АННОТАЦИЯ

Дипломный проект на тему «Мобильное приложение для генерации рецептов SmartChef» включает в себя программное средство, пояснительную записку и графическую часть.

Разработанное программное средство предназначено для генерации рецептов на основе фото продуктов с использованием нейросетевых технологий. Приложение позволяет пользователю загружать фотографии продуктов, которые затем распознаются, а на основе этого генерируются рецепты с подробным описанием ингредиентов, КБЖУ и шагов приготовления. Приложение предоставляет интуитивно понятный интерфейс для пользователей, позволяя улучшить их кулинарные навыки и рацион.

Программная часть реализована с использованием современных технологий: клиентская часть построена на фреймворке Kotlin с использованием Jetpack Compose для интерфейса, а для распознавания продуктов и генерации рецептов используется Gemini API. Для хранения данных используется облачная платформа Firebase, обеспечивающая надёжную и масштабируемую инфраструктуру для хранения информации о пользователях, рецептах и предпочтениях. В приложении реализованы функции поиска рецептов, сохранения и сортировки по ингредиентам и диетическим предпочтениям.

Пояснительная записка включает подробное описание архитектуры приложения, обоснование выбора технологий, описание пользовательских сценариев, а также результаты тестирования. Общий объём пояснительной записки составляет 146 страниц и включает четыре основных раздела и два приложения.

Графическая часть проекта представлена на двух листах и содержит диаграмму последовательности и диаграмму вариантов использования, иллюстрирующие основные процессы взаимодействия пользователя с системой.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc199226234)

[1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ 5](#_Toc199226235)

[2. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ 7](#_Toc199226236)

[2.1. Постановка задачи 7](#_Toc199226237)

[2.2. Описание организации данных 7](#_Toc199226238)

[2.3. Описание программных средств 9](#_Toc199226239)

[2.4. Проектирование интерфейса приложения 11](#_Toc199226240)

[2.5. Описание физической структуры сайта 19](#_Toc199226241)

[3. ТЕСТИРОВАНИЕ И АНАЛИЗ ПОЛУЧЕННЫХ РЕЗУЛЬТАТОВ 20](#_Toc199226242)

[4. РУКОВОДСТВО ПО ИСПОЛЬЗОВАНИЮ ПРОГРАММНОГО СРЕДСТВА 28](#_Toc199226243)

[5. ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ РАЗРАБОТКИ ПРОГРАММНОГО СРЕДСТВА 33](#_Toc199226244)

[6. ОХРАНА ТРУДА 40](#_Toc199226245)

[6.1. Правовые и организационные вопросы охраны труда 40](#_Toc199226246)

[6.2. Производственная санитария и гигиена труда 41](#_Toc199226247)

[6.3. Техника безопасности (электробезопасность) 43](#_Toc199226248)

[6.4. Пожарная безопасность 44](#_Toc199226249)

[ЗАКЛЮЧЕНИЕ 47](#_Toc199226250)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 49](#_Toc199226251)

[ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ 51](#_Toc199226252)

[ПРИЛОЖЕНИЕ 1 (ОБЯЗАТЕЛЬНОЕ) – ЛИСТИНГ ПРОГРАММЫ 52](#_Toc199226253)

# 

# ВВЕДЕНИЕ

В современном мире стремительное развитие мобильных технологий и искусственного интеллекта оказывает значительное влияние на повседневную жизнь и подходы к кулинарии. Всё больше людей ищут способы сделать процесс приготовления пищи более осознанным и удобным, используя современные инструменты для поиска рецептов и планирования меню. В этих условиях особую актуальность приобретают мобильные приложения, которые позволяют пользователям эффективно генерировать рецепты на основе имеющихся продуктов и получать персонализированные рекомендации.

Данный дипломный проект посвящён разработке мобильного приложения «SmartChef» — виртуального помощника для генерации рецептов на основе фотографий продуктов. Приложение ориентировано на широкий круг пользователей, желающих легко и быстро находить рецепты, используя фотографии своих продуктов. Приложение использует технологии искусственного интеллекта для распознавания продуктов и генерации рецептов с подробным описанием ингредиентов, калорийности и шагов приготовления. Особое внимание уделено удобству интерфейса, визуализации рецептов и автоматизации процесса поиска и создания блюд, что позволяет сделать процесс приготовления пищи простым и увлекательным.

В условиях растущей популярности умных технологий для улучшения качества жизни, SmartChef предоставляет пользователям возможность генерировать рецепты, основываясь на их предпочтениях и наличии продуктов. Приложение реализует функции поиска рецептов, сохранения их в избранное, а также сортировки по типу блюда, ингредиентам и диетическим предпочтениям.

Кроме того, проект демонстрирует современные подходы к разработке мобильных приложений, использованию API для распознавания объектов и генерации контента, а также эффективному взаимодействию с базой данных для хранения информации о рецептах. В ходе работы были применены такие инструменты, как Kotlin для разработки Android-приложения и Gemini API для распознавания продуктов и генерации рецептов.

Таким образом, создание мобильного приложения SmartChef является актуальным и востребованным проектом, отвечающим современным тенденциям цифровизации в сфере кулинарии и растущим потребностям пользователей в удобных и персонализированных решениях для приготовления пищи. В последующих разделах подробно рассматриваются этапы проектирования, реализации и тестирования приложения, а также вопросы обеспечения безопасности и удобства работы пользователей.

# 1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

Web-приложение «SmartChef» разработано с целью предоставления пользователям современного инструмента для планирования и приготовления пищи на основе имеющихся продуктов. Приложение предназначено для поиска рецептов, формирования меню, а также для учета и планирования ингредиентов. Основная аудитория SmartChef — это люди, стремящиеся к удобству в приготовлении пищи, а также те, кто хочет рационально использовать продукты и избегать их излишков.

Приложение может использоваться как в домашних условиях, так и на кухне, помогая пользователям оптимизировать процесс приготовления пищи и рационально подходить к выбору продуктов. Благодаря мобильной платформе, SmartChef доступно на смартфонах и планшетах на базе Android, что обеспечивает удобство использования на ходу.

Ключевые преимущества приложения SmartChef:

1. Интуитивно понятный и современный интерфейс, обеспечивающий быстрый доступ ко всем функциям.
2. Генерация рецептов на основе фотографий продуктов с использованием технологий искусственного интеллекта.
3. Возможность сортировки рецептов по типу блюда, ингредиентам и диетическим предпочтениям.
4. Поддержка персонализированных рекомендаций по составлению меню.
5. Поддержка русского языка и адаптация интерфейса под разные размеры экранов.

Требования к аппаратным и программным ресурсам:

* Операционная система Android версии 8.0 и выше.
* Минимум 2 ГБ оперативной памяти.
* Стабильное подключение к интернету для синхронизации данных и работы с сервером.

В процессе разработки приложения SmartChef был проведён анализ существующих решений в области поиска рецептов и планирования питания, таких как Yummly, Tasty, Cookpad и других. Были выявлены их сильные и слабые стороны, что позволило определить уникальные преимущества SmartChef:

Достоинства конкурентов:

* Широкий выбор рецептов и функциональность для сохранения рецептов в избранное.
* Возможность фильтрации рецептов по типу блюда и ингредиентам.
* Интеграция с магазинами продуктов для заказа ингредиентов.

Недостатки конкурентов:

* Сложность в поиске рецептов на основе имеющихся продуктов.
* Отсутствие персонализированных рекомендаций и учета предпочтений пользователя.
* Неоптимизированный интерфейс для пользователей с ограниченным опытом в приготовлении пищи.

Уникальные преимущества SmartChef:

* Генерация рецептов на основе фотографий продуктов с использованием искусственного интеллекта.
* Простой и удобный интерфейс, подходящий для всех пользователей.
* Возможность фильтрации рецептов по ингредиентам, типу блюда и диетическим предпочтениям.
* Поддержка персонализированных рекомендаций и планирования меню на основе наличных продуктов.

Таким образом, SmartChef занимает уникальную нишу среди приложений для планирования питания, ориентируясь на инновационные подходы к использованию технологий распознавания изображений и обеспечивая простоту и удобство использования.

# 2. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

# 2.1. Постановка задачи

Web-приложение SmartChef предназначено для персонального контроля питания и генерации рецептов, основанных на данных о продуктах с использованием искусственного интеллекта. Приложение предоставляет пользователю следующие основные функции:

1. Регистрация и авторизация: создание личного профиля с возможностью указания индивидуальных параметров (возраст, вес, рост, цели, предпочтения в питании и т.д.).
2. Поиск рецептов по фото: загрузка изображений продуктов для генерации рецептов, основанных на распознавании ингредиентов с использованием искусственного интеллекта.
3. Генерация персонализированных рецептов: автоматическое создание списка рецептов на основе предпочтений, целей пользователя и доступных ингредиентов.
4. Управление профилем: редактирование личной информации, настройка целей (например, потеря веса, улучшение питания).
5. Поддержка нескольких устройств: доступ к приложению на смартфонах, планшетах и компьютерах с современными браузерами.

Приложение реализовано на Android Studio с использованием языка Kotlin и интеграцией Gemini API для распознавания продуктов и генерации рецептов.

Нефункциональные требования:

1. Безопасность: защита персональных данных пользователей.
2. Удобный интерфейс: современный дизайн, адаптивная верстка для различных устройств.
3. Производительность: быстрая загрузка страниц, высокая отзывчивость интерфейса.
4. Кроссплатформенность: работа на устройствах с Android.

# 2.2. Описание организации данных

Данные приложения хранятся в реляционной базе данных, структура которой включает следующие основные таблицы:

* users — информация о пользователях;
* ingredients — ингредиенты рецептов;
* recipes — рецепты;
* user\_recipes — рецепты, сохраненные пользователем;

Ниже приведено описание структуры таблиц:

Таблица 2.1 – Структура данных таблицы «users»

|  |  |  |
| --- | --- | --- |
| Наименование столбца | Тип данных | Описание |
| id | Число | Уникальный идентификатор пользователя |
| name | Текст | |  | | --- | | Имя пользователя |  |  | | --- | |  | |
| email | Текст | |  | | --- | | Электронная почта |  |  | | --- | |  | |
| password\_hash | Текст | |  | | --- | | Хэш пароля |  |  | | --- | |  | |
| age | Число | |  | | --- | | Возраст пользователя |  |  | | --- | |  | |
| gender | Текст | |  | | --- | | Пол пользователя |  |  | | --- | |  | |
| created\_at | Дата/Время | Дата регистрации |

Таблица 2.2 – Структура данных таблицы «ingredients»

|  |  |  |
| --- | --- | --- |
| Наименование столбца | Тип данных | Описание |
| id | Число | Уникальный идентификатор пользователя |
| name | Текст | |  | | --- | | Название рецепта |  |  | | --- | |  | |
| description | Текст | |  |  |  | | --- | --- | --- | | |  | | --- | | Описание ингредиента |  |  | | --- | |  | |  |  | | --- | |  | |
| calories\_per\_100g | Число | |  |  |  | | --- | --- | --- | | |  | | --- | | Калории на 100 г ингредиента |  |  | | --- | |  | |  |  | | --- | |  | |
| proteins\_per\_100g | Число | |  |  |  | | --- | --- | --- | | |  | | --- | | Белки на 100 г ингредиента |  |  | | --- | |  | |  |  | | --- | |  | |
| fats\_per\_100g | Число | |  | | --- | | Жиры на 100 г ингредиента |  |  | | --- | |  | |
| carbohydrates\_per\_100g | Число | Углеводы на 100 г ингредиента |

Таблица 2.3 – Структура данных таблицы «recipes»

|  |  |  |
| --- | --- | --- |
| Наименование столбца | Тип данных | Описание |
| id | Число | Уникальный идентификатор рецепта |
| name | Текст | Название рецепта |
| description | Текст | |  | | --- | | Описание рецепта |  |  | | --- | |  | |
| cooking\_time | Число | |  | | --- | | Время приготовления (в минутах) |  |  | | --- | |  | |
| instructions | Текст | |  | | --- | | Инструкции по приготовлению |  |  | | --- | |  | |
| image\_url | Текст | Ссылка на изображение рецепта |

Таблица 2.4 – Структура данных таблицы «user\_recipes»

|  |  |  |
| --- | --- | --- |
| Наименование столбца | Тип данных | Описание |
| id | Число | Уникальный идентификатор записи рецепта |
| user\_id | Число | Идентификатор пользователя |
| recipe\_id | Число | Идентификатор рецепта |
| date\_added | Дата | Дата добавления рецепта пользователем |

Входные данные программы:

* логин, пароль, email пользователя
* ингредиенты, загруженные пользователем для поиска рецептов
* предпочтения пользователя для генерации рецептов (например, диетические ограничения, предпочтения по кухне)
* данные о рецептах (название, ингредиенты, шаги приготовления)

Выходные данные программы:

* генерируемые рецепты на основе загруженных ингредиентов
* графики и отчёты о рецептах, сохранённых пользователем
* история изменений в сохранённых рецептах
* уведомления и напоминания о новых рецептах, сохранённых пользователем

# 2.3. Описание программных средств

Программное средство Nutrio разработано с использованием современных технологий и инструментов, обеспечивающих кроссплатформенность, масштабируемость и удобство для пользователя.

Серверная часть реализована на платформе Java Spring Boot. Этот фреймворк обеспечивает высокую производительность, безопасность и гибкость при построении RESTful API, а также хорошо подходит для интеграции с модулями искусственного интеллекта и работы с реляционными базами данных.

Клиентская часть приложения разработана с использованием Flutter — кроссплатформенного фреймворка, позволяющего создавать современные мобильные и web-интерфейсы с единой кодовой базой. Flutter обеспечивает высокую скорость разработки, адаптивность интерфейса и отличную производительность на различных устройствах.

База данных приложения спроектирована и реализована с использованием облачной платформы Firebase, которая предоставляет гибкое и масштабируемое решение для хранения и синхронизации данных в режиме реального времени. Для администрирования и визуализации структуры данных используется web-интерфейс Firebase Console, а также инструменты аналитики и контроля доступа, встроенные в Firebase.

Краткая характеристика используемых технологий

Java Spring Boot

Преимущества:

1. Высокая производительность и масштабируемость.
2. Богатый набор встроенных средств для безопасности, работы с БД, интеграции с внешними сервисами.
3. Активное сообщество и обширная документация.

Недостатки:

1. Более высокий порог вхождения для начинающих разработчиков.
2. Требует большего объёма ресурсов по сравнению с легковесными решениями.

Flutter

Преимущества:

1. Кроссплатформенность: единая кодовая база для Android, iOS и Web.
2. Быстрая разработка благодаря hot reload и большому количеству готовых виджетов.
3. Современный и гибкий UI.

Недостатки:

1. Большой размер итогового приложения.
2. Некоторые ограничения по нативным возможностям на web-платформе.

