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

Направление подготовки     09.03.04 Программная инженерия

Профиль               Беспрофильное

**КУРСОВОЙ ПРОЕКТ**

по дисциплине Разработка, анализ и управление программными проектами

на тему: «Интернет-магазин бытовой техники»

Выполнила студентка      4      курсагруппы 17-КБ-ПР1

Белоусова Елизавета Дмитриевна

Допущен к защите .

Руководитель (нормоконтролер)

работы                               к.т.н., доцент Попова О.Б

Защищен \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члены комиссии .

.

.

Краснодар

2021

ФГБОУ ВО «Кубанский государственный технологический университет»

(ФГБОУ ВО КубГТУ)

Институт Компьютерных систем и информационной безопасности

Кафедра Информационных систем и программирования

Направление подготовки     09.03.04 Программная инженерия

Профиль               Беспрофильное

УТВЕРЖДАЮ

Зав. кафедрой \_\_\_\_\_\_М.В. Янаева.

«08 » 2021 г.

**ЗАДАНИЕ**

на курсовой проект

Студентке: Белоусовой Е.Д группы 17-КБ-ПР1 курса 4

Тема проекта: «Интернет-магазин бытовой техники»

(утверждена указанием директора института № (от 2021г.)

План работы:

1. Изучение предметной области

2. Проектирование системы .

Объем работы:

а) пояснительная записка 24  с.

Рекомендуемая литература

Гагарина Л.Г., Кокорева Е.В., Виснадул Б.Д. Технология разработки программного обеспечения: учебное пособие / Под ред. Л.Г. Гагариной. – М.: ИД «Форум»: Инфра-М, 2013. – 400 с

Срок выполнения: с « » по « » 2021г.

Срок защиты:                      « » 2021г.

Дата выдачи задания:                « » 2021г.

Дата сдачи работы на кафедру:       « » 2020г.

Руководитель работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ к.т.н., доцент Попова О.Б

Задание приняла студентка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Белоусова Е.Д

**Реферат**

Пояснительная записка к курсовому проекту 24 с., 12 рисунков, 6 источников.

ПРОЕКТИРОВАНИЕ ПРОГРАМНЫХ СИСТЕМ, АРХИТЕКТРУА ПРОГРАММНЫХ СИСТЕМ, UML, ИЕРАРХИЯ ПРОЕКТНЫХ КЛАССОВ СИСТЕМ, ДИАГРАММЫ.

Объектом исследованием является проектирование программных систем в различных сферах деятельности.

Цель работы состоит в проектировании программного обеспечения табло для кодового замка.

К полученным результатам относится различного вида диаграммы, описывающие суть заданного проекта.

**Содержание**

[Введение 5](#_Toc64484302)

[1 Нормативные ссылки 6](#_Toc64484303)

[2 Постановка задачи 7](#_Toc64484304)

[2.1 Задание на проектирование 7](#_Toc64484305)

[2.2 Утверждение целей проекта 8](#_Toc64484306)

[2.3 Оценка затрат на проект 11](#_Toc64484307)

[2.2 Глоссарий 12](#_Toc64484308)

[3 Проектирование системы 14](#_Toc64484309)

[3.1 Начальное проектирование 14](#_Toc64484310)

[3.2 Построение иерархии классов 16](#_Toc64484311)

[3.4 Функциональной проектирование 18](#_Toc64484312)

[Заключение 22](#_Toc64484313)

[Список использованных источников 23](#_Toc64484314)

[Приложение А - Оригинальность работы 24](#_Toc64484315)

**Введение**

Разработка программных систем – процесс трудоемкий и в то же время творческий. В условиях развитого рынка программного обеспечения разработчики вынуждены решать две во многом противоречивые задачи: с одной стороны, создание программного комплекса в заданное время в соответствии с формальными требованиями заказчика, а с другой стороны, творческий процесс исследования для его оптимальной реализации.

Программное обеспечение компьютерных технологий создаётся в большинстве случаев коллективно и только правильное управление программными проектами приводит их разработчиков к успеху. Правильное управление в свою очередь связано с планированием, организацией, мотивацией и контролем процессов разработки.

Управление программными проектами – область деятельности, в ходе которой определяются и достигаются четкие цели программного проекта при балансировании между требованиями к системе и ПО, объёмом работ, потребными ресурсами (такими как деньги, труд, материалы, энергия, пространство и др.), временем, качеством и рисками.

В данном курсовом проекте будет закрепление основ и углубление знаний в области дисциплины, получение дополнительных практических навыков и изучение приемов разработки проектов программных продуктов с использованием языка UML. В процессе выполнения будет выполнятся все этапы создания проекта программного продукта от постановки задачи до сопровождающейся документацией и инструкциями по его использованию.

