Гуманитарлық-техникалық колледж

Бекітемін / Утверждаю

ОІЖ директор орынбасары /

Заместитель директора по УР

\_\_\_\_\_\_\_\_\_\_ А.К. Макишев

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2024 жыл

Дипломдық жобаға түсініктеме хат

Пояснительная записка к дипломному проекту

Тақырып / Тема: «Информационно-поисковая система по стрелковому оружию вооружённых сил Республики Казахстан»

1304.ДП.19.000.00.ПЗ

Мамандық / Специальность: «Вычислительная техника и программное обеспечение»

Топ / Группа 404

|  |  |  |
| --- | --- | --- |
| Дипломшы / Дипломник | \_\_\_\_\_\_\_\_\_\_\_\_ | Р.М. Жусупов |
| Жетекші / Руководитель | \_\_\_\_\_\_\_\_\_\_\_\_ | А.Ю. Бодряков |
| Нормобақылау / Нормоконтролер | \_\_\_\_\_\_\_\_\_\_\_\_ | Г.К. Бисенова |

Петропавл – 2024

Содержание

[Введение 4](#_Toc169877081)

[1. Аналитическая часть 6](#_Toc169877082)

[1.1. Вооружённые силы Республики Казахстан. 6](#_Toc169877083)

[1.2. Стрелковое оружие. 7](#_Toc169877084)

[1.3. Анализ возможностей автоматизации. 7](#_Toc169877085)

[1.4. Обоснование необходимости использования ВТ и создания информационной системы. 8](#_Toc169877086)

[2. Проектная часть 10](#_Toc169877087)

[2.1 Информационное обеспечение проекта. 10](#_Toc169877088)

[2.2. Обоснование выбора инструмента разработки 11](#_Toc169877089)

[2.3. Этапы реализации проекта. 14](#_Toc169877090)

[2.4. Основные объекты базы данных. 14](#_Toc169877091)

[2.5. Web-интерфейс приложения. 16](#_Toc169877092)

[2.6. Описание запросов к базе данных. 17](#_Toc169877093)

[2.7 Описание программной логики взаимодействия с базой данных. 18](#_Toc169877094)

[2.8. Эксплуатационные требования необходимые для работы приложения 22](#_Toc169877095)

[Заключение 23](#_Toc169877096)

[Список литературы 24](#_Toc169877097)

[Приложение. Код программы 25](#_Toc169877098)

Введение

В наше время Интернет становиться все более развитой средой для осуществления коммуникаций с потребителями. Существенным является и тот факт, что Интернет становиться удобной и достаточно дешевой «торговой площадкой». Повлиял он также и на современный бизнес, потому что многие компании уже имеют собственный сайт, который используется в качестве визитной карточки или рекламы, все большее количество фирм старается представить свою продукцию в on-line среде. При этом такое представление не ограничивается только лишь созданием промо-сайтов и размещением рекламных баннеров и статей в электронных журналах и на информационных порталах.

Компьютерные системы, особенно в области коммуникаций и хранения данных, развиваются несколько быстрее, чем остальные, потому что им приходится идти в ногу с технологическими и социальными изменениями в мире. В связи с тем, что объемы информации, подлежащей хранению, растут высокими темпам, производители программного обеспечения вынуждены разрабатывать новые гибкие подходы к управлению большими объемами данных.

На протяжении последних лет интернет-технологии стремительно развивались, и сегодня большинство людей предпочитает обращаться к онлайн-ресурсам для поиска необходимой информации и услуг. Веб-сайт - один из таких ресурсов - отвечает этому тренду, предоставляя удобную платформу для взаимодействия между автором и посетителями.

Веб-сайт предоставляет удобную платформу для коммуникации и взаимодействия с клиентами. Через формы обратной связи и получить консультацию или уточнить информацию. Такое прямое взаимодействие с клиентами способствует улучшению обслуживания и повышению уровня доверия.

Актуальность проблемы.

В вооружённых силах Республики Казахстан используется большая номенклатура стрелкового оружия, каждый вид которого обладает определёнными параметрами, например: калибр, дальность поражения, ёмкость магазина и т.п. У многих людей вызывает интерес тема оружия, и в частности, оружия используемого в армии Казахстана, и в частности стрелкового оружия. Однако трудно найти информационный источник, где в одном месте собрана вся информация о стрелковом оружии используемом в армии Казахстана и его характеристиках. Web-сайт разработанный в рамках данного дипломного проекта призван решить эту проблему.

Цель дипломного проекта.

Создание информационно-поисковой системы по стрелковому оружию вооружённых сил Республики Казахстан, в виде web-сайта, в котором, в удобной и наглядной форме, будет отражена информация о типах и характеристиках стрелкового оружия ВС РК.

Задачи дипломного проекта:

1. Сбор и анализ информации о стрелковом оружии в вооружённых силах Республики Казахстан, его видах и характеристиках;

2. Рассмотрение потребностей и возможностей визуализации этой информации в виде web-сайта;

3. Проектирование удобной и функциональной базы данных, хранящей информацию о стрелковом оружии и его характеристиках.

4. Создание комплекса запросов к базе данных позволяющих извлекать информацию из базы данных по запросу пользователя в удобном для него формате.

5. Разработка web-сайта в качестве web-интерфейса к базе данных.

6. Тестирование.

# 1. Аналитическая часть

## 1.1. Вооружённые силы Республики Казахстан.

Вооружённые силы Республики Казахстан (каз. Қазақстан Республикасының қарулы күштері) — военная организация Республики Казахстан для защиты независимости и территориальной целостности государства от возможных внешних оппонентов.

Согласно закону Республики Казахстан «Об обороне и Вооружённых Силах Республики Казахстан» от 16 февраля 2012 года, к вооружённым силам относятся все формирования и учреждения находящиеся в структуре Министерства обороны РК.

В систему органов военного управления Вооружённые силы Республики Казахстан входят:

органы военного управления;

виды вооружённых сил:

сухопутные войска;

специальные войска;

ракетные войска и артиллерия;

силы воздушной обороны;

военно-морские силы;

десантно-штурмовые войска;

силы специальных операций;

военно-учебные заведения;

военно-научные учреждения;

другие организации.

Верховным Главнокомандующим ВС РК является президент страны, который осуществляет общее руководство строительством, подготовкой и применением военной организации, обеспечением военной безопасности государства.

Парламент РК принимает законы по вопросам военной безопасности и обороны государства, решает вопросы войны и мира, принимает решение об использовании Вооружённых Сил для выполнения международных обязательств по поддержанию мира и безопасности, ратифицирует и денонсирует международные договоры по вопросам обороны и военного сотрудничества.