Firebase

Преимущества:

1. Облачное хранение данных без необходимости настройки серверной инфраструктуры.
2. Синхронизация данных в реальном времени между клиентами.
3. Встроенные механизмы аутентификации, аналитики и безопасности.

Недостатки:

1. Возможные ограничения по сложным SQL-запросам.
2. Зависимость от интернет-соединения и внешнего облачного сервиса.

Firebase Console

Преимущества:

1. Удобный web-интерфейс для управления базой данных, пользователями и хранилищем.
2. Гибкая настройка прав доступа, мониторинг и логирование.

Недостатки:

1. Требует подключения к интернету.
2. Ограниченные возможности в части ручного SQL-запроса и сложной бизнес-логики по сравнению с традиционными СУБД.

Важной особенностью мобильного приложения SmartChef является интеграция модуля искусственного интеллекта, который на основе распознанных продуктов с фотографии формирует персонализированные рецепты. Для этого используются современные алгоритмы обработки изображений и генерации текстового контента, что позволяет учитывать предпочтения пользователя и создавать блюда с точным расчётом ингредиентов, КБЖУ и пошаговыми инструкциями. Такой подход делает кулинарный процесс более осознанным, удобным и адаптированным под конкретные потребности пользователя.

# 2.4. Проектирование интерфейса приложения

В данном проекте пользовательская часть Web-приложение будет представлена пятью основными страницами: страница авторизации, страница регистрации, главная страница, страница рецепта, страница профиля.

На странице авторизации пользователь может авторизоваться в свой аккаунт. Страница «Авторизация» изображена на картинке 2.1. Элементы управления страницей «Авторизация» отображено в таблице 2.8.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рис.2.1. Страница «Авторизация»

Таблица 2.8 – Элементы управления страницей «Авторизация»

|  |  |  |
| --- | --- | --- |
| Элементы управления | Свойства | Назначение |
| Input1 | Class=input | Поле ввода адреса электронной почты пользователя |
| Input2 | Class=input | Поле ввода пароля пользователя |
| Button1 | Class=button | Кнопка для входа в аккаунт |
| Button2 | Class=button | Кнопка для перехода на страницу «Регистрации» |

На странице «Регистрация» пользователь ведет выбирание им ранее товары, показано на рисунке 2.2. Элементы управления страницей «Регистрация» отображены в таблице 2.9.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рис.2.2. Страница «Регистрация»

Таблица 2.9 – Элементы управления страницей «Регистрация»

|  |  |  |
| --- | --- | --- |
| Элементы управления | Свойства | Назначение |
| Input1 | Class=input | Поле ввода адреса электронной почты пользователя |
| Input2 | Class=input | Поле ввода пароля |
| Input3 | Class=input | Поле подтверждения пороля |
| Button1 | Class=button | Кнопка для регистрации |
| Button2 | Class=button | Кнопка для перехода на страницу «Авторизации» |

На странице «Главная» рисунок 2.3, пользователь может перейти на любую страницу приложения. Элементы страницы «Главная» представлены в таблице 2.9.

Изображение выглядит как текст, снимок экрана, диаграмма, шаблон

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рис.2.3. Страница «Главная»

Таблица 2.9 – Элементы управления страница «Главная»

|  |  |  |
| --- | --- | --- |
| Элементы управления | Свойства | Назначение |
| Image1 | Class=image | Фотография, которую загрузил пользователь |
| Button1-3 | Class=button | Кнопки для перехода на страницу «Рецепт» |
| Button4 | Class=button | Кнопка перехода на страницу «Главная» |
| Button5 | Class=button | Кнопка для перехода на камеру |
| Button6 | Class=button | Кнопка для перехода на галерею |
| Button7 | Class=button | Кнопка перехода на страницу «Профиль» |

На странице «Профиль» рисунок 2.4, пользователь видит данные своего профиля и избранные рецепты. Элементы страницы «Профиль» показаны в таблице 2.10. Элементы страницы «Описания товара» показаны в таблице 2.11.

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 2.4 Страница «Профиль»

Таблица №2.10 – Элементы страницы «Профиль»

|  |  |  |
| --- | --- | --- |
| Элементы управления | Свойства | Назначение |
| Button1 | Class=button | Кнопка для перехода на страницу «Главная» | |
| Button2 | Class=button | Кнопка выхода из аккаунта | |
| Button3-6 | Class=button | Кнопка перехода на страницу «Избранный рецепт» | |

На странице «План питания» рисунок 2.5, пользователь может создать свой план питания. Элементы страницы «План питания» показаны в таблице 2.11.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 2.5 Страница «Рецепт»

Таблица №2.11 – Элементы страницы «План питания»

|  |  |  |
| --- | --- | --- |
| Элементы управления | Свойства | Назначение |
| Button1 | Class=button | Кнопка для перехода на страницу «Главная» |
| Button2 | Class=button | Кнопка избранного |
| Checkbox1-3 | Class=checkbox | Переключатель для шагов приготовления |

# 2.5. Описание физической структуры сайта

Программа содержит следующие подкаталоги: lib (основной подкаталог), assets (подкаталог ресурсов), test (подкаталог тестов), основной файл main.dart

Подкаталог lib хранит следующие файлы:

1) models/auth\_user.dart – файл модели пользователя для аутентификации.

2) models/day\_meal\_plan.dart – файл модели дневного плана питания.

3) models/meal.dart – файл модели приема пищи.

4) screens/register\_screen.dart – файл экрана регистрации новых пользователей.

5) screens/login\_screen.dart – файл экрана входа в систему.

6) screens/home\_screen.dart – файл главного экрана приложения.

7) screens/meal\_plan\_screen.dart – файл экрана плана питания.

8) screens/profile\_screen.dart – файл экрана профиля пользователя.

9) services/api\_service.dart – файл сервиса для работы с API.

10) services/auth\_service.dart – файл сервиса аутентификации.

11) services/meal\_plan\_service.dart – файл сервиса для работы с планами питания.

12) widgets/meal\_card.dart – файл виджета карточки приема пищи.

13) widgets/nutrition\_info.dart – файл виджета с информацией о питании.

14) widgets/custom\_button.dart – файл кастомной кнопки.

15) app.dart – файл корневого виджета приложения.

16) routes.dart – файл конфигурации маршрутизации.

17) theme.dart – файл темы приложения.

18) main.dart – файл точки входа в приложение.

19) constants.dart – файл с константами приложения.

20) utils/validators.dart – файл с валидаторами форм.

21) utils/helpers.dart – файл с вспомогательными функциями.

22) utils/extensions.dart – файл с расширениями для классов.

# 3. ТЕСТИРОВАНИЕ И АНАЛИЗ ПОЛУЧЕННЫХ РЕЗУЛЬТАТОВ

Тестирование программного обеспечения – процесс исследования, испытания программного продукта, имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением на конечном наборе тестов, выбранных определённым образом.

Метод «черного ящика». Основан на принципе «вход-выход». Программе подаются некоторые данные на вход и проверяются результаты, в надежде найти несоответствия. При этом как именно работает программа, считается несущественным.

Этапы процесса тестирования:

1. Проверка приложения в нормальных условиях. Представляет собой тестирование с исходными данными, требуемыми для нормального функционирования программы.
2. Проверка в экстремальных условиях. Представляет собой тестирование программы с граничными и неверными исходными данными. Программа должна работать с граничными исходными данными и отвергать неверные исходные данные.

В проекте использовался метод тестирования «черного ящика». Программа была протестирована на следующих тестах:

1) Тест №1. Проверка на запуск приложения.

2) Тест №2. Проверка авторизации с корректными данными.

3) Тест №3. Проверка открытия камеры.

4) Тест №4. Проверка открытия окна загрузки фотографий.

5) Тест №5. Проверка открытия профиля.

Тест №1. При запуске приложения загружается страница авторизации, что видно на рисунке 3.1.

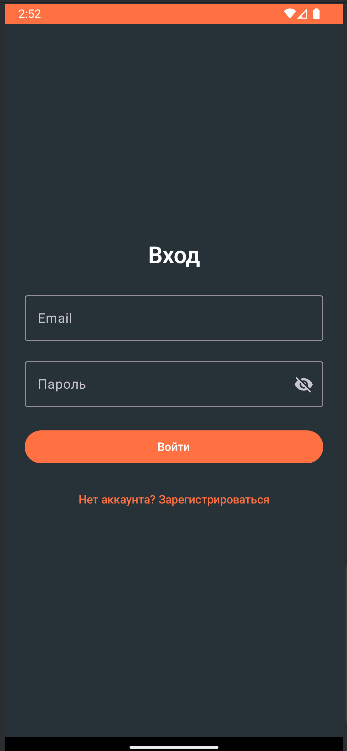


Рис. 3.1. Страница авторизации

Тест №2. Во время нажатия кнопки «Войти» при правильно заполненных полях, программа авторизует пользователя в систему и войдет в его аккаунт, что показано на рисунках 3.2., 3.3.

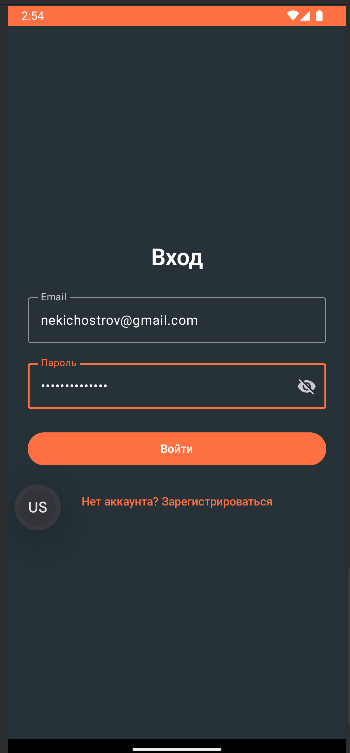


Рис. 3.2. Авторизация пользователя

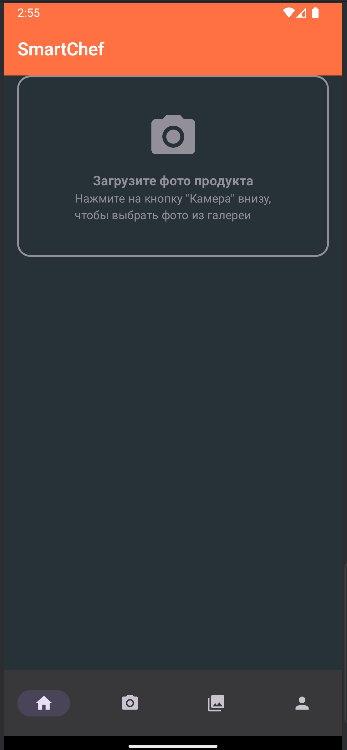


Рис. 3.3. Авторизация пользователя

Тест №3. При нажатии на иконку камеры, открывается камера, что показано на рисунке 3.4.



Рис. 3.4. Открытие камеры

Тест №4. При нажатии на иконку картинок открывается окно загрузки фотографий, что представлено на рисунке 3.5.

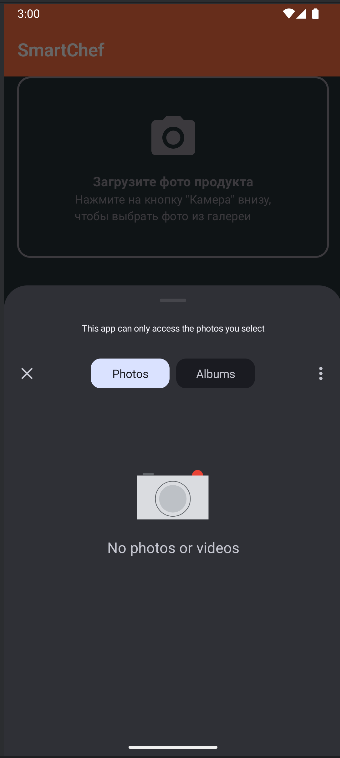


Рис. 3.5. Окно загрузки фотографий

Тест №5. При нажатии на иконку профиля открывается профиль пользователя, что представлено на рисунке 3.6.

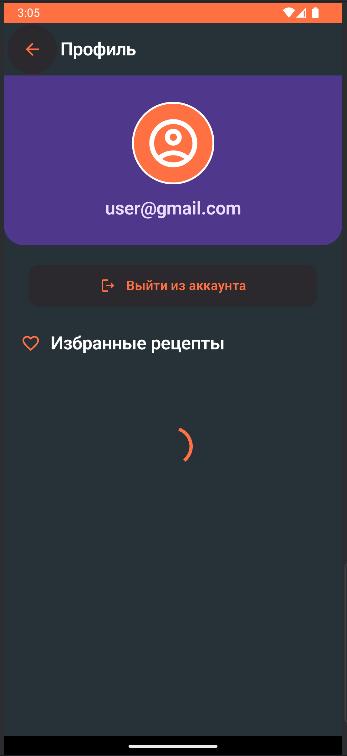


Рис. 3.6. Профиль пользователя

# 4. РУКОВОДСТВО ПО ИСПОЛЬЗОВАНИЮ ПРОГРАММНОГО СРЕДСТВА

Для того, чтобы зайти в приложение необходимо нажать на соответствующую иконку, как указано на рисунке 4.1.



Рисунок 4.1. Иконка приложения

После чего пользователь попадает на страницу «Авторизации», где может авторизоваться, либо перейдя на страницу «Регистрация», зарегистрировать себе новый аккаунт, что показано на рисунке 4.2.

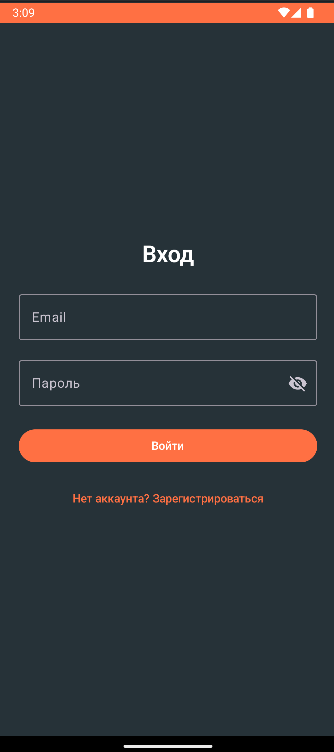


Рисунок 4.2. Страница авторизации

После успешной авторизации, пользователь попадает на главную страницу, которая показана на рисунке 4.3. На главной странице пользователь может перейти на страницы «Камеры», «Загрузки фото», «Профиля», нажав на соответствующий пункт, после чего, приложение направит его на выбранную страницу, которые представлены на рисунках 4.4., 4.5, 4.6.