В качестве результата должна являться разработка проекта приложения с использованием CASE-технологий. Для решения этих задач при описании поведения проектируемого программного обеспечения в настоящее время используется UML (Unified Modeling Language) – унифицированный язык моделирования.

# **1 Нормативные ссылки**

В данных методических указаниях использованы ссылки на следующие стандарты:

ГОСТ Р 1.5-2012. Стандартизация в Российской Федерации. Стандарты национальные. Правила построения, изложения, оформления и обозначения

ГОСТ Р 7.0.5-2008 СИБИД. Библиографическая ссылка. Общие требования и правила составления

ГОСТ Р 7.0.12-2011 СИБИД. Библиографическая запись. Сокращение слов и словосочетаний на русском языке. Общие требования и правила

ГОСТ 2.301-68 ЕСКД. Форматы

ГОСТ 7.9-95 СИБИД. Реферат и аннотация. Общие требования

ГОСТ 7.82-2001 СИБИД. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила со-сдавления

# **2 Постановка задачи**

## **2.1 Задание на проектирование**

Требуется разработать модель программного обеспечения встроенного микропроцессора для кодового замка, регулирующего доступ в помещение.

Кодовый замок состоит из панели с кнопками (цифры «0»...«9», кнопка «Вызов», кнопка «Контроль»), цифрового дисплея, электромеханического замка, звонка.

Панель с кнопками устанавливается с наружной стороны двери, замок устанавливается с внутренней стороны двери, звонок устанавливается внутри охраняемого помещения. В обычном состоянии замок закрыт.

Доступ в помещение осуществляется после набора кода доступа, состоящего из четырех цифр.

Во время набора кода введенные цифры отображаются на дисплее. Если код набран правильно, то замок открывается на некоторое время, после чего дверь снова закрывается.

Содержимое дисплея очищается. Кнопка «Вызов» используется для подачи звукового сигнала внутри помещения. Кнопка «Контроль» используется для смены кодов.

Смена кода доступа осуществляется следующим образом. При открытой двери нужно набрать код контроля, состоящий из четырех цифр, и новый код доступа. Для смены кода контроля нужно при открытой двери и нажатой кнопке «Вызов» набрать код контроля, после чего – новый код контроля.

Перед началом проектирования системы построим диаграмму, для более точного представления задачи.

