**User stories**

Как пользователь я могу добавлять, редактировать и удалять выбранную книгу в таблице, просматривать поля, а также иметь возможность поиска книги в базе по любому параметру

**Use case**

**Название:** «Может быть инициирован»

**Цель:** Отображение пользовательского интерфейса

**Начальное состояние:** Выполнение открытия программы и инициализация

**Основной сценарий:**

1. Пользователь открывает программу
2. Отображается таблица с данными

**Альтернативный сценарий:**

* 1. Если сервер не отвечает, по тем или иным причинам, выводится сообщение об ошибке
  2. Пользователь закрывает программу

**Сценарий обработки ошибок**

Предусловие: на шаге 1 основного сценария происходит ошибка отображения книг из базы (сбой сети и т. п.)

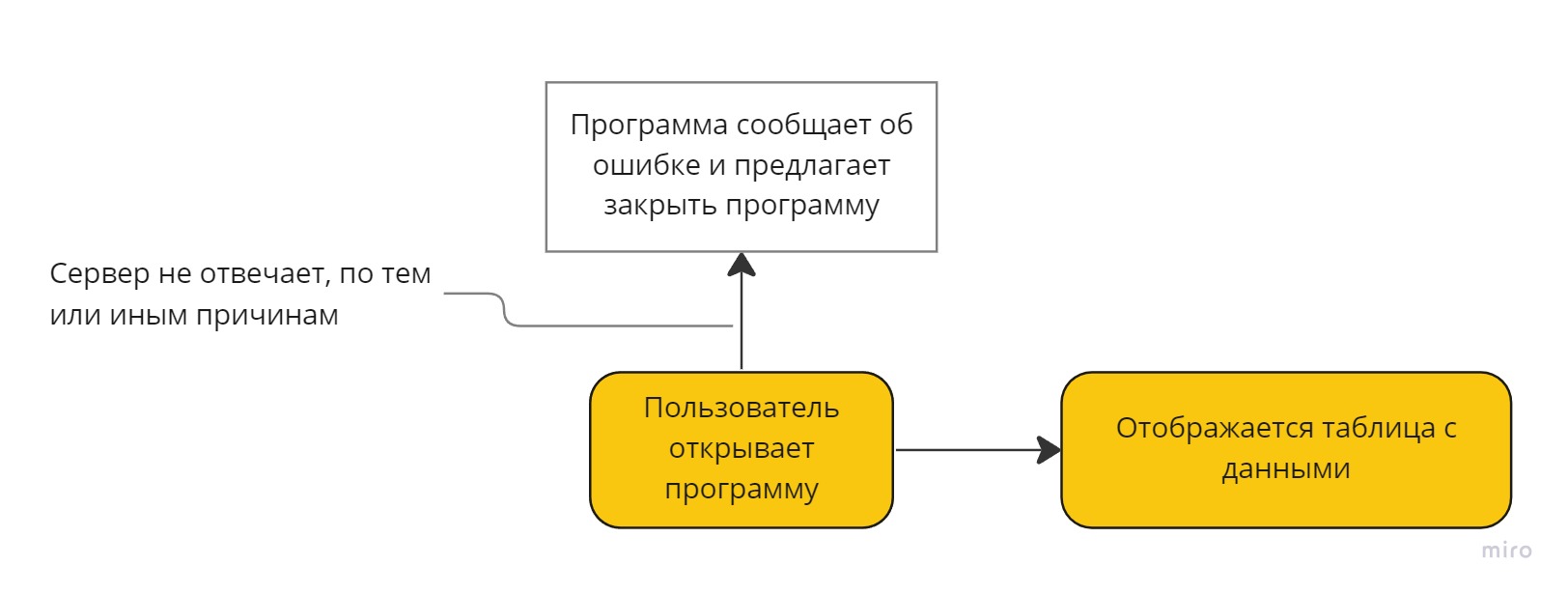


Рис. 1 Графическое изображение альтернативного сценария

**Название:** «Можно добавить новую книгу»

**Цель:** Добавить книгу в базу

**Начальное состояние:** Программа инициализирована

**Основной сценарий:**

1. Происходит нажатие на кнопку «добавить книгу»
2. Пользователь заполняет поля
3. Нажимает на кнопку «сохранить»
4. Проверка корректности ввода
5. Сохранение книги в базе

**Альтернативный сценарий:**

* 1. На шаге 3 пользователь имеет возможность закрыть диалоговое окно с редактированием новой книги, и она не добавится
  2. Отображение ошибки при неправильном вводе полей
  3. Если сервер не отвечает, по тем или иным причинам, выводится сообщение об ошибке

**Сценарий обработки ошибок**

Предусловие: на шаге 5 основного сценария происходит ошибка сохранения книги в базе (сбой сети и т. п.)

