

Web-приложение «AIVVA» состоит из следующих разделов:

1. Главная. В данном разделе изложена общая информация о салоне и его услугах.

2. Услуги. Из предоставленных услуг, пользователь может выбрать нужную услугу и ознакомиться с ней, так же в этом разделе расположена кнопка записи, где пользователь может записаться на услугу.

3. О нас. Раздел содержит информацию о салоне, его акциях и скидках, информацию о мастерах.

4. Фотогалерея. Данный раздел содержит фото салона, фото мастеров и фото работ. Фотоальбом представляет из себя слайдер изображений.

5. Отзывы. В данном разделе содержатся отзывы клиентов и ответы на них.

6. Контакты. В данном разделе содержится контактная информация: месторасположение, телефон, электронная почта, время работы, карта.

При разработке сайта были использованы язык разметки HTML, CSS, язык программирования JavaScript. IDE: PhpStorm.

П. М. Баль

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **М. А. Подалов**, ст. преподаватель

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОГРАММНОГО И АППАРАТНОГО РЕШЕНИЙ ДЛЯ СИСТЕМЫ КРАТКОСРОЧНОЙ АРЕНДЫ ЭЛЕКТРОСАМОКАТОВ

Данный проект направлен на создание системы краткосрочной аренды электросамокатов, которая предоставит пользователям удобный и экологически чистый способ передвижения по городу [1, 2]. Для реализации данной системы необходимо провести проектирование и разработку программного и аппаратного обеспечения, которые позволят контролировать состояние самокатов и их местоположение, а также обеспечат удобный интерфейс для пользователей.

В рамках проекта необходимо решить следующие задачи:

Разработка программного обеспечения, которое позволит пользователям зарегистрироваться в системе, выбирать доступные самокаты для аренды, оплачивать услуги. Важными особенностями программного обеспечения будут надежность и безопасность, чтобы пользователь мог быть уверен в сохранности своих личных данных.

Разработка аппаратного обеспечения, которое позволит контролировать состояние самокатов и их местоположение в режиме реального времени. Для этого необходимо установить на самокаты GPS-навигаторы и датчики уровня заряда батареи. Это позволит администраторам системы получать информацию о местоположении каждого самоката, а также отслеживать состояние батарей и своевременно заменять их, если это необходимо.

Разработка мобильного приложения для пользователей, которое позволит быстро и удобно выбирать доступные самокаты, оплачивать услуги и получать информацию о акциях и купонах. Важными особенностями мобильного приложения будут его удобный интерфейс и быстрое действие. Пользователь должен иметь возможность быстро выбрать самокат и оплатить услугу, не тратя много времени на ожидание. Для этого необходимо создать удобный интерфейс, который будет понятен не только опытным пользователям, но и новичкам.

Разработка системы управления, которая будет позволять администраторам системы контролировать работу самокатов, получать статистическую информацию и управлять ценообразованием. Важными особенностями системы управления будут ее надежность и безопасность. Администраторы должны иметь возможность получать актуальную информацию о состоянии каждого самоката, а также менять цены на услуги в зависимости от спроса.

Таким образом, разработка данной системы требует комплексного подхода и включает в себя решение задач как на программном, так и на аппаратном уровне. Однако, реализация проекта позволит создать удобную и экологически чистую систему аренды электросамокатов, которая будет полезна как для жителей города, так и для его гостей.

Данный проект будет строиться на клиент-серверной архитектуре, где клиентом выступит мобильное приложение, а сервером – RESTful сервис на Java с использованием фреймворка Spring. Клиент-серверная архитектура позволит разделить приложение на две части, что позволит повысить его масштабируемость и удобство сопровождения. Кроме того, REST API позволит создать универсальный интерфейс для взаимодействия между клиентом и сервером, что сделает приложение более гибким и расширяемым [3].

REST API – это набор правил и протоколов, которые определяют стандарты взаимодействия между клиентом и сервером. Он позволяет клиенту отправлять запросы на сервер и получать ответы в формате JSON. REST API является универсальным и гибким интерфейсом, который может быть использован для взаимодействия с любым типом сервера.

При создании мобильного приложения будет использоваться фреймворк Flutter Flow. Flutter Flow позволяет создавать красивые и функциональные мобильные приложения за короткое время. Его основными преимуществами являются: быстрое создание интерфейса с помощью визуального редактора, возможность переиспользования компонентов и легкая интеграция с RESTful сервисом на Java [4].

Использование Java и фреймворка Spring для создания RESTful сервиса, а также использование Flutter Flow для создания мобильного приложения, позволят создать удобную и функциональную систему краткосрочной аренды электросамокатов. Кроме того, клиент-серверная архитектура и REST API сделают приложение более гибким и расширяемым, что позволит его успешно развиваться в будущем.

Литература

1. Электросамокат [Электронный ресурс] / Свободная энциклопедия «Wikipedia». – Режим доступа: <https://ru.wikipedia.org/wiki/Электросамокат>. – Дата доступа: 21.03.2023.
2. Кикшеринг [Электронный ресурс] / Свободная энциклопедия «Wikipedia». – Режим доступа: <https://ru.wikipedia.org/wiki/Кикшеринг>. – Дата доступа: 21.03.2023.
3. REST [Электронный ресурс] / Свободная энциклопедия «Wikipedia». – Режим доступа: <https://ru.wikipedia.org/wiki/REST>. – Дата доступа: 21.03.2023.
4. Java [Электронный ресурс] / Свободная энциклопедия «Wikipedia». – Режим доступа: <https://ru.wikipedia.org/wiki/Java>. – Дата доступа: 21.03.2023.

А. П. Барабанов

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **Е. М. Березовская**, канд. физ.-мат. наук, доцент

РАЗРАБОТКА WEB-ПРИЛОЖЕНИЯ «БИБЛИОТЕКА» НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON С ПРИМЕНЕНИЕМ ФРЕЙМВОРКА DJANGO

В современном мире обычные приложения отходят на второй план. Все мы помним, как какое-нибудь приложение необходимо было установить для его дальнейшего использования, что отнимало достаточно много времени, особенно, если в процессе инсталляции появлялись