

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

**«Дальневосточный федеральный университет»**

(ДВФУ)

**ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ**

**(ШКОЛА)**

**Департамент программной инженерии и искусственного интеллекта**

ТКАЧЕВ АНДРЕЙ ВЛАДИМИРОВИЧ

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ РАБОТЫ СО СПРАВОЧНИКАМИ ПРЕДМЕТНОЙ ОБЛАСТИ «ВЕТЕРИНАРНАЯ КЛИНИКА»

**КУРСОВОЙ ПРОЕКТ**

по дисциплине «Фундаментальные структуры данных и алгоритмы»

по образовательной программе подготовки бакалавров по направлению

09.03.04 «Программная инженерия», профиль «Программная инженерия»

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | |  |  | Студент группы Б9123-09.03.04 | | | | | | | |
|  |  |  | | | | Ткачев А.В. | | | |
|  | | | | | | |  |  | (подпись) | | | |  | | | |
|  | | | | | | | | | | | | | | | | |
|  | | | | | | |  |  | Руководитель | | | | | | | |
|  | | | | | | |  |  | ассистент ДПИиИИ | | | | | | | |
|  | | | | | | |  |  | ученая степень, должность | | | | |  | Решетнев Н.Я. | |
|  | | | | | | |  |  | (подпись) | | | | |  | (ФИО) | |
|  | | | | | | | | | | | | | | | | |
| Регистрационный № | | | | | | |  |  | Защищен с оценкой | | | | | | | |
|  | | | | | | |  |  |  | | | | | | | |
|  | | | |  |  | |  |  | « |  | » |  |  | | | 2025 г. |
| (подпись) | | | |  | (ФИО) | |  |  |  | | | | | | | |
| « |  | » |  | | | 2025 г. |  |  |  | | | | | | | |

г. Владивосток

2025

# Оглавление

[Оглавление 2](#_Toc203082564)

[Введение 3](#_Toc203082565)

[1 Анализ предметной области 4](#_Toc203082566)

[1.1 Объект предметной области 4](#_Toc203082567)

[1.2 Законы ПО 8](#_Toc203082568)

[1.3 Постановка задачи «Формирование отчета «Посещение питомца»» 8](#_Toc203082569)

[2 Теоретическая часть 9](#_Toc203082570)

[2.1 Хеш-таблица 10](#_Toc203082571)

[2.1.1 Хеш-функция 10](#_Toc203082572)

[2.1.2 Разрешение коллизий методом открытая адресация 10](#_Toc203082573)

[2.1.3 Указываете, какая цепочка используется в ХТ (если цепочка только в дереве, то этот пункт должен быть ниже после деревьев) 11](#_Toc203082574)

[2.1.4 Пример хеш-таблицы на данных из ПО 12](#_Toc203082575)

[2.2 Указываете, какое дерево используется 12](#_Toc203082576)

[2.2.1 Указываете, какая цепочка используется в БДП 12](#_Toc203082577)

[2.2.2 Пример дерева на данных из ПО 12](#_Toc203082578)

[3 Требования к информационной системе 13](#_Toc203082579)

[3.1 Требования к данным 13](#_Toc203082580)

[3.1.1 Требования к входным данным 13](#_Toc203082581)

[3.1.2 Требования к выходным данным 13](#_Toc203082582)

[3.2 Функциональные требования 14](#_Toc203082583)

[3.2.1 Общие требования 14](#_Toc203082584)

[3.2.2 Требования для работы со справочником «Питомцы» 14](#_Toc203082585)

[3.2.3 Требования для работы со справочником «Приемы» 15](#_Toc203082586)

[4 Реализация 16](#_Toc203082587)

[4.1 Спецификация структур данных 16](#_Toc203082588)

[4.2 Описание среды разработки 16](#_Toc203082589)

[4.3 Руководство пользователя 17](#_Toc203082590)

[4.3.1 Работа со справочником «\_\_\_\_\_\_\_» 17](#_Toc203082591)

[4.4 Тестирование 17](#_Toc203082592)

[Заключение 19](#_Toc203082593)

[Список литературы 20](#_Toc203082594)

# Введение

ё

Ветеринарные клиники — это специализированные учреждения, занимающиеся приемом и лечением животных, проведением вакцинаций, хирургических операций, ведением медицинских карт пациентов, а также решением задач, таких как запись на прием и учет лекарственных препаратов.