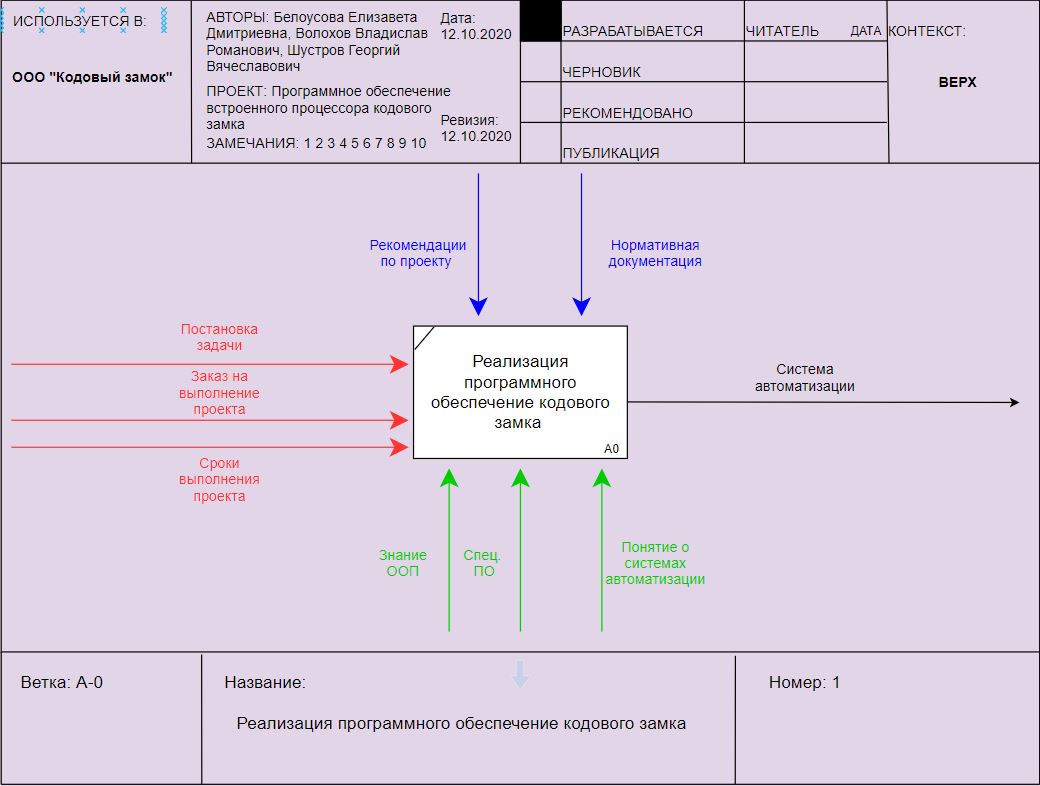


Рисунок 1 – Диаграмма IDEF

## **2.2 Утверждение целей проекта**

Цель данного раздела: утверждение целей проекта, требований к результатам, ресурсных затрат проекта, организационной структуры и ответственности в проекте, процедур проекта.

Цель – автоматизация отдельных видов деятельности компании ООО “Кодовый замок”.

Задачи проекта – в период с 1.01.2021 г. по 5.04.2021 г.:

1. Разработать документ “Модель основных бизнес-процессов ООО “Кодовый замок” по типу “как есть”;
2. Разработать документ “Функциональная модель предприятия ООО “Кодовый замок”;
3. Разработать документ “Высокоуровневая модель данных предприятия ООО “Кодовый замок”
4. Разработать документ “Предложение по автоматизации ООО “Кодовый замок”;
5. Разработать документ “Модель автоматизируемых бизнес-процессов ООО “Кодовый замок” по типу “как должно быть”;
6. Разработать документ “Требования к разрабатываемой информационной системе”;
7. Разработать документ “Результаты анализа и проектирования информационной системы”;
8. Провести реализацию информационной системы;
9. Провести тестирование информационной системы и разработать документ “Результаты тестирования информационной системы”;
10. Произвести развертывание и внедрение информационной системы в промышленную эксплуатацию компанией ООО “Кодовый замок”

Для более подробного графика работы над проектом ниже приставлена таблица основных вех и результатов.

Таблица 1 – Основные вехи и результаты

|  |  |  |
| --- | --- | --- |
| Название вехи | Срок | Результаты |
| Начало проекта | 01.01.2021 | Подготовлено и проведено стартовое совещание |
| Проектная команда сформирована | 20.01.2021 | Этап формирования проектной команды завершен |
| Структурный анализ завершен | 10.02.2021 | - Разработана модель основных бизнес-процессов предприятия ООО «AsPowder ” по типу “как есть”; |

Окончание таблицы 1

|  |  |  |
| --- | --- | --- |
| Анализ проектирования завершен | 20.02.2021 | - Спроектированы подсистемы;  - Спроектированы классы;  - Сформированы прецеденты;  - Спроектированы БД; |
| Реализация завершена | 15.03.2021 | - Модули реализованы;  -Программный код проведен;  - Модули интегрированы |
| Тестирование завершено | 20.03.2021 | - Проведено модульное тестирование;  - Проведено функциональное тестирование;  - Проведено приемочное тестирование;  - Проведено нагрузочное тестирование; |
| Развертывание и внедрение завершено | 25.03.2021 | - Разработана документация;  - ИТ-инфраструктура закуплена;  - Пользователи обучены;  -Система развернута на рабочих местах;  - Проведены приемо-сдаточные испытания |
| Конец проекта | 05.04.2021 | Формальное закрытие проекта |

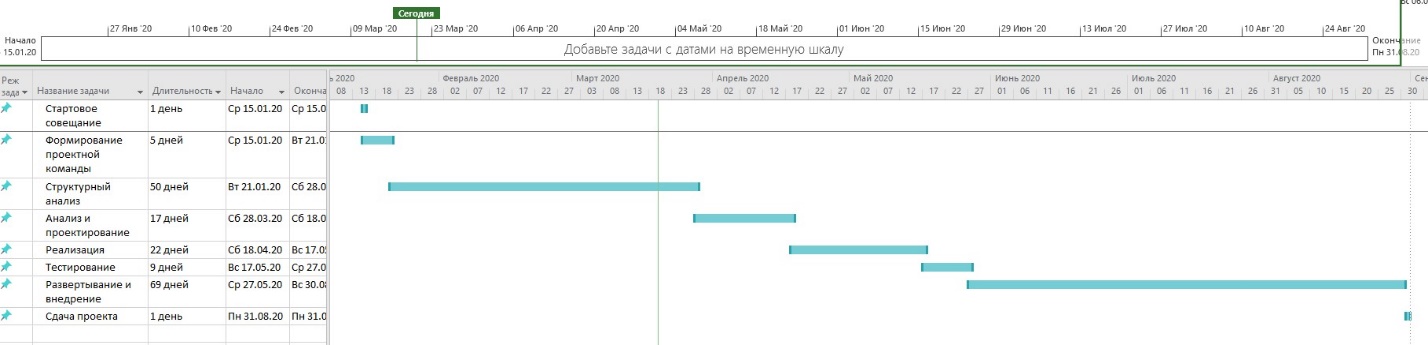


Рисунок 2 - Диаграмма Ганта для проекта

## **2.3 Оценка затрат на проект**

Оценка стоимости — процесс определения всех затрат, необходимых для успешной и полной реализации проекта.

