**Содержание**

Введение

1. Постановка задачи
   1. Описание предметной области
   2. Актуальность решаемой задачи
   3. Характеристика решаемой задачи
2. Проектирование программного продукта
   1. Разработка модели данных
   2. Выбор программного обеспечения
   3. Определение требований к техническим средствам
   4. Защита информации
3. Разработка программного обеспечения
   1. Определение формы представления входных и выходных

данных

* 1. Разработка справочной системы
  2. Тестирование программного модуля
  3. Описание разрабатываемого программного продукта

1. Экономическая часть
2. Охрана труда
3. Энерго- и ресурсосбережение

Заключение

Список используемых источников

Приложение А (обязательное) Входные и выходные формы

Приложение Б (обязательное) Текст программы

Приложение В (обязательное) Результаты тестирования

Приложение Г (обязательное) Описание программы

Приложение Д (обязательное) Документация пользователя

Приложение Е (обязательное) Расчет затрат на оплату труда и

отчислений на социальные нужды

# **Введение**

Базы данных и системы управления базами данных являются важнейшими компонентами современного модернизированного делового мира. Поскольку компании накапливают огромные объемы данных, потребность в обученных специалистах, которые могут управлять этой информацией и анализировать ее, становится все более важной. Базы данных используются в большинстве программных средств, даже в обычных браузерах, поэтому выбор базы данных как способа хранения информации в программах сейчас актуален как никогда.

В этом дипломе мы рассмотрим разработку комплексной программы «Автоматизированная система мониторинга жилого сектора», которая облегчит учет и контроль мониторинга обхода жилого сектора. В результате дипломного проектирования должны быть использованы умения и навыки, полученные в ходе преддипломной практики для реализации дипломного проекта.

Целью дипломного проектирования является разработка информационная системы «Автоматизированная система мониторинга жилого сектора»

Задачами дипломного проектирования являются:

– анализ и сбор сведений по предметной области;

– разработка пользовательского интерфейса программы;

– реализация программного модуля;

– создание структуры данных;

– написание технической документации;

– апробация разработанного программного продукта.

Для разработки ПО был выбран язык Object Pascal (Delphi).с использованием базы данных SQLite.

**1 Постановка задачи**

**1.1 Описание предметной области**

Информационная система «Автоматизированная система мониторинга жилого сектора» имеет определенные особенности среды функционирования, которые необходимо учитывать при ее создании.

Прежде всего, программа должна быть универсальной и доступной для пользователей. Для этого она должна работать на различных операционных системах семейства WINDOWS, иметь возможность удобного и быстрого доступа к информации.

Одной из ключевых особенностей среды функционирования программы является ее простота и функциональность.

Кроме того, программа должна работать в различных условиях, таких как отсутствие высокоскоростного интернета или отсутствие подключения к интернету, работа на компьютерах старых конфигураций. Поэтому для ее создания необходимо разрабатывать оптимизированный код и использовать технологии, которые позволят работать на устройствах с любыми операционными системами и разными уровнями производительности.

Обход жилого сектора сотрудниками МЧС является актуальной темой, особенно в отопительный период, когда увеличивается опасность пожаров и отравления угарным газом. Вот некоторые причины, по которым это важно:

* Профилактика пожаров.
* Обучение населения.
* Обход жилого сектора.
* Установка датчиков.

Таким образом, обход жилого сектора сотрудниками МЧС играет важную роль в профилактике пожаров и обучении населения правилам пожарной безопасности.

Положительный эффект от внедрения программы может быть замечен по улучшенным показателям отчетности по обходу жилого сектора, предупреждению ЧС, предотвращению пожаров.

**1.2 Актуальность решаемой задачи**

Актуальность разработки программы информационной системы «Автоматизированная система мониторинга жилого сектора» заключается в обеспечении контроля и учета обхода жилого сектора с целью предупреждения ЧС. Планируемые и проведенные обходы жилого сектора необходимо учитывать, чтобы исключить возможность пропустить какой-либо дом в обходе или потратить лишнее время на повторный обход уже посещенного домовладения. Для организации работы используются планы-графики обхода с указанием адреса жилого домовладения и ответственного за обход. Такой план-график печатается и выдается на руки сотруднику, который осуществляет обход.

По окончании периода обхода (день или неделя) сотрудник предоставляет информацию о тех владениях, среди жителей которых он провел профилактическую работу и данные о результативных обходах вносятся в базу данных. Таким образом данные о профилактической работе среди населения находятся постоянно в актуальном состоянии позволяя тем самым контролировать состояние информированности граждан и корректировать план-график следующих обходов, добавляя в него жителей тех владений, которые отсутствовали дома в моменты прошлых обходов.

Внедрив информационную систему, работники МЧС могут контролировать сроки проведения как плановых профилактических акций по предотвращению ЧС, совершаемых регулярно (например, осенние обходы к периоду отопления или весенние о паводках) так и внеплановые, вызванные аномальными и непредсказуемыми погодными условиями.

Разработка информационной системы «Автоматизированная система мониторинга жилого сектора» призвана упростить процессы контроля и учета обхода, автоматизировать выполняемые задачи.

**1.3 Характеристика решаемой задачи**

Цель разработки информационной системы «Автоматизированная система мониторинга жилого сектора» - предоставить пользователям удобный интерфейс для учета обходов жилого сектора с использованием современных компьютерных технологий. Функционально программа разделена на следующие основные части:

* раздел программного средства позволяющий составлять графики обхода жилого сектора;
* раздел программного средства позволяющий контролировать сроки и ответственных за обход, осуществлять поиск в базе данных;
* раздел позволяющий вносить новых сотрудников, осуществляющих обходы, редактировать их данные и удалять уволенных сотрудников;
* раздел позволяющий формировать отчеты об обходах и план-графики обходов.