Значительную часть работы клиник составляет документооборот. Ручная обработка данных требует много времени и усилий, поэтому оптимизация этих процессов — важный шаг к повышению общей эффективности работы учреждения.

В связи с этим возникает необходимость в создании программной системы, автоматизирующей управление данными в ветеринарных клиниках. Такой подход предполагает формализацию и структурирование информации, что значительно упрощает и ускоряет работу персонала. В рамках проекта будет разработан удобный интерфейс для взаимодействия с системой, а также реализованы функции хранения данных о пациентах и приемах.

Целью курсового проекта является: разработка информационной системы для работы со справочниками предметной области «Ветеринарная клиника».

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Провести анализ предметной области «Ветеринарная клиника».

2. Изучить теоретические основы методов построения справочников.

3. Определить требования к информационной системе.

4. Реализовать информационную систему и провести тестирование.

# 1 Анализ предметной области

Требуется разработать информационную систему для предметной области (ПО) ‹‹Ветеринарная клиника››.

Система должна решать следующие задачи:

1. хранить, позволять пересматривать, добавлять и удалять информацию о питомцах (Кличка; Вид; Владелец) их приемах в ветеринарную клинику (Кличка; Владелец; Диагноз; Врач; Дата приема).
2. позволять искать информацию о питомцах (Кличка;  Вид; Владелец) по имени владельца питомца и приёмах в клинику (Кличка; Владелец; Диагноз; Врач; Дата приема) по имени врача;
3. формировать и позволять просматривать список приёмов питомцев в клинике ;
4. проверять целостность информации, представленной в справочниках.

1.1 Объект предметной области

Исходя из задач, которые должна решать информационная система, выделены объекты ПО, представленный в таблице 1.

|  |  |
| --- | --- |
| Тип и название объекта | Описание объекта |
| Справочник ‹‹Питомцы›› | Хранит информацию по каждому питомцу |
| Справочник ‹‹Приемы›› | Хранит информацию по каждому приему в ветеринарной клинике |
| Отчет ‹‹Посещение питомца›› | Хранит информацию по каждому питомцу и посещенному им приему |

Таблица 1 – Объекты ПО

Каждый питомец характеризируется следующими параметрами: кличка, вид, владелец.

**Кличка** ̶ слово, состоящее из букв русского алфавита, первая буква ̶ заглавная, остальные ̶ строчные.

**Вид** – слово, полностью состоящее из строчных букв.

**Владелец**  ̶ строка, состоящая из нескольких слов, которые состоят из букв русского алфавита, первые буквы каждого слова ̶ заглавные, остальные ̶ строчные.

Пример справочника ‹‹Питомцы›› представлен в таблице 2.

Таблица 2 – Справочник ‹‹Питомцы››.

|  |  |  |
| --- | --- | --- |
| **Кличка** | **Вид** | **Владелец** |
| Тоша | попугай | Иванов Иван Сергеевич |
| Пуфик | собака | Федоров Николай Вадимович |
| Василек | кот | Степанов Олег Сергеевич |
| Комочек | шиншилла | Иванов Иван Сергеевич |
| Барсик | кошка | Соколова Анна Михайловна |
| Рекс | собака | Волков Дмитрий Игоревич |
| Муся | кошка | Николаева Татьяна Петровна |
| Грей | кот | Кузнецов Павел Олегович |
| Шарик | собака | Морозов Сергей Владимирович |
| Зефир | кролик | Ковалева Екатерина Дмитриевна |

Каждый прием характеризируется следующими параметрами: кличка, владелец, диагноз, врача, дата приема.