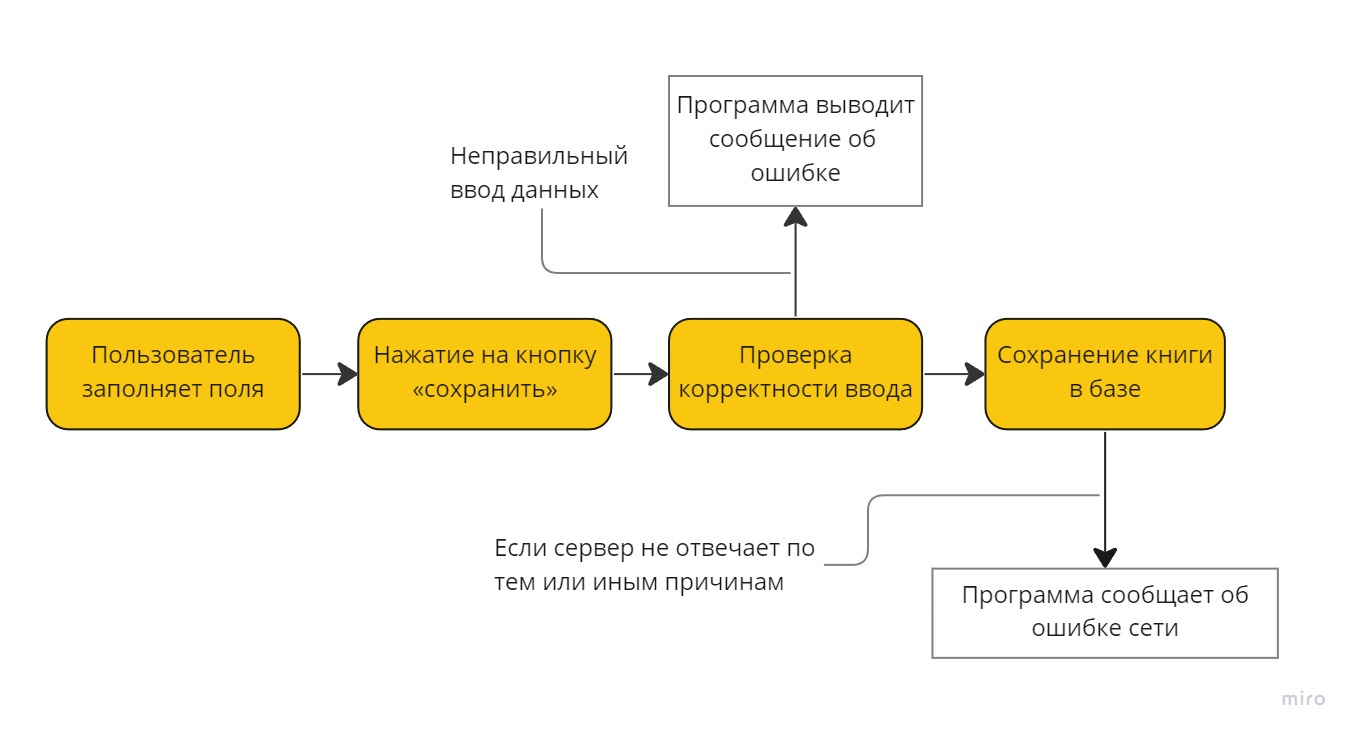


Рис. 2 Графическое изображение альтернативного сценария

**Название:** «Можно найти книгу»

**Цель:** Получить результат поискового запроса

**Начальное состояние:** Программа инициализирована

**Основной сценарий:**

1. Юзер набирает текст в строку поиска
2. Программа в режиме реального времени выводит книги, подходящие под запрос

**Название:** «Можно редактировать книгу»

**Цель:** Изменить данные книги

**Начальное состояние:** Программа инициализирована

**Основной сценарий:**

1. Юзер выбирает в таблице запись
2. Нажимает кнопку редактировать
3. Появляется окно редактирования книги
4. Пользователь вносит изменения в поля
5. Нажимает кнопку «сохранить»
6. Изменения отображаются в таблице

**Альтернативный сценарий:**

* 1. Книга не выбрана – всплытие окна, уведомляющем об ошибке
  2. На шаге 4 пользователь может не изменять книгу
  3. Сервер может не отвечать, по тем или иным причинам, в таком случае выводится сообщение об ошибке

**Сценарий обработки ошибок**

Предусловие: на шаге 5 основного сценария происходит ошибка изменения книги в базе (сбой сети и т. п.)

