МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Вятский государственный университет»**

**Колледж ВятГУ**

**КУРСОВОЙ ПРОЕКТ**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| По междисциплинарному курсу | | | | | МДК 05.01 Проектирование и дизайн информационных | | | | |
| систем | | | | |
| Тема | Проектирование информационной системы научной библиотеки | | | | | | | | | |
|  | | | | | | | | | |
| Студента (ки) | | | | Максимова Дмитрия Александровича | | | | | | |
|  | | | *ФИО (полностью в родительном падеже)* | | | | | | | |
| Курс | | 3 | | | | Форма обучения | очная | |
|  | | *(арабской цифрой)* | | | |  | | *(очная, заочная)* | | |
| Основная профессиональная образовательная программа по специальности | | | | | | | | | | |
| 09.02.07 Информационные системы и программирование | | | | | | | | | | |
| (код и наименование специальности без кавычек) | | | | | | | | | | |

Руководитель курсового проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись) (фамилия, инициалы)

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(прописью, без сокращений)*

Киров, 2022

**РЕФЕРАТ**

Брагина С. А.  Разработка технического задания на создание автоматизированного программного обеспечения для ведения клиентской базы спортивного клуба «Fitness Energy», а также для продажи товаров и услуг, предоставляемых компанией. ПЗ: курс. Проект / Колледж ВятГУ; рук. Мамаева Е. А. - Киров, 2021. ПЗ 41 с., 23 рис., 6 источников, 3 табл.

Обзор предметной области, расширенное техническое задание, схемы бизнес-процессов и их описания, проектирование программного обеспечения.

Объект исследования - программные обеспечения и системы для ведения клиентской базы и работы спортивного клуба.

Цель работы – разработать техническое задание и модельную часть программного обеспечения по созданию работы спортивного клуба с учетом ведения клиентской базы и предложения услуг и товаров.

Задание на курсовой проект – разработать техническое задание и модель программного обеспечения в соответствии с выбранной темой.

**СОДЕРЖАНИЕ**

[**ЦЕЛЬ ПРОВЕДЕНИЯ ОБСЛЕДОВАНИЯ** 5](#_Toc91500529)

[**1.** **Организационная структура спорт клуба «Fitness Energy»** 6](#_Toc91500530)

[1.1. Назначение и информационные связи подразделений 6](#_Toc91500531)

[1.2. Требования к программному обеспечению 7](#_Toc91500532)

[1.3. Анализ подразделений 8](#_Toc91500533)

[**2.** **Расширенное техническое задание** 10](#_Toc91500534)

[2.1. Введение 10](#_Toc91500535)

[2.2. Сокращение слов 10](#_Toc91500536)

[2.3. Наименование программы: 10](#_Toc91500537)

[2.4. Область применения 11](#_Toc91500538)

[2.5. Основания для разработки 11](#_Toc91500539)

[2.6. Назначение разработки 11](#_Toc91500540)

[2.6.1. Функциональное назначение 11](#_Toc91500541)

[2.7. Эксплуатационное назначение 11](#_Toc91500542)

[2.8. Требования к программе 11](#_Toc91500543)

[2.8.1. Требования к функциональным характеристикам 11](#_Toc91500544)

[2.8.2. Требования к составу выполняемых функций 11](#_Toc91500545)

[2.9. Требования к надёжности 12](#_Toc91500546)

[2.10. Требования к обеспечению надёжного функционирования программы 13](#_Toc91500547)

[2.11. Требования к информационной и программной совместимости 13](#_Toc91500548)

[2.12. Требования к информационной и программной совместимости 13](#_Toc91500549)

[2.13. Требования к программной документации 13](#_Toc91500550)

[2.14. Технико-экономические показатели 14](#_Toc91500551)

[2.15. Стадии и этапы разработки 14](#_Toc91500552)

[2.16. Порядок контроля и приемки 15](#_Toc91500553)

[**3.** **Схемы бизнес-процессов и их описание** 16](#_Toc91500554)

[3.1. Функциональный анализ ПО 16](#_Toc91500555)

[2.2. Структурный анализ системы 20](#_Toc91500556)

[2.3. Логическая модель ПО 23](#_Toc91500557)

[2.4. Физическая модель ПО 27](#_Toc91500558)

[**4.** **Проектирование ПО** 31](#_Toc91500559)

[4.1. Формирование диаграммы Use Case 31](#_Toc91500560)

[4.2. Формирование схемы алгоритма функционирования АРМ ИС 32](#_Toc91500561)

[4.3. Экранные формы 33](#_Toc91500562)

[**ЗАКЛЮЧЕНИЕ** 40](#_Toc91500563)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 41](#_Toc91500564)

# **ЦЕЛЬ ПРОВЕДЕНИЯ ОБСЛЕДОВАНИЯ**

В процессе предпроектного обследования научной библиотеки «Science Knowledge» изучаются основные направления деятельности, оказываемые услуги для читателей, и организационная структура библиотеки. Определяются функции подразделений, существующие информационные взаимосвязи между подразделениями, внутренний и внешний документооборот.

На основе анализа указанной информации определяются требуемые учетные подсистемы, охватывающие несколько подразделений, каждое из которых заинтересовано в оперативности и актуальности данных. Разрабатываются рекомендации по усовершенствованию документооборота, исключению дублирования информации.

Изучается текущий уровень автоматизации: определяется перечень разработанных подсистем, состав автоматизированных рабочих мест и круг решаемых задач с целью определения функциональной полноты системы и автоматизации учетных функций. Разрабатываются предложения по требуемому составу выбранных подсистем, уточнению перечня задач, подлежащих автоматизации, и расширению состава автоматизированных рабочих мест с целью получения полной оперативной информации по бухгалтерскому, оперативному и управленческому учету производственно-хозяйственной деятельности научной библиотеки «Science Knowledge».

Осуществляется обследование существующих бизнес-процессов и бизнес-процедур.

Формируется организационно-функциональная схема автоматизации и разрабатываются требования к проектируемой системе.

1. **Организационная структура научной библиотеки**
   1. Назначение и информационные связи подразделений

* Генеральный директор руководит производственно-хозяйственной и финансово-экономической деятельностью ООО ««Science Knowledge»», неся всю полноту ответственности за последствия принимаемых решений, финансово-хозяйственные результаты деятельности. Организует работу и взаимодействие всех структурных подразделений, направляет их деятельность на развитие и совершенствование производства, повышение эффективности работы, увеличение прибыли, качества и конкурентоспособности услуг.
* Администратор работает с читателями и книгами (выдает книги, добавляет новых читателей и книги). Ведет записи, работает с программой управленческого учета и CRM. Руководит обслуживающим персоналом.
* Обслуживающий персонал информирует читателей о книжном фонде библиотеки, помогает подбирать книги и выдает их читателям.
* Бухгалтер отвечает за управление денежными средствами предприятия, обеспечение своевременного выполнения компанией обязательств. Обеспечивает работу по анализу планово-экономической деятельности производства и подготовке бухгалтерской, налоговой, управленческой отчетности. Осуществляет контроль за информационно-техническим отделом. Координирует работу с банками и лизинг-компаниями.
* Клининговый работник отвечает за чистоту библиотечных залов и дополнительных помещений.
* Системный администратор поддерживает компьютеры, сервера, сети и другое оборудование в рабочем состоянии.

