## СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc160199098)

[1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc160199099)

[1.1 Постановка задачи 6](#_Toc160199100)

[1.2 Словарь терминов 6](#_Toc160199101)

[1.3 Описание предметной области 7](#_Toc160199102)

[2 ПРОЕКТИРОВАНИЕ ВЕБ-САЙТА «УЧЕТ ВНУТРИОФИСНЫХ РАСХОДОВ» 9](#_Toc160199103)

[2.1 Диаграмма прецедентов и расширенное описание прецедентов 9](#_Toc160199104)

[2.2 Диаграмма классов 10](#_Toc160199105)

[2.3 Диаграмма состояний 11](#_Toc160199106)

[3 РАЗРАБОТКА ПРОТОТИПА ВЕБ-САЙТА «УЧЕТ ВНУТРИОФИСНЫХ РАСХОДОВ» 14](#_Toc160199107)

[3.1 Общие принципы организации системы 14](#_Toc160199108)

[3.2 Организация доступа к данным 15](#_Toc160199109)

[3.3 Организация бизнес-логики и веб-интерфейса 16](#_Toc160199110)

[3.3.1 Органи*з*ация бизнес-логики 16](#_Toc160199111)

[3.4 Организация веб-интерфейса 17](#_Toc160199112)

[4 РЕАЛИЗАЦИЯ ЧАСТИ ФУНКЦИОНАЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ «УЧЕТ ВНУТРИОФИСНЫХ РАСХОДОВ» 19](#_Toc160199113)

[4.1 Соединение с базой данных 19](#_Toc160199114)

[4.2 Описание классов 19](#_Toc160199115)

[4.3 Создание страниц 20](#_Toc160199116)

[4.4 Реализация функционала и бизнес-логика 22](#_Toc160199117)

[ЗАКЛЮЧЕНИЕ 24](#_Toc160199118)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 25](#_Toc160199119)

[ПРИЛОЖЕНИЕ А ТЕХНИЧЕСКОЕ ЗАДАНИЕ 27](#_Toc160199120)

[ПРИЛОЖЕНИИ Б ЛИСТИНГ ПРОГРАММНОГО КОДА 33](#_Toc160199132)

# ВВЕДЕНИЕ

В современном мире эффективное управление финансами играет ключевую роль в обеспечении устойчивого развития организации. Одним из важных аспектов финансового управления является учет внутриофисных расходов. Внутриофисные расходы включают в себя все затраты, связанные с офисной деятельностью, такие как покупка офисного оборудования, расходные материалы, коммунальные услуги, услуги связи и другие операционные издержки.

Целью данного курсового проекта является изучение и анализ учета внутриофисных расходов для оптимизации затрат, контроля над финансами и планирования бюджета организации. Учет внутриофисных расходов позволяет не только фиксировать и классифицировать расходы, но и проводить анализ данных для принятия обоснованных управленческих решений.

В рамках данного курсового проекта будут рассмотрены основные компоненты учета внутриофисных расходов, преимущества данного процесса, а также технологические средства, которые могут быть использованы для эффективного управления расходами в офисной среде.

Изучение данной темы имеет практическое значение для компаний, поскольку позволяет улучшить финансовую дисциплину, эффективно управлять бюджетом и выявлять потенциальные области для сбережений. Таким образом, разработка системы учета внутриофисных расходов может стать важным инструментом для повышения эффективности финансового управления в организации.

Задачи курсового проекта:

* изучение предметной области учета внутриофисных расходов;
* анализ существующих подходов и методов учета внутриофисных расходов;
* определение требований к разрабатываемой системе;
* проектирование архитектуры и структуры будущей системы;
* создание базы данных для хранения информации;
* разработка модулей системы для учета, анализа расходов и формирования отчетов;
* разработка пользовательского интерфейса;
* внедрение системы на сервере и настройка окружения.

Курсового проекта включает в себя введение, две главы, заключение, список использованных источников и приложения. Во введении описывается проблема учета внутриофисных расходов и предлагается ее решение с помощью информационной системы. Кратко описывается предметная область и используемые термины.

В первой главе рассматриваются существующие подходы к учету расходов, их достоинства и недостатки. Анализируются возможности информационных систем для автоматизации этого процесса.

Вторая глава посвящена разработке информационной системы для учета внутриофисных расходов. Описываются основные бизнес-процессы, связанные с расходами, и предлагается модель их автоматизации. Разрабатывается база данных для хранения информации о расходах и интерфейс для работы с ней.

Заключение содержит выводы о проделанной работе и результаты внедрения разработанной системы.

# АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Учет внутриофисных расходов является важным аспектом финансового управления организацией. Это включает в себя запись, отслеживание и анализ всех расходов, связанных с офисной деятельностью, таких как расходы на офисное оборудование, расходные материалы, коммунальные услуги, услуги связи и другие операционные издержки.

## Постановка задачи

Необходимо разработать веб сайт, который позволит фирме по закупкам бытовой техники отслеживать и управлять внутриофисными расходами.

Фирма по закупки бытовой технике состоит из отделов. Каждый отдел имеет название. В каждом отделе работает определенное количество сотрудников. Сотрудники могут осуществлять покупки в соответствии с видами расходов.

Каждый вид расходов имеет название, некоторое описание и предельную сумму средств, которые могут быть потрачены в месяц. При каждой покупке сотрудник оформляет документ, где указывает вид расхода, дату, сумму и отдел. Техническое задание представлено в ПРИЛОЖЕНИИ А.

## Словарь терминов

Администратор – пользователь информационной системы, который обладает наибольшим набором прав в системе, и ведущий основную работу с ней.

Фирма – это организация, которая занимается коммерческой деятельностью, то есть продажей товаров или оказанием услуг с целью получения прибыли.

Расходы - это затраты, которые несет компания на свою деятельность. Они могут быть связаны с производством товаров или услуг, с арендой помещений, с оплатой труда сотрудников и т.д.

Заявка — это документ, оформляемый сотрудником при осуществлении покупки в рамках определенного вида расходов

Бухгалтерский учёт — это система, которая собирает, регистрирует и анализирует финансовую информацию о компании. Эта информация используется для принятия решений, контроля над деятельностью компании и управления ею.

Внутрикорпоративные расходы — это затраты, которые несет компания на свою деятельность, включая затраты на персонал, аренду, коммунальные услуги, рекламу и другие. Эти расходы учитываются при принятии решений о распределении ресурсов и контроле над деятельностью компании.

## Описание предметной области

Внутриофисные расходы – это расходы, связанные с деятельностью компании, но не связанные непосредственно с производством или продажей товаров или услуг. Это могут быть расходы на аренду помещений, коммунальные услуги, зарплату сотрудникам, рекламу и другие расходы.

Целью учета внутриофисных расходов является сбор информации о этих расходах, анализ их и планирование, контроль за их соблюдение. Для этого используются методы бухгалтерского учета, управленческого учета, экономического анализа.

Один из методов учета – метод учета затрат по функциям. Он основан на расчете затрат на выполнение функций или задач в компании. Этот метод помогает определить, какие функции являются наиболее затратными, и как можно сократить затраты на них.

Контроль за соблюдением расходов проводится с помощью бюджетирования, контроля затрат, анализа отклонений от бюджета.

Таким образом, учет внутриофисных расходов – это сложный процесс. Он требует знаний и опыта в бухгалтерском учете, экономическом анализе и других областях.

Взаимодействие с веб-сайтом должно осуществляться с помощью веб интерфейса, который можно будет запустить с любого устройства, которое имеет доступ в глобальную сеть Интернет, при этом доступ к системе можно будет получить практически из любой точки мира. В системе будут участвовать два вида ролей пользователей: администратор и оператор. Каждая из них будет иметь свои права и функциональные возможности. Так оператору будет дана возможность:

* Изменение информации о пользователях;
* управление доступом пользователей к различным функциям сайта;
* просмотр статистики сайта;
* связь с технической поддержкой сайта;
* оплата услуг сайта с помощью различных платежных систем.