Рисунок 4.3 Главная страница



Рисунок 4.4 Страница камеры

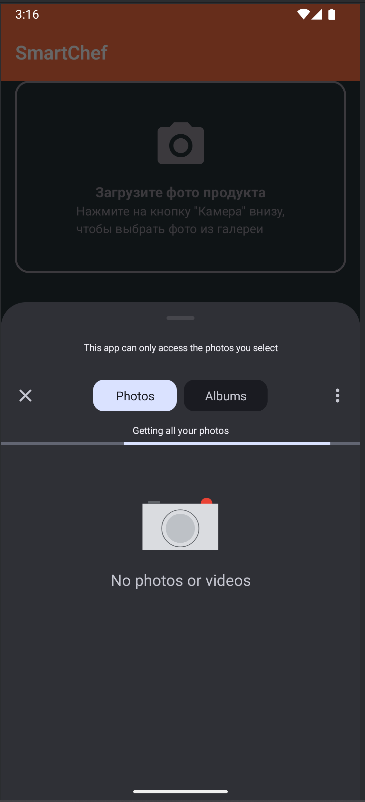


Рисунок 4.5 Страница загрузки фото

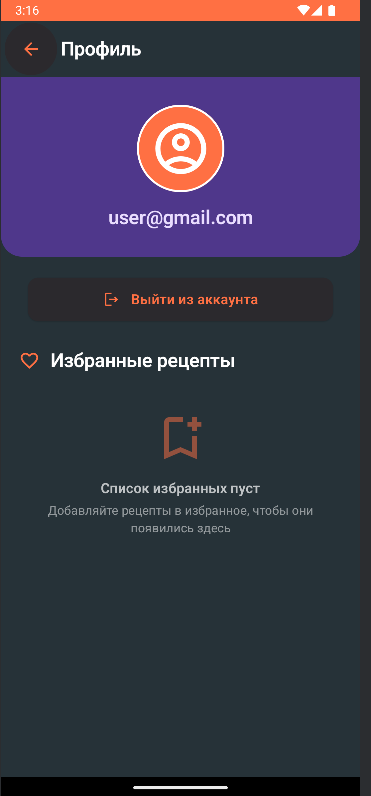


Рисунок 4.6 Страница профиля

# 5. ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ РАЗРАБОТКИ ПРОГРАММНОГО СРЕДСТВА

В рыночных условиях программное средство преимущественно представлено в виде продукции научно-технических организаций, представляющей собой функционально завершенные и имеющие товарный вид программные продукты, поставляемые заказчикам и продаваемые покупателям по рыночным ценам. Все завершенные разработки программного обеспечения являются научно-технической продукцией.

Стоимостная оценка программного обеспечения (ПО) и определение экономического эффекта у разработчика предполагают составление сметы затрат, которая в денежном выражении включает следующие статьи расходов:

* заработную плату исполнителей, основную (Зо) и дополнительную (Зд);
* отчисления на социальные нужды (Р соц.);
* налоги, от фонда оплаты труда (Не);
* материалы и комплектующие (М);
* спецоборудование (Рc);
* машинное время (Рм);
* расходы на научные командировки (Рнк);
* прочие прямые расходы (Пз);
* накладные расходы (Рн).

На основе общей суммы расходов по всем статьям (Ср) и результатов маркетинговых исследований на рынке ПО определяется плановая отпускная цена (Цo) с учётом прибыли (рентабельности) и налогов, включаемых в цену.

По уточненному объёму ПО и нормативам затрат труда в расчёте на единицу объема определяются нормативная и общая трудоемкость разработки ПО.

Нормативная трудоёмкость разработки ПО (ТЕн) определяется на основании принятого к расчёту объема (Vy) и категории сложности ПО (чел./дн.), указанных в приложении 1, таблице1.1, уточняется с учётом сложности и новизны проекта, степени использования стандартных модулей при разработке. Нормативная трудоёмкость разработки ПО служит основой для определения общей трудоёмкости.

Общая трудоёмкость разработки ПО (ТЕр) определяется по формуле (5.1):

TЕр = TЕн × Kc × Kн × Kт × Кур, (5.1)

где ТЕн – нормативная трудоёмкость разработки программного средства, определяемая относительно общего количества строк исходного кода;

Кс – коэффициент сложности;

Кн – коэффициент, учитывающий степень новизны;

Кт – коэффициент, учитывающий использование стандартных модулей;

Кур – коэффициент, учитывающий средства разработки.

Категория сложности ПO определяется в зависимости от наличия (отсутствия) следующих характеристик:

* + высокий уровень языкового интерфейса с пользователем;
  + режим работы в реальном времени;
  + управление удаленными объектами;
  + машинная графика, многомашинные комплексы;
  + существенное распараллеливание вычислений;
  + нестандартная конфигурация технических средств;
  + оптимизационные и особо сложные инженерные и научные расчёты;
  + переносимость ПО.

Влияние фактора сложности на трудоемкость учитывается умножением нормативной трудоемкости на соответствующий коэффициент сложности.

Коэффициент сложности (Kc) учитывает дополнительные затраты труда, связанные со сложностью разрабатываемого программного продукта, при расчёте которого необходимо воспользоваться дополнительными коэффициентами сложности ПО, указанными в приложении 1, таблице 1.3.

Коэффициент, учитывающий степень использования при разработке ПО стандартных модулей (Kт) определяется их удельным весом в общем объеме проектируемого продукта (Приложение 1, таблица 1.3). При определении влияния этого фактора на трудоемкость он учитывается путем умножения нормативной трудоемкости на соответствующий коэффициент.

Коэффициент новизны разрабатываемого ПО (Kн) определяется путём сравнения характеристик разрабатываемого ПО с имеющимися аналогами, позволяет определить экспертным путем степень его новизны. Влияние фактора новизны на трудоемкость учитывается путем умножения трудоемкости на соответствующий коэффициент новизны. Для расчета данного показателя используются поправочные коэффициенты, учитывающие новизну ПО (Kн), указанные в приложении 1, таблице 1.4.

TЕp = 26 × 0,26 × 1,0 × 0,65 × 1,3 = 6 чел./дн.

Основной статьей расходов на создание ПО является заработная плата разработчиков (исполнителей) проекта, в число которых принято включать инженеров-программистов, участвующих в написании кода, руководителей проекта, системных архитекторов, дизайнеров, разрабатывающих пользовательский интерфейс, разработчиков баз данных, Web-мастеров и других специалистов, необходимых для решения специальных задач в команде. Заработная плата руководителей организации и работников вспомогательных служб (инфраструктуры) учитывается в накладных расходах.

В соответствии со статьями 1 и 63 Трудового кодекса Республики Беларусь оплата труда работника включает в себя тарифную ставку (тарифный оклад), стимулирующие (надбавки, премии, бонусы и др.) и компенсирующие выплаты. При этом конкретные формы, системы и размеры оплаты труда работников устанавливаются нанимателем на основании коллективного договора, соглашения, иных локальных правовых актов и трудового договора. В рамках данного дипломного проекта применяется стандартизированная методика калькуляции трудовых затрат, основанная на следующих принципах:

1. В состав расходов на оплату труда включены:

- основная заработная плата, начисляемая по тарифной системе;

- стимулирующие выплаты (премиальная составляющая);

- оплата фактически отработанного времени.

2. Нормативная база расчета:

- единицей измерения принят человеко-день;

- расчетные нормы установлены для одного программного средства;

- базовый режим труда: 5-дневная рабочая неделя, 8-часовая продолжительность рабочего дня.

Данная методика соответствует принципам нормирования труда в ИТ-отрасли и позволяет осуществить корректную оценку трудозатрат на разработку программного средства.

Основная заработная плата исполнителей на конкретное программное обеспечение рассчитывается по формуле (5.2):

, (5.2)

где ТСч – часовая тарифная ставка соответствующего разряда, руб.;

ТЕр – трудоемкость разработки, часов;

Кпр – коэффициент премирования.

Расчёт дополнительной заработной платы исполнителей за конкретное ПО (ЗПдоп) включает выплаты, предусмотренные законодательством о труде (оплата отпусков, льготных часов, времени выполнения государственных обязанностей и других выплат, не связанных с основной деятельностью исполнителей), и определяется по нормативу в процентах к основной заработной плате, согласно формуле (5.3):

ЗПдоп=(ЗПосн×%ЗПдоп)/100, (5.3)

где ЗПосн – основная заработная плата исполнителя, руб.;

% ЗПдоп – процент дополнительной заработной платы, %.

Обязательным условием является определение отчислений на социальные нужды (Рсоц) согласно формуле (5.4):

, (5.4)

где ЗПОСН – основная заработная плата исполнителей, руб.;

ЗПДОП – дополнительная заработная плата исполнителей, руб.;

Нсоц– норматив отчислений в ФСЗН и Белгосстрах (в соответствии с действующим законодательством)

Расходы на материалы и комплектующие определяются на основании сметы затрат, разрабатываемой на программное обеспечение с учетом действующих нормативов. В данные затраты включают расходы на магнитные носители, бумагу, красящие ленты и другие материалы, необходимые для разработки программного обеспечения. Нормы расхода материалов в суммарном выражении определяются в расчете на 100 строк исходного кода или по нормативу в процентах основной заработной платы разработчиков, который устанавливается организацией.

Сумма затрат на расходные материалы рассчитывается по формуле (5.5):

, (5.5)

где ЗПОСН – основная заработная плата исполнителей, руб.;

Нм – норматив расхода на материалы, %.

Расходы по статье «Машинное время» включают оплату машинного времени, необходимого для разработки и отладки ПО, которое определяется по нормативам (в машино-часах) на 100 строк исходного кода машинного времени в зависимости от характера решаемых задач и типа ПК (0,6 машино-часов (5.6):

(5.6)

где Цм– цена одного машино-часа, руб.;

Vстрок– общий объем ПО (количество строк исходного кода);

Нмв– норматив расхода машинного времени на отладку 100 строк исходного кода (машино-часов).

.

Общехозяйственные расходы связанны с необходимостью содержания аппарата управления, вспомогательных хозяйств и опытных (экспериментальных) производств, а также с расходами на общехозяйственные нужды, и относятся на конкретное программное обеспечение по нормативу в процентном отношении к основной заработной плате исполнителей. Норматив устанавливается в целом по организации (30-80 %) и рассчитывается по формуле (5.7):

, (5.7)

где % ОХР – норматив общехозяйственных расходов, %.

Прочие прямые затраты – это затраты на приобретение и подготовку специальной научно-технической информации и специальной литературы. Они определяются по нормативу, разрабатываемому в целом по организации, в процентах к основной заработной плате и рассчитывается по формуле (5.8):

, (5.8)

где Нп.з. – норматив прочих прямых затрат, %.

Результаты расчёта затрат внесём в таблицу 5.1.

Таблица 5.1 – Расчет полной себестоимости программного обеспечения

|  |  |
| --- | --- |
| Наименование статей затрат | Сумма, руб. |
| Материальные затраты | 34,53 |
| Основная заработная плата | 690,60 |
| Дополнительная заработная плата | 138,10 |
| Отчисления в ФСЗН и Белгосстрах | 281,86 |
| Машинное время | 33,00 |
| Общехозяйственные расходы | 241,70 |

*Продолжение таблицы 5.1*

|  |  |
| --- | --- |
| Наименование статей затрат | Сумма, руб. |
| Прочие прямые затраты | 69,06 |
| Полная себестоимость | 1488,85 |

Для определения цены ПО необходимо рассчитать плановую прибыль. Рентабельность и прибыль создаваемого ПП определяется исходя из результатов анализа рыночных условий переговоров с заказчиком и согласования с ним отпускной цены.

Прибыль рассчитывается по формуле (5.9):

(5.9)

где С/сПОЛН – полная себестоимость ПО, руб.;

Ппл – плановая прибыль от реализации ПО, руб.;

Р – уровень рентабельности ПО, %.

Отпускная цена без НДС определяется по формуле (5.10):

ЦОТП(без НДС)= С/сПОЛН. + ППЛ, (5.10)

ЦОТП(без НДС)= 1488,85+297,77 = 1786,62 руб.

Сумма НДС определяется по формуле (5.11):

(5.11)

где % НДС – ставка НДС, %.

Отпускная цена ПО с НДС(ЦОТП(с НДС) рассчитывается по формуле (5.12):

ЦОТП(с НДС**)**= ЦОТП **(**без НДС**)**+ НДС, (5.12)

где НДС – сумма налога на добавленную стоимость, руб.

ЦОТП(с НДС**)**= 1786, 62 + = 2143,94 руб.

Полученные результаты расчетов внесём в таблицу 5.2.

Таблица 5.2 – Расчет отпускной цены программного обеспечения

|  |  |
| --- | --- |
| Показатели | Сумма, руб. |
| Полная себестоимость | 1488,85 |
| Прибыль | 297,77 |
| Цена без НДС | 1786,62 |
| НДС | 357,32 |
| Отпускная цена с НДС | 2143,94 |

Экономическим эффектом от создания программного обеспечения для разработчика является чистая прибыль. Она определяется после уплаты налога на прибыль, формула (5.13):

, (5.13)

где %Нпр – ставка налога на прибыль, %.

.

Затраты на разработку мобильного приложения SmartChef составили 1488,85 руб. Отпускная прогнозируемая цена (с учётом НДС) составляет 2143,94 руб. Прибыль разработчика составит 297,77 руб., а чистая прибыль после уплаты налога на прибыль — 238,22 руб.

Рассчитанная цена реализуемого программного обеспечения позволяет полностью покрыть затраты на разработку и обеспечивает экономически целесообразную прибыль, что свидетельствует о целесообразности внедрения приложения SmartChef.

# 6. ОХРАНА ТРУДА

# 6.1. Правовые и организационные вопросы охраны труда

Данный дипломный проект может быть использован в деятельности как на предприятиях с большой численностью работников, так и индивидуальными предпринимателями. Для безопасной реализации дипломного проекта необходимо обратить внимание на вопросы охраны труда — системы обеспечения безопасности жизни и здоровья работников в процессе трудовой деятельности, включающей правовые, социально-экономические, организационно-технические, санитарно-гигиенические, лечебно-профилактические, реабилитационные и иные мероприятия и средства.

Общий контроль и руководство по соблюдению правил охраны труда в Минской ЦРКБ осуществляет главный инженер по охране труда А.Н. Карачун. Он координирует работу по проведению всех видов инструктажей, контролю за состоянием охраны труда в Минской ЦРКБ, планирует и проводит мероприятия, направленные на профилактику травматизма и профессиональных заболеваний.

Перед тем как вновь принятые сотрудники или практиканты в Минской ЦРКБ прибудут на свое рабочее место, с ними проводится вводный инструктаж. При проведении инструктажа были рассмотрены вопросы внутреннего трудового распорядка, прав и обязанностей работника и работодателя. Также сотрудники Минской ЦРКБ получили общую информации об организации и описание опасных и вредных факторов по предприятию в целом.