Правительство РК организует оснащение Вооружённых Сил вооружением, военной и специальной техникой, их финансовое и материальное обеспечение, оперативное оборудование территории страны в интересах обороны, гражданскую и территориальную оборону, определяет перечень стратегических объектов, подлежащих физической защите, штатную численность Вооружённых Сил, порядок предоставления и использования для нужд обороны всех природных ресурсов.

Министерство обороны РК через Генеральный штаб ВС осуществляет централизованное руководство Вооружёнными Силами в мирное время, решает задачи, связанные с обеспечением военной безопасности и обороны страны, координирует разработку основных направлений военной реформы, программ развития Вооружённых Сил, определяет их структуру и штатную численность, проводит единую военно-техническую политику в государстве, осуществляет международное военное сотрудничество.

Генеральный штаб ВС РК разрабатывает основные направления военной реформы, планы строительства и развития Вооружённых Сил, их оперативной, боевой и мобилизационной подготовки, организует и осуществляет планирование применения Вооружённых Сил, разрабатывает план оперативного оборудования территории страны в интересах обороны.

Аналогичным образом осуществляется руководство другими войсками и воинскими формированиями, предназначенными для выполнения задач военными методами, — к ним относятся Служба государственной охраны, Национальная гвардия МВД РК, Пограничная служба КНБ РК и органы управления Гражданской обороны КЧС МВД РК.

## 1.2. Стрелковое оружие.

Стрелковое оружие обычно включает в себя такие виды огнестрельного оружия, как пистолеты и револьверы, винтовки и автоматы, карабины и ружья, а также ручные пулемёты и пистолеты-пулемёты.

Cтрелковое оружие определяется, как «ствольное оружие калибром менее 20 мм, предназначенное для метания пули, дроби или картечи». Таким образом, под стрелковое оружие также попадают крупнокалиберные пулемёты. Отдельно к стрелковому оружию относят стрелково-гранатомётные комплексы.

Cтрелковое оружие - это подвид огнестрельного оружия.

Огнестрельное оружие — оружие дальнего и ближнего боя, в котором для разгона и выбрасывания снаряда (мины, пули) из канала ствола используется сила давления газов, образующихся при сгорании топливо-окислительных горючих смесей, как правило пороха. Огнестрельное оружие сочетает в себе средство непосредственного поражения (артиллерийский снаряд, мина, пуля) и средство метания их к цели (автоматы, пистолеты, пушки и др.).

## 1.3. Анализ возможностей автоматизации.

Информационным фундаментом ИС являются базы данных и знаний.

Понятие базы данных можно применить к любой связанной между собой по определенному признаку информации, хранимой и организованной особым образом - как правило, в виде таблиц. По сути, БД – это некоторое подобие электронной картотеки, электронного хранилища данных, которое хранится в компьютере в виде одного или нескольких данных. Это данные снабженные их определением.

Одной из современных форм организации информации являются банки данных. «Банк данных – это система специальным образом организованных данных (баз данных), программных, технических, организационно – методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования этих данных».

Для принятия обоснованных и эффективных решений в производственной деятельности, в управлении экономикой и в политике современный специалист должен уметь с помощью компьютеров и средств получать, накапливать, хранить и обрабатывать данные, представляя результат в виде наглядных документов.

Современная жизнь немыслима без эффективного управления. Важной категорией являются системы обработки информации, от которых во многом зависит эффективность работы любого предприятия ли учреждения.

На сегодняшний день не существует такой организации или предприятия, которое бы в своей деятельности не сталкивалось с рядом проблем. Если же все-таки явные проблемы не обнаружены, то задачи, связанные с усовершенствованием, обновлением оборудования, программного обеспечения, повышением эффективности управления, автоматизацией ручного труда были актуальны в любой сфере деятельности. Поэтому основной задачей предприятия является автоматизация повседневной работы.

Важность программного обеспечения определяется тем, что именно в программном обеспечении закладываются и реализуются функции автоматизированных систем управления процессом производства, контролем за его ходом и возможностью корректировки производства с целью снижения затрат и времени. Программное изделие должно быть разработано так, чтобы оно выполняло свои функции без лишних затрат ресурсов: оперативной памяти, машинного времени, пропускной способности каналов, передачи данных – на стадии функционирования и на стадии создания программных изделий. Нормирование труда в процессе создания программных изделий, а следовательно затраты времени и денежные расходы обусловлены определенными трудностями, в связи с отсутствием методики определения норм времени на создание постоянно изменяющихся требований заказчика программных изделий. Оценка труда программистов по времени возможна установлением жестко заданных сроков исполнения, однако представляется возможным, на основе экспертных оценок и существующих норм на создание базовых программ.

## 1.4. Обоснование необходимости использования ВТ и создания информационной системы.

Использование самой вычислительной техники необходимо по ряду многих причин, в частности:

постоянно растущие объемы обрабатываемой информации

сложности возникающие с созданием большого количества разнообразных документов.

оперативность получения необходимых данных. (Имеется в виду не только быстрое составление готовых отчетов, но и возможность корректировки уже введенной информации по отдельным элементам базы данных).

Отсюда вытекает одно немаловажное требование к программе:

создание простого интерфейса, что позволяет пользователю более быстро освоить общение с компьютером.

Все выше перечисленное позволяют сказать с уверенностью, что на данном рабочем месте просто необходимо использовать вычислительную машину с соответствующим программным обеспечением.

Для того чтобы эффективнее использовать работу служащих отдела кадров необходимо на базе имеющегося оборудования и программного обеспечения создать автоматизированное рабочее место (ИС).

К информационно-поисковой системе предъявляются следующие требования:

наличие распределенной баз данных и знаний, постоянно пополняемых оперативной и достоверной информацией;

к отдельным базам данных и знаний или их фрагментам может иметь доступ только ограниченный круг лиц, а к отдельным элементам – только секретарь;

обеспечения оперативного поиска информации;

наличие программных средств обеспечения принятия управленческого решения;

простота работы;

обеспечения оперативной связи с другими источниками информации в пределах организационной структуры предприятия или его подразделения и т.д.

ИС обеспечивает решения задач профессиональной деятельности на основе как локальных баз данных и знаний, так и распределенных баз данных и прикладного программного обеспечения.

ИС реализует функции:

ввод информации;

ведения баз данных;

обработка входящей и исходящей документации;

контроль исполнительской деятельности.

Применение баз данных в несколько раз снижает затраты на документооборот, повышает скорость и качество подготовки документов, упорядочивает организационную структуру управления и тем самым повышает эффективность управления.

