

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Кубанский государственный технологический университет»
(ФГБОУ ВО КубГТУ)

Институт Компьютерных систем и информационной безопасности
Кафедра Информационных систем и программирования
Специальность 10.05.03 Информационная безопасность автоматизированных систем
Профиль Защищенные автоматизированные системы управления

КУРСОВАЯ РАБОТА

по дисциплине Технологии и методы программирования
(наименование дисциплины)

на тему: «Холодильник»
(тема курсовой работы)

Выполнил студент 2 курса группы 18-К-АС1

Лаврененков Г.Ю.
(Ф.И.О.)

Допущен к защите _____

Руководитель (нормоконтролер) работы О.Б. Попова

Защищен _____ Оценка _____
(дата)

Члены комиссии
Н.В. Кушнир _____
К.Е. Тотухов _____

Краснодар
2020

ФГБОУ ВО «Кубанский государственный технологический университет»
(ФГБОУ ВО КубГТУ)

Институт Компьютерных систем и информационной безопасности
Кафедра Информационных систем и программирования
Специальность 10.05.03 Информационная безопасность автоматизированных систем
Профиль Защищенные автоматизированные системы управления

УТВЕРЖДАЮ
Зав. кафедрой _____ М.В. Янаева
«12» февраля 2020 г.

ЗАДАНИЕ
на курсовую работу

Студенту: Лаврененкову Г.Ю. группы 18-К-АС1 курса 2
(Ф.И.О.) (№ группы и курса)

Тема проекта: «Холодильник»

План работы:

1. Изучение предметной области
2. Проектирование
3. Описание реализованных диаграмм

Объем работы:

а) пояснительная записка 24 с.

Рекомендуемая литература

1. Йордон. «Объектно-ориентированный анализ и проектирование систем»
2. Роберт А. Максимчук. «UML для простых смертных»
3. «Автоматизация проектирования вычислительных систем.» ред. М.Брейер

Срок выполнения: с «15» февраля по «11» мая 2020г.

Срок защиты: с «11» мая по «14» июня 2020 г.

Дата выдачи задания «15» февраля 2020г.

Дата сдачи работы на кафедру «01» июня 2020 г.

Руководитель работы _____ к.т.н., доцент Попова О.Б.
(должность, подпись,)

Задание принял студент Лаврененков Г.Ю. Ф.И.О.

Реферат

Курсовая работа: 33 страница, 32 рисунка, 6 используемых источников.

Ключевые слова: ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ПРОЕКТИРОВАНИЕ, ПОСЛЕДОВАТЕЛЬНОСТИ, МОДЕЛЬ, КЛАСС, ДОМОФОН, UML, BPMN, ГАНТ, EPC, FURPS+ , IDEF0, DFD, ДИАГРАММЫ.

В процессе выполнения данного задания было разработано программное обеспечение для холодильника, способного значительно упростить деятельность работников данной организации.

Целью работы является разработка проекта холодильника с использованием диаграмм разного вида, в полной мере описывающих как внутреннее устройство исследуемой системы, так и всевозможные взаимодействия между её компонентами.

В конечном итоге были получены диаграммы, обладающие исчерпывающей информацией о программном обеспечении холодильника.

К ним относятся: диаграмма Ганта, UML-диаграмма, IDEF0-диаграмма, DFD-диаграмма, EPC-диаграмма, BPMN «As-Is» и BPMN «To be», документ FURPS+.

Содержание

Введение.....	5
1 Формулировка задачи.....	6
2 Диаграмма Ганта	7
3 Создание модели As-Is в стандарте IDEF0.....	8
4 Диаграмма потоков данных (DFD)	11
5 UML.....	12
6 EPC	13
8 FURPS+	15
9 Результаты машинного тестирования программы	16
10 Системные требования	19
11 Руководство пользователя.....	20
Заключение	21
Список использованных источников	22
Приложение А – Проверка на антиплагиат.....	23
Приложение Б – Диаграмма Ганта.....	24

Введение

В настоящее время информационных технологий происходит автоматизация всех процессов, что значительно упрощает работу в организациях. Программное обеспечение позволяет не только сократить время и объем работы, но также обеспечить избежание ошибок человеческого фактора.

Программа, основанная на WinForms позволяет избежать использования большого количества бумажных носителей информации. Гарантирован моментальный доступ к двум основным базам данных: список товаров (цена и количество) и список товаров, перемещённых в корзину (название и цена).

Таким образом, программа позволяет значительно упростить и ускорить взаимодействие с клиентом, а также повысить качество обслуживания.

1 Формулировка задачи

Программное обеспечение встроенного процессора холодильника. Холодильник состоит из нескольких холодильных камер для хранения продуктов. В каждой холодильной камере имеется регулятор температуры, мотор, термометр, индикатор, таймер, датчик открытия двери камеры и устройство для подачи звуковых сигналов.

При помощи терморегулятора устанавливается максимально допустимая температура в данной камере. Мотор предназначен для поддержания низкой температуры. Термометр постоянно измеряет температуру внутри камеры, а индикатор температуры, расположенный на дверце, постоянно высвечивает ее значение. При повышении температуры выше предела, определяемого текущим положением регулятора, включается мотор. При снижении температуры ниже некоторого другого значения, связанного с первым, мотор отключается.

Доступ в камеру осуществляется через дверцу. Если дверь холодильной камеры открыта в течение слишком долгого времени, подается звуковой сигнал. Звуковой сигнал также подается в любых нештатных ситуациях (например, при поломке мотора). Холодильник ведет электронный журнал, в котором отмечаются все происходящие события:

- изменение положения терморегулятора камеры;
- включение и отключение мотора;
- доступ в камеру;
- нештатные ситуации.

2 Диаграмма Ганта

Диаграмма Ганта — «это популярный тип столбчатых диаграмм (гистограмм), который используется для иллюстрации плана, графика работ по какому-либо проекту. Является одним из методов планирования проектов. Придумал американский инженер Генри Гант (Henry Gantt). Выглядит это как горизонтальные полосы, расположенные между двумя осями: списком задач по вертикали и датами по горизонтали.

На диаграмме видны не только сами задачи, но и их последовательность. Это позволяет ни о чём не забыть и делать всё своевременно.

Ключевым понятием диаграммы Ганта является «веха» — метка значимого момента в ходе выполнения работ, общая граница двух или более задач. Вехи позволяют наглядно отобразить необходимость синхронизации, последовательности в выполнении различных работ. Вехи, как и другие границы на диаграмме, не являются календарными датами. Сдвиг вехи приводит к сдвигу всего проекта. Поэтому диаграмма Ганта не является, строго говоря, графиком работ. Кроме того, диаграмма Ганта не отображает значимости или ресурсоемкости работ, не отображает сущности работ (области действия). Для крупных проектов диаграмма Ганта становится чрезмерно тяжеловесной и теряет всякую наглядность.»