Оценка стоимости — итеративный процесс получения примерных данных о стоимости работ и ресурсов. Оценки могут уточняться по ходу проекта. Допустимая погрешность оценок зависит от назначения получаемых данных и от фазы проекта.

Таким образом, любая оценка стоимости в проекте приблизительная. Но все же приблизительно — это от слова «близко». По ходу проекта оценки должны уточняться и становиться более реальными. На начальном этапе приблизительная оценка затрат представлена в таблице 2.

Таблица 2 – Оценка затрат на проект

|  |  |  |
| --- | --- | --- |
| № | Статья затрат | Финансирование, руб. |
| Расходы на оборудование | | |
| 1 | Провода | 50000 |
| 2 | Замок | 50000 |
| 3 | Рабочие места | 25000 |
| 4 | Сетевое оборудование | 20000 |
|  | Итого: | 100000 |

Окончание таблицы 2

|  |  |  |
| --- | --- | --- |
| Расходы на этапы проекта | | |
| 1 | Формирование проектной команды | 50000 |
| 2 | Разработка функциональной модели | 100000 |
| 3 | Разработка высокоуровневой модели | 15000 |
| 4 | Проведение структурного анализа | 100000 |
| 5 | Проектирование | 40000 |
| 6 | Реализация проекта | 35000 |
| 7 | Тестирование | 15000 |
| 8 | Развертывание и внедрение | 65000 |
| 9 | Завершение проекта | 35000 |
|  | Итого: | 340000 |

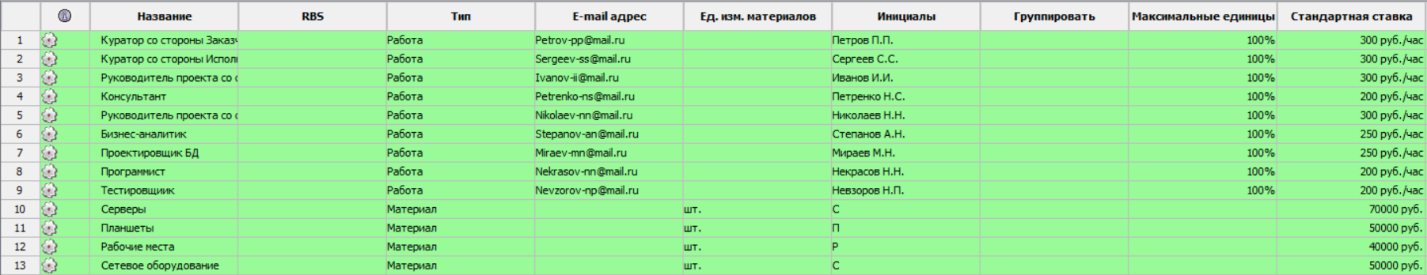


Рисунок 3 - Список ресурсов проекта

## **2.2 Глоссарий**

Глоссарий – это набор терминов, составленный для конкретного проекта. Наличие словаря помогает добиться целостности содержания. Глоссарий определяет, какие термины переводятся, а какие переводятся. Каждое словарное понятие имеет небольшое определение.

В нашем случае глоссарий будет выглядеть как в Таблице 1. В нем представлены основные термины и понятия, которые сопровождают весь курсовой проект.

Таблица 3 – Глоссарий

|  |  |
| --- | --- |
| Пользователь | Лицо, использующие данное программное обеспечение |
| Контроль | Набор действий по контролю ПО: введение нового кода, сбрасывание |
| Вызов | Сигнал подающий звонок в помещение |
| Набор кода | Поле ввода кода для входа в помещение |
| Сброс и закрытие замка | После входа в помещение, система автоматически сбрасывается и закрывается |
| Меню | Интерфейс ПО |

# **3 Проектирование системы**

## **3.1** **Начальное проектирование**

Визуальное моделирование в UML можно представить, как процесс, переходящий от наиболее общей и абстрактной концептуальной модели исходной системы к логической, а затем и к физической модели соответствующей программной системы. Для достижения этих целей сначала строится модель в виде так называемой диаграммы вариантов использования, которая описывает функциональное назначение системы, другими словами, что система делает в процессе своей работы. Диаграмма вариантов использования - это начальное концептуальное представление или концептуальная модель системы во время ее проектирования и разработки.