В результате внедрения информационной системы «Автоматизированная система мониторинга жилого сектора» ожидается повышение эффективности сотрудников МЧС, снижение количество непредупрежденных и не проверенных жителей жилого сектора. Хранение информации централизованно в единой базе данных позволит координировать действия различных сотрудников с целью оптимизации их работы.

**2 Проектирование программного продукта**

**2.1 Разработка модели данных**

Все действия и данные, производимые программой, можно показать с помощью диаграмм UML. С помощью диаграммы вариантов использования проектируемая система представляется в виде множества сущностей или актёров, взаимодействующих с системой с помощью вариантов использования. Диаграмма вариантов использования – диаграмма, описывающая, какой функционал разрабатываемой программной системы доступен каждой группе пользователей.

Диаграмма прецедентов представлена на рисунке 1.

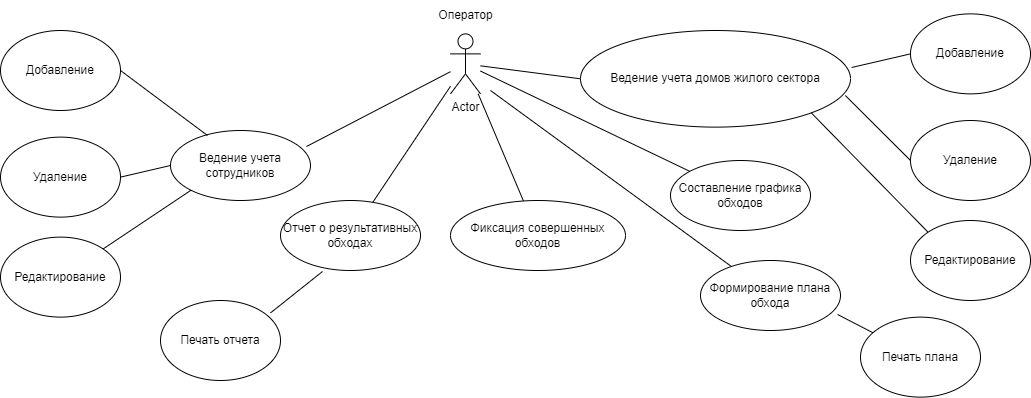


Рисунок 1 – Диаграмма вариантов использования UML

Основным средством для предоставления статических моделей являются диаграммы классов.

Статические модели обеспечивают представление структуры систем в терминах базовых строительных блоков и отношений между ними. «Статичность» этих моделей состоит в том, что здесь не показывается динамика изменений системы во времени. Вместе с тем, эти модели несут в себе не только структурные описания, но и описания операций, реализующих заданное поведение системы.

Вершины диаграммы классов нагружены классами, а дуги (ребра) – отношениями между ними. На рисунке 2 представлена диаграмма классов.

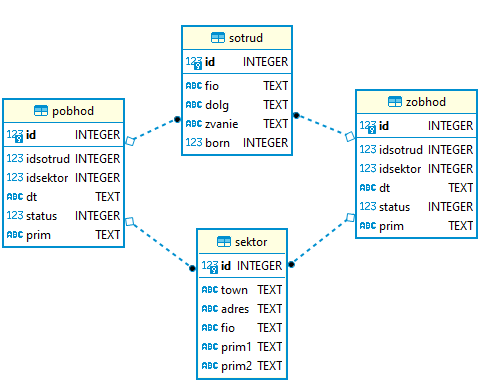


Рисунок 2 – Диаграмма классов

На рисунках 3-7 представлены диаграммы последовательностей

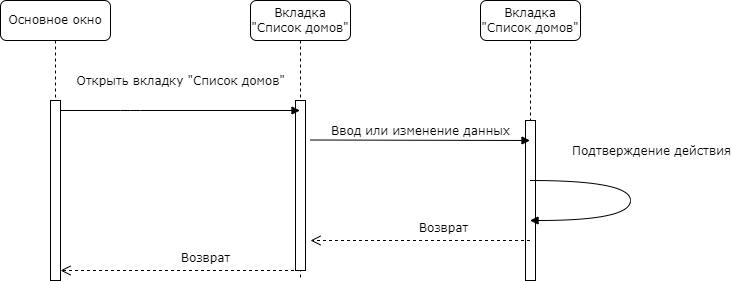


Рисунок 3 – Диаграмма последовательности ведение учета домов.

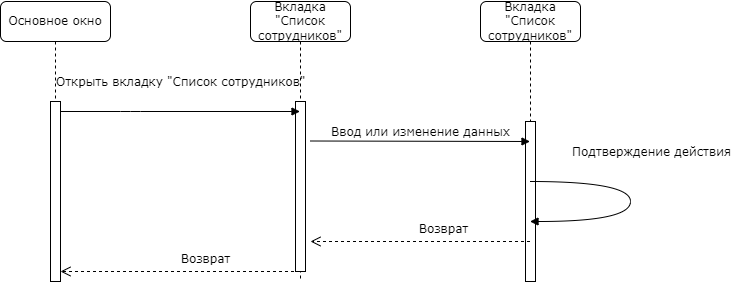


Рисунок 4 – Диаграмма последовательности ведение учета сотрудников.

