

**КУРСОВОЙ ПРОЕКТ**

**по дисциплине: МДК 01.01 Разработка программных модулей**

**Тема: Разработка системы классов для базы данных обменного пункта.**

**Специальность 09.02.07 «Информационные системы и программирование»**

|  |  |  |
| --- | --- | --- |
| **Выполнил студент(ка) группы 309ИС-22** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **П.Д. Донских** |
| **Руководитель** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Л.Б. Гусятинер** |

**Москва 2024**



**УТВЕРЖДАЮ**

**Зам. директора КМПО**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_С.Ф. Гасанов**

**«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.**

**ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ**

**по дисциплине: МДК 01.01 Разработка программных модулей**

**Специальность 09.02.07 «Информационные системы и программирование»**

**Студент группы 309ИС-22 Павел Донских**

**ТЕМА: «Разработка системы классов для базы данных обменного пункта»**

Дата выдачи задания «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г.

Срок сдачи проекта «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2024 г.

**Москва 2024**

**ОТЗЫВ**

**научного руководителя на курсовой проект**

**Студента(ки**)

(фамилия, имя и отчество)

**группа**  **курс**  3

**Дисциплина**  МДК 01.01 РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ

**Тема работы**

Отмечаются следующие моменты: актуальность темы исследования; степень разработанности проблемы, наиболее интересно исследованные вопросы. Оценивается степень самостоятельности и творчества студента; уровень его теоретической подготовки; умение анализировать научные материалы, делать практические выводы. Знание основных концепций, научной и специальной литературы по избранной теме.

**Научный руководитель**

(Уч. степень, уч. звание, должность, ФИО)

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 5](#_Toc184680255)

[Глава 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 8](#_Toc184680256)

[1.1 Информационное обеспечение задачи 8](#_Toc184680257)

[1.2 Обзор и анализ существующих программных решений 9](#_Toc184680258)

[1.3 Постановка задачи 12](#_Toc184680259)

[1.3.1 Основные задачи и функции. 12](#_Toc184680260)

[1.3.2 Структура входной и выходной информаций 12](#_Toc184680261)

[Глава 2. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА МОДУЛЯ 13](#_Toc184680262)

[2.1 Выбор инструментов 13](#_Toc184680263)

[2.2 Проектирование сценария 14](#_Toc184680264)

[2.3 Таблицы структур 16](#_Toc184680265)

[2.4 Описание главного модуля 18](#_Toc184680266)

[2.5 Описание модулей 22](#_Toc184680267)

[Глава 3. ТЕСТИРОВАНИЕ И ОТЛАДКА ПРОГРАММНОГО МОДУЛЯ 24](#_Toc184680268)

[ЗАКЛЮЧЕНИЕ 25](#_Toc184680269)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 26](#_Toc184680270)

[Приложение 1. Скрины работы программы для сотрудника 28](#_Toc184680271)

[Приложение 2. Скрины работы программы для администратора 29](#_Toc184680272)

[Приложение 3. Таблица план тестирования. 30](#_Toc184680273)

# ВВЕДЕНИЕ

Первая признанная международная система валютного обмена — «Золотой стандарт», установленный в 1944 году на Бреттон-Вудской конференции. Этот стандарт действовал до начала 70-х годов и способствовал созданию Международного Валютного Фонда, который контролировал валютные курсы и предлагал консультации. На Ямайской конференции 1976 года произошли изменения, которые привели к переходу от фиксированных к плавающим обменным курсам, и валютный рынок получил название «Форекс».

Обмен валют играет важную роль в финансовой системе каждой страны, обеспечивая возможность конвертации фиатной валюты как для физических лиц, так и для юридических лиц. С каждым днем интерес к операциям конвертации только растет, что связано с увеличением международной торговли и развитием туризма. Кроме того, с ростом интернет-транзакций значение онлайн-обменников становится все более актуальным.

Использование автоматизированных систем не только повышает общую эффективность, но и обеспечивает высокую точность проводимых обменов. Автоматизация процессов помогает снизить ошибки, возникающие при ручной обработке данных. Внедрение таких систем позволяет обменным пунктам быстрее реагировать на изменения на рынке и предлагать клиентам безопасные способы проведения операций.

Объектом исследования является система, которая включает в себя все процессы, связанные с обменом валют, его анализом и отчетностью.

