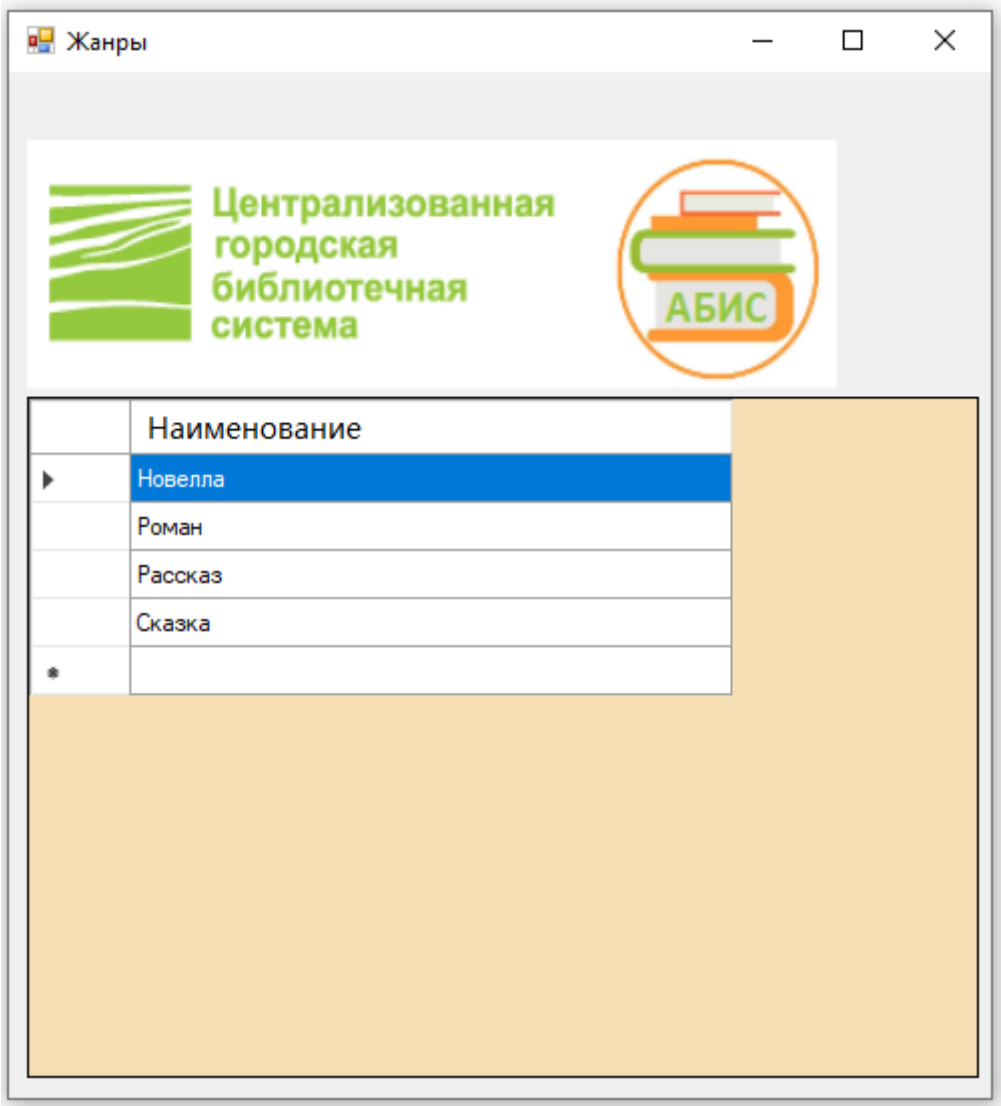


Рисунок А.13 – Форма «Жанры» при авторизации пользователя с ролью «Читатель»