**Кличка** ̶ слово, состоящее из букв русского алфавита, первая буква ̶ заглавная, остальные ̶ строчные.

**Владелец**  ̶ строка, состоящая из нескольких слов, которые состоят из букв русского алфавита, первые буквы каждого слова ̶ заглавные, остальные ̶ строчные.

**Диагноз**  ̶ строка, состоящая из одного или несколько слов. Каждое слово состоит из заглавных или прописных букв.

**Врач**  ̶ строка, состоящая из нескольких слов, которые состоят из букв русского алфавита, первые буквы каждого слова ̶ заглавные, остальные ̶ строчные.

**Дата приема** – строка, формата ‹‹DD месяц YYYY››, где DD – день (целое число со значением от 1 до 31), месяц - x { “январь”, “февраль”, “март”, “апрель”, “мая”, “июнь”, “июль”, “август”, “сентябрь”, “октябрь”, “ноябрь”, “декабрь” }, YYYY – число от 2000 до 2025.

Пример справочника ‹‹Приемы›› представлен в таблице 3.

Таблица 3 – Справочник ‹‹Приемы››.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Кличка** | **Владелец** | **Диагноз** | **Врач** | **Дата приема** |
| Тоша | Иванов Иван Сергеевич | Травма крыла | Семёнова Мария Петровна | 21 февраль 2018 |
| Пуфик | Федоров Николай Вадимович | Аллергия | Петров Алексей Дмитриевич | 16 март 2019 |
| Василёк | Степанов Олег Сергеевич | Конъюнктивит | Козлова Елена Викторовна | 24 март 2019 |
| Комочек | Иванов Иван Сергеевич | Профилактика | Смирнов Денис Олегович | 05 апрель 2019 |
| Барсик | Соколова Анна Михайловна | Стрижка | Андреева Ольга Сергеевна | 10 сентябрь 2020 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Кличка** | **Владелец** | **Диагноз** | **Врач** | **Дата приема** |
| Рекс | Волков Дмитрий Игоревич | Перелом | Фёдоров Артём Викторович | 30 октябрь 2020 |
| Муся | Николаева Татьяна Петровна | Кастрация | Григорьева Ирина Алексеевна | 14 ноябрь 2021 |
| Грей | Кузнецов Павел Олегович | Годовой осмотр | Семёнов Игорь Дмитриевич | 02 декабрь 2021 |
| Шарик | Морозов Сергей Владимирович | Вакцинация | Лебедева Наталья Ивановна | 15 январь 2022 |
| Зефир | Ковалёва Екатерина Дмитриевна | Чистка зубов | Власов Михаил Сергеевич | 08 февраль 2022 |

Каждая запись отчета характеризуется следующими полями: кличка, вид, владелец, диагноз, врача, дата приема.

**Кличка** ̶ слово, состоящее из букв русского алфавита, первая буква ̶ заглавная, остальные ̶ строчные.

**Вид** – слово, полностью состоящее из строчных букв.

**Владелец**  ̶ строка, состоящая из нескольких слов, которые состоят из букв русского алфавита, первые буквы каждого слова ̶ заглавные, остальные ̶ строчные.

**Диагноз** – строка, состоящая из одного или несколько слов. Каждое слово состоит из заглавных или прописных букв.

**Врач**  ̶ строка, состоящая из нескольких слов, которые состоят из букв русского алфавита, первые буквы каждого слова ̶ заглавные, остальные ̶ строчные.

**Дата приема** – строка, формата ‹‹DD месяц YYYY››, где DD – день (целое число со значением от 1 до 31), месяц - x { “январь”, “февраль”, “март”, “апрель”, “мая”, “июнь”, “июль”, “август”, “сентябрь”, “октябрь”, “ноябрь”, “декабрь” }, YYYY – число от 2000 до 2025.