Диаграмма Ганта для проекта «Холодильник» находится в «Приложении Б».

3 Создание модели As-Is в стандарте IDEF0

Чтобы оценить возможности, разрабатываемой системы, необходимо построить её базовую модель, которую можно представить в виде диаграммы As-Is.

Диаграмма As-Is – это функциональная модель системы «как есть», позволяющая узнать где находятся слабые места, в чём будут состоять преимущества и недостатки, протекающих в ней бизнес-процессов относительно конкурентов. Применение данной модели позволит чётко зафиксировать какие информационные объекты принимают участие в жизненном цикле системы, какая информация будет поступать на вход и что будет получаться на выходе. Модель As-Is, строится с использованием нотации IDEF0.

IDEF0 – это графическая нотация, предназначенная для описания бизнес-процессов. Система, описываемая в данной нотации, проходит через декомпозицию или, иными словами, разбиение на взаимосвязанные функции. Для каждой функции существует правило сторон:

- стрелкой слева обозначаются входные данные;
- стрелкой сверху – управление;
- стрелкой справа – выходные данные;
- стрелкой снизу – механизм.

Учитывая всё вышеперечисленное на рисунке 1 была составлена модель As-Is проекта «Холодильник».

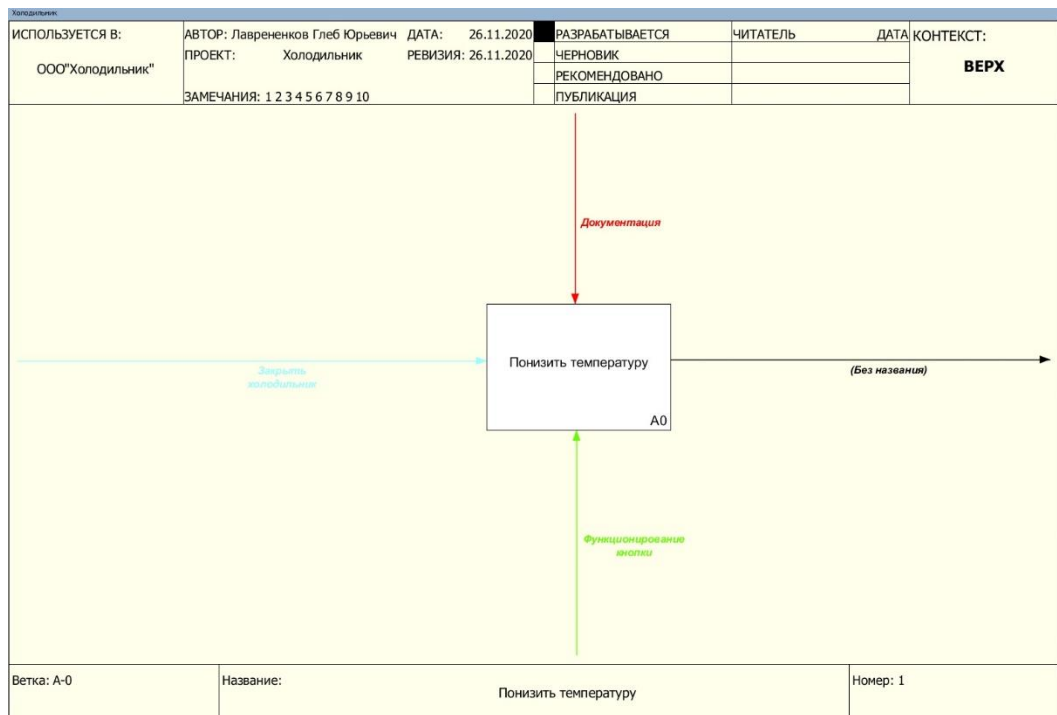


Рисунок 3 – Декомпозиция системы «Холодильник»

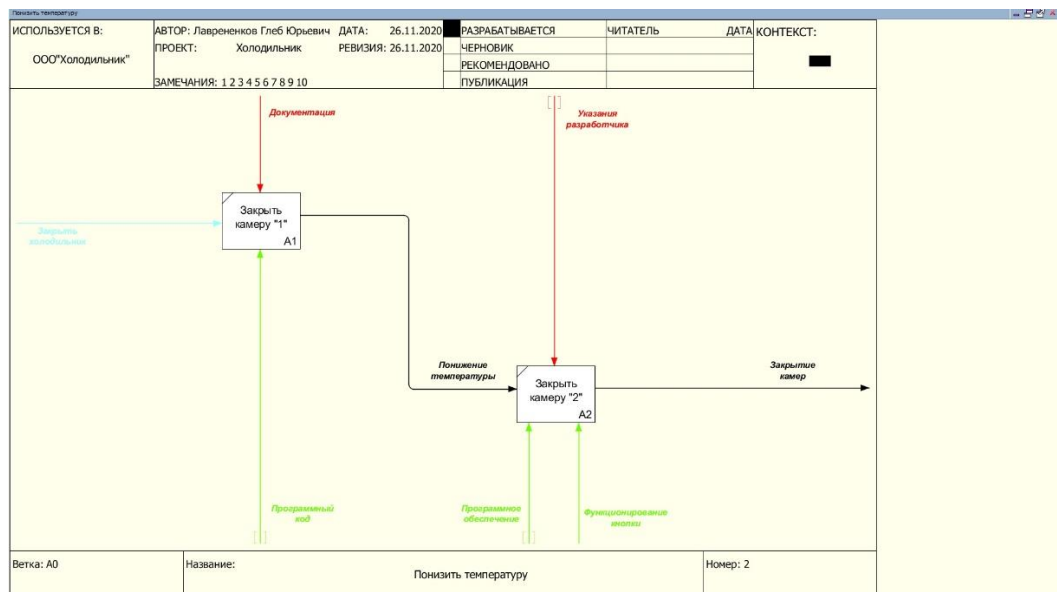


Рисунок 4 – Декомпозиция кода ПО «Холодильник»

Входными данными для данной системы являются действия человека, необходимые для того, чтобы открыть камеру.

Управление происходит при помощи сенсорного экрана и кнопок на корпусе холодильника, а также правилам, описанным в инструкции.

4 Диаграмма потоков данных (DFD)

Диаграмма потоков данных DFD (DataFlowDiagrams) – «это методология графического структурного анализа, описывающая внешние по отношению к системе источники и адресаты данных, логические функции, потоки данных и хранилища данных, к которым осуществляется доступ. Диаграмма DFD – это один из основных инструментов структурного анализа и проектирования информационных систем, существовавших до широкого распространения UML.»