Форма «Бронь» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

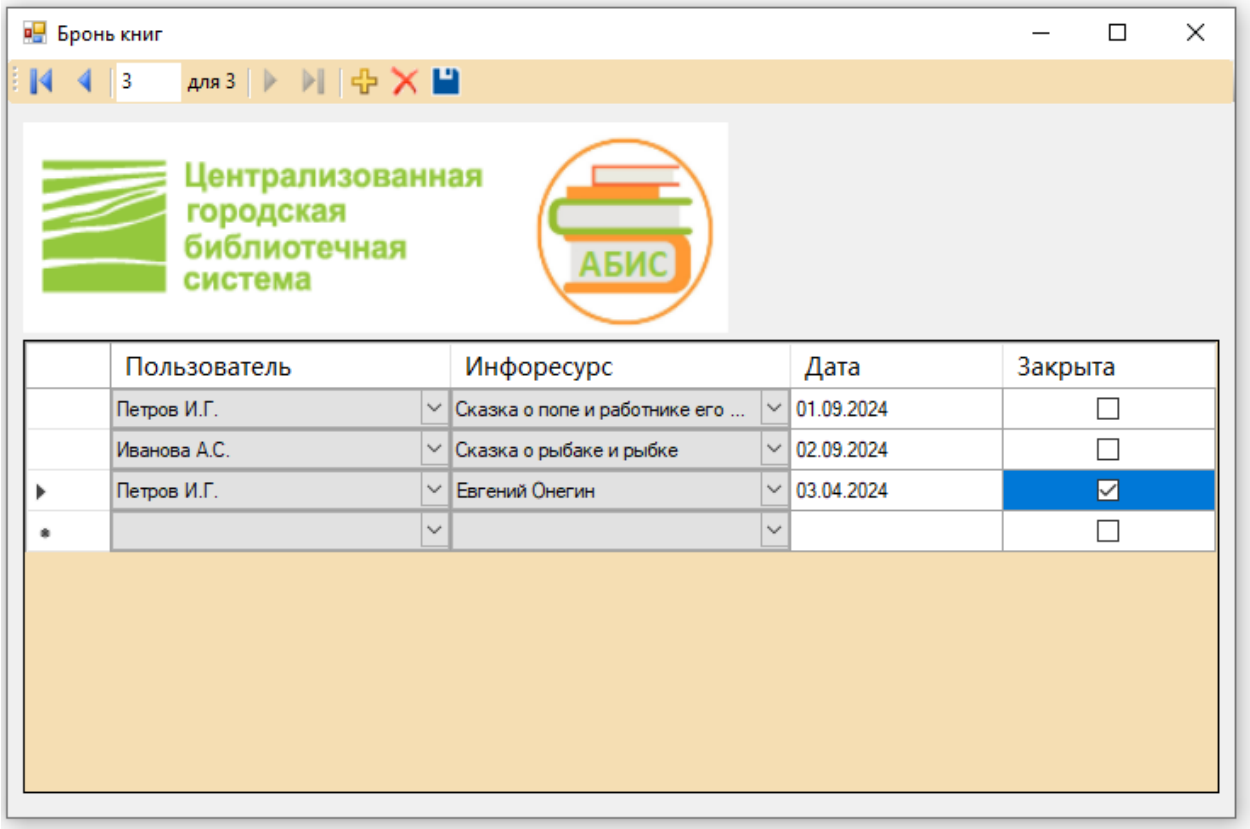


Рисунок А.14 – Форма «Бронь» при авторизации пользователя с ролью «Библиотекарь»

Форма «Бронь» при авторизации пользователя с ролью «Читатель» выглядит следующим образом.