Пример отчета ‹‹Посещение питомца›› представлен в таблице 4.

Таблица 4 – Отчет ‹‹Приемы питомцев››.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Кличка** | **Вид** | **Владелец** | **Диагноз** | **Врач** | **Дата приема** |
| Тоша | попугай | Иванов Иван Сергеевич | Травма крыла | Семёнова Мария Петровна | 21 февраль 2018 |
| Пуфик | собака | Федоров Николай Вадимович | Аллергия | Петров Алексей Дмитриевич | 16 март 2019 |
| Василёк | кошка | Степанов Олег Сергеевич | Конъюнктивит | Козлова Елена Викторовна | 24 март 2019 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Кличка** | **Вид** | **Владелец** | **Диагноз** | **Врач** | **Дата приема** |
| Комочек | шиншилла | Иванов Иван Сергеевич | Профилактика | Смирнов Денис Олегович | 05 апрель 2019 |
| Барсик | кошка | Соколова Анна Михайловна | Стрижка | Андреева Ольга Сергеевна | 10 сентябрь 2020 |
| Рекс | собака | Волков Дмитрий Игоревич | Перелом | Фёдоров Артём Викторович | 30 октябрь 2020 |
| Муся | кошка | Николаева Татьяна Петровна | Кастрация | Григорьева Ирина Алексеевна | 14 ноябрь 2021 |
| Грей | кот | Кузнецов Павел Олегович | Годовой осмотр | Семёнов Игорь Дмитриевич | 02 декабрь 2021 |
| Шарик | собака | Морозов Сергей Владимирович | Вакцинация | Лебедева Наталья Ивановна | 15 январь 2022 |
| Зефир | кролик | Ковалёва Екатерина Дмитриевна | Чистка зубов | Власов Михаил Сергеевич | 08 февраль 2022 |

1.2 Законы ПО

1. Данные о приёме не могут существовать без соответствующей записи о питомце.
2. Питомец однозначно идентифицируется парой (Кличка, Владелец).
3. Отчёт о посещениях должен содержать полную историю приёмов для выбранного питомца.
4. Дата приёма не может быть будущей относительно текущей даты системы и должна соответствовать формату
5. Каждый прием одного и того же питомца может иметь как одинаковый, так и разный диагноз, или диагноз может вообще отсутствовать.
6. Даты приёмов для одного питомца должны идти в хронологическом порядке

1.3 Постановка задачи «Формирование отчета «Посещение питомца»»

Входные данные: Справочник «Питомцы», Справочник «Приёмы», дата начала периода, дата конца периода, Владелец и диагноз.

Выходные данные: Отчет «Приёмы питомцев».

Связь:

Отчет «Приёмы питомцев».Врач = Справочник «Приёмы».Врач

Отчет «Приёмы питомцев».Дата = Справочник «Приёмы».Дата

Отчет «Приёмы питомцев».Кличка = Справочник «Питомцы».Кличка

Отчет «Приёмы питомцев».Владелец = Справочник «Питомцы».Владелец

Отчет «Приёмы питомцев».Вид = Справочник «Питомцы».Вид

Отчет «Приёмы питомцев».Диагноз = Справочник «Приёмы».Диагноз

Если существует запись в справочнике «Приёмы»:

Справочник «Приёмы».Дата приёма >= дата конца периода &&

Справочник «Приёмы».Дата приёма <= дата конца периода&&

Справочник «Приёмы».Владелец = заданный владелец &&

Справочник «Приёмы».Диагноз = Диагноз &&

Причем должна существовать запись в справочнике «Питомцы»:

Справочник «Питомцы».Кличка = Справочник «Приёмы».Кличка

Справочник «Питомцы».Владелец = Справочник «Приёмы».Владелец

# 2 Теоретическая часть

В рамках курсового проекта необходимо осуществлять поиск информации о приёмах питомцах на основании клички питомца и его владельца.