Итак, построим общую диаграмму вариантов использования для заданного проекта. Она является исходным концептуальным представлением системы в процессе ее проектирования и разработки. Данная диаграмма состоит из актеров, вариантов использования и отношений между ними (см. Рис. 1).

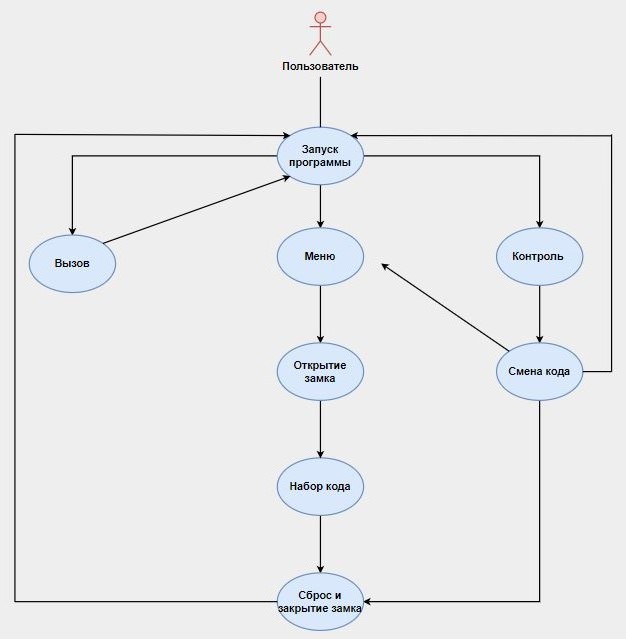


Рисунок 4 – Общая диаграмма вариантов использования

Так же в первый этап проектирования необходимо построить диаграмму деятельности. Она отражает динамические аспекты поведения системы. По существу, эта диаграмма представляет собой блок-схему, которая наглядно показывает, как поток управления переходит от одной деятельности к другой.

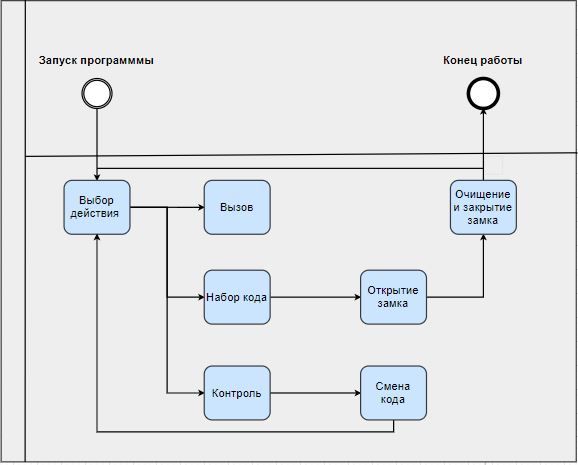


Рисунок 5 – Диаграмма деятельности

Событийная цепочка процессов (EPC, event-driven process chain) - тип диаграмм, используемых для моделирования, анализа и реорганизации бизнес-процессов (функционального моделирования). В то же время EPC-диаграммы могут использоваться для моделирования поведения отдельных частей системы при реализации функций и служить заменой традиционных блок-схем (поведенческого моделирования).

Диаграмма процесса (функции) в нотации EPC представляет собой упорядоченную комбинацию событий и функций. Для каждой функции могут быть определены начальные и конечные события, участники, исполнители, материальные и информационные потоки, сопровождающие её, а также проведена декомпозиция на более низкие уровни.

В нашем случае диаграмма процессов будет показана на рисунке 3.