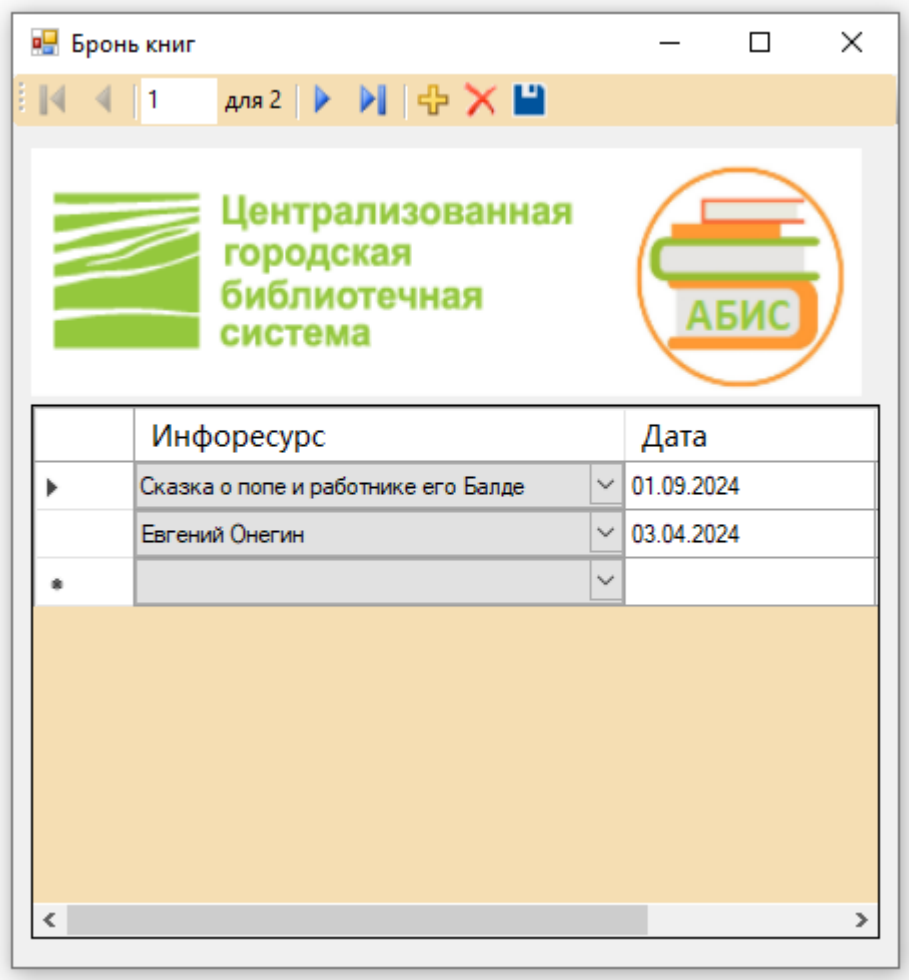


Рисунок А.15 – Форма «Бронь» при авторизации пользователя с ролью «Читатель»