Со всеми вновь принятыми сотрудниками или практикантами в Минской ЦРКБ проводится первичный инструктаж, выделяется персональное рабочее место с установленными техническими средствами.

При проведении первичного инструктажа главным инженером по охране труда А.Н. Карачуном особое внимание уделялось самостоятельному подключению приборов к электросети. Было дополнительно подчёркнуто, что при пользовании средствами вычислительной техники и периферийным оборудованием каждый работник должен внимательно и осторожно обращаться с электропроводкой, приборами и аппаратами, и всегда помнить, что пренебрежение правилами безопасности угрожает и здоровью, и жизни человека.

В инструкции, с которой я был ознакомлен в первый рабочий день, указывалось о необходимости постоянного контроля на своем рабочем месте за исправным состоянием электропроводки, выключателей, штепсельных розеток, при помощи которых оборудование включается в сеть, и заземления. Делался упор на то, что при обнаружении неисправности необходимо немедленно обесточить электрооборудование и оповестить администрацию. Продолжение работы возможно только после устранения неисправности;

В общие правила, соблюдение которых является обязательным для всех работников Минской ЦРКБ, включены и те, которые касаются профилактики повреждения изоляции проводов и возникновения коротких замыканий, а именно не разрешается:

‒ вешать что-либо на провода;

‒ закрашивать и белить шнуры и провода;

‒ закладывать провода и шнуры за газовые и водопроводные трубы, за батареи отопительной системы;

‒ выдергивать штепсельную вилку из розетки за шнур, усилие должно быть приложено к корпусу вилки.

Среди основных правил, направленных на исключение поражения электрическим током, можно назвать следующие:

‒ запрещается часто включать и выключать компьютер без необходимости;

‒ запрещается прикасаться к экрану и к тыльной стороне блоков компьютера;

‒ не разрешается работать на средствах вычислительной техники и периферийном оборудовании мокрыми руками;

‒ не разрешается работать на средствах вычислительной техники и периферийном оборудовании, имеющих нарушения целостности корпуса, нарушения изоляции проводов, неисправную индикацию включения питания, с признаками электрического напряжения на корпусе

‒ запрещается класть на средства вычислительной техники и периферийном оборудовании посторонние предметы.

В действующих инструкциях Минской ЦРКБ сотруднику при работе за монитором рекомендуется раз в 45 минут делать зарядку для глаз.

При проверке соблюдения правил охраны труда указывалось на необходимость поддержания рабочего места и инвентаря в чистоте — одна из задач работника.

# 6.2. Производственная санитария и гигиена труда

В Минской ЦРКБ у каждого работника есть свое рабочее место, оборудованное техническими средствами: монитор, системный блок, клавиатура, мышь. Площадь помещения, выделенная одному работнику — 4,5 кв.м. Расстояние между столами сотрудников — около 2 метров. Каждый стол оснащен дополнительными преградами от проводов и удобным стулом с регуляторами высоты и длины сиденья.

В инструкциях Минской ЦРКБ указано, что работник обязан содержать в чистоте своё рабочее место, не допускать приёма пищи за рабочим столом, тщательно мыть руки, следить за порядком и не допускать скопления мусора.

Работа программиста в организациях и на предприятиях связана с вычислительными машинами, обработкой различных документов и информации, и подразумевает работу без физического напряжения, но под влиянием большого количества вредных производственных факторов.

При работе с персональным компьютером на программиста действуют следующие вредные факторы:

* ультрафиолетовое и инфракрасное излучения;
* электромагнитные излучения;
* рентгеновское излучение;
* статистическое электричество;
* блики и мерцание экрана;
* акустические шумы работающих принтеров, электродвигателей приводов винчестера, компакт – дисков, вентилятора процессора.

Кроме того, работа программиста за компьютером сопряжена с повышенными физическими и психологическими нагрузками, обусловленными следующими причинами:

* необходимость длительного напряжения, сохранение неподвижной позы во время работы. Длительная неподвижность приводит к усталости, болям в позвоночнике, шее, плечевых суставах;

− работа с клавиатурой вызывает боль в локтевых суставах, предплечьях, запястьях, кистях и пальцах рук;

* перегрузка зрительного аппарата.

Основная нагрузка на глаза обусловлена тем, что большая часть информации воспринимается визуально. Излишняя нагрузка на глаза может быть ослаблена правильным выбором монитора и его настройкой.

В Минской ЦРКБ реализуется комплекс мероприятий по снижению воздействия вредных факторов.

В рабочих кабинетах поддерживаются следующие показатели микроклимата:

− температура воздуха от плюс 190 до плюс 25°С;

− относительная влажность от 40 до 50%;

− скорость движения воздуха до 0,1 м/с;

− запыленность до 10 мг/м3.

Для борьбы с шумом применяются звукопоглощающие материалы.

Обязательно проводится проветривание помещений, так как в офисе становится очень душно, что может плохо сказаться на работоспособности, проветривание осуществляется три или четыре раза в течение рабочего дня.

Для создания благоприятных условий труда к организации рабочего места программиста клавиатура должны быть размещена на столе на высоте по отношению к полу 700 мм. Дисплей должен быть помещен на столе так, чтобы его экран располагался в вертикальной плоскости под углом менее 15 градусов. Во избежание перенапряжения органов зрения должно соблюдаться расстояние до монитора 40-50 см. Данные требования соблюдались мной на рабочем месте в Минской ЦРКБ.

В помещении, где находилось моё рабочее место имелось естественное и искусственное освещение, что помогает избежать проблемы со зрением. Освещенность на поверхности стола в зоне размещения рабочего документа составляла 300 – 500 люкс. Освещение не создавало бликов на поверхности экрана.

Для предотвращения утомления и повышения работоспособности был установлен рациональный режим труда и отдыха. В Минской ЦРКБ установлен регламентированный режим труда и отдыха. Предприятие начинает работу с 8:30 утра и заканчивает в 17:00. Обеденное время начинается с 13:00 и заканчивается в 13:30. Помимо обеденного перерыва сотрудник имеет право взять технологический перерыв: через каждый час работы введён перерыв на 5-10 минут, а через два часа – на 15 минут. Рабочая неделя является пятидневной, начало недели в понедельник и окончание в пятницу.

# 6.3. Техника безопасности (электробезопасность)

Место преддипломной практики относилось к помещению без повышенной опасности. Вся оргтехника Минской ЦРКБ подключена к источникам бесперебойного питания, что исключает потерю информации при отключении электричества.

Периодически проводится комплексный осмотр оборудования на внешние повреждения.

А также проводится перечень мероприятий для обеспечения электробезопасности:

1. Отключают напряжение и принимают меры, исключающие его ошибочную подачу к месту работы.
2. Вывешивают предупредительные плакаты на коммутационной аппаратуре, на постоянных и временных ограждениях.
3. Проверяют, есть ли напряжение на отключенной для работы части установки и накладывают на токоведущие части установки переносное заземление.

Нагрузка на электросеть в Минской ЦРКБ оптимизирована. Предусмотрены резервные генераторы, что обеспечивает стабильную работу даже в случае аварии. В экстренной ситуации:

* немедленно прекращается работа, выключается аппаратура, о всех недостатках сообщается должностному лицу;
* после окончания работы приостанавливается функционирование всех использованных приложений и программ.

При проведении первичного инструктажа особое внимание уделялось самостоятельному подключению приборов к электросети. Было дополнительно подчёркнуто, что при пользовании средствами вычислительной техники и периферийным оборудованием каждый работник должен внимательно и осторожно обращаться с электропроводкой, приборами и аппаратами и всегда помнить, что пренебрежение правилами безопасности угрожает и здоровью, и жизни человека.

Главный инженер охраны труда А.Н. Карачун, неоднократно указывалось на то, что необходимо постоянно следить на своем рабочем месте за исправным состоянием электропроводки, выключателей, штепсельных розеток, при помощи которых оборудование включается в сеть, и заземления. Делался упор на то, что при обнаружении неисправности немедленно обесточить электрооборудование, оповестить администрацию. Продолжение работы возможно только после устранения неисправности;

Во избежание повреждения изоляции проводов и возникновения коротких замыканий не разрешается:

* вешать что-либо на провода;
* закрашивать и белить шнуры и провода;
* закладывать провода и шнуры за газовые и водопроводные трубы, за батареи отопительной системы;
* выдергивать штепсельную вилку из розетки за шнур, усилие должно быть приложено к корпусу вилки.

Для исключения поражения электрическим током запрещается:

* часто включать и выключать компьютер без необходимости;
* прикасаться к экрану и к тыльной стороне блоков компьютера;
* работать на средствах вычислительной техники и периферийном оборудовании мокрыми руками;
* работать на средствах вычислительной техники и периферийном оборудовании, имеющих нарушения целостности корпуса, нарушения изоляции проводов, неисправную индикацию включения питания, с признаками электрического напряжения на корпусе.

# 6.4. Пожарная безопасность

Среди возможных причин возникновения пожара при эксплуатации ПЭВМ (возникновение аварийных ситуаций) можно назвать следующие:

* короткие замыкания;
* перегрузки;
* повышение переходных сопротивлений в электрических контактах;
* перенапряжение;
* возникновение токов утечки.

Для предотвращения возникновения и распространения пожара в Минской ЦРКБ реализуются следующие меры пожарной безопасности:

1) Для понижения воспламеняемости и способности распространять пламя кабели покрывают огнезащитным покрытием;

2) При ремонтно-профилактических работах строго соблюдаются правила пожарной безопасности;

3) Каждое из помещений, где производится эксплуатация устройств ПЭВМ, оборудовано первичными средствами пожаротушения и обеспечено инструкциями по их применению. В качестве средств пожаротушения в УП «КАНПЛАСТ» разрешается использование углекислотного огнетушителя типа ОУ-2, ОУ-5, а также порошковый тип. Применение пенных огнетушителей не допускается, так как жидкость пропускает ток;

4) Устройства ПЭВМ установлены вдали отопительных и нагревательных приборов (расстояние не менее 1 м и в местах, где не затруднена их вентиляция и нет прямых солнечных лучей);

5) Разработаны организационные меры по обучению персонала навыкам ликвидации пожара имеющимися в наличии средствами тушения пожара до прибытия пожарного подразделения.

К первичным средствам тушения небольших очагов возгорания в помещениях ЦРКБ относятся:

* пожарные стволы;
* внутренние пожарные водопроводы;
* огнетушители;
* сухой песок;
* асбестовые одеяла.

Для обеспечения готовности персонала к чрезвычайным ситуациям в здании разработаны и размещены планы эвакуации людей и материальных ценностей. Эти планы вывешены в каждом кабинете и коридоре учреждения. Также в каждом помещении установлены стенды с актуальной информацией по охране труда, пожарной безопасности, а также указаны ответственные лица.

В ЦРКБ ведётся систематическая работа по профилактике пожаров: своевременно удаляется горючий мусор, электроинструмент хранится в специально отведённых местах, осуществляется контроль за исправностью электросетей и оборудования.

В Минской ЦРКБ регулярно проводятся мероприятия, направленные на устранение потенциальных причин возгорания. В учреждениях осуществляется постоянная уборка помещений с удалением горючих отходов и материалов, способных вызвать возгорание. Электроинструмент и техническое оборудование хранятся в специально отведённых безопасных условиях, с соблюдением установленных норм.

В больнице внедрён комплекс организационно-технических мер, направленных на предотвращение пожаров: обустроены пути эвакуации для персонала и пациентов, обеспечен свободный доступ к выходам, размещены планы эвакуации на видимых местах, а также установлены стенды с памятками по оказанию первой медицинской помощи и действиям в случае пожара. Все помещения оснащены необходимыми средствами пожаротушения, обеспечивающими возможность оперативного реагирования до прибытия спасательных служб.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения дипломного проекта было разработано мобильное приложение «Nutrio», которое позволяет пользователям получать персонализированные планы питания и отслеживать свой рацион. Проект демонстрирует значительный потенциал в области улучшения качества жизни пользователей, предоставляя им инструмент для поддержания здорового образа жизни.

Разработанное веб-приложение успешно сочетает в себе удобство и функциональность, предлагая пользователям интуитивно понятный интерфейс и персонализированный подход к планированию питания. Интеграция системы учета индивидуальных параметров и предпочтений способствует созданию уникального опыта для каждого пользователя, что делает приложение «Nutrio» не просто инструментом для планирования питания, но и персональным помощником в достижении целей по здоровому образу жизни.

Программное средство разработано в среде программирования VS Code с использованием языка программирования Dart и фреймворка Flutter. База данных приложения спроектирована и реализована с использованием REST API, что обеспечивает надежное хранение и обработку данных пользователей.

Тестирование веб-приложения подтвердило его эффективность и надежность, а адаптивный дизайн обеспечивает удобство использования на различных устройствах. Экономическое обоснование разработки подчеркивает целесообразность вложений в проект, учитывая растущий спрос на приложения для здорового образа жизни.

Затраты на разработку веб-приложения «Nutrio» составили 3862,61 руб. Отпускная прогнозируемая цена составляет 3708,10руб. Прибыль разработчика составит 772,53 руб. Рассчитанная цена разработанного программного обеспечения позволит полностью покрыть все затраты.

В заключение, веб-приложение «Nutrio» является успешным примером инновационного подхода к разработке приложений в сфере здорового питания. Оно отвечает современным требованиям и ожиданиям пользователей, предлагая им не только удобный инструмент для планирования питания, но и возможность достижения своих целей в области здоровья и фитнеса. Результаты проекта могут быть использованы в качестве основы для дальнейшего развития и улучшения приложений, ориентированных на здоровый образ жизни и персонализированное питание.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Беляцкая, Т.Н. Предпринимательская деятельность и управление проектами в ИТ-сфере: учеб. пособие / Т.Н. Беляцкая. – Минск: РИПО, 2023. – 245 с.
2. Палицын, В. А. Технико-экономическое обоснование дипломных проектов: метод. пособие для студентов всех специальностей БГУИР: в 4 ч. Ч. 4 : Проекты программного обеспечения / В. А. Палицын. – Минск: БГУИР, 2006. – 76 с.
3. Хабр [Электронный ресурс] / SQL запросы быстро. Часть 1. Режим доступа: https://habr.com/ru/post/480838 – Дата доступа: 10.04.2025.
4. Auth0 [Электронный ресурс] / Введение в управление идентификацией и доступом (IAM). Режим доступа: Auth0.com – Дата доступа: 22.02.2025.
5. CometChat [Электронный ресурс] / Разработка чата: пошаговый процесс создания чата. Режим доступа: CometChat.com – Дата доступа: 27.02.2025.
6. freeCodeCamp [Электронный ресурс] / Как добавить функциональность поиска во фронтенд-приложение. Режим доступа: freeCodeCamp.org – Дата доступа: 27.02.2025.
7. Gknt.gov.by [Электронный ресурс] / Методические рекомендации. Режим доступа: http://www.gknt.gov.by:443/rules/metodicheskie-rekomendatsii/ Дата доступа: 29.02.2025.
8. Langformula [Электронный ресурс] / Упражнения по английскому языку. Режим доступа: https://langformula.ru/exercises – Дата доступа: 13.03.2025.
9. Metanit [Электронный ресурс] / Текстовое поле TextBox. Режим доступа: https://metanit.com/sharp/windowsforms/4.3.php – Дата доступа : 02.03.2025.
10. MouseDC [Электронный ресурс] / Удаление данных из базы (DELETE). Режим доступа: https://www.mousedc.ru/learning/66-sql-delete-update – Дата доступа: 25.03.2025.
11. News-newsby-org.narod.ru [Электронный ресурс] / Постановление Министерства труда и социальной защиты Республики Беларусь. Режим доступа: https://news-newsby-org.narod.ru/doc-razn/pos05/postn05236/index.htm Дата доступа: 30.02.2025.
12. Sling Academy [Электронный ресурс] / Как реализовать поиск по ключевым словам с использованием PHP и MySQL. Режим доступа: SlingAcademy.com – Дата доступа: 27.02.2025.
13. Unetway [Электронный ресурс] / SQL - Оператор LIKE. Режим доступа: https://unetway.com/tutorial/sql-operator-like – Дата доступа: 09.03.2025.

# ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ

БД – база данных

ПО – программное обеспечение

ДП – дипломный проект

ИИ – искусственный интеллект

СУБД – система управления базами данных

# ПРИЛОЖЕНИЕ 1 (ОБЯЗАТЕЛЬНОЕ) – ЛИСТИНГ ПРОГРАММЫ

<?php

define( 'WP\_USE\_THEMES', true );

require \_\_DIR\_\_ . '/wp-blog-header.php';

<?php

\* @package WordPress

define( 'WP\_INSTALLING', true );

require \_\_DIR\_\_ . '/wp-load.php';

require \_\_DIR\_\_ . '/wp-blog-header.php';

if ( ! is\_multisite() ) {

wp\_redirect( wp\_registration\_url() );

die();

}

$valid\_error\_codes = array( 'already\_active', 'blog\_taken' );

list( $activate\_path ) = explode( '?', wp\_unslash( $\_SERVER['REQUEST\_URI'] ) );

$activate\_cookie = 'wp-activate-' . COOKIEHASH;

$key = '';

$result = null;

if ( isset( $\_GET['key'] ) && isset( $\_POST['key'] ) && $\_GET['key'] !== $\_POST['key'] ) {

wp\_die( \_\_( 'A key value mismatch has been detected. Please follow the link provided in your activation email.' ), \_\_( 'An error occurred during the activation' ), 400 );

} elseif ( ! empty( $\_GET['key'] ) ) {

$key = $\_GET['key'];

} elseif ( ! empty( $\_POST['key'] ) ) {

$key = $\_POST['key'];

}

if ( $key ) {

$redirect\_url = remove\_query\_arg( 'key' );

if ( remove\_query\_arg( false ) !== $redirect\_url ) {

setcookie( $activate\_cookie, $key, 0, $activate\_path, COOKIE\_DOMAIN, is\_ssl(), true );

wp\_safe\_redirect( $redirect\_url );

exit;

} else {

$result = wpmu\_activate\_signup( $key );

}

}

if ( null === $result && isset( $\_COOKIE[ $activate\_cookie ] ) ) {

$key = $\_COOKIE[ $activate\_cookie ];

$result = wpmu\_activate\_signup( $key );

setcookie( $activate\_cookie, ' ', time() - YEAR\_IN\_SECONDS, $activate\_path, COOKIE\_DOMAIN, is\_ssl(), true );

}

if ( null === $result || ( is\_wp\_error( $result ) && 'invalid\_key' === $result->get\_error\_code() ) ) {

status\_header( 404 );

} elseif ( is\_wp\_error( $result ) ) {

$error\_code = $result->get\_error\_code();

if ( ! in\_array( $error\_code, $valid\_error\_codes, true ) ) {

status\_header( 400 );

}

}

nocache\_headers();

if ( is\_object( $wp\_object\_cache ) ) {

$wp\_object\_cache->cache\_enabled = false;

}

// Fix for page title.

$wp\_query->is\_404 = false;

do\_action( 'activate\_header' );

function do\_activate\_header() {

/\*\*

\* Fires within the `<head>` section of the Site Activation page.

\*

\* Fires on the {@see 'wp\_head'} action.

\*

\* @since 3.0.0

\*/

do\_action( 'activate\_wp\_head' );

}

add\_action( 'wp\_head', 'do\_activate\_header' );

function wpmu\_activate\_stylesheet() {

?>

<style type="text/css">

.wp-activate-container { width: 90%; margin: 0 auto; }

.wp-activate-container form { margin-top: 2em; }

#submit, #key { width: 100%; font-size: 24px; box-sizing: border-box; }

#language { margin-top: 0.5em; }

.wp-activate-container .error { background: #f66; color: #333; }

span.h3 { padding: 0 8px; font-size: 1.3em; font-weight: 600; }

</style>

<?php

}

add\_action( 'wp\_head', 'wpmu\_activate\_stylesheet' );

add\_action( 'wp\_head', 'wp\_strict\_cross\_origin\_referrer' );

add\_filter( 'wp\_robots', 'wp\_robots\_sensitive\_page' );

get\_header( 'wp-activate' );

$blog\_details = get\_site();

?>

<div id="signup-content" class="widecolumn">

<div class="wp-activate-container">

<?php if ( ! $key ) { ?>

<h2><?php \_e( 'Activation Key Required' ); ?></h2>

<form name="activateform" id="activateform" method="post" action="<?php echo esc\_url( network\_site\_url( $blog\_details->path . 'wp-activate.php' ) ); ?>">

<p>

<label for="key"><?php \_e( 'Activation Key:' ); ?></label>

<br /><input type="text" name="key" id="key" value="" size="50" autofocus="autofocus" />

</p>

<p class="submit">

<input id="submit" type="submit" name="Submit" class="submit" value="<?php esc\_attr\_e( 'Activate' ); ?>" />

</p>

</form>

<?php

} else {

if ( is\_wp\_error( $result ) && in\_array( $result->get\_error\_code(), $valid\_error\_codes, true ) ) {

$signup = $result->get\_error\_data();

?>

<h2><?php \_e( 'Your account is now active!' ); ?></h2>

<?php

echo '<p class="lead-in">';

if ( '' === $signup->domain . $signup->path ) {

printf(

/\* translators: 1: Login URL, 2: Username, 3: User email address, 4: Lost password URL. \*/

\_\_( 'Your account has been activated. You may now <a href="%1$s">log in</a> to the site using your chosen username of &#8220;%2$s&#8221;. Please check your email inbox at %3$s for your password and login instructions. If you do not receive an email, please check your junk or spam folder. If you still do not receive an email within an hour, you can <a href="%4$s">reset your password</a>.' ),

esc\_url( network\_site\_url( $blog\_details->path . 'wp-login.php', 'login' ) ),

esc\_html( $signup->user\_login ),

esc\_html( $signup->user\_email ),

esc\_url( wp\_lostpassword\_url() )

);

} else {

printf(

/\* translators: 1: Site URL, 2: Username, 3: User email address, 4: Lost password URL. \*/

\_\_( 'Your site at %1$s is active. You may now log in to your site using your chosen username of &#8220;%2$s&#8221;. Please check your email inbox at %3$s for your password and login instructions. If you do not receive an email, please check your junk or spam folder. If you still do not receive an email within an hour, you can <a href="%4$s">reset your password</a>.' ),

sprintf( '<a href="http://%1$s">%1$s</a>', esc\_url( $signup->domain . $blog\_details->path ) ),

esc\_html( $signup->user\_login ),

esc\_html( $signup->user\_email ),

esc\_url( wp\_lostpassword\_url() )

);

}

echo '</p>';

} elseif ( null === $result || is\_wp\_error( $result ) ) {

?>

<h2><?php \_e( 'An error occurred during the activation' ); ?></h2>

<?php if ( is\_wp\_error( $result ) ) : ?>

<p><?php echo esc\_html( $result->get\_error\_message() ); ?></p>

<?php endif; ?>

<?php

} else {

$url = isset( $result['blog\_id'] ) ? esc\_url( get\_home\_url( (int) $result['blog\_id'] ) ) : '';

$user = get\_userdata( (int) $result['user\_id'] );

?>

<h2><?php \_e( 'Your account is now active!' ); ?></h2>

<div id="signup-welcome">

<p><span class="h3"><?php \_e( 'Username:' ); ?></span> <?php echo esc\_html( $user->user\_login ); ?></p>

<p><span class="h3"><?php \_e( 'Password:' ); ?></span> <?php echo esc\_html( $result['password'] ); ?></p>

</div>

<?php

if ( $url && network\_home\_url( '', 'http' ) !== $url ) :

switch\_to\_blog( (int) $result['blog\_id'] );

$login\_url = wp\_login\_url();

restore\_current\_blog();

?>

<p class="view">

<?php

/\* translators: 1: Site URL, 2: Login URL. \*/

printf( \_\_( 'Your account is now activated. <a href="%1$s">View your site</a> or <a href="%2$s">Log in</a>' ), esc\_url( $url ), esc\_url( $login\_url ) );

?>

</p>

<?php else : ?>

<p class="view">

<?php

printf(

/\* translators: 1: Login URL, 2: Network home URL. \*/

\_\_( 'Your account is now activated. <a href="%1$s">Log in</a> or go back to the <a href="%2$s">homepage</a>.' ),

esc\_url( network\_site\_url( $blog\_details->path . 'wp-login.php', 'login' ) ),

esc\_url( network\_home\_url( $blog\_details->path ) )

);

?>

</p>

<?php

endif;

}

}

?>

</div>

</div>

<?php

get\_footer( 'wp-activate' );

<?php

if ( ! isset( $wp\_did\_header ) ) {

$wp\_did\_header = true;

// Load the WordPress library.

require\_once \_\_DIR\_\_ . '/wp-load.php';

// Set up the WordPress query.

wp();

// Load the theme template.

require\_once ABSPATH . WPINC . '/template-loader.php';

}

<?php

if ( 'POST' !== $\_SERVER['REQUEST\_METHOD'] ) {

$protocol = $\_SERVER['SERVER\_PROTOCOL'];

if ( ! in\_array( $protocol, array( 'HTTP/1.1', 'HTTP/2', 'HTTP/2.0', 'HTTP/3' ), true ) ) {

$protocol = 'HTTP/1.0';

}

header( 'Allow: POST' );

header( "$protocol 405 Method Not Allowed" );

header( 'Content-Type: text/plain' );

exit;

}

/\*\* Sets up the WordPress Environment. \*/

require \_\_DIR\_\_ . '/wp-load.php';

nocache\_headers();

$comment = wp\_handle\_comment\_submission( wp\_unslash( $\_POST ) );

if ( is\_wp\_error( $comment ) ) {

$data = (int) $comment->get\_error\_data();

if ( ! empty( $data ) ) {

wp\_die(

'<p>' . $comment->get\_error\_message() . '</p>',

\_\_( 'Comment Submission Failure' ),

array(

'response' => $data,

'back\_link' => true,

)

);

} else {

exit;

}

}

$user = wp\_get\_current\_user();

$cookies\_consent = ( isset( $\_POST['wp-comment-cookies-consent'] ) );

do\_action( 'set\_comment\_cookies', $comment, $user, $cookies\_consent );

$location = empty( $\_POST['redirect\_to'] ) ? get\_comment\_link( $comment ) : $\_POST['redirect\_to'] . '#comment-' . $comment->comment\_ID;

// If user didn't consent to cookies, add specific query arguments to display the awaiting moderation message.

if ( ! $cookies\_consent && 'unapproved' === wp\_get\_comment\_status( $comment ) && ! empty( $comment->comment\_author\_email ) ) {

$location = add\_query\_arg(

array(

'unapproved' => $comment->comment\_ID,

'moderation-hash' => wp\_hash( $comment->comment\_date\_gmt ),

),

$location

);

}

$location = apply\_filters( 'comment\_post\_redirect', $location, $comment );

wp\_safe\_redirect( $location );

exit;

ignore\_user\_abort( true );

if ( ! headers\_sent() ) {

header( 'Expires: Wed, 11 Jan 1984 05:00:00 GMT' );

header( 'Cache-Control: no-cache, must-revalidate, max-age=0' );

}

// Don't run cron until the request finishes, if possible.

if ( function\_exists( 'fastcgi\_finish\_request' ) ) {

fastcgi\_finish\_request();

} elseif ( function\_exists( 'litespeed\_finish\_request' ) ) {

litespeed\_finish\_request();

}

if ( ! empty( $\_POST ) || defined( 'DOING\_AJAX' ) || defined( 'DOING\_CRON' ) ) {

die();

define( 'DOING\_CRON', true );

if ( ! defined( 'ABSPATH' ) ) {

/\*\* Set up WordPress environment \*/

require\_once \_\_DIR\_\_ . '/wp-load.php';

}

function \_get\_cron\_lock() {

global $wpdb;

$value = 0;

if ( wp\_using\_ext\_object\_cache() ) {

/\*

\* Skip local cache and force re-fetch of doing\_cron transient

\* in case another process updated the cache.