Администраторы имеют следующие возможности:

* Управление пользователями: администраторы могут добавлять новых пользователей, изменять информацию о существующих пользователях и удалять неактивных пользователей;
* управление доступом: администраторы могут предоставлять или ограничивать доступ к различным разделам сайта для разных групп пользователей;
* управление контентом: администраторы могут создавать и редактировать статьи, новости, товары и другие элементы контента на сайте;
* управление рекламой: администраторы могут размещать рекламные баннеры на сайте и управлять ими;
* управление платежами: администраторы могут подключать и настраивать платежные системы для приема платежей от пользователей.

Вывод по главе

В данной главе был проведен анализ предметной области и определена необходимость создания веб-приложения для учёта внутриофисных расходов в сфере ремонта компьютеров. Были рассмотрены основные термины и понятия, используемые в данной отрасли, а также определены роли администратора и пользователей в системе.

# ПРОЕКТИРОВАНИЕ ВЕБ-САЙТА «УЧЕТ ВНУТРИОФИСНЫХ РАСХОДОВ»

В данной главе будет описано проектирование веб-сайта «Учет внутриофисных расходов», включая описание его структуры, функций, интерфейса и других аспектов. Будет спроектирована база данных для хранения информации о расходах и создадим необходимые таблицы и связи между ними.

## 2.1 Диаграмма прецедентов и расширенное описание прецедентов

Диаграмма прецедентов - это диаграмма, которая показывает отношения между актерами и прецедентами в системе. Она помогает определить, какие функции системы важны для пользователей, и как эти функции связаны друг с другом.

На рисунке 1 изображена диаграмма прецедентов для прототипа веб-сайта «Учет внутриофисных расходов». На данной диаграмме изображены все варианты использования разрабатываемого веб сайта.

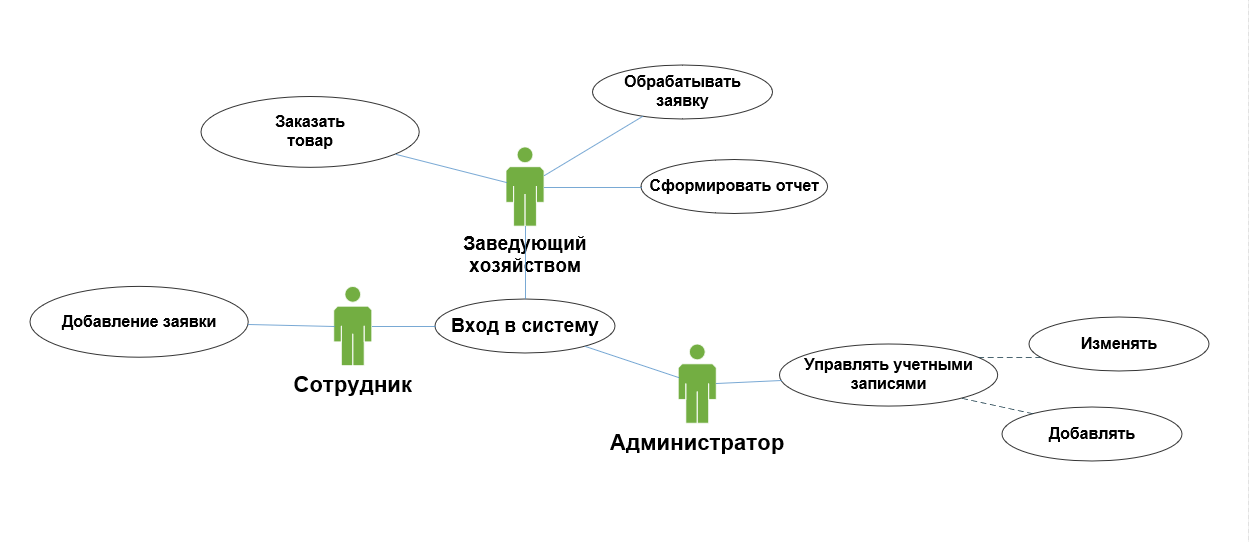


Рисунок 1 – Диаграмма прецедентов

Опишем диаграмму прецедентов более подробно в таблице 1

Таблица 1 – Описание диаграммы прецедентов

| № | Роли | Действие |
| --- | --- | --- |
| 1 | Заведующий хозяйством | Заказать товар,  Обработать заявку,  Сформировать отчет. |
| 2 | Сотрудник | Добавление заявки |
| 3 | Администратор | Управление учетными записями   * Изменять, * добавлять. |

## 2.2 Диаграмма классов

Диаграмма классов - это диаграмма, которая отображает структуру системы в виде классов и отношений между ними. Она помогает понять, какие объекты существуют в системе, какие свойства и методы у них есть, а также как они связаны друг с другом.

В процессе анализа диаграммы прецедентов были выделены 4 сущности, на основании которых была построена диаграмма классов, изображённая на рисунке 2.

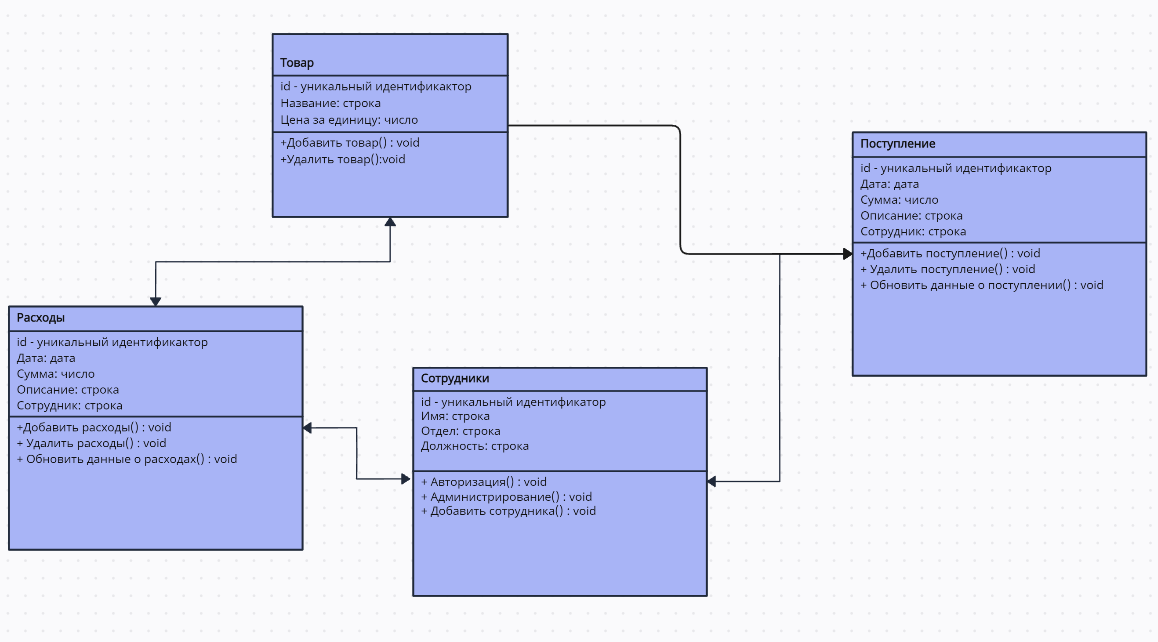


Рисунок 2 – Диаграмма классов

Опишем более подробно основные сущности в таблице 2.