Предметом исследования является разработка программного модуля, который будет автоматизировать процессы учета обменов и формирования отчетов.

Актуальность разработки обоснована рядом факторов, касающихся современного бизнеса и управления трудозатратами.

1. Увеличение конкуренции на рынке

* В условиях растущей конкуренции предприятия стремятся оптимизировать свои процессы для повышения эффективности и снижения затрат.
* Необходимость адаптации: компании должны адаптироваться к изменениям, чтобы оставаться конкурентоспособными.

1. Автоматизация процессов

* Снижение ручного труда: ручной учет обменов часто приводит к ошибкам, задержкам и увеличению затрат времени.
* Упрощение работы: Автоматизированные системы упрощают работу менеджеров и сотрудников, позволяя им сосредоточиться на более важных задачах.

1. Повышение точности учета

* Точность данных: автоматизированные системы обеспечивают более точный учет обменов, что способствует правильному расчету финансовых показателей.
* Снижение ошибок: исключение человеческого фактора уменьшает вероятность ошибок в данных, что критически важно для финансовой отчетности.

1. Отчетность

* Автоматизированные системы могут генерировать отчеты, необходимые для соблюдения законодательных требований, что упрощает процесс отчетности.

Цель работы: создание функционального программного модуля, который будет обеспечивать:

Упрощение процесса записи истории обменов в файл

Представление инструментов для генерации отчетов и анализа обменов за текущий год.

Для достижения цели необходимо:

* Проанализировать предметную область;
* Проанализировать готовое решение;
* Подготовить техническое задание;
* Обосновать выбор инструментов и средств разработки;
* Подготовить план тестирования;
* Описать реализацию технического задания;
* Выполнить тестирование.

**Глава 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

* 1. Информационное обеспечение задачи

Обменный пункт — это учреждение предназначено для предоставления услуг по обмену валют. Основные клиенты обменного пункта — это туристы, бизнесмены и местные жители, нуждающиеся в наличной валюте для повседневных нужд. Назначение разрабатываемого программного продукта заключается в автоматизации процессов учета операций обмена валют, управления данными о проведенных сделках и формирования отчетности. Программное обеспечение должно обеспечить эффективное хранение данных о проведенных операциях, включая даты обмена, суммы и валютные пары, а также возможность генерации отчетов по месяцам и за текущий год.

Основные функции обменного пункты:

* Установление курсов валют: курсы должны обновляться в реальном времени на основе рыночных данных;
* Обработка заявок на обмен: система должна обеспечивать возможность ввода данных о сделках;
* Учет проведенных операций: все операции должны фиксироваться для дальнейшего анализа;
* Генерация отчетов: система должна формировать отчеты по сделкам за определенные периоды.

Основные термины:

1. Валюта — денежные знаки, используемые в разных странах (например, $ — доллар, ₽ — рубль);
2. Валютная пара – это финансовый инструмент, состоящий из 2-х разных валют. Обозначается краткими кодами валют через дробь, например EUR/USD – это валютная пара евро/американский доллар. Или USD/JPY – валютная пара американский доллар/японская йена;
3. Курс валюты — цена одной валюты, выраженная в другой валюте, устанавливаемая на основе рыночного спроса и предложения;
4. Обменный курс — курс, по которому происходит обмен валют (включает комиссию обменного пункта);
5. Базовая валюта — валюта, которую клиент хочет получить;
6. Котируемая валюта — валюта, за которую клиент получает базовую валюту;
7. Спред — разница цены покупки и продажи;
8. ASK — цена покупки;
9. BID — цена продажи;
10. Транзакция — любая сделка или операция, для совершения которой используется банковский счет, при этом баланс на нем меняется в меньшую или большую сторону.
    1. Обзор и анализ существующих программных решений