В результате декомпозиции системы «программное обеспечение для «Холодильник» была получена следующая диаграмма DFD (рис. 5).

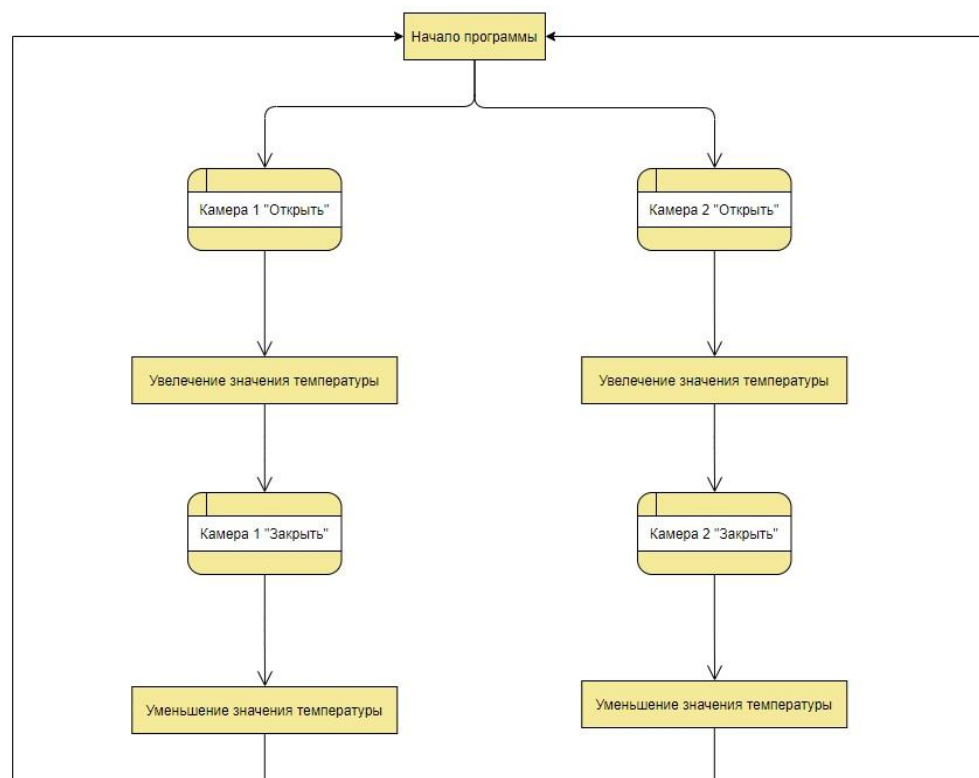


Рисунок 13 – Диаграмма DFD-системы «программное обеспечение для «Холодильника»

Внешними сущностями данной системы является оператор устройства, под управлением которого находится программа, а также сама программа.

5 UML

UML (англ. Unified Modeling Language — «унифицированный язык моделирования») — язык графического описания для объектного моделирования в области разработки программного обеспечения, для моделирования бизнес-процессов, системного проектирования и отображения организационных структур.

UML является языком широкого профиля, это — открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML-моделей возможна генерация кода.»

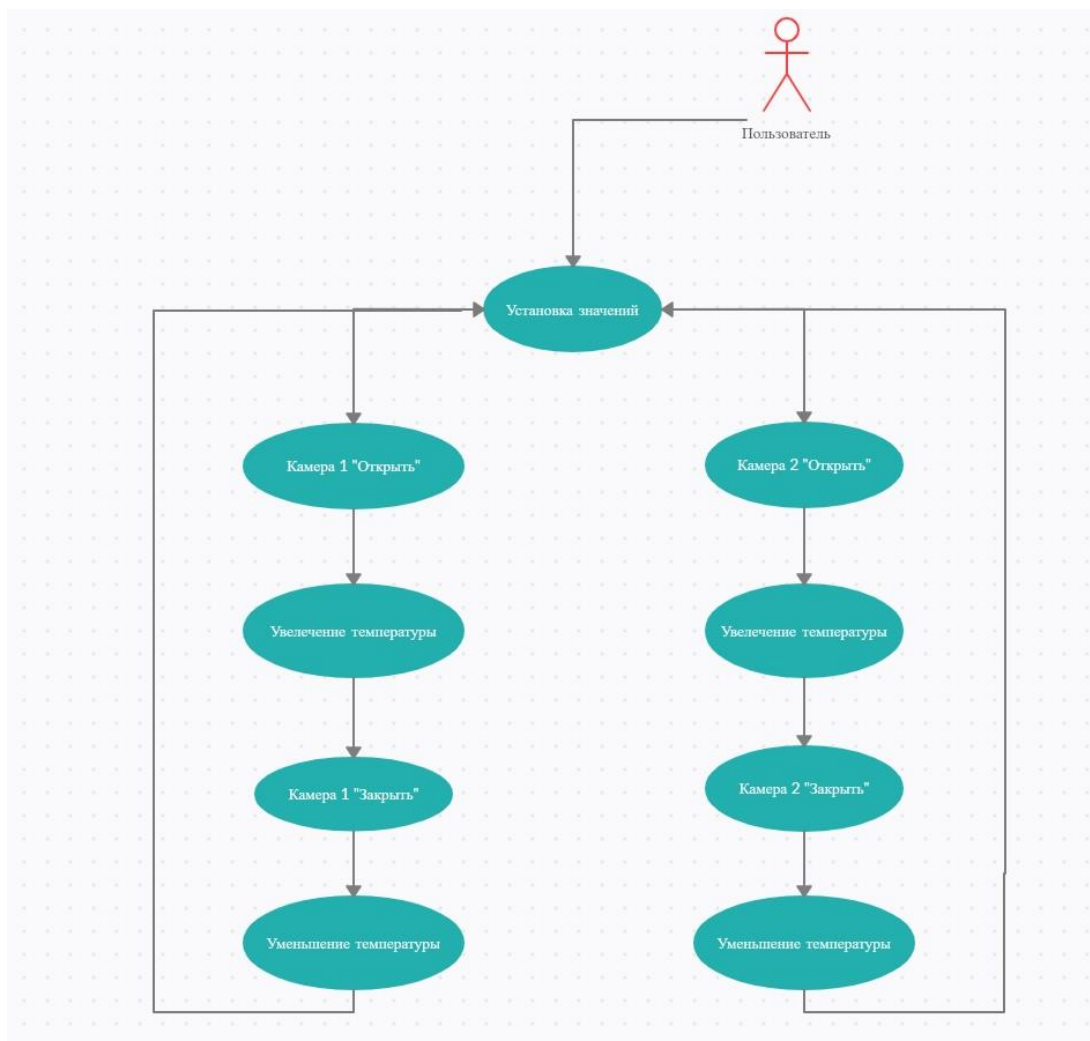


Рисунок 14– UML-диаграмма программного обеспечения «Холодильника»

6 EPC

Событийная цепочка процессов (EPC-диаграмма, англ. event-driven process chain) — «тип блок-схемы, используемой для бизнес-моделирования. EPC может быть использована для настройки системы планирования ресурсов предприятия (ERP), и для улучшений бизнес-процессов.