Таблица 2 – Описание сущностей

| Название | Описание |
| --- | --- |
| Название сущности | Товар |
| Словесное описание | Наличие товаров |
| Атрибуты | Id: integer  Название: string  Цена за единицу: integer |
| Название сущности | Сотрудники |
| Словесное описание | Сотрудники отдела |
| Атрибуты | Id: integer  Имя: string  Отдел: string  Должность: string |
| Название сущности | Расходы |
| Словесное описание | Отслеживание расходов фирмы |
| Атрибуты | Id: integer  Дата:date  Сумма: integer  Описание: string  Сотрудник: string |
| Название сущности | Поступление |
| Словесное описание | Отслеживание поступлений фирмы |
| Атрибуты | Id: integer  Дата:date  Сумма: integer  Описание: string  Сотрудник: string |

## 2.3 Диаграмма состояний

Диаграмма состояний - это диаграмма, которая описывает поведение объекта в системе с точки зрения его возможных состояний и переходов между ними. Она помогает понять, как объект может изменяться в процессе выполнения, какие события могут вызывать эти изменения и какие действия объект может выполнять в каждом состоянии.

На рисунке 3 изображена диаграмма состояний для прецедента, она показывает состояния, которые он может иметь.

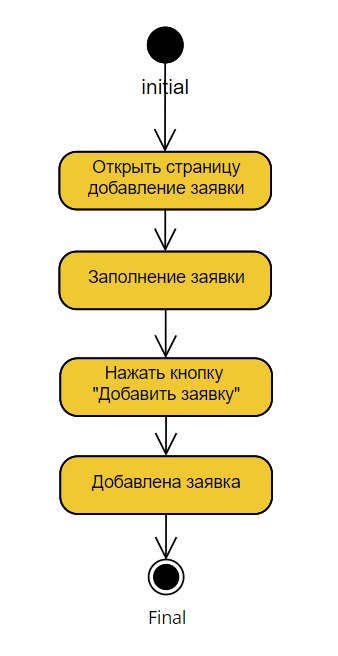


Рисунок 3 – Диаграмма состояний

Процесс добавление заявки начинается с того, что сотрудник открывает страницу добавление заявки. Затем заполняет ее по следующим критериям:

* Название товара;
* категория товара;
* сумма расходов;
* дата покупки;
* загрузить чек (необязательно);
* комментарии.

Затем сотрудник добавляет заявку на покупку товара, с последующей её обработкой от зав.хоза

В данной главе были определены основные требования и спецификации для разрабатываемого веб-приложения. Создана диаграмма прецедентов для описания основных функций и сценариев использования системы, а также определена структура объектов и их взаимосвязи в системе с помощью диаграммы классов. Разработана база данных для хранения информации о внутриофисных расходах.

Для определения возможных сценариев работы с системой и учета всех возможных состояний объектов разработана диаграмма состояний. В результате, были получены основные элементы проекта, необходимые для разработки веб-приложения, включая функциональные требования, спецификации, диаграммы прецедентов, классов и состояний. Эти элементы будут использоваться в последующих главах для разработки и тестирования веб-приложения. Кроме того, были выделены основные этапы разработки веб-приложения и описаны роли участников проекта. Разработан план проекта, который включает задачи по разработки, и поддержке веб-приложения

# 3 РАЗРАБОТКА ПРОТОТИПА ВЕБ-САЙТА «УЧЕТ ВНУТРИОФИСНЫХ РАСХОДОВ»

Разработка прототипа информационной системы (ИС) – это процесс создания предварительной версии системы, которая демонстрирует основные функциональные возможности и интерфейс системы. Прототип разрабатывается в начальной стадии проекта, чтобы позволить заказчику или пользователям оценить и проверить концепцию и функциональность системы.

## 3.1 Общие принципы организации системы

Веб-сайт - это особый тип программы, работающей по архитектуре «клиент-сервер». Главное приложение сайта находится на сервере, а пользователь (клиент) получает только результаты его работы. Сайт получает запросы от пользователя, обрабатывает их и выдает результат. Все это происходит через интернет.

Отображением результатов запросов, а также приемом данных от клиента и их передачей на сервер обычно занимается специальное приложение — браузер (Internet Explorer, Mozilla, Opera и т. д.). Как известно, одной из функций браузера является отображение данных, полученных из Интернета, в виде страницы, описанной на языке HTML, следовательно, результат, передаваемый сервером клиенту, должен быть представлен на этом языке.

На стороне сервера Web-приложение выполняется специальным программным обеспечением (Web-сервером), который и принимает запросы клиентов, обрабатывает их, формирует ответ в виде страницы, описанной на языке HTML, и передает его клиенту. Одним из таких Web-серверов является Internet Information Services (IIS) компании Microsoft.

## 3.2 Организация доступа к данным

Данные веб-сайта хранятся в базе данных реляционного типа. Таким образом, прежде выделенные сущности были преобразованы в таблицы. На рисунке 4 представлена схема базы данных

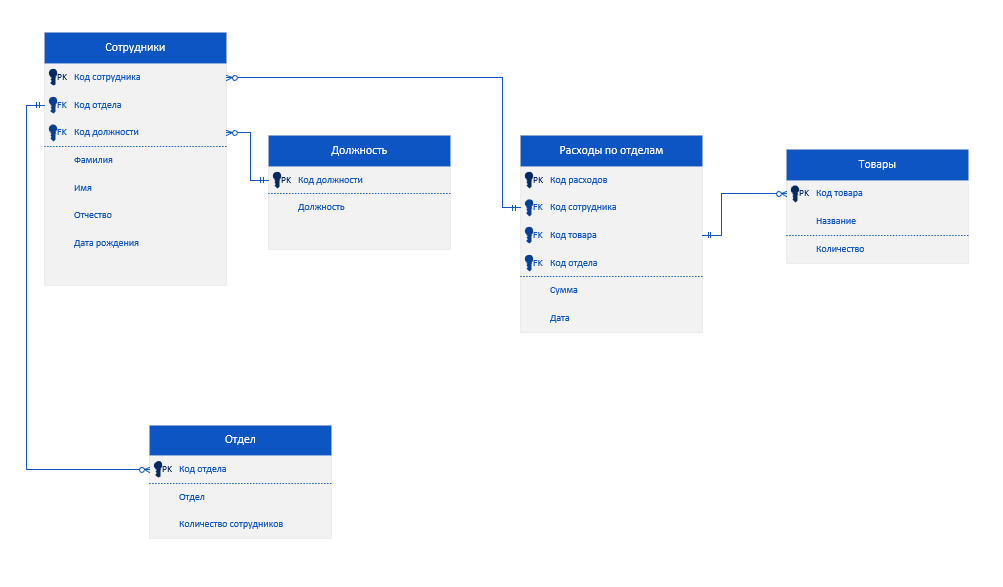


Рисунок 4 – Логическая модель данных

Ориентируясь на созданную логическую модель данных информационной системы, можно вынести таблицы:

* Сотрудники;
* отдел;
* должность;
* расходы по отделам;
* товар.

Описание сущностей(таблиц):

* К объекту «Сотрудники» относятся: ФИО, Дата рождения;
* к объекту «Отдел» относятся: Отдел, Количество сотрудников;
* к объекту «Должность» относятся: Должность;
* к объекту «Расходы по отделам» относятся: Сумма, Дата;
* к объекту «Товар» относятся: Название, Количество.

Описание связей:

Сотрудник имеет возможность осуществлять мелкие покупки для нужд фирмы, поэтому между объектами «Сотрудник» и «Расходы по отделам» имеется связь «Один ко многим». У каждого сотрудника есть своя должность, поэтому между объектами «Сотрудник» и «Должность» имеется связь «Многие ко многим». В каждом отделе работает определенное количество сотрудников, поэтому между объектами «Отдел» и «Сотрудники» имеется связь «Один ко многим». По каждому расходу имеется свой вид расхода, либо же товар, поэтому между объектами «Расходы по отделам» и «Товар», имеется связь «Один ко многим».