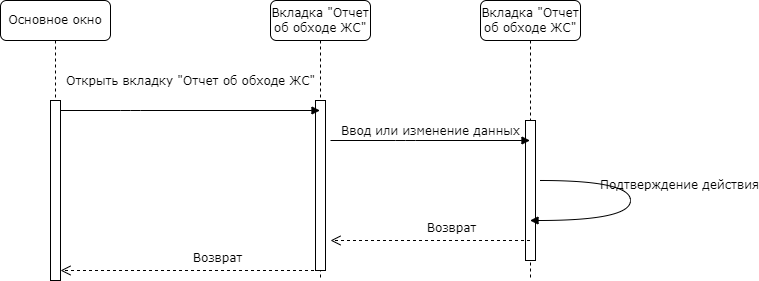


Рисунок 5 – Диаграмма последовательности отчет о результативных обходах.

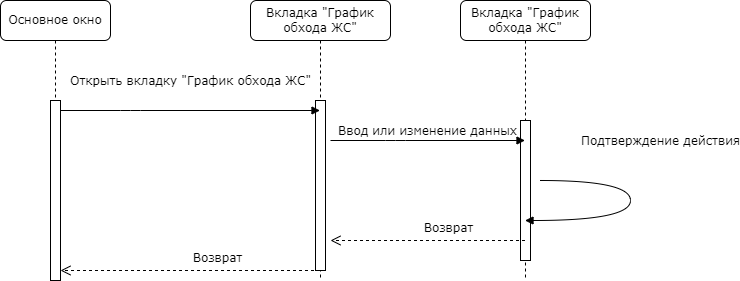


Рисунок 6 – Диаграмма последовательности составление графика обходов.

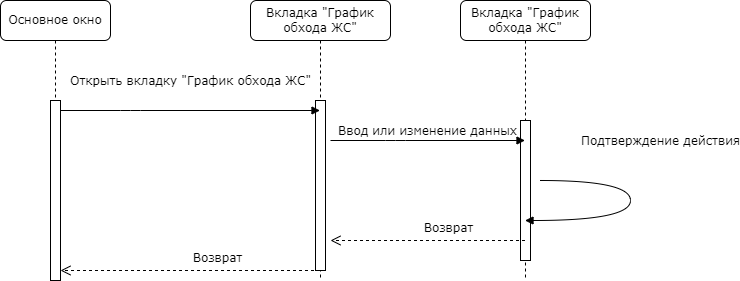


Рисунок 7 – Диаграмма последовательности формирование плана обходов.

На рисунке 8 представлена диаграмма состояний

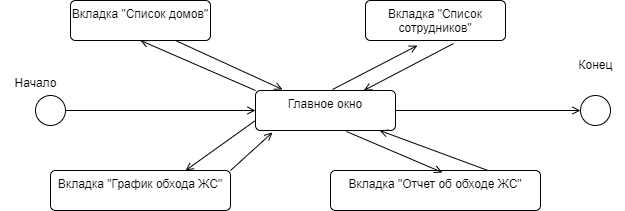


Рисунок 8 – Диаграмма состояний

На рисунке 9 представлена диаграмма компонентов.

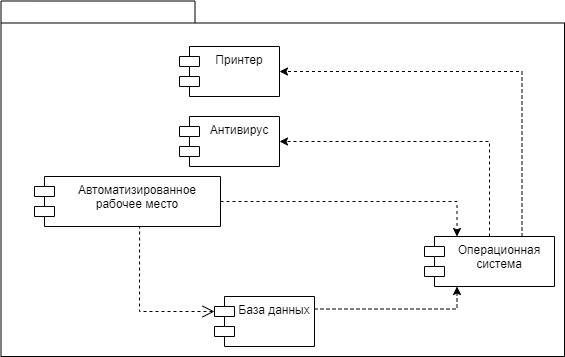


Рисунок 9 – Диаграмма компонентов

Диаграмма развертывания представлена на рисунке 10.

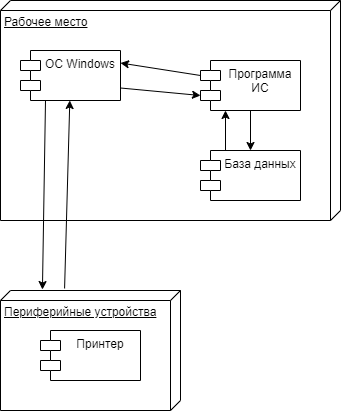


Рисунок 10 – Диаграмма развертывания

**2.2 Выбор программного обеспечения**

В качестве задания было предложено разработать автоматизированную систему мониторинга жилого сектора. Для хранения данных централизованно и облегчения резервного копирования требуется создать базу данных жилого сектора города с целью инвентаризации противопожарного состояния объектов, вынесения замечаний и предупреждений жильцам и контроля выполнения графика по обходу жилого сектора.

Для данных целей будет использоваться язык программирования Embarcadero Delphi.

Delphi — это язык программирования, основанный на [Pascal](https://blog.skillfactory.ru/glossary/pascal/). Изначально он создавался для разработки приложений для ОС Windows, потом стал применяться и для других целей. Сейчас он не слишком популярен, а сфера его использования довольно узкая, но язык все еще нужен.

Синтаксис Delphi похож на Pascal, ведь этот язык — по сути, диалект «паскаля». Отличие Delphi в том, что он [объектно-ориентированный](https://blog.skillfactory.ru/glossary/oop-obektno-orientirovannoe-programmirovanie/), то есть сущности в нем представлены как объекты с методами и свойствами.