1. Программа сообщает об ошибке

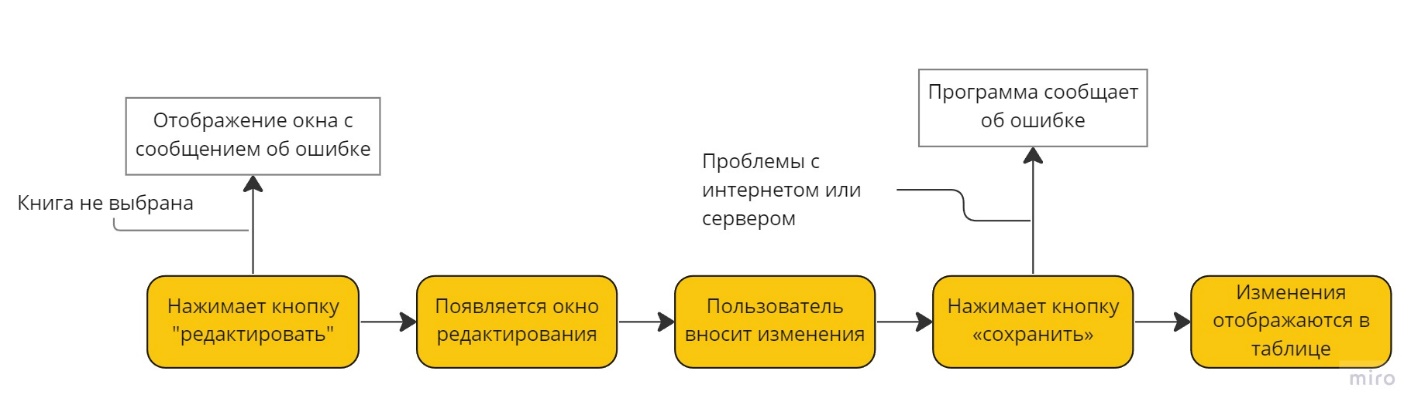


Рис. 3 Графическое изображение альтернативного сценария

**Название:** «Можно удалить книгу»

**Цель:** Удаление данных книги

**Начальное состояние:** Программа инициализирована, пользователь выбрал книгу

**Основной сценарий:**

1. Пользователь выбирает в таблице запись
2. Нажимает кнопку удалить
3. В таблице исчезнет выбранная запись

**Альтернативный сценарий:**

* 1. Книга не выбрана – всплытие окна, уведомляющем об ошибке
  2. Если сервер не отвечает, по тем или иным причинам, выводится сообщение об ошибке и пользователю предлагается сохранить данные в файл

**Сценарий обработки ошибок**

Предусловие: на шаге 2 основного сценария происходит ошибка удаления книги (сбой сети и т. п.)

