СОДЕРЖАНИЕ

[Введение 4](#_Toc122133327)

[1 Обзор и анализ информационной системы приюта для бездомных   
животных 6](#_Toc122133328)

[1.1 Существующие аналоги информационной системы 6](#_Toc122133329)

[1.2 Обоснование современных технологий разработки информационной системы 8](#_Toc122133330)

[1.3 Выводы и постановка задач на дипломное проектирование 9](#_Toc122133331)

[2 Эргономическое проектирование информационной системы 11](#_Toc122133332)

[2.1 Анализ функций и их распределение в проектируемой системе 11](#_Toc122133333)

[2.2 Разработка алгоритмов работы пользователей программного комплекса 14](#_Toc122133334)

[2.3 Разработка эргономических требований и сценария информационного взаимодействия 18](#_Toc122133335)

[3 Разработка информационной системы приюта для бездомных животных 35](#_Toc122133336)

[3.1 Разработка структуры информационной системы 35](#_Toc122133337)

[3.2 Разработка алгоритмов работы информационной системы 39](#_Toc122133338)

[3.3 Разработка и тестирование программного модуля 42](#_Toc122133339)

[4 Технико-экономическое обоснование эффективности разработки и использования информационной системы приюта для бездомных животных 52](#_Toc122133340)

[4.1 Краткое описание разработки 52](#_Toc122133341)

[4.2 Расчёт затрат на разработку программного средства 52](#_Toc122133342)

[4.3 Экономический эффект при разработке по для свободной реализации на рынке информационных технологий 56](#_Toc122133343)

[5 Охрана труда. Реализация пространственно-антропометрической эргономической совместимости работника и технического средства при организации рабочего места 58](#_Toc122133344)

[Заключение 64](#_Toc122133345)

[Список использованных источников 66](#_Toc122133346)

[Приложение А (обязательное). Листинг программы 68](#_Toc122133347)

# ВВЕДЕНИЕ

Стремительное развитие программных средств, наделенных различными функциональными возможностями, способствовало формированию новых принципов в различных сферах деятельности человека. На современном этапе развития вычислительной техники прослеживается тенденция внедрения информационных технологий, в том числе различного рода систем автоматизации, что приводит к оптимизации деятельности организации и позволяет в последствии экономить финансовые, трудовые и иных видов ресурсов. Это тенденция не обошла стороной и сферу помощи брошенным животным.

В настоящее время зооприюты и иные организации, оказывающие помощь питомцам, остро нуждаются в специализированных программных продуктах. Таким образом, создание информационной системы зооприюта является актуальным и своевременным решением.

Целью диплома является разработка веб-приложения, позволяющего автоматизировать работу зооприюта.

Для достижения цели поставлены следующие задачи:

* проанализировать аналоги существующих информационных систем;
* произвести эргономическое проектирование информационной системы;
* обосновать экономический эффект от разработки информационной системы;
* произвести реализацию пространственно-антропометрической эргономической совместимости работника и технического средства при организации рабочего места;
* произвести наладку и тестирование программного обеспечения.

# **1 ОБЗОР И АНАЛИЗ ИНФОРМАЦИОННОЙ СИСТЕМЫ** ПРИЮТА ДЛЯ БЕЗДОМНЫХ ЖИВОТНЫХ

# Существующие аналоги информационной системы

Информационная система – это компьютерная система (программа или веб-сайт), обеспечивающая хранение информации, организованной в определенные структуры и предоставление доступа к хранимой информации одновременно множеству заинтересованных лиц [1].

Каждый день десятки животных становятся бездомными, кого-то выбросили, кто-то потерялся, кто-то был рожден на улиц. Проблема бездомных животных всегда есть, и работа приютов направлена на её решение.

Существует огромное множество интернет-ресурсов, поэтому у каждой компании есть свой веб-сайт. Посетители таких ресурсов напрямую взаимодействуют с компанией в сети. Веб-ресурс – это визитная карточка любой компании. Создание и разработка веб-ресурса – это способ расширения границ деятельности компании.

Под веб-ресурсом, обычно подразумевается одна или несколько логически связанных между собой веб-страниц. Как правило, сайт в интернете представляет собой массив связанных данных, имеющий уникальный адрес и воспринимаемый пользователями как единое целое [2].

К сожалению, сайтов зооприютов и похожей тематики крайне мало в РБ, поэтому актуальность темы помощи брошенным животным высока.

Примерами сайтов зооприютов являются:

* сайт «egida.by»;
* сайт «zooshans.by».

Рассмотрим каждый из них более подробно.

Сайт «egida.by». Это благотворительный портал защиты и прав животных. Эгида – это общественная организация основанная 12 женщинами, которые в свободное время ведут волонтерскую работу в приюте [3].

Интерфейс сайта представлен на рисунке 1.1.

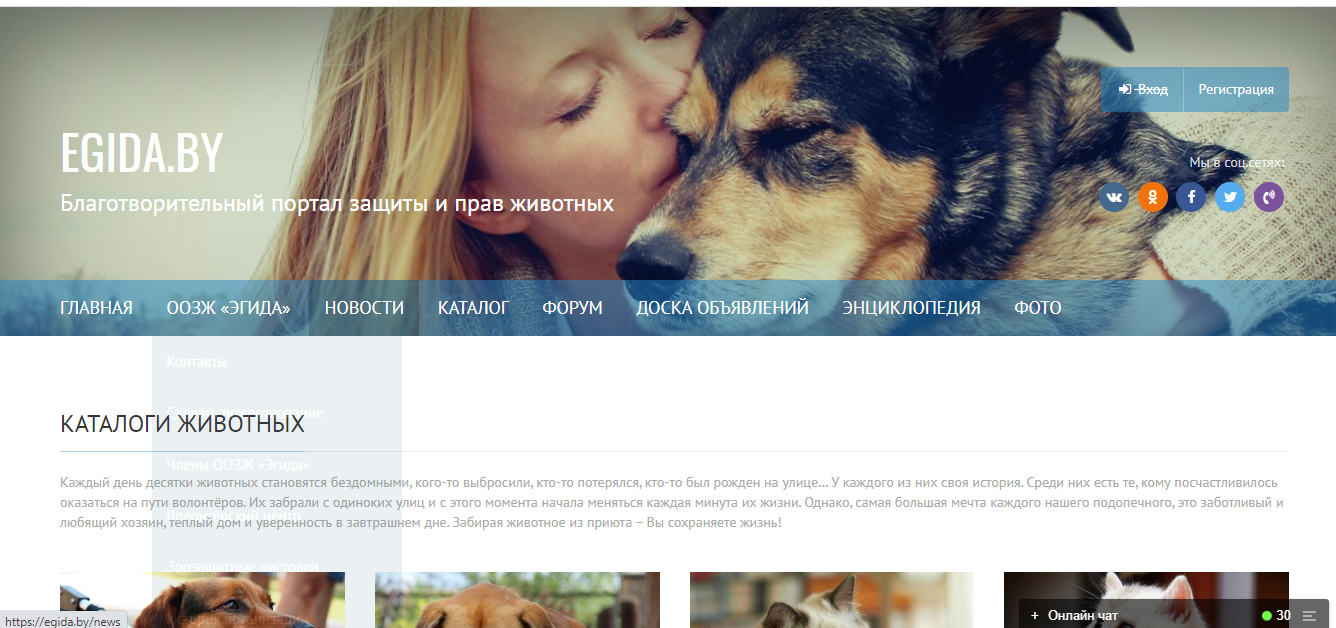


Рисунок 1.1 – Интерфейс сайта «egida.by»

Достоинства информационной системы являются:

* наличие энциклопедии;
* наличие доски объявлений;
* наличие форума.

Недостатки информационной системы являются

* отсутствие поисковой системы;
* отсутствие оставления заявки онлайн;
* маленький список категорий животных.

Сайт «zooshans.by». Это сайт благотворительного объединения “ЗООшанс”, помогающего бездомным животным [4].

Интерфейс сайта представлен на рисунке 1.2.

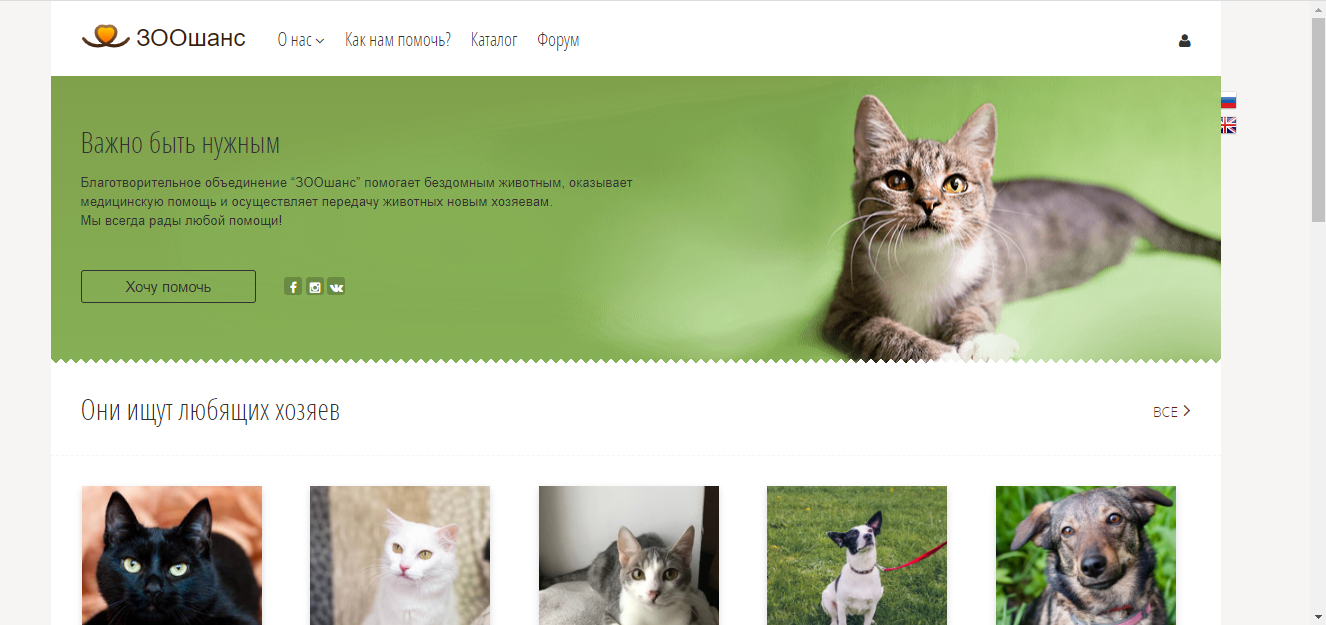


Рисунок 1.2 – Интерфейс сайта «zooshans.by»

Достоинства информационной системы являются:

* наличие 2 языков поддержки;
* наличие форума.

Недостатки информационной системы являются

* отсутствие поисковой системы;
* отсутствие новостей.

# Обоснование современных технологий разработки информационной системы

Для реализации дипломного проекта был выбран объектно-ориентированный язык программирования C# на платформе ASP.NET Core. База данных реализуется с помощью SQL Server. Доступ к данным реализуется через Entity Framework [5].

ASP.NET Core – свободно-распространяемый кроссплатформенный фреймворк для создания веб-приложений на платформе .NET с открытым исходным кодом. Данная платформа разрабатывается компанией Майкрософт совместно с сообществом и имеет большую производительность по сравнению с ASP.NET [6].

ASP.NET Core предоставляет следующие преимущества:

* единое решение для создания пользовательского веб-интерфейса и веб-API (Application Programming Interface);
* разработано для тестируемости;
* Razor Pages упрощает написание кода для сценариев страниц и повышает его эффективность;
* возможность разработки и запуска в ОС Windows, macOS и Linux;
* открытый исходный код и ориентация на сообщество;
* встроенное введение зависимостей;
* облачная система конфигурации на основе среды;
* интеграция современных клиентских платформ и рабочих процессов разработки;
* упрощенный высокопроизводительный модульный конвейер HTTP-запросов (HyperText Transfer Protocol – протокол передачи гипертекста);
* Razor Pages упрощает написание кода для сценариев страниц и повышает его эффективность.

ASP.NET MVC Framework – фреймворк для создания веб-приложений, который реализует шаблон Model-view-controller. Концепция паттерна MVC предполагает разделение приложения на три компонента: модель, представление, контроллер [7].

Entity Framework представляет собой объектно-ориентированную, легковесную и расширяемую технологию от компании Microsoft для доступа к данным. EF Core является ORM-инструментом (object-relational mapping - отображения данных на реальные объекты) [8].

SQL – язык структурированных запросов, основной задачей которого является предоставление простого способа считывания и записи информации в базу данных [9].

MS SQL Server – это платформа для решения критически важных задач в масштабе предприятия, обладающая высокой доступностью, повышенной производительностью и безопасностью. Решение представляет собой хорошо масштабируемый, полностью реляционный, быстродействующий сервер, способный обрабатывать большие объемы данных для клиент-серверных приложений [10].

# Выводы и постановка задач на дипломное проектирование

Целью диплома является разработка информационной системы приюта для бездомных животных, которая представляет собой веб-приложение, реализующее возможность просмотра питомцев и автоматизации работы администратора. В результате анализа существующих информационных систем, были выявлены следующие недостатки: отсутствие системы поиска, возможности оставления заявки онлайн, отсутствие новостей. Данные недостатки будут учтены при разработке информационной системы в рамках дипломного проекта.

Для достижения цели поставлены следующее задачи:

* проанализировать аналоги существующих информационных систем;
* произвести эргономическое проектирование информационной системы;
* обосновать экономический эффект от разработки информационной системы;
* произвести реализацию пространственно-антропометрической эргономической совместимости работника и технического средства при организации рабочего места.

Разрабатываемая информационная система должна реализовывать следующие функции:

– регистрация пользователя;

– авторизация пользователя;

– просмотра всех питомцев в выбранной категории;

– просмотра новостей на главной странице;

– поиск питомцев по имени;

– составление заявки о желании взять питомца;

– добавления питомца в базу данных;

– удаление питомца из базы данных;

– редактирование информации о питомце;

– добавление новости в базу данных;

– удаление новости из базы данных;

– редактирование новости;

– удаление заявки из базы данных;

– редактирование заявки о взятии питомца себе.

