

Министерство образования и молодежной политики
Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Специальность 09.02.07: «Информационные системы и программирование»

Разработка АРМ Работника библиотеки

Пояснительная записка

к курсовому проекту

КП-ПР-31-14-2023-ПЗ

Разработал:

Студент гр. ПР-31

_____/Д.А. Пинигина

Руководитель

_____/Д.О. Гариев

2023

Министерство образования и молодежной политики
Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Специальность 09.02.07: «Информационные системы и программирование»

Разработка АРМ работника библиотеки

Курсовой проект

КП-ПР-31-14-2023-ПЗ

ОГЛАВЛЕНИЕ

Введение.....	4
1 Постановка задачи.....	5
1.1 Описание предметной области	5
1.2 Требования к программному продукту	8
2 Проектирование логической структуры	14
3 Техническое проектирование	18
3.1 Выбор состава технических программных средств	18
3.2 Физическая структура программы	20
4 Тестирование	22
Заключение	24
Список использованной литературы.....	25
Приложение 1	28

Инф. № подл.	Подп. и дата	Взам. инф. №	Инф. № дубл.	Подп. и дата	<div>КП-ПР-31-14-2023-ПЗ</div> <div>Разработка АРМ работника библиотеки</div>	Лит	Лист	Листов
	Лит	Изм.	№ докум.	Подп.	Дата			
	Разраб.	Линигина Д.А.						
	Пров.	Гариев Д.О.						
	Н. контр.	Гариев Д.О.						
	Утв.							
						3		
						ГАПОУ СО «ЕКТС»		

ВВЕДЕНИЕ

Актуальность выбранной темы заключается в том, чтобы помочь библиотекарям с упрощением их деятельности.

Вечное добавление книг, регистрация читателя, особенно оформление выдачи. Всё это является тяжелым процессом, чтобы держать библиотеку в актуальном состоянии, следить за сроками выдач, удалять списанные книги, если они на данный момент уже не подходят и пылятся на полке. При работе возникает куча ошибок, из-за которых могут произойти неполадки и трудности, которые при этом нужно будет решить.

Исходя из этого, данная тема точно нуждается в разработке автоматизированного рабочего места.

Цель работы – разработка автоматизированного рабочего места работника библиотеки.

Основные задачи работы:

1. Провести анализ предметной области.
2. Выделить необходимые требования для создания программного продукта.
3. Спроектировать логическую структуру.
4. Провести техническое проектирование программного продукта.
5. Провести тестированию по созданному программному продукту.

Инв. № подл.	Подп. и дата	Инв. № дцбл.	Взам. инв. №	Подп. и дата						
Инв. № подл.	Подп. и дата	Инв. № дцбл.	Взам. инв. №	Подп. и дата	КП-ПР-31-14-2023-ПЗ					Лист
Ли	Изм.	№ докум.	Подп.	Дата						4

Инв. № подл.	Подп. и дата	Инв. № дудл.	Взам. инв. №	Подп. и дата

После изучения предметной области необходимо составить необходимый перечень требований и задач, которые обязано выполнять программное обеспечение на выходе.

Библиотека – организация, которая в свои обязанности включает сбор и хранение всех возможных рукописных трудов культовых авторов, а также их выдачу людям для их ознакомления и прочтения. Данное учреждение называют «храмом знаний».

Для полноценной деятельности важно не забывать о читателях. Они в свою очередь порождают спрос на посещение библиотек и чтение книг. Для каждого такого гостя предоставляется свой читательский билет, который содержит имя, фамилию, отчество читателя, а также его контактный номер, адрес электронной почты и данные паспорта. К каждому читателю необходимо относиться одинаково,

уважая его выбор и интересы, а также предоставляя ему необходимую помощь в выборе книг. Если же посетитель не проявляет какой-либо интерес, то необходимо приложить все усилия, чтобы он смог определиться и выбрать заинтересовавшую его книгу.

Для более упрощенного поиска определенной книги, каждое произведение направляют в свой отдел, исходя от его жанра. Каждый том должен знать своё место на полках библиотеки.

При сотрудничестве с поставщиками литературы важно иметь при себе все возможные связи. От них зависит, появятся ли новые книги на полках библиотеки или же нет. Особенно необходимо держать связи с поставщиками через электронную почту, ведь если необходимо срочно связаться с ними, то никаких проблем с планами не произойдет.

Выдача книг читателю является довольно простой работой. Важно лишь указывать, что та или иная книга в руках у того, кто её взял на время. Без необходимой информации о читателе произведение будет считаться потерянным, что может привести к различным негативным последствиям, которые в свою очередь могут сильно ударить по репутации организации.

Среди всех возможностей пользования библиотекой, необходимо знать и об ограничениях, которые важны для правильной работы организации. Срок выдачи книг составляет месяц, который читатель вправе продлить два раза. В случае, если произведение вернули позже срока, на того, кто брал книгу будет заведён штраф. При трёх задержках читателю больше не будут предоставлять книги. Это также касается отказа об уплате штрафа.

Важным человеком в библиотеке является сам библиотекарь. Его основные задачи – ведение учета книг и читателей. Под этими двумя заданиями хранятся куда более серьёзные вещи. Ими являются следующее:

1. Принимать книги от поставщиков, добавлять их в базу данных, присваивая им свой идентификационный код и направлять в нужный отдел. Для упрощения в дальнейшей работе с помощью читателей, важно соблюдать необходимые

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата						Листм	
Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата	Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-31-14-2023-ПЗ	6

действия, ведь такой ответственный шаг поможет библиотекарю в поисках нужного произведения.