|  |  |  |
| --- | --- | --- |
| Таблица 1. Обзор существующих программных решений. | | |
| Название системы | Описание | Минусы системы |
| "Currency Tracker" | Система предоставляет пользователям возможность отслеживать количество операций обмена валюты за заданный период. Функционал включает в себя создание отчетов, которые отображают объем сделок по каждой валютной. | * Ограниченная функциональность: может не поддерживать интеграцию с другими системами, что затрудняет обмен данными. * Сложность в использовании: интерфейс, может быть, неинтуитивным для новых пользователей, что требует времени на обучение. * Ограниченные возможности кастомизации отчетов, что может не удовлетворять специфические потребности бизнеса. |
| "Forex Report Generator" | Программное обеспечение предлагает инструменты для создания детализированных отчетов о сделках на рынке Forex. Пользователи могут анализировать количество сделок и прибыль по каждой валютной паре за выбранные временные интервалы. | * Высокая стоимость лицензии, что может быть неприемлемо для небольших обменных пунктов. * Возможные проблемы с производительностью при обработке больших объемов данных, что может замедлить генерацию отчетов. |
| "Exchange Statistics Generator" | Система предназначена для автоматической генерации отчетов о статистике обменных операций, включая объемы обмена, популярные валюты и временные интервалы. | * Ограниченные возможности фильтрации данных: пользователи могут не иметь возможности детализировать отчеты по специфическим критериям, что затрудняет анализ. * Сложности с интеграцией: система может не поддерживать интеграцию с другими финансовыми инструментами, что ограничивает функциональность. * Необходимость в ручном обновлении данных: пользователи могут сталкиваться с необходимостью вручную обновлять информацию, что увеличивает риск ошибок. |

Ограниченная функциональность и интеграция: многие системы, такие как "Currency Tracker" и "Exchange Statistics Generator", имеют проблемы с интеграцией с другими платформами. Это может затруднить обмен данными и снизить общую эффективность работы обменного пункта.

Сложности в использовании: интерфейсы некоторых программ могут быть неинтуитивными для новых пользователей, что требует дополнительного времени на обучение и адаптацию. Это может стать препятствием для быстрого внедрения системы в рабочие процессы.

Высокая стоимость: программное обеспечение, например, "Forex Report Generator", может иметь высокую стоимость лицензии, что делает его недоступным для небольших обменных пунктов и индивидуальных инвесторов. Это ограничивает выбор доступных решений для малых бизнесов.

Необходимость ручного обновления данных: в системах, таких как "Exchange Statistics Generator", пользователи могут сталкиваться с необходимостью вручную обновлять данные. Это увеличивает риск ошибок и затрудняет процесс генерации отчетов.

Таким образом, при выборе программного решения для генерации отчетов в обменном пункте важно учитывать не только функциональные возможности, но и удобство использования, интеграцию с другими системами.

* 1. Постановка задачи

Разработка программного модуля информационной системы направлена на автоматизацию ключевых бизнес-процессов, улучшения качества отчетности обменного пункта.

* + 1. Основные задачи и функции.

Ввод, удаление, редактирование данных о проведенных операциях обмена и менеджерах;

Учет количества сделок по каждой валютной паре;

Формирование отчетов о количестве операций и суммах обмененных денег за определенные периоды;

Возникает необходимость в компьютерной обработке вводимых данных о проведенных операциях с целью хранения этой информации на носителях и анализа эффективности работы обменного пункта.

* + 1. Структура входной и выходной информаций

Структура входной информации:

* Данные о менеджерах: номер менеджера, ФИО менеджера;
* Данные об обменах: номер обмена, дата обмена, номер менеджера, валюта, сумма в рублях.

Структура выходной информации:

* Готовый документ: текущая дата, количество обменов за каждый месяц, сумма обмененных денег по месяцам за текущий год, место для подписи.

# Глава 2. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА МОДУЛЯ

* 1. Выбор инструментов

При выборе инструмента и языка программирования было проведено сравнение по критериям, которые представлены в таблицах 1 и 3.

Степень важности критерия выбиралась из: низкая, ниже средней, средняя, ниже высокой, высокая.

|  |  |
| --- | --- |
| Таблица 2. Критерии выбора инструмента. | |
| Критерий | Важность критерия |
| Функционал | Ниже высокой |
| Удобство использования | Высокая |
| Скорость разработки | Ниже высокой |

Исходя из данных критериев, я сравнил 2 языка программирования от 0 до 10 баллов за критерий

|  |  |  |
| --- | --- | --- |
| Таблица 3. Оценка языков программирования. | | |
| Критерий/Язык программирования | Python | C |
| Функционал | 10 | 8 |
| Удобство использования | 7 | 9 |
| Скорость разработки | 7 | 7 |
| Итого | 24 | 24 |

|  |  |
| --- | --- |
| Таблица 4. Критерии выбора среды разработки. | |
| Критерий | Важность критерия |
| Простота | Средняя |
| Функционал | Высокая |
| Удобство использования | Высокая |
| Документация на русском языке | Ниже средней |