\*/

$value = wp\_cache\_get( 'doing\_cron', 'transient', true );

} else {

$row = $wpdb->get\_row( $wpdb->prepare( "SELECT option\_value FROM $wpdb->options WHERE option\_name = %s LIMIT 1", '\_transient\_doing\_cron' ) );

if ( is\_object( $row ) ) {

$value = $row->option\_value;

}

}

return $value;

}

$crons = wp\_get\_ready\_cron\_jobs();

if ( empty( $crons ) ) {

die();

}

$gmt\_time = microtime( true );

$doing\_cron\_transient = get\_transient( 'doing\_cron' );

if ( empty( $doing\_wp\_cron ) ) {

if ( empty( $\_GET['doing\_wp\_cron'] ) ) {

// Called from external script/job. Try setting a lock.

if ( $doing\_cron\_transient && ( $doing\_cron\_transient + WP\_CRON\_LOCK\_TIMEOUT > $gmt\_time ) ) {

return;

}

$doing\_wp\_cron = sprintf( '%.22F', microtime( true ) );

$doing\_cron\_transient = $doing\_wp\_cron;

set\_transient( 'doing\_cron', $doing\_wp\_cron );

} else {

$doing\_wp\_cron = $\_GET['doing\_wp\_cron'];

}

}

if ( $doing\_cron\_transient !== $doing\_wp\_cron ) {

return;

}

foreach ( $crons as $timestamp => $cronhooks ) {

if ( $timestamp > $gmt\_time ) {

break;

}

foreach ( $cronhooks as $hook => $keys ) {

foreach ( $keys as $k => $v ) {

$schedule = $v['schedule'];

if ( $schedule ) {

$result = wp\_reschedule\_event( $timestamp, $schedule, $hook, $v['args'], true );

if ( is\_wp\_error( $result ) ) {

error\_log(

sprintf(

\_\_( 'Cron reschedule event error for hook: %1$s, Error code: %2$s, Error message: %3$s, Data: %4$s' ),

$hook,

$result->get\_error\_code(),

$result->get\_error\_message(),

wp\_json\_encode( $v )

)

);

do\_action( 'cron\_reschedule\_event\_error', $result, $hook, $v );

}

}

$result = wp\_unschedule\_event( $timestamp, $hook, $v['args'], true );

if ( is\_wp\_error( $result ) ) {

error\_log(

sprintf(

\_\_( 'Cron unschedule event error for hook: %1$s, Error code: %2$s, Error message: %3$s, Data: %4$s' ),

$hook,

$result->get\_error\_code(),

$result->get\_error\_message(),

wp\_json\_encode( $v )

)

);

do\_action( 'cron\_unschedule\_event\_error', $result, $hook, $v );

}

do\_action\_ref\_array( $hook, $v['args'] );

// If the hook ran too long and another cron process stole the lock, quit.

if ( \_get\_cron\_lock() !== $doing\_wp\_cron ) {

return;

}

}

}

}

if ( \_get\_cron\_lock() === $doing\_wp\_cron ) {

delete\_transient( 'doing\_cron' );

}

die();

<?php

require\_once \_\_DIR\_\_ . '/wp-load.php';

header( 'Content-Type: text/xml; charset=' . get\_option( 'blog\_charset' ), true );

$link\_cat = '';

if ( ! empty( $\_GET['link\_cat'] ) ) {

$link\_cat = $\_GET['link\_cat'];

if ( ! in\_array( $link\_cat, array( 'all', '0' ), true ) ) {

$link\_cat = absint( (string) urldecode( $link\_cat ) );

}

}

echo '<?xml version="1.0"?' . ">\n";

?>

<opml version="1.0">

<head>

<title>

<?php

/\* translators: %s: Site title. \*/

printf( \_\_( 'Links for %s' ), esc\_attr( get\_bloginfo( 'name', 'display' ) ) );

?>

</title>

<dateCreated><?php echo gmdate( 'D, d M Y H:i:s' ); ?> GMT</dateCreated>

<?php

/\*\*

\* Fires in the OPML header.

\*

\* @since 3.0.0

\*/

do\_action( 'opml\_head' );

?>

</head>

<body>

<?php

if ( empty( $link\_cat ) ) {

$cats = get\_categories(

array(

'taxonomy' => 'link\_category',

'hierarchical' => 0,

)

);

} else {

$cats = get\_categories(

array(

'taxonomy' => 'link\_category',

'hierarchical' => 0,

'include' => $link\_cat,

)

);

}

foreach ( (array) $cats as $cat ) :

$catname = apply\_filters( 'link\_category', $cat->name );

?>

<outline type="category" title="<?php echo esc\_attr( $catname ); ?>">

<?php

$bookmarks = get\_bookmarks( array( 'category' => $cat->term\_id ) );

foreach ( (array) $bookmarks as $bookmark ) :

/\*\*

\* Filters the OPML outline link title text.

\*

\* @since 2.2.0

\*

\* @param string $title The OPML outline title text.

\*/

$title = apply\_filters( 'link\_title', $bookmark->link\_name );

?>

<outline text="<?php echo esc\_attr( $title ); ?>" type="link" xmlUrl="<?php echo esc\_url( $bookmark->link\_rss ); ?>" htmlUrl="<?php echo esc\_url( $bookmark->link\_url ); ?>" updated="

<?php

if ( '0000-00-00 00:00:00' !== $bookmark->link\_updated ) {

echo $bookmark->link\_updated;

}

?>

" />

<?php

endforeach; // $bookmarks

?>

</outline>

<?php

endforeach; // $cats

?>

</body>

</opml>

<?php

if ( ! defined( 'ABSPATH' ) ) {

define( 'ABSPATH', \_\_DIR\_\_ . '/' );

}

error\_reporting( E\_CORE\_ERROR | E\_CORE\_WARNING | E\_COMPILE\_ERROR | E\_ERROR | E\_WARNING | E\_PARSE | E\_USER\_ERROR | E\_USER\_WARNING | E\_RECOVERABLE\_ERROR );

}

if ( file\_exists( ABSPATH . 'wp-config.php' ) ) {

/\*\* The config file resides in ABSPATH \*/

require\_once ABSPATH . 'wp-config.php';

} elseif ( @file\_exists( dirname( ABSPATH ) . '/wp-config.php' ) && ! @file\_exists( dirname( ABSPATH ) . '/wp-settings.php' ) ) {

/\*\* The config file resides one level above ABSPATH but is not part of another installation \*/

require\_once dirname( ABSPATH ) . '/wp-config.php';

} else {

$path = wp\_guess\_url() . '/wp-admin/setup-config.php';

// Redirect to setup-config.php.

if ( ! str\_contains( $\_SERVER['REQUEST\_URI'], 'setup-config' ) ) {

header( 'Location: ' . $path );

exit;

}

wp\_load\_translations\_early();

// Die with an error message.

$die = '<p>' . sprintf(

/\* translators: %s: wp-config.php \*/

\_\_( "There doesn't seem to be a %s file. It is needed before the installation can continue." ),

'<code>wp-config.php</code>'

) . '</p>';

$die .= '<p>' . sprintf(

/\* translators: 1: Documentation URL, 2: wp-config.php \*/

\_\_( 'Need more help? <a href="%1$s">Read the support article on %2$s</a>.' ),

\_\_( 'https://developer.wordpress.org/advanced-administration/wordpress/wp-config/' ),

'<code>wp-config.php</code>'

) . '</p>';

$die .= '<p>' . sprintf(

/\* translators: %s: wp-config.php \*/

\_\_( "You can create a %s file through a web interface, but this doesn't work for all server setups. The safest way is to manually create the file." ),

'<code>wp-config.php</code>'

) . '</p>';

$die .= '<p><a href="' . $path . '" class="button button-large">' . \_\_( 'Create a Configuration File' ) . '</a></p>';

wp\_die( $die, \_\_( 'WordPress &rsaquo; Error' ) );

}

require \_\_DIR\_\_ . '/wp-load.php';

// Redirect to HTTPS login if forced to use SSL.

if ( force\_ssl\_admin() && ! is\_ssl() ) {

if ( str\_starts\_with( $\_SERVER['REQUEST\_URI'], 'http' ) ) {

wp\_safe\_redirect( set\_url\_scheme( $\_SERVER['REQUEST\_URI'], 'https' ) );

exit;

} else {

wp\_safe\_redirect( 'https://' . $\_SERVER['HTTP\_HOST'] . $\_SERVER['REQUEST\_URI'] );

exit;

}

}

function login\_header( $title = null, $message = '', $wp\_error = null ) {

global $error, $interim\_login, $action;

if ( null === $title ) {

$title = \_\_( 'Log In' );

}

// Don't index any of these forms.

add\_filter( 'wp\_robots', 'wp\_robots\_sensitive\_page' );

add\_action( 'login\_head', 'wp\_strict\_cross\_origin\_referrer' );

add\_action( 'login\_head', 'wp\_login\_viewport\_meta' );

if ( ! is\_wp\_error( $wp\_error ) ) {

$wp\_error = new WP\_Error();

}

// Shake it!

$shake\_error\_codes = array( 'empty\_password', 'empty\_email', 'invalid\_email', 'invalidcombo', 'empty\_username', 'invalid\_username', 'incorrect\_password', 'retrieve\_password\_email\_failure' );

$shake\_error\_codes = apply\_filters( 'shake\_error\_codes', $shake\_error\_codes );

if ( $shake\_error\_codes && $wp\_error->has\_errors() && in\_array( $wp\_error->get\_error\_code(), $shake\_error\_codes, true ) ) {

add\_action( 'login\_footer', 'wp\_shake\_js', 12 );

}

$login\_title = get\_bloginfo( 'name', 'display' );

$login\_title = sprintf( \_\_( '%1$s &lsaquo; %2$s &#8212; WordPress' ), $title, $login\_title );

if ( wp\_is\_recovery\_mode() ) {

/\* translators: %s: Login screen title. \*/

$login\_title = sprintf( \_\_( 'Recovery Mode &#8212; %s' ), $login\_title );

}

$login\_title = apply\_filters( 'login\_title', $login\_title, $title );

?><!DOCTYPE html>

<html <?php language\_attributes(); ?>>

<head>

<meta http-equiv="Content-Type" content="<?php bloginfo( 'html\_type' ); ?>; charset=<?php bloginfo( 'charset' ); ?>" />

<title><?php echo $login\_title; ?></title>

<?php

wp\_enqueue\_style( 'login' );

if ( 'loggedout' === $wp\_error->get\_error\_code() ) {

ob\_start();

?>

<script>if("sessionStorage" in window){try{for(var key in sessionStorage){if(key.indexOf("wp-autosave-")!=-1){sessionStorage.removeItem(key)}}}catch(e){}};</script>

<?php

wp\_print\_inline\_script\_tag( wp\_remove\_surrounding\_empty\_script\_tags( ob\_get\_clean() ) );

}

do\_action( 'login\_head' );

$login\_header\_url = \_\_( 'https://wordpress.org/' );

$login\_header\_url = apply\_filters( 'login\_headerurl', $login\_header\_url );

$login\_header\_title = '';

$login\_header\_title = apply\_filters\_deprecated(

'login\_headertitle',

array( $login\_header\_title ),

'5.2.0',

'login\_headertext',

\_\_( 'Usage of the title attribute on the login logo is not recommended for accessibility reasons. Use the link text instead.' )

);

$login\_header\_text = empty( $login\_header\_title ) ? \_\_( 'Powered by WordPress' ) : $login\_header\_title;

$login\_header\_text = apply\_filters( 'login\_headertext', $login\_header\_text );

$classes = array( 'login-action-' . $action, 'wp-core-ui' );

if ( is\_rtl() ) {

$classes[] = 'rtl';

}

if ( $interim\_login ) {

$classes[] = 'interim-login';

?>

<style type="text/css">html{background-color: transparent;}</style>

<?php

if ( 'success' === $interim\_login ) {

$classes[] = 'interim-login-success';

}

}

$classes[] = ' locale-' . sanitize\_html\_class( strtolower( str\_replace( '\_', '-', get\_locale() ) ) );

$classes = apply\_filters( 'login\_body\_class', $classes, $action );

?>

</head>

<body class="login no-js <?php echo esc\_attr( implode( ' ', $classes ) ); ?>">

<?php

wp\_print\_inline\_script\_tag( "document.body.className = document.body.className.replace('no-js','js');" );

?>

<?php

do\_action( 'login\_header' );

?>

<?php

if ( 'confirm\_admin\_email' !== $action && ! empty( $title ) ) :

?>

<h1 class="screen-reader-text"><?php echo $title; ?></h1>

<?php

endif;

?>

<div id="login">

<h1 role="presentation" class="wp-login-logo"><a href="<?php echo esc\_url( $login\_header\_url ); ?>"><?php echo $login\_header\_text; ?></a></h1>

<?php

$message = apply\_filters( 'login\_message', $message );

if ( ! empty( $message ) ) {

echo $message . "\n";

}

// In case a plugin uses $error rather than the $wp\_errors object.

if ( ! empty( $error ) ) {

$wp\_error->add( 'error', $error );

unset( $error );

}

if ( $wp\_error->has\_errors() ) {

$error\_list = array();

$messages = '';

foreach ( $wp\_error->get\_error\_codes() as $code ) {

$severity = $wp\_error->get\_error\_data( $code );

foreach ( $wp\_error->get\_error\_messages( $code ) as $error\_message ) {

if ( 'message' === $severity ) {

$messages .= '<p>' . $error\_message . '</p>';

} else {

$error\_list[] = $error\_message;

}

}

}

if ( ! empty( $error\_list ) ) {

$errors = '';

if ( count( $error\_list ) > 1 ) {

$errors .= '<ul class="login-error-list">';

foreach ( $error\_list as $item ) {

$errors .= '<li>' . $item . '</li>';

}

$errors .= '</ul>';

} else {

$errors .= '<p>' . $error\_list[0] . '</p>';

}

$errors = apply\_filters( 'login\_errors', $errors );

wp\_admin\_notice(

$errors,

array(

'type' => 'error',

'id' => 'login\_error',

'paragraph\_wrap' => false,

)

);

}

if ( ! empty( $messages ) ) {

$messages = apply\_filters( 'login\_messages', $messages );

wp\_admin\_notice(

$messages,

array(

'type' => 'info',

'id' => 'login-message',

'additional\_classes' => array( 'message' ),

'paragraph\_wrap' => false,

)

);

}

}

}

function login\_footer( $input\_id = '' ) {

global $interim\_login;

// Don't allow interim logins to navigate away from the page.

if ( ! $interim\_login ) {

?>

<p id="backtoblog">

<?php

$html\_link = sprintf(

'<a href="%s">%s</a>',

esc\_url( home\_url( '/' ) ),

sprintf(

/\* translators: %s: Site title. \*/

\_x( '&larr; Go to %s', 'site' ),

get\_bloginfo( 'title', 'display' )

)

);

/\*\*

\* Filters the "Go to site" link displayed in the login page footer.

\*

\* @since 5.7.0

\*

\* @param string $link HTML link to the home URL of the current site.