Таким образом будет разработана информационная система, использование которой повысит качество и эффективность работы зооприютов, дает возможность выбрать себе питомца, а также окажет положительный социальный эффект.

Для разработки информационной системы будет использован следующий стек технологий: C#, Entity Framework, SQL Server, ASP.NET Core.

Интерфейс должен быть интуитивно понятным, удобным и эргономичным.

# ЭРГОНОМИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ

# 2.1 Анализ функций и их распределение в проектируемой системе

Системой будут пользоваться три группы пользователей: администраторы, неавторизированные пользователи, авторизованные пользователи.

Неавторизированному пользователю доступны следующие функции:

1. Регистрация пользователя осуществляется после нажатия кнопки «Регистрация». Пользователю будет представлена форма, в которую необходимо ввести логин и пароль.
2. Просмотр информации о питомцах и новостях, представленной в информационной системе. Для этого пользователю необходимо перейти по релевантной ссылке.
3. Поиск информации о питомцах по имени. Для этого в поле поиска в верхней части экрана необходимо ввести кличку питомца.

Все нижеперечисленные функции, выполняемые системой «человек – компьютер – среда» (СЧКС), доступны только авторизованному сотруднику:

1. Авторизация осуществляется после ввода логина, пароля и последующем нажатии кнопки «Войти». При повторном входе в систему возможно автоматическое заполнение формы входа для удобства и экономии времени.
2. Оставление заявки для взятия питомца себе. Для этого необходимо кликнуть на кнопку «Подробнее» на выбранном питомце, и после перехода кликнуть на кнопку «Взять себе».
3. Для выхода из системы, необходимо нажать кнопку «Выйти» в верхнем правом углу экрана.

Все нижеперечисленные функции, выполняемые системой «человек – компьютер – среда» (СЧКС), доступны только привилегированным сотрудникам – администраторам:

1. Для добавления питомца, необходимо перейти на страницу работы с питомцами, и нажать на кнопку «Добавить». После перехода на страницу добавления, необходимо корректно заполнить все поля ввода, и нажать на кнопку «Добавить».
2. Для редактирования информации о питомце, необходимо на странице работы с питомцами нажать на кнопку «Редактировать» на конкретном питомце. После перехода на страницу редактирования, необходимо исправить информацию в полях для ввода, после чего нажать на кнопку «Сохранить».
3. Для удаления данных о питомце, необходимо на странице работы с питомцами нажать на кнопку «Удалить» на конкретном питомце. После перехода на страницу удаления, необходимо подтвердить удаление, после чего нажать на кнопку «Удалить».
4. Для добавления новости, необходимо перейти на страницу работы с новостями, и нажать на кнопку «Добавить». После перехода на страницу добавления, необходимо корректно заполнить все поля ввода, и нажать на кнопку «Добавить».
5. Для редактирования информации о новости, необходимо на странице работы с новостями нажать на кнопку «Редактировать» на конкретной новости. После перехода на страницу редактирования, необходимо исправить информацию в полях для ввода, после чего нажать на кнопку «Сохранить».
6. Для удаления новости, необходимо на странице работы с новостями нажать на кнопку «Удалить» на конкретной новости. После перехода на страницу удаления, необходимо подтвердить удаление, после чего нажать на кнопку «Удалить».
7. Для добавления заявки, необходимо перейти на страницу работы с заявками, и нажать на кнопку «Добавить». После перехода на страницу добавления, необходимо корректно заполнить все поля ввода, и нажать на кнопку «Добавить».
8. Для редактирования информации о заявке, необходимо на странице работы с заявками нажать на кнопку «Редактировать» на конкретной заявке. После перехода на страницу редактирования, необходимо исправить информацию в полях для ввода, после чего нажать на кнопку «Сохранить».
9. Для удаления заявки, необходимо на странице работы с заявками нажать на кнопку «Удалить» на конкретной заявке. После перехода на страницу удаления, необходимо подтвердить удаление, после чего нажать на кнопку «Удалить».

Распределение функций в СЧМС между человеком и техническими устройствами осуществляется на основе следующих принципов:

1. Человек выполняет те функции, которые нет возможности автоматизировать, либо их автоматизация понесет за собой большие затраты.
2. Машина, наоборот, будет выполнять только те функции, которые не требуют непосредственного контроля человека. Некоторая функциональность зачастую выполняется машиной, чтобы исключить монотонность труда человека.

Учитывая названные принципы, проведем анализ функций нашей системы с целью распределения их между человеком и компьютером. Результаты работы представим в таблице 2.1.

Таблица 2.1 – Распределение функций между человеком и компьютером в проектируемой СЧКС

|  |  |  |
| --- | --- | --- |
| Название функции | Кто ее выполняет в СЧМС | Обоснование распределения функций |
| 1 | 2 | 3 |
| Регистрация | Пользователь | Функция остается за человеком в соответствии с назначением системы. |
| Авторизация | Администратор, пользователь | Система делает это четко и однозначно, работая в соответствии с программой. |
| Добавление новости. | Администратор | Функция остается за человеком в соответствии с назначением системы. |
| Просмотр данных | Администратор, пользователь | Функция остается за человеком в соответствии с назначением системы. |
| Удаление новости. | Администратор | Функция остается за человеком в соответствии с назначением системы. |
| Редактирование новости | Администратор | Функция остается за человеком в соответствии с назначением системы. |
| Оформление заявки. | Администратор, пользователь | Функция остается за человеком в соответствии с назначением системы. |
| Добавление питомца. | Администратор | Функция остается за человеком в соответствии с назначением системы. |
| Редактирование данных о питомце | Администратор | Функция остается за человеком в соответствии с назначением системы. |

Продолжение таблицы 2.1

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Удаление данных о питомце. | Администратор | Функция остается за человеком в соответствии с назначением системы. |
| Удаление заявки. | Администратор. | Функция остается за человеком в соответствии с назначением системы. |
| Поиск данных о питомце | Администратор, пользователь | Функция остается за человеком в соответствии с назначением системы |
| Редактирование заявки | Администратор. | Функция остается за человеком в соответствии с назначением системы. |
| Выход из программы. | Администратор, пользователь | Функция остается за человеком в соответствии с назначением системы. |

В этом подразделе мы установили, что в приложении будут 2 системы:

* пользователь-компьютер-среда;
* администратор-компьютер-среда.

# 2.2 Разработка алгоритмов работы пользователей программного комплекса

Пользователями являются:

* администратор, который осуществляет работу с данными;
* пользователь, просматривающий контент и взаимодействующий с ним.

Ниже описаны следующие пользователи и функции, которые они выполняют:

Администратор:

* авторизация как администратор;
* добавление контента;
* редактирование контента;
* удаление контента;
* просмотр контента;
* поиск контента по заданным параметрам.

Пользователь:

* просмотр контента;
* поиск контента по заданным параметрам;
* авторизация и регистрация;
* оставление заявки о взятии питомца.

Администратору также доступны все возможности пользователя.

Алгоритм работы человека в системе «Пользователь – компьютер – среда» представлен в таблице 2.2.

Таблица 2.2 – Алгоритм работы человека в подсистеме «Пользователь – компьютер – среда»

|  |  |  |
| --- | --- | --- |
| Содержание операции | Обращения к средствам отображения информации | Обращения к органам управления |
| 1 | 2 | 3 |
| Запуск веб-ресурса. | Ссылка в браузере | Щелчок мышкой по ссылке |
| Ввод логина и пароля для входа в систему | Форма с полями «Логин» и «Пароль» | Манипуляции с мышью, клавиатурой |
| Переход в программу | Кнопка «Войти» | Щелчок мышкой |
| Просмотр новостей | Пункт меню «Главная» | Щелчок мышкой |
| Просмотр питомцев-собак | Пункт меню «Собаки» | Щелчок мышкой |
| Просмотр питомцев-котов | Пункт меню «Коты» | Щелчок мышкой |
| Поиск питомца | Поле «Поиск» и кнопка «Поиск» | Манипуляции с мышью, клавиатурой |
| Переход на страницу с подробным отображением новости | Пункт меню «Главная» и кнопка «Подробнее» под выбранной новостью | Щелчок мышкой |
| Переход на страницу с картой питомца | Пункт меню «Собаки» и кнопка «Подробнее» под выбранным питомцем | Щелчок мышкой |

Продолжение таблицы 2.2

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Оставление заявки | Страница с картой питомца и кнопка «Оставить заявку» | Щелчок мышкой |
| Выход из системы | Кнопка «Выход» | Щелчок мышкой |

Алгоритм работы человека в подсистеме «Администратор – компьютер – среда» представлен в таблице 2.3.

Таблица 2.3 – Алгоритм работы человека в подсистеме «Администратор – ПК – среда»

|  |  |  |
| --- | --- | --- |
| Содержание операции | Обращения к средствам отображения информации | Обращения к органам управления |
| 1 | 2 | 3 |
| Запуск веб-ресурса. | Ссылка в браузере | Щелчок мышкой по ссылке |
| Ввод логина и пароля для входа в систему | Форма с полями «Логин» и «Пароль» | Манипуляции с мышью, клавиатурой |
| Переход в программу | Кнопка «Войти» | Щелчок мышкой |
| Просмотр данных о питомцах | Пункт меню «Работа с питомцами» | Щелчок мышкой |
| Переход на страницу добаления питомца | Кнопка «Добавление питомцев» | Щелчок мышкой |
| Добавление нового питомца | Поля для добавления и кнопка «Добавить» | Манипуляции с мышью, клавиатурой |
| Переход на страницу редактирования данных конкретного питомца | Кнопка «Редактировать» | Щелчок мышкой |
| Переход на страницу добаления питомца | Кнопка «Добавление питомцев» | Щелчок мышкой |
| Добавление нового питомца | Поля для добавления и кнопка «Добавить» | Манипуляции с мышью, клавиатурой |

Продолжение таблицы 2.3

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Переход на страницу редактирования данных конкретного питомца | Кнопка «Редактировать» | Щелчок мышкой |
| Сохранения новых данных о питомце | Кнопка «Сохранить» | Манипуляции с мышью, клавиатурой |
| Переход на страницу удаления данных о питомце | Кнопка «Удаление» | Щелчок мышкой |
| Удаление данных о питомце | Кнопка «Удалить» | Манипуляции с мышью |
| Просмотр данных о заявках | Пункт меню «Работа с обращениями» | Просмотр данных о питомцах |
| Переход на страницу добаления заявки | Кнопка «Добавление обращений» | Щелчок мышкой |
| Добавление новой заявки | Поля для добавления и кнопка «Добавить» | Манипуляции с мышью, клавиатурой |
| Переход на страницу редактирования данных конкретного заявки | Кнопка «Редактировать» | Щелчок мышкой |
| Сохранения новых данных по заявке | Кнопка «Сохранить» | Манипуляции с мышью, клавиатурой |
| Переход на страницу удаления заявки | Кнопка «Удаление» | Щелчок мышкой |
| Удаление заявки | Кнопка «Удалить» | Манипуляции с мышью |
| Просмотр данных о новостях | Пункт меню «Работа с новостями» | Просмотр данных о питомцах |
| Переход на страницу добаления новости | Кнопка «Добавление новостей» | Щелчок мышкой |

Продолжение таблицы 2.3

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Добавление новой новости | Поля для добавления и кнопка «Добавить» | Манипуляции с мышью, клавиатурой |
| Переход на страницу редактирования данных конкретной новости | Кнопка «Редактировать» | Щелчок мышкой |
| Сохранения новых данных о новости | Кнопка «Сохранить» | Манипуляции с мышью, клавиатурой |
| Переход на страницу удаления новости | Кнопка «Удаление» | Щелчок мышкой |
| Удаление новости | Кнопка «Удалить» | Манипуляции с мышью |
| Переход на страницу добавления нового ветеринара | Кнопка «Добавить» | Щелчок мышкой |
| Выход из системы | Кнопка «Выход» | Щелчок мышкой |

# 2.3 Разработка эргономических требований и сценария информационного взаимодействия

Под эргономическими требованиями понимаются требования к СЧМС, ее отдельным подсистемам и рабочей среде, определяемые свойствами человека и устанавливаемые для обеспечения его комфортной, эффективной и безопасной деятельности.

Эргономические требования (ЭТ) определяются свойствами человека-оператора и устанавливаются с целью оптимизации его деятельности, они являются базовыми при проектировании СЧМС на основе антропоцентрического подхода.

Эргономические требования должны предъявляться как к характеристикам машины (и к различным ее элементам), так и к человеку-оператору, включенному в эргатическую систему, создаваемую на основе данной машины.

Сформулировать эти требования и реализовать их не просто, т. к. свойства человека весьма многочисленны и разнообразны, они характеризуют человека как анатомическую, физиологическую, психофизиологическую и психологическую системы.

Под эргономическими требованиями к СЧКС понимаются требования к СЧКС, ее отдельным подсистемам, оборудованию, рабочей среде, определяемые свойствами человека и устанавливаемые для обеспечения его эффективной и безопасной деятельности. Эргономические требования определяются свойствами человека-оператора и классифицируются в соответствии с ними:

* антропометрические требования – определяются анатомическими, морфологическими и биомеханическими свойствами человека;
* физиологические требования – учитывают энергетические и скоростные возможности человека;
* психофизиологические требования – обусловлены возможностями и особенностями сенсорных систем человека;
* психологические требования – определяют соответствие СЧКС и ее элементов психологическим особенностям человека;
* гигиенические требования – определяют безопасные условия работы человека;
* социально-психологические требования – определяют соответствие конструкции машины и организации рабочих мест характеру и степени группового взаимодействия [11].

При формулировании эргономических требований и их влиянии на характеристики эргатической системы имеет место логическая последовательность, изображенная на рисунке 2.1.