На рисунке 1 представлена структурная схема научной библиотеки «Science Knowledge».

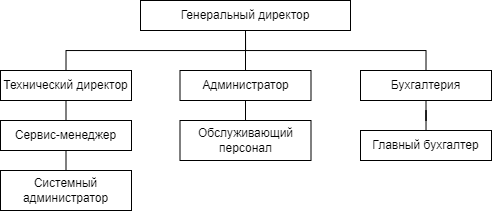


Рисунок 1 – Структурная схема библиотеки «Science Knowledge»

* 1. Требования к программному обеспечению

ПО рабочего места представляет собой программный комплекс, разработанный независимо от других, и работающий автономно. Программа написано в системе СУБД SQLite.

На основании описания предметной области можно выделить требование, которому должна удовлетворять проектируемое ПО.

Разрабатываемая система создается для ведения читательской базы, сотрудников и книжного фонда в научной библиотеке. Она предназначена для внесения, хранения, вывода информации в удобном формате.

Функции системы:

1. Возможность редактирования и просмотра сотрудников, читателей и книг согласно предоставляемому доступу администратором.
2. Возможность выдачи книг читателям, а также оформление сдачи книги читателем.

Основная задача системы – это обеспечить состояния работы с информацией, при котором могут происходить внесения и редактирования читателей, книг и сотрудников в базе данных, заполнение информации о взятых читателем книгах и сроках их возврата.

Программа реализована для эксплуатации ее сотрудниками.

* 1. Анализ подразделений

*Управленческий блок (Администрация)*

Управленческий блок непосредственно Генеральному директору. Структурные подразделения данного блока обеспечивают административную работу всего остального производства. Они передают и получают информацию из других отделов, анализирует ее, преобразуя в необходимую отчетность, как внутреннюю, так и внешнюю.

*Бухгалтерия* собирает всю документарную первичную информацию и на основании ее ведет бухгалтерский учет хозяйственно-финансовой деятельности на предприятии.

*Обслуживающий персонал* занимается обслуживанием читателей, помещений и зданий, техники, а также сопровождением торговых или производственных процессов.

*IT отдел* обеспечивает техническую поддержку библиотеки, обслуживание серверов, компьютеров и оргтехники, установку и поддержку нормальной работы операционных систем.

1. **Расширенное техническое задание**

# Введение

# Сокращение слов

Термины, которые имеют лёгкий смысл для восприятия, могут в тексте сокращаться. К ним относятся следующие термины, представленные в таблице 1.

Таблица 1 – Термины и их сокращения

|  |  |
| --- | --- |
| Термин | Сокращение |
| Автоматизированная система - это организационно-техническая система, обеспечивающая выработку решений на основе автоматизации информационных процессов в различных сферах деятельности | АС |
| Программное обеспечение - программа или множество программ, используемых для управления компьютером | ПО |
| Система Управления Базой Данных - Специализированная программа, предназначенная для организации и ведения базы данных. | СУБД |

# Наименование программы:

Научная библиотека «Science Knowledge»

Данное ПО является частью единой автоматизированной системы по ведению деятельности компании, которая предоставляет книги читателям на определенный срок.

# Область применения

Область применения программного продукта является сфера деятельности научной библиотеки «Science Knowledge».

# Основания для разработки

Основанием для разработки является задание, полученное в рамках дисциплины МДК 05.01 «Проектирование и дизайн информационных систем», выданное преподавателем Мамаевой Екатериной Александровной.

# Назначение разработки

# Функциональное назначение

Для сотрудников научной библиотеки программа предоставляет возможность заполнения информации о сотрудниках, читателях, книгах, а также возможность отслеживания взятия и сдачи книг читателем.

# Эксплуатационное назначение

Программа должна эксплуатироваться как приложение на ПК и предназначено для сотрудников научной библиотеки «Science Knowledge».

# Требования к программе

# Требования к функциональным характеристикам

# Требования к составу выполняемых функций

После запуска программы пользователю должно отобразиться окно для авторизации, куда он должен будет ввести свой логин и пароль, который ранее должен храниться в базе данных.

В системе будет существовать 1 тип пользователей — это сотрудники, поскольку программа разрабатывается для них.

После авторизации должно отобразиться главное меню. В этом меню должны быть кнопки для перехода в следующие разделы: Сотрудники, Читатели, Книги, Профиль, Выдача книги, Сдача книги.

При нажатии на кнопку «Сотрудники» программа проверяет наличие разрешения просмотра сотрудников у данного пользователя. Если такое разрешение есть, открывается окно с таблицей, где отображены все пользователи (сотрудники) системы. Здесь пользователь может изменить информацию о любом пользователе, включая возможность просмотра и редактирования сотрудников, и возможность редактирования книг и читателей, а также добавить нового пользователя. Если разрешения у данного пользователя на просмотр сотрудников отсутствует, то появляется соответствующее сообщение в главном меню.

При нажатии на кнопку «Профиль» отображается окно с информацией о текущем пользователе. Здесь пользователь может поменять некоторую информацию о себе, включая пароль для входа в систему.

При нажатии на кнопку «Читатели» открывается окно с таблицей, где отображаются все читатели библиотеки. Пользователь может просматривать читателей и, если такое разрешение у пользователя есть, добавлять новых читателей и редактировать старых.

При нажатии на кнопку «Книги» открывается окно с таблицей, где отображаются все книги библиотеки. Пользователь может просматривать книги и, если такое разрешение у пользователя есть, добавлять новые книги и редактировать старые.

При нажатии на кнопку «Выдача книги» открывается окно, где пользователь может выбрать читателя и книгу, которую нужно выдать читателю, а также срок сдачи книги читателем и информацию о том, берет ли читатель книгу на дом или в читательский зал.

При нажатии на кнопку «Сдача книги» открывается окно, где пользователь должен ввести ID читателя. Если такого ID не существует, то высвечивается соответствующее сообщение, иначе открывается окно, где показаны все несданные читателем книги. Здесь пользователь может выбрать книгу и пометить её как сданную. Также можно отобразить уже сданные читателем книги.

# Требования к надёжности

Вероятность безотказной работы системы должна составлять не менее 99.99% при условии исправности сети.

# Требования к обеспечению надёжного функционирования программы

В связи с тем, что в базе данных хранятся конфиденциальные данные — базу данных стоит резервировать, то есть сохранять.

Надежное функционирование программы должно быть обеспечено выполнением заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

* организацией бесперебойного питания технических средств;
* использованием лицензионного программного обеспечения;
* выполнения требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

# Требования к информационной и программной совместимости

Состав технических средств:

Компьютер пользователя, включающий в себя:

* + процессор x86 с тактовой частотой, не менее 1 ГГц;
  + оперативную память объемом, не менее 2 Гб;
  + видеокарту, монитор, мышь.

# Требования к информационной и программной совместимости

Программа обмениваются с СУБД сообщениями локально.

# Требования к программной документации

Предварительный состав программной документации:

* техническое задание (включает описание применения);
* программа и методика испытаний;
* руководство системного программиста;
* руководство программиста;
* ведомость эксплуатационных документов.

# Технико-экономические показатели

Программа пригодна для библиотек, которым нужно хранить и обрабатывать информацию о взятых читателями книгах. Скорее всего программа будет использоваться в не больших предприятиях, которым не требуются многофункциональная система.