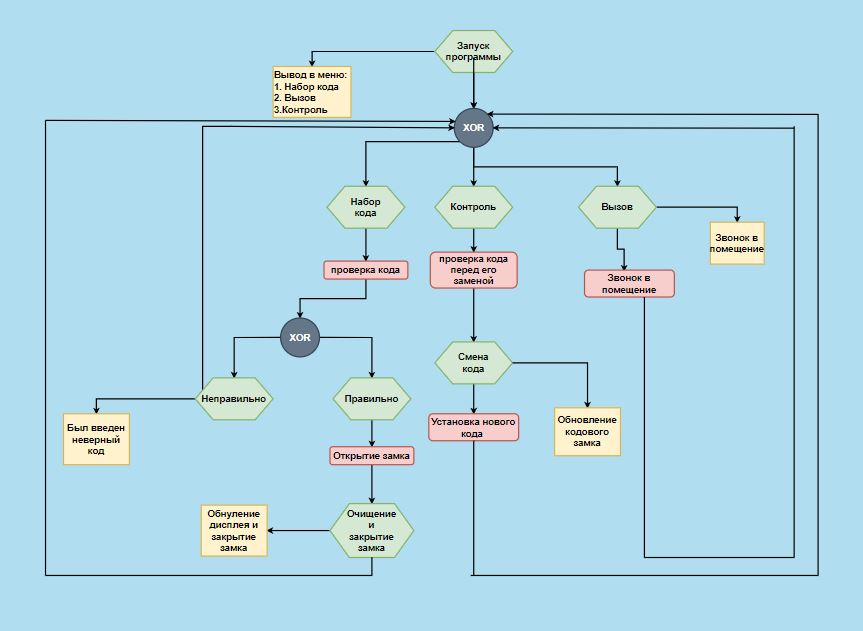


Рисунок 6 – Диаграмма процесса

## **3.2 Построение иерархии классов**

Диаграмма классов используется для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. В частности, диаграмма классов может отражать различные отношения между сущностями предметной области, такими как объекты и подсистемы, а также описывать их внутреннюю структуру и типы отношений. Эта диаграмма не дает информации о временных аспектах работы системы.

Диаграмма классов – это определенный граф, вершинами которого являются элементы типа «классификатор», связанные различными типами структурных отношений. Следует отметить, что схема класса может также включать интерфейсы, пакеты, отношения и даже отдельные экземпляры, такие как объекты и отношения. Когда говорят об этой схеме, имеют в виду статическую структурную модель проектируемой системы. Следовательно, логическая модель системы схемы классов считается графическим представлением структурных взаимосвязей, которые являются независимыми или постоянными во времени.

Таблица 4 – Класс Набор кода

|  |  |  |
| --- | --- | --- |
| Имя функции | Возвращаемый тип | Описание |
| useAlphabet | Integer | Использует алфавит 0-9 |
| lenghtCode | Integer | Имеет 4-ч кодовую длину |
| openDoor | Integer | Открытие двери |

Таблица 5 – Класс Вызов

|  |  |  |
| --- | --- | --- |
| Имя функции | Возвращаемый тип | Описание |
| ringDoor | String | Проигрывает звонок в помещение |
| BacktoMenu | String | Возвращает в меню |

Таблица 6 – Класс Контроль

|  |  |  |
| --- | --- | --- |
| Имя Атрибута | Тип | Описание |
| LastCode | Bool | Прошлый код |
| Имя фукнции | Возвращаемый тип | Описание |
| makeNewCode | Bool | Задает новый код |
| ResetCode |  | Сбрасывает предыдущий код |

Полная диаграмма классов изображена на рисунке 5.

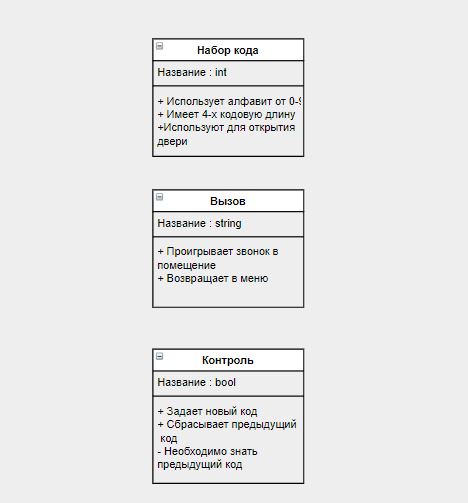


Рисунок 7 – Диаграмма классов

## **3.4 Функциональной проектирование**

Нотация **IDEF0** обычно используется для описания процессов верхнего уровня, хотя и позволяет описать всю деятельность компании. Особенностью метки является возможность отображения не только входов и выходов каждого блока, но также «органов управления» и «механизмов». Например, не всегда ясно, что технические стандарты и спецификации следует относить к «руководству», но не следует относить описания должностных обязанностей и руководителей производства. Споры возникают и по поводу «механизмов» управления процессами, поскольку каждый профессионал имеет тенденцию интерпретировать это понятие по-своему.

Диаграммы IDEF3 чаще используются для создания процессов нижнего уровня, их также можно использовать для декомпозиции блоков процессов IDEF0. В отличие от IDEF0, эта нотация не поддерживает отображение «механизмов» и «элементов управления», но указывает порядок, в котором выполнялся персонал.