Организации используют EPC-диаграммы для планирования потоков работ бизнес-процессов. Существует ряд инструментов для создания EPC-диаграмм, некоторые из этих средств поддерживают инструментонезависимый формат обмена данными EPC — язык разметки EPML. EPC-диаграммы используют символы нескольких видов, чтобы показать структуру потока управления (последовательность решений, функции, события и другие элементы) бизнес-процесса.

EPC-метод был разработан Августом-Вильгельмом Шеером в рамках работ над созданием ARIS в начале 1990-х годов. Используется многими организациями для моделирования, анализа и реорганизации бизнес-процессов.»

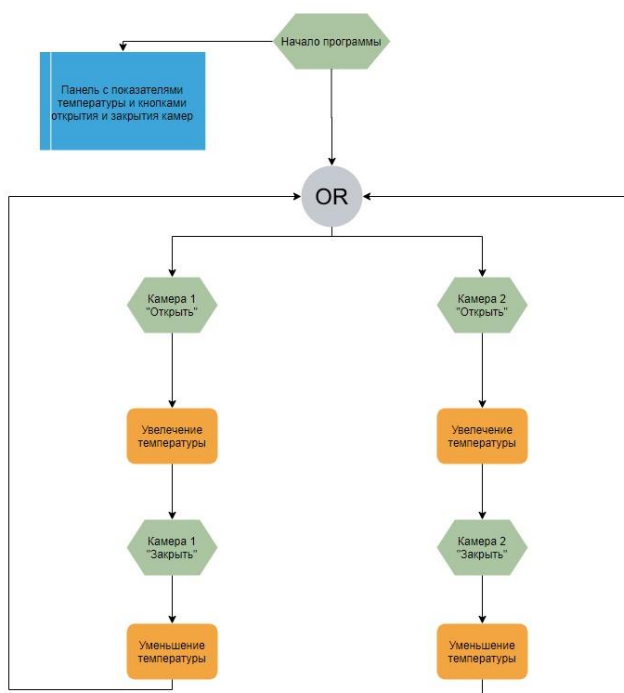


Рисунок 15 – EPC-диаграмма программного обеспечения «Холодильника»

7 BPMN

BPMN (Business Process Management Notation) – «это язык моделирования бизнес-процессов, который является промежуточным звеном между формализацией/визуализацией и воплощением бизнес-процесса.

Говоря проще, такая нотация представляет собой описание графических элементов, используемых для построения схемы протекания бизнес-процесса.

Как минимум, такая схема нужна, чтобы выстроить в соответствии с ней бизнес процесс и понятно регламентировать его для всех участников.

Как максимум, моделирование BPMN позволяет впоследствии провести автоматизацию бизнес-процессов в соответствии с имеющейся схемой.»

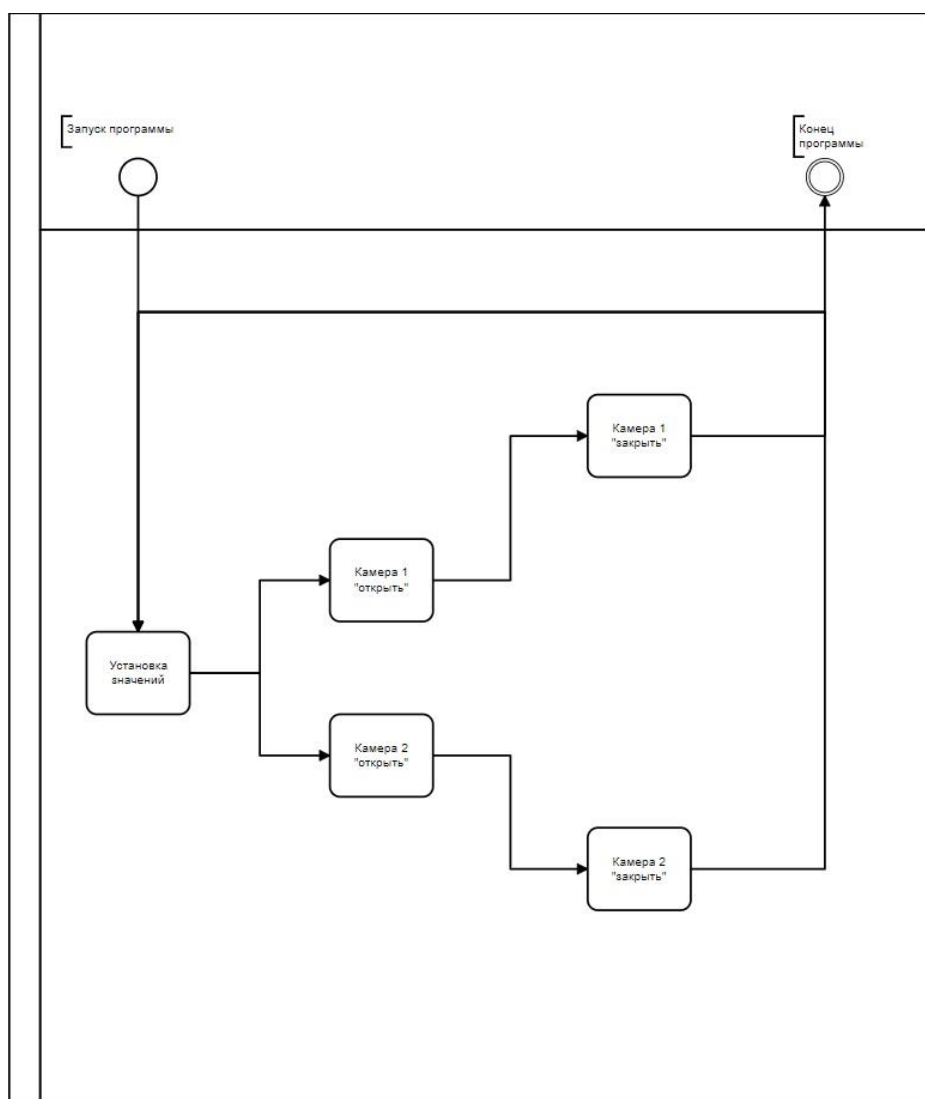


Рисунок 16 – Диаграмма BPMN «As-Is» и «To be»

8 FURPS+

Классификация требований к системе FURPS+ была разработана Робертом Грэйди (Robert Grady) из Hewlett-Packard и предложена в 1992 году.