\*/

echo apply\_filters( 'login\_site\_html\_link', $html\_link );

?>

</p>

<?php

the\_privacy\_policy\_link( '<div class="privacy-policy-page-link">', '</div>' );

}

?>

</div><?php // End of <div id="login">. ?>

<?php

if (! $interim\_login &&apply\_filters( 'login\_display\_language\_dropdown', true )) {

$languages = get\_available\_languages();

if ( ! empty( $languages ) ) {

?>

<div class="language-switcher">

<form id="language-switcher" method="get">

<label for="language-switcher-locales">

<span class="dashicons dashicons-translation" aria-hidden="true"></span>

<span class="screen-reader-text">

<?php

/\* translators: Hidden accessibility text. \*/

\_e( 'Language' );

?>

</span>

</label>

<?php

$args = array(

'id' => 'language-switcher-locales',

'name' => 'wp\_lang',

'selected' => determine\_locale(),

'show\_available\_translations' => false,

'explicit\_option\_en\_us' => true,

'languages' => $languages,

);

wp\_dropdown\_languages( apply\_filters( 'login\_language\_dropdown\_args', $args ) );

?>

<?php if ( $interim\_login ) { ?>

<input type="hidden" name="interim-login" value="1" />

<?php } ?>

<?php if ( isset( $\_GET['redirect\_to'] ) && '' !== $\_GET['redirect\_to'] ) { ?>

<input type="hidden" name="redirect\_to" value="<?php echo sanitize\_url( $\_GET['redirect\_to'] ); ?>" />

<?php } ?>

<?php if ( isset( $\_GET['action'] ) && '' !== $\_GET['action'] ) { ?>

<input type="hidden" name="action" value="<?php echo esc\_attr( $\_GET['action'] ); ?>" />

<?php } ?>

<input type="submit" class="button" value="<?php esc\_attr\_e( 'Change' ); ?>">

</form>

</div>

<?php } ?>

<?php } ?>

<?php

if ( ! empty( $input\_id ) ) {

ob\_start();

?>

<script>

try{document.getElementById('<?php echo $input\_id; ?>').focus();}catch(e){}

if(typeof wpOnload==='function')wpOnload();

</script>

<?php

wp\_print\_inline\_script\_tag( wp\_remove\_surrounding\_empty\_script\_tags( ob\_get\_clean() ) );

}

do\_action( 'login\_footer' );

?>

</body>

</html>

<?php

}

function wp\_shake\_js() {

wp\_print\_inline\_script\_tag( "document.querySelector('form').classList.add('shake');" );

}

function wp\_login\_viewport\_meta() {

?>

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

<?php

}

$action = isset( $\_REQUEST['action'] ) ? $\_REQUEST['action'] : 'login';

$errors = new WP\_Error();

if ( isset( $\_GET['key'] ) ) {

$action = 'resetpass';

}

if ( isset( $\_GET['checkemail'] ) ) {

$action = 'checkemail';

}

$default\_actions = array(

'confirm\_admin\_email',

'postpass',

'logout',

'lostpassword',

'retrievepassword',

'resetpass',

'rp',

'register',

'checkemail',

'confirmaction',

'login',

WP\_Recovery\_Mode\_Link\_Service::LOGIN\_ACTION\_ENTERED,

);

// Validate action so as to default to the login screen.

if ( ! in\_array( $action, $default\_actions, true ) && false === has\_filter( 'login\_form\_' . $action ) ) {

$action = 'login';

}

nocache\_headers();

header( 'Content-Type: ' . get\_bloginfo( 'html\_type' ) . '; charset=' . get\_bloginfo( 'charset' ) );

if ( defined( 'RELOCATE' ) && RELOCATE ) { // Move flag is set.

if ( isset( $\_SERVER['PATH\_INFO'] ) && ( $\_SERVER['PATH\_INFO'] !== $\_SERVER['PHP\_SELF'] ) ) {

$\_SERVER['PHP\_SELF'] = str\_replace( $\_SERVER['PATH\_INFO'], '', $\_SERVER['PHP\_SELF'] );

}

$url = dirname( set\_url\_scheme( 'http://' . $\_SERVER['HTTP\_HOST'] . $\_SERVER['PHP\_SELF'] ) );

if ( get\_option( 'siteurl' ) !== $url ) {

update\_option( 'siteurl', $url );

}

}

$secure = ( 'https' === parse\_url( wp\_login\_url(), PHP\_URL\_SCHEME ) );

setcookie( TEST\_COOKIE, 'WP Cookie check', 0, COOKIEPATH, COOKIE\_DOMAIN, $secure );

if ( SITECOOKIEPATH !== COOKIEPATH ) {

setcookie( TEST\_COOKIE, 'WP Cookie check', 0, SITECOOKIEPATH, COOKIE\_DOMAIN, $secure );

}

if ( isset( $\_GET['wp\_lang'] ) ) {

setcookie( 'wp\_lang', sanitize\_text\_field( $\_GET['wp\_lang'] ), 0, COOKIEPATH, COOKIE\_DOMAIN, $secure );

}

do\_action( 'login\_init' );

do\_action( "login\_form\_{$action}" );

$http\_post = ( 'POST' === $\_SERVER['REQUEST\_METHOD'] );

$interim\_login = isset( $\_REQUEST['interim-login'] );

$login\_link\_separator = apply\_filters( 'login\_link\_separator', ' | ' );

switch ( $action ) {

case 'confirm\_admin\_email':

/\*

\* Note that `is\_user\_logged\_in()` will return false immediately after logging in

\* as the current user is not set, see wp-includes/pluggable.php.

\* However this action runs on a redirect after logging in.

\*/

if ( ! is\_user\_logged\_in() ) {

wp\_safe\_redirect( wp\_login\_url() );

exit;

}

if ( ! empty( $\_REQUEST['redirect\_to'] ) ) {

$redirect\_to = $\_REQUEST['redirect\_to'];

} else {

$redirect\_to = admin\_url();

}

if ( current\_user\_can( 'manage\_options' ) ) {

$admin\_email = get\_option( 'admin\_email' );

} else {

wp\_safe\_redirect( $redirect\_to );

exit;

}

/\*\*

\* Filters the interval for dismissing the admin email confirmation screen.

\*

\* If `0` (zero) is returned, the "Remind me later" link will not be displayed.

\*

\* @since 5.3.1

\*

\* @param int $interval Interval time (in seconds). Default is 3 days.

\*/

$remind\_interval = (int) apply\_filters( 'admin\_email\_remind\_interval', 3 \* DAY\_IN\_SECONDS );

if ( ! empty( $\_GET['remind\_me\_later'] ) ) {

if ( ! wp\_verify\_nonce( $\_GET['remind\_me\_later'], 'remind\_me\_later\_nonce' ) ) {

wp\_safe\_redirect( wp\_login\_url() );

exit;

}

if ( $remind\_interval > 0 ) {

update\_option( 'admin\_email\_lifespan', time() + $remind\_interval );

}

$redirect\_to = add\_query\_arg( 'admin\_email\_remind\_later', 1, $redirect\_to );

wp\_safe\_redirect( $redirect\_to );

exit;

}

if ( ! empty( $\_POST['correct-admin-email'] ) ) {

if ( ! check\_admin\_referer( 'confirm\_admin\_email', 'confirm\_admin\_email\_nonce' ) ) {

wp\_safe\_redirect( wp\_login\_url() );

exit;

}

/\*\*

\* Filters the interval for redirecting the user to the admin email confirmation screen.

\*

\* If `0` (zero) is returned, the user will not be redirected.

\*

\* @since 5.3.0

\*

\* @param int $interval Interval time (in seconds). Default is 6 months.

\*/

$admin\_email\_check\_interval = (int) apply\_filters( 'admin\_email\_check\_interval', 6 \* MONTH\_IN\_SECONDS );

if ( $admin\_email\_check\_interval > 0 ) {

update\_option( 'admin\_email\_lifespan', time() + $admin\_email\_check\_interval );

}

wp\_safe\_redirect( $redirect\_to );

exit;

}

login\_header( \_\_( 'Confirm your administration email' ), '', $errors );

do\_action( 'admin\_email\_confirm', $errors );

?>

<form class="admin-email-confirm-form" name="admin-email-confirm-form" action="<?php echo esc\_url( site\_url( 'wp-login.php?action=confirm\_admin\_email', 'login\_post' ) ); ?>" method="post">

<?php

/\*\*

\* Fires inside the admin-email-confirm-form form tags, before the hidden fields.

\*

\* @since 5.3.0

\*/

do\_action( 'admin\_email\_confirm\_form' );

wp\_nonce\_field( 'confirm\_admin\_email', 'confirm\_admin\_email\_nonce' );

?>

<input type="hidden" name="redirect\_to" value="<?php echo esc\_attr( $redirect\_to ); ?>" />

<h1 class="admin-email\_\_heading">

<?php \_e( 'Administration email verification' ); ?>

</h1>

<p class="admin-email\_\_details">

<?php \_e( 'Please verify that the <strong>administration email</strong> for this website is still correct.' ); ?>

<?php

/\* translators: URL to the WordPress help section about admin email. \*/

$admin\_email\_help\_url = \_\_( 'https://wordpress.org/documentation/article/settings-general-screen/#email-address' );

$accessibility\_text = sprintf(

'<span class="screen-reader-text"> %s</span>',

/\* translators: Hidden accessibility text. \*/

\_\_( '(opens in a new tab)' )

);

printf(

'<a href="%s" target="\_blank">%s%s</a>',

esc\_url( $admin\_email\_help\_url ),

\_\_( 'Why is this important?' ),

$accessibility\_text

);

?>

</p>

<p class="admin-email\_\_details">

<?php

printf(

/\* translators: %s: Admin email address. \*/

\_\_( 'Current administration email: %s' ),

'<strong>' . esc\_html( $admin\_email ) . '</strong>'

);

?>

</p>

<p class="admin-email\_\_details">

<?php \_e( 'This email may be different from your personal email address.' ); ?>

</p>

<div class="admin-email\_\_actions">

<div class="admin-email\_\_actions-primary">

<?php

$change\_link = admin\_url( 'options-general.php' );

$change\_link = add\_query\_arg( 'highlight', 'confirm\_admin\_email', $change\_link );

?>

<a class="button button-large" href="<?php echo esc\_url( $change\_link ); ?>"><?php \_e( 'Update' ); ?></a>

<input type="submit" name="correct-admin-email" id="correct-admin-email" class="button button-primary button-large" value="<?php esc\_attr\_e( 'The email is correct' ); ?>" />

</div>

<?php if ( $remind\_interval > 0 ) : ?>

<div class="admin-email\_\_actions-secondary">

<?php

$remind\_me\_link = wp\_login\_url( $redirect\_to );

$remind\_me\_link = add\_query\_arg(

array(

'action' => 'confirm\_admin\_email',

'remind\_me\_later' => wp\_create\_nonce( 'remind\_me\_later\_nonce' ),),$remind\_me\_link);?>

<a href="<?php echo esc\_url( $remind\_me\_link ); ?>"><?php \_e( 'Remind me later' ); ?></a>

</div>

<?php endif; ?>

</div>

</form>

<?php

login\_footer();

break;

case 'postpass':

if ( ! isset( $\_POST['post\_password'] ) || ! is\_string( $\_POST['post\_password'] ) ) {

wp\_safe\_redirect( wp\_get\_referer() );

exit;

}

require\_once ABSPATH . WPINC . '/class-phpass.php';

$hasher = new PasswordHash( 8, true );

$expire = apply\_filters( 'post\_password\_expires', time() + 10 \* DAY\_IN\_SECONDS );

$referer = wp\_get\_referer();

if ( $referer ) {

$secure = ( 'https' === parse\_url( $referer, PHP\_URL\_SCHEME ) );

} else {

$secure = false;

}

setcookie( 'wp-postpass\_' . COOKIEHASH, $hasher->HashPassword( wp\_unslash( $\_POST['post\_password'] ) ), $expire, COOKIEPATH, COOKIE\_DOMAIN, $secure );

wp\_safe\_redirect( wp\_get\_referer() );

exit;

case 'logout':

check\_admin\_referer( 'log-out' );

$user = wp\_get\_current\_user();

wp\_logout();

if ( ! empty( $\_REQUEST['redirect\_to'] ) && is\_string( $\_REQUEST['redirect\_to'] ) ) {

$redirect\_to = $\_REQUEST['redirect\_to'];

$requested\_redirect\_to = $redirect\_to;

} else {

$redirect\_to = add\_query\_arg(

array(

'loggedout' => 'true',

'wp\_lang' => get\_user\_locale( $user ),

),

wp\_login\_url()

);

$requested\_redirect\_to = '';

}

$redirect\_to = apply\_filters( 'logout\_redirect', $redirect\_to, $requested\_redirect\_to, $user );

wp\_safe\_redirect( $redirect\_to );

exit;

case 'lostpassword':

case 'retrievepassword':

if ( $http\_post ) {

$errors = retrieve\_password();

if ( ! is\_wp\_error( $errors ) ) {

$redirect\_to = ! empty( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : 'wp-login.php?checkemail=confirm';

wp\_safe\_redirect( $redirect\_to );

exit;

}

}

if ( isset( $\_GET['error'] ) ) {

if ( 'invalidkey' === $\_GET['error'] ) {

$errors->add( 'invalidkey', \_\_( '<strong>Error:</strong> Your password reset link appears to be invalid. Please request a new link below.' ) );

} elseif ( 'expiredkey' === $\_GET['error'] ) {

$errors->add( 'expiredkey', \_\_( '<strong>Error:</strong> Your password reset link has expired. Please request a new link below.' ) );

}

}

$lostpassword\_redirect = ! empty( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : '';

$redirect\_to = apply\_filters( 'lostpassword\_redirect', $lostpassword\_redirect );

do\_action( 'lost\_password', $errors );

login\_header(

\_\_( 'Lost Password' ),

wp\_get\_admin\_notice(

\_\_( 'Please enter your username or email address. You will receive an email message with instructions on how to reset your password.' ),

array(

'type' => 'info',

'additional\_classes' => array( 'message' ),

)

),

$errors

);

$user\_login = '';

if ( isset( $\_POST['user\_login'] ) && is\_string( $\_POST['user\_login'] ) ) {

$user\_login = wp\_unslash( $\_POST['user\_login'] );

}

?>

<form name="lostpasswordform" id="lostpasswordform" action="<?php echo esc\_url( network\_site\_url( 'wp-login.php?action=lostpassword', 'login\_post' ) ); ?>" method="post">

<p>

<label for="user\_login"><?php \_e( 'Username or Email Address' ); ?></label>

<input type="text" name="user\_login" id="user\_login" class="input" value="<?php echo esc\_attr( $user\_login ); ?>" size="20" autocapitalize="off" autocomplete="username" required="required" />

</p>

<?php

do\_action( 'lostpassword\_form' );

?>

<input type="hidden" name="redirect\_to" value="<?php echo esc\_attr( $redirect\_to ); ?>" />

<p class="submit">

<input type="submit" name="wp-submit" id="wp-submit" class="button button-primary button-large" value="<?php esc\_attr\_e( 'Get New Password' ); ?>" />

</p>

</form>

<p id="nav">

<a class="wp-login-log-in" href="<?php echo esc\_url( wp\_login\_url() ); ?>"><?php \_e( 'Log in' ); ?></a>

<?php

if ( get\_option( 'users\_can\_register' ) ) {

$registration\_url = sprintf( '<a class="wp-login-register" href="%s">%s</a>', esc\_url( wp\_registration\_url() ), \_\_( 'Register' ) );

echo esc\_html( $login\_link\_separator );

/\*\* This filter is documented in wp-includes/general-template.php \*/

echo apply\_filters( 'register', $registration\_url );

}

?>

</p>

<?php

login\_footer( 'user\_login' );

break;

case 'resetpass':

case 'rp':

list( $rp\_path ) = explode( '?', wp\_unslash( $\_SERVER['REQUEST\_URI'] ) );