Для хранения данных о враче в Справочнике «Питомцы» будет использоваться структура данных «Хеш-таблица», исходя из того, что пара «Кличка – владелец» должны быть уникальны в справочнике, и необходимо обеспечить быстрое выполнение операций поиска, добавления, удаления.

Для хранения данных о приёме в Справочнике «Приёмы» будет использоваться структура данных «АВЛ-дерево», так как питомец может иметь несколько приёмов, и необходимо обеспечить быстрое выполнение операций поиска, добавления, удаления.

2.1 Хеш-таблица

Хеш-таблица представляет собой эффективную структуру данных для реализации словарей. Хеш-таблица обобщает обычный массив. Возможность прямой индексации элементов массива обеспечивает доступ к произвольному элементу массива за время O(1). Позволяет выполнять операции поиска, вставки и удаления[Кормен.А ].

2.1.1 Хеш-функция

Теоретический материал со ссылками на литературу, что такое ХФ [номер из списка литературы].

Теоретический материал про метод хеширования, который будете использовать.

Пример на тех записях, которые приводили в анализе ПО (подобрать такой, где значения ХФ одинаковые) -> подводите к коллизии + указываете, какие методы разрешения коллизий существуют и какой вы будете использовать.

2.1.2 Разрешение коллизий методом открытая адресация

Открытая адресация — это метод разрешения коллизий, при котором все элементы хранятся непосредственно в самой хеш-таблице. Каждая ячейка в этой хеш-таблице может находиться в одном из трех состояний:

Пустая (статус 0)

Занятая (статус 1)

Удаленная (статус 2)

При поиске элемента последовательно исследуются (пробируются) ячейки таблицы до тех пор, пока не будет найден искомый элемент или пустая ячейка.

Для вставки при открытой адресации последовательно исследуются ячейки хеш-таблицы до тех пор, пока не будет найдена пустая ячейка для размещения вставляемого ключа. Вместо фиксированного порядка исследования ячеек, последовательность исследуемых ячеек зависит от вставляемого ключа.

Особенность операции удаления в открытой адресации заключается в том, что простое опустошение слота после удаления ключа может нарушить корректность последующих операций поиска. Если ключ, который был удален, находился на пути пробирования для другого ключа (то есть, поиск другого ключа должен был пройти через этот слот), то после его удаления поиск мог бы ошибочно остановиться, посчитав, что искомый ключ отсутствует.

Для предотвращения таких ошибок, удаленные слоты специально помечаются как "удаленные". Это позволяет операции вставки использовать такой слот для размещения нового элемента. Однако, при поиске, алгоритм не останавливается на "удаленном" слоте; он продолжает пробирование так, как если бы слот был занят, пока не найдет искомый ключ или действительно пустой слот.

2.1.3 Указываете, какая цепочка используется в ХТ (если цепочка только в дереве, то этот пункт должен быть ниже после деревьев)

Теоретический материал что из себя представляет цепочка.

Рисуете пример построенной ХТ для справочников п. 1.1.

В следующем параграфе пишете про вторую вашу структуру данных – БДП, АВЛ или КЧ.

2.1.4 Пример хеш-таблицы на данных из ПО

В рамках курсового проекта в хеш-таблице хранится указываете какой ключ и значение. На рисунке номер представлен пример для справочника «Название».

Рисуете пример построенной ХТ для справочников п. 1.1, как выполняется поиск, что происходит после добавления и удаления.

2.2 Указываете, какое дерево используется

В этом параграфе пишете про БДП, АВЛ или КЧ. Если БДП с ХТ, значит переносите в 2.1.

Теоретический материал про дерево + операции работы (для удаления указать на что замена).

Рисуете пример построенного дерева для справочника п. 1.1.

2.2.1 Указываете, какая цепочка используется в БДП

Если в элементе дерева хранится цепочка, то добавится этот пункт. Теоретический материал что из себя представляет цепочка в дереве.