Сокращение FURPS расшифровывается так:

1. Functionality, функциональность
2. Usability, удобство использования
3. Reliability, надежность
4. Performance, производительность
5. Supportability, поддерживаемость
6. + необходимо помнить о таких возможных ограничениях, как:
 - ограничения проектирования, design
 - ограничения разработки, implementation
 - ограничения на интерфейсы, interface
 - физические ограничения, physical

Если применить к этой классификации популярное разделение требований на функциональные и нефункциональные, то к последним следует отнести все перечисленные выше группы кроме первой, т.е. URPS+.

F – стандартный набор функций;

U – приятный дизайн, интуитивно понятный интерфейс;

R – 1 сбой/5 лет; среднее время сбоя – 3 секунды; время готовности системы к работе – 5 сек.

P – время отклика системы 0.01 сек, 100% эффективность работы, пропускная способность 30 запросов в минуту; потребление ресурсов – 60 Вт/час;

S – легкая настройка;

+ - ограничение температуры

Рисунок 9 – FURPS+ для системы «Холодильник»

9 Результаты машинного тестирования программы

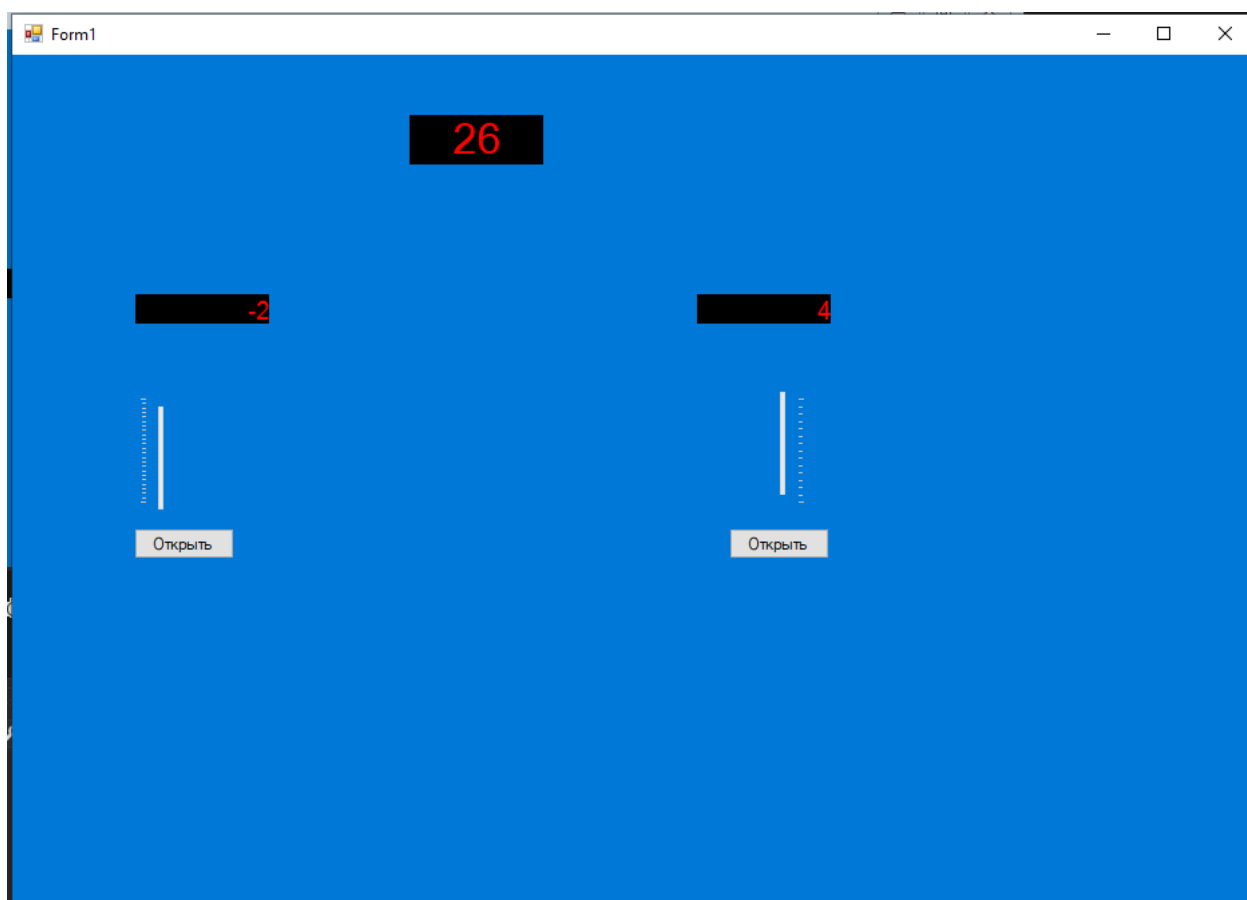


Рисунок 15 – Основное меню.