В связи с тем, что предприятиям работающими с клиентской базой и услугами, приходиться использовать многофункциональные системы, есть вероятность, что распространение ПО подобного типа в будущем будет расширяться. Однако, в случае бесплатного распространения программы, потребность в ней может быть весьма высокой, поскольку в ней имеются все функции, которые потребуются для предприятий малого или же среднего уровня.

# Стадии и этапы разработки

Разработка должна быть проведена в три стадии:

* техническое задание;
* технический проект;
* внедрение программы.

На стадии «Техническое задание» должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии «Технический проект» должен быть выполнен перечисленные ниже этапы работ:

* разработка программы;
* разработка программной документации;
* испытания программы.

На стадии «Внедрение» должен быть выполнен этап разработки «Подготовка и передача программы».

Содержание работ по этапам: На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

* постановка задачи;
* определение требований к программе;
* определение стадий, этапов и сроков разработки программы и документации на нее.
* согласование и утверждение технического задания.

На этапе разработки программы должна быть выполнена работа по программированию и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.101-77.

На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

* разработка, согласование и утверждение порядка и методики испытаний;
* проведение приемо-сдаточных испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на объектах заказчика.

# Порядок контроля и приемки

Приемосдаточные испытания программы должны проводиться согласно разработанной исполнителем и согласованной заказчиком «Программы и методики испытаний».

Ход проведения приемо-сдаточных испытаний заказчик и исполнитель документируют в протоколе испытаний.

На основании протокола испытаний исполнитель совместно с заказчиком подписывают акт приемки-сдачи программы в эксплуатацию.

1. **Схемы бизнес-процессов и их описание**
   1. Функциональный анализ ПО

Методология IDEF0 предназначена для функционального моделирования, то есть моделирования выполнения функций объекта, путем создания описательной графической модели, показывающей, что, как и кем делается в рамках функционирования любого предприятия.

Разработанные IDEF0 модели предназначены для документирования процессов производства, отображения того, какая информация и ресурсы используются на каждом этапе.

На рисунке 2 изображена функциональная модель программного обеспечения для деятельности научной библиотеки.

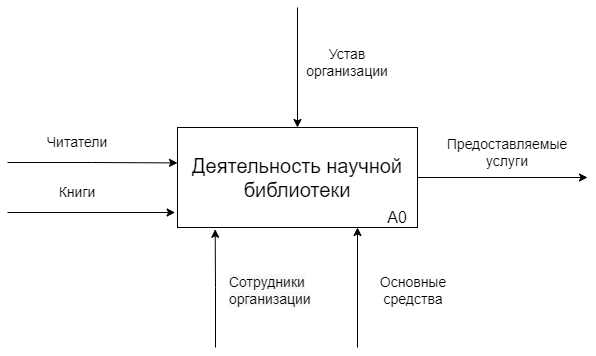


Рисунок 2 – ПО деятельности научной библиотеки

Внешней входящей информацией являются:

* читатели;
* книги.

Управляющим воздействием будет являться:

* устав организации

Механизмом управления будут являться:

* сотрудники организации;
* основные средства;
* база данных.

Внешней исходящей информацией будет являться:

* предоставляемые услуги.

Для разбиения сложного процесса на составляющие подфункции применяется декомпозиция.

Диаграмма декомпозиции, представляет контекстную функцию, разложенную на 2 составляющих более простых функций.

1. Проверка читателя в базе данных.
2. Выдача книги.

На рисунке 3 показана декомпозиция основного блока для выдачи книги.

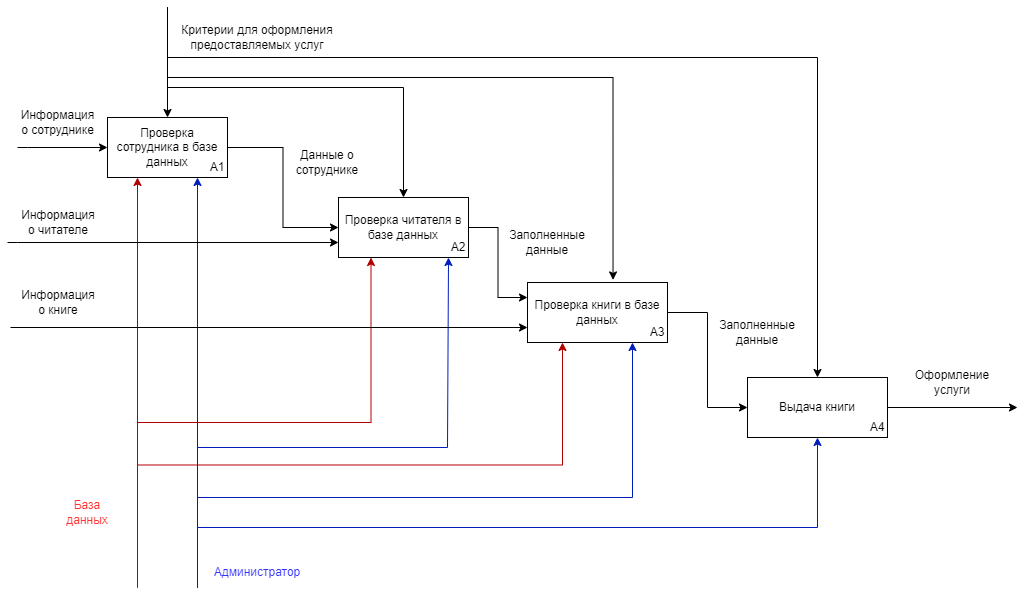


Рисунок 3 - IDEF0 Декомпозиция ПО для деятельности научной библиотеки

**Декомпозиция блока «Проверка читателя в базе данных»**

Данный блок декомпозируется на 2 функции:

1. Поиск читателя в базе данных.
2. Добавление читателя в базу данных.

Для определения данных, в соответствии с которыми будет создан новый читатель в базе, необходимо проверять существует ли такой читатель в базе и передать данные о нём далее.

На рисунке 4 представлена декомпозиция блока «Проверка читателя в базе данных».

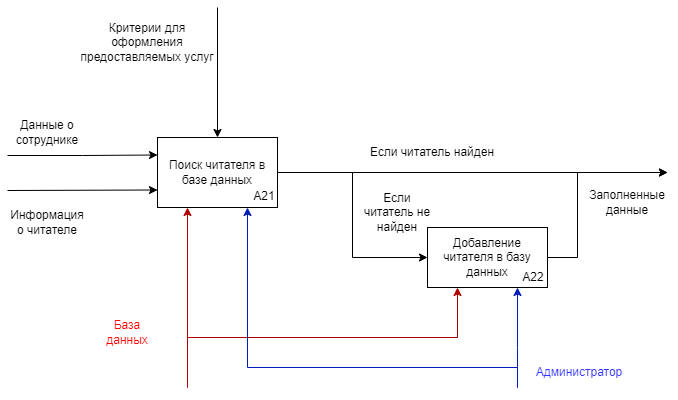


Рисунок 4 - IDEF0 Декомпозиция блока «Проверка читателя в базе данных»

Система, в свою очередь, проверяет введённые данные читателя, определяет, заполнены ли все обязательные поля.