Более раннее название Delphi — Object Pascal. Также Delphi называется среда разработки для языка. Сейчас ее полное название — Embarcadero Delphi, хотя за время существования она сменила несколько имен.

Изначально язык создавался для разработки компьютерных приложений. Сейчас сфера его применения шире и одновременно уже. Им пользуются крупные проекты с многолетней историей для поддержки своего ПО, он задействован в ряде специфических сфер — например, системы контроля и учета доступа, электронные очереди, внутренние корпоративные программы. Также Delphi используется при работе с базами данных. Часто его используют для поддержки legacy-кода — такого, который был написан давно и считается морально устаревшим.

Delphi можно встретить в сфере [инди-геймдева](https://blog.skillfactory.ru/glossary/gamedev/). Например, на нем написан популярный движок GameMaker для создания простых игр.

Еще один вариант применения языка — обучение. Pascal, на котором он основан, создавался в том числе как учебный язык, и отсюда выросли его особенности: четкость и понятность кода, строгость синтаксиса. Это было нужно, чтобы не запутывать начинающих, а в итоге привело к тому, что язык стало легче расширять. Сейчас от обучения на Pascal и Delphi понемногу отходят, но эта практика до сих пор сохранилась, например в вузах.

**Объектно-ориентированный подход.** Delphi создан для разработки в парадигме [ООП](https://blog.skillfactory.ru/glossary/oop-obektno-orientirovannoe-programmirovanie/). В нем реализованы все основные принципы объектно-ориентированного программирования: наследование, полиморфизм, инкапсуляция, есть объекты и классы как особый тип данных. При этом он позволяет писать не только в объектно-ориентированном подходе — просто ориентирован на него.

**Четкость и структурированность.** Язык создан очень ясным и однозначным. У него строгий синтаксис, а значительная часть вещей задается явно и вручную. Изначально это было сделано для обучения, но позже позволило избежать сильного усложнения языка, когда он расширился. Синтаксис устроен так, чтобы не допускать неоднозначности. Многие вещи, которые теоретически позволены в других, менее «жестких» языках, в Delphi считаются ошибкой.

**Строгая типизация.** Типы в Delphi присваиваются строго, явно и статически. Это означает, что при создании переменной разработчик сразу указывает ее тип, и он не меняется в дальнейшем. При попытке присвоить этой переменной значение другого типа программа не [компилируется](https://blog.skillfactory.ru/glossary/kompilyator/) и выдает ошибку. Такой подход делает язык менее гибким, но защищает от неожиданных ошибок. То есть если разработчик присвоит переменной что-то не то, он поймет это еще на этапе компиляции, а не после отправки работающей с виду программы на деплой.

**Ручное управление памятью.** Большинство современных языков, на которых пишут десктопные или мобильные приложения, работают с памятью автоматически. Специальные процессы выделяют часть оперативной памяти для решения задач программы, а сборщики мусора позже очищают неиспользуемую память. Разработчику не нужно управлять этим процессом. Но в Delphi все не так.

В Delphi разработчик и выделяет, и очищает память вручную, отдельными командами. Автоматической работы с оперативной памятью по умолчанию не предусмотрено. Кажется, будто это делает написание кода сложнее, но на практике это позволяет сделать приложение более быстрым и менее требовательным к ресурсам компьютера. Правда, от разработчика требуется определенный уровень квалификации, иначе можно ошибиться и допустить утечку памяти.

**Консервативность.** Язык меняется медленно, и его основные принципы остаются неизменными. Программирование на Delphi сейчас строится примерно по тем же особенностям, что раньше. Консервативный подход поддерживается и строгой типизацией, и общей «жесткостью» синтаксиса. Сейчас, когда языки постоянно меняются, эта особенность — и плюс, и минус. Минус — потому что языку тяжело успевать за новыми тенденциями. Плюс — потому что программы на Delphi остаются предсказуемыми. Поэтому язык нередко используют там, где важно, чтобы система оставалась стабильной долгое время.

**2.3 Определение требований к техническим средствам**

Программа учета персональных компьютеров функционирует на компьютерах с определенным аппаратным и программным обеспечением.

Минимальные требования для корректной работы системы:

* процессор с тактовой частотой 1.5 ГГц;
* оперативная память – 1Гб
* 700 Мбайт свободного пространства на жестком диске;
* операционная система - Windows 7 и выше;
* офисный пакет Microsoft Office.

Рекомендуемая конфигурация системы:

* процессор с тактовой частотой 1.9 ГГц и выше;
* оперативная память – 2Гб и более;
* 700 Мбайт свободного пространства на жестком диске и более для большой базы данных;
* операционная система - Windows 7 или выше;
* офисный пакет Microsoft Office.

Для безопасной работы программы и сохранности полученной информации рекомендуется использование источника бесперебойного питания.

**2.4 Защита информации**

Защита информации — это процесс обеспечения конфиденциальности, целостности и доступности данных.

Шифрование данных. Это процесс преобразования данных в зашифрованный вид, который может быть прочитан только с помощью ключа. Шифрование может использоваться для защиты конфиденциальных данных, таких как пароли, номера кредитных карт и другие личные данные.

Установка паролей. Это может быть пароль на доступ к компьютеру, пароль на доступ к файлам или пароль на доступ к сети. Пароли должны быть достаточно сложными и надежными для защиты от взлома.

Защита информации в программе предусмотрена путем ограничения доступа к компьютеру на уровне операционной системы. Данные информационной системы не несут коммерческой тайны, поэтому явной авторизации при запуске программы не требуется.