Форма «Выдача» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

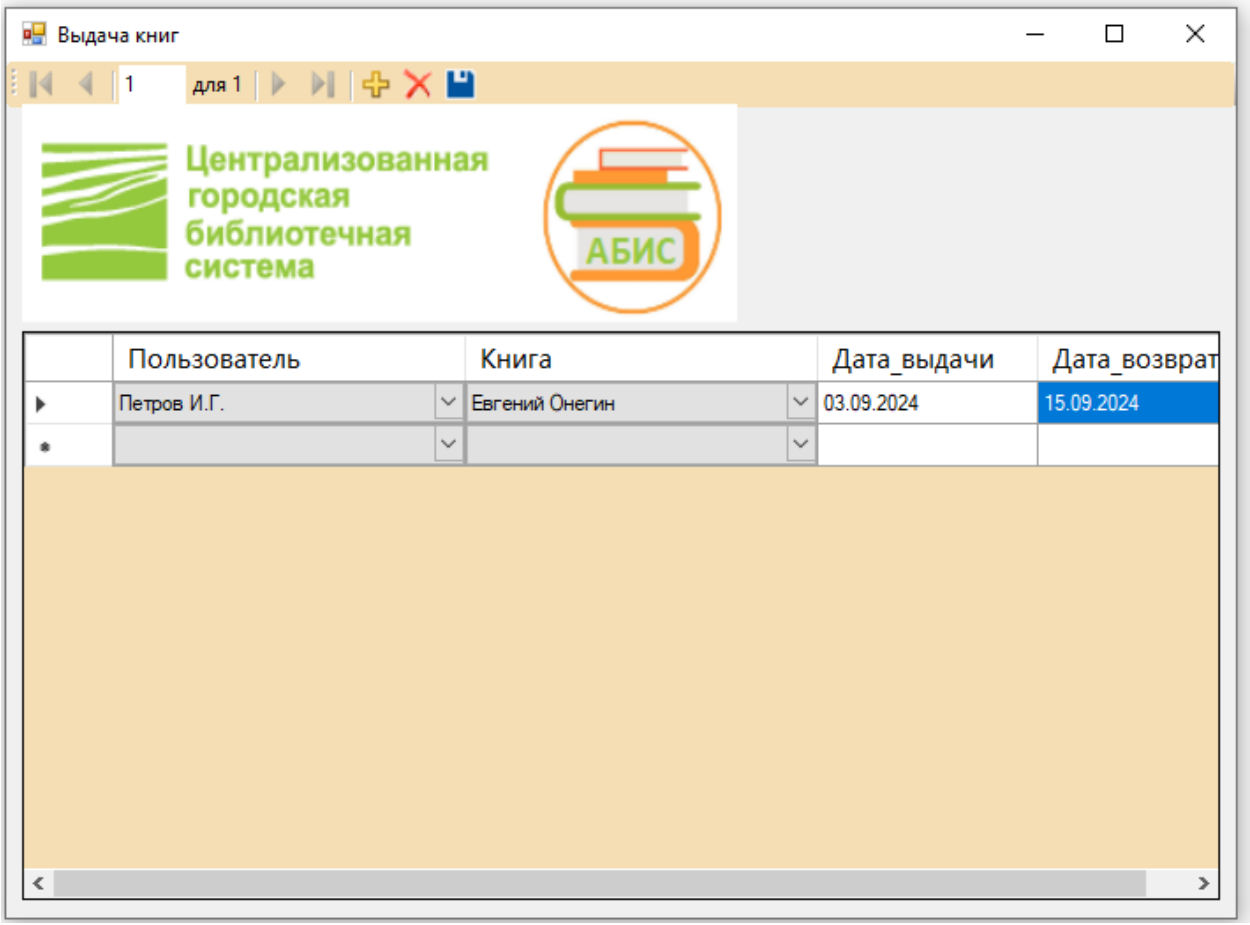


Рисунок А.16 – Форма «выдача» при авторизации пользователя с ролью «Библиотекарь»