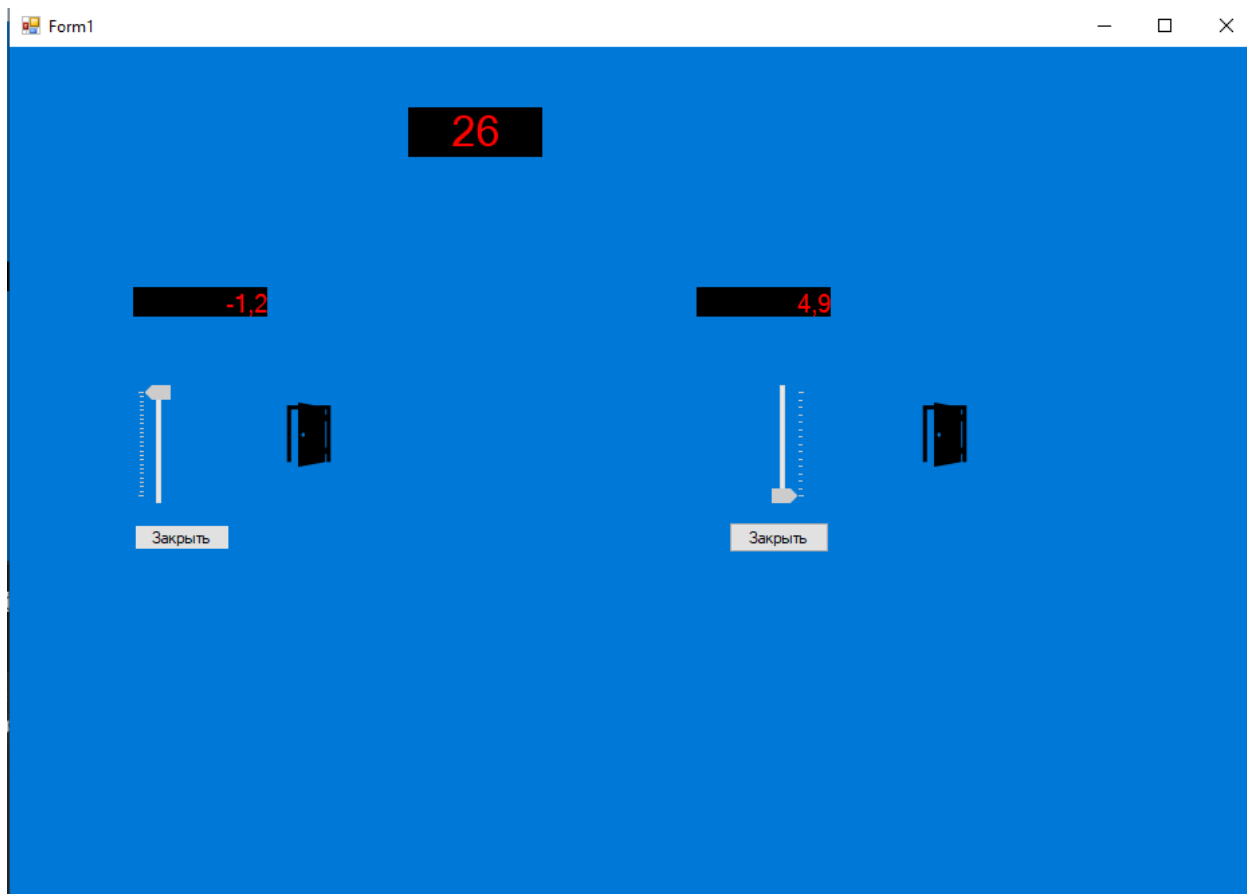


Рисунок 16 – Открытые камеры

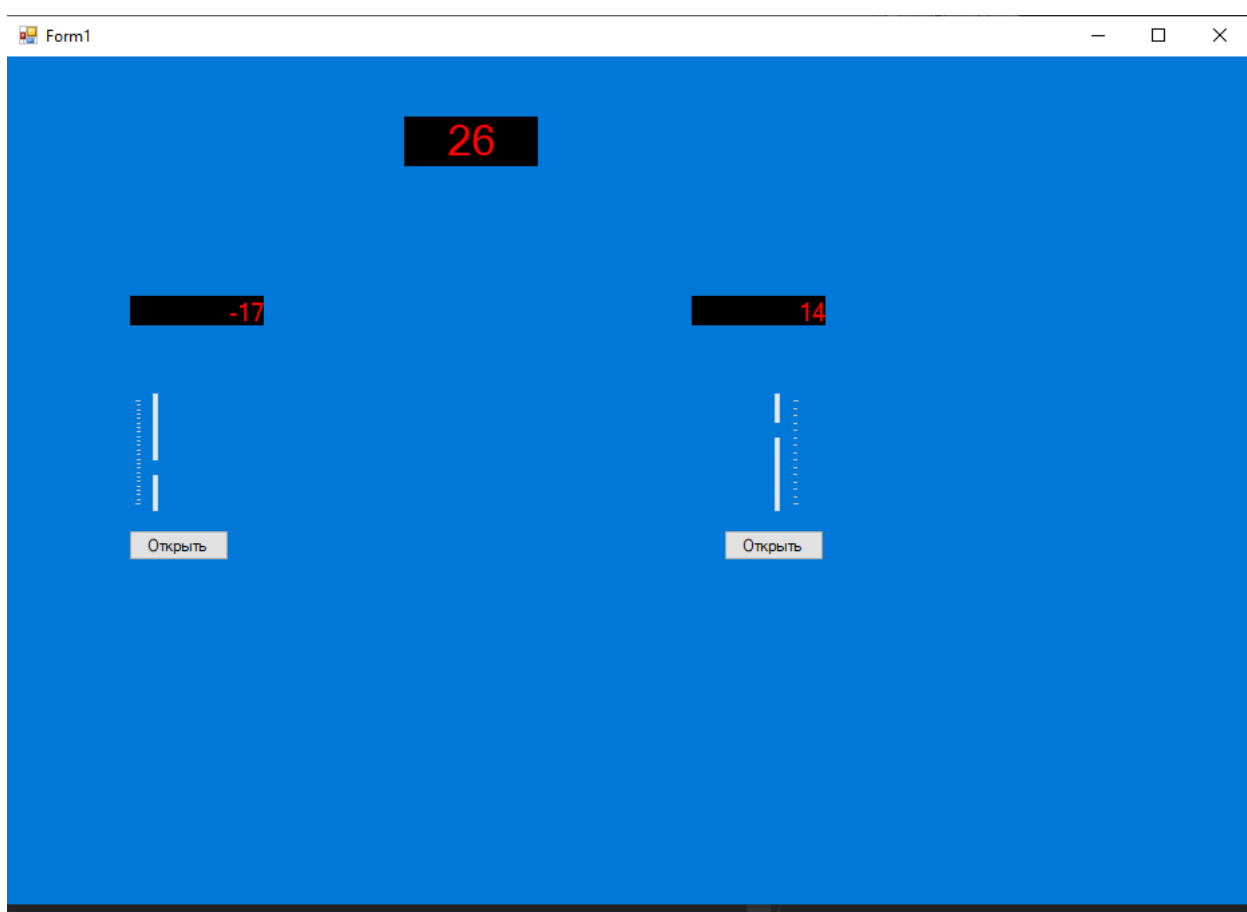


Рисунок 17 – Температуру можно вводить вручную

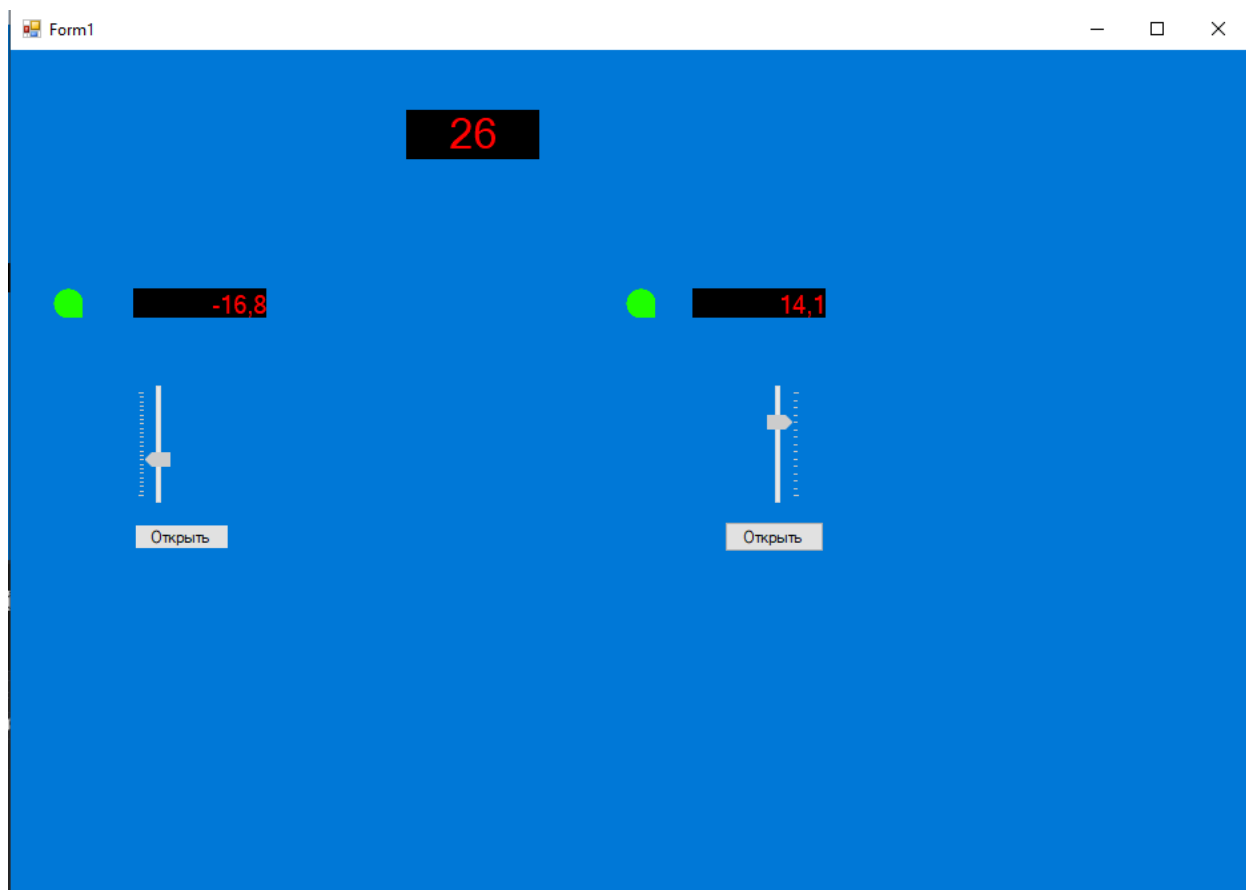


Рисунок 18 – Результаты, если закрыть камеру

10 Системные требования

Таблица 1 – Системные требования программы

Процессор	2.8 ГГц
Оперативная память	150 Мб
Монитор	1600 x 900
Свободное место на носителе	50 Мб
Устройства взаимодействия с пользователем	Клавиатура и мышь