**3 Разработка программного обеспечения**

**3.1 Определение формы представления входных и выходных данных**

После прохождения авторизации открывается главная форма программы, представленная на рисунке 1 приложения А. По умолчанию открывается первая вкладка со списком домов жилого сектора, с адресом и ФИО ответственного квартиросъемщика.

Выбором необходимой вкладки осуществляется выбор режима работы программы, например, для работы со списком сотрудников имеющих право совершать обходы и выносить предписания необходимо выбрать вкладку «Список сотрудников». Результат представлен на рисунке 2 приложения А.

Вкладка «Отчет об обходе ЖС» предназначена для просмотра результатов обхода и формирования отчета об обходах за период времени для контроля исправления выявленных недостатков или планирования следующих сроков обхода. Внешний вид вкладки представлен на рисунке 3 приложения А.

Вкладка «График обхода ЖС» служит для планирования обходов жилого сектора для последующей выдачи сотрудникам имеющим право обхода и вынесения предупреждений и замечаний. Вид вкладки представлен на рисунке 4 приложения А.

**3.2 Разработка справочной системы**

Справочная система играет ключевую роль в программном обеспечении, так как она помогает пользователям решать все возможные проблемы при работе с этими программами. С помощью справки пользователи могут получить полную информацию о том, как использовать программное обеспечение и принципах его работы. Использование справочной системы является важным фактором для успешной работы с программным обеспечением, так как она позволяет пользователям быстро и эффективно решать все возможные неполадки.

**3.3 Тестирование программного модуля**

Тестирование программного обеспечения является важным процессом, требующим глубокого понимания особенностей программного продукта и требований к нему. В отличие от тестирования сайта, тестирование программного обеспечения требует строгого и четкого следования процедурам и правилам. Тестирование программного обеспечения должно проводиться на специализированном оборудовании и с использованием специальных инструментов и программных средств. В результате любое тестирование программного обеспечения должно быть выполнено качественно, чтобы обеспечить стабильную работу приложения и минимизировать возможные ошибки и несоответствия требованиям.

Тестирование для информационной системы «Автоматизированная система мониторинга жилого сектора» проводилась в два этапа:

* тестирование устойчивости – проверка реакции программы на переход по внутренним вкладкам и элементам;

– тестирование функциональности – скорость реакции программы при различных действиях пользователя.

При тестировании программы вручную получены те же результаты, что и при работе программы на реальном устройстве с установленной на нём операционной системе Windows 7. Вычислительный процесс устойчив. Нарушений в работе не наблюдалось.

Программа корректно реагирует на все запросы пользователя. Переход по всем формам осуществляется незамедлительно.

По результатам тестирования можно сделать вывод, о том, что программа работает корректно и готова к эксплуатации.

Время выполнения одной операции в программе составляет не более одной секунды.

Тестирование программы проводилось по всем вариантам использования. Результаты тестирования представлены в таблице 1.

Таблица 1 – Журнал тестирования

|  |  |  |
| --- | --- | --- |
| Действие актера | Действие программного модуля | Отметка о правильной работе или описание ошибки |
| Запустить программу | Запуск | Действие выполнено успешно |
| Нажатие на кнопку логина | Нажатие на кнопку | Действие выполнено успешно |
| Перейти на главную форму | Переход | Действие выполнено успешно |
| Открыть список домов | Открытие | Действие выполнено успешно |
| Перейти на вкладку управления сотрудниками | Переход | Действие выполнено успешно |
| Переход по вкладкам | Переход | Действие выполнено успешно |
| Перейти на вкладку графика | Переход | Действие выполнено успешно |
| Перейти на вкладку отчета | Переход | Действие выполнено успешно |
| Управление сотрудниками | Управление | Действие выполнено успешно |

**3.4 Описание разрабатываемого программного продукта**

В современном мире программное обеспечение играет огромную роль в нашей жизни. Оно используется практически во всех сферах деятельности, начиная от обычного пользования компьютерами и заканчивая управлением крупными предприятиями. Разработка программного обеспечения – это сложный и трудоемкий процесс, требующий участия множества специалистов, начиная от системных аналитиков и заканчивая техническими писателями.

Каждый этап процесса разработки имеет свои особенности и нюансы. Например, системный анализ – это этап, на котором определяются роли каждого элемента в системе и их функции. Анализ требований – это этап, на котором уточняются особенности и характеристики программного продукта. Проектирование – это этап, где создаются представления о структуре приложения, его модулях и интерфейсах. Кодирование – это этап, на котором осуществляется написание кода на языке программирования. Тестирование – это последний этап перед запуском программного продукта, на котором проверяются его функции и работоспособность. Сопровождение – это этап, на котором вносятся изменения в уже существующий программный продукт, чтобы он соответствовал меняющимся требованиям рынка и заказчика.

Разработка программного обеспечения – это процесс, который требует работы над ошибками и совершенствования. Важная роль тут отводится тестированию, так как именно благодаря этому этапу можно увидеть проблемы и недостатки продукта, а также исправить их вовремя.

Последовательность и расчет трудоемкости разработки программного продукта оформляется в виде таблицы 2.