## 3.3 Организация бизнес-логики и веб-интерфейса

Для успешной реализации проекта по разработке веб-сайта для отслеживания внутриофисных расходов необходимо продумать организацию бизнес-логики и создание удобного веб-интерфейса для пользователей. Важно, чтобы функционал сайта был интуитивно понятен и удобен в использовании для всех пользователей.

### 3.3.1 Органи*з*ация бизнес-логики

Регистрация отделов и управление сотрудниками:

* разработка функционала для добавления новых отделов с указанием их названия и описания;
* возможность добавления, удаления и управления сотрудниками в каждом отделе.

Классификация видов расходов:

* создание категорий расходов с возможностью добавления новых категорий и установки предельной суммы расходов;
* возможность просмотра статистики расходов по каждой категории.

Оформление покупок и отслеживание расходов:

* функционал для оформления новых товаров с указанием даты, суммы, категории расхода и отдела;
* возможность просмотра и анализа всех совершенных покупок с фильтрацией по различным параметрам.

## 3.4 Организация веб-интерфейса

В веб-приложении используется технология ASP.NET для декларативного модульного описания шаблона страниц. ASP.NET позволяет избавиться от недостатков, присущих скриптовым языкам. В ASP.NET HTML-код страницы и программный код, выполняемый на сервере (серверный код), выделены в разные файлы \*.aspx и \*.aspx.cs соответственно. Содержимое ASP.NET-страниц помимо тэгов HTML включает в себя так называемые серверные элементы управления – специальные тэги, которые имеют объектное представление в серверном коде.

Карта сайта представлена на рисунке 5.

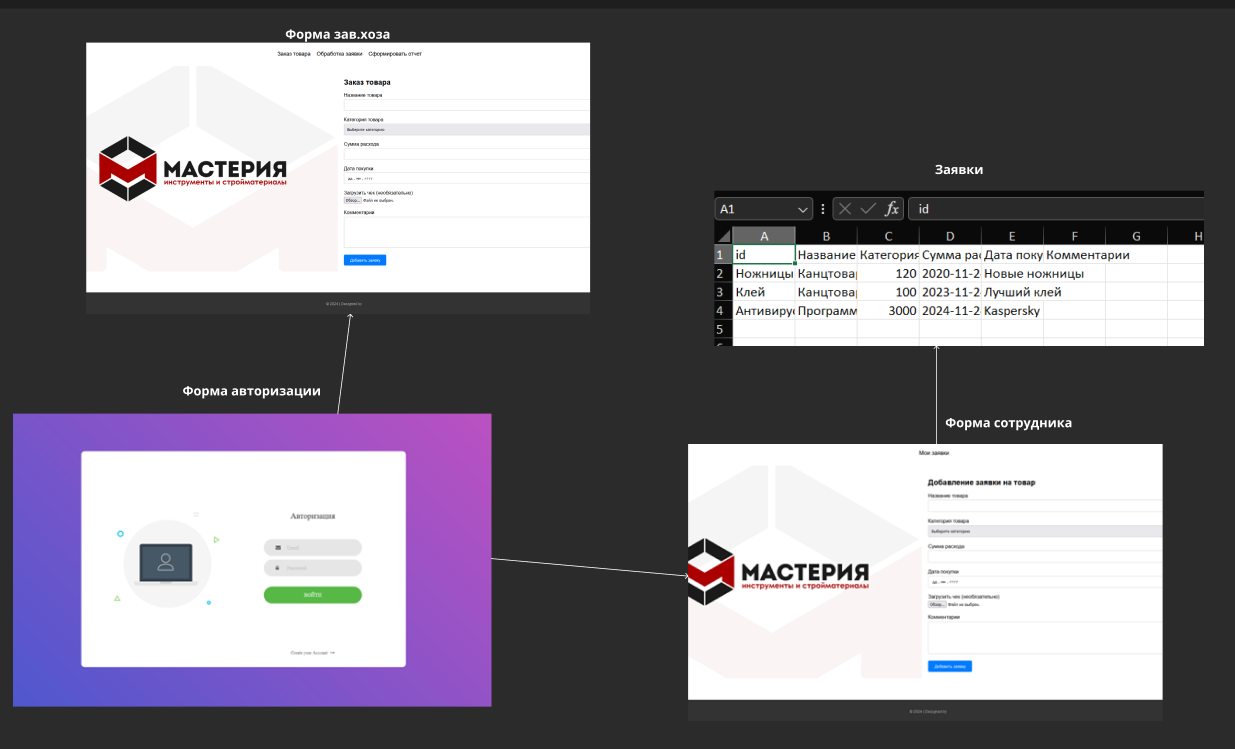


Рисунок 5 – Карта сайта

Вывод по главе

В данной главе был разработан прототип веб-сайта «Учет внутриофисных расходов».

В рамках разработки прототипа были определены общие принципы организации системы, такие как использование базы данных для хранения информации и применение объектно-ориентированного подхода к разработке. Также была организована работа с данными, определена структура базы данных и основные сущности, участвующие в бизнес-процессах. Особое внимание было уделено организации бизнес-логики системы и разработке алгоритмов обработки данных. Были определены основные функции системы, сценарии использования и требования к функциональности. Наконец, была разработана структура веб-интерфейса, определены основные компоненты и элементы управления, обеспечивающие удобство и эффективность работы пользователей с системой.

# 4 РЕАЛИЗАЦИЯ ЧАСТИ ФУНКЦИОНАЛА ИНФОРМАЦИОННОЙ СИСТЕМЫ «УЧЕТ ВНУТРИОФИСНЫХ РАСХОДОВ»

В данной главе будет подробно описан процесс реализации части функционала информационной системы, предназначенной для учета и контроля внутриофисных расходов в организации. Целью данной части функционала является автоматизация процесса учета расходов, оптимизация бюджетного контроля и обеспечение эффективного управления финансами внутри компании.

## 4.1 Соединение с базой данных

Файл web.config хранит в себе информацию о подключении к базе данных, далее следует листинг фрагмента этого файла, который отвечает за связь с БД.

const mysql = require('mysql2');

// create the connection to database

const connection = mysql.createConnection({

host: 'localhost',

user: 'root',

pass: 123,

database: 'test'

});

// simple query

connection.query(

'SELECT \* FROM `table',

function(err, results, fields) {

console.log(results); // results contains rows returned by server

console.log(fields); // fields contains extra meta data about results, if available

}

);

## 4.2 Описание классов

В среде Visual Studio Code в проекте APS.NET, файлы классов описаний объектов хранится в папке Models. В данном проекте здесь находятся следующие модули: Expense, Employee, Departament

Модуль Expense:

class Expense:

def \_\_init\_\_(self, category, amount, date):

self.category = category

self.amount = amount

self.date = date

def \_\_str\_\_(self):

return f"{self.date}: {self.category} - {self.amount}"

Модуль Employee:

class Employee:

def \_\_init\_\_(self, name, position):

self.name = name

self.position = position

self.expenses = []

def add\_expense(self, category, amount, date):

expense = Expense(category, amount, date)

self.expenses.append(expense)

def total\_expenses(self):

return sum(expense.amount for expense in self.expenses)

Модуль Departament:

class Department:

def \_\_init\_\_(self, name):

self.name = name

self.employees = []

def add\_employee(self, employee):

self.employees.append(employee)

def total\_department\_expenses(self):

return sum(employee.total\_expenses() for employee in self.employees)

## 4.3 Создание страниц

Для создания веб-страниц использовался редактор Visual Studio Code. Виртуальное отображение этих страниц можно увидеть на рисунках.