Одним из основных требований к технологии баз данных является автоматизация экранного диалога, особенно важны три показателя: дружественность, гибкость, продуктивность.

Понятие ИС можно применить к любой связанной между собой по определенному признаку информации, хранимой и организованной особым образом - как правило, в виде таблиц.

По сути, БД – это некоторое подобие электронной картотеки, электронного хранилища данных, которое хранится в компьютере в виде одного или нескольких данных. Это данные снабженные их определением.

Одной из современных форм организации информации являются банки данных.

«Банк данных – это система специальным образом организованных данных (баз данных), программных, технических, организационно – методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования этих данных».

# 2. Проектная часть

## 2.1 Информационное обеспечение проекта.

Цель инфологического моделирования – обеспечение наиболее естественных для человека способов сбора и представления той информации, которую предполагается хранить в создаваемых базах данных. Основными конструктивными элементами инфологических моделей являются сущности, связи между ними и их свойства (атрибуты).

Сущность – любой различимый объект, информацию о котором необходимо хранить в базе данных.

Атрибут – поименованная характеристика сущности. Атрибуты используются для определения того, какая информация должна быть собрана о сущности. Примерами атрибутов для сущности "Студенты" являются фамилия, имя, отчество, профессия, нострификация, адрес и т.д.

Связь – ассоциирование двух или более сущностей. Если бы назначением базы данных было только хранение отдельных, не связанных между собой данных, то ее структура могла бы быть очень простой. Однако одно из основных требований к организации базы данных – это обеспечение возможности отыскания одних сущностей по значениям других, для чего необходимо установить между ними определенные связи.

В данном проекте имеются следующие объекты (таблицы) и связи (рисунок 1.):

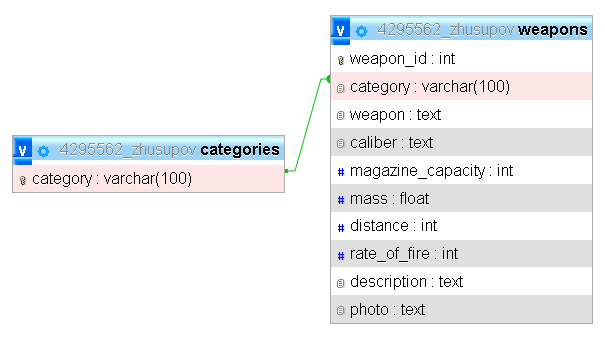


Рисунок 1. Инфологическая модель.

## 2.2. Обоснование выбора инструмента разработки

### 2.2.1. Обоснования выбора системы управления базами данных

MySQL – это реляционная система управления базами данных. То есть данные в ее базах хранятся в виде логически связанных между собой таблиц, доступ к которым осуществляется с помощью языка запросов SQL. Mysql – свободно распространяемая система, т.е. платить за ее применение не нужно. Кроме того, это достаточно быстрая, надежная и, главное, простая в использовании СУБД, вполне подходящая для не слишком глобальных проектов.

Работать с Mysql можно не только в текстовом режиме, но и в графическом. Существует очень популярный визуальный интерфейс (кстати, написанный на PHP) для работы с этой СУБД. Называется он PhpMyAdmin. Этот интерфейс позволяет значительно упростить работу с базами данных в Mysql.

PhpMyAdmin позволяет пользоваться всеми достоинствами браузера, включая прокрутку изображения, если оно не умещается на экран. Многие из базовых SQLфункций работы с данными в PhpMyAdmin сведены к интуитивно понятным интерфейсам и действиям, напоминающим переход по ссылкам в Internet. Но, тем не менее, стоит все же поработать и в текстовом режиме.

Перед тем как переходить к детальному изучению языка SQL, несколько слов об установке Mysql и подготовке к работе. Если вы не собираетесь заниматься администрированием сервера, то информация, приведенная ниже, пригодится вам только для общего развития. Итак, устанавливается Mysql очень просто – автоматически, пару раз нажмите OK, и все. После этого вы можете зайти в директорию, где лежат файлы типа Mysql.exe, Mysqld.exe и т.п. (у нас под Windows XP это c:\Mysql\bin) Последний файл запускает Mysql-сервер. В некоторых системах сервер запускается в виде сервиса. После запуска сервера следует запустить Mysql-клиент, запустив программу Mysql.exe. Здесь даже пароля не спросят. Более того, если вы наберете shell> Mysql.exe -u root или shell>Mysql -u root Mysql то получите все права администратора Mysql сервера. Кстати, выполнять эти команды надо, находясь в той директории, где лежат файлы Mysql.exe.

Для начала, не вдаваясь в подробности команд, исправим эти два недочета (отсутствие пароля у администратора и возможность входа анонимным пользователям):

### 2.2.2. Обоснования выбора web-сервера.

Open Server — это портативный локальный WAMP/WNMP сервер, имеющий многофункциональную управляющую программу и большой выбор подключаемых компонентов. Представленный пакет программ не является очередной любительской сборкой собранной «на коленке», это первый полноценный профессиональный инструмент, созданный специально для веб-разработчиков с учётом их рекомендаций и пожеланий.

Компоненты и инструменты

Для отладки скриптов в различном окружении Open Server предлагает на выбор сразу два вида HTTP серверов, различные версии PHP и СУБД модулей, а так же возможность быстрого переключения между ними.

HTTP модули: Apache 2.2.21 и Nginx 1.0.11;

СУБД модули: MySQL 5.1.61, MySQL 5.5.20 и PostgreSQL 9.1.1;

PHP модули: PHP 5.2.17 (IMagick 2.2.1, Zend Optimizer 3.3.3, IonCube Loader 4.0.7, Memcache 2.2.4) и PHP 5.3.9 (IMagick 2.3.0, Xdebug 2.1.3, IonCube Loader 4.0.10, Memcache 2.2.6);

Отличный набор инструментов: HeidiSQL, Adminer, PHPMyAdmin, PHPPgAdmin, PgAdmin.

В состав пакета так же включены такие вкусности как: Perl, FTP сервер, Sendmail, Memcached сервер!

Open Server — это единственный известный мне проект, в который включён Nginx! Причём здесь реализовано удобное подключение правил реврайта через файлы .nxaccess в корне домена, а PHP работает в режиме True FastCGI (неубиваемый).

Все компоненты взяты с официальных репозиториев и всегда обновляются до актуальных версий с каждым обновлением пакета.