$rp\_cookie = 'wp-resetpass-' . COOKIEHASH;

if ( isset( $\_GET['key'] ) && isset( $\_GET['login'] ) ) {

$value = sprintf( '%s:%s', wp\_unslash( $\_GET['login'] ), wp\_unslash( $\_GET['key'] ) );

setcookie( $rp\_cookie, $value, 0, $rp\_path, COOKIE\_DOMAIN, is\_ssl(), true );

wp\_safe\_redirect( remove\_query\_arg( array( 'key', 'login' ) ) );

exit;

}

if ( isset( $\_COOKIE[ $rp\_cookie ] ) && 0 < strpos( $\_COOKIE[ $rp\_cookie ], ':' ) ) {

list( $rp\_login, $rp\_key ) = explode( ':', wp\_unslash( $\_COOKIE[ $rp\_cookie ] ), 2 );

$user = check\_password\_reset\_key( $rp\_key, $rp\_login );

if ( isset( $\_POST['pass1'] ) && ! hash\_equals( $rp\_key, $\_POST['rp\_key'] ) ) {

$user = false;

}

} else {

$user = false;

}

if ( ! $user || is\_wp\_error( $user ) ) {

setcookie( $rp\_cookie, ' ', time() - YEAR\_IN\_SECONDS, $rp\_path, COOKIE\_DOMAIN, is\_ssl(), true );

if ( $user && $user->get\_error\_code() === 'expired\_key' ) {

wp\_redirect( site\_url( 'wp-login.php?action=lostpassword&error=expiredkey' ) );

} else {

wp\_redirect( site\_url( 'wp-login.php?action=lostpassword&error=invalidkey' ) );

}

exit;

}

$errors = new WP\_Error();

// Check if password is one or all empty spaces.

if ( ! empty( $\_POST['pass1'] ) ) {

$\_POST['pass1'] = trim( $\_POST['pass1'] );

if ( empty( $\_POST['pass1'] ) ) {

$errors->add( 'password\_reset\_empty\_space', \_\_( 'The password cannot be a space or all spaces.' ) );

}

}

// Check if password fields do not match.

if ( ! empty( $\_POST['pass1'] ) && trim( $\_POST['pass2'] ) !== $\_POST['pass1'] ) {

$errors->add( 'password\_reset\_mismatch', \_\_( '<strong>Error:</strong> The passwords do not match.' ) );

}

do\_action( 'validate\_password\_reset', $errors, $user );

if ( ( ! $errors->has\_errors() ) && isset( $\_POST['pass1'] ) && ! empty( $\_POST['pass1'] ) ) {

reset\_password( $user, $\_POST['pass1'] );

setcookie( $rp\_cookie, ' ', time() - YEAR\_IN\_SECONDS, $rp\_path, COOKIE\_DOMAIN, is\_ssl(), true );

login\_header(

\_\_( 'Password Reset' ),

wp\_get\_admin\_notice(

\_\_( 'Your password has been reset.' ) . ' <a href="' . esc\_url( wp\_login\_url() ) . '">' . \_\_( 'Log in' ) . '</a>',

array(

'type' => 'info',

'additional\_classes' => array( 'message', 'reset-pass' ),

)

)

);

login\_footer();

exit;

}

wp\_enqueue\_script( 'utils' );

wp\_enqueue\_script( 'user-profile' );

login\_header(

\_\_( 'Reset Password' ),

wp\_get\_admin\_notice(

\_\_( 'Enter your new password below or generate one.' ),

array(

'type' => 'info',

'additional\_classes' => array( 'message', 'reset-pass' ),

)

),

$errors

);

?>

<form name="resetpassform" id="resetpassform" action="<?php echo esc\_url( network\_site\_url( 'wp-login.php?action=resetpass', 'login\_post' ) ); ?>" method="post" autocomplete="off">

<input type="hidden" id="user\_login" value="<?php echo esc\_attr( $rp\_login ); ?>" autocomplete="off" />

<div class="user-pass1-wrap">

<p>

<label for="pass1"><?php \_e( 'New password' ); ?></label>

</p>

<div class="wp-pwd">

<input type="password" name="pass1" id="pass1" class="input password-input" size="24" value="" autocomplete="new-password" spellcheck="false" data-reveal="1" data-pw="<?php echo esc\_attr( wp\_generate\_password( 16 ) ); ?>" aria-describedby="pass-strength-result" />

<button type="button" class="button button-secondary wp-hide-pw hide-if-no-js" data-toggle="0" aria-label="<?php esc\_attr\_e( 'Hide password' ); ?>">

<span class="dashicons dashicons-hidden" aria-hidden="true"></span>

</button>

<div id="pass-strength-result" class="hide-if-no-js" aria-live="polite"><?php \_e( 'Strength indicator' ); ?></div>

</div>

<div class="pw-weak">

<input type="checkbox" name="pw\_weak" id="pw-weak" class="pw-checkbox" />

<label for="pw-weak"><?php \_e( 'Confirm use of weak password' ); ?></label>

</div>

</div>

<p class="user-pass2-wrap">

<label for="pass2"><?php \_e( 'Confirm new password' ); ?></label>

<input type="password" name="pass2" id="pass2" class="input" size="20" value="" autocomplete="new-password" spellcheck="false" />

</p>

<p class="description indicator-hint"><?php echo wp\_get\_password\_hint(); ?></p>

<?php

do\_action( 'resetpass\_form', $user );

?>

<input type="hidden" name="rp\_key" value="<?php echo esc\_attr( $rp\_key ); ?>" />

<p class="submit reset-pass-submit">

<button type="button" class="button wp-generate-pw hide-if-no-js skip-aria-expanded"><?php \_e( 'Generate Password' ); ?></button>

<input type="submit" name="wp-submit" id="wp-submit" class="button button-primary button-large" value="<?php esc\_attr\_e( 'Save Password' ); ?>" />

</p>

</form>

<p id="nav">

<a class="wp-login-log-in" href="<?php echo esc\_url( wp\_login\_url() ); ?>"><?php \_e( 'Log in' ); ?></a>

<?php

if ( get\_option( 'users\_can\_register' ) ) {

$registration\_url = sprintf( '<a class="wp-login-register" href="%s">%s</a>', esc\_url( wp\_registration\_url() ), \_\_( 'Register' ) );

echo esc\_html( $login\_link\_separator );

/\*\* This filter is documented in wp-includes/general-template.php \*/

echo apply\_filters( 'register', $registration\_url );}?>

</p>

<?php

login\_footer( 'pass1' );

break;

case 'register':

if ( is\_multisite() ) {

wp\_redirect( apply\_filters( 'wp\_signup\_location', network\_site\_url( 'wp-signup.php' ) ) );exit;}

if ( ! get\_option( 'users\_can\_register' ) ) {

wp\_redirect( site\_url( 'wp-login.php?registration=disabled' ) );

exit;

}

$user\_login = '';

$user\_email = '';

if ( $http\_post ) {

if ( isset( $\_POST['user\_login'] ) && is\_string( $\_POST['user\_login'] ) ) {

$user\_login = wp\_unslash( $\_POST['user\_login'] );

}

if ( isset( $\_POST['user\_email'] ) && is\_string( $\_POST['user\_email'] ) ) {

$user\_email = wp\_unslash( $\_POST['user\_email'] );

}

$errors = register\_new\_user( $user\_login, $user\_email );

if ( ! is\_wp\_error( $errors ) ) {

$redirect\_to = ! empty( $\_POST['redirect\_to'] ) ? $\_POST['redirect\_to'] : 'wp-login.php?checkemail=registered';

wp\_safe\_redirect( $redirect\_to );

exit;

}

}

$registration\_redirect = ! empty( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : '';

$redirect\_to = apply\_filters( 'registration\_redirect', $registration\_redirect, $errors );

login\_header(

\_\_( 'Registration Form' ),

wp\_get\_admin\_notice(

\_\_( 'Register For This Site' ),

array(

'type' => 'info',

'additional\_classes' => array( 'message', 'register' ),)),$errors);

?>

<form name="registerform" id="registerform" action="<?php echo esc\_url( site\_url( 'wp-login.php?action=register', 'login\_post' ) ); ?>" method="post" novalidate="novalidate">

<p>

<label for="user\_login"><?php \_e( 'Username' ); ?></label>

<input type="text" name="user\_login" id="user\_login" class="input" value="<?php echo esc\_attr( $user\_login ); ?>" size="20" autocapitalize="off" autocomplete="username" required="required" />

</p>

<p>

<label for="user\_email"><?php \_e( 'Email' ); ?></label>

<input type="email" name="user\_email" id="user\_email" class="input" value="<?php echo esc\_attr( $user\_email ); ?>" size="25" autocomplete="email" required="required" />

</p>

<?php do\_action( 'register\_form' );?>

<p id="reg\_passmail">

<?php \_e( 'Registration confirmation will be emailed to you.' ); ?>

</p>

<input type="hidden" name="redirect\_to" value="<?php echo esc\_attr( $redirect\_to ); ?>" />

<p class="submit">

<input type="submit" name="wp-submit" id="wp-submit" class="button button-primary button-large" value="<?php esc\_attr\_e( 'Register' ); ?>" />

</p>

</form>

<p id="nav">

<a class="wp-login-log-in" href="<?php echo esc\_url( wp\_login\_url() ); ?>"><?php \_e( 'Log in' ); ?></a>

<?php

echo esc\_html( $login\_link\_separator );

$html\_link = sprintf( '<a class="wp-login-lost-password" href="%s">%s</a>', esc\_url( wp\_lostpassword\_url() ), \_\_( 'Lost your password?' ) );

echo apply\_filters( 'lost\_password\_html\_link', $html\_link );?>

</p>

<?php

login\_footer( 'user\_login' );

break;

case 'checkemail':

$redirect\_to = admin\_url();

$errors = new WP\_Error();

if ( 'confirm' === $\_GET['checkemail'] ) {

$errors->add(

'confirm',

sprintf(( 'Check your email for the confirmation link, then visit the <a href="%s">login page</a>.' ),wp\_login\_url()),'message');

} elseif ( 'registered' === $\_GET['checkemail'] ) {

$errors->add(

'registered',

sprintf(

/\* translators: %s: Link to the login page. \*/

\_\_( 'Registration complete. Please check your email, then visit the <a href="%s">login page</a>.' ),

wp\_login\_url()

),

'message'

);

}

$errors = apply\_filters( 'wp\_login\_errors', $errors, $redirect\_to );

login\_header( \_\_( 'Check your email' ), '', $errors );

login\_footer();

break;

case 'confirmaction':

if ( ! isset( $\_GET['request\_id'] ) ) {

wp\_die( \_\_( 'Missing request ID.' ) );

}

if ( ! isset( $\_GET['confirm\_key'] ) ) {

wp\_die( \_\_( 'Missing confirm key.' ) );

}

$request\_id = (int) $\_GET['request\_id'];

$key = sanitize\_text\_field( wp\_unslash( $\_GET['confirm\_key'] ) );

$result = wp\_validate\_user\_request\_key( $request\_id, $key );

if ( is\_wp\_error( $result ) ) {

wp\_die( $result );

}

do\_action( 'user\_request\_action\_confirmed', $request\_id );

$message = \_wp\_privacy\_account\_request\_confirmed\_message( $request\_id );

login\_header( \_\_( 'User action confirmed.' ), $message );

login\_footer();

exit;

case 'login':

default:

$secure\_cookie = '';

$customize\_login = isset( $\_REQUEST['customize-login'] );

if ( $customize\_login ) {

wp\_enqueue\_script( 'customize-base' );

}

// If the user wants SSL but the session is not SSL, force a secure cookie.

if ( ! empty( $\_POST['log'] ) && ! force\_ssl\_admin() ) {

$user\_name = sanitize\_user( wp\_unslash( $\_POST['log'] ) );

$user = get\_user\_by( 'login', $user\_name );

if ( ! $user && strpos( $user\_name, '@' ) ) {

$user = get\_user\_by( 'email', $user\_name );}

if ( $user ) {

if ( get\_user\_option( 'use\_ssl', $user->ID ) ) {

$secure\_cookie = true;

force\_ssl\_admin( true );}}}

if ( isset( $\_REQUEST['redirect\_to'] ) && is\_string( $\_REQUEST['redirect\_to'] ) ) {

$redirect\_to = $\_REQUEST['redirect\_to'];

// Redirect to HTTPS if user wants SSL.

if ( $secure\_cookie && str\_contains( $redirect\_to, 'wp-admin' ) ) {

$redirect\_to = preg\_replace( '|^http://|', 'https://', $redirect\_to );}

} else {$redirect\_to = admin\_url();}

$reauth = empty( $\_REQUEST['reauth'] ) ? false : true;

$user = wp\_signon( array(), $secure\_cookie );

if ( empty( $\_COOKIE[ LOGGED\_IN\_COOKIE ] ) ) {

if ( headers\_sent() ) {

$user = new WP\_Error(

'test\_cookie',

sprintf(

\_\_( '<strong>Error:</strong> Cookies are blocked due to unexpected output. For help, please see <a href="%1$s">this documentation</a> or try the <a href="%2$s">support forums</a>.' ),

\_\_( 'https://developer.wordpress.org/advanced-administration/wordpress/cookies/' ),

\_\_( 'https://wordpress.org/support/forums/' )

)

);

} elseif ( isset( $\_POST['testcookie'] ) && empty( $\_COOKIE[ TEST\_COOKIE ] ) ) {

// If cookies are disabled, the user can't log in even with a valid username and password.

$user = new WP\_Error(

'test\_cookie',

sprintf(

/\* translators: %s: Browser cookie documentation URL. \*/

\_\_( '<strong>Error:</strong> Cookies are blocked or not supported by your browser. You must <a href="%s">enable cookies</a> to use WordPress.' ),

\_\_( 'https://developer.wordpress.org/advanced-administration/wordpress/cookies/#enable-cookies-in-your-browser' )));}}

$requested\_redirect\_to = isset( $\_REQUEST['redirect\_to'] ) && is\_string( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : '';

$redirect\_to = apply\_filters( 'login\_redirect', $redirect\_to, $requested\_redirect\_to, $user );

if ( ! is\_wp\_error( $user ) && ! $reauth ) {

if ( $interim\_login ) {

$message = '<p class="message">' . \_\_( 'You have logged in successfully.' ) . '</p>';

$interim\_login = 'success';

login\_header( '', $message );?>

</div>

<?php

do\_action( 'login\_footer' );

if ( $customize\_login ) {

ob\_start();

?>

<script>setTimeout( function(){ new wp.customize.Messenger({ url: '<?php echo wp\_customize\_url(); ?>', channel: 'login' }).send('login') }, 1000 );</script>

<?php

wp\_print\_inline\_script\_tag( wp\_remove\_surrounding\_empty\_script\_tags( ob\_get\_clean() ) )}?>