2. Иметь доступ к информации о книгах. Книги – важная составляющая библиотеки, поэтому за ними нужно обязательно следить.

3. Узнавать, какие книги, на данный момент, находятся в библиотеке. Чтобы не происходило неприятных ситуаций между библиотекарем и читателем, необходимо прослеживать точную и недавнюю информацию о тех или иных произведениях. Если уж и произошла подобная ситуация, то библиотекарю в срочном порядке предоставить альтернативы той книги, которую так требовал читатель.

4. Списывать книги, если осуществляется переиздание или книга устарела с годами. Без данного действия у библиотекаря может произойти негативные последствия, связанные с нехваткой места на полках и отделах, из-за чего новым книгам не будет места несмотря на то, что старые издания уже никто не спрашивает и не берет.

5. Указывать возврат, если читатель вернул книгу. Это очень необходимо для избежания ложных штрафов и возможного увольнения библиотекаря.

6. Искать выбранную книгу, исходя из запроса читателя. Библиотекарь обязан помогать своим клиентам, так как от доброжелательного и своевременного обслуживания зависит репутация библиотеки.

7. Давать читателю возможность воспользоваться библиотекой.

8. Наблюдать за штрафами, если срок сдачи книги у читателя истёк. Данная мера является административной, поэтому вынуждена взимать определенную сумму в целях компенсации несоблюдения библиотечных правил.

9. Оформлять выдачу на основе выбранной книги.

Работая в библиотеке, всё должно соответствовать Федеральному закону "О библиотечном деле" от 29.12.1994 N 78-ФЗ.

Администрация библиотеки обязана контролировать дисциплину, правила и нормы труда, следить за правомерным обслуживанием посетителей, устранять все возможные вопросы, касаемые проблем данной организации.

Инф. № подл.	Подп. и дата	Инф. № дцбл.	Взам. инф. №	Подп. и дата
Ли	Изм.	№ докум.	Подп.	Дата
КП-ПР-31-14-2023-ПЗ				Листм
				7

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

производительности, надежности и безопасности, обработки ошибок, интерфейсные, ограничения.

Производительность:

1. Программа должна выводить данные о выбранной книге за 1 секунду.
2. Программа должна автоматически обновлять данные о книгах за 1 секунду.
3. Программа должна автоматически обновлять данные о читателях за 1 секунду.
4. Программа должна заносить данные о забронированной книге за 1 секунду.
5. Программа должна удалять данные о читателях за 1 секунду.
6. Программа должна удалять данные о книгах за 1 секунду.
7. Программа должна находить нужную книгу за 1 секунду.
8. Программа должна автоматически обновлять количество штрафов за 1 секунду.
9. Программа должна занимать меньше 100 МБ памяти.

Надежность и безопасность:

1. Все данные книг должны быть доступны только для прочтения, если не произошла авторизация.
2. Данные выдач и читателей не доступны без авторизации.
3. Изменение и добавление данных недоступно без авторизации.
4. Данные книг доступны в открытом доступе, если работник библиотеки был авторизован.
5. Данные читателей доступны в открытом доступе, если работник библиотеки был авторизован.
6. Данные выдач доступны в открытом доступе, если работник библиотеки был авторизован.

Обработки ошибок:

1. Если не получается авторизоваться, то программа должна вывести сообщение о неудачной авторизации.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.
Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-31-14-2023-ПЗ
					Лист 10

2. Если не получилось ввести новую книгу, то программа должна вывести сообщение о неправильно введенных данных.

3. Если не получилось ввести нового читателя, то программа должна вывести сообщение о неправильно введенных данных.

4. Если не получилось ввести новую выдачу, то программа должна вывести сообщение о неправильно выбранных данных.

5. Если не получилось изменить читателя, то программа должна вывести сообщение о неправильно введенных данных.

6. Если не получилось изменить книгу, то программа должна вывести сообщение о неправильно введенных данных.

7. Если не получилось изменить выдачу, то программа должна вывести сообщение о неправильно введенных данных.

8. Если не получилось удалить читателя, то программа должна вывести сообщение об ошибке удаления.

9. Если не получилось удалить книгу, то программа должна вывести сообщение об ошибке удаления.

10. Если не получилось удалить выдачу, то программа должна вывести сообщение об ошибке удаления.

Интерфейсные:

1. В окне авторизации должно быть текстовое поле и поле для ввода пароля для авторизации пользователя.

2. В окне авторизации должен быть флажок «Войти как гость».

3. В окне авторизации должен происходить выбор по нажатию флажка, после которого ввод данных будет недоступной.

4. На главной странице в основном блоке должен располагаться лист со всеми книгами.

5. На главной странице слева должны располагаться навигационный кнопки.

6. Для каждой книги в листе указывается название, автор, жанр, год издания, количество, ссылка на изображение.

Инф. № подл.	Подп. и дата
Инф. № докл.	Взам. инв. №
Подп. и дата	Инф. № подл.
Инф. № подл.	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата
----	------	----------	-------	------

КП-ПР-31-14-2023-ПЗ

7. К каждой книге из листа должна быть кнопка «Оформить выдачу».