Возможности

Прежде всего хочу отметить, что Open Server — это целиком и полностью портативный сервер. Никаких системных сервисов, куч мусора в реестре и system32. Вы можете везде таскать его с собой на флешке (желательно на скоростной), запускать на рабочей/домашней машине без опаски что у вас что-то не заработает.

В случае отсутствия на компьютере нужных системных компонентов Open Server установит их сам, достаточно выбрать в меню [Инструменты — Первый запуск] если сервер запускается на компьютере впервые.

Не хочу описывать основные возможности программы, поскольку в большинстве своём они стандартны для такого рода софта. Самой собой что с помощью Open Server можно запустить/остановить сервер или открыть нужный домен. Гораздо интереснее вам будет узнать о специфических «фичах», которые делают Open Server особенным, действительно особенным:

— подробный просмотр логов всех компонентов в реальном времени;

— выбор HTTP, СУБД и PHP модулей в любом сочетании;

— поддержка SSL и кириллических доменов из коробки;

— поддержка алиасов или по другому доменных указателей, а так же удобная форма их настройки (привет любителям мультисайтинга в Drupal!);

— создание локального поддомена без потери видимости основного домена в сети интернет;

— доступ к доменам (в один клик) и быстрый доступ к шаблонам конфигурации модулей;

— мультиязычный интерфейс (Русский, Украинский, Белорусский, Английский).

### 2.2.3. Обоснования выбора языка web-программирования.

PHP (рекурсивный акроним словосочетания PHP: Hypertext Preprocessor) - это распространенный язык программирования общего назначения с открытым исходным кодом. PHP сконструирован специально для ведения Web-разработок и его код может внедряться непосредственно в HTML.

Вместо рутинного вывода HTML-кода командами языка (как это происходит, например, в Perl или C), скрипт PHP содержит HTML с вкраплениями кода (в нашем случае, это вывод текста "Привет, я - скрипт PHP!"). Код PHP отделяется специальными начальным и конечным тегами <?php и ?>, которые позволяют "переключаться" в "PHP-режим" и выходить из него.

PHP отличается от JavaScript тем, что PHP-скрипты выполняются на сервере и генерируют HTML, который посылается клиенту. Если бы у вас на сервере был размещен скрипт, подобный вышеприведенному, клиент получил бы только результат его выполнения, но не смог бы выяснить, какой именно код его произвел. Вы даже можете настроить свой сервер таким образом, чтобы обычные HTML-файлы обрабатывались процессором PHP, так что клиенты даже не смогут узнать, получают ли они обычный HTML-файл или результат выполнения скрипта.

PHP крайне прост для освоения, но вместе с тем способен удовлетворить запросы профессиональных программистов. Не пугайтесь длинного списка возможностей PHP. Вы можете быстро начать, и уже в течение нескольких часов сможете создавать простые PHP-скрипты.

Хотя PHP, главным образом, предназначен для работы в среде web-серверов, область его применения не ограничивается только этим. Читайте дальше и не пропустите главу Возможности PHP либо, начните непосредственно с Вводного руководства, если вас интересует исключительно веб-программирование.

### 2.2.4. Обоснования выбора технологий построения web-интерфейса.

Язык разметки гипертекстовых страниц (HTML – Hypertext Markup Language) представляет собой язык, разработанный специально для создания Web-документов. Он определяет синтаксис и размещение специальных инструкций (тегов), которые не выводятся на экран, но указывают браузеру, как отображать содержимое документа. Он также используется для создания ссылок на другие документы, локальные или сетевые, например, находящиеся в сети Интернет.

Стандарт HTML и другие стандарты для Web разработаны под руководством консорциума W3C (World Wide Web Consortium). Стандарты, спецификации и проекты новых предложений можно найти на сайте http://www.3w.org/. В настоящее время действует спецификация HTML 4.0, поддержка которой со стороны основных браузеров постоянно растет.

На практике на стандарт HTML большое влияние оказывает наличие тегов, предложенных и поддерживаемых наиболее известными браузерами, такими как Microsoft Internet Explorer и Netscape Navigator. Эти теги в данный момент могут как входить, так и не входить в состав действующей спецификации HTML.

## 2.3. Этапы реализации проекта.

Во время разработки данного дипломного проекта были пройдены следующие этапы:

1. Сбор информации о стрелковом оружии ВС РК.

2. Поиск и анализ подобных информационных ресурсов.

3. Разработка общей концепции проекта

4. Разработка структуры базы данных.

5. Заполнение БД информацией.

6. Создание PHP приложения.

7. Разработка программной логики получения данных из базы данных с помощью SQL-запросов.

9. Создание web-страниц – интерфейса к приложению

10. Тестирование и отладка базы данных, приложения и web-сайта.

## 2.4. Основные объекты базы данных.

В процессе разработки базы данных была выработана следующая структура базы данных, которая отображена на схеме (рисунок 2).

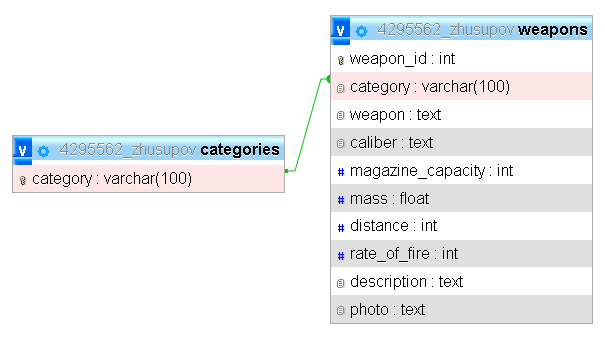


Рисунок 2. Структура базы данных.

Таблица БД Weapons (оружие) является основной таблицей базы данных. Таблица Categories (категории), хранящая список категорий стрелкового оружия, связана с главной отношением «один ко многим» и является справочной для облегчения и убыстрения ввода информации (выбора одного пункта из нескольких доступных). Структура таблицы БД Weapons показана на рисунке 3.



Рисунок 3. Структура основной таблицы БД Weapons.

Кроме таблиц в базе данных имеются другие объекты.