Таблица 2-Трудоемкость разработки программного продукта

|  |  |  |  |
| --- | --- | --- | --- |
| Виды работ | Кол-во  операций | Норма времени, ч | |
| на одну  операцию | на все  операции |
| 1 | 2 | 3 | 4 |
| 1. Подготовка исходных данных | 16 | 0,30 | 4,8 |
| 1. Реализация алгоритмов контрольных задач с использованием ПС ПЭВМ | 15 | 0,30 | 4,5 |
| 1. Обработка данных и получение результатов | 10 | 0,28 | 2,8 |
| 1. Анализ ошибок обработки данных и подготовка заключения о результатах проверки | 6 | 0,35 | 2,1 |
| 1. Ознакомление сотрудников службы сопровождения с содержанием задач, структурой входных и выходных данных | 2 | 0,79 | 1,58 |
| 1. Определение параметров настройки | 2 | 0,17 | 0,34 |
| 1. Ознакомление с объектом внедрения | 1 | 0,94 | 0,94 |
| 1. Консультации по подготовке пользователями исходных данных в соответствии с требованиями и ограничениями ОС ПЭВМ | 3 | 1,00 | 3 |
| 1. Разработка рекомендаций по реализации алгоритмов и требований пользователя к обработке данных с использованием ППП ПЭВМ по подготовке задач к опытной эксплуатации | 6 | 2,20 | 13,2 |
| 1. Оценка соответствия функциональных и эксплуатационных характеристик ПС требованиям к обработке данных | 4 | 1,10 | 4,4 |
| 1. Проведение консультаций и анализ ошибок комплексирования в период опытной эксплуатации | 4 | 1,70 | 6,8 |
| 1. Разработка рекомендаций по созданию программных средств сопряжения (программ и блоков пользователя, осуществляющих промежуточную обработку данных) | 1 | 1,70 | 1,7 |
| 1. Корректировка программ с целью изменения незначительных функциональных характеристик | 5 | 1,06 | 5,3 |
| 1. Разработка дополнительных модулей и включение их в состав ПС | 1 | 2,40 | 2,4 |
| 1. Анализ требований задач пользователя к обработке данных и характеристик среды их функционирования | 3 | 0,70 | 2,1 |
| 1. Разработка требований к тестированию и подготовка тестовых единиц | 7 | 0,70 | 4,9 |
| 1. Анализ результатов прогона и разработка функциональных спецификаций на корректировку ПС | 4 | 0,60 | 2,4 |
| 1. Внесение изменений в программы и эксплуатационную документацию ПС у пользователя | 7 | 0,29 | 2,03 |
| 1. Демонстрация функционирования на контрольных задачах службы сопровождения | 5 | 0,20 | 1 |
| 1. Анализ организационно- экономических и технических характеристик объекта внедрения ПС | 1 | 0,40 | 0,4 |
| 1. Разработка требований к выбору ПС для реализации задач пользователя | 1 | 0,40 | 0,4 |
| 1. Рекомендации по выбору ПС | 1 | 0,23 | 0,23 |
| 1. Оценка полноты охвата функциональными возможностями ПС | 6 | 0,40 | 2,4 |
| 1. Требования задач пользователей данного класса | 2 | 0,40 | 0,8 |
| 1. Выработка рекомендаций по расширению функциональных возможностей ПС | 1 | 0,10 | 0,1 |
| 1. Оценка необходимости проведения обучения работе по утвержденной технологии | 3 | 0,50 | 1,5 |
| 1. Анализ характеристик и производственных условий разработки, изготовления и сопровождения ПС у пользователя | 1 | 0,60 | 0,6 |
| 1. Разработка рекомендаций по применению у пользователя поставляемых технологических процессов разработки, производства и сопровождения ПС | 4 | 1,10 | 4,4 |
| 1. Практическая работа пользователей с ПС в вычислительном центре службы сопровождения | 5 | 1,40 | 7 |
| 1. Разработка рекомендаций по подготовке исходных данных в соответствии с требованиями и ограничениями ПС, по созданию технологии обработки данных с помощью ПС | 3 | 0,58 | 1,74 |
| 1. Консультации по устранению ошибок подготовки данных и решению задач | 4 | 0,3 | 1,2 |
| 1. Вывод на печать | 105 | 0,0028 | 0,29 |
| Итого трудоемкость  в т.ч ПЭВМ,  принтер |  |  | 87,89  87,60  0,29 |

Расход бумаги составит 105 листов, носителей информации – 1 диск DVD‑R.

Кроме программного модуля разработана сопровождающая программная документация в соответствии с ГОСТ 19.402-2000 «Описание программы», которая представлена в приложении Г и документация пользователя в соответствии с ГОСТ ИСО 9127-2002 «Документация пользователя», которая представлена в приложении Д.

**4 Экономическая часть**

**4.1 Расчет материальных затрат**

К материальным затратам относятся затраты на расходные материалы и затраты на электроэнергию на технологические цели.

Материальные затраты МЗ, руб., рассчитываются по формуле

МЗ = Ср.м + Сэн,

где Ср.м - стоимость расходных материалов, руб.;

Сэн - стоимость электроэнергии, руб.

МЗ = 38,14 + 11,18 = 49,32 руб.

Затраты на расходные материалы Ср.м, руб., определяются по формуле

Ср.м = Сб + Ск + Сн,

где Сб - стоимость бумаги, руб.;

Ск - стоимость картриджа для принтера, руб.;

Сн - стоимость носителя информации, руб.

Ср.м = 3,82 + 32,42 + 1,90 = 38,14 руб.

Затраты на бумагу определяются по формуле

Сб = Цб × Рб,

где Цб - цена за 1 лист бумаги, руб,;  
 Рб - расход листов бумаги при разработке и печати программного

продукта, шт;

Сб = ×105 = 3,82 руб.