8. К каждой книге из листа должны быть расположены кнопки изменения и удаления в правом верхнем и нижнем углу.

9. В окне для выдачи у каждой выдачи указывается название книги, автор, жанр, год издания, кому отдана, когда и до какой даты нужно вернуть.

10. Если срок подошёл к концу, то выдачи горят красным и высвечивается поле со штрафом.

11. В окне для выдачи для каждой выдач должна быть кнопка для принятия книг и изменения данных.

12. В окне читателей указывается ФИО, почта и телефон.

13. В окне читателей для каждого читателя из списка есть кнопки изменения и удаления.

14. В окне оформления выдачи по порядку расположен список с читателями, список с книгами, поле для ввода даты выдачи, поле для даты сдачи.

15. В окне добавления книг располагается поля для ввода ISBN, названия, выпадающий список с авторами, поле для ввода года издания, списки с жанрами и издательствами, поля с количеством, ценой и ссылкой на изображение.

16. В окне добавления читателя располагается поля для ввода ФИО, электронной почты и телефона.

17. Внизу каждого окна для добавлений и изменений данных должны быть кнопки сохранения и отмены.

Ограничения:

1. Точность – максимальная.
2. Все окна должны придерживаться одного стиля.
3. Используются стандарты ГОСТ.

Существуют обратные требования, которые не позволяют выполнять определенные действия.

Обратные требования:

1. Программа не должна оформлять выдачу книги, если количество таких книг равна 0.

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Инв. № подл.
Инв. № подл.	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-31-14-2023-ПЗ

2. Программа не должна допускать доступ неавторизованным пользователям к изменению и обработке данных.

Инв. № подл.	Подп. и дата	Инв. № дцбл.	Взам. инв. №	Подп. и дата	КП-ПР-31-14-2023-ПЗ	Лист
						13
Инв. № подл.	Подп. и дата	Инв. № дцбл.	Взам. инв. №	Подп. и дата		
Ли	Изм.	№ докум.	Подп.	Дата		

2 ПРОЕКТИРОВАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

ER-диаграмма позволяет понять, как связаны сущности в базе данных между друг-другом и их элементы.

Сущности: книга, поставщик, жанр, читатель, выдача.

Используется связь «Один ко многим» и «Многие ко многим».

Книга содержит:

- ISBN (Международный стандартный книжный номер);
- название;
- автор;
- поставщик;
- жанр;
- количество;
- цена;
- год издания;
- изображение.

Жанр содержит только его название.

Поставщик содержит имя и адрес электронной почты.

Читатель содержит:

- имя;
- фамилия;
- отчество;
- адрес электронной почты;
- телефон.

Связующая таблица выдачи содержит:

- книга;
- читатель;
- дата выдачи;

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	КП-ПР-31-14-2023-ПЗ	Лист
						14
Ли	Изм.	№ докум.	Подп.	Дата		

- дата сдачи.

Для полного понятия логики базы данных составляется следующая таблица (Рисунок 1):

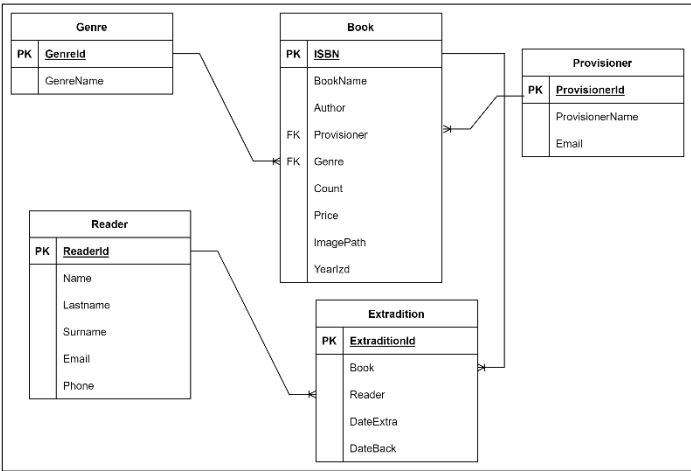


Рисунок 1 — ER-диаграмма

Use-case диаграмма позволяет описать систему на концептуальном уровне. В её основу входят актёры и прецеденты. Такая модель показывает то, как именно связаны актёры и прецеденты. Из предметной области можно сделать вывод, что актёрами будут: читатель, поставщик и библиотекарь. Все они играют огромную роль в работе библиотеки (Рисунок 2).

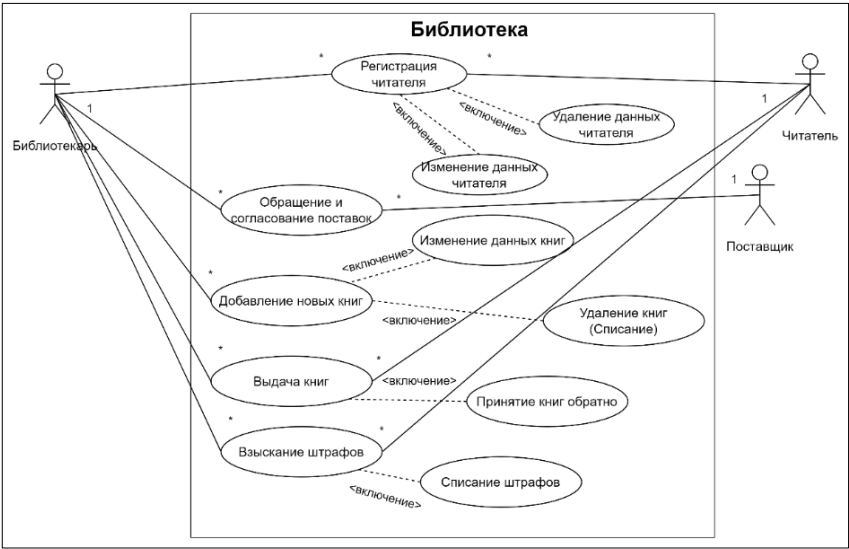


Рисунок 2 — Use-case диаграмма

Statechar диаграмма показывает переход между двумя состояниями. Используя данную предметную область, можно показать, как книга на выдачу становится выданной книгой (Рисунок 3).