Таблица Weapons заполненная данными частично показана на рисунке 4:

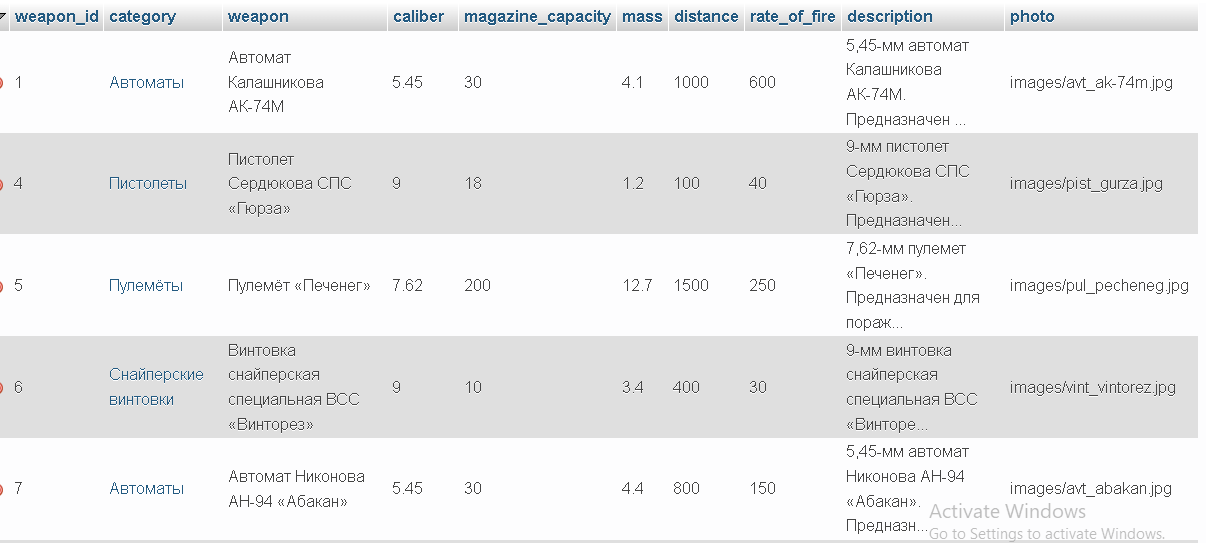


Рисунок 4. Частичный вид основной таблицы БД Weapons.

Таблица Categories связана с основной таблицей базы данных отношением один-ко-многим и является справочником категорий (видов) стрелкового оружия. Таблица Categories заполненная данными показана на рисунке 5:

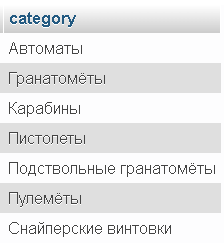


Рисунок 5. Таблицы БД Categories с информацией.

## 2.5. Web-интерфейс приложения.

Web-интерфейс приложения состоит из нескольких web-страниц HTML оформленных с помощью стилей CSS. Сначала открывается главная (стартовая) страница (рисунок 6).



Рисунок 6. Стартовая страница приложения.

Далее посетитель может, выбрав соответствующий пункт меню, перейти на одну из следующих страниц:

1. Все данные - страница отображает в табличном виде всю информацию из таблицы БД оружие. Т.е. всё оружие со всеми характеристиками.

2. Сортировка - страница отображает также всю информацию об оружии в табличном виде, но с возможностью сортировки по некоторым столбцам (полям).

3. По категориям - страница содержит поле со списком для выбора категории оружия. После выбора пользователем соответствующей категории - в таблице отображается оружие только этой категории.

4. По калибру - страница содержит поле со списком для выбора калибра оружия. После выбора пользователем соответствующего калибра - в таблице отображается оружие только этого калибра.

5. Фотогаллерея - карточки с фотографиями соответствующих образцов оружия с подписями.

6. Обо мне - информация о авторе-разработчике приложения.

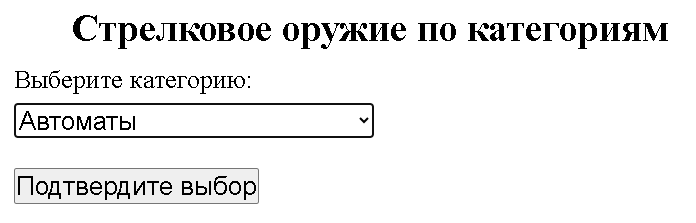


Рисунок 7. Частичный вид страницы "По категориям".

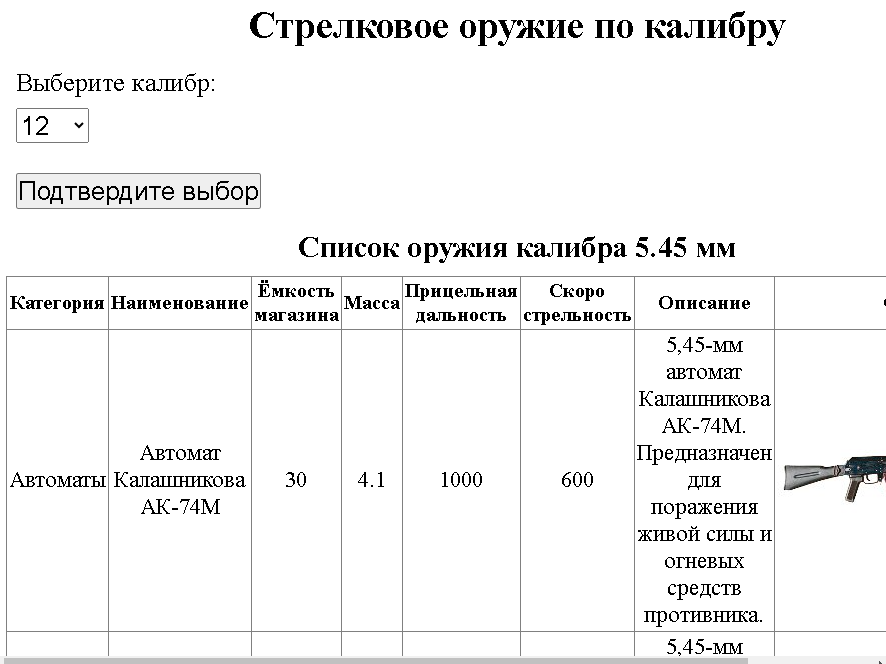


Рисунок 8. Частичный вид страницы "По калибру".

## 2.6. Описание запросов к базе данных.

Все запросы к базе данных MySQL реализованы на языке SQL.

1. Запрос выводящий все данные из таблицы Оружие:

SELECT \* FROM weapons ORDER BY category ASC

2. Запрос выводящий оружие выбранного калибра:

SELECT DISTINCT caliber FROM weapons ORDER BY caliber ASC

;

3. Запрос выводящий оружие данной категории:

  $sql\_selected\_category = "SELECT \* FROM weapons WHERE category = ?";

  $query\_selected\_category = $pdo->prepare($sql\_selected\_category);

  $query\_selected\_category->execute([$category]);

## 2.7 Описание программной логики взаимодействия с базой данных.

Программная логика взаимодействия с базой данных реализована на языке РНР. Ниже приводится пример РНР-кода реализующий получение списка оружия выбранной категории категории.