Программное обеспечение	Visual Studio 2019 года последней версии

11 Руководство пользователя

Запуск программы осуществляется открытием исполняемого файла .sln.

Теперь перед нами открылось стартовое и основное меню программы.

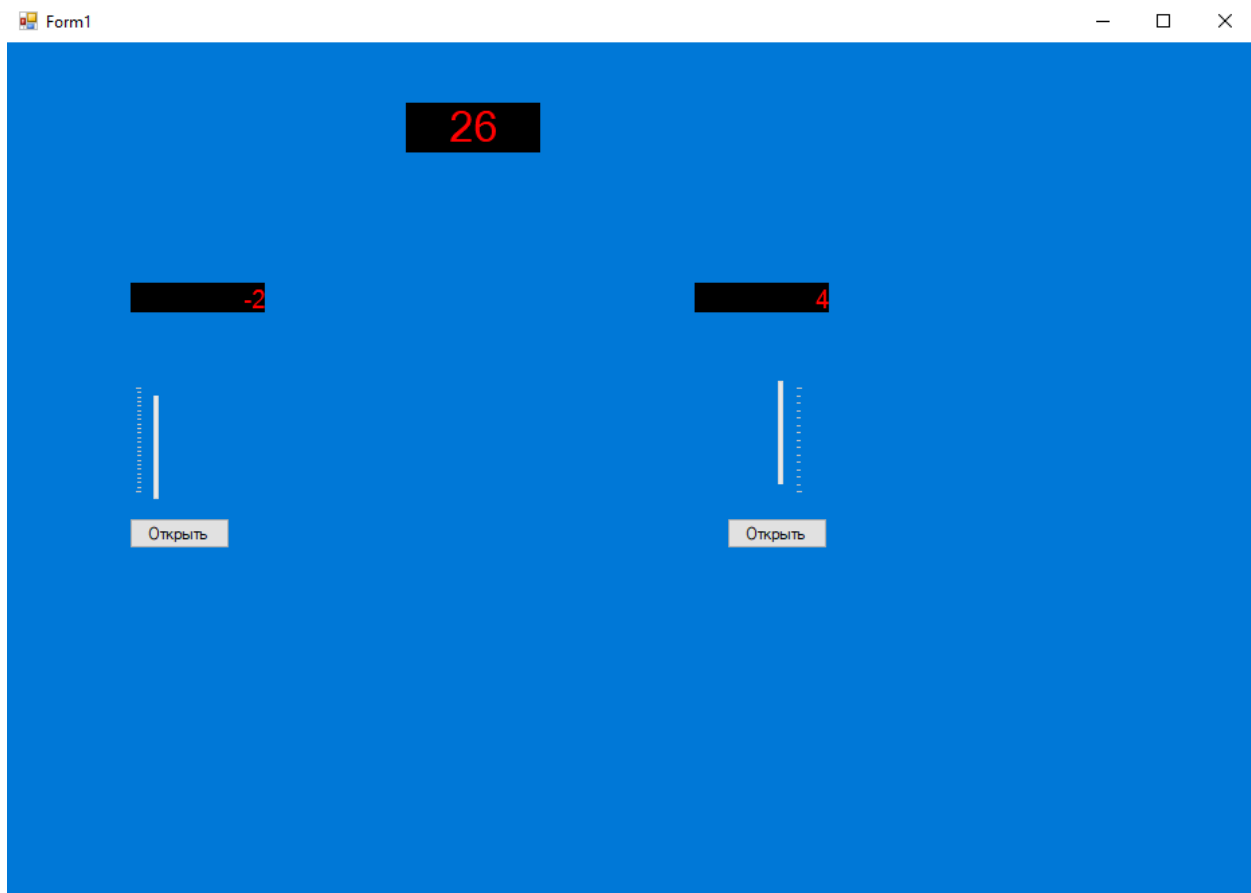


Рисунок 26 – Начальный интерфейс программы

Первым делом необходимо выбрать нужную для вас температуру, используя специальные ячейки. Интерфейс интуитивно понятен, основные возможности данного ПО подробно разобраны.