**Декомпозиция блока «Выдача книги»**

Данный блок состоит из 4 функций:

1. Ввод даты выдачи книги.
2. Ввод планируемой даты сдачи книги.
3. Выбор выдачи книги для чтения в зале или выдачи на дом.
4. Оформление услуги.

Для определения данных, в соответствии с которыми будет сформирована услуга, необходимо их получить от сотрудника, преобразовать в данные программы и передать далее.

Декомпозиция блока «Оформление услуги» представлена на рисунке 5.

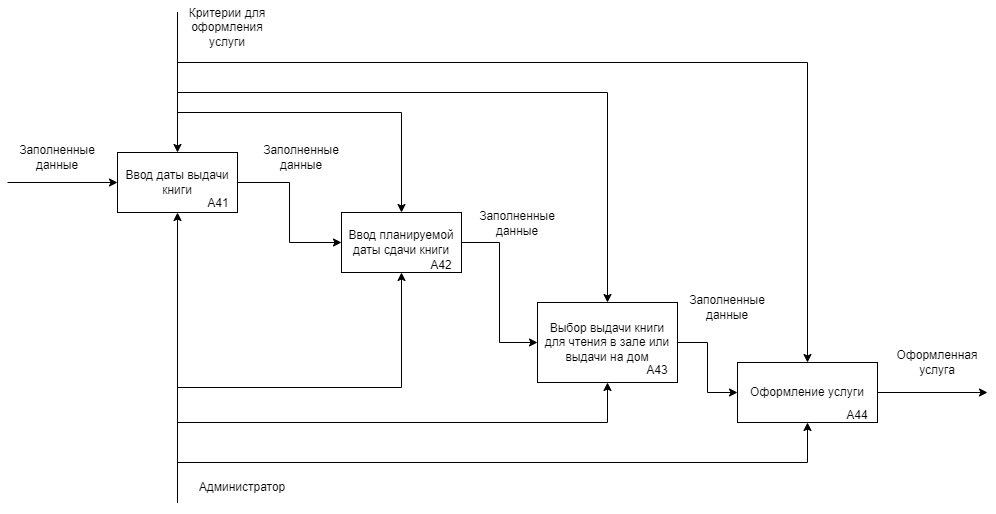


Рисунок 5 - IDEF0 Декомпозиция блока «Оформление услуги»

Система, в свою очередь, определяет, заполнены ли все обязательные поля и заносит информацию о выданной книге в базу данных.

1. 1. Структурный анализ системы

Функциональная модель DFD позволяет описать потоки объектов между узлами обработки, а также источники и приемники объектов и места их хранения.

В DFD используются следующие типы элементов – внешние сущности (для представления источников и получателей данных, обрабатываемых в системе) и хранилища данных (для моделирования любых средств, позволяющих хранить данные в течение произвольного промежутка времени для их последующего применения по мере необходимости).

Верхний уровень иерархии приведен на рисунке 7, где отображена контекстная диаграмма, представляющая всю программу одним функциональным блоком, взаимодействующим с набором внешних сущностей.

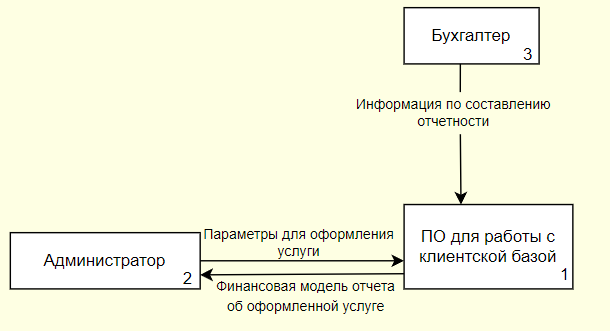


Рисунок 7 - DFD Формирование отчетности по услугам ПО

Внешними сущностями контекстной диаграммы являются:

* администратор;
* бухгалтер.

После составления контекстной диаграммы DFD строится диаграмма декомпозиции, представляющая контекстный блок разложением на набор составляющих его более простых блоков. В данном случае выполняется декомпозиция контекстной диаграммы на 3 блока:

1. Проверка клиента в базе.
2. Оформление услуги.
3. Подсчёт общей стоимости заказа.

Основной уровень декомпозиции ПО представлен на рисунке 8.

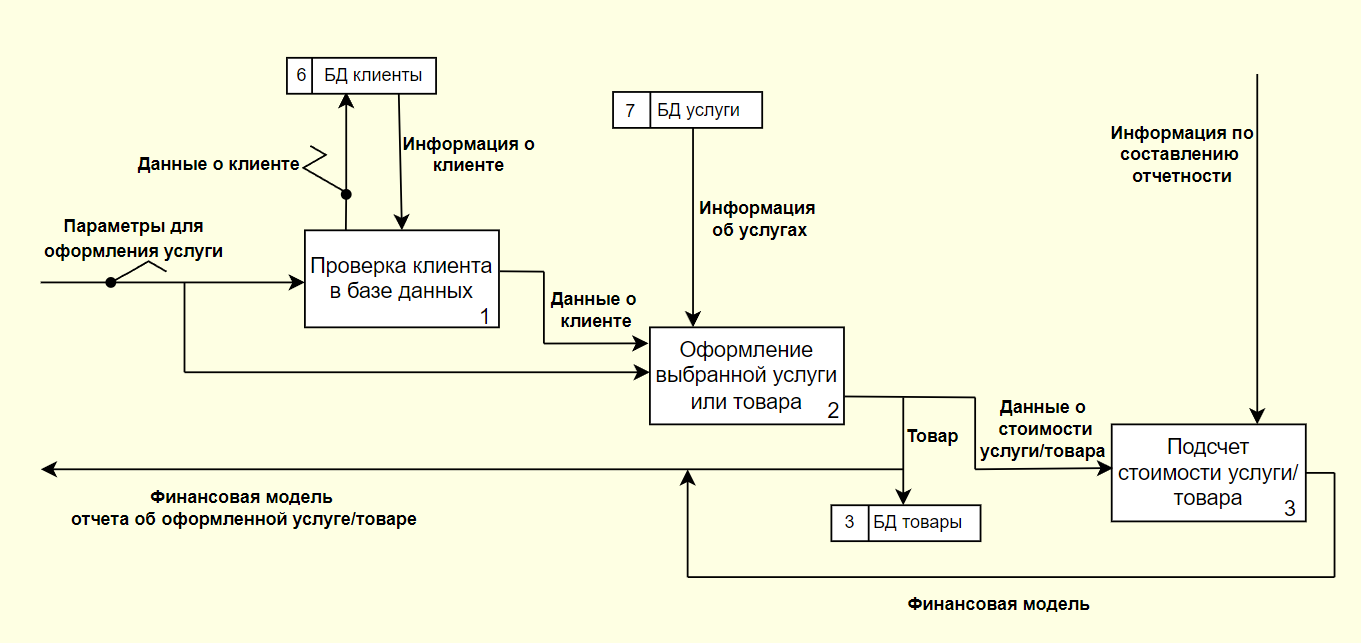


Рисунок 8 - Уровень декомпозиции модели DFD

Хранилищами данных в уровне декомпозиции диаграммы объявляются:

* база данных клиенты;
* база данных услуги;
* база данных товары.