<?

// Получение списка калибров оружия из базы данных

$sql\_caliber = "SELECT DISTINCT caliber FROM weapons ORDER BY caliber ASC";

$query\_caliber = $pdo->query($sql\_caliber)->fetchall(PDO::FETCH\_COLUMN);

?>

<?

// Функция вывода таблицы, вызываемая, когда выбран калибр оружия

function table\_output($pdo, $caliber) {

  // Выбранный калибр передаётся в функцию в переменной $caliber

  echo "<h3> Список оружия калибра $caliber мм </h3>";

  // Делаем SQL-запрос выбирая оружие только данного калибра

  $sql\_selected\_caliber = "SELECT \* FROM weapons WHERE caliber = ? ORDER BY category ASC";

  $query\_selected\_caliber = $pdo->prepare($sql\_selected\_caliber);

  $query\_selected\_caliber->execute([$caliber]);

  // Формируем таблицу вывода оружия данного калибра

  echo "<table border=1>";

  echo "<tr>

    <th>Категория</th>

    <th>Наименование</th>

    <th>Ёмкость магазина</th>

    <th>Масса</th>

    <th>Прицельная дальность</th>

    <th>Скоро<br>стрельность</th>

    <th>Описание</th>

    <th>Фото</th>

  </tr>";

  foreach ($query\_selected\_caliber as $row) {

    $pole2=$row['category'];

    $pole3=$row['weapon'];

    $pole5=$row['magazine\_capacity'];

    $pole6=$row['mass'];

    $pole7=$row['distance'];

    $pole8=$row['rate\_of\_fire'];

    $pole9=$row['description'];

    $pole10=$row['photo'];

    echo "<tr>

      <td>$pole2</td> <td>$pole3</td> <td>$pole5</td> <td>$pole6</td>

      <td>$pole7</td> <td>$pole8</td> <td>$pole9</td> <td><img src=$pole10></td>

    </tr>";

  }

  echo "</table>";

}

?>

<!-- Основное содержимое страницы "Оружие по калибру" -->

<main class='other-content'>

<h2> Стрелковое оружие по калибру </h2>

<section class="section-form">

  <!-- Форма списка калибров оружия -->

  <form name="form\_caliber" method="post">

    <p>Выберите калибр:</p>

    <select name="calibres">

      <!-- Заполняем список калибров из поля 'калибр' таблицы 'оружие' базы данных -->

      <?

        foreach ($query\_caliber as $row) {

          echo "<option>".$row."</option>";

        }

      ?>

    </select>

    <br><br>

    <input name="submit" type="submit" value="Подтвердите выбор">

  </form>

</section>

<?

// Код обработки выбора калибра оружия из списка

if(isset($\_POST["submit"])){

  if(isset($\_POST["calibres"])){

// Если категория выбрана и кнопка подтверждения нажата :

    $caliber = $\_POST["calibres"];

    // Запускаем функцию вывода оружия выбранной калибра

    table\_output($pdo, $caliber);

  }

}

?>

<?

  // Количество обращений по категориям

  // Если в списке ниже выбранна категория и нажата кнопка Отправить категорию

  if(isset($\_POST["submit\_category"])) {

    if(isset($\_POST["categories"])) {

      // Получаем выбранную категорию

      $selected\_category = $\_POST["categories"];

      // и в SQL-запросе считаем количество обращений по выбранной категории

      $sql = "SELECT COUNT('appeal\_category') FROM appeals WHERE appeal\_category = '$selected\_category'";

      // Получаем результат SQL-запроса и записываем их в переменную

      $selected\_category\_count = $pdo->query($sql)->fetchColumn();

    }

  }

?>

<?

  // Для построения диаграммы пишем SQL-запрос в ктором

  // считаем количество обращений по каждой категории

  $sql = "SELECT appeal\_category, COUNT(appeal\_category) FROM appeals GROUP BY appeal\_category";

  // Получаем результат SQL-запроса и записываем их в переменную

  $appeals\_by\_categories = $pdo->query($sql)->fetchAll();

?>

  Количество обращений по категориям

  <!-- Сначала сформируем раскрывающийся список выбора категории -->

  <form method="POST" name="selecting\_categories" >

    <select name="categories" class="select-categories">

    <?

      // Заполним список категорий из базы данных

      // из таблицы appeal\_categories

      $sql = "SELECT \* FROM appeal\_categories";

      $query\_categories = $pdo->query($sql);

      foreach ($query\_categories as $row) {

        $category=$row['category'];

        echo "<option value=" . $category . ">";

        echo "$category";

        echo "</option>";

      }

    ?>

    </select>

    <!-- После выбора категории посетитель нажимает кнопку

        и отправляет выбранную категорию в обработчик

        в начале этой страницы -->

    <input name="submit\_category" type="submit" value="Выбрать категорию" class="button">

  </form>

</p>

<!-- Показ количества обращений по выбранной категории  -->

<?

  // Сделаем так, чтобы весь функционал обращений по категориям,

  // работал только если нажата кнопка "Отправить категорию"

  if(isset($\_POST["submit\_category"])){

    if(isset($\_POST["categories"])){

?>

<!-- Выводим количество обращений по категориям полученных в коде вначале страницы -->

<p>

  Количество обращений категории

  <span class="param-text"> <? echo $selected\_category; ?> </span> :

  <span class="param">

    <? echo $selected\_category\_count; ?>

  </span>

</p>

<?

    }

  }

?>

<!-- КОНЕЦ раздела "Количество обращений по категориям" -->

## 2.8. Эксплуатационные требования необходимые для работы приложения

### [2.8.1. Программное обеспечение необходимое для работы клиентской части приложения](#_Toc104342514)

Клиентская часть рассматриваемого приложения будет работать в любых современных браузерах (хотя предпочтительней Google Chrome) на операционных системах: Windows, Android, iOS, Linux и на любых устройствах (смартфоны, планшеты, мониторы и др.)

### [2.8.2. Программное обеспечение необходимое для работы серверной части приложения](#_Toc104342514)

Разработанное приложение должно функционировать на web-сервере с установленным сервером Apachi версии не ранее 2.2, интерпретатором PHP версии не ранее 5.6, СУБД MySQL версии не ранее 5.2.

# 

Заключение

В результате разработки данного проекта была реализована цель проекта - создание информационно-поисковой системы по стрелковому оружию вооружённых сил Республики Казахстан, в виде web-сайта, в котором, в удобной и наглядной форме, отражена информация о типах и характеристиках стрелкового оружия ВС РК.