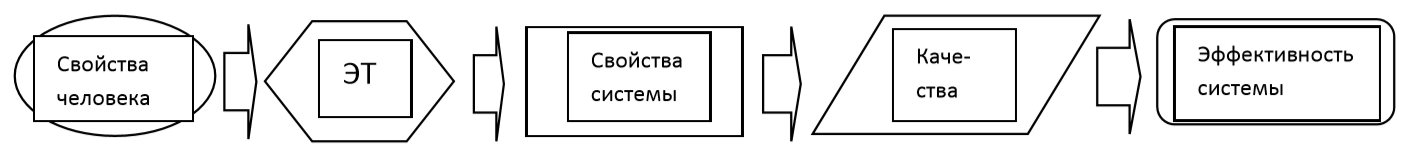


Рисунок 2.1– Логика формулирования эргономических требований и их влияния на характеристики СЧМ

Под свойствами человека-оператора понимают его антропометрические, психофизиологические, физиологические и психологические характеристики, возможности и особенности.

Свойства системы определяются структурными, функциональными, энергетическими, информационными взаимодействиями и отношениями составляющих ее элементов.

Эргономические требования формируются на основании экспериментальных исследований и опыта эксплуатации систем «человек − машина – среда».

Эргономические требования необходимы для обеспечения:

* рационального распределения функций в СЧМ;
* оптимальной организации рабочего места на основе учета в конструкции рабочих характеристик и свойств человека;
* соответствия технических средств возможностям человека по приему и переработке информации и осуществлению управляющих воздействий;
* оптимальных для жизнедеятельности и работоспособности человека показателей производственной среды.

Описание конкретных свойств, определяющих значения групповых эргономических показателей качества СЧКС таких, как антропометрического, физиологического, психофизиологического, психологического и гигиенического находятся в таблице 2.4.

Таблица 2.4 – Общие эргономические требования проектируемой СЧКС

|  |  |
| --- | --- |
| Группа эргономических требований | Номенклатура эргономических требований |
| 1 | 2 |
| Психологические | – привлечение внимания пользователей к важным сообщениям;  – наличие индикатора степени выполнения операций;  – отсутствие в текстовых сообщениях приложения нецензурной лексики;  – краткое и понятное пользователю сообщение об ошибках;  – использование в проектировании приложения привычных пользователю паттернов управления и взаимодействия;  – отсутствие действий, для которых нужна предварительная подготовка системы или пользователя;  – наличие предупреждений о нежелательных последствиях некоторых действий; |

Продолжение таблицы 2.4

|  |  |
| --- | --- |
| 1 | 2 |
| Физиологические ЭТ | ‒ отсутствие жестов управления, не соответствующим физиологическим возможностям пользователей;  ‒ соответствие компоновки ОУ принципам экономии рабочих движений; |
| Антропометрические ЭТ | ‒ отсутствие действий управления системой, выходящих за пределы антропометрических возможностей человека;  ‒ соответствие размеров кликабельных областей антропометрическим характеристикам человека; |
| Психофизиологические ЭТ | – соответствие размеров знаков на экране дисплея оперативному порогу зрения человека;  – соответствие контраста знаков и фона оптимальным условиям восприятия;  – соответствие вида контраста знаков и фона уровню освещенности рабочего места;  – соответствие расположения надписей условиям их оптимального считывания;  – соответствие размеров текста в зависимости от его важности;  – отсуствие повышенной нагрузки на зрительную систему пользователя со стороны графического оформления приложения;  ‒ отсутствие требований, связанных с обслуживанием системы, несоответствующих возможностям органов чувств человека. |
| Гигиенические ЭТ | * соответствие параметров изображения на экране дисплея условиям комфорта зрительной работы пользователей (отсутствие мельканий, слепящих яркостей и т.п.);   соответствие параметров рабочей среды гигиеническим нормативам; |

Продолжение таблицы 2.4

|  |  |
| --- | --- |
| 1 | 2 |
| Физиологические ЭТ | ‒ отсутствие жестов управления, не соответствующим физиологическим возможностям пользователей;  ‒ соответствие компоновки ОУ принципам экономии рабочих движений; |

Далее следует рассмотреть более детально номенклатуру эргономических требований к разрабатываемому программному продукту. В связи с тем, что разрабатываемый продукт является системой автоматизированного тестирования, выделить определенные характеристики, которые будут отвечать рабочему месту пользователя будет сложно. Поэтому основными эргономическими требованиями являются требования к информации, предъявляемой на экране.

Яркость знаков не должна быть менее 35 кд/м2 для дисплеев на электронно-лучевых трубках и не менее 20 кд/м2 для дисплеев с плоскими экранами. Разница яркости рабочего поля экрана и знаковых элементов не должна быть более 20%. Соотношение яркостной контраст изображения и фона должен быть не менее, чем 3:1. Ширина контура знака может варьироваться в пределах от 0,25 до 0,5 мм, а изменение размеров однотипных знаков на рабочем поле не должно превышать 5%.

Число цветов, одновременно отображаемых на экране должно быть минимальным, а при необходимости проведения быстрого поиска, основанного на опознании цветов, и когда параметры цвета вызываются из памяти электронной вычислительной машины, следует применять не более 6 цветов.

Контраст изображения по отношению к фону должен быть оптимальным: для графической информации необходимо использование прямого контраста, для текстовой – обратного.

Яркость цветов объектов по отношению к фону должна обеспечивать равномерное распределение яркости, яркостной контраст должен быть не менее, чем 60%.

Следует использовать оптимальный выбор цветов для смыслового противопоставления объектов: красный – зеленый, синий – желтый, белый – черный.

Должно обеспечиваться оптимальное сочетание цвета и яркости изображения: красный – при высокой яркости, зеленый – в среднем диапазоне, желтый – в широком диапазоне, синий – при малой яркости

Экран не должен быть засорен элементами. Пользователю необходимо предъявлять только те элементы, которые нужны ему для достижения цели, заданной его действиями. На рабочем экране допускается размещение не более чем 4-6 второстепенных объектов.

Надписи, обозначающие объекты или органы управления, должны быть краткими, однозначно воспринимаемыми и читаться так, как принято в целевом регионе (в данном случае – слева направо). Словарный состав приложения должен включать общеупотребимые, известные пользователю слова. Разрешается использование терминологии, если это соотносится с целевой аудиторией. Сокращение слов нежелательно, но в крайнем случае допускается использование стандартных сокращений.

Параметры предъявляемого на экране текста должны удовлетворять следующим требованиям:

* высота знака – не менее 3 мм;
* отношение ширины буквы, цифры к высоте – в пределах 0.76 – 0.80;
* толщина линии обводки в прямом контрасте – в пределах 10 – 15% от высоты знака;
* в обратном контрасте – в пределах 12 – 16% от высоты знака;
* расстояние между знаками – не менее 30% от высоты знака;
* расстояние между строками – 1,5 – 2 высоты знака;
* длина строки – 40 – 80 знакомест.

Запуск информационной системы производится путем запуска браузера, после чего вводит в адресную строку адрес веб-приложения. Окно главной страницы предоставлено на рисунке 2.1

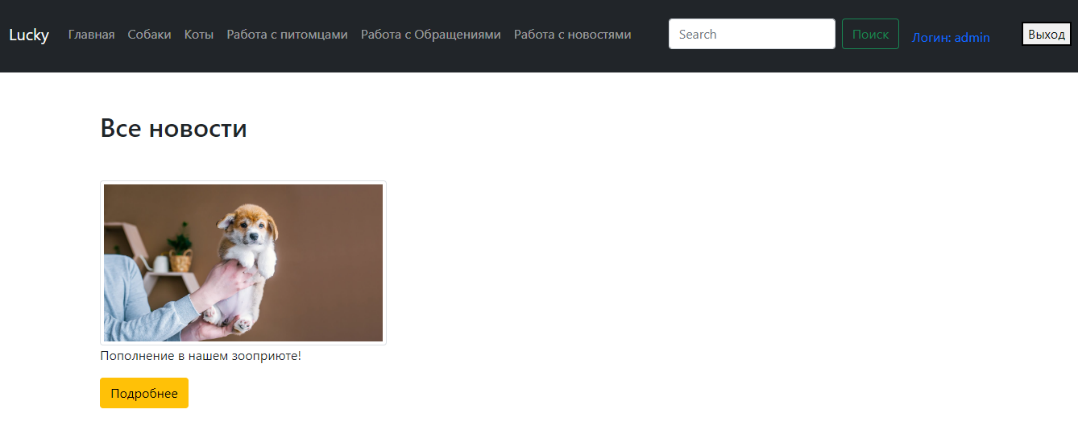


Рисунок 2.1 – Главная страница

Если пользователь не авторизован, он может перейти на страницу авторизации. Окно авторизации представлено на рисунке 2.2.

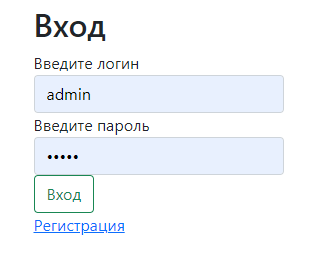


Рисунок 2.2 – Окно авторизации

Если пользователь не зарегистрирован, то он может перейти на страницу регистрации. Окно регистрации представлено на рисунке 2.3.

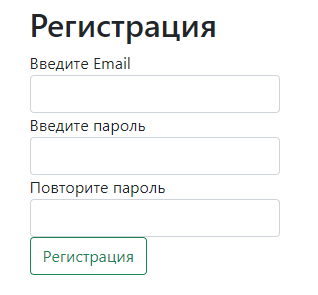


Рисунок 2.3 – Окно регистрации

Просматривая питомцев, пользователь может перейти на страницу подробной информации о питомце. Окно с подробной информацией о питомце представлено на рисунке 2.4.

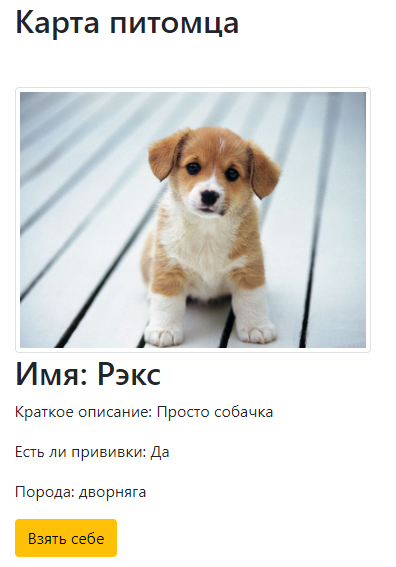


Рисунок 2.4 – Окно с подробной информацией о питомце

На рисунке 2.5 изображён интерфейс страницы работы с питомцем, доступная администратору.

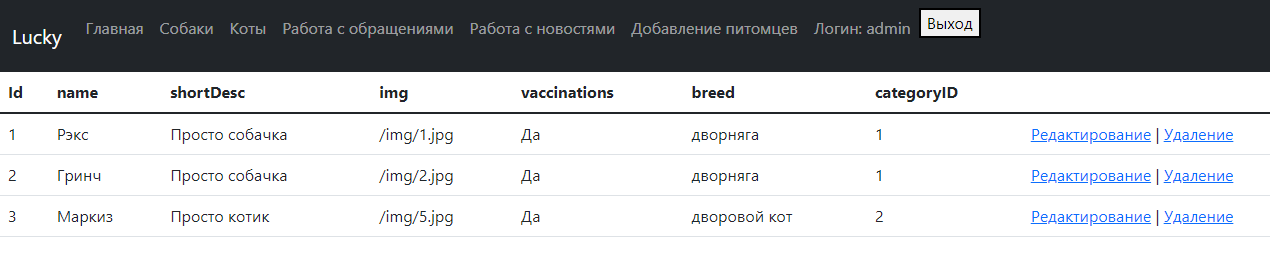


Рисунок 2.5 – Страница работы с питомцами

На рисунке 2.6 изображён интерфейс страницы добавления питомца, доступная администратору.

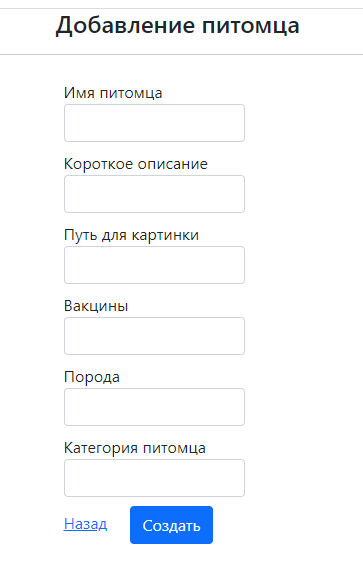


Рисунок 2.6 – Страница добавления питомца

На рисунке 2.7 изображён интерфейс страницы редактирования информации о питомце, доступная администратору.

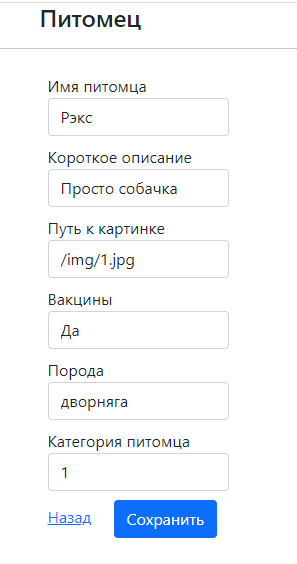


Рисунок 2.7 – Страница редактирования информации о питомце

На рисунке 2.8 изображён интерфейс страницы удаления информации о питомце, доступная администратору.

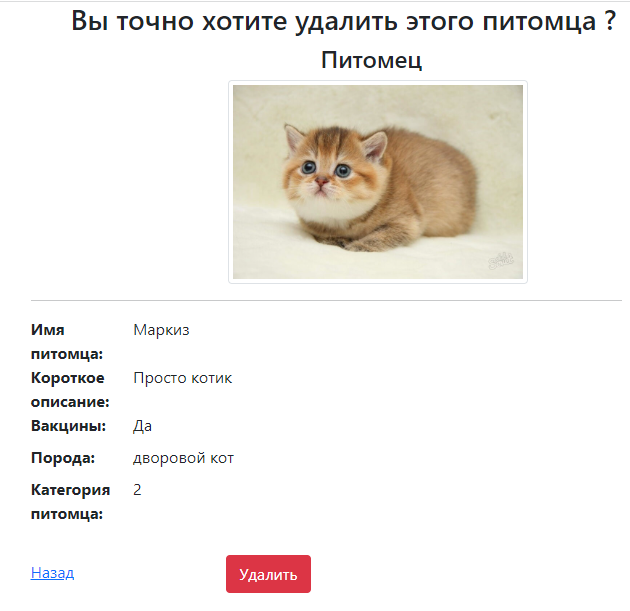


Рисунок 2.8 – Страница удаления информации о питомце

На рисунке 2.9 изображён интерфейс страницы с поиском питомца.