На уровне декомпозиции выполняются следующие процессы:

* проверка клиента в базе;
* оформление выбранной услуги/товара;
* подсчёт общей стоимости услуги/товара.
  1. Логическая модель ПО

Логическая модель данных является визуальным графическим представлением структур данных, их атрибутов и связей. Логическая модель представляет данные таким образом, чтобы они легко воспринимались.

Основными компонентами логической модели являются:

* сущности;
* атрибуты сущности;
* связи между сущностями.

Для описания мы используем нотацию IDEF1X, которая представляет структуру данных, описывающего предприятия. Методология IDEF1X – один из подходов к моделированию данных, основанный на концепции "сущность – связь".

Сущность в методологии IDEF1X является не зависимой, если каждый экземпляр сущности может быть однозначно идентифицирован без определения его отношений с другими сущностями, а если однозначная идентификация экземпляра сущности зависит от его отношения к другой сущности, то сущность такая называется зависимой.

В программе существует 4 основные сущности:

1. Сотрудники.
2. Читатели.
3. Книги.
4. История выдачи книг.

Каждая сущность в базе является зависимой и имеет связь. Имя каждой связи между двумя данными сущностями является уникальным. Логическая модель будет иметь следующий вид связи между сущностями:

* «Книги» – «История выдачи книг»: один ко многим, одна книга может выдаваться много раз;
* «Читатели» – «История выдачи книг»: один ко многим, один читатель может взять много книг;
* «Сотрудники» – «История выдачи книг»: один ко многим, один сотрудник может выдать много книг;

Логическая модель по нотации IDEF1X представлена на рисунке 9.

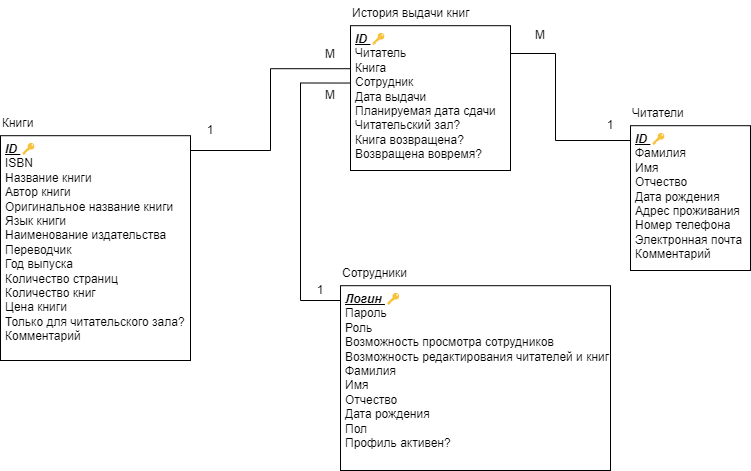


Рисунок 9 – Логическая модель программного обеспечения

Набор сущностей с их свойствами, которые представляются в базе данных представлены в таблице 2.

Таблица 2 – Описание сущностей и их свойств

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сущность | Свойство | Тип | Уникальность | Обязательность заполнения | Ограничения |
| Пользователи | Логин | текстовый | Уникальный | обязательное | PK, NOT NULL |
| Пароль | текстовый |  | обязательное | NOT NULL |
| Роль | текстовый |  | обязательное | NOT NULL |
| Возможность просмотра сотрудников | логический |  | обязательное | NOT NULL |
| Возможность редактирования читателей и книг | логический |  | обязательное | NOT NULL |
| Фамилия | текстовый |  | обязательное | NOT NULL |
| Имя | текстовый |  | обязательное | NOT NULL |
| Отчество | текстовый |  | необязательное |  |
| Дата рождения | текстовый |  | обязательное | NOT NULL |
| Пол | текстовый |  | обязательное | NOT NULL |
| Профиль активен? | логический |  | обязательное | NOT NULL |
| Читатели | ID | числовой | Уникальный | обязательное | PK, NOT NULL |
| Фамилия | текстовый |  | обязательное | NOT NULL |
| Имя | текстовый |  | обязательное | NOT NULL |
| Отчество | текстовый |  | необязательное |  |
| Дата рождения | текстовый |  | обязательное | NOT NULL |
| Адрес проживания | текстовый |  | обязательное | NOT NULL |
| Номер телефона | текстовый |  | необязательное |  |
| Электронная почта | текстовый |  | необязательное |  |
| Комментарий | текстовый |  | необязательное |  |
| Книги | ID | числовой |  | обязательное | FK, NOT NULL |
|  | ISBN | числовой |  | обязательное | FK, NOT NULL |
| Вид тренировки | Название книги | числовой | Уникальный | обязательное | PK, NOT NULL |
| Автор | строчный |  | обязательное | NOT NULL |
| Услуги | Оригинальное название книги | числовой | Уникальный | обязательное | PK, NOT NULL |
| Язык книги | строчный |  | обязательное | NOT NULL |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Абонемент | Код | числовой | Уникальный | обязательное | PK, NOT NULL |
| Вид тренировки | строчный |  | обязательное | NOT NULL |
| Цена | числовой |  | обязательное | NOT NULL |
| Количество посещений | числовой |  | обязательное | NOT NULL |
| Количество дней | числовой |  | обязательное | NOT NULL |
| Вид услуги | строчный |  | обязательное | NOT NULL |
| Продажа абонемента | Номер | числовой | Уникальный | обязательное | PK, NOT NULL |
| Дата начала | числовой |  | обязательное | NOT NULL |
| Дата окончания | числовой |  | обязательное | NOT NULL |
| Код абонемента | числовой |  | обязательное | FK, NOT NULL |
| Код клиента | числовой |  | обязательное | FK, NOT NULL |
| Учет посещений | Номер | числовой | Уникальный | обязательное | PK, NOT NULL |
| Номер карты | числовой |  | обязательное | NOT NULL |
| Номер записи | числовой |  | обязательное | NOT NULL |

* 1. Физическая модель ПО

Физическая модель базы данных содержит все детали, необходимые конкретной СУБД для создания базы: наименования таблиц и столбцов, типы данных, определения первичных и внешних ключей.

Физическая модель строится на основе логической с учетом ограничений, накладываемых возможностями выбранной СУБД.

На рисунке 10 представлена физическая модель программы, которая описывает 10 таблиц, которые между собой имеют связи.