Заключение

В результате выполнения данного курсового проекта было разработано ПО для симулятора телевизора. Программирование осуществлялось на языке высокого уровня C#, использование данного языка позволило удобно разработать симулятор и наглядно показать результат работы при помощи встроенного в Visual Studio интерфейса программирования приложений Windows Forms. Также были спроектированы диаграммы, которые позволяют детально изучить не только процесс машинного выполнения программы, но также и оценить процесс создания (проектирования и реализации) данного проекта.

При построении диаграмм применялись основные правила и принципы моделирования, включающие графическое представление объектов и связей между ними, иерархическое построение, а также названия, отражающие назначение той или иной сущности, или взаимодействия.


Благодаря детальному разбору проекта при помощи диаграмм проектирования, полученных в процессе разработки, можно с уверенностью сказать, что разработанное ПО удовлетворяет все потребности потребителя, позволяет комфортно использовать телевизор.


Были получены важные знания и практические навыки как в области использования объектно-ориентированных языков программирования в целом, так и в области построения диаграмм проектирования, отображающих поведение различных организационных структур.


Список использованных источников


1. Ларман, Крэг. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную разработку / Крэг Ларман. - Москва: Гостехиздат, 2017. - 736 с.
2. Роберт А. Максимчук. UML для простых смертных / Роберт А. Максимчук, Эрик Дж. Нейбург. - Москва: СИНТЕГ, 2014. - 272 с.
3. Йордон, Эдвард. Объектно-ориентированный анализ и проектирование систем / Эдвард Йордон, Карл Аргила. - М.: ЛОРИ, 2014. - 264 с.
4. SoloLearn – C# Tutorial. [Электронный ресурс]: - Режим доступа: <https://www.sololearn.com/Course/CSharp/> (Дата обращения 13.03.2020).
5. Википедия. [Электронный ресурс]: - Режим доступа: <https://ru.wikipedia.org> (Дата обращения 17.09.2019).
6. GitHub – pavelshevchenko/TV. [Электронный ресурс]: - Режим доступа: <https://github.com/pavelshevchenko/TV> (Дата обращения 9.05.2020).
7. Comindware – Нотация BPMN 2.0 [Электронный ресурс]: - Режим доступа: <https://comindware.com/ru/blog-нотация-bpmn-2-0-элементы-и-описание/> (Дата обращения 28.02.2020)
8. SysAna– Требования к системе: классификация FURPS+ [Электронный ресурс]: - Режим доступа: <https://sysana.wordpress.com/2010/09/16/furps/> (Дата обращения 03.03.2020)



Приложение А – Проверка на антиплагиат


**АНТИПЛАГИАТ**
ТВОРИТЕ СОБСТВЕННЫМ УМОМ




 **ПОЛЬЗОВАТЕЛЬ**
gleblav00@mail.ru

 **БАЛЛОВ**
0

 **ТАРИФ** 
Бесплатный доступ (0/0)

 **МОДУЛИ И КОЛЛЕКЦИИ**
Подключено: 1 [смотреть](#)

 **МЕНЮ**

ru

ГЛАВНАЯ / КАБИНЕТ / РЕЗУЛЬТАТЫ ПРОВЕРКИ /

Краткий отчет

получить полный отчет

[ПАРАМЕТРЫ ПРОВЕРКИ](#) [ЭКСПОРТ](#) [ИСТОРИЯ ОТЧЕТОВ](#) [ВЫЙТИ В КАБИНЕТ](#) [ЕЩЕ...](#)

Лаврененков Глеб

ПРОВЕРЕНО: 26.11.2020 05:41:02

№	Доля в отчете	Источник	Актуальна на	Модуль поиска
[01]	1,53%	Основные методологии разработки программного обеспечения	02 Дек 2019	Модуль поиска Интернет
[02]	0,02%	Основные методологии разработки программного обеспечения _ Авторские прогр...	04 Янв 2016	Модуль поиска Интернет
[03]	1,11%	2017_М_ПИ_ЗИС_МД_Матицин_Илья_Николаевич_ЗВВКР.pdf	10 Авг 2018	Модуль поиска Интернет

ЗАИМСТВОВАНИЯ
26,29%

САМОЦИТИРОВАНИЯ
0%

ЦИТИРОВАНИЯ
0%

ОРИГИНАЛЬНОСТЬ
73,71%

ИСТОЧНИКОВ: 20

ЕЩЕ НАЙДЕНО ИСТОЧНИКОВ: 17

ЗАИМСТВОВАНИЯ: 23,64%

Приложение Б – Диаграмма Ганта



	Иконка	Название	Продолжи...	Начало	Окончание	Предает...	Название ресурса
1		Стартовое совещание	1 день	16.02.20 ...	17.02.20 17:00		Куратор со стороны Заказчика[50%];Куратор со стороны Исполнителя[50%]
2		Формирование проектной команды	3 дней	18.02.20 ...	20.02.20 17:00	1	Куратор со стороны Заказчика;Куратор со стороны Исполнителя;Руководитель проекта
3		Разработка функциональной модели	2 дней	21.02.20 ...	24.02.20 17:00	2	Консультант
4		Разработка высокоуровневой модели	2 дней	25.02.20 ...	26.02.20 17:00	3	Консультант
5		Проведение анализа и проектирование	3 дней	27.02.20 ...	02.03.20 17:00	4	Консультант;Куратор со стороны Заказчика[50%];Куратор со стороны Исполнителя[50%]
6		Реализация проекта	60 дней	03.03.20 ...	25.05.20 17:00	5	Программист
7		Тестирование проекта	10 дней	26.05.20 ...	08.06.20 17:00	6	Программист;Тестировщик
8		Развертывание и внедрение	0 дней	09.06.20 ...	09.06.20 8:00	7	Поставщик оборудования
9		Окончание проекта	2 дней	09.06.20 ...	10.06.20 17:00	8	Руководитель проекта со стороны Заказчика;Руководитель проекта со стороны Исполнителя