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	Инв. № дубл.
Инв. № подл.	Взам. инв. №
Инв. № подл.	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

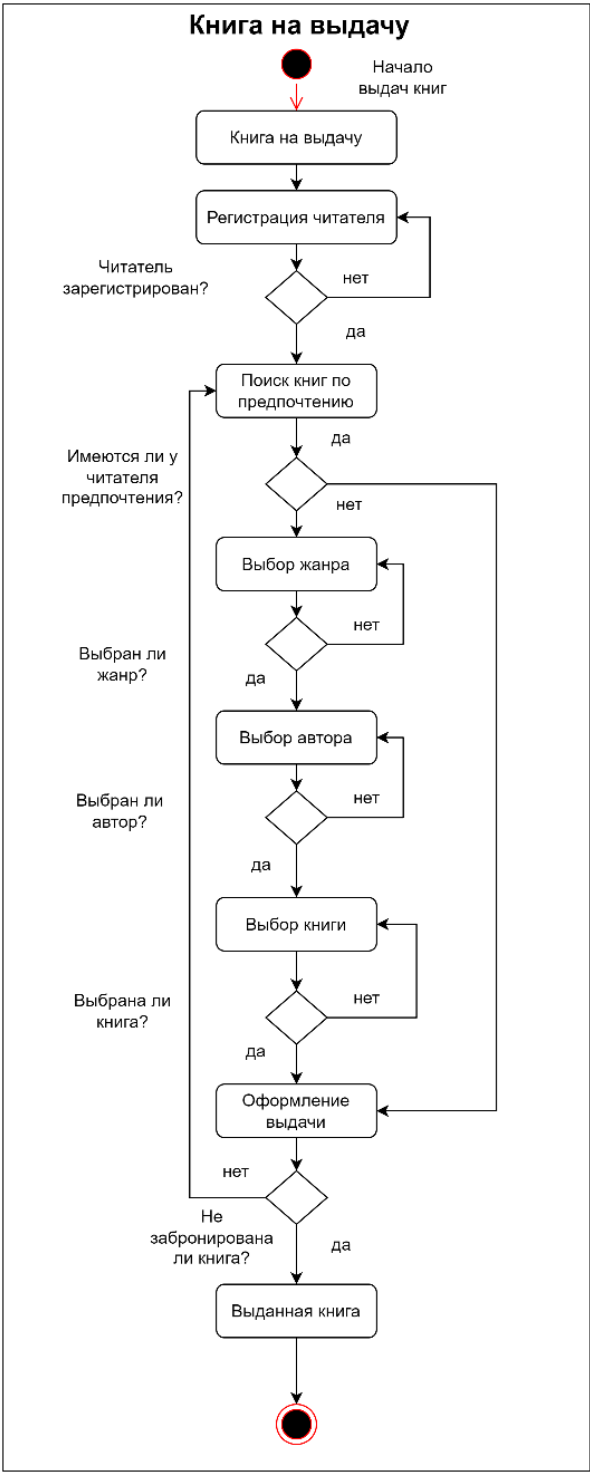


Рисунок 3 — Statechar диаграмма

Activity диаграмма отражает динамические аспекты поведения системы. Благодаря ей можно понять, как поток управления переходит от одной деятельности к другой. Такая диаграмма разбивается на несколько частей, чтобы было понятнее кому какая достанется деятельность. Каждая роль отвечает за свои действия (Рисунок 4).