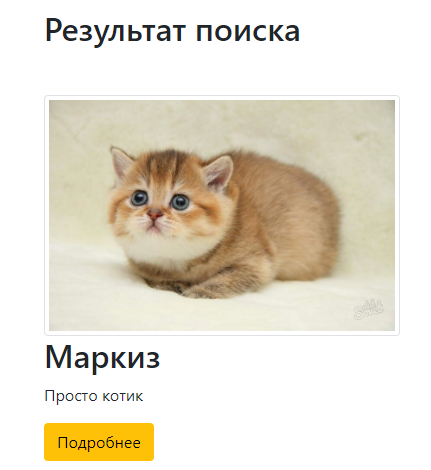


Рисунок 2.9 – Страница поиска питомца

Эргономическая оценка инженерных решений – это комплекс научно-технических и организационно-методических мероприятий по оценке выполнения в проектных документах и в образцах системы «человек — машина» (СЧМ) эргономических требований технического задания, нормативно-технических и руководящих документов, а также разработка рекомендаций для устранения отступлений от этих требований. Указанная оценка проводится при обосновании выполнения каждого этапа опытно-конструкторской разработки: технического предложения, эскизного проекта, рабочего проекта.

На практике эргономическая оценка представляет собой определение соответствия показателей объекта оценки эргономическим требованиям (ЭТ) и установление эргономического уровня качества оцениваемого объекта, т. е. степени реализации эргономических требований.

Эргономическая оценка производится на основании номенклатуры эргономических требований и показателей, отношения которых характеризуются определенной иерархической зависимостью.

Для оценки степени соответствия характеристик конкретной СЧМ эргономическим требованиям могут применяться экспериментальные, расчётные и экспертный методы.

Экспериментальный метод основан на использовании специальной аппаратуры и методик, позволяющих объективно оценить степень соответствия СЧМ эргономическим требованиям (например, по изменению показателей функционального состояния человека-оператора, по количеству допущенных ошибок и т.п.) Данный метод даёт надёжные результаты, но его реализация связана с трудностями, т.к. требует значительных затрат времени и средств, кроме того необходимо иметь опытный образец технического звена системы.

Расчетный метод позволяет определить значения эргономических показателей качества СЧМ в зависимости от ее параметров на основе использования теоретических или эмпирических зависимостей. На сегодняшний день теоретико-эмпирические зависимости между параметрами СЧМ и эргономическими показателями практически не разработаны, что существенно затрудняет применение данного метода.

Экспертный метод в настоящее время является наиболее распространенным. Его сущность заключается в проведении экспертами интуитивно-логического анализа с количественной оценкой суждений и обработкой результатов. При этом достоверность экспертизы зависит от количества экспертов и их квалификации.

Для проведения эргономической оценки пользовательского интерфейса проектируемой системы будем использовать экспертный метод.

Общие эргономические требования к проектируемой системе приведены в таблице 2.5.

Таблица 2.5 – Общие эргономические требования к проектируемой системе и соответствующие им единичные эргономические показатели

|  |  |  |
| --- | --- | --- |
| Группа | Эргономические требования | Единичные эргономические показатели |
| 1 | 2 | 3 |
| Психофизиологические | ПФ-1. Соответствие размеров знаков на экране дисплея оперативному порогу зрения человека | Размеры шрифта текста не более 30 пикселей |
| ПФ-2. Соответствие контраста знаков и фона оптимальным условиям восприятия | Использование темного цвета на белом фоне |
| ПФ-3. Отображение недоступных пунктов меню хорошо различимым блеклым цветом | Цвет недоступных пунктов меню выделить серым цветом |
| ПФ-4. Соответствие расположения надписей условиям их оптимального считывания | Удобное расположение на экране дисплея |
| ПФ-5. Использование пролистываемых и раскрывающихся списков в целях экономии экранного пространства | Наличие пролистываемых списков |
| Психологические | П-1. Наличие указаний на проблемы, возникающие в процессе обслуживания системы | Сообщения об ошибочных действиях пользователей |
| П-2. Наличие подсказок о следующих шагах работы в системе | Интуитивно понятный интерфейс |
| П-3. Наличие предупреждений о нежелательных последствиях некоторых действий | Предупреждение о последствиях |

Продолжение таблицы 2.5

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Психологические | П-4. Соответствие цветов знаков и надписей сформированным стереотипам восприятия цвета | Цвета ошибок выделить красным |
| П-5. Отсутствие в текстовых сообщениях аббревиатур, непонятных слов и сокращений | Отсутствие аббревиатур и слов, способных вызвать недопонимание. |
| П-6. Привлечение внимания пользователей к важным сообщениям | Выделение текста жирным |
| П-7. Наличие индикатора степени выполнения заданий (операций) | Наличие полоски показывающая результат выполнения |
| П-8. Наличие кратких и понятных заголовков окон | Краткие и понятные обозначения |
| П-9. Использование для названий пунктов меню одного слова (глагола для действий, существительного для объектов) | Использование пунктов из одного слова для обозначение действия и объекта |
| П-10. Применение в названиях пунктов меню норм использования заглавных букв, принятых в языке. | Использование заглавных букв в названии |
| П-11. Соответствие опций элементов интерфейса установленным, привычным нормам (например, использование клавиши Enter) | Соответствие привычным нормам |
| П-12. Отсутствие у пользователей сложностей в поиске необходимых директив (элементов интерфейса) для управления процессом решения поставленной задачи | Интуитивно понятный интерфейс |
| П-13. Сообщение об ошибке должно указывать, почему произошла ошибка | Сообщение указывает причину ошибки |
| П-14. Вежливое и понятное пользователю сообщение об ошибках | Сообщение об ошибке вежливое и корректное |

Продолжение таблицы 2.5

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Физиологические | Ф-1. Соответствие размеров зон установки курсора физиологическим возможностям движений | Размер кнопок не должен быть более 50 пикселей. Размер полей для ввода не должен быть более 200 пикселей. |
| Ф-2. Использование крутилок для ввода числовых значений | Наличие крутилок для ввода числовых значений |
| Ф-3. Отсутствие требований к пользователям вводить информацию, которая была предварительно введена или которая может быть автоматически получена из системы | При редактировании информация подгружается системой |
| Ф-4. Использование командных кнопок для ввода явных действий | Наличие командных кнопок для ввода явных действий |
| Гигиенические | Г-1. Соответствие параметров рабочей среды гигиеническим нормативам | Соответствие параметров рабочей среды безопасности человека для его здоровья |
| Социально-психологические | СП-1. Минимизация возможности возникновения ситуаций, ведущих к дискомфорту пользователей | Использовать разграничение ролей |

Далее проводим оценку значений единичных эргономических показателей. При этом рекомендуемые значения единичных эргономических показателей устанавливаются на основе действующих нормативно-технических документов и эргономических справочников.

Единичные эргономические показатели оцениваются по бинарной шкале, они принимают значение, равное 1, если фактическое значение показателя соответствует рекомендуемому, и равное 0, если оно ему не соответствует.

Групповой эргономический показатель рассчитывается как общая оценка по группе единичных показателей:

ЭПгр = ∑ 1 ∕ (∑ 1 + ∑ 0), (2.1)

где − суммарное число случаев, когда имеет место соответствие;

− суммарное число случаев, когда соответствия нет.

Результаты оценки значений единичных и групповых эргономических показателей приведены в таблице 2.6.

Таблица 2.6 – Значения единичных и групповых эргономических показателей проектируемой системы

|  |  |  |
| --- | --- | --- |
| Группа ЭП | Значения единичных ЭП | Значения групповых ЭП |
| Психофизиологические ЭП | ПФ-1, ПФ-2, ПФ-3, ПФ-4, ПФ-5 = 1 | 5 / (5 + 0) = 1 |
| Психологические ЭП | П-1, П-2, П-3, П-5, П-6, П-8, П-9, П-10, П-11, П-12, П-13,  П-14 = 1;  П-7, П-15 = 0 | 12 / (12+2) = 0,857 |
| Физиологические ЭП | Ф-1, Ф-2, Ф-3 = 1  Ф-4 = 0 | 3 / (3+1) = 0,75 |
| Гигиенические ЭП | Г-1 = 1 | 1 / (1+ 0) = 1 |
| Социально-психологические ЭП | СП-1 =1 | 1 / (1+ 0) = 1 |
| Антропометрические ЭП | Не актуальны для данной СЧМ |  |

Далее оцениваются эргономические свойства СЧМ. Однако поскольку для нашей системы значимым является только одно свойство – «управляемость» именно это свойство будет определять эргономичность системы в целом.

Эргономические свойства СЧМ определяются по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (2.2) |

где *α*н*i* – нормированные весовые коэффициенты, сумма которых должна быть равна единице.

Для оцениваемого эргономического свойства «управляемость» выбираем величины весовых коэффициентов. Результаты распределения представлены в таблице 2.7

Таблица 2.7 – Значения весовых коэффициентов для оценки эргономического свойства «управляемость»

|  |  |
| --- | --- |
| Групповой ЭП | Значение весового коэффициента |
| Психофизиологический | 0,25 |
| Психологический | 0,4 |
| Физиологический | 0,15 |
| Гигиенический | 0,1 |
| Социально-психологический | 0,1 |

По формуле (2.2), определим количественное значение указанных выше эргономических свойств:

ЭСВуправляемость = 0,251 + 0,40,857 + 0,150,75 + 0,11 + 0,11 = 0,905.

Поскольку в нашей системе значимым с точки зрения формирования интегральной оценки – эргономичности - является только одно эргономическое свойство – «управляемость» принимаем за оценку эргономичности полученное значение.

Следовательно, эргономичность нашей системы равна 0,905

Полученное значение группового эргономического показателя оценивается с учетом следующей градации:

* 0,8 – 1,0 – «отлично» – эргономические характеристики изделия соответствуют базовым значениям;
* 0,5 – 0,8 – «хорошо» – приближается к базовым, но требуется совершенствование изделия;
* 0,2 – 0,5 – «удовлетворительно» – далеки от базовых, требуется значительное улучшение изделия;
* 0 – 0,2 – «неудовлетворительно» – практически не обеспечивается необходимая производительность, удобство и безопасность труда человека – оператора.

Согласно расписанной выше системе оценивания, уровень эргономичности данной СЧКС оценивается отметкой «отлично», что свидетельствует о тщательном подходе в ее создании и детально проработанном взаимодействия пользователя с приложением. Однако так как показатель эргономичности не равен 1, нужно провести анализ единичных показателей, которые не соответствуют предъявляемым требованиям. Результаты анализа представлены в таблице 2.8.

Таблица 2.8 – Рекомендации по улучшению эргономичности, проектируемой СЧКС

|  |  |
| --- | --- |
| Невыполненное эргономическое требование | Предложение по улучшению эргономичности |
| П-7. Наличие индикатора степени выполнения заданий (операций) | Добавить полоску индикатор, демонстрирующую ход выполнения задания |
| П-15. Наличие подсказок о следующих шагах работы в системе | Добавить подсказки для пользователя о следующих шагах в информационной системе |
| Ф-4. Использование командных кнопок для ввода явных действий | Добавить командные кнопки, для ввода явных действий |

В разделе 2 была представлена информация об эргономической составляющей разрабатываемой информационной системы. Также в виде таблицы были отображены эргономические требования, показатели и условия к программному продукту. Также были представлены различные варианты взаимодействия пользователя и информационной системы, и их внешний вид.

# 3 РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ПРИЮТА ДЛЯ БЕЗДОМНЫХ ЖИВОТНЫХ

# 3.1 Разработка структуры информационной системы

Для грамотного построение структуры прежде всего нужно выбрать архитектуру информационной системы. Архитектура информационной системы.

Архитектура информационной системы – это набор ключевых решений, неизменных при изменении бизнес–технологии в рамках бизнес–видения [12].

Для реализации информационной системы была выбрана «клиент-серверная» архитектура. Данная архитектура позволяет разделять процессы предоставление услуг и отправки запросов на них на разные компьютеры (клиенты) находящийся в сети, каждый из которых выполняет свои задачи независимо от других пользователей. Схематически данную архитектуру можно представить, как показано на рисунке 3.1.

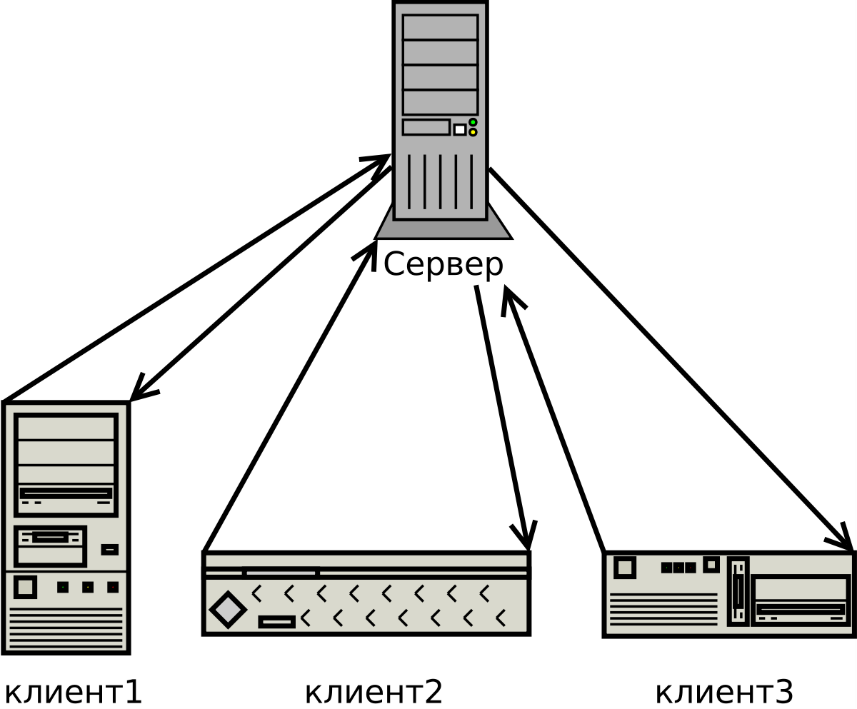


Рисунок 3.1 – Клиент-серверная архитектура

Следующим шагом к построению структуры информационной системы является выбор шаблона проектирования. Шаблоны проектирования – это многократно применяемая архитектурная конструкция, предоставляющая решение общей проблемы проектирования в рамках конкретного контекста и описывающая значимость этого решения. Паттерн не является законченным образцом проекта, который может быть прямо преобразован в код. Для данной информационной системы был выбран шаблон проектирования MVC (Model, View, Controller).