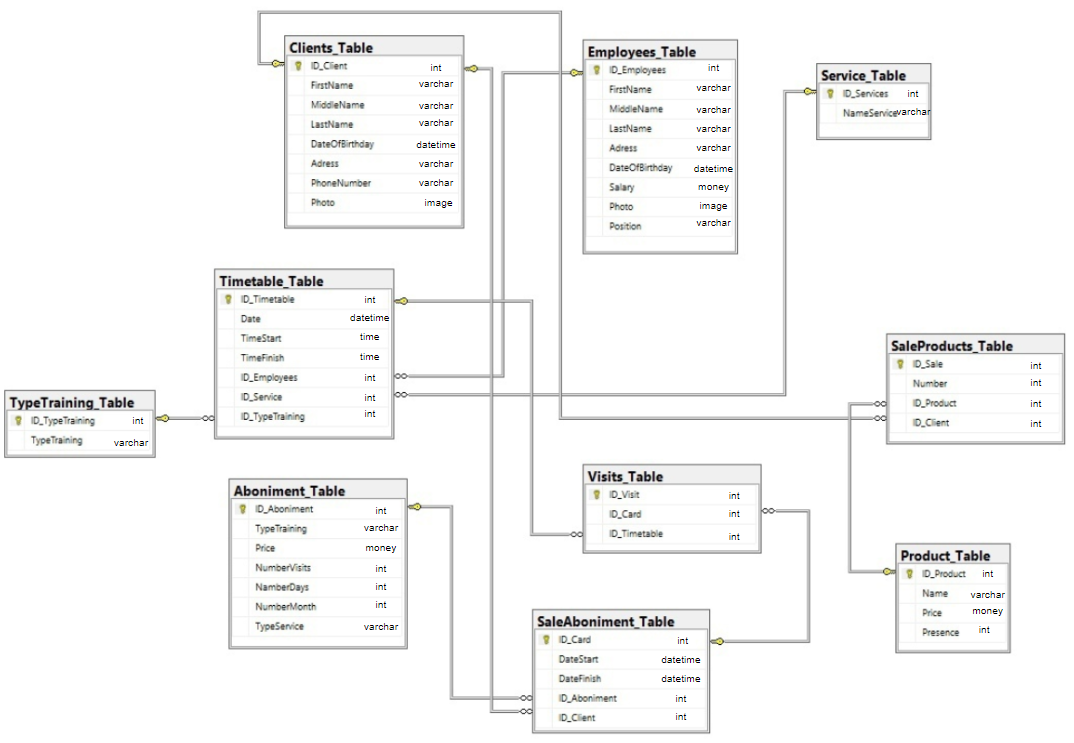


Рисунок 10 – Физическая модель программного обеспечения

Физическая модель имеет описание всех атрибутов, их типов данных, и их размер, которое представлено в таблице 3.

Таблица 3 – Описание полей созданных таблиц

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица | Атрибут | Тип | Размер |
| Clients | Id client | Int | 50 |
| FirstName | Varchar | 255 |
| MiddleName | Varchar | 255 |
| LastName | Varchar | 255 |
| DataOfBirthday | Datetime | 16 |
| Adress | Varchar | 255 |
| PhoneNumber | Varchar | 255 |
| Timetable | Id Timetable | Int | 50 |
| Date | Datetime | 16 |
| TimeStart | Time | 16 |
| TimeFinish | Time | 16 |
| Id Employees | Int | 50 |
| Id Service | Int | 50 |
|  | ID TypeTraning | Int | 50 |
| Product | Id Product | Int | 50 |
| Name | Varchar | 255 |
| Price | Money | 50 |
| Presence | Int | 50 |
| SaleProducts | Id Sale | Int | 50 |
| Number | Int | 50 |
| Id Product | Int | 50 |
| Id Client | Int | 50 |
| Aboniment | Id Aboniment | Int | 50 |
| TypeTraining | Varchar | 255 |
| Price | Money | 50 |
| NumberVisits | Int | 50 |
| NumberDays | Int | 50 |
| NumberMonth | Int | 50 |
| TypeService | Varchar | 255 |
| SaleAboniment | Id Card | Int | 50 |
| DateStart | Datetime | 16 |
| DateFinish | Datetime | 16 |
| Id Aboniment | Int | 50 |
| Id Client | Int | 50 |
| Visits | Id Visit | Int | 50 |
| Id Card | Int | 50 |
| Id Timetable | Int | 50 |
| Service | Id Services | Int | 50 |
| NameService | Varchar | 255 |
| TypeTrainig | Id TypeTraining | Int | 50 |
| TypeTrainig | Varchar | 255 |
| Employees | Id Employees | Int | 50 |
| FirstName | Varchar | 255 |
| MiddleName | Varchar | 255 |
| LastName | Varchar | 255 |
| Adress | Varchar | 255 |
| DateOfBirthday | Datetime | 16 |
| Salary | Money | 50 |
| Photo | Image | 25 |
| Position | Varchar | 255 |

1. **Проектирование ПО**
   1. Формирование диаграммы Use Case

Диаграмма «Use Case» - диаграмма вариантов использования, которая отражает отношения между актерами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать программу на концептуальном уровне.

Она состоит из графической диаграммы, описывающей действующие лица, прецеденты (то есть варианты использования), спецификации.

Диаграмма вариантов использования по программному обеспечению представлена на рисунке 11.

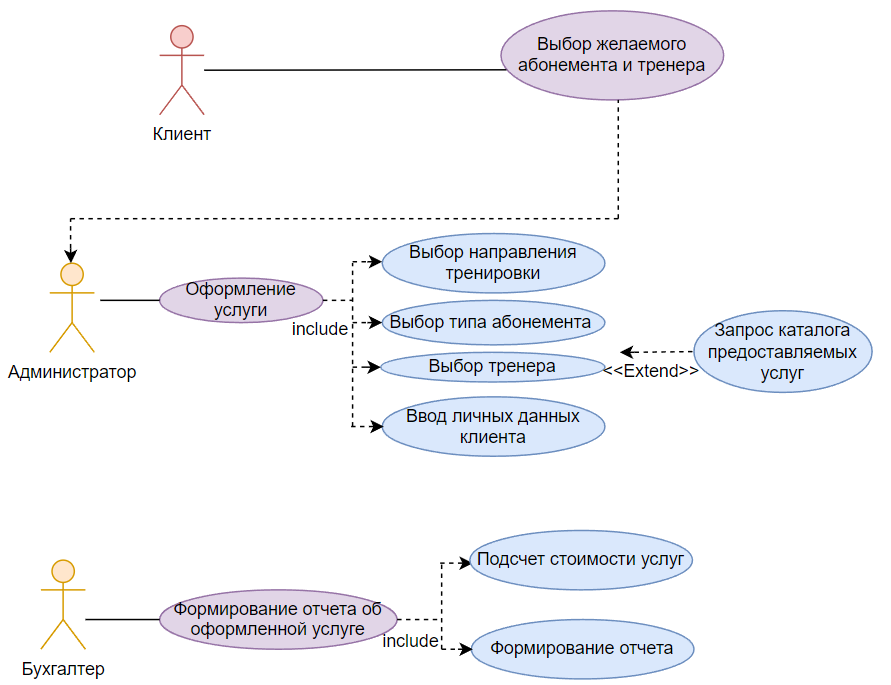


Рисунок 11 – Диаграмма Use Case для оформления услуг

* 1. Формирование схемы алгоритма функционирования АРМ ИС

Линейность алгоритма заключается в том, что все действия в основном модуле программы происходят последовательно. Ветвящийся алгоритм необходим для того, чтобы выбирать необходимые данные из файлов. Это значит, что при нажатии определённой кнопки следует процесс, который был описан в событии по нажатию каждой кнопки.

Схема алгоритма функционирования АРМ ИС бизнес-процесса представлена на рисунка 12–13.