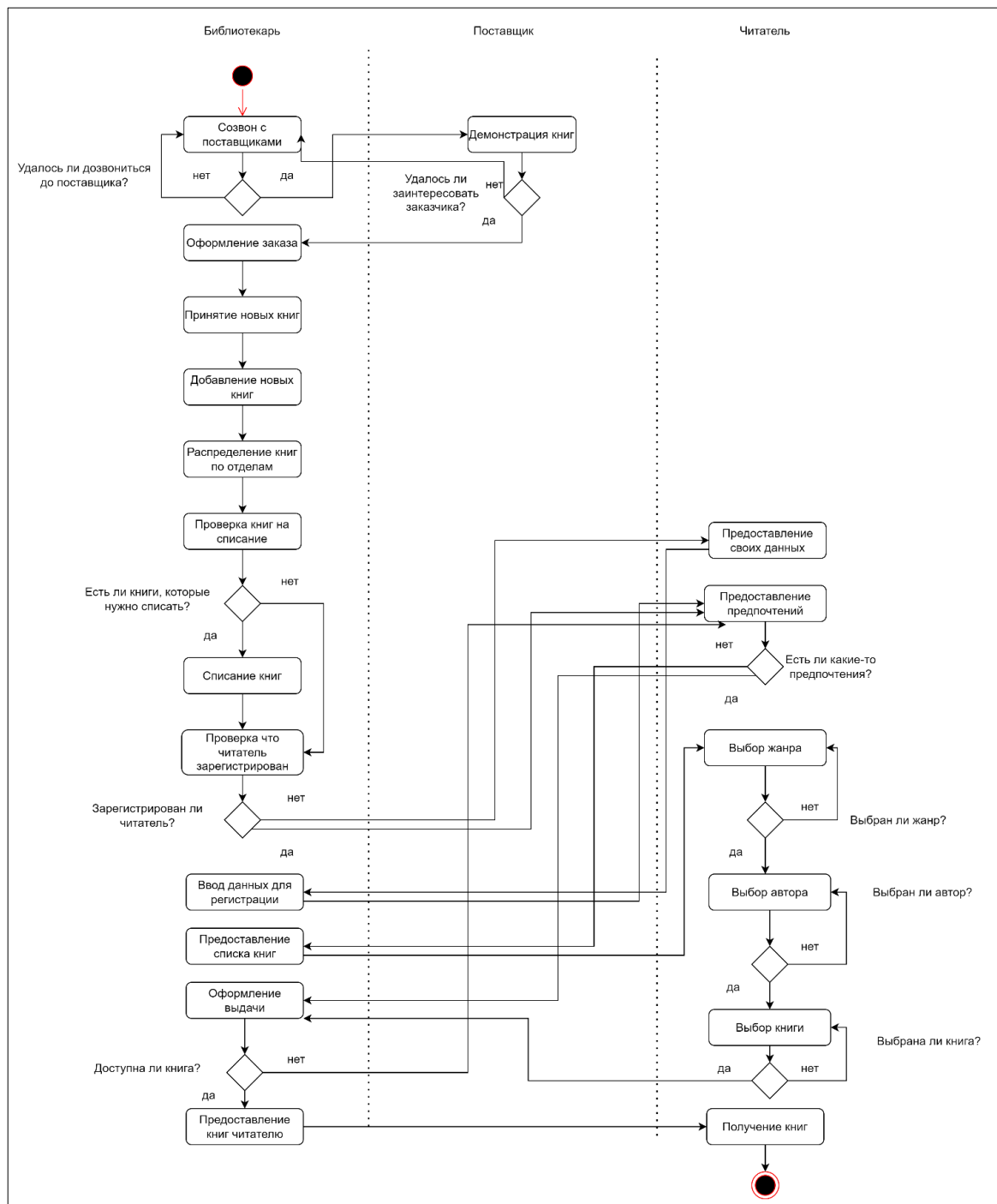


Рисунок 4 — Activity диаграмма

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	Инв. № дубл.

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-31-14-2023-ПЗ

3 ТЕХНИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

3.1 Выбор состава технических программных средств

Программно-технические средства – совокупность определенных программных и технических средств автоматизации, которые предназначены для создания необходимых проектов и информационных систем.

За основу создания необходимой программы используются технические средства, среди которых есть ЭВМ, благодаря которой программа будет реализована. Для удобного и безошибочного использования того или иного проекта необходимо подбирать подходящие технические составляющие.

Выделены следующие средства:

- центральный процессор x64 не менее 4-х ядер;
- оперативная память не менее 4 ГБ;
- компьютерная мышь и клавиатура;
- накопители информации не менее 512 ГБ.

Так как программа будет не особо требовательной, поэтому и технические составляющие будут не особо внушительными. Работа с проектом включает в себя ввод данных, поэтому для полноценной реализации и демонстрации функционала необходимы мышь и клавиатуры.

В процессе построения и эксплуатации необходимы программные средства, благодаря которым на выходе выйдет продукт, что сможет конкурировать с другими автоматизированными рабочими местами.

Для обеспечения функционала и вычислений, разработки понадобится операционная система Windows 10. Так как эта ОС довольно распространена на рынке, имеет понятный функционал, для создания программы она хорошо подойдет. Также она вполне доступная для большинства людей. Если сравнивать с другими конкурентами, то типу Windows 7 и Windows 11, то тут всё уходит в преимущество Windows 10. Windows 7 очень уязвима к кибератакам и работает в режиме

Инф. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.	Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-31-14-2023-ПЗ	Лист 18
--------------	--------------	--------------	--------------	--------------	--------------	----	------	----------	-------	------	---------------------	------------

эмуляции, что снижает производительность, так как разработчики уже не пишут ПО для неактуальных ОС. Windows 11 считается недавней версией, но на данный момент оставляет желать лучшего. Так как время от времени нестабильно себя ведёт и имеет множество функций, которые заблокированы. Поэтому выбор остался за Windows 10.

Для разработки самой программы используется среда разработки Visual Studio 2022. Её компилятор использует библиотеки Windows, что помогает делать проекты под Windows. Сама среда довольно отзывчивая и понятная для разработчика. Если сравнивать её с Eclipse, то она сложная и непонятная для новичков и некоторых разработчиков. У неё также нет гарантий, что работа будет надёжной. Так как разработчики плагинов данной среды – само сообщество. Code::Blocks тоже не подходит для разработки. Он не подходит для создания комплексных приложений, которые обладают огромным функционалом.