Таким образом в данном проекте необходимо при построение данных диаграмм учесть следующие требования:

• модель должна отражать весь указанный в описании функционал, а также четко отражать существующие потоки данных и описывать правила их движения;

• наличие в модели не менее трех уровней;

• не менее двух уровней декомпозиции в стандарте IDEF0

• на диаграмме 1-го уровня (A0) не менее 4-х функциональных блоков;

• на диаграмме 2-го и далее уровнях должна быть декомпозиция в стандарте IDEF3, на каждой диаграмме не менее 2-х функциональных блоков.

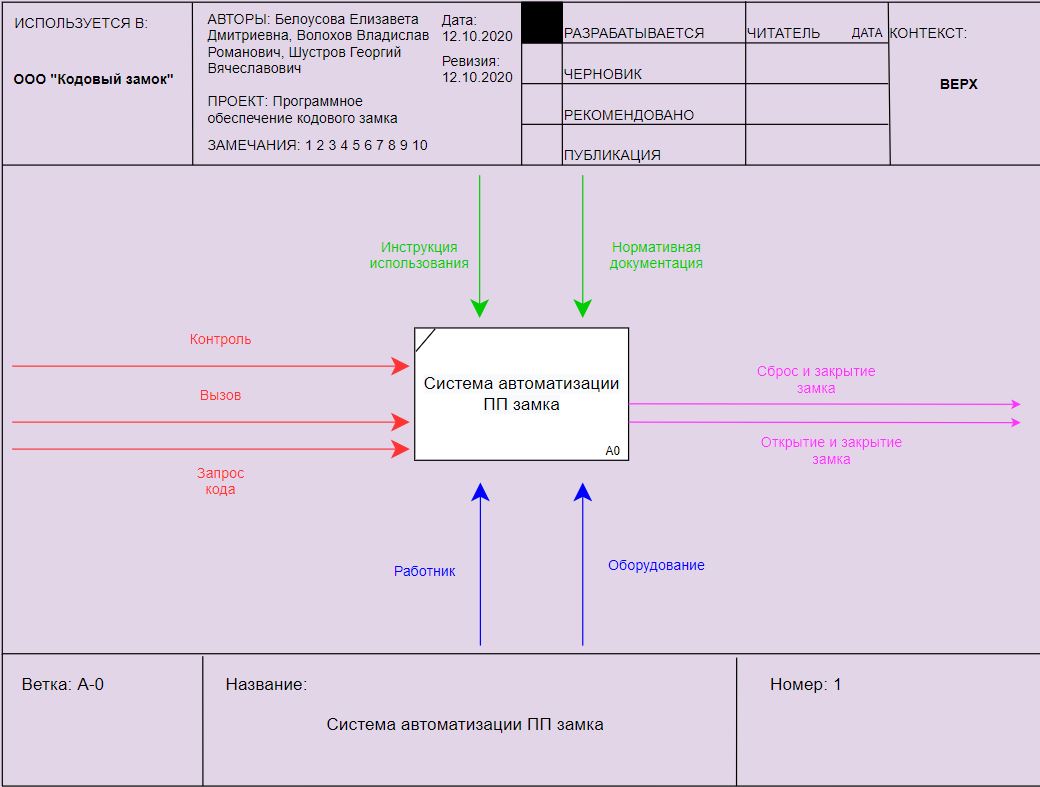


Рисунок 8 – Диаграмма IDEF0

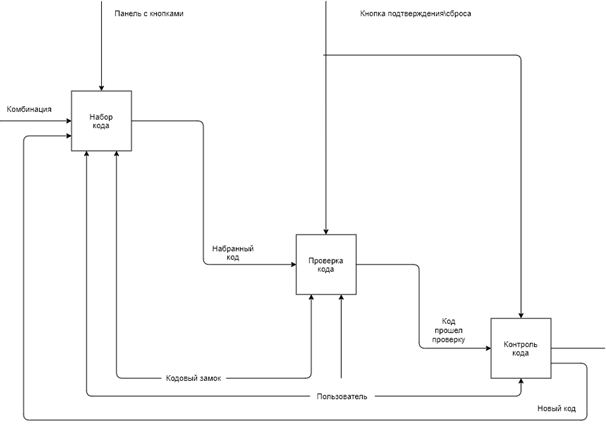


Рисунок 9 – Диаграмма IDEF0

Для построения диаграммы IDEF3 был использован следующий рисунок:



Рисунок 10 – Дополнительная информация

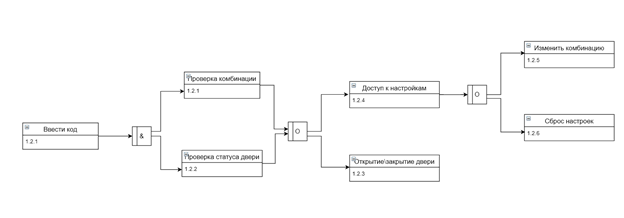


Рисунок 11 – Диаграмма IDEF3

# **Заключение**

Задачей курсового проекта было спроектировать систему «Кодовый замок». В качестве результата проект спроектирован с использованием CASE-технологий. Для решения этих задач при описании поведения проектируемого программного обеспечения использовался UML (Unified Modeling Language) – унифицированный язык моделирования.