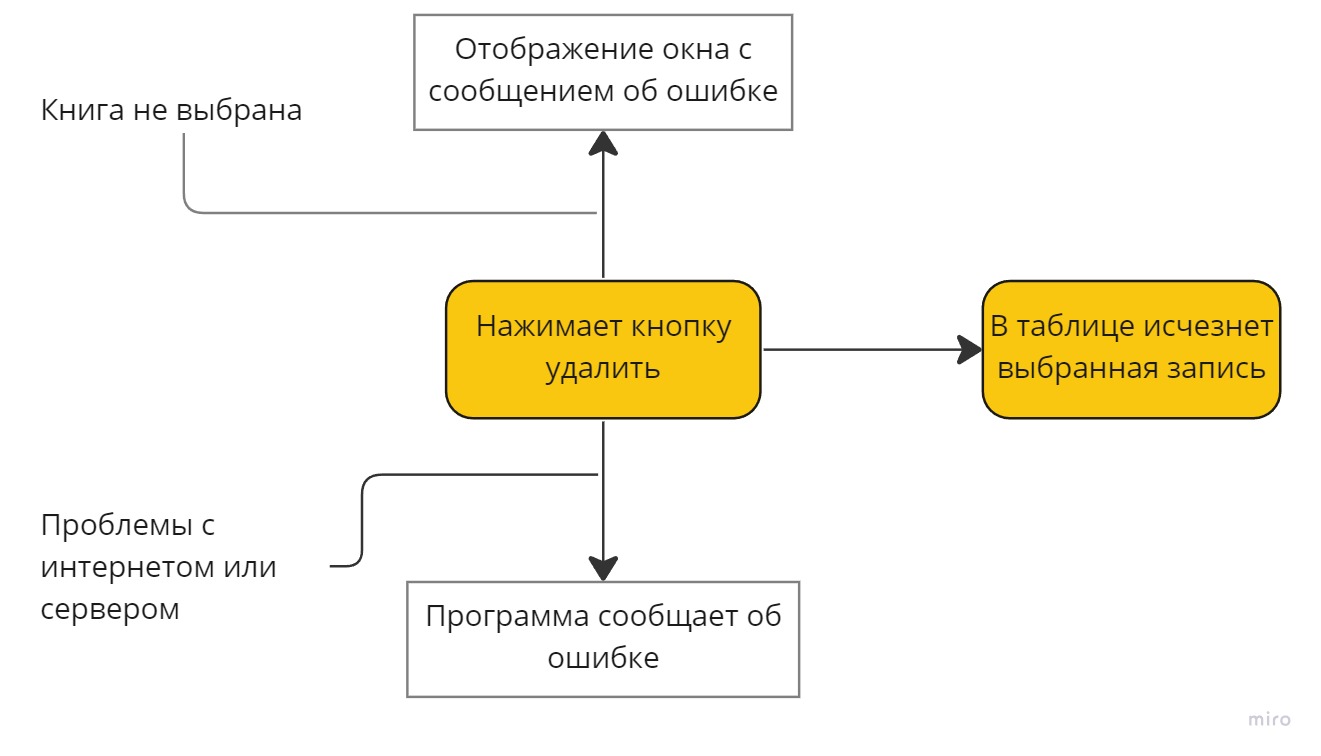
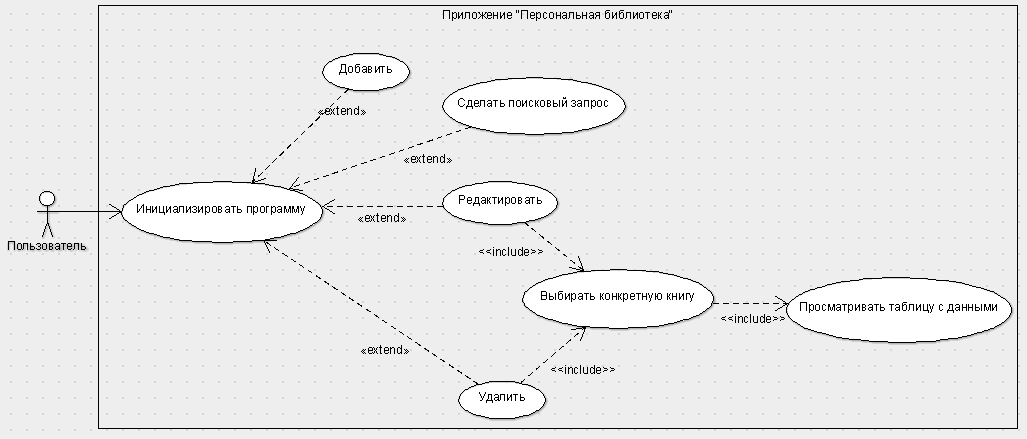


Рис. 4 Графическое изображение альтернативного сценария

**Use case diagram**



**Specification**

1. **Общие сведения**
2. **Полное наименование системы и ее условное обозначение**

Наименование системы: «Личная библиотека»

1. **Наименование разработчика системы и реквизиты заказчика**

Заказчик – Заведующий отделом информационных технологий, Преподаватель первой квалификационной категории Смольняков Александр Витальевич.