Интерфейс данных страниц представлен на рисунках 6 – 8.

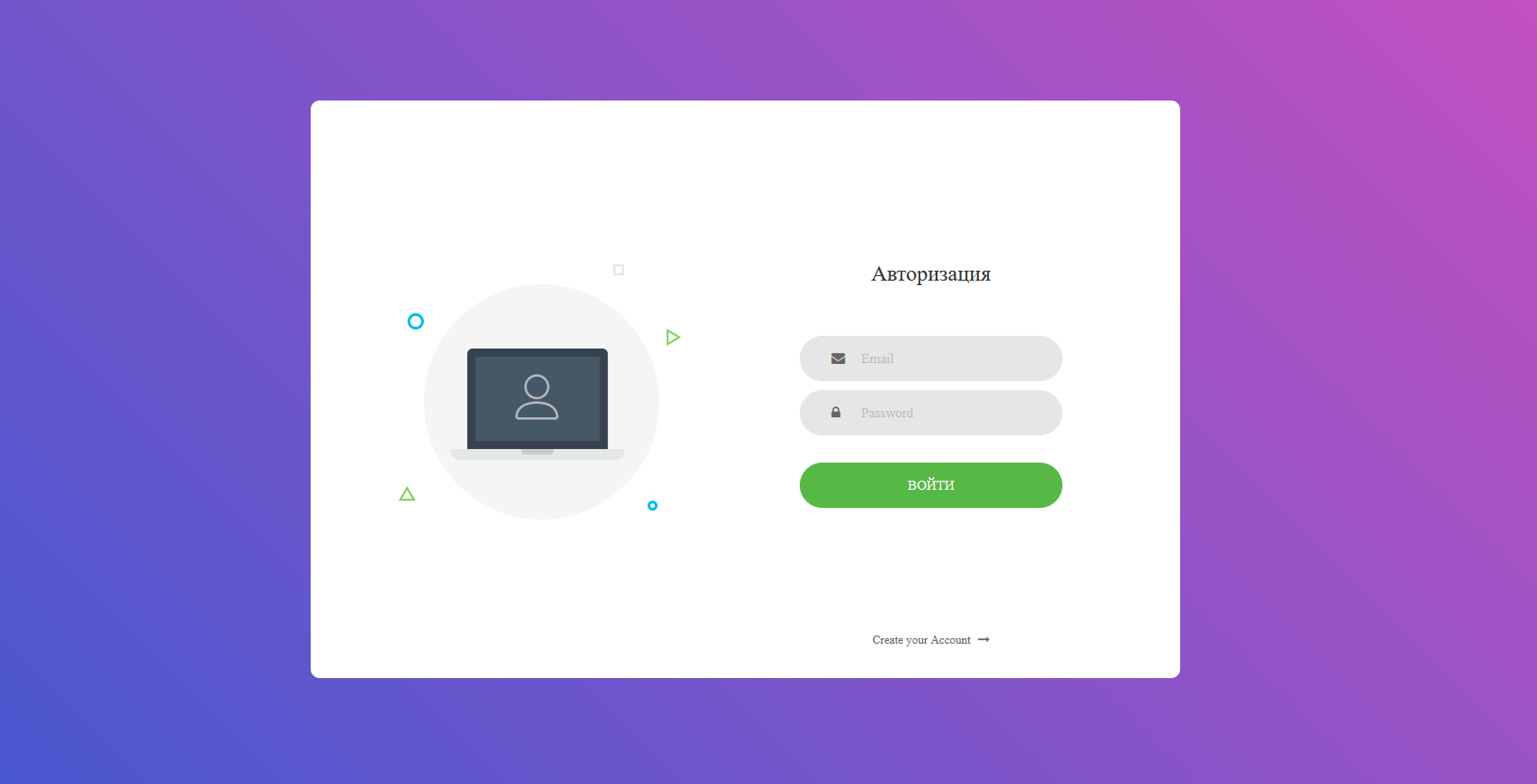


Рисунок 6 – Форма Авторизации

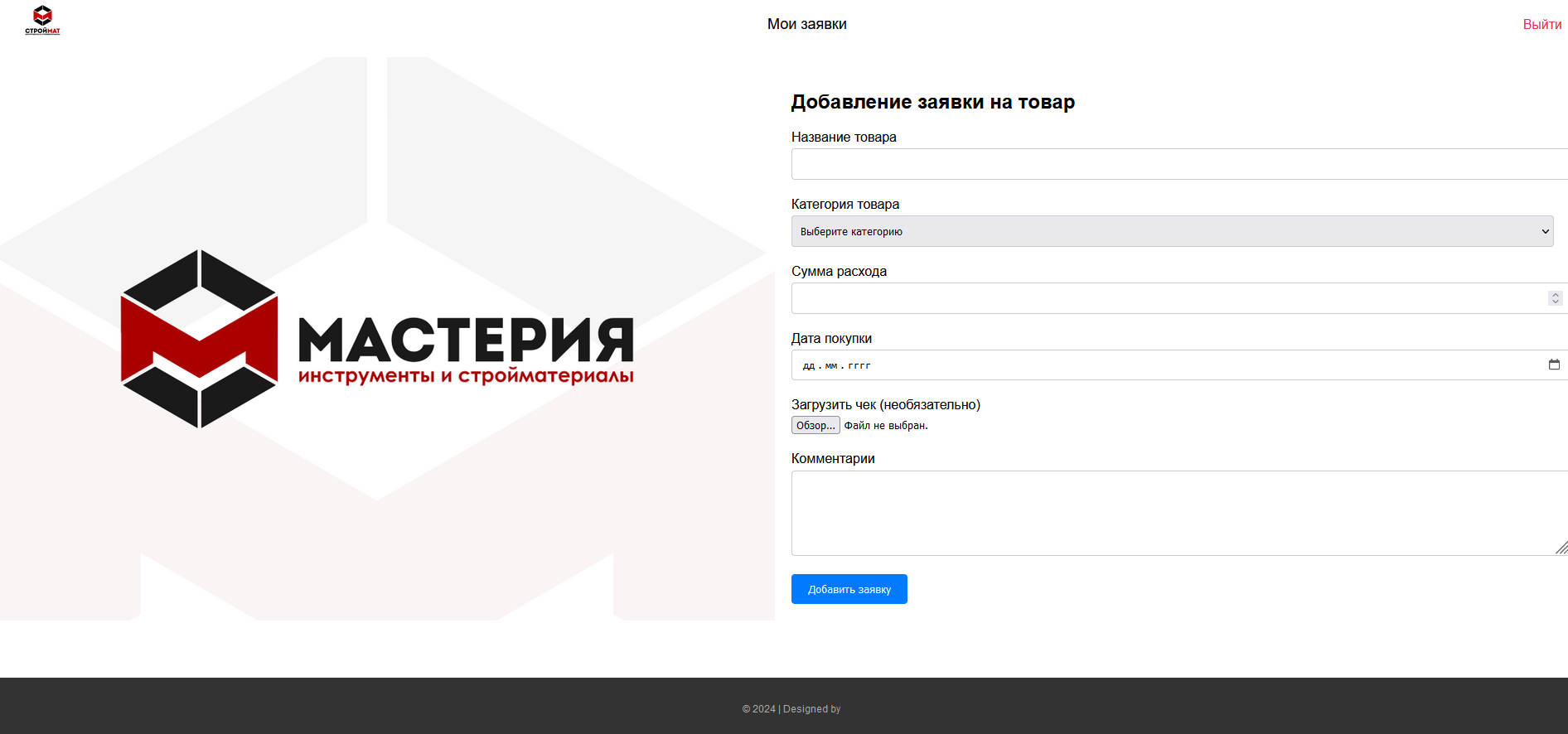


Рисунок 7 – Форма сотрудника

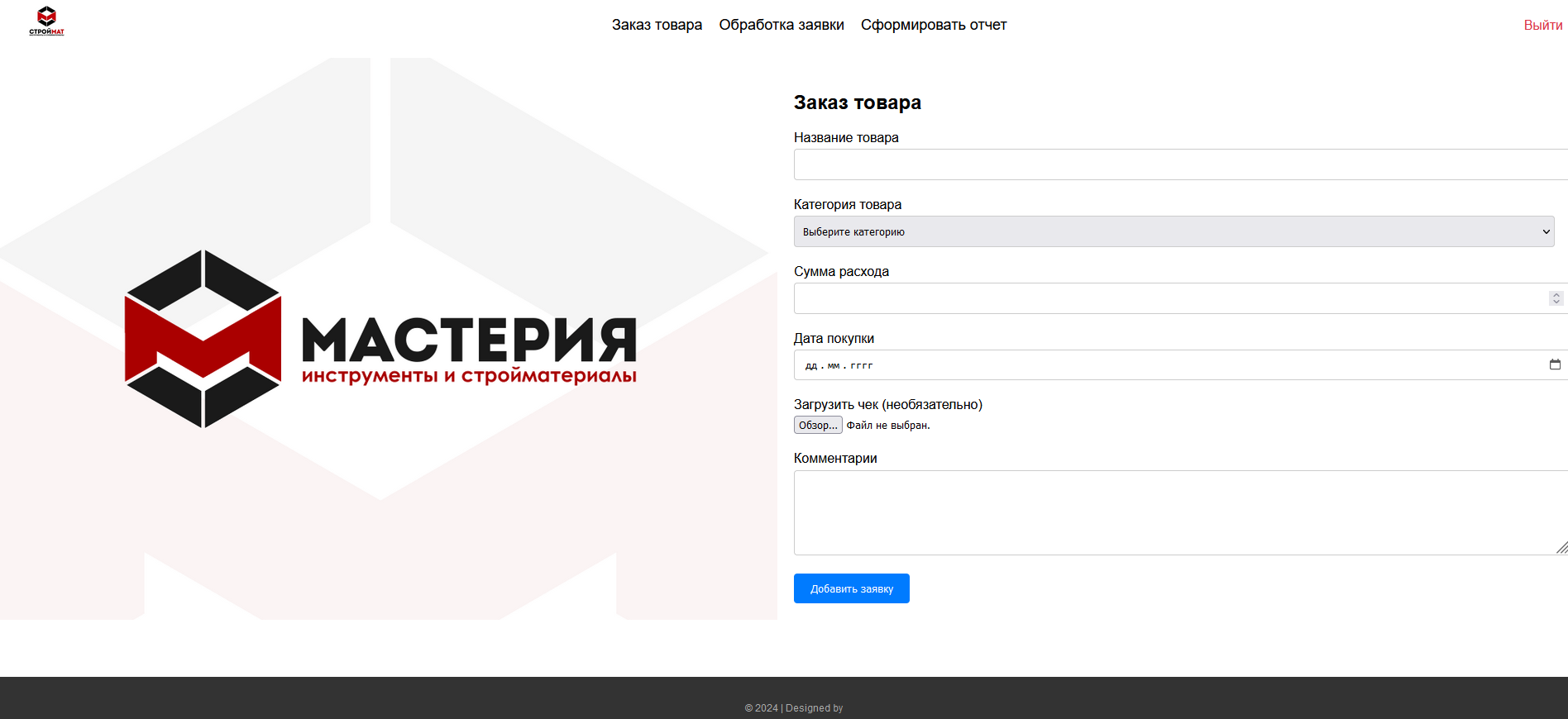


Рисунок 8 – Форма Зав.хоза

Листинг программного кода представлен в ПРИЛОЖЕНИИ Б.

## 4.4 Реализация функционала и бизнес-логика

В проекте APS.NET бизнес-логика хранится в папке Controllers.

Листинг программного кода безнес-логики веб-сайта

from flask import Flask, render\_template, send\_file, request, redirect

import sqlite3

import openpyxl

from io import BytesIO

app = Flask(\_\_name\_\_)

@app.route('/')

def index():

    return render\_template('content.html')

@app.route('/download\_excel')

def download\_excel():

    # Соединяемся с базой данных SQLite

    conn = sqlite3.connect('requests.db')

    cursor = conn.cursor()

    # Получаем данные из базы данных

    cursor.execute("SELECT id, item\_name, item\_category, item\_cost, purchase\_date, comments FROM requests")

    data = cursor.fetchall()

    # Создаем Excel-файл и заполняем его данными

    wb = openpyxl.Workbook()

    ws = wb.active

    ws.append(['id','Название товара', 'Категория товара', 'Сумма расхода', 'Дата покупки', 'Комментарии'])

    for row in data:

        ws.append(row)

    # Создаем временный буфер для хранения файла Excel

    excel\_buffer = BytesIO()

    wb.save(excel\_buffer)

    excel\_buffer.seek(0)

    # Отправляем файл Excel как ответ на запрос

    return send\_file(

        excel\_buffer,

        mimetype='application/vnd.openxmlformats-officedocument.spreadsheetml.sheet',

        as\_attachment=True,

        download\_name='Заявки.xlsx'

    )

@app.route('/add\_request', methods=['POST'])

def add\_request():

item\_name = request.form['item\_name']

item\_category = request.form['item\_category']

item\_cost = request.form['item\_cost']

purchase\_date = request.form['purchase\_date']

comments = request.form['comments']

# Вставляем данные в базу данных

conn = sqlite3.connect('requests.db')

cursor = conn.cursor()