Исходя из данных критериев, я сравнил 2 среды разработки от 0 до 10 баллов за критерий.

|  |  |  |
| --- | --- | --- |
| Таблица 5. Оценка сред разработки. | | |
| Критерий/среда разработки | CodeBlocks | Microsoft Visual Studio |
| Простота | 7 | 5 |
| Функционал | 10 | 10 |
| Удобство использования | 8 | 6 |
| Документация на русском языке | 6 | 9 |
| Итого | 31 | 30 |

По результатам сравнения была выбрана среда разработки CodeBlocks

CodeBlocks — это интегрированная среда разработки (IDE) с открытым исходным кодом, предназначенная для программирования на языках C, C++ и Fortran. Она предоставляет пользователям удобный интерфейс для написания, компиляции и отладки кода. CodeBlocks поддерживает различные компиляторы и может быть расширена с помощью плагинов, что делает её гибким инструментом для разработчиков.

* 1. Проектирование сценария

В данном разделе приведён пример сценария использования программы сотрудником и администратором (см. рисунок 1).

После запуска программы менеджер должен загрузить файлы менеджеров, операций обменов. Затем он может редактировать данные в файлах или записать новые даные. После этих действий программа выдаст отчет об обменах.

Также, для администратора предусмотрен режим редактирования данных, в который можно получить доступ, используя пароль. В нем администратору доступны: добавление и удаление данных о менеджерах. Ниже представлена схема сценария использования.

Рисунок 1. Сценарий использования

* 1. Таблицы структур

В таблицах структур (см. таблицы 5, 6) представлены две структуры, manager (для хранения информации о менеджере), exchange (для представления данных обменов).

В правом блоке, расположен блок с операциями (функциями, где используются структуры).

|  |  |  |
| --- | --- | --- |
| Таблица 5. Структуры Exchange | |  |
| Элементы | Функции | Описание функции |
| int exchange\_id — идентификатор продажи. | int load\_exchanges(char \*fname, Exchange exchanges[], int limit); | Загружает данные о обменах из файла и сохраняет их в массиве exchanges, возвращая количество загруженных обменов. |
| struct Date date — дата продажи (содержит поля day, month, year). | void exchanges\_print(Exchange exchanges[], int limit); | Печатает список обменов, используя данные из массива exchanges. |
| int manager\_id — идентификатор менеджера. | void save\_new\_exchange(char \*fname); | Сохраняет новый обмен в файл, запрашивая у пользователя необходимые данные. |
| char currency — валюта. | void edit\_exchange(char \*fname); | Позволяет редактировать существующий обмен, запрашивая у пользователя новый ввод данных. |
| int price — цена обмена. | void do\_query1(Exchange exchanges[], int limit, const char \*fname); | Выполняет запрос по обменам, анализируя данные и записывая результаты в файл. |
|  | void input\_exchange(Exchange \*exchange, char \*fname); | Функция для ввода данных о новом обмене. |
|  | int check\_exchange\_id\_exists(char \*fname, int exchange\_id); | Функция для проверки существования ID обмена в файле. |
|  | void exchange\_to\_s(char s[100], Exchange exchange); | Преобразует объект Exchange в строку, включая дату, и записывает результат в переданную строку. |

|  |  |  |
| --- | --- | --- |
| Таблица 6. Структуры Manager | |  |
| Элементы | Функции | Описание функции |
| int manager\_id — идентификатор продажи. | int load\_managers(char \*fname, Manager managers[], int limit); | Загружает данные о менеджерах из файла и сохраняет их в массиве managers, возвращая количество загруженных менеджеров. |
| struct Date date — дата продажи (содержит поля day, month, year). | void managers\_print(Manager managers[], int limit); | Печатает список менеджеров, используя данные из массива managers. |
| int manager\_id — идентификатор менеджера. | void manage\_managers(Manager managers[], int \*count); | Позволяет управлять менеджерами, предлагая пользователю добавить или удалить менеджера. |
| char currency — валюта. | void add\_manager(Manager managers[], int \*count); | Добавляет нового менеджера в массив managers, запрашивая у пользователя необходимые данные. |
| int price — цена продажи. | void remove\_manager(Manager managers[], int \*count); | Удаляет менеджера из массива managers по указанному ID, сдвигая оставшиеся элементы. |
|  | void input\_manager(Manager \*manager); | Функция для ввода данных о новом менеджере. |
|  | int check\_manager\_id\_exists(Manager managers[], int count, int manager\_id); | Функция для проверки существования ID менеджера в массиве. |

* 1. Описание главного модуля

Главный модуль представляет собой main.c

К главному модулю подключаются остальные модули, содержащие в себе структуры и функции.