Разработчик – студент группы ИС-3 Молчанов Павел Дмитриевич

1. **Основания для разработки АС**

Работа по созданию и изучению клиент-серверной программы, которая позволяет администрировать библиотеку

1. **Порядок оформления и предъявления заказчику результатов работ по созданию системы:**

К результатам труда разработчика относится:

* оригинальное программное обеспечение
* типовые проектные решения и особенности построения клиент-серверной архитектуры
* проектная и рабочая документация

Заказчику передаются:

* программное обеспечение
* документация

Заказчик приобретает у третьих лиц:

* пассивное сетевое оборудование
* серверное оборудование
* аппаратное оборудование

1. **Назначение и цели создания системы**
2. **Назначение системы**

Программа позволяет вести личную электронную библиотеку

1. **Цели создания системы**

Целью создания системы является:

* Получение новых знаний в сфере разработки архитектуры ПО
* Изучение технологий java fx и spring framework
* Упрощение взаимодействия с программой

1. **Характеристика объекта автоматизации**
2. **Краткие сведения об объекте автоматизации**

Объектом автоматизации является система по администрированию библиотеки через интернет. Требуется автоматизировать поиск, удаление, сохранение информации в базе данных

1. **Требования к системе**
2. **Требование к структуре и функционированию системы**

Структура системы реализована по концепции MVC, где моделью является данные книги, автора, издательства.

Книга имеет следующие поля:

* Автор (сущность)
  + Имя (может быть пустым, но не одновременно)
  + Фамилия (может быть пустым, но не одновременно)
  + Отчество (может быть пустым, но не одновременно)
* Издательство (сущность)
  + Название
  + Город издательства
* Год издания
* Название
* Раздел библиотеки (специальная литература, хобби, беллетристика и т.п.)
* Происхождение книги

Сущность Автор имеет следующие поля:

* Имя (может быть пустым, но не одновременно)
* Фамилия (может быть пустым, но не одновременно)
* Отчество (может быть пустым, но не одновременно)
* Список написанных книг

Сущность Издательство имеет следующие поля:

* Название
* Город издательства
* Список изданных книг

ИС администрирования библиотеки должна включать в себя 2 подсистемы:

функционал клиента:

* добавление книги
* удаление выбранной книги
* редактирование выбранной книги
* поиск по любым полям

функционал сервера:

* получение списка книг из БД
* добавление книги в БД
* редактирование книги в БД
* удаления книги по идентификатору в БД
* коннект с базой данных
* выводы ошибок, при их наличии
* отправка данных через api
* получение данных через api

1. **Требования к средствам и способам связи для информационного обмена между компонентами системы**

Для информационного обмена между компонентами системы должна быть организована компьютерная сеть. ИС функционирует как на клиенте, так и на сервере, к которому имеют доступ пользователи через эту сеть

1. **Требования по эргономике и технической эстетике**

Программа, работающая на клиенте, должна иметь минималистичный интерфейс:

* Основной цвет: светло-зеленый
* Другой цвет должен гармонировать с основным
* Шрифт должен хорошо читаться
* Размер шрифта в диапазоне от 12 до 32 пунктиров
* Все кнопки должны находится в ожидаемых местах

1. **Требования по сохранности информации**

Сохранность информации должна быть обеспечена в следующих случаях:

* Выход из строя аппаратных систем
* Ошибки в программных средствах
* Неверные действия пользователя

Для сохранности информации необходимо предусмотреть использование блоков бесперебойного питания для защиты данных от повреждения в случае отключения питания, для надёжного хранения данных необходимо производить ежедневное резервное копирование БД на несколько дисков

Для выполнения операции отката и повышения надёжности хранения базы данных предусмотреть раздельное хранение двух дополнительных копий (с возможностью сохранения на различных физических носителях)

1. **Состав и содержание работ по созданию системы**

Перечень документов, предъявляемых по окончании соответствующих стадий по созданию системы, представлен ниже:

1. Эскизный проект - описание функций, функций подсистем, их целей
2. Технический проект - описание ПО, информационной базы, интерфейса
3. Ввод в действие – испытание и устранение неполадок
4. **Порядок контроля и приемки системы**

Установить контроль и приемку результатов работ на каждой стадии создания системы в соответствии с разделом 5.0

На стадии 3 принимается готовая версия программного продукта.

Приемка этапа заключается в рассмотрении и оценке проведенного объема работ и предъявленной технической документации в соответствии с требованиями настоящего технического задания.

Ответственность за организацию и проведение приемки системы должен нести заказчик. Приемка системы должна производиться по завершению приемки всех задач системы. При этом необходимо предоставить обеспечение материальной частью (технические средства), проектной документацией.

Завершающим этапом при приемке системы должно быть составление акта приемки

1. **Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие**

Для обеспечения готовности объекта к вводу в действие, провести установку технических и программных средств

1. **Требования к документированию**

Проектная документация должна быть разработана в соответствии с ГОСТ 34.201-89.