Язык программирования – C#. Данный язык довольно востребован и отлично подходит для разработки в Visual Studio. C# автоматически очищает данные, совместим с Windows, имеет строгую типизацию. Сравнивая его с C++, то он имеет в доступе множество полезных библиотек в свободном доступе. Сам код выглядит лаконичнее у C#, чем у C++. Если же брать для сравнения Python, то ему не хватает производительности и скорости, поэтому он не самый подходящий вариант для создания программного продукта.

Платформа – .NET. Она производительная, кроссплатформенная, имеет мощную библиотеку классов. Возможно использовать и .NET Framework, но эта платформа уже не актуальна, так как последнее обновление было в 2019 году. Она может помешать работе других приложений. .NET Framework зависит от версии ОС, для приложений на данной платформе требуется установщик. Поэтому, выбор сделан в пользу актуальности и удобства.

Для обеспечения решений задач предметной области и определения области применения программы нам необходимы и другие прикладные программы.

Для хранения данных в БД, понадобится СУБД, одна из которых является SQL Server 2019, которая в свою очередь будет работать с помощью среды SQL

Инв. № подл.	Подп. и дата
Инв. № докл.	Взам. инв. №
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-31-14-2023-ПЗ	Лист
						19

Server Management Studio. Она подходит для разработчиков и имеет полный функционал работы. Если сравнивать её с PostgreSQL, то она в свою очередь обладает конфигурацией, которая может заставить напрячься неподготовленных пользователей. Скорость данной СУБД может время от времени падать в зависимости от запросов. Для работы с MySQL придётся приложить огромное количество усилий, после которых получится выполнить определенную задачу. Поэтому был сделан выбор в пользу SQL Server 2019.

3.2 Физическая структура программы

Для создания необходимого программного продукта были учтены следующие подходы:

1. Имена переменных, классов и функций прямо указывают на их назначение.
2. Всегда удалять лишний код вместо того, чтобы его комментировать.
3. Оставлять необходимые комментарии, описывающие важные куски кода.
4. Функции всегда выполняют только одно действие.
5. Данные должны быть валидными.

Созданный программный продукт содержит в себе следующую файловую структуру: папку с изображениями, папку с конвертерами, папку с классами из базы данных и сами файлы с окнами и их взаимодействиями.

В папке с изображениями (Images) содержатся:

- change.png – изображение для кнопки изменения;
- cross.png – изображение для кнопки удаления;
- pngegg.png – основное изображение, логотип программы;
- exit.png – изображение для кнопки выхода из программы;
- url.png – изображение для кнопки для просмотра изображения;
- pngegg.ico – изображение для иконки программного продукта.

Инв. № подл.	Подп. и дата	Инв. № докл.	Взам. инв. №	Подп. и дата						
Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-31-14-2023-ПЗ					Лист
										20

В папке с конвертерами (Converters) содержится только файл DateToBackground.cs, который в свою очередь помогает демонстрировать выдачи, у которых пропущен срок сдачи, отмечая данный элемент красным цветом.

Папка с классами (Models) создаётся при подключении базы данных, содержащая в себе:

1. Book.cs – класс со свойствами столбцов из таблицы «Книги».
2. Reader.cs – класс со свойствами столбцов из таблицы «Читатель».
3. Genre.cs – класс со свойствами столбцов из таблицы «Жанр».
4. Provisioner.cs – класс со свойствами столбцов из таблицы «Издательство».
5. Extradition.cs – класс со свойствами столбцов из таблицы «Выдача».
6. WellLibraryContext.cs – контекст, необходимый для работы с данными из базы данных.

Далее идут файлы окон:

1. MainWindow.xaml/cs – окно, от которого начинается включение программы.
2. BookMain.xaml/cs – окно, в котором доступен весь список возможностей и лист с книгами.
3. Readers.xaml/cs – окно, в котором доступен список занесенных читателей, возможность их изменять и удалять.
4. AddReaderWindow.xaml/cs – окно, необходимое для занесения новых данных о читателе.
5. AddBookWindow.xaml/cs – окно, необходимое для занесения и изменений данных книг.
6. AddVidWindow.xaml/cs – окно, необходимое для занесения и изменения выдач книг читателю.
7. VidBooks.xaml/cs – окно, необходимое для просмотром сроков выдач и возможных их изменений.
8. App.xaml – файл для ресурсов, стилей, триггеров.
9. Session.cs – файл, необходимый для работы с контекстом подключенной базы данных.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.	КП-ПР-31-14-2023-ПЗ				Листм
										21
Ли	Изм.	№ докум.	Подп.	Дата						

4 ТЕСТИРОВАНИЕ

Для успешной работы программного продукта в дальнейшем проводится тестирование. Тестирование — процесс исследования программы с целью получения информации о качестве продукта.

Для полной и качественной проверки продукта нужно выбрать необходимые методы и подходы тестирования, которые в свою очередь смогут дать необходимую информацию.

Проверка делится на 3 уровня:

1. Модульное – тестируется минимально возможный для тестирования компонент, например, отдельный класс или функция.
2. Интеграционное – тестируются интерфейсы между компонентами, подсистемами.
3. Системное – тестируется интегрированная система на её соответствие требованиям.

При интеграционном программировании необходимо будет проверить взаимодействие модулей между собой, интерфейсы работ с базой данных, внешние аппаратные интерфейсы, неправильные обработки исключений.