Форма «Списания» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

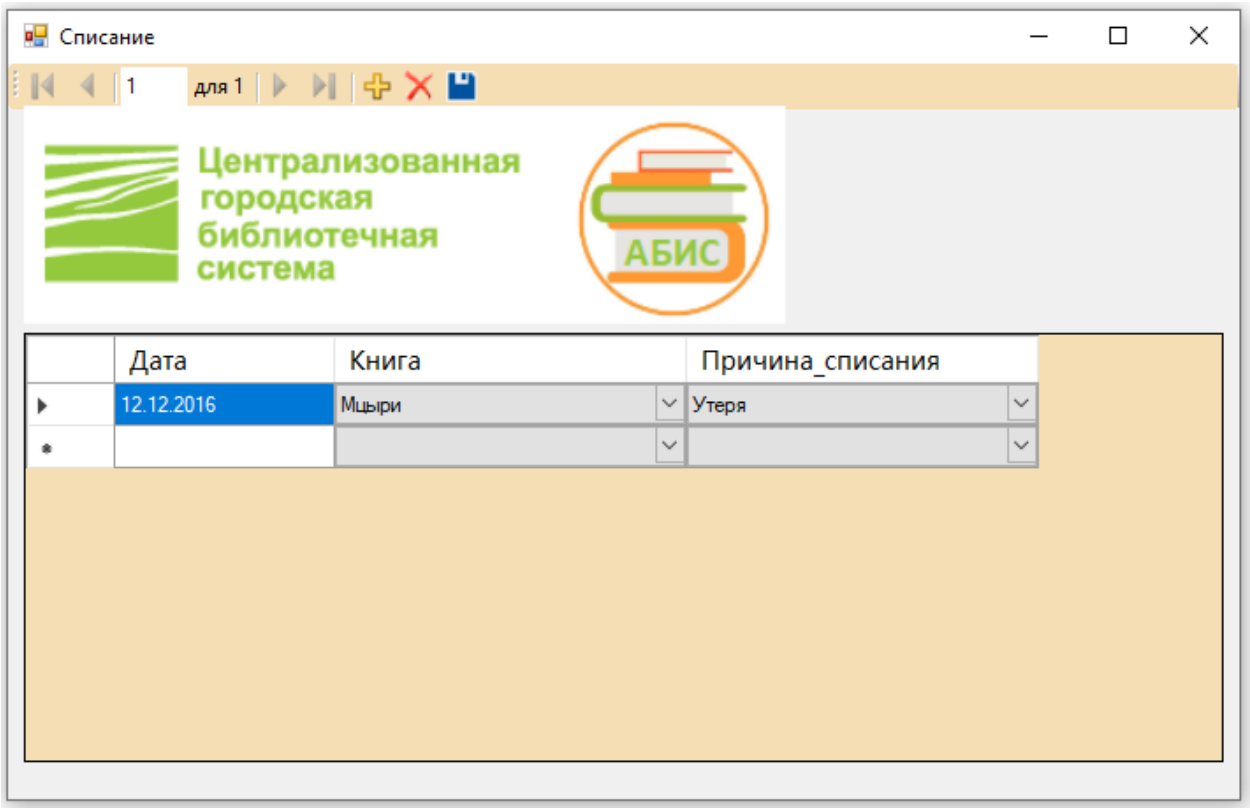


Рисунок А.17 – Форма «Списание» при авторизации пользователя с ролью «Библиотекарь»

Форма «Причины списания» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

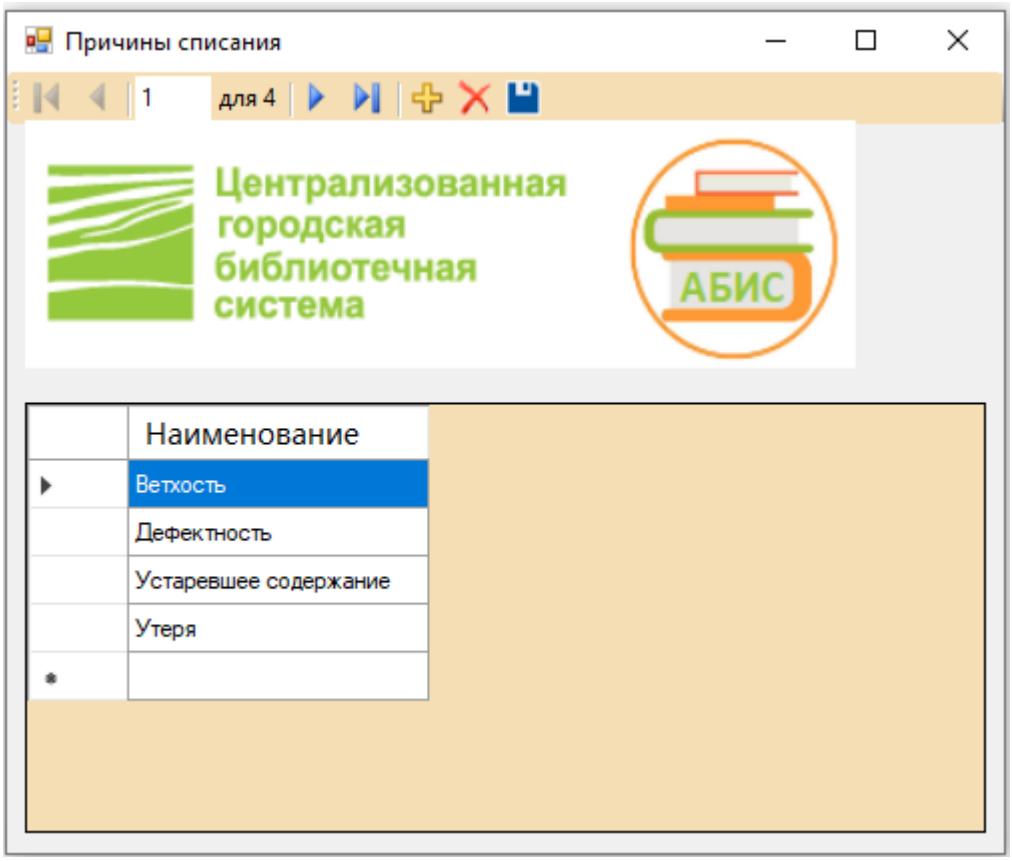


Рисунок А.18 – Форма «Причины списания» при авторизации пользователя с ролью «Библиотекарь»

Форма «Пользователи» при авторизации пользователя с ролью «Библиотекарь» выглядит следующим образом.