Концепция шаблона проектирования MVC предпологает разделения приложения на три компонента:

* Модель (model): описывает используемые в приложении данные, а также логику, которая связана непосредственно с данными, например, логику валидации данных. Как правило, объекты моделей хранятся в базе данных;
* Представление (view): отвечают за визуальную часть или пользовательский интерфейс, нередко html-страница, через который пользователь взаимодействует с приложением. Также представление может содержать логику, связанную с отображением данных. В то же время представление не должно содержать логику обработки запроса пользователя или управления данными [13];
* Контроллер (controller): представляет центральный компонент MVC, который обеспечивает связь между пользователем и приложением, представлением и хранилищем данных. Он содержит логику обработки запроса пользователя. Контроллер получает вводимые пользователем данные и обрабатывает их. И в зависимости от результатов обработки отправляет пользователю определенный вывод, например, в виде представления, наполненного данными модели [14].

В данном дипломном проекте имеются следующие файлы, отвечающие за структуру сайта и его функционал:

Администратор:

* «CRUD\_NEWSController.cs» – этот файл отвечает за работу контроллера новостей;
* «CRUD\_ORDERController.cs» – этот файл отвечает за работу контроллера обращений пользователей;
* «CRUD\_Controller.cs» – этот файл отвечает за работу контроллера с питомцами;
* «PetCartController.cs» – этот файл отвечает за работу контроллера с подробной информацией о питомце;
* «SearchController.cs» – этот файл отвечает за работу контроллера с поиском;
* «AccountController.cs» – этот файл отвечает за работу контроллера с авторизацией и регистрацией;
* «PetRepository.cs» – этот файл отвечает за доступ к источнику данных;
* «Layout.cshtml» – этот файл отвечает за отображение главной страницы;
* «dbsettings.json» – этот файл отвечает за настройки использующейся базы данных.

В данном программном средстве присутствуют следующие функции:

* «Search» – отвечает за поиск;
* «List» – отвечает за вывод информации о всех питомцах;
* «Create» – отвечает за добавление информации о питомце в базу данных;
* «Error» ­ – отвечает за вывод ошибки, в случае неправильного ввода данных;
* «Update» ­ – отвечает за обновление информации о питомце;
* «Delete» ­ – отвечает за удаление питомца из базы данных;
* «Login» ­ – отвечает за авторизацию пользователя;
* «Lagout» ­– отвечает за выход пользователя из системы;
* «Register» ­ – отвечает за регистрацию пользователя.

Физическая структура базы данных определяет тип и свойства данных, которые будут записаны в память компьютера. Правила перехода к физической модели следующие: каждое отношение превращается в файл базы данных, каждый столбец - в поле файла, каждая строка – в запись файла.

Этап физического моделирования базы данных включает в себя определение состава файлов и их заполнение исходными данными в соответствии с ограничениями, допущениями и особенностями предметной области.

Описание сущности «Pet», предназначенное для хранения данных о питомцах, представлено в таблице 3.1.

Таблица 3.1 – Описание отношения «Pet»

|  |  |  |
| --- | --- | --- |
| Название атрибута | Тип данных | Описание |
| Id | Счетчик | Идентификационный номер питомца |
| Name\_pet | Текстовой | Имя питомца |
| shortDecs | Текстовой | Краткое описание |
| img | Текстовой | Путь к изображению |
| vaccinations | Текстовой | Прививки |
| breed | Текстовой | Порода |
| categoryId | Числовой | Идентификационный номер категории |

Описание сущности «Category», предназначенное для хранения данных о категориях, представлено в таблице 3.2.

Таблица 3.2 – Описание отношения «Category»

|  |  |  |
| --- | --- | --- |
| Название атрибута | Тип данных | Описание |
| Id | Счетчик | Идентификационный номер категории |
| category\_name | Текстовой | Название категории |
| Decs | Текстовой | Описание категории |

Описание сущности «User», предназначенное для хранения данных о пользователях, представлено в таблице 3.3.

Таблица 3.3 – Описание отношения «User»

|  |  |  |
| --- | --- | --- |
| Название атрибута | Тип данных | Описание |
| Id | Счетчик | Идентификационный номер пользователя |
| Login | Текстовой | Никнейм пользователя |
| password | Числовой | Пароль пользователя |
| roleId | Числовой | Идентификационный номер роли пользователя |

Описание сущности «Order», предназначенное для хранения данных о заявках пользователей, представлено в таблице 3.4.

Таблица 3.4 – Описание отношения «Order»

|  |  |  |
| --- | --- | --- |
| Название атрибута | Тип данных | Описание |
| Id | Счетчик | Идентификационный номер заявки |
| dateOrd | Текстовой | Дата заказа |
| Email | Текстовой | Почта |
| petId | Числовой | Идентификационный номер питомца |

Описание сущности «News», предназначенное для хранения данных о новостях, представлено в таблице 3.5.

Таблица 3.5 – Описание отношения «News»

|  |  |  |
| --- | --- | --- |
| Название атрибута | Тип данных | Описание |
| Id | Счетчик | Идентификационный номер новости |
| shortDecs | Текстовой | Краткое описание |
| img | Текстовой | Путь к изображению |
| Desc | Текстовой | Полное описание |

Описание сущности «Role», предназначенное для хранения данных о ролях пользователей, представлено в таблице 3.6.

Таблица 3.6 – Описание отношения «Role»

|  |  |  |
| --- | --- | --- |
| Название атрибута | Тип данных | Описание |
| Id | Счетчик | Идентификационный номер роли |
| name | Текстовой | Название роли |

Описание связей базы данных:

* «Role» – «User». Связь «Один-ко-многим» (1:M). Одна роль закрепляется за одним или несколькими пользователями. Пользователь имеет одну роль;
* «Category» – «Pet». Связь «Один-ко-многим» (1:M). В одной категории может быть множество питомцев;
* «Pet» – «Order». Связь «Один-ко-многим» (1:M). На одного питомца может быть несколько заявок.

Для разработки базы данных используется MS SQL – компактная СУБД. Схема данных представлена в графических материалах.

# 3.2 Разработка алгоритмов работы информационной системы

Для разработки алгоритмов работы автоматизированной системы зооприюта, необходимо сформулировать общие требования к функциональному поведению проектируемой системы, выполнить исходную структуризацию предметной области [15].

Таким образом, рассмотрим диаграмму вариантов использования, представленную в графических материалах.

В разрабатываемой системе предусмотрено два типа пользователей, а именно пользователи и администраторы. Для входа необязательно авторизоваться в информационной системе. Для авторизации нужно ввести имя пользователя и пароль.

Если учетная запись не создана, можно зарегистрироваться. Страница регистрации должна содержать следующие элементы пользовательского интерфейса:

– поле для ввода Email;

– поле для ввода пароля;

– поле для подтверждения введенного пароля.

При авторизации как пользователь доступна главная страница с новостями, а также страницы с питомцами и подробным описанием питомца.

На главной странице отображаются следующие элементы пользовательского интерфейса:

– список новостей с картинкой и коротким описанием;

– поле для поиска питомца;

– кнопка «Подробнее» на новостях.

При нажатии на кнопку подробнее, будет совершен переход на страницу выбранной новости.

На странице с новостями отображаются следующие элементы пользовательского интерфейса:

– картинка новости;

– полное текст новости.

На странице с питомцами отображаются следующие элементы пользовательского интерфейса:

– список питомцев с картинкой и коротким описанием;

– поле для поиска питомца;

– кнопка «Подробнее».

На странице с подробной информацией о питомце отображаются следующие элементы пользовательского интерфейса:

– картинка питомца;

– короткое описание питомца;

– информация о прививках;

– информация о породе;

– кнопка «Взять себе».

Используя систему с ролью администратора, открывается другой экран, нежели с ролью пользователь. Администратору кроме стандартных, доступны также страницы:

– работа с новостями;

– работа с питомцами;

– работа с обращениями.

Каждая из этих страниц имеет одинаковую структуру. На них располагается, кнопка добавления, редактирования, удаления и список соответствующей информации. Администратор может создавать, редактировать и удалять питомцев, новости и заявки.

Рассмотрим процесс добавления, редактирования и удаления питомца. Для добавления питомца необходимо перейти на соответствующую вкладку и нажать на кнопку «Добавление питомца», затем заполнить форму и нажать на кнопку «Создать». Для редактирования нужно перейти на соответствующую вкладку, выбрать необходимого питомца и нажать на кнопку «Редактирование», далее внести изменения в появившихся полях и нажать на кнопку «Сохранить». Для удаления нужно перейти на соответствующую вкладку, выбрать необходимого питомца и нажать на кнопку «Удаление», далее в появившемся окне нажать на кнопку «Удалить».

Далее рассмотрим, страницы добавления каждой сущности:

1 Добавление питомца содержит:

– поле для ввода клички;

– поле для ввода короткого описания;

– поле для ввода пути картинки;

– поле для ввода данных о прививках;

– поле для ввода информации о породе;

– поле для ввода информации о категории;

– кнопку «Назад»;

– кнопку «Создать».

2 Добавление новости содержит:

– поле для ввода короткого описания;

– поле для ввода полного описания;

– поле для ввода пути картинки;

– кнопку «Назад»;

– кнопку «Создать».

3 Добавление заявки содержит:

– поле для ввода даты обращения;

– поле для ввода email;

– поле для ввода идентификационного номера питомца;

– кнопку «Назад»;

– кнопку «Создать».

Окна программы можно увидеть на плакатах в графическом материале дипломного проекта.

# 3.3 Разработка и тестирование программного модуля

Для оценки работоспособности информационной системы, разработанной в дипломном проекте, было проведено функциональное тестирование [16].

В рамках функционального тестирования информационная система была исследована на соответствие требованиям, сформулированным ранее в разделе 1.3. Для удобства проверки информационная система была разделена на модули и подмодули. Далее на основании требований был составлен чек-лист проверок. Также было проведено MAT тестировавние – тестирование системы или ее части только на валидных данных. Чек-лист и результат проверки представлен в таблице 3.7. Результаты проверок получены на основе сопоставления ожидаемых и полученных результатов [17].

Таблица 3.7 – Чек-лист проверок

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Модуль | Подмодуль | Проверка | Уровень проверки | Результат |
| 1 | 2 | 3 | 4 | 5 |
| Главная страница | Основная часть | Графический пользовательский интерфейс | Smoke | Проверка выполнена успешно |
| Главная страница | Основная часть | Содержание главной страницы | Smoke | Проверка выполнена успешно |
| Главная страница | Пункты меню | Отображение пунктов для администратора только для администратора | Smoke | Проверка выполнена успешно |
| Главная страница | Пункты меню | Отображение поля ввода для поиска | Smoke | Проверка выполнена успешно |

Продолжение таблицы 3.7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| Главная страница | Пункты меню | Поиск при помощи фильтра | Smoke | Проверка выполнена успешно |
| Главная страница | Пункты меню | Переход на страницу работы с питомцами | Smoke | Проверка выполнена успешно |
| Страница работы с питомцами | Основная часть | Графический пользовательский интерфейс | Smoke | Проверка выполнена успешно |
| Страница добавления питомца | Основная часть | Валидация полей при неправильном вводе | AT | Проверка выполнена успешно |
| Страница подробной информации о питомце | Основная часть | Нажатие на кнопку «Взять себе» | MAT | Проверка выполнена успешно |
| Страница добавления питомца | Основная часть | Добавление при правильном вводе | Smoke | Проверка выполнена успешно |
| Страница добавления питомца | Основная часть | Добавление при неправильном вводе | Smoke | Проверка выполнена успешно |
| Страница редактирования питомца | Основная часть | Редактирование при правильном вводе | Smoke | Проверка выполнена успешно |

Продолжение таблицы 3.7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| Страница редактирования питомца | Основная часть | Редактирование при неправильном вводе | Smoke | Проверка выполнена успешно |
| Страница удаления питомца | Основная часть | Удаление питомца | Smoke | Проверка выполнена успешно |
| Страница добавления новости | Основная часть | Добавление при правильном вводе | Smoke | Проверка выполнена успешно |
| Страница добавления новости | Основная часть | Добавление при неправильном вводе | Smoke | Проверка выполнена успешно |
| Страница редактирования новости | Основная часть | Редактирование при правильном вводе | Smoke | Проверка выполнена успешно |
| Страница редактирования новости | Основная часть | Редактирование при неправильном вводе | Smoke | Проверка выполнена успешно |
| Страница удаления новости | Основная часть | Удаление питомца | Smoke | Проверка выполнена успешно |
| Страница добавления обращения | Основная часть | Добавление при правильном вводе | Smoke | Проверка выполнена успешно |

Продолжение таблицы 3.7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| Страница добавления обращения | Основная часть | Добавление при неправильном вводе | Smoke | Проверка выполнена успешно |
| Страница редактирования обращения | Основная часть | Редактирование при правильном вводе | Smoke | Проверка выполнена успешно |
| Страница редактирования обращения | Основная часть | Редактирование при неправильном вводе | Smoke | Проверка выполнена успешно |
| Страница удаления обращения | Основная часть | Удаление питомца | Smoke | Проверка выполнена успешно |

Для базовых функций информационной системы было проведено Smoke-тестирование [18].

Smoke Test – поверхностное тестирование для определения пригодности сборки для дальнейшего тестирования, должно покрывать базовые функции программного обеспечения.