Разработанный проект включает в себя след диаграммы:

* 1 глоссарий;
* 1 диаграмма вариантов использования;
* 1 диаграмма деятельности;
* 1 диаграмм последовательностей;
* 1 диаграмма процессов;
* 1 диаграмма классов;
* 1 диаграмма IDEF0;
* 1 диаграмма IDEF3.

Итого система «Кодовый замок» полностью спроектирована и готова к дальнейшему использованию.

**Список использованных источников**

1. Разработка, анализ и управление программными проектами [Методическое пособие] методические указания по выполнению курсового проекта для студентов всех форм обучения направления 09.03.04 (231000.62) Программная инженерия Краснодар: Изд. ФГБОУ ВПО «КубГТУ», 2015. - 28с.
2. Гагарина Л.Г., Кокорева Е.В., Виснадул Б.Д. Технология разработ-ки программного обеспечения: учебное пособие / Под ред. Л.Г. Гагариной. – М.: ИД «Форум»: Инфра-М, 2013. – 400 с.
3. Диаграмма размещения [Электронный ресурс] URL: <http://www.maksakov-sa.ru/ModelUML/DaigrRazm/index.html> (дата обращения: 01. 12.2020);
4. Диаграммы пакетов [Электронный ресурс] URL: <https://itteach.ru/rational-rose/diagrammi-paketov-komponentov-i-razmescheniya> (дата обращения: 03.12.2020);
5. Диаграмма компонентов [Электронный ресурс] URL: <https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema15/tema15_2> (дата обращения: 03.12.2020);
6. Диаграмма деятельности [Электронный ресурс] URL: <https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema14/tema14_3> (дата обращения: 13.12.2020).

# **Приложение А - Оригинальность работы**

Проверка была выполнена на официальном сайте Антиплагиат.

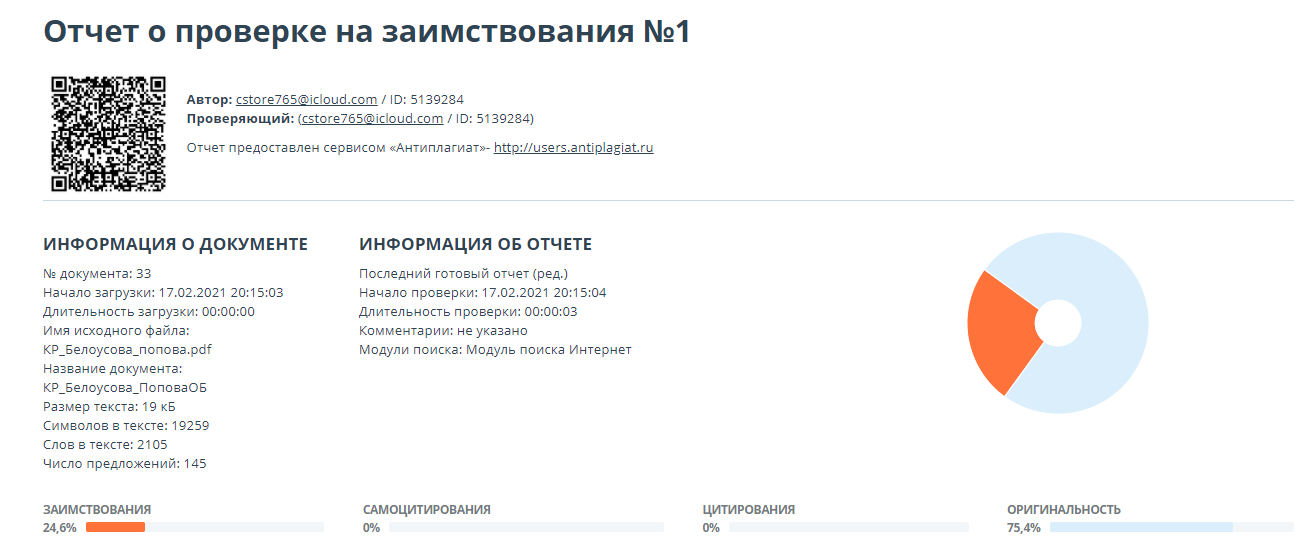


Рисунок 12 – Результат проверки