Затраты на носители информации определяются по формуле

Сн = Цн × Рн,

где Цн - цена носителя информации, руб.;

Рн - расход дискет, CD, CD-RW, шт.

Сн =1,90×1 = 1,90 руб.

Затраты на электроэнергию определяют исходя из загруженности персонального компьютера программиста и частично занятости принтера за время разработки (учитывая, что при разработке программного продукта принтер используется меньше, чем персональный компьютер).

Затраты на электроэнергию определяются по формуле

Сэн = Цэн × (Тпк × Wпк + Тприн × Wприн),

где Цэн - тариф за 1 кВт-ч электроэнергии, руб.;

Тпк - время работы персонального компьютера, ч;

Тприн - время работы принтера, ч;

Wпк - потребляемая мощность ПК, кВт-ч;

Wприн - потребляемая мощность принтера, кВт-ч.

Сэн = 0,317×(87,89×0,40+0,29×0,35) = 11,18 руб.

**Расчет затрат на оплату труда и отчислений на социальные нужды**

Расчет затрат на оплату труда и отчислений на социальные нужды осуществляется при помощи прикладной программы на ПЭВМ. Пример данного расчета приведен в приложении Е.

**Расчет себестоимости разработки программного продукта**

Себестоимость разработки программного продукта Спол., руб., рассчитывается по формуле

Спол=МЗ+ФОТ+Осн+Ао+Зпр,

где Ао – амортизационные отчисления основных средств и

нематериальных активов,руб .;

Зпр – прочие затраты ,руб.

Спол = 49,32+ 611,25+207,83+5,28+1893,255=2766,94 руб.

Амортизационные отчисления рассчитываются по формуле

Ао = ,

К = ,

Фд = Фн×(1-%)

где ОС – стоимость основных средств и нематериальных активов , 1100,00 руб.;

Тн – нормативный срок службы ,10 лет ;

К – коэффициент, учитывающий долю занятости ПЭВМ;

Фд – действительный фонд времени работы ПЭВМ, ч;

Фн – утвержденный номинальный годовой фонд времени ,2016 ч;

%П – процент простоя оборудования в ремонте ,%;

Фд = 2016×(1- ) = 1834,56 ч.

К = = 0,048

АО = = 5,28 руб

Прочие затраты включают оплату услуг связи ,ВЦ, банков, сигнализаций , консультаций, аудиторских и рекламных услуг , за охрану ; вознаграждение ра-ботникам за изобретения и рационализаторские предложения; арендная плата за имущество; плата по процентам за краткосрочный и долгосрочный кредиты под пополнение оборотных средств; земельный налог; налог за пользование природ- ными ресурсами и другие налоги ,включаемые в себестоимость , рассчитываются по формуле

Зпр = ,

где %Зпр – процент прочих затрат ,%;

Зпр =  = 1893,26 руб

Результаты расчетов заносим в таблицу 3.

Таблица 3. Результаты расчетов

|  |  |
| --- | --- |
| Наименование  элементов затрат | Сумма, руб. |
| 1 Материальные затраты (за вычетом стоимости возвратных отходов) | 49,32 |
| 2 Затраты на оплату труда | 611,25 |
| 3 Отчисления на социальные нужды | 207,83 |
| 4 Амортизация основных средств и нематериальных активов, используемых в предпринимательской деятельности | 5,28 |
| 5 Прочие затраты | 1893,26 |
| Итого затрат на производство и реализацию продукции | 2766,94 |

**Расчет отпускной цены разработки программного продукта**

Отпускная цена разработки программного продукта без учета НДС Ц, руб., рассчитывается по формуле

Ц = Спол + Пр.

Ц = 2766,94 + 276,69 = 3043,63 руб.

Прибыль рассчитывается по формуле

Пр = ,

где НР – норматив рентабельности , 10%;

Пр = = 276,69 руб.

Цена отпускная с учетом НДС Цотп,руб.,рассчитывается по формуле

Цотп = Ц + НДС,

НДС = ,

где НДС – налог на добавленную стоимость ,руб.;

hндс – ставка налога на добавленную стоимость , %;

НДС =  = 608,73

Цотп = 3043,63+608,73 = 3652,40руб.

Результаты расчетов заносим в таблицу 3

Таблица 3 – Калькуляция отпускной цены разработки программного продукта

|  |  |
| --- | --- |
| Наименование статей калькуляции | Сумма, руб. |
| 1 | 2 |
| 1 Стоимость расходных материалов | 38,14 |
| 2 Стоимость электроэнергии | 11,18 |
| Итого материальные затраты | 49,32 |
| 3 Затраты на оплату труда | 611,25 |
| 4 Отчисления на социальные нужды | 207,83 |
| 5 Амортизация основных средств и нематериальных активов , используемых в предпринимательской деятельности | 5,28 |
| 6 Прочие затраты | 1893,25 |
| Итого полная себестоимость | 2766,94 |
| 7 Прибыль | 276,69 |
| Отпускная цена без учета НДС | 3043,63 |
| 8 Налог на добавленную стоимость | 608,73 |
| Итого отпускная цена с учетом НДС | 3652,40 |

**Расчет структуры себестоимости**

Под структурой себестоимости понимают процентное соотношение отдельных видов затрат, составляющих себестоимость к полной себестоимости.

Удельный вес отдельных калькуляционных статей Уд.в, %, в себестоимости определяется по формуле

Уд.вi= ×100,

где Зi – величина затрат i-той статьи себестоимости, руб.