Листинг 1. Код главного модуля

// main.c

#include <stdio.h>

#include <locale.h>

#include <stdlib.h>

#include <Windows.h>

#include "func.h"

#include "exchange.h"

#include "manager.h"

#define AR\_MAN\_LEN 100

#define AR\_EXCHANGE\_LEN 1000

int main() {

setlocale(LC\_ALL, "");

int man\_count = 0; // Количество менеджеров

int exchange\_count = 0; // Количество обменов

Exchange exchanges[AR\_EXCHANGE\_LEN]; // Массив для хранения обменов

Manager managers[AR\_MAN\_LEN]; // Массив для хранения менеджеров

man\_count = load\_managers("managers.txt", managers, AR\_MAN\_LEN); // Загрузить менеджеров

printf("Файл менеджеров загружен в память\n\n");

int item = -1; // Переменная для выбора пункта меню

while (item != 0) {

printf("Главное меню:\n");

printf("0. Выход\n");

printf("1. О программе\n");

printf("2. Я менеджер\n");

printf("3. Я администратор\n");

printf("Выберите действие: ");

scanf("%d", &item); // Ввод выбора пользователя

system("cls"); // Очистка консоли после выбора

switch (item) {

case 0:

printf("Спасибо за использование нашей разработки\n\n");

break;

case 1:

printf("309ИС-22 Донских П. Д.\n");

printf("Генерация отчетов для обменного пункта валют\n\n");

break;

case 2:

manager\_login(managers, man\_count, exchanges, &exchange\_count); // Вход для менеджера

break;

case 3:

admin\_login(managers, &man\_count); // Вход для администратора

break;

default:

printf("Ошибка. Пункты от 0 до 3\n\n");

break;

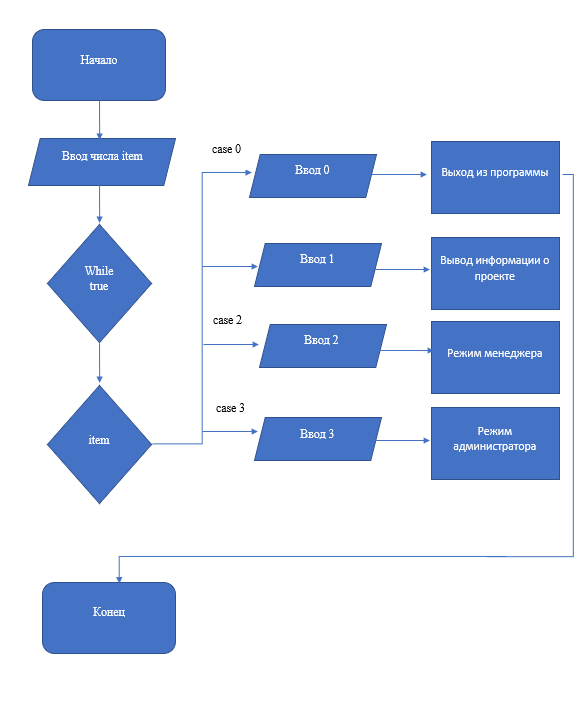
}

}

return 0;

}

Ниже представлена блок-схема главного модуля.

Рисунок 2. Блок-схема главного модуля

* 1. Описание модулей
  2. Главный модуль

Главный модуль представляет собой файл main.c. К главному модулю подключаются остальные модули, содержащие в себе структуры и функции.