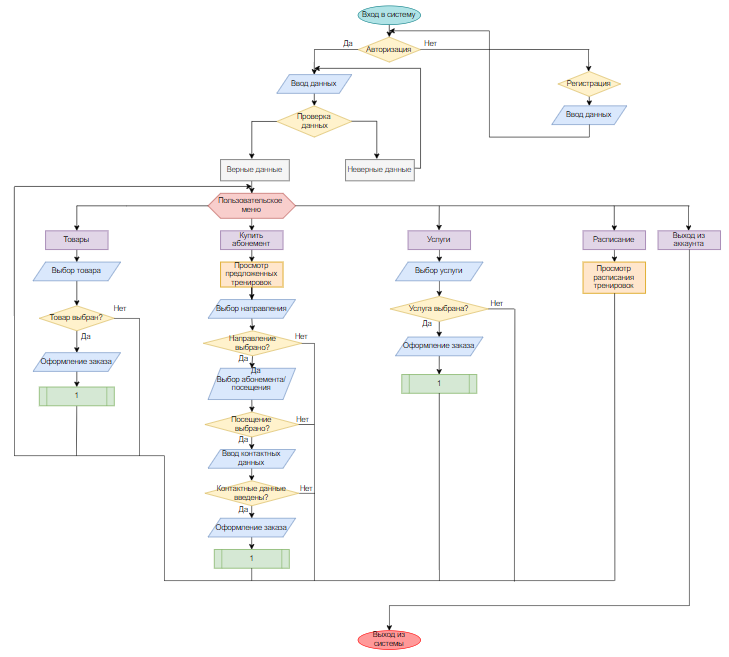


Рисунок 12 – Схема алгоритма функционирования АРМ ИС

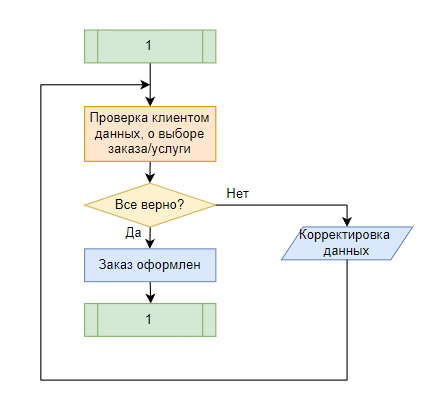


Рисунок 13 – Схема алгоритма функционирования АРМ ИС

* 1. Экранные формы

Программное обеспечение включает в себя 3 рабочих экранных-форм, которые представляю основную работу программы и позволяют выполнять действие, связанные с формированием заказа.

Первая экранная форма – форма авторизации, которая требует предъявить свой уникальный логин и пароль, чтобы программа распознала пользователя. Экранная форма представлена на рисунке 14. Исходя из результата работы на форме авторизации, может появиться экранная форма предупреждения.

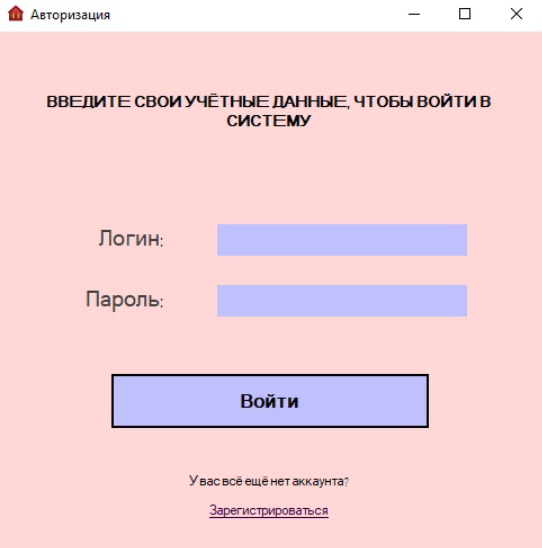


Рисунок 14 – Экранная форма при авторизации

Если пользователь вводит неверные данные, которых нет в базе данных, высвечивается диалоговое окно с ошибкой, если же данные ведены верно, и они есть в базе данных появляется диалоговое окно с успешным входом в систему. Диалоговые окна представлены на рисунках 15, 16 соответственно.

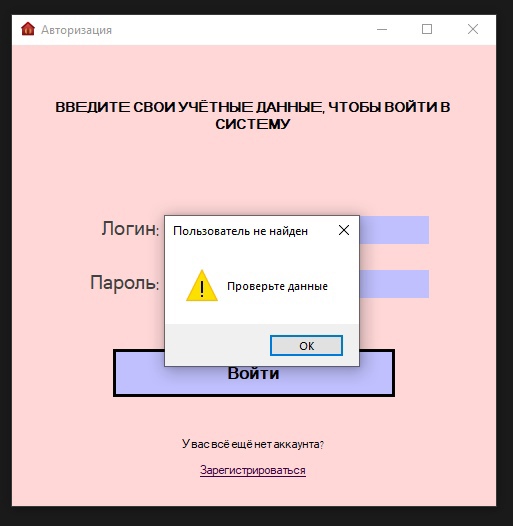


Рисунок 15 – Диалоговое окно с просьбой о проверке данных пользователя

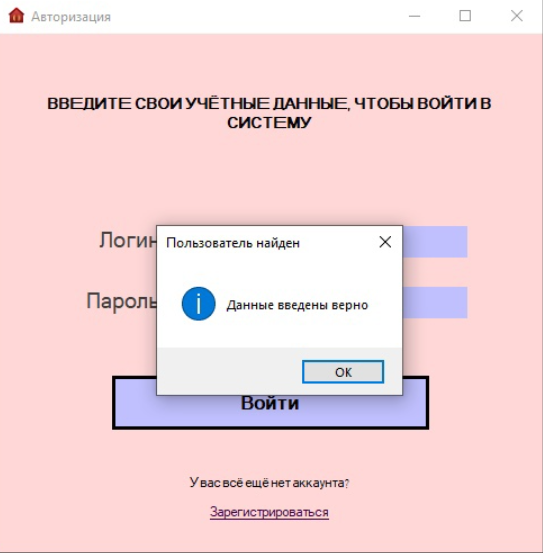


Рисунок 16 – Диалоговое окно об успешно введенных данных

Вторая экранная форма – форма регистрации, которая позволяет зарегистрировать нового пользователя в системе, форма представлена на рисунке 17.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 17 –Экранная форма при регистрации

Третья экранная форма – форма главного меню. Форма имеет 3 раздела:

1. Услуги.
2. Абонементы.
3. Товары.

В зависимости от выбора раздела на главном меню, будут открываться новые экранные формы в соответствии с выбранным пунктом.

На рисунке 18 представлена экранная форма главного меню.

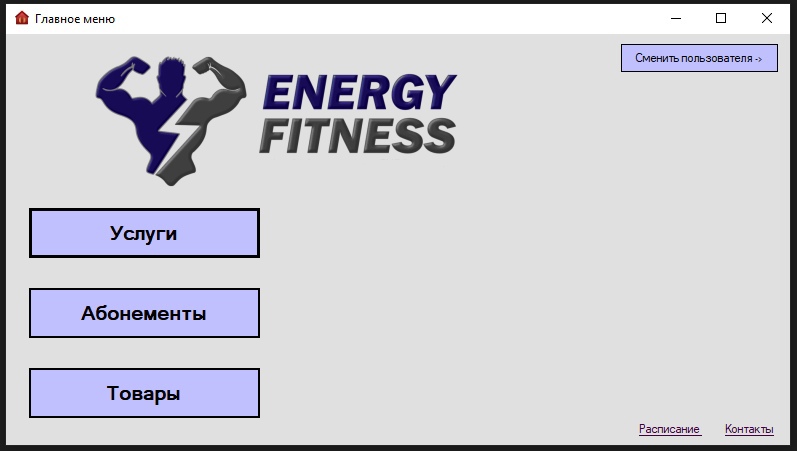


Рисунок 18 – Экранная форма главного меню

Если нажать на кнопку «Абонементы» появляется экранная форма с разделами о выборе срока действия абонемента, направления услуги и выбора специалиста. После заполнения всех полей появляется итоговая стоимость. Экранная форма «Абонементы» представлена на рисунке 19.