2.2.2 Пример дерева на данных из ПО

В рамках курсового проекта в дереве хранится указываете какой ключ и значение. На рисунке номер представлен пример для справочника «Название».

Рисуете пример построенной дерева для справочников п. 1.1, как выполняется поиск, что происходит после добавления и удаления.

И аналогично для дерева, которое используется при фильтрации для формирования отчета.

# 3 Требования к информационной системе

В данной главе описываются требования к информационной системе «Ветеринарная клиника», а именно: требования к входным данным системы, требования к ее выходным данным и требования к ее функционалу.

3.1 Требования к данным

3.1.1 Требования к входным данным

Основываясь на анализе ПО, входными данными является:

* текстовый файл «Pets.txt», каждая строка которого содержит информацию о питомце из Справочника «Питомцы»: Кличка, Владелец, Вид, разделенные символом «;»;

Пример текстового файла:

#-----------------------------------------------------------------------------------------#

* текстовый файл «Visit.txt», каждая строка которого содержит информацию о приёме из Справочника «Приемы»: Кличка, Владелец, Диагноз, Врач, Дата разделенные символом «;»;

Пример текстового файла:

#-----------------------------------------------------------------------------------------#

3.1.2 Требования к выходным данным

Выходными данными являются:

* текстовый файл \_\_\_\_, каждая строка файла содержит информацию об объекте \_\_\_\_\_, а именно \_\_\_\_\_, разделенный символом \_\_\_\_;
* сообщения об ошибках (каких).

У вас на выходе два текстовых файла, аналогично и для второго.

3.2 Функциональные требования

3.2.1 Общие требования

Информационная система «Ветеринарная клиника» должна позволять:

1. считывать данные для справочника «Питомцы» из текстового файла «Pets.txt» в массив, который выводиться в виде таблицы в графическом интерфейсе;
2. считывать данные для справочника «Приёмы» из текстового файла «Visits.txt» в массив, который выводиться в виде таблицы в графическом интерфейсе
3. сохранять данные справочника «Питомцы» в текстовый файл «changePets.txt»;
4. сохранять данные справочника «Приемы» в текстовый файл «changeVisits.txt»
5. работать со справочниками «Питомцы» и «Приемы» при помощи графического интерфейса;
6. проверять корректность вводимых пользователем данных для справочников «Питомцы» и «Приемы» (см. п. 1.1);
7. формировать список для отчёта «Посещение питомца» и выводить в окно графического интерфейса;
8. проверять целостность данных, хранимых в справочниках.

3.2.2 Требования для работы со справочником «Питомцы»

Информационная система «Юридическая фирма» должна позволять:

* 1. хранить информацию о питомцах: Кличка, Владелец, Вид;
  2. просматривать информацию о питомцах, используя графический интерфейс информационной системы;
  3. осуществлять поиск информации о питомце на основе его клички и владельце. Поиск должен осуществляться через графический интерфейс с помощью полей для ввода данных. Если в справочнике отсутствует искомая запись, программа возвращает соответствующее сообщение. Иначе выводится сообщение о нахождении нужного клиента;
  4. добавлять информацию о питомце в справочник «Питомцы». Добавление должно осуществляться через графический интерфейс с помощью полей для ввода данных. Каждая запись уникальна, при наличии добавляемой записи в справочнике выводится сообщение об ошибке;
  5. удалять информацию о питомце из справочника «Питомцы». Удаление должно осуществляться через графический интерфейс с помощью полей для ввода данных. После удаления должен быть проверен справочник «Приемы» на наличие записей с удалённым питомцем, в случае нахождения записи должны быть удалены. Если в справочнике «Питомцы» изначально не был найден удаляемый клиент, система возвращает соответствующее сообщение;

3.2.3 Требования для работы со справочником «Приемы»

Информационная система «Юридическая фирма» должна позволять:

1. хранить информацию о посещениях: Кличка, Владелец, Диагноз, Врач, Дата;
2. просматривать информацию о посещениях, используя графический интерфейс информационной системы;
3. осуществлять поиск информации о посещениях на основе клички питомца и владельце. Поиск должен осуществляться через графический интерфейс с помощью полей для ввода данных. Если в справочнике отсутствует искомая запись, программа возвращает соответствующее сообщение. Иначе выводится сообщение о нахождении нужного клиента;
4. добавлять информацию о посещение в справочник «Приемы». Добавление должно осуществляться через графический интерфейс с помощью полей для ввода данных. Каждая запись уникальна, при наличии добавляемой записи в справочнике выводится сообщение об ошибке. Если в справочнике «Питомцы» не существует клиента, указанного в добавляемой консультации, возвращается сообщение об ошибке;
5. удалять информацию о посещение из справочника «Приёмы». Удаление должно осуществляться через графический интерфейс с помощью полей для ввода данных. Если в справочнике «Приёмы» изначально не был найден удаляемый клиент, система возвращает соответствующее сообщение;

# 4 Реализация

В данной главе содержится спецификация реализованных структур данных, описание графического интерфейса, а также приведены результаты тестирования структур данных.

4.1 Спецификация структур данных

Тут описание СД (можно в табличном виде)

Класс \_\_\_\_ - класс, описывающий \_\_\_\_.

Поля:

* перечисляете поля с комментариями

Методы:

* перечисляете методы – заголовок, назначение, список формальных параметров, входные данные, выходные данные

Для класса, реализующего хеш-таблицу, в методе для хеш-функции приводите пример значений для ключей из анализа.

4.2 Описание среды разработки

Для реализации информационной системы «Название» была выбрана среда разработки указываете какая. Язык реализации указываете какой.

4.3 Руководство пользователя

При запуске приложения указываете какой файл запускается, какое стартовое окно открывается со ссылками на рисунки.

4.3.1 Работа со справочником «\_\_\_\_\_\_\_»

Что нужно сделать (какие кнопочки нажать) для каждой функции из функциональных требований с рисунками.

4.4 Тестирование

Тестируете отдельно работу СД и отдельно работу со справочником.

Тестирование проводилось методом черного ящика, результаты которого представлены в Таблицах \_\_\_\_\_.

Таблица \_\_\_\_ – Тестирование работы со справочником «Название»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Описание тестовой ситуации | | Входные данные | | Выходные данные | |
| Ваш справочник | Ваши поля | Ваш справочник | Сообщение |
| Добавление | | | | | |
| 1 | Название тестовой ситуации |  |  |  |  |
| 2 | и т.д. |  |  |  |  |
| Удаление | | | | | |
| 3 | Название тестовой ситуации |  |  |  |  |
| 4 | и т.д. |  |  |  |  |
| Поиск | | | | | |
| 5 | Название тестовой ситуации |  |  |  |  |
| 6 | и т.д. |  |  |  |  |

Таблица \_\_\_\_ – Тестирование структуры данных \_\_\_\_\_\_

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Описание тестовой ситуации | | Входные данные | | Выходные данные | |
| Ваша структура | Ваши поля | Ваша структура | Ваши поля |
| Добавление | | | | | |
| 1 | Название тестовой ситуации |  |  |  |  |
| 2 | и т.д. |  |  |  |  |
| Удаление | | | | | |
| 3 | Название тестовой ситуации |  |  |  |  |
| 4 | и т.д. |  |  |  |  |
| Поиск | | | | | |
| 5 | Название тестовой ситуации |  |  |  |  |
| 6 | и т.д. |  |  |  |  |

Аналогично для второго справочника и для отчета.

# Заключение

Целью курсового проекта было: повторение из введения.

Цель достигнута. Для достижения поставленной цели были выполнены следующие задачи:

Перечисление задач + что было дополнительно изучено для решения каждой (например, среда разработки и т. д.).

# Список литературы

должны быть ссылки на учебники (в том числе интернет-источники), из которых брали информацию теоретической части