Изменить

* 1. Модуль функций

В модуле функций находятся функции, используемые в проекте:

1. void exchange\_to\_s(char s[100], Exchange exchange) — преобразует объект Exchange в строку, включая дату, и записывает результат в переданную строку.
2. void manager\_to\_s(char s[150], Manager manager) — преобразует объект Manager в строку и записывает результат в переданную строку.
3. void show\_menu() — отображает меню с доступными действиями для пользователя.
4. int check\_exchange\_id\_exists(char \*fname, int exchange\_id) —проверяет, существует ли ID обмена в указанном файле.
5. void input\_exchange(Exchange \*exchange, char \*fname) —запрашивает у пользователя данные для нового обмена и заполняет объект Exchange.
6. void save\_new\_exchange(char \*fname) — сохраняет новый обмен в файл, запрашивая у пользователя необходимые данные.
7. void edit\_exchange(char \*fname) — позволяет редактировать существующий обмен, запрашивая у пользователя новый ввод данных.
8. void exchanges\_print(Exchange exchanges[], int limit) — печатает список обменов, используя данные из массива exchanges.
9. int load\_exchanges(char \*fname, Exchange exchanges[], int limit) —загружает данные о обменах из файла и сохраняет их в массиве exchanges, возвращая количество загруженных обменов.
10. int load\_managers(char \*fname, Manager managers[], int limit) — загружает данные о менеджерах из файла и сохраняет их в массиве managers, возвращая количество загруженных менеджеров.
11. void managers\_print(Manager managers[], int limit) — печатает список менеджеров, используя данные из массива managers.
12. void do\_query1(Exchange exchanges[], int limit, const char \*fname) — выполняет запрос по обменам, анализируя данные и записывая результаты в файл.
13. void manage\_managers(Manager managers[], int \*count) —Позволяет управлять менеджерами, предлагая пользователю добавить или удалить менеджера.
14. void add\_manager(Manager managers[], int \*count) —добавляет нового менеджера в массив managers, запрашивая у пользователя необходимые данные.
15. void remove\_manager(Manager managers[], int \*count) — yдаляет менеджера из массива managers по указанному ID, сдвигая оставшиеся элементы.
16. void input\_manager(Manager \*manager) — функция может используется для ввода данных о новом менеджере.
17. int check\_manager\_id\_exists(Manager managers[], int count, int manager\_id) — функция проверяет, существует ли ID менеджера в массиве менеджеров.
18. void admin\_menu(Manager managers[], int \*man\_count) — функция для отображения меню администратора, позволяя управлять менеджерами (добавление, удаление).
19. void manager\_menu(Manager managers[], int man\_count, Exchange exchanges[], int \*exchange\_count) — функция для отображения меню менеджера, позволяя выполнять действия, связанные с обменами.
20. void get\_current\_date(char \*date\_str) — функция для получения текущей даты в строковом формате.

# Глава 3. ТЕСТИРОВАНИЕ И ОТЛАДКА ПРОГРАММНОГО МОДУЛЯ

**Добавить тест-кейсы**

# ЗАКЛЮЧЕНИЕ

В результате выполнения курсового проекта была написана программа "CurrencyReport" для автоматизации процесса отчетности обменного пункта.

В ходе работы были проанализированы предметная область, существующие разработки, посвященные данному направлению, получены практические навыки с работой в CodeBlocks.

Так же планируется продолжить работу над данным проектом с целью расширения возможностей и удобства для пользователей. Планы по доработке представлены ниже.

1. Реализация методов и функций, учитывающих количество оставшихся в наличии валют;
2. Добавить реализацию функций для управления обменами;
3. Обновление курса валют в режиме реального времени;
4. Создание и последующая доработка интерфейса;
5. Создание и последующая доработка баз данных;

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Федеральные законы:

* Федеральный закон "Об информации, информационных технологиях и о защите информации" от 27.07.2006 N 149-ФЗ — <https://www.consultant.ru/document/cons_doc_LAW_61798/>
* Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ — <https://www.consultant.ru/document/cons_doc_LAW_61801/>
* Федеральный закон "О безопасности критической информационной инфраструктуры Российской Федерации" от 26.07.2017 N 187-ФЗ — <https://www.consultant.ru/document/cons_doc_LAW_220885/>

ГОСТы:

* РОССТАНДАРТ — <https://www.gost.ru/portal/gost/home/standarts/catalognational>
* ГОСТ Р ИСО/МЭК 12207 — <https://docs.cntd.ru/document/1200082859>
* ЕСПД — <https://www.swrit.ru/gost-espd.html>

Источники информации об обменных пунктах:

* Кузнецов, А. П. (2019). Обменные пункты: организация и управление. Санкт-Петербург: Издательство "Экономика".
* Сидоров, В. Н. (2021). Анализ валютного рынка. Москва: Издательство "Наука".
* Официальный сайт Центрального банка Российской Федерации. (2023). Курсы валют – <https://cbr.ru/>

Источники для C:

* Ввод/выводи работа с файлами в C — <https://metanit.com/c/tutorial/7.1.php>
* Структуры — <https://metanit.com/c/tutorial/6.1.php>
* Функции — <https://metanit.com/c/tutorial/4.1.php>
* Указатели — <https://metanit.com/c/tutorial/5.1.php>

# Приложение 1. Скрины работы программы для сотрудника

# Приложение 2. Скрины работы программы для администратора

# Приложение 3. Таблица план тестирования.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Таблица 6. План тестирования для программного модуля обменного-пункта. | | | | | |
| **ID теста** | **Описание теста (тип)** | **Предусловия** | **Шаги для воспроизведения** | **Ожидаемый результат** | **Фактический результат** |
| **1** | Ввод данных об обмене (позитивный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 1. 4. Ввести данные в поля «id обмена», «дата обмена», «id менеджера», «валюта», «сумма обмена». 5. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «данные успешно сохранены в файл». | Вывод на экран:  «данные успешно сохранены в файл». |
| **2** | Ввод данных об обмене (негативный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 1. 4. Ввести некорректные данные в поле «дата обмена». 5. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «ошибка ввода даты»; «введите дату заново». | Вывод на экран:  «ошибка ввода даты»; «введите дату заново». |
| **3** | Ввод данных об обмене (негативный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 1. 4. Ввести id обмена, который уже существует в файле. 5. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «обмен с таким ID уже существует», «введите ID заново». | Вывод на экран:  «обмен с таким ID уже существует», «введите ID заново». |
| **4** | Ввод данных об обмене (позитивный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера 3. Выбрать действие 1. 4. Ввести корректный id менеджера. 5. Нажать кнопку «Enter» на клавиатуре. | Продолжение ввода данных об обмене. | Продолжение ввода данных об обмене. |
| **5** | Ввод данных об обмене (негативный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 1. 4. Ввести некорректное обозначение валюты. 5. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «неверный ввод валюты»;  «введите заново». | Вывод на экран:  «неверный ввод валюты»;  «введите заново». |
| **6** | Ввод данных об обмене (негативный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 1. 4. Ввести некорректное значение суммы обмена 5. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «ошибка ввода»; «введите заново». | Вывод на экран:  «ошибка ввода»; «введите заново». |
| **7** | Загрузка данных из файла обмена (позитивный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 4. 4. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран: «файл обмена загружен в память». | Вывод на экран: «файл обмена загружен в память». |
| **8** | Загрузка данных их файла менеджеров (позитивный). | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 3. 4. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран: «файл менеджеров загружен в память». | Вывод на экран: «файл менеджеров загружен в память». |
| **9** | Редактирование данных об обмене (негативный) | Пользователь находится в меню менеджера. | 1. Выбрать действие 2. 2. Ввести id менеджера. 3. Выбрать действие 2. 4. Ввести некорректные данные в поле «дата продажи». 5. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «ошибка ввода даты»; «введите дату заново». | Вывод на экран:  «ошибка ввода даты»; «введите дату заново». |
| **10** | Удаление менеджера (негативный) | Пользователь находится в меню администратора. | 1. Выбрать действие 3. 2. Ввести пароль админа. 3. Выбрать действие 9. 4. Выбрать 2 (удалить менеджера) 5. Ввести некорректный id менеджера. 6. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «менеджер не найден». | Вывод на экран:  «менеджер не найден». |
| **11** | Добавление менеджеров (позитивный) | Пользователь находится в меню редактирования данных о менеджере. | 1. Выбрать действие 3. 2. Ввести пароль админа. 3. Выбрать действие 9. 4. Выбрать действие 1 5. Ввести корректное данные: id\_менеджера, ФИО. 6. Нажать кнопку «Enter» на клавиатуре. | Вывод на экран:  «менеджер успешно добавлен». | Вывод на экран:  «менеджер успешно добавлен». |
| **12** | Выполнение запроса по обменам | Пользователь находится в основном меню. | 1. Выбрать действие 2. 2. Ввести id админа 3. Выбрать действие 5. 4. Нажать кнопку «Enter» на клавиатуре. | 1. Вывод на экран:  «Результат сохранен в файле query1.txt» 2. Сохранение результата в файле query1.txt. | 1. Вывод на экран:  «Результат сохранен в файле query1.txt» 2. Сохранение результата в файле query1.txt. |