Для интеграционного тестирования необходимо добавление модулей в одну общую систему. Для этого был выбран нисходящий подход. Таким образом, будет проще локализовать баги, можно уже на ранних этапах получать возможность тестирования, для этого понадобится много заглушек, но чем раньше начнётся тестирование, тем проще будет на более поздних этапах.

Во время тестов будут составляться тестовые сценарии, благодаря которым будут фиксироваться все возможные ошибки. Всё это будет повторяться до тех пор, пока все ошибки не будут замечены, чтобы потом отправить их на исправление.

Для проверки программы будет использоваться метод черного ящика, с его помощью и будут создаваться необходимые сценарии для тестов. В учёт не будет входить внутренняя структура программы, только сам интерфейс и взаимодействие

Инф. № подл.	Подп. и дата	Взам. инв. №	Инв. № дцкл.	Подп. и дата	Инв. № подл.	Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-31-14-2023-ПЗ	Лист 22
--------------	--------------	--------------	--------------	--------------	--------------	----	------	----------	-------	------	---------------------	------------

При тестировании черным ящиком будут анализироваться классы эквивалентности и граничных значений, особенно второй анализ, потому что именно в нём происходит множество ошибок. Таким образом, возможность понять, соблюдаются ли границы у необходимых данных, правильны ли булевы условия и верны ли определенные значения.

На данном этапе все тесты прошли проверку, фактические результаты соответствуют ожидаемым. Протоколы тестирования приведены в приложении 1.

[illegible]

Инв. № подл.	Подп. и дата	Инв. № дудл.	Взам. инв. №	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

Лист
24

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Профессия Библиотекарь [Электронный ресурс]. – Режим доступа: <https://www.profguide.io/professions/Bibliotekar.html> (дата обращения: 22.02.2023).
2. Библиотека – Википедия [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Библиотека> (дата обращения: 22.02.2023).
3. Функциональные и нефункциональные требования: полное руководство [Электронный ресурс]. – Режим доступа: <https://bestprogrammer.ru/izuchenie/funktsionalnye-i-nefunktsionalnye-trebovaniya-polnoe-rukovodstvo> (дата обращения: 28.02.2023).
4. Диаграмма вариантов использования (UseCase diagram) | Flexberry PLATFORM Documentation [Электронный ресурс]. – Режим доступа: https://flexberry.github.io/ru/fd_use-case-diagram.html (дата обращения: 10.03.2023).
5. Диаграмма состояний (Statechart diagram) | Flexberry PLATFORM Documentation [Электронный ресурс]. – Режим доступа: https://flexberry.github.io/ru/fd_statechart-diagram.html (дата обращения: 10.03.2023).
6. Диаграмма активностей (Activity diagram) | Flexberry PLATFORM Documentation [Электронный ресурс]. – Режим доступа: https://flexberry.github.io/ru/fd_activity-diagram.html (дата обращения: 10.03.2023).
7. Требования к системе Visual Studio 2022 | Microsoft Learn [Электронный ресурс]. – Режим доступа: <https://learn.microsoft.com/ru-ru/visualstudio/releases/2022/system-requirements> (дата обращения: 21.03.2023).
8. Выбор между .NET и .NET Framework для серверных приложений | Microsoft Learn [Электронный ресурс]. – Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/standard/choosing-core-framework-server> (дата обращения: 21.03.2023).
9. Лучшие IDE для разработки на C# [Электронный ресурс]. – Режим доступа: https://gb.ru/posts/c_sharp_ides (дата обращения: 21.03.2023).

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	КП-ПР-31-14-2023-ПЗ					Лист 25
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Ли	Изм.	№ докум.	Подп.	Дата	

18. Виды тестирования по доступу к коду – Школа седого тестировщика [Электронный ресурс]. – Режим доступа: <https://sedtest-school.ru/types-of-testing/vidy-testirovaniya-po-dostupu-k-kodu/> (дата обращения: 02.04.2023).

19. Тестирование по стратегии чёрного ящика – Википедия [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Тестирование_по_стратегии_чёрного_ящика (дата обращения: 02.04.2023).

Инв. № подл.	Подп. и дата				Взам. инв. №	Инв. № дцбл.	Подп. и дата	Инв. № подл.	
Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-31-14-2023-ПЗ				Лист
									27

ПРИЛОЖЕНИЕ 1

Таблица 1 — Окно для входа

Поле	Описание
Название проекта	Тестирование АРМ работника библиотеки
Рабочая версия	1.0
Имя тестирующего	Пинигина Дарья
Дата(ы) теста	29.03.2023
Тестовый пример #	authorization_is_librarian_wrong_login_password #1
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Авторизация в программе с неправильным логином и паролем.
Краткое изложение теста	Ввод неверного логина и пароля.
Этапы теста	<ol style="list-style-type: none"> 1. Запускаем программу. 2. В поле «Логин» вводим неправильные данные. 3. В поле «Пароль» вводим неправильные данные. 4. Нажимаем на кнопку «Войти». 5. Ожидаем результат.
Тестовые данные	Неверный логин «library», пароль «110378».
Ожидаемый результат	Вывод сообщения: «Неверные данные».
Фактический результат	Вывод сообщения: «Неверные данные».
Предварительное условие	Запустить среду разработки Visual Studio.
Постусловие	Стабильное, без вылетов, кодов ошибок, сбоев.
Статус (Зачет/Незачет)	Зачет.
Примечания/комментарии	Главное не забывать заполнить данные.