Также выполнены все поставленные задачи, а именно:

1. Сбор и анализ информации о стрелковом оружии в вооружённых силах Республики Казахстан, его видах и характеристиках;

2. Рассмотрение потребностей и возможностей визуализации этой информации в виде web-сайта;

3. Проектирование удобной и функциональной базы данных, хранящей информацию о стрелковом оружии и его характеристиках.

4. Создание комплекса запросов к базе данных позволяющих извлекать информацию из базы данных по запросу пользователя в удобном для него формате.

5. Разработка web-сайта в качестве web-интерфейса к базе данных.

6. Тестирование.

Приложение расположено на временном бесплатном хостинге для публичного доступа и готово к тестированию и использованию.

В дальнейшем возможно улучшение пользовательского интерфейса, дизайна, а также добавление дополнительных возможностей.

Список литературы

1. ГОСТ 19.105-78 ЕСПД. Общие требования к программным документам;
2. ГОСТ 19.404-79 ЕСПД. Пояснительная записка. Требования к содержанию и оформлению;
3. Гринченко Н. Н., Гусев Е. В., Макаров Н. П., Пылькин А. Н., Цуканова Н. И.. Проектирование баз данных. СУБД MySQL. Горячая линия – Телеком, 2022. – 236 с.
4. Гончаров А.Ю. MYSQL 2024. КУДИЦ-ОБРАЗ, 2024. – 272 с.
5. PHP и jQuery для профессионалов: Джейсон Ленгсторф — Санкт-Петербург, Вильямс, 2021 г.- 362 с.
6. Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript: Робин Никсон — Москва, Питер, 2023 г.- 496 с.
7. Сценарии для Web-сайта. PHP и JavaScript: Вадим Дунаев — Санкт-Петербург, БХВ-Петербург, 2018 г.- 576 с.
8. CSS. Рецепты программирования, 3-е издание. Кристофер Шмитт. Русская Редакция, БХВ-Петербург. 2022.
9. HTML5 за 10 минут. Стивен Хольцнер. Издательство: Вильямс. 2022 г.
10. HTML, javascript, PHP и MySQL. Джентльменский набор Web-мастера. Прохоренок Н. А. БХВ-Петербург. 2017 г.
11. Visual Studio Code [Электронный ресурс]. – URL: https://code.visualstudio.com
12. OpenServer [Электронный ресурс]. – URL: https://ospanel.io (11.06.2023)
13. RSS [Электронный ресурс]. – URL: http://www.rss.ru/Content/Pages/pay\_service

Приложение. Код программы

Стартовая страница приложения

<?

  require "template.php";

?>

<main class='index-content'>

<section class="section-text">

  <p>

  Стрелковое оружие обычно включает в себя такие виды огнестрельного оружия, как пистолеты и револьверы, винтовки и автоматы, карабины и ружья, а также ручные пулемёты и пистолеты-пулемёты.

  </p>

  <p>

    Cтрелковое оружие определяется, как «ствольное оружие калибром менее 20 мм, предназначенное для метания пули, дроби или картечи». Таким образом, под стрелковое оружие также попадают крупнокалиберные пулемёты. Отдельно к стрелковому оружию относят стрелково-гранатомётные комплексы.

  </p>

  <p>

    Cтрелковое оружие - это подвид огнестрельного оружия.

  </p>

  <p>

    Огнестрельное оружие — оружие дальнего и ближнего боя, в котором для разгона и выбрасывания снаряда (мины, пули) из канала ствола используется сила давления газов, образующихся при сгорании топливо-окислительных горючих смесей, как правило пороха.

    Огнестрельное оружие сочетает в себе средство непосредственного поражения (артиллерийский снаряд, мина, пуля) и средство метания их к цели (автоматы, пистолеты, пушки и др.).

  </p>

</section>

<section class="section-photo">

  <img src="images/z0.jpg" alt="">

</section>

<p> </p>

</main>

<?

  require "footer.php";

?>

Код подключения базы данных.

<?

 $dsn = "mysql:host=localhost;dbname=4295562\_zhusupov;charset=utf8";

 $user = '4295562\_zhusupov';

 $pass = 'zkhzdZdA4qb91q):';

 $opt = [

  PDO::ATTR\_ERRMODE            => PDO::ERRMODE\_EXCEPTION,

  PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

  PDO::ATTR\_EMULATE\_PREPARES   => false,

  ];

 $pdo = new PDO($dsn, $user, $pass, $opt);

//  echo "<p>Вы подключились к MySQL!</p>";

?>

Страница "оружие по категориям"

<?

  require "template.php";

?>

<?

// Получение списка категорий оружия из базы данных

$sql\_categories = "SELECT category FROM categories";

$query\_categories = $pdo->query($sql\_categories)->fetchall(PDO::FETCH\_COLUMN);

?>

<?

// Функция вывода таблицы, вызываемая, когда выбрана категория оружия

function table\_output($pdo, $category) {

  // Выбранная категория передаётся в функцию в переменной $category

  echo "<h3> Список оружия категории $category </h3>";

  // Делаем SQL-запрос выбирая оружие только данной категории

  $sql\_selected\_category = "SELECT \* FROM weapons WHERE category = ?";

  $query\_selected\_category = $pdo->prepare($sql\_selected\_category);

  $query\_selected\_category->execute([$category]);

  // Формируем таблицу вывода оружия данной категории

  echo "<table border=1>";

  echo "<tr>

    <th>Наименование</th>

    <th>Калибр</th>

    <th>Ёмкость магазина</th>

    <th>Масса</th>

    <th>Прицельная дальность</th>

    <th>Скоро<br>стрельность</th>

    <th>Описание</th>

    <th>Фото</th>

  </tr>";

  foreach ($query\_selected\_category as $row) {

    $pole3=$row['weapon'];

    $pole4=$row['caliber'];

    $pole5=$row['magazine\_capacity'];

    $pole6=$row['mass'];

    $pole7=$row['distance'];

    $pole8=$row['rate\_of\_fire'];

    $pole9=$row['description'];

    $pole10=$row['photo'];

    echo "<tr>

      <td>$pole3</td> <td>$pole4</td> <td>$pole5</td> <td>$pole6</td>

      <td>$pole7</td> <td>$pole8</td> <td>$pole9</td> <td><img src=$pole10></td>

    </tr>";

  }

  echo "</table>";

}

?>

<!-- Основное содержимое страницы "Оружие по категориям" -->

<main class='other-content'>

<h2> Стрелковое оружие по категориям </h2>

<section class="section-form">

<!-- Форма списка категорий оружия -->

  <form name="form\_categories" method="post">

    <p>Выберите категорию:</p>

    <select name="categories">

      <!-- Заполняем список категорий из таблицы категорий базы данных -->

      <?

        foreach ($query\_categories as $row) {

          echo "<option>".$row."</option>";

        }

      ?>

    </select>

    <br><br>

    <input name="submit" type="submit" value="Подтвердите выбор">

  </form>

</section>

<p> </p>

<?