Для проверок на уровне Smoke-тестирования были разработаны низкоуровневые тест-кейсы. Тест-кейс представляет собой набор входных данных, условий выполнения и ожидаемых результатов, разработанный с целью проверки того или иного свойства. Низкоуровневым называется тест-кейс с конкретными входными данными и ожидаемыми результатами.

Общая структура тест-кейса включает:

* идентификатор;
* модуль и подмодуль приложения;
* заглавие тест-кейса;
* исходные данные, приготовления к тест-кейсу;
* шаги тест-кейса;
* ожидаемые результаты по каждому шагу тест-кейса.

Тест кейсы разделяются по ожидаемому результату на позитивные и негативные:

– позитивный тест-кейс использует только корректные данные и проверяет, что приложение правильно выполнило вызываемую функцию;

– негативный тест-кейс оперирует как корректными, так и некорректными данными (минимум 1 некорректный параметр) и ставит целью проверку исключительных ситуаций (срабатывание валидаторов), а также проверяет, что вызываемая приложением функция не выполняется при срабатывании валидатора.

Рассмотрим позитивные и негативные тест-кейсы для каждого сценария.

В таблице 3.8 представлено описание позитивных тест-кейсов.

Таблица 3.8 – Позитивные тест-кейсы

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | Приоритет | Модуль | Подмодуль | Заголовок | Шаги | Ожидаемые результаты |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | Smoke | Главная страница | Основная часть | Просмотр новостей | 1 Перейти на сайт. | 1.Осуществляется переход на главную страницу. |
| 2 | MAT | Страница подробной информации о питомце | Основная часть | Попытка оставить заявку будучи неавторизованным | 1 Перейти на сайт.  2. Перейти на вкладку с питомцами.  3.Кликнуть на кнопку «Подробнее»  4.Кликнуть на кнопку «Взять себе» | 1. Переход на страницу авторизации. |
| 3 | MAT | Главная страница | Основная часть | Попытка выйти из аккаунта | 1 Перейти на сайт.  2. Нажать на кнопку «Выйти» | 1. Переход на страницу авторизации. |

Продолжение таблицы 3.8

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 4 | MAT | Страница авторизации | Основная часть | Попытка авторизоваться | 1. Ввести данные в поле  2. Нажать на кнопку «Вход» | 1. Переход на страницу авторизации. |
| 5 | MAT | Страница поиска | Основная часть | Попытка ввода корректных данных | 1.Перейти на сайт.  2.Клик по полю ввода.  3. Ввод значений в поле ввода.  4.Нажатие на кнопку «Поиск» | Выдача результата поиска |
| 6 | MAT | Страница подробной информации о питомце | Основная часть | Попытка оставить заявку будучи авторизованным | 1 Перейти на сайт.  2. Перейти на вкладку с питомцами.  3.Кликнуть на кнопку «Подробнее»  4.Кликнуть на кнопку «Взять себе» | 1.Успешная попытка оставить заявку. 2.Автоматический переход на страницу с уведомлением о успешной заявке. |
| 7 |  | Страница просмотра информации о питомце | Основная часть | Удаление питомца | 1. Перейти на панель администратора.  2. Нажать кнопку «Удаление».  3. Нажать на кнопку «Удалить» | 1. Удаление информации о питомце из бд.  2. Переход на страницу просмотра информации. |

Продолжение таблицы 3.8

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | MAT | Страница подробной информации о питомце | Основная часть | Попытка оставить заявку будучи авторизованным | 1 Перейти на сайт.  2. Перейти на вкладку с питомцами.  3.Кликнуть на кнопку «Подробнее»  4.Кликнуть на кнопку «Взять себе» | 1. Переход на страницу с уведомлением об успешно оставленном обращении. |

Вышеизложенные позитивные тест-кейсы пройдены успешно.

Рассмотрим негативные тест-кейсы. В таблице 3.9 представлено подробное описание негативных тест-кейсов.

Таблица 3.9 – Негативные тест-кейсы

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | Приоритет | Модуль | Подмодуль | Заголовок | Шаги | Ожидаемые результаты |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | MAT | Страница регистрации | Основная часть | Попытка ввода другого пароля при подтверждении | 1.Перейти на сайт.  2. Перейти на страницу регистрации.  3. Ввод 2 разных паролей.  4.Нажатие на кнопку «Зарегистрироваться» | 1.Вывод ошибки. |

Продолжение таблицы 3.9

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 2 | MAT | Страница добавления информации о питомце | Основная часть | Попытка добавления питомца в несуществующую категорию | 1.Перейти на сайт будучи администратором.  2. Перейти на страницу панель администратора.  3. Переход на страницу добавления питомцев.  4. Заполнение некорректными данными  5. Нажатие кнопки «Создать» | 1.Вывод ошибки. |
| 3 | MAT | Страница авторизации | Основная часть | Попытка авторизации без ввода логина | 1.Перейти на сайт.  2. Перейти на страницу авторизации.  3. Заполнение некорректными данными  4. Нажатие кнопки «Создать» | 1.Вывод ошибки. |

В ходе тестирования пункта 1 была обнаружена и устранена ошибка. Все остальные тест-кейсы пройдены успешно.

Визуальное отображение вывода сообщения о соответствующих ошибках представлено на рисунках 3.2 – 3.4.

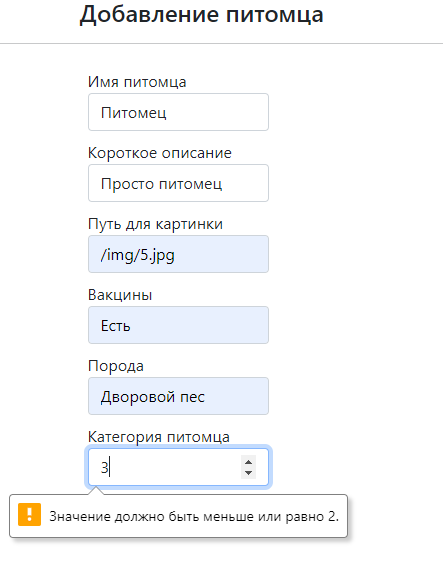


Рисунок 3.2 – Отображение ошибки при добавлении

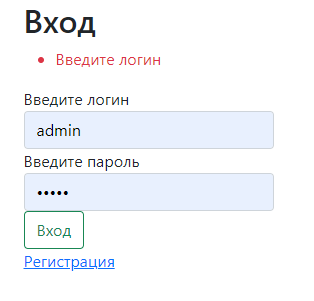


Рисунок 3.3 – Отображение ошибки при авторизации

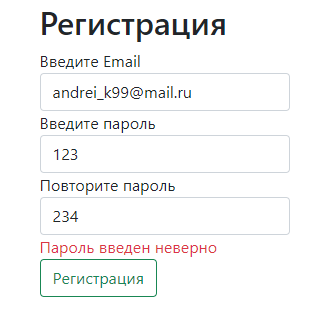


Рисунок 3.4 – Отображение ошибки при регистрации

В главе 3 были разработаны структура и алгоритмы работы системы, а также проведено тестирование работоспособности системы. По результатам тестирования было выявлено, что функционал системы работает корректно и без сбоев.

# 4 ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ЭФФЕКТИВНОСТИ РАЗРАБОТКИ И ИСПОЛЬЗОВАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПРИЮТА ДЛЯ БЕЗДОМНЫХ ЖИВОТНЫХ

# Краткое описание разработки

Разработанная в дипломном проекте информационная система представляет сайт зооприюта на платформе ASP.NET Core и ее эргономическое обеспечение. Данная система представляет собой клиент-серверное веб-приложение, которое предоставляет возможность получить информацию о питомцах зооприюта. Система будет использоваться зооприютом. Оно позволит публиковать новости приюта и размещать информацию о питомцах в сети интернет, что облегчит поиск новых хозяев для питомцев.

Преимуществом перед аналогами является то, что данная информационная система была выполнена с использование современных технологий, такими как:

* + ASP.NET Core – свободно-распространяемый кроссплатформенный фреймворк для создания веб-приложений на платформе .NET с открытым исходным кодом;
  + SQL Server – это платформа является одной из наиболее популярных систем управления базами данных (СУБД) в мире. Данная СУБД подходит для самых различных проектов: от небольших приложений до больших высоконагруженных проектов;
* Entity Framework представляет собой объектно-ориентированную, легковесную и расширяемую технологию от компании Microsoft для доступа к данным. EF Core является ORM-инструментом (object-relational mapping – отображения данных на реальные объекты).

Данные технологии позволяют системе выдерживать высокую нагрузку на сервер, обрабатывать большой объём данных.

# Расчёт затрат на разработку программного средства

Формирование отпускной цены программного средства осуществляется на основе расчета затрат на его разработку. В разработке программного средства принимают участие 4 исполнителя: руководитель проекта, разработчик тестов, инженер-программист, специалист по тестированию программного обеспечения, дизайнер.

Основная заработная плата исполнителей проекта определяется по формуле 4.1:

где n – количество исполнителей, занятых разработкой конкретного ПО;

Кпр – коэффициент, учитывающий процент премий (Кпр= 1,2);

Зчi – часовая заработная плата i-го исполнителя, р.;

ti – трудоёмкость работ, выполняемых i-м исполнителем, ч.

Расчёт основой заработной платы представлена в таблице 4.1.

Таблица 4.1 – Расчёт основной заработной платы разработчиков

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование должности разработчика | Вид выполняемой работы | Месячная заработная плата, р. | Часовая  заработная плата, р. | Трудоём-кость работ, ч | Итого, руб. |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Руководитель проекта | Контроль разработки | 4200 | 25 | 50 | 1250 |
| Инженер-программист | Создание бизнес-логики, базы данных | 3360 | 20 | 70 | 1400 |

Продолжение таблицы 4.1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Специалист по тестированию программного обеспечения | Тестирование системы | 2520 | 15 | 55 | 825 |
| Дизайнер | Создание дизайна системы | 3024 | 18 | 60 | 1080 |
| Всего | | | | | 4555 |
| Премия (20%) | | | | | 911 |
| Всего, затраты на основную заработную плату | | | | | 5466 |

После подстановки значений в формулу (4.1) основная заработная плата составит:

Зо = 1,2 × (1250 + 1400 + 825 + 1080) = 5466 р.

Затраты на дополнительную заработную плату разработчиков включают выплаты, предусмотренные законодательством о труде (оплата трудовых отпусков, льготных часов, времени выполнения государственных обязанностей и других выплат, не связанных с основной деятельностью исполнителей), и определяется по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (4.2) |

где Нд – норматив дополнительной заработной платы(10%);

Зо – затраты на основную заработную плату, р.

После подстановки значений в формулу (4.2) дополнительная заработная плата составит:

Отчисления на социальные нужды определяются в соответствии с действующими законодательными актами по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (4.3) |

где Нсоц – норматив отчислений от фонда оплаты труда, 34,6% (согласно действующему законодательству).

Подставив значения в формулу (4.3), получим:

Прочие затраты включают, как напрямую связанные с разработкой конкретного программного продукта в соответствии с планируемой суммой затрат на эти мероприятия, так и затраты, связанные с функционированием в организации разработчика в целом. Данные затраты включаются в себестоимость разработки программного обеспечения в процентах от затрат на основную заработную плату разработчиков по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (4.4) |
|  |  |  |

где Нпр – норматив прочих затрат (100%).

После подстановки значений в формулу (4.4) прочие расходы составят:

Общая сумма затрат по всем статьям сметы (Зр) на разработку ПО рассчитывается по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (4.5) |
|  |  |  |

По формуле (4.5) получаем сумму затрат по всем статьям сметы:

Зр = + + += .

# Экономический эффект при разработке ПО для свободной реализации на рынке информационных технологий

Для расчета экономического эффекта при разработке программного обеспечения потребуется найти прибыль (чистую прибыль) от продажи потребителям, которая определяется по формулам 4.6 – 4.8.

Цена программного обеспечения была определена в процессе переговоров с потенциальными заказчиками и установлена в размере 3750 рублей. Расчёт среднегодового объёма продаж произведён, основываясь на количестве компаний, которые согласны приобрести данное программное обеспечение (3), и компаний, заинтересованных в покупке, но на данный момент не готовых принять решение (2) Исходя из вышесказанного среднегодовой объём продаж составляет 5 реализуемых копий.

|  |  |  |
| --- | --- | --- |
|  |  | (4.6) |
|  |  |  |

где Ц – цена программного обеспечения, (3750 р.);

N – предполагаемый объем продаж (5);

НДС – налог на добавленную стоимость (формула 4.7);

Зр – затраты на разработку.

Прибыль разработчика по формуле 4.6 составит:

Налог на добавленную стоимость определяется по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (4.7) |
|  |  |  |

где НДС – ставка налога на добавленную стоимость согласно действующему законодательству, (Ндс = 20%).

Налог на добавленную стоимость по формуле 4.7 составит:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Чистая прибыль разработчика определяется по формуле 4.8:

|  |  |  |
| --- | --- | --- |
|  |  | (4.8) |
|  |  |  |

где ‒ ставка налога на прибыль, (18%).

Чистая прибыль разработчика по формуле 4.8 составит:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | |  |
|  | |  |  |

Рентабельность затрат на разработку программного средства определяется по формуле 4.9:

где ‒ прирост чистой прибыли разработчика, полученной от разработки программного средства, р;

‒ затраты на разработку программного средства, р.

Уровень рентабельности составит:

В результате технико-экономического обоснования разработки и использования веб-ресурса были получены следующие данные:

* + сумму затрат по всем статьям сметы составила рублей;
* чистая прибыль разработчика составила ;
  + рентабельность затрат на разработку программного средства составила 12,49%

Таким образом, использования информационной системы сайта зооприюта является экономически эффективным.

В главе 4 были рассчитаны затраты на разработку информационной системы, а также целесообразность ее использования.