cursor.execute("INSERT INTO requests (item\_name, item\_category, item\_cost, purchase\_date, comments) VALUES (?, ?, ?, ?, ?)",

(item\_name, item\_category, item\_cost, purchase\_date, comments))

conn.commit()

conn.close()

# Перенаправляем пользователя обратно на страницу, где он добавлял заявку

return redirect('/')

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True, port=5000)

Вывод по главе

В данной главе была реализована часть функционала информационной системы «Учет внутриофисных расходов».

В рамках реализации были выполнены следующие задачи:

Соединение с базой данных: было реализовано соединение с базой данных, что позволило системе сохранять и извлекать данные.  
Описание классов: были определены и описаны классы, которые представляют собой основные структурные единицы системы.

Создание страниц: были созданы необходимые веб-страницы для отображения информации и выполнения основных функций системы.

Реализация функционала и бизнес-логика: был реализован основной функционал системы, включая обработку данных, генерацию отчетов и выполнение расчетов

# ЗАКЛЮЧЕНИЕ

В рамках данного курсового проекта были рассмотрены основные компоненты учета внутриофисных расходов, преимущества данного процесса, а также технологические средства, которые использованы для эффективного управления расходами в офисной среде.

Целью данного курсового проекта являлась изучение и анализ учета внутриофисных расходов для оптимизации затрат, контроля над финансами и планирования бюджета организации. Учет внутриофисных расходов позволяет не только фиксировать и классифицировать расходы, но и проводить анализ данных для принятия обоснованных управленческих решений.

Были выполнены следующие задачи:

* Изучение предметной области учета внутриофисных расходов;
* анализ существующих подходов и методов учета внутриофисных расходов;
* определение требований к разрабатываемой системе;
* проектирование архитектуры и структуры будущей системы;
* создание базы данных для хранения информации;
* разработка модулей системы для учета, анализа расходов и формирования отчетов;
* разработка пользовательского интерфейса;
* внедрение системы на сервере и настройка окружения.

