**Слайд 1.**

Добрый день уважаемый председатель и члены государственной экзаменационной комиссии. Я буду представлять дипломный проект на тему «Веб-приложение для раздельного сбора бытовых отходов».

**Слайд 2.**

В современном мире, на фоне роста экологической осознанности, все больше людей стремятся внести свой вклад в сохранение природы.

Одновременно с этим ощущается явный дефицит доступных и эффективных цифровых решений, способных мотивировать, обучать и поддерживать пользователей на пути к осознанному потреблению. Для тех, кто только начинает разделять отходы, процесс часто кажется сложным из-за нехватки информации, понятных инструкций и удобных сервисов. Это делает необходимым создание технологичных решений, которые упростят сортировку, повысят вовлечённость и помогут каждому сделать свой вклад в заботу о планете.

Веб-приложение для раздельного сбора отходов может стать таким решением – удобным, интуитивным и мотивирующим инструментом, который превратит экологичные привычки в простую рутину.

**Слайд 3.**

Исходя из вышесказанного, цель дипломного проектирования – Разработка веб-приложения для раздельного сбора бытовых отходов.

Для достижения представленной цели были поставлены следующие задачи:

1. провести анализ аналогичных решений;
2. спроектировать составляющие приложения;
3. реализовать клиентскую и серверную части согласно требованиям;
4. протестировать функционал приложения;
5. составить руководство пользования;
6. провести технико-экономическое обоснование проекта.

**Слайд 4-7.**

Для решения схожих задач на рынке уже существуют программные решения и платформы. На слайдах представлены три популярные платформы — Зеленая карта (4), rsbor (5) и target99 (6). Эти сервисы широко используются покупателями. Однако, несмотря на популярность, у этих решений имеются определённые ограничения.

Итоговое сравнение вы можете увидеть в таблице (7)

**Слайд 8.**

В рамках этапа проектирования была разработана диаграмма вариантов использования, которая содержит в себе все функциональные возможности для каждой из доступных пользователям ролей веб-приложения. Для роли «Гость» доступны только минимальные возможности ознакомления с системой (можно сказать, какие). Роль «Пользователя» предоставляет расширенный функционал предыдущей роли. Роль «Администратор» содержит функционал для управления системой.

**Слайд 9.**

Также, в рамках этапа проектирования была разработана логическая схема базы данных веб-приложения. База данных содержит в себе 12 таблиц. База данных MySQL.

**Слайд 10.**

(Архитектура) Представленная на диаграмме развертывания, включает в себя базу данных, серверную часть приложения на Node.js и клиентскую часть, реализованную на React. Все основные сервисы развернуты в контейнерах Docker под управлением Docker Compose на базе операционной системы Windows Server 2022. Также можно увидеть подключенные сторонние сервисы.

**Слайд 11.**

(Диаграмма последовательности)

Чтобы добавить новый вид вторсырья в систему, администратор сначала взаимодействует с графическим интерфейсом (GUI), где вводит информацию о новом виде вторсырья и отправляет соответствующий запрос. После этого данные передаются в блок обработки запроса, который выполняет первичную валидацию и направляет информацию дальше, в блок сервиса по управлению вторсырьём. Этот сервис, в свою очередь, обращается к блоку базы данных, чтобы записать новый тип вторсырья в соответствующую таблицу. Если во время выполнения возникает ошибка, например, при подключении к базе данных или при нарушении уникальности записей, система формирует ответ с ошибкой, который возвращается из блока базы данных в сервисный блок, далее в блок обработки запроса, а затем обратно в графический интерфейс, где администратору отображается сообщение об ошибке. Если операция проходит успешно, база данных сохраняет новый вид вторсырья, где администратор получает уведомление об успешном добавлении нового вида вторсырья в систему.

**Слайд 12.**

(Блок-схема алгоритма сдачи вторсырья)

Чтобы отметить сданное вторсырье в приложении, пользователь должен сначала зарегистрироваться или войти в веб-приложение. Затем на странице приема вторсырья пользователь должен ввести два ключа, которые были выданы в пункте приема, в соответствующую форму. После ввода ключей начинаются проверки. Вначале происходит проверка секретного ключа. Если ключ существует, проверка продолжается, в противном случае выводится ошибка, сообщающая о его отсутствии. Затем происходит проверка, связан ли данный ключ с конкретным пунктом приема. Если связь существует, проверка продолжается, в противном случае выводится сообщение о недействительности ключа. Далее происходит проверка второго ключа для подтверждения веса. Если ключ верен, работа алгоритма продолжается, в противном случае выводится сообщение о недействительности ключа. Исходя из введенного ключа находится информация о вторсырье, которое пользователь сдал, включая количество баллов, начисляемых за один килограмм, и количество новой продукции, произведенной из одного сданного килограмма вторсырья. Затем проверяется количество баллов, имеющихся у пользователя. Рассчитываются новые баллы, вес и количество произведенной продукции. Новые баллы записываются пользователю, а информация о приеме добавляется в таблицу для статистики. Код для пункта приема обновляется, поскольку одноразовые ключи не могут быть повторно использованы. В конечном итоге пользователю выводится сообщение о его баллах, количество начисленных баллов и количество новой продукции, которая будет произведена из сданного пользователем вторсырья.

**Слайд 13.**

(Блок-схема алгоритма просмотра скидок и открытия промокодов)

Для того чтобы получить и активировать промокод в приложении, пользователь должен сначала авторизоваться. После успешной авторизации происходит получение данных с сервера, в том числе информации о текущей роли пользователя. Далее осуществляется проверка, обладает ли пользователь ролью «Пользователь». Если роль не соответствует, алгоритм прекращается. В случае, если роль подходит, происходит извлечение данных пользователя из базы данных. 23 Затем проверяется наличие баллов у пользователя. Если баллы отсутствуют, пользователю выводится сообщение «Скидок нет», и выполнение алгоритма завершается. В противном случае осуществляется вывод списка доступных скидок. Далее пользователь может открыть промокод, после чего с его аккаунта списываются баллы, соответствующие стоимости промокода. Затем происходит вывод информации об открытом промокоде, после чего алгоритм завершается.

**Слайд 14.**

(стек технологий)

**Слайд 15.**

Тестирование серверной части веб-приложения проводилось с помощью нескольких подходов: негативный, позитивный, функциональный. Позитивное и негативное на слайде. В рамках тестирования было разработано **29 функциональных тест-кейсов** В результате проведённого тестирования удалось достичь **100% покрытия функциональности** серверной части.

**Слайд 16**

Также было проведено технико-экономическое обоснование программного средства. Результаты данной работы представлены в итоговой таблице.

**Слайд 17.**

Далее переходим к переходим демонстрации видео с обзором общего функционала веб-приложения.

**Слайд 18**

В ходе выполнения дипломного проекта было разработано веб-приложение для раздельного сбора бытовых отходов и выполнены следующие задачи:

* произведен анализ существующих решений;
* спроектированы составляющие приложения;
* разработана клиентская и серверная части приложения;
* протестирован функционал приложения;
* составлено руководство пользователя и технико-экономическое обоснование для приложения.

Разработанный проект, поддерживает работу 3 пользовательских ролей, включает в себя 12 таблиц базы данных. Было произведено тестирование на 29 функциональных тест-кейсов, покрытие составило 100%. Авторский объем кода составляет примерно 11 000 строк.

Веб-приложение поддерживает архитектуру, которая позволяет в дальнейшем быстро и безопасно расширить функционал.

Доклад окончен, спасибо за внимание.