// Код обработки выбора категории оружия из списка

if(isset($\_POST["submit"])){

  if(isset($\_POST["categories"])){

    // Если категория выбрана и кнопка подтверждения нажата :

    $category = $\_POST["categories"];

    // Запускаем функцию вывода оружия выбранной категории

    table\_output($pdo, $category);

  }

}

?>

<p>

  <br>

</p>

<a href="index.php">На главную</a>

</main>

<?

  require "footer.php";

?>

Страница "оружие по калибру"

<?

  require "template.php";

?>

<?

// Получение списка калибров оружия из базы данных

$sql\_caliber = "SELECT DISTINCT caliber FROM weapons ORDER BY caliber ASC";

$query\_caliber = $pdo->query($sql\_caliber)->fetchall(PDO::FETCH\_COLUMN);

?>

<?

// Функция вывода таблицы, вызываемая, когда выбран калибр оружия

function table\_output($pdo, $caliber) {

  // Выбранный калибр передаётся в функцию в переменной $caliber

  echo "<h3> Список оружия калибра $caliber мм </h3>";

  // Делаем SQL-запрос выбирая оружие только данного калибра

  $sql\_selected\_caliber = "SELECT \* FROM weapons WHERE caliber = ? ORDER BY category ASC";

  $query\_selected\_caliber = $pdo->prepare($sql\_selected\_caliber);

  $query\_selected\_caliber->execute([$caliber]);

  // Формируем таблицу вывода оружия данного калибра

  echo "<table border=1>";

  echo "<tr>

    <th>Категория</th>

    <th>Наименование</th>

    <th>Ёмкость магазина</th>

    <th>Масса</th>

    <th>Прицельная дальность</th>

    <th>Скоро<br>стрельность</th>

    <th>Описание</th>

    <th>Фото</th>

  </tr>";

  foreach ($query\_selected\_caliber as $row) {

    $pole2=$row['category'];

    $pole3=$row['weapon'];

    $pole5=$row['magazine\_capacity'];

    $pole6=$row['mass'];

    $pole7=$row['distance'];

    $pole8=$row['rate\_of\_fire'];

    $pole9=$row['description'];

    $pole10=$row['photo'];

    echo "<tr>

      <td>$pole2</td> <td>$pole3</td> <td>$pole5</td> <td>$pole6</td>

      <td>$pole7</td> <td>$pole8</td> <td>$pole9</td> <td><img src=$pole10></td>

    </tr>";

  }

  echo "</table>";

}

?>

<!-- Основное содержимое страницы "Оружие по калибру" -->

<main class='other-content'>

<h2> Стрелковое оружие по калибру </h2>

<section class="section-form">

  <!-- Форма списка калибров оружия -->

  <form name="form\_caliber" method="post">

    <p>Выберите калибр:</p>

    <select name="calibres">

      <!-- Заполняем список калибров из поля 'калибр' таблицы 'оружие' базы данных -->

      <?

        foreach ($query\_caliber as $row) {

          echo "<option>".$row."</option>";

        }

      ?>

    </select>

    <br><br>

    <input name="submit" type="submit" value="Подтвердите выбор">

  </form>

</section>

<p> </p>

<?

// Код обработки выбора калибра оружия из списка

if(isset($\_POST["submit"])){

  if(isset($\_POST["calibres"])){

    // Если категория выбрана и кнопка подтверждения нажата :

    $caliber = $\_POST["calibres"];

    // Запускаем функцию вывода оружия выбранной калибра

    table\_output($pdo, $caliber);

  }

}

?>

<p>

  <br>

</p>

<a href="index.php">На главную</a>

</main>

<?

  require "footer.php";

?>

Код CSS

:root {

  --index: calc(1vw + 1vh);

}

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

html, body {

  /\* min-height: 100%; \*/

}

body{

  /\* outline: 2px dashed blue; \*/

  margin: 1%;

  padding: 0.5%;

  display: grid;

  grid-template-rows: minmax(12vh, auto) minmax(8vh, auto) minmax(60vh, auto) minmax(6vh, auto);

  grid-gap: 2vh;

  font-size: 130%;

}

header, nav, main, footer {

  padding: 0.5%;

}

header {

  /\* outline: 2px solid gold; \*/

}

.main-header {

  background-image: url(images/z-collage.png);

  background-position: center;

  background-size: cover;

  width: 100%;

  height: auto;

}

nav {

  /\* outline: 2px solid green; \*/

}

.main-menu\_\_list {

  /\* outline: 2px dotted magenta; \*/

  list-style-type: none;

  padding: 0.3%;

  display:grid;

  grid-template-columns: repeat(7, 1fr);

  justify-items: center;

}

.main-menu\_\_list li {

  /\* outline: 1px solid blue; \*/

  padding: 1%;

  width: fit-content;

  text-align: center;

}

.main-menu\_\_list li a {

  text-decoration: none;

}

main {

  /\* outline: 2px solid red; \*/

}

.index-content {

  display: grid;

  grid-template-columns: repeat(2, 1fr);

  grid-gap: 1%;

}

section {

  padding: 1%;

}

.section-text {

  /\* outline: 2px dashed blue; \*/

}

.section-photo {

  /\* outline: 2px dashed blue; \*/

  background-image: url(images/z.jpg);

  background-repeat: no-repeat;

  background-position: center;

  background-size: cover;

  width: 100%;

  height: auto;

}

.other-content {

  /\* outline: 2px solid lime; \*/

}

footer {

  outline: 2px solid olive;

  text-align: center;

}

h1, h2, h3 {

  text-align: center;

}

h1 {

  font-size: 180%;

}

p {

  text-align: justify;

  margin-top: 1%;

}

/\* img {

  display: block;

  width: 100%;

  height: auto;

} \*/

.table {

  margin-top: 1%;

  margin-bottom: 1%;

  border-collapse: collapse;

  /\* width: 500px; \*/

  /\* background-color: red; \*/

  /\* font-size: 500px;

  table-layout: fixed; \*/

}

td, th {

  padding: 2px;

}