Удельный вес материальных затрат рассчитывается по формуле

= ×100,

= ×100 = 1,78 %

Удельный вес затрат на оплату труда рассчитывается по формуле

= ×100,

= ×100 = 22,09 %

Удельный вес отчислений на социальные нужды в себестоимости рассчитываются по формуле

= ×100,

= ×100 = 7,51%

Удельный вес амортизации основных и нематериальных активов рассчитывается по формуле .

= ×100,

= ×100 = 0,19%

Удельный вес прочих затрат рассчитывается по формуле

= ×100,

= ×100 = 68,42%

Результаты заносим в таблицу 4

Таблица4 – Структура себестоимости

|  |  |  |
| --- | --- | --- |
| Показатели | Сумма, руб. | % |
| 1 Материальные затраты | 49,32 | 1,78 |
| 2 Затрат на оплату труда | 611,25 | 22,09 |
| 3 Отчисления в Фонд социальной защиты населения на социальные нужды | 207,83 | 7,51 |
| 4 Амортизация основных средств и нематериальных активов, используемых в предпринимательской деятельности | 5,28 | 0,19 |
| 5 Прочие затраты | 1893,26 | 68,42 |
| Итого полная себестоимость | 2766,94 | 100 |

Структура себестоимости оформлена в виде диаграммы и вынесена на лист 1 формата А3 графической части.

**Расчет технико-экономических показателей**

Эффективность разработки программного продукта подтверждается технико-экономическими показателями:

– трудоемкость разработки программного продукта;

– полная себестоимость;

– прибыль;

– отпускная цена;

– рентабельность продукта;

– материалоемкость;

– дельный вес ТЭР в себестоимости;

–затраты на 1 рубль реализованной продукции.

Рентабельность продукта R %, – показатель оценки эффективного использования текущих затрат на разработку программного продукта и рассчитывается по формуле

R = ×100

R =  ×100=10%

Материалоемкость Ме, руб/руб., – показывает долю материальных затрат в выручке продукции и рассчитывается по формуле

Ме =

Ме =  = 0,14руб./руб.

Удельный вес топливно-энергетических ресурсов в себестоимости УдТЭР %, – показывает долю топливно-энергетических затрат в себестоимости продукции и рассчитывается по формуле

= ×100

=×100 = 0,30%

Затраты на 1 рубль реализованной продукции Зреал,руб/руб., - это один из показателей эффективности производства и определяется по формуле

Зреал =

Зреал = = 0,76руб./руб.

Таблица 5 –Технико-экономические показатели

|  |  |  |
| --- | --- | --- |
| Наименование показателей | Единица измерения | Величина показателя |
| 1 Трудоемкость разработки программного продукта | ч. | 87,60 |
| 2 Полная себестоимость | руб. | 2766,94 |
| 3 Прибыль | руб. | 276,69 |
| 4 Рентабельность программного продукта | % | 10 |
| 5 Отпускная цена изделия с учетом НДС | руб. | 3652,40 |
| 6 Материалоемкость | руб/руб. | 0,14 |
| 7 Удельный вес топливно-энергетических ресурсов в себестоимости | % | 0,30 |
| 8 Затраты на 1 рубль реализованной продукции | руб/руб. | 0,76 |

Отпускная цена программного продукта с учетом НДС составит 3652,40 руб.

**5. Мероприятия по ТБ и промсанитарии, охрана окружающей**

**среды**

# Под техникой безопасности подразумевается комплекс мероприятий технического характера, направленных на создание безопасных условий труда и предотвращение несчастных случаев на производстве.

На любом предприятии принимаются меры к тому, чтобы труд работающих был безопасным, и для осуществления этих целей выделяются большие средства.

Министерство по чрезвычайным ситуациям в своей деятельности принимает широкий спектр мероприятий по технике безопасности (ТБ) и промышленной санитарии, а также охране окружающей среды. Эти мероприятия направлены на обеспечение безопасности сотрудников, предотвращение аварий и чрезвычайных ситуаций, а также минимизацию вреда для окружающей среды. Ниже перечислены некоторые основные мероприятия:

**Обучение и тренировки:**

Проведение обучающих курсов и тренировок по технике безопасности и промышленной санитарии для сотрудников МЧС.

Подготовка кадров к действиям в чрезвычайных ситуациях с учетом принципов техники безопасности.

**Разработка стандартов и инструкций:**

Разработка и внедрение стандартов и инструкций по безопасной эксплуатации оборудования и техники.

Установление процедур по управлению рисками и мерам по предотвращению аварий.

**Проверка и обслуживание оборудования:**

Регулярная проверка и обслуживание техники и оборудования для обеспечения их надежной работы и безопасности.

Применение технического контроля и профилактического обслуживания для предотвращения возможных аварийных ситуаций.

**Использование средств защиты:**

Предоставление сотрудникам средств индивидуальной защиты (СИЗ) и требование их использования при работе в опасных условиях.

Обеспечение обучения персонала по правильному использованию и хранению средств защиты.

**Мониторинг и анализ:**

Проведение регулярного мониторинга условий труда и экологической обстановки на объектах МЧС.

Анализ данных о происшествиях и авариях с целью выявления причин и разработки мер по их предотвращению в будущем.

**Охрана окружающей среды:**

Внедрение мер по сортировке и утилизации отходов на объектах МЧС.

Применение экологически чистых технологий и материалов при проведении операций и строительстве объектов.

Эти мероприятия помогают МЧС обеспечивать безопасные условия труда для своих сотрудников, предотвращать аварии и минимизировать вред для окружающей среды в своей деятельности.