# 5 ОХРАНА ТРУДА. РЕАЛИЗАЦИЯ ПРОСТРАНСТВЕННО-АНТРОПОМЕТРИЧЕСКОЙ ЭРГОНОМИЧЕСКОЙ СОВМЕСТИМОСТИ РАБОТНИКА И ТЕХНИЧЕСКОГО СРЕДСТВА ПРИ ОРГАНИЗАЦИИ РАБОЧЕГО МЕСТА

Цель данного дипломного проекта – проектирование и разработка информационной системы приюта для бездомных животных. В качестве основного сценария применения системы рассматривается использование в сети интернет для получения оперативной сводки и консолидации данных, обслуживания запросов пользователей. Автоматизированная информационная система (АИС) позволяет повысить качество и количество получаемой информации по торговой точке, а также анализировать эти данные с помощью отчетов и сводок и более эффективно использовать компьютерные ресурсы предприятия. Программное средство написано на языке C#. Для его работы требуется комплекс ПК на платформе x86 или x64 с установленной на каждом ОС Windows версии XP, Vista, Seven или новее.

Использование АИС на предприятии предполагает работу оператора ПК, обслуживающего вычислительную систему, за компьютером. Такая работа требует использования видеодисплейных терминалов (ВДТ), клавиатур и механических манипуляторов «мышь». Для повышения производительности труда и предотвращения переутомления оператора при работе с этими устройствами необходимо создать оптимальные условия для деятельности работника в системе «человек-машина». Изучением взаимодействия человека, машины и окружающей среды, разработкой подходов к оптимальному планированию человеко-машинных систем занимается эргономика [19].

Для эффективного и безопасного функционирования таких систем необходимо, прежде всего, обеспечить совместимость характеристик машины и человека. Совместимость человека с машиной определяется пространственно-антропометрической, сенсомоторной, энергетической (биомеханической) и психофизиологической совместимостью [20].

Сенсомоторная совместимость предполагает учет скорости операций человека и его реакций на различные раздражители. Энергетическая совместимость предусматривает согласование усилий, мощности и скорости движений органов управления с биомеханическими возможностями человека. Психофизиологическая совместимость должна учитывать реакцию человека на цвет, форму и другие эстетические параметры машины.

Пространственно-антропометрическая совместимость предполагает необходимость учета размеров тела человека, его возможностей обзора внешнего пространства, определения зоны досягаемости для конечностей, позы при организации рабочего места [21].

Для комфортной работы оператора необходимо обеспечить соответствие параметров рабочего места его антропометрических характеристикам. Антропометрические характеристики человека подразделяются на статические и динамические. К статическим характеристикам относятся размеры тела и его отдельных частей – рук, ног, кистей, стоп и др., к динамическим – возможные углы поворота отдельных частей тела, зоны досягаемости [22].

Задачей работника при работе с АИС заключается в анализе получаемых отчетов и сводок, и мониторинге работы АИС. Такая деятельность требует длительной работы за ПК, которая оказывает негативное влияние на здоровье, выражающееся в повышенном зрительном напряжении, психологической перегрузке, длительном неизменном положении тела (гипокинезия), а также воздействием некоторых физических вредных факторов (электромагнитное излучение, статическое электричество, ультрафиолетовое и рентгеновское излучения) [23].

Гипокинезия может являться условием формирования сердечно-сосудистых нарушений у человека. Все эти факторы негативно сказываются на продуктивности работы оператора ПК, приводят к переутомлению, а при длительном воздействии могут привести к серьёзным заболеваниям.

Для снижения влияния вредных факторов при работе за ПК, обеспечения пространственно-антропометрической совместимости работника и технического средства рабочее место оператора должно отвечать требованиям, учитывающим его антропометрические характеристики. Одним из важнейших требований является оптимальное расположение органов управления в пространстве рабочего места, учитывающее зоны досягаемости человека.

Пространство рабочего места с размещенными средствами отображения информации, органами управления и другими техническими средствами, в котором осуществляются двигательные действия человека в процессе труда, называется моторным полем. В моторном поле рабочего места оператора выделяют три зоны: зону досягаемости, легкой досягаемости и оптимальную зону. Зона досягаемости ограничивается дугами, описываемыми максимально вытянутыми руками при движении их в плечевом суставе, зона легкой досягаемости – расслабленными руками при движении их в плечевом суставе, оптимальная зона моторного поля – предплечьями при движении в локтевых суставах с опорой. При организации рабочего места необходимо стремиться к обеспечению выполнения всех трудовых операций в зоне досягаемости моторного поля. Если операции совершаются часто (две и более в 1 мин), они должны концентрироваться в зоне легкой досягаемости или оптимальной зоне моторного поля.

Зонирование рабочего место представлено на рисунке 5.1.

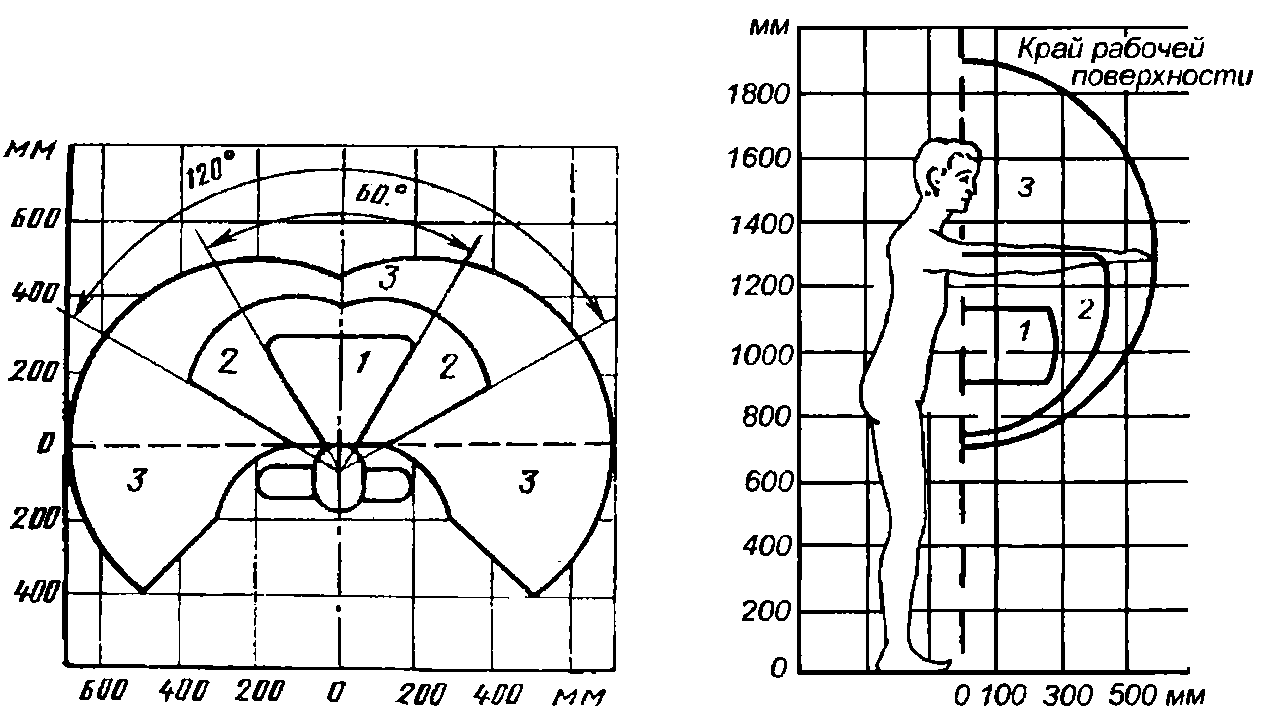


Рисунок 5.1 – Зонирование рабочего места

Рабочие зоны, представленные на рисунке:

1 Зона для размещения наиболее важных и очень часто используемых органов управления (оптимальная зона моторного поля).

2 Зона для размещения часто используемых органов управления (зона лёгкой досягаемости моторного поля).

3 Зона для размещения редко используемых органов управления (зона досягаемости моторного поля) [24].

С учётом зонирования моторного поля к рабочему месту оператора предъявляются следующие конкретные требования. Пространство для нахождения человека-оператора должно составлять по глубине не менее 850 мм. Должно иметься пространство для стоп глубиной и высотой не менее 150 мм и шириной не менее 530 мм. Все органы ручного управления должно находиться в зоне лёгкой досягаемости моторного поля: по высоте – 900-1300 мм, по глубине – 400-500 мм [25].

Схема размещения рабочих мест должна учитывать расстояния между рабочими столами с видеомониторами (в направлении тыла поверхности одного видеомонитора и экрана другого видеомонитора), которое должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов - не менее 1,2 м.

Планировка рабочего места оператора, отвечающая предъявленным требованиям, показана на рисунке 5.2.

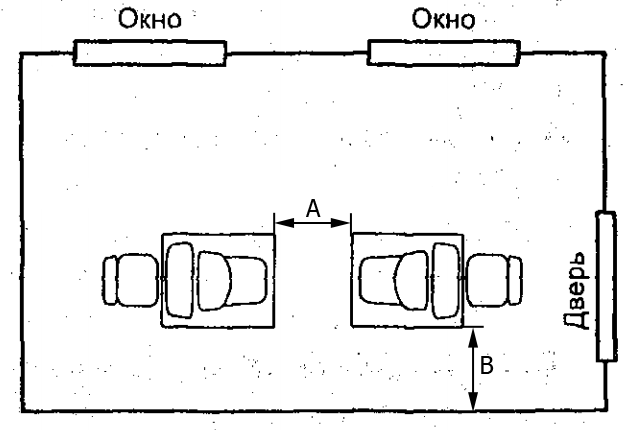


Рисунок 5.2 – Рекомендуемое расположение рабочих мест.

Расстояние А ­­­­ не менее 2 м. Расстояние В не менее 1,2 м.

Рабочее место оператора ПК включает в себя следующие основные элементы: рабочий стол, стул, ПК, ВДТ, клавиатуру и манипулятор «мышь». Каждое из этих устройств должно обеспечивать пространственно-антропометрическую эргономическую совместимость с оператором.

Экран ВДТ должен находиться в месте рабочей зоны, обеспечивающем удобство зрительного наблюдения в вертикальной плоскости под углом ± 30° от нормальной линии взгляда оператора, и иметь возможность поворота вокруг горизонтальной и вертикальной осей. Экран видеомонитора должен находиться от глаз пользователя на оптимальном расстоянии 600-700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов.

Высота рабочей поверхности стола должна регулироваться в пределах 680-800 мм, при отсутствии такой возможности высота должна составлять 725 мм. Рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной – не менее 500 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм. Конструкция рабочего стула (кресла) должна позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления [26].

Большое значение для обеспечения комфорта пользователя имеет рабочий стул (кресло). К нему предъявляется ряд требований. Поверхность сиденья должна иметь ширину и глубину не менее 400 мм, регулировку по высоте в пределах 400-550 мм и по углам наклона до 15° и назад до 5°. Спинка сиденья должна иметь ширину не менее 380 мм и высоту опорной поверхности 300 ± 20 мм. Угол наклона спинки должен иметь возможность регулировки в пределах 0 ± 30°. Кресло должно иметь подлокотники длиной не менее 250 мм и шириной 50-70 мм, с регулировкой по высоте в пределах 230 ± 30 мм и внутреннего расстояния между подлокотниками в пределах 350-500 мм.

Клавиатура должна располагаться на расстоянии не менее чем 300 мм от края стола, или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

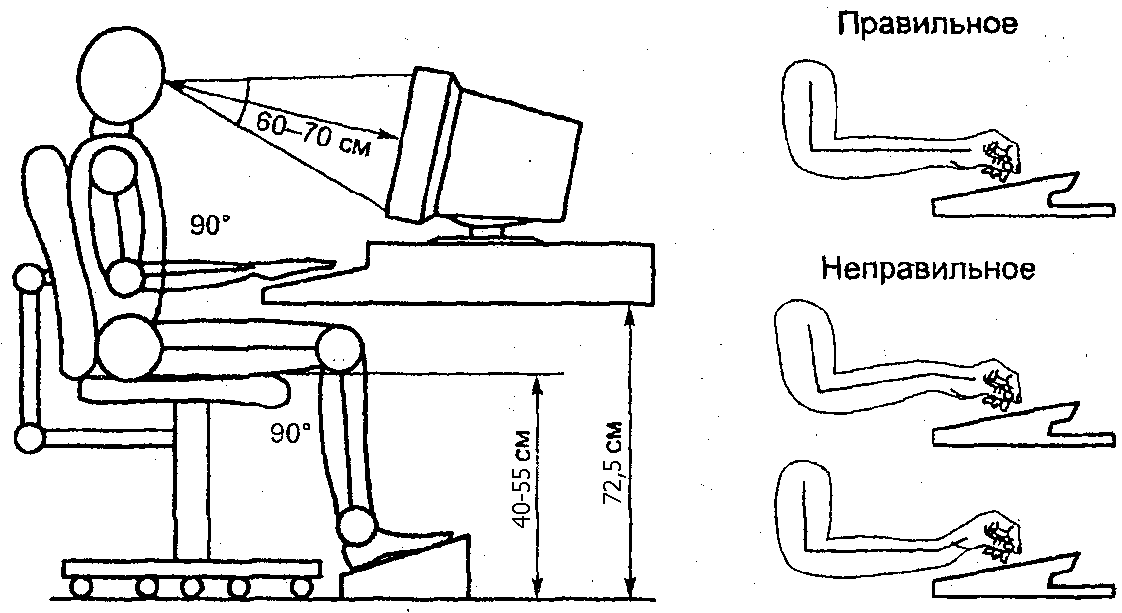


Рисунок 5.3 – Правильная позиция оператора и положение его рук при работе.

Также рабочее место должно быть оборудовано подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах до 150 мм и по углу наклона опорной поверхности подставки до 20 градусов. Поверхность подставки должна быть рифленой и иметь по переднему краю бортик высотой 10 мм [27].

Меры по охране труда, перечисленные выше, обеспечивают оптимальную пространственно-антропометрическую эргономическую совместимость работника и технического средства при организации рабочего место оператора ПК. При соблюдении требований пространственно-антропометрической эргономической совместимости все необходимые органы управления ПК находятся в зонах досягаемости и легко доступны для оператора, а расположение кресла, стола и подставки для ног учитывает антропометрические характеристики работника и предотвращает ухудшение осанки. Все эти факторы вместе создают комфортные условия для работы. В результате снижается утомляемость работника, увеличивается производительность труда и снижается риск ухудшения его здоровья при работе с АИС.