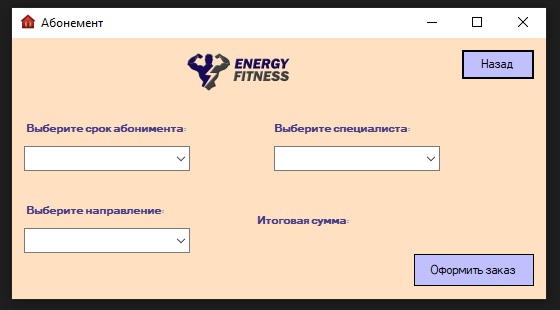


Рисунок 19 – Экранная форма «Абонементы»

Далее после заполнения всех данных на форме и нажатия кнопки «Оформить заказ» появляется сформированный чек по оформленной услуге. На рисунке 20 представлена форма чека.

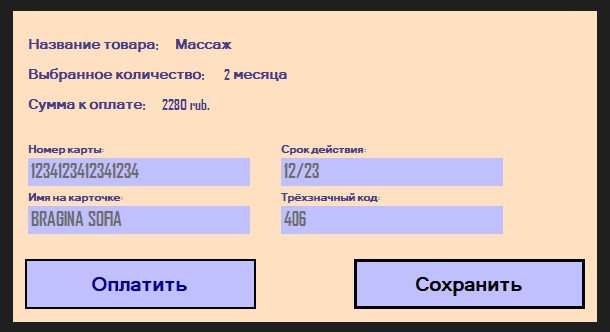


Рисунок 20 – Форма чека

По нажатию кнопки «Сохранить» чек сохраняется в формате .txt на компьютере. На рисунке 21 представлен сохраненный чек об оплате услуги.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 21 – Сохраненный чек об оплате

Также пользователь может приобрести товар в спортивном клубе. Перейдя с главной формы по кнопке «Товары», появится экранная форма с товарами, имеющимися в спортивном клубе. Пользователь может выбрать товар и отметить в каком количестве он хочет его приобрести. На рисунке 19 представлена экранная форма «Товары». После оформления заказа появиться диалоговое окно, представленное на рисунке 22, об успешном оформлении заказа.

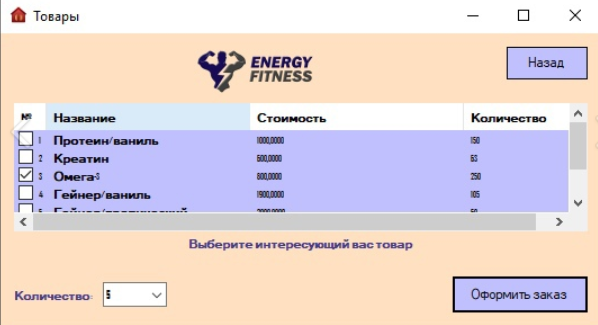


Рисунок 22 – Экранная форма «Товары»

После оформления заказа аналогичным образом, как при работе с формой «Абонементы» появляется форма с заполнением чека, который после заполнения всех данных тоже можно сохрнаить в формает .txt.

На рисунке 23 представлена экранная форма «Услуги».

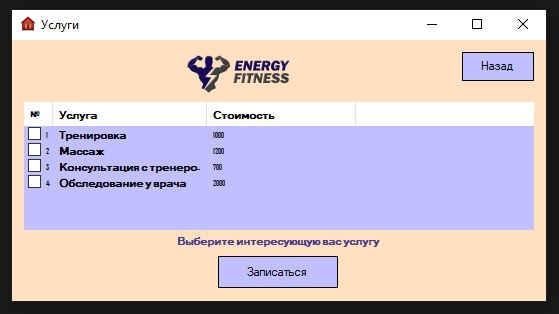


Рисунок 23 – Экранная форма «Услуги»

После записи услуги также формируется чек об успешном оформлении оплаты.

**ЗАКЛЮЧЕНИЕ**

В результате исследования был поставлен и проведён анализ поставленных проблем, с дальнейшим решением их, а также усвоена работа аналогичных систем. Благодаря чему, был решён список поставленных проблем, а также выполнены задачи.

При разработке были приобретены навыки, которые были применены при разработке технического задания на создание автоматизированного программного обеспечения для ведения клиентской базы, товаров и услуг, предоставляемых спортивным клубом «Fitness Energy», а также при формирование общей модели программного обеспечения.

В дальнейшем планируется усовершенствование программы, поскольку выполнена она не совсем идеальна. Не удалось реализовать вывод статической отчетности по процессам, происходящим в программе. Также планируется реализация еще нескольких типов пользователей, например администратора, с другим интерфейсом и особым набором функциональных возможностей, в отличие от клиентов. Еще один вид пользователей, который хотелось бы добавить – это тренерский состав, чтобы каждый тренер мог заранее просматривать свою загруженность на день, по записям клиентов на тренировки и т.п.

Разработанное техническое задание и программное обеспечение может облегчить и ускорить качество работы сотрудникам, и компаниям, которые работают в сфере продажи товаров и услуг.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

* 1. ГОСТ Р 19.201-78. Единая система программной документации. Техническое задание. Требования к содержанию и оформлению = Unified system of software documentation. Technical specification. requirements for content and design : национальный стандарт Российской Федерации : издание официальное : утвержден и введён в действие постановлением государственного комитета СССР по стандартам от 18 декабря 1978 г. N 3351-ст : введён впервые: дата введения 1980.01.01 / подготовлен АО «Кодекс». – Москва : Издательство стандартов, 1980. – III, 4 с. ; 29 см. – Текст: непосредственный.
  2. RuGost – разработка документации по ГОСТ 34, 19, РД-50 : [сайт]. – Москва, 2012 – . – URL: http://www.rugost.com (дата обращения: 10.23.2020). – Текст: электронный
  3. Документирование в разработке ПО : [сайт]. – Москва, 2006 – . – URL: https://habr.com/ru/post/215837 (дата обращения: 18.11.2020). – Текст: электронный
  4. ГОСТ Р 34.602-89. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы = Information technology. Set of standards for automated systems. Technical directions for automated system making   : национальный стандарт Российской Федерации : издание официальное : утвержден и введён в действие постановлением государственного комитета СССР по стандартам от 18 декабря 1989 г. N 3231-ст : введён впервые: дата введения 1990.01.01 / подготовлен АО «Кодекс». – Москва : Издательство стандартов, 1990. – III, 26 с. ; 29 см. – Текст: непосредственный.
  5. Документация программного обеспечения : [сайт]. – Екатеринбург, 2017 – . – URL: https://unetway.com/tutorial/dokumentacia-programmnogo-obespecenia (дата обращения: 28.11.2020). – Текст: электронный.
  6. Методы и средства инженерии программного обеспечения : [сайт]. – Москва, 2003 – . – URL: https://intuit.ru/studies/courses/2190/237 (дата обращения: 11.11.2020). – Текст: электронный.