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Figma [Электронный ресурс]: Режим доступа: https://www.figma.com/files/recent?fuid=1042359483770556320 Дата обращения - 15.12.2023
2. Canva [Электронный ресурс]: Режим доступа: https://www.canva.com/design/DAFjz\_fpFss/Uy4oDR4Ao52CwmvemoUGVQ/edit Дата обращения - 15.12.2023
3. Ангарский Промышленно Экономический Техникум [Электронный ресурс]: Режим доступа: https://a-pet.ru/moodle/user/files.php Дата обращения - 15.12.2023
4. Checko [Электронный ресурс]: Режим доступа: https://checko.ru/company/alfa-pro-1163850088544 Дата обращения - 15.12.2023
5. Seonews [Электронный ресурс]: Режим доступа: https://www.seonews.ru/analytics/sayt-vizitka-prakticheskoe-rukovodstvo-i-primery/ Дата обращения - 15.12.2023
6. Microsoft Visual Studio [Электронный ресурс] – Wikipedia, 2022. Режим доступа: https://ru.wikipedia.org/wiki/Microsoft\_Visual\_Studio (дата обращения: 22.02.2024).
7. W3Schools. HTML Tutorial [Электронный ресурс]. Режим доступа: https://www.w3schools.com/html/ (дата обращения: 22.02.2024).
8. MDN Web Docs. CSS [Электронный ресурс]. Режим доступа: https://developer.mozilla.org/en-US/docs/Web/CSS (дата обращения: 22.02.2024).
9. JavaScript MDN Web Docs. JavaScript [Электронный ресурс]. Режим доступа: https://developer.mozilla.org/en-US/docs/Web/JavaScript (дата обращения: 22.02.2024).
10. SQLCourse. SQLCourse.com: an interactive online training course for SQL beginners [Электронный ресурс]. Режим доступа: <https://www.sqlcourse.com/> (дата обращения: 22.02.2024).
11. MySQL Tutorial. MySQL Tutorial [Электронный ресурс]. Режим доступа: <https://www.mysqltutorial.org/> (дата обращения: 22.02.2024).
12. PHP: Hypertext Preprocessor. PHP: Hypertext Preprocessor [Электронный ресурс]. Режим доступа: <https://www.php.net/> (дата обращения: 22.02.2024).
13. MongoDB. MongoDB Documentation [Электронный ресурс]. Режим доступа: https://docs.mongodb.com/ (дата обращения: 22.02.2024).
14. React. React – A JavaScript library for building user interfaces [Электронный ресурс]. Режим доступа: <https://reactjs.org/> (дата обращения: 22.02.2024).
15. Angular. Angular [Электронный ресурс]. Режим доступа: <https://angular.io/> (дата обращения: 22.02.2024).

# ПРИЛОЖЕНИЕ А

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Министерство образования Иркутской области

Государственное бюджетное профессиональное образовательное учреждение

Иркутской области

**«Ангарский промышленно-экономический техникум»**

(ГБПОУ ИО «АПЭТ»)

|  |
| --- |
| Автоматизация системы инвестирование свободных средств |
| Наименование ИС |
| УЧЕТ ВНУТРИОФИСИНЫХ РАСХОДОВ |
| Сокращенное наименование ИС |
| **ТЕХНИЧЕСКОЕ ЗАДАНИЕ** |
| Действует с 27.02.2024 |

|  |
| --- |
| Составитель |
| Березин Данил Александрович |
| ФИО сотавителя |
| Разработчик |
| Должность составителя |

г. Ангарск, 2024 г.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ. 3](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338856)

[1. Основание для разработки……………………………………………………...3](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338857)

[2. Назначение разработки…………………………………………………………3](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338858)

[3. Технические требования к программе или программному изделию………..4](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338859)

[3.1. Требования к функциональным характеристикам 4](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338860)

[3.2. Требования к надежности 4](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338861)

[3.3. Условия эксплуатации 4](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338862)

[3.4. Требования к составу и параметрам технических средств 5](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338863)

[3.5. Требования к информационной и программной совместимости 5](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338864)

[3.6. Требования к транспортированию и хранению 5](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338865)

[3.7. Специальные требования 5](file:///\\apetfs\ИСПП-3\Коновалов_СВ\4%20курс\Учебная%20практика\ИС%20Автотрансопорт\Доки\Приложения\1%20ТЗ..docx#_Toc28338866)

# ВВЕДЕНИЕ

В современном мире, где технологии играют ключевую роль, процессы автоматизации и разработки информационных систем выделяются как особенно трудоемкие и значимые направления. Это обусловлено тем, что автоматизация становится катализатором глубоких трансформаций в управленческой деятельности, оказывая существенное влияние на жизнь людей и общество в целом.

Такое развитие приводит к созданию управленческих систем, цель которых - повышение эффективности и оптимизация работы объектов управления за счет комплексного применения технологий для сбора, обработки, передачи данных и выполнения управленческих решений.

Информационные системы, в этом контексте, служат инструментом для предоставления данных и информации, необходимых уполномоченным лицам организации. Главная задача таких систем - обеспечение доступа к актуальной информации для грамотного и эффективного управления ресурсами организации.

# Основание для разработки

Основой для данной инициативы является соглашение между заинтересованными сторонами. Предметом разработки выступает «Автоматизация системы учета внутриофисных расходов».

# Назначение разработки

Актуальность создания автоматизированной системы учета внутриофисных расходов обусловлена потребностью в ведении эффективного контроля за финансовыми потоками внутри компании, вызванной увеличением объемов внутренних расходов, необходимостью их оптимизации, упрощением процессов утверждения и согласования расходных операций, автоматизацией учета и анализа финансовых данных, экономией времени сотрудников на рутинные операции, обеспечением прозрачности и контролем за эффективностью использования ресурсов предприятия.

# Технические требования к программе или программному изделию

## Требования к функциональным характеристикам.

* Система должна обеспечивать регистрацию сотрудников и авторизацию для доступа к учету внутриофисных расходов.
* Сотрудники должны иметь возможность подавать заявки на расходы, определять их категории и прикреплять соответствующие документы.
* Система должна предоставлять отчеты о запланированных и фактических расходах, анализируя их соответствие бюджетным ограничениям.
* Сотрудники должны иметь доступ к инструкциям и рекомендациям по оформлению заявок на расходы.

## Требования к надежности

Для обеспечения надежности необходимо проверять корректность вводимых данных сотрудником

## Условия эксплуатации

Данный программный продукт разработан для использования в условиях, удобных для оператора компьютера. Предусмотрено, что за поддержку и обслуживание данного программного продукта будет отвечать один сотрудник – администратор. Администратор будет отвечать за стабильную работу программы и ввод данных о сотрудниках, использующих этот программный продукт для учета внутриофисных расходов.

## Требования к составу и параметрам технических средств

Для хорошего функционирования программного средства должны соответствовать:

* процессор Intel Pentium III с тактовой частотой от 75 Мгц и выше;
* оперативная память от 128 Мб;
* жесткий диск c объемом памяти не менее 60 Мб свободного дискового пространства;
* видеокарта, с объемом оперативной памяти 512 Мб;

## Требования к информационной и программной совместимости

Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией операционной системы:

* Windows NT 10.0 (2018),
* Windows NT 10.0 (2019),
* Windows NT 10.0 (2022).

## Требования к транспортированию и хранению

Программа поставляется на разных носителях таких как Usb флеш накопителе или на лазерном носителе информации.

## Специальные требования.

Разрабатываемый программный продукт должен обладать следующими требованиями:

* Программа должна иметь простой интуитивно понятный интерфейс;
* редактировать устаревшую информацию;
* обеспечивать целостность хранимой информации.