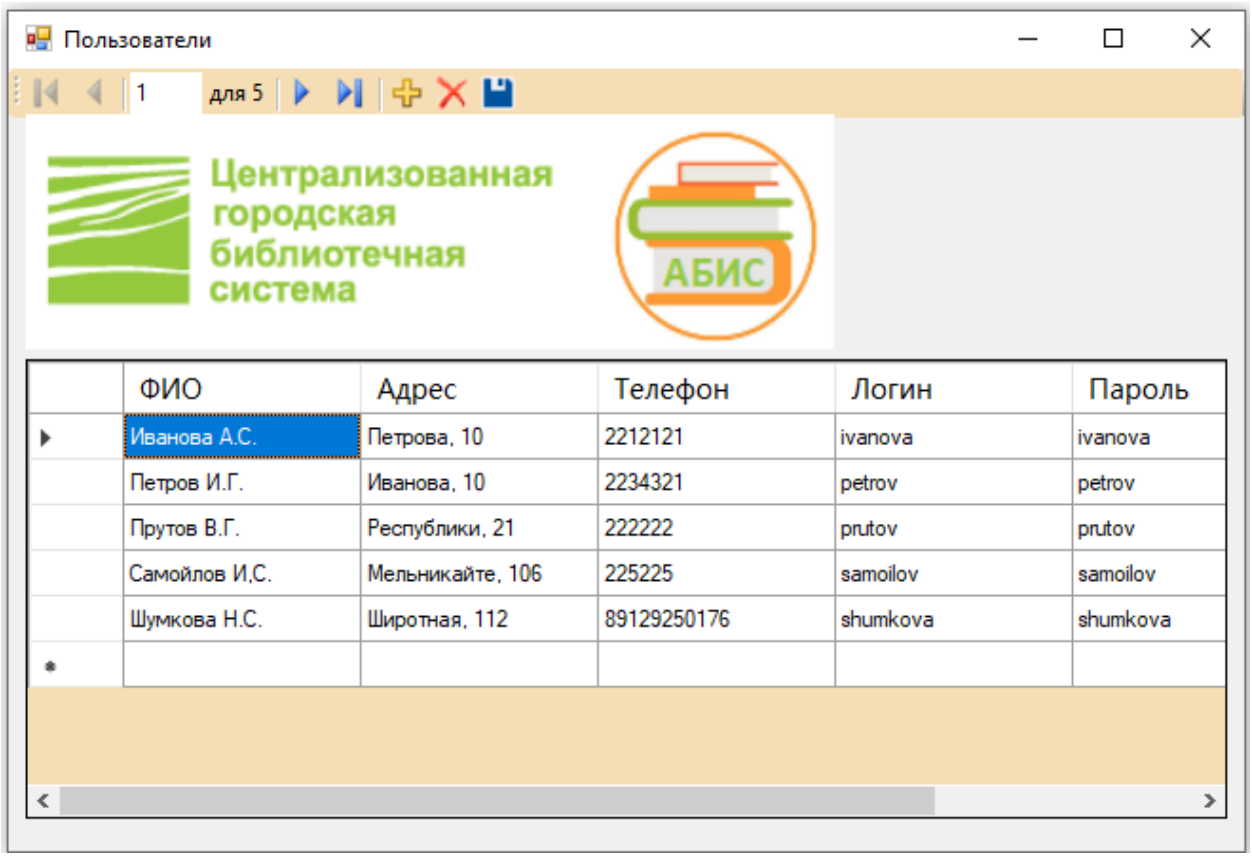


Рисунок А.19 – Форма «Пользователи»