Конец таблицы 1

Инв. № подл.	Подп. и дата	Взам. инв. №	Подп. и дата	Инв. № дубл.
Ли	Изм.	№ докум.	Подп.	Дата
КП-ПР-31-14-2023-ПЗ				Лист 28

Таблица 2 — Окно добавления книги

Поле	Описание
Название проекта	Тестирование АРМ работника библиотеки
Рабочая версия	1.0
Имя тестирующего	Пинигина Дарья
Дата(ы) теста	29.03.2023
Тестовый пример #	add_book #2
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Добавление книги в базу данных с неверным ISBN.
Краткое изложение теста	Добавление книги с неверным ISBN.
Этапы теста	<ol style="list-style-type: none"> 1. Запускаем программу. 2. В поле «Логин» вводим правильные данные. 3. В поле «Пароль» вводим правильные данные. 4. Нажимаем на кнопку «Войти». 5. Нажимаем кнопку «Добавить книгу». 6. Вводим неправильный ISBN, название, автора, жанр, издательство, год издания, количество, цену. 7. Нажимаем «Добавить». 8. Проверяем результат.
Тестовые данные	ISBN – «9785051759650», Название – «Золотые земли. Сокол и Ворон», Автор – «Ульяна Черкасова», Жанр – «Фэнтэзи», Издательство – «Эксмо», Количество – 5, Год издания – 2023, Цена – 535.
Ожидаемый результат	Сообщение «Введены некорректные данные».
Фактический результат	Сообщение «Введены некорректные данные».
Предварительное условие	Запустить среду разработки Visual Studio.
Постусловие	Стабильное, без вылетов, кодов ошибок, сбоев.
Статус (Зачет/Незачет)	Зачет.

Инв. № подл.	Подп. и дата
Инв. № докл.	Взам. инв. №
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-31-14-2023-ПЗ

Лист

29

Примечания/комментарии	Важно, чтобы это главное окно стабильно работало и была подключена база данных.
------------------------	---

Конец таблицы 2

Таблица 3 — Окно добавления читателя

Поле	Описание
Название проекта	Тестирование АРМ работника библиотеки
Рабочая версия	1.0
Имя тестирующего	Пинигина Дарья
Дата(ы) теста	29.03.2023
Тестовый пример #	add_reader_wrong_email_phone #3
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Добавление читателя с неверной почтой и телефоном.
Краткое изложение теста	Неверная почта и телефон.
Этапы теста	<ol style="list-style-type: none"> 1. Запускаем программу. 2. В поле «Логин» вводим правильные данные. 3. В поле «Пароль» вводим правильные данные. 4. Нажимаем на кнопку «Войти». 5. Нажимаем кнопку «Добавить читателя». 6. Вводим фамилию, имя, отчество, неверную почту и неверный телефон. 7. Нажимаем «Добавить». 8. Проверяем результат.
Тестовые данные	Фамилия – Сорокин, Имя – Владислав, Отчество – Васильевич, Почта – sdsdk.gmail, Телефон – 9904756029.
Ожидаемый результат	Сообщение «Введены неверные данные».
Фактический результат	Сообщение «Введены неверные данные».
Предварительное условие	Запустить среду разработки Visual Studio.
Постусловие	Стабильное, без вылетов, кодов ошибок, сбоев.

Инв. № подл.	Подп. и дата
Инв. № докл.	Взам. инв. №
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-31-14-2023-ПЗ

Лист

30

Статус (Зачет/Незачет)	Зачет.
Примечания/комментарии	Важно, чтобы это главное окно стабильно работало и была подключена база данных.

Конец таблицы 3

Таблица 4 — Окно оформления выдачи

Поле	Описание
Название проекта	Тестирование АРМ работника библиотеки
Рабочая версия	1.0
Имя тестирующего	Пинигина Дарья
Дата(ы) теста	29.03.2023
Тестовый пример #	add_extradition_wrong #4
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Оформление выдачи с неверными сроками.
Краткое изложение теста	Неверный срок.
Этапы теста	<ol style="list-style-type: none"> 1. Запускаем программу. 2. В поле «Логин» вводим правильные данные. 3. В поле «Пароль» вводим правильные данные. 4. Нажимаем на кнопку «Войти». 5. Выбираем из списка книгу. 6. Нажимаем кнопку «Оформить выдачу». 7. Выбираем читателя из списка. 8. Указываем дату выдачи и дату сдачи. 9. Нажимаем «Добавить». 10. Проверяем результат.
Тестовые данные	Читатель – Сорокин Владислав Васильевич, дата выдачи – 06.04.23, дата сдачи – 13.04.23.
Ожидаемый результат	Сообщение «Некорректные данные».
Фактический результат	Сообщение «Некорректные данные».

Инв. № подл.	Подп. и дата
Инв. № докл.	Взам. инв. №
Подп. и дата	Инв. № подл.
Инв. № подл.	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата
----	------	----------	-------	------

КП-ПР-31-14-2023-ПЗ

Лист

31

Предварительное условие	Запустить среду разработки Visual Studio.
Постусловие	Стабильное, без вылетов, кодов ошибок, сбоев.
Статус (Зачет/Незачет)	Зачет.
Примечания/комментарии	Важно, чтобы это главное окно стабильно работало и была подключена база данных.

Конец таблицы 4

[illegible]

КП-ПР-31-14-2023-ПЗ

Луст

32