# ПРИЛОЖЕНИИ Б

ЛИСТИНГ ПРОГРАММНОГО КОДА

Представлен листинг приведённых выше веб-страниц

Страница авторизации:

<html lang="en">

<head>

<title>Авторизация</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="icon" type="image/png" href="images/icons/favicon.ico">

<link rel="stylesheet" type="text/css" href="vendor/bootstrap/css/bootstrap.min.css">

<link rel="stylesheet" type="text/css" href="fonts/font-awesome-4.7.0/css/font-awesome.min.css">

<link rel="stylesheet" type="text/css" href="vendor/animate/animate.css">

<link rel="stylesheet" type="text/css" href="vendor/css-hamburgers/hamburgers.min.css">

<link rel="stylesheet" type="text/css" href="vendor/select2/select2.min.css">

<link rel="stylesheet" type="text/css" href="css/util.css">

<link rel="stylesheet" type="text/css" href="css/main.css">

<meta name="robots" content="noindex, follow">

</head>

<body>

<div class="limiter">

<div class="container-login100">

<div class="wrap-login100">

<div class="login100-pic js-tilt" data-tilt>

<img src="images/img-01.png" alt="IMG">

</div>

<form class="login100-form validate-form">

<span class="login100-form-title">

Авторизация

</span>

<div class="wrap-input100 validate-input" data-validate="Valid email is required: ex@abc.xyz">

<input class="input100" type="text" name="email" placeholder="Email">

<span class="focus-input100"></span>

<span class="symbol-input100">

<i class="fa fa-envelope" aria-hidden="true"></i>

</span>

</div>

<div class="wrap-input100 validate-input" data-validate="Password is required">

<input class="input100" type="password" name="pass" placeholder="Password">

<span class="focus-input100"></span>

<span class="symbol-input100">

<i class="fa fa-lock" aria-hidden="true"></i>

</span>

</div>

  <div class="container-login100-form-btn">

    <button class="login100-form-btn">

      <a href="content.html" class="login100-form-btn">Войти</a>

    </button>

 </div>

<div class="text-center p-t-136">

<a class="txt2" href="#">

Create your Account

<i class="fa fa-long-arrow-right m-l-5" aria-hidden="true"></i>

</a>

</div>

</form>

</div>

</div>

</div>

<script src = "vendor/jquery/jquery-3.2.1.min.js"></script>

<script src="vendor/bootstrap/js/popper.js"></script>

<script src="vendor/bootstrap/js/bootstrap.min.js"></script>

<script src="vendor/select2/select2.min.js"></script>

<script src="vendor/tilt/tilt.jquery.min.js"></script>

<script>

    $('.js-tilt').tilt({

      scale: 1.1

    })

  </script>

<script async src="https://www.googletagmanager.com/gtag/js?id=UA-23581568-13"></script>

<script>

  window.dataLayer = window.dataLayer || [];

  function gtag(){dataLayer.push(arguments);}

  gtag('js', new Date());

  gtag('config', 'UA-23581568-13');

</script>

<script src="js/main.js"></script>

<script defer src="https://static.cloudflareinsights.com/beacon.min.js/v84a3a4012de94ce1a686ba8c167c359c1696973893317" integrity="sha512-euoFGowhlaLqXsPWQ48qSkBSCFs3DPRyiwVu3FjR96cMPx+Fr+gpWRhIafcHwqwCqWS42RZhIudOvEI+Ckf6MA==" data-cf-beacon='{"rayId":"85b629f269977035","b":1,"version":"2024.2.1","token":"cd0b4b3a733644fc843ef0b185f98241"}' crossorigin="anonymous"></script>

</body>

</html>

Страница сотрудника:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="css/style.css">

    <title>Document</title>

</head>

<body>

    <header>

        <div class="container">

            <div class="logo">

                <a href="/">Логотип</a>

            </div>

            <nav>

                <ul class="nav-links">

                 <!-- Замените # на реальные ссылки -->

                    <li><a href="#">Мои заявки</a></li>

                    <!-- Добавьте дополнительные ссылки по необходимости -->

                </ul>

            </nav>

            <div class="user-info">

                <span>Привет, {Имя Пользователя}</span> <!-- Динамически замените Имя Пользователя -->

                <a href="/index.html">Выйти</a> <!-- Ссылка для выхода -->

            </div>

        </div>

    </header>

    <div class="content\_container">

        <div class="background-image"></div>

        <div class="form-container">

            <h2>Добавление заявки на товар</h2>

            <form id="expense-form">

                <div class="form-group">

                    <label for="item-name">Название товара</label>

                    <input type="text" id="item-name" name="item-name" required>

                </div>

                <div class="form-group">

                    <label for="item-category">Категория товара</label>

                    <select id="item-category" name="item-category" required>

                        <option value="">Выберите категорию</option>

                        <option value="office-supplies">Канцтовары</option>

                        <option value="software">Программное обеспечение</option>

                        <option value="household-items">Хозтовары</option>

                        <!-- Добавьте другие категории по необходимости -->

                    </select>

                </div>

                <div class="form-group">

                    <label for="item-cost">Сумма расхода</label>

                    <input type="number" id="item-cost" name="item-cost" required>

                </div>

                <div class="form-group">

                    <label for="purchase-date">Дата покупки</label>

                    <input type="date" id="purchase-date" name="purchase-date" required>

                </div>

                <div class="form-group">

                    <label for="receipt-upload">Загрузить чек (необязательно)</label>

                    <input type="file" id="receipt-upload" name="receipt-upload">

                </div>

                <div class="form-group">

                    <label for="comments">Комментарии</label>

                    <textarea id="comments" name="comments" rows="4"></textarea>

                </div>

                <button type="submit">Добавить заявку</button>

            </form>

        </div>

    </div>

    <div id="notification" class="notification">

        Заявка успешно отправлена!

    </div>

    <script src="main.js"></script>

</body>

</html>

Страница Зав.хоза

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="css/style.css">

<title>Страница заведующего хозяйством</title>

</head>

<body>

<header>

<div class="container">

<div class="logo">

<a href="/">Логотип</a>

</div>

<nav>

<ul class="nav-links">

<li><a href="#">Мои заявки</a></li>

<li><a href="#">Обработать заявки</a></li>

<li><a href="#">Сформировать отчет</a></li>

<!-- Добавьте другие ссылки по необходимости -->

</ul>

</nav>

<div class="user-info">

<span>Привет, {Имя Пользователя}</span> <!-- Динамически замените Имя Пользователя -->

<a href="/index.html">Выйти</a> <!-- Ссылка для выхода -->

</div>

</div>

</header>

<div class="content\_container">

<div class="background-image"></div>

<div class="form-container">

<h2>Заказ товара</h2>

<form id="order-form">

<div class="form-group">

<label for="item-name">Название товара</label>

<input type="text" id="item-name" name="item-name" required>

</div>

<div class="form-group">

<label for="item-category">Категория товара</label>

<select id="item-category" name="item-category" required>

<option value="">Выберите категорию</option>

<option value="office-supplies">Канцтовары</option>

<option value="software">Программное обеспечение</option>

<option value="household-items">Хозтовары</option>

<!-- Добавьте другие категории по необходимости -->

</select>

</div>

<div class="form-group">

<label for="item-quantity">Количество</label>

<input type="number" id="item-quantity" name="item-quantity" required>

</div>

<button type="submit">Заказать товар</button>

</form>

</div>

</div>

<div class="content\_container">

<div class="background-image"></div>

<div class="form-container">

<h2>Обработка заявок</h2>

<!-- Здесь может быть форма или таблица для обработки заявок -->

</div>

</div>

<div class="content\_container">

<div class="background-image"></div>

<div class="form-container">

<h2>Формирование отчета</h2>

<!-- Здесь может быть форма или таблица для формирования отчета -->

</div>

</div>

<div id="notification" class="notification">

Успешное действие!

</div>

<script src="main.js"></script>

</body>

</html>