# ЗАКЛЮЧЕНИЕ

В процессе выполнения дипломного проекта была создана информационная система приюта для бездомных животных.

Разработанное приложение позволит пользователям получать информацию о новостях приюта, а также о питомцах приюта.

Реализация продукта осуществлялась в среде разработки Visual Studio с использованием следующего стека технологий C#, Entity Framework, SQL Server, ASP.NET Core.

В процессе проектирования были решены следующие задачи:

* проведен анализ существующих аналогов информационной системы зооприюта;
* выполнено эргономическое проектирование, в ходе которого были проведены анализ и распределение функций, разработаны алгоритмы работы пользователя и сценарий взаимодействия пользователя и информационной системы;
* разработаны графический интерфейс пользователя с учетом принципов инженерно-психологического проектирования;
* в части программного проектирования выбраны и обоснованы язык программирования и среда разработки, разработана архитектура информационной системы;
* проведено технико-экономическое обоснование эффективности разрабатываемого программного продукта. Расчёты показали, что рентабельность затрат на разработку составит 12,49%, следовательно, разработка данного программного средства является экономически эффективной и ее целесообразно осуществлять;
* в разделе по охране труда разработаны рекомендации по реализации пространственно-антропометрической эргономической совместимости работника и технического средства при организации рабочего места.

В результате программной реализации информационная система выполняет следующие функции:

– регистрация пользователя;

– авторизация пользователя;

– просмотра всех питомцев;

– просмотра всех новостей;

– поиск питомцев по имени;

– составление заявки о желании взять питомца;

– добавления питомца;

– удаление питомца;

– редактирование информации о питомце;

– добавления новости;

– удаление новости;

– редактирование новости;

– удаление заявки;

– редактирование заявки.

Преимуществами разработанного продукта является наличие системы поиска и возможности оставления заявки онлайн. Таким образом, в ходе дипломного проектирования были выполнены все поставленные цели и задачи.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Информационные технологии: теория, опыт, проблемы [Электронный ресурс]. – Режим доступ: https://baa.by/informacionnie-tehnologii.

[2] HTML и CSS. Разработка и дизайн веб-сайтов: Дакетт Джон – Вильямс, 2022. – 480 c.

[3] egida.by – [Электронный ресурс]. – Режим доступ: https://egida.by/

[4] zooshans.by – [Электронный ресурс]. – Режим доступ: https://zooshans.by/

[5] Фреймворки в веб-разработке [Электронный ресурс]. – Режим доступ: https://web-creator.ru/articles/about\_frameworks

[6] Введение в ASP.NET Core - Metanit [Электронный ресурс]. – Режим доступ: https://metanit.com/sharp/aspnet5/1.1.php

[7] Руководство по ASP.NET MVC 5 - Metanit [Электронный ресурс]. – Режим доступ: https://metanit.com/sharp/mvc5/

[8] Введение в Entity Framework Core - Metanit [Электронный ресурс]. – Режим доступ: https://metanit.com/sharp/entityframeworkcore/1.1.php

[9] Что такое SQL? - Информационные порталы и сервисы [Электронный ресурс]. – Режим доступ: https://aws.amazon.com/ru/what-is/sql/

[10] Microsoft SQL Server – Информационные порталы и сервисы [Электронный ресурс]. – Режим доступ: https://ipos.by/server/

[11] Шупейко И. Г. Эргономическое проектирование систем «человек-машина»: пособие / И. Г. Шупейко. – Минск: БГУИР, 2017. - 76 с.

[12] Справочник по HTML [Электронный ресурс]. – Режим доступ: http://htmlbook.ru/html

[13] Структуры данных и алгоритмы: Ульман Джеффри Д. / Ахо Альфред В. – Вильямс, 2019. – 400 c.

[14] Архитектура информационной системы, оценка рисков [Электронный ресурс]. – Режим доступ: https://www.osp.ru/cio/2002/06/172179

[15] Общие сведения ASP.NET Core MVC - Microsoft Learn [Электронный ресурс]. – Режим доступ: https://learn.microsoft.com/ru-ru/aspnet/core/mvc/views/overview?view=aspnetcore-7.0

[16] Хроники детерминированности | (об IT) [Электронный ресурс]. – Режим доступ: http://svyatoslav.biz/software\_testing\_book/

[17] Тестирование - Тестовый случай - Test Case - Про Тестинг [Электронный ресурс]. – Режим доступ: http://www.protesting.ru/testing/testcase.html

[18] Smoke-тестирование [Электронный ресурс]. – Режим доступ: https://blog.skillfactory.ru/glossary/smoke-test/

[19] Андруш, В. Г. Охрана труда: учебник / В. Г. Андруш, П. Т. Ткачёва, К. Д. Яшин. – 2-е изд., исправленное и дополненное. – Минск: РИПО, 2021. – 334 с.

[20] Вершина, Г. А. Охрана труда: учебник / Г. А. Вершина, А. М. Лазаренков. – Изд. 2–е, дополненное и переработанное. – Минск : ИВЦ Минфина, 2020. – 562 с.

[21] Пивоварчик, А. А. Охрана труда: учебно-методическое пособие / А. А. Пивоварчик. – Гродно: ГрГУ, 2021. – 431 с.

[22] Попов, Ю. П. Охрана труда: учебное / Ю. П. Попов, В. В. Колтунов. – 6-е изд., исправленное и дополненное. – Москва: КноРус, 2020. – 226 с.

[23] Пособие по охране труда в вопросах и ответах / [сост. В.К. Янковский]. – Минск: Центр охраны труда и промышленной безопасности, 2016. – 279 с.

[24] Челноков, А. А. Охрана труда: учебник / А. А. Челноков, И. Н. Жмыхов, В. Н. Цап. – Минск: Вышэйшая школа, 2020. – 542 с.

[25] Янковский, В.К. Пособие по охране труда в вопросах и ответах / составитель В. К. Янковский. – Минск: Центр охраны труда и промышленной безопасности, 2017. – 278 с.

[26] Беляков, Г.И. Безопасность жизнедеятельности. Охрана труда: учебник / Г. И. Беляков. – Москва: Юрайт, 2013. – 572 с.

[27] Графкина, М.В. Охрана труда в непроизводственной сфере: учебное пособие / М. В. Графкина. – Москва: Форум, 2013. – 319 с.

# ПРИЛОЖЕНИЕ А

# (обязательное)

# Листинг программы

using Microsoft.AspNetCore.Mvc;

using Lucky.Date;

using Lucky.Date.interfaces;

using Lucky.Date.Models;

using Lucky.ViewModels;

using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Authorization;

namespace Lucky.controllers

{

//[Authorize(Roles = "user, admin")]

public class PetsController : Controller

{

private readonly IAllPets \_allPets;

private readonly IPetsCategory \_allCategories;

public PetsController(IAllPets iAllPets, IPetsCategory iPetCat)

{

\_allPets = iAllPets;

\_allCategories = iPetCat;

}

[Route("Pets/List")]

[Route("Pets/List/{category}")]

public ViewResult List(string category)

{

string \_category = category;

IEnumerable<Pet> pets = new List<Pet>();

string currCategory = "";

if (string.IsNullOrEmpty(category))

{

pets = \_allPets.Pets.OrderBy(i => i.Id);

}

else

{

if (string.Equals("category1", category, StringComparison.OrdinalIgnoreCase))

{

Console.WriteLine(category);

pets = \_allPets.Pets.Where(i => i.Category.category\_name.Equals("Dogs")).OrderBy(i => i.Id);

currCategory = \_category;

Console.WriteLine(category);

}

else if (string.Equals("category2", category, StringComparison.OrdinalIgnoreCase))

{

pets = \_allPets.Pets.Where(i => i.Category.category\_name.Equals("Cats")).OrderBy(i => i.Id);

currCategory = \_category;

}

}

var carObj = new PetsListViewModel

{

allPets = pets,

currCategory = currCategory

};

ViewBag.Title = "Pets page";

return View(carObj);

}

}

}

using Microsoft.AspNetCore.Mvc;

using Lucky.Date;

using Lucky.Date.Models;

using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Authorization;

namespace Lucky.controllers

{

[Authorize(Roles = "admin")]

public class CRUDController : Controller

{

private readonly AppDbContent appDBContent;

public CRUDController(AppDbContent appDBContnent)

{

this.appDBContent = appDBContnent;

}

public ActionResult create()

{

return View();

}

[HttpPost]

public ActionResult create(Pet model)

{

using (appDBContent)

{

Console.WriteLine(model);

if ( model.name != null && model.vaccinations != null && model.breed != null && model.shortDesc != null && model.img != null && model.categoryID != 0)

{

appDBContent.Pet.Add(model);

appDBContent.SaveChanges();

Console.WriteLine(model);

return RedirectToAction("Read");

}

else

if( model.name==null)

ModelState.AddModelError(""," Введите имя");

if (model.vaccinations == null)

ModelState.AddModelError("", " Есть ли прививки");

if (model.breed == null)

ModelState.AddModelError("", " Введите породу");

if (model.shortDesc == null)

ModelState.AddModelError("", " Введите описание");

if (model.img == null)

ModelState.AddModelError("", " Введите путь для картинки");

if (model.categoryID == 0)

ModelState.AddModelError("", " Введите категорию");

}

return View();

}

public IActionResult Error()

{

ViewBag.Message = "Данные заполнены неверно";

return View();

}

[HttpGet]

public ActionResult Read()

{

using (appDBContent)

{

var data = appDBContent.Pet.ToList();

return View(data);

}

}

[HttpGet]

public ActionResult Update(Pet pet)

{

using (appDBContent)

{

var data = appDBContent.Pet.Where(x => x.Id == pet.Id).SingleOrDefault();

return View(data);

}

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Update(int CourseId, Pet model)

{

using (appDBContent)

{

var data = appDBContent.Pet.FirstOrDefault(x => x.Id == model.Id);

if (data != null && model.name != null && model.vaccinations != null && model.breed != null && model.shortDesc != null && model.img != null && model.categoryID != 0)

{

data.name = model.name;

data.shortDesc = model.shortDesc;

data.img = model.img;

data.vaccinations = model.vaccinations;

data.breed = model.breed;

data.categoryID = model.categoryID;

appDBContent.SaveChanges();

return RedirectToAction("Read");

}

else

{

return RedirectToAction("Error");

}

}

}

public ActionResult Delete(int id)

{

var data = appDBContent.Pet.Where(x => x.Id == id).SingleOrDefault();

return View(data);

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult

Delete(int id, Pet pet)

{

using (appDBContent)

{

var data = appDBContent.Pet.FirstOrDefault(x => x.Id == pet.Id);

if (data != null)

{

appDBContent.Pet.Remove(data);

appDBContent.SaveChanges();

return RedirectToAction("Read");

}

else

return View();

}

}

}

}

using Microsoft.EntityFrameworkCore;

using Lucky.Date.Models;

using Lucky.Date.Models.Account;

namespace Lucky.Date

{

public class AppDbContent : DbContext

{

public AppDbContent(DbContextOptions<AppDbContent> options) : base(options)

{

}

public DbSet<Pet> Pet { get; set; }

public DbSet<Category> Category { get; set; }

public DbSet<Order> Order { get; set; }

public DbSet<User> User { get; set; }

public DbSet<Role> Role { get; set; }

public DbSet<News> News { get; set; }

}

}

@model IEnumerable<Lucky.Date.Models.Pet>

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width" />

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

<title>Read</title>

</head>

<body>

<header>

<nav class="navbar navbar-expand-md navbar-dark bg-dark">

<div class="container-fluid">

<a class="navbar-brand">Lucky</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarCollapse" aria-controls="navbarCollapse" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarCollapse">

<ul class="navbar-nav me-auto mb-2 mb-md-0">

<li class="nav-item">

<a class="nav-link" asp-controller="Home" asp-action="Index">Главная</a>

</li>

<li class="nav-item">

<a class="nav-link" href="/Pets/List/category1">Собаки</a>

</li>

<li class="nav-item">

<a class="nav-link" href="/Pets/List/category2">Коты</a>

</li>

<li class="nav-item">

<a class="nav-link" asp-controller="CRUD\_ORDER" asp-action="Read">Работа с обращениями</a>

</li>

<li class="nav-item">

<a class="nav-link" asp-controller="CRUD\_NEWS" asp-action="Read">Работа с новостями</a>

</li>

<li class="nav-item">

<a class="nav-link" asp-controller="CRUD" asp-action="Create">Добавление питомцев</a>

</li>

@if (User.Identity.Name != null)

{

<li class="nav-item">

<p class="nav-link"> Логин: @User.Identity.Name</p>

</li>

}

<li class="nav-item">

@if (User.Identity.Name != null)

{

<a asp-controller="Account" asp-action="Lagout" class="header-a">

<button class="exit-button">

Выход

</button>

</a>

}

</li>

</ul>

</div>

</div>

</nav>

</header>

<table class="table">

<thead>

<tr>

<th>

@Html.DisplayNameFor(model => model.Id)

</th>

<th>

@Html.DisplayNameFor(model => model.name)

</th>

<th>

@Html.DisplayNameFor(model => model.shortDesc)

</th>

<th>

@Html.DisplayNameFor(model => model.img)

</th>

<th>

@Html.DisplayNameFor(model => model.vaccinations)

</th>

<th>

@Html.DisplayNameFor(model => model.breed)

</th>

<th>

@Html.DisplayNameFor(model => model.categoryID)

</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var item in Model)

{

<tr>

<td>

@Html.DisplayFor(modelItem => item.Id)

</td>

<td>

@Html.DisplayFor(modelItem => item.name)

</td>

<td>

@Html.DisplayFor(modelItem => item.shortDesc)

</td>

<td>

@Html.DisplayFor(modelItem => item.img)

</td>

<td>

@Html.DisplayFor(modelItem => item.vaccinations)

</td>

<td>

@Html.DisplayFor(modelItem => item.breed)

</td>

<td>

@Html.DisplayFor(modelItem => item.categoryID)

</td>

<td>

<a asp-action="Update" asp-route-id="@item.Id">Редактирование</a> |

<a asp-action="Delete" asp-route-id="@item.Id">Удаление</a>

</td>

</tr>

}

</tbody>

</table>

</body>

</html>