

**М. С. Городнянская**  
(ГГУ имени Ф. Скорины, Гомель)  
Науч. рук. **М. А. Подалов**, ст. преподаватель

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ СКАНИРОВАНИЯ И ГЕНЕРАЦИИ QR-КОДОВ**

В современном мире QR-коды стали неотъемлемой частью повседневной жизни, применяясь в самых различных сферах — от коммерции и маркетинга до образования и здравоохранения. Благодаря простоте использования и возможности быстрого считывания информации, QR-коды помогают значительно упростить процесс обмена данными. Актуальность темы разработки приложения для сканирования и генерации QR-кодов обусловлена высокой востребованностью технологий, которые позволяют быстро и безопасно передавать информацию.

Целью данного проекта является создание приложения, способного сканировать и генерировать QR-коды, обеспечивая быстрое действие, надежность и удобство использования.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Изучить существующие подходы к разработке приложений для работы с QR-кодами.
2. Разработать функционал сканирования QR-кодов с помощью камеры мобильного устройства.
3. Реализовать возможность генерации QR-кодов на основе введенной пользователем информации.
4. Обеспечить удобный и интуитивно понятный интерфейс, который позволит пользователю быстро получать доступ к функциям приложения.

Таким образом, разработка данного приложения представляет собой не только исследовательскую, но и практическую задачу, направленную на удовлетворение современных потребностей пользователей в удобных и эффективных средствах обмена данными.

Мобильная разработка представляет собой один из самых востребованных сегментов в сфере информационных технологий, поскольку число пользователей мобильных устройств и спрос на мобильные приложения продолжают стремительно расти. Разработка мобильных приложений позволяет создавать удобные и функциональные решения, которые могут работать на различных платформах, таких как Android и iOS.

На сегодняшний день существуют несколько подходов к разработке мобильных приложений:

1. Нативная разработка — это создание приложений, предназначенных для определенной платформы (Android или iOS) с использованием платформенных языков и инструментов. Для Android это, как правило, Java или Kotlin и среда Android Studio, а для iOS — Swift или Objective-C с использованием Xcode. Нативные приложения обеспечивают высокий уровень производительности и доступ к полному функционалу устройства, но требуют разработки отдельных версий для каждой платформы, что увеличивает затраты времени и ресурсов.
2. Кроссплатформенная разработка — подход, при котором приложение разрабатывается сразу для нескольких платформ с использованием единого кода. Среди популярных кроссплатформенных фреймворков можно выделить Flutter, React Native и Xamarin. Flutter, разработанный компанией Google, особенно выделяется благодаря высокому уровню производительности, богатому набору виджетов и возможности создавать интерфейсы, идентичные для Android и iOS.
3. Гибридные приложения — приложения, созданные с помощью веб-технологий, таких как HTML, CSS и JavaScript, и работающие внутри оболочки, имитирующей нативное приложение. Такие решения позволяют снизить затраты на разработку, но часто

уступают нативным и кроссплатформенным приложениям в производительности и доступе к платформенным функциям [1–3].

Для разработки приложения для сканирования и генерации QR-кодов была выбрана среда разработки, такая как Android Studio, которая предлагает интеграцию с Flutter SDK, удобные инструменты отладки и профилирования, а также поддержку эмуляторов Android и iOS [4–5].

Организует переходы между экранами, обеспечивая интуитивное взаимодействие пользователя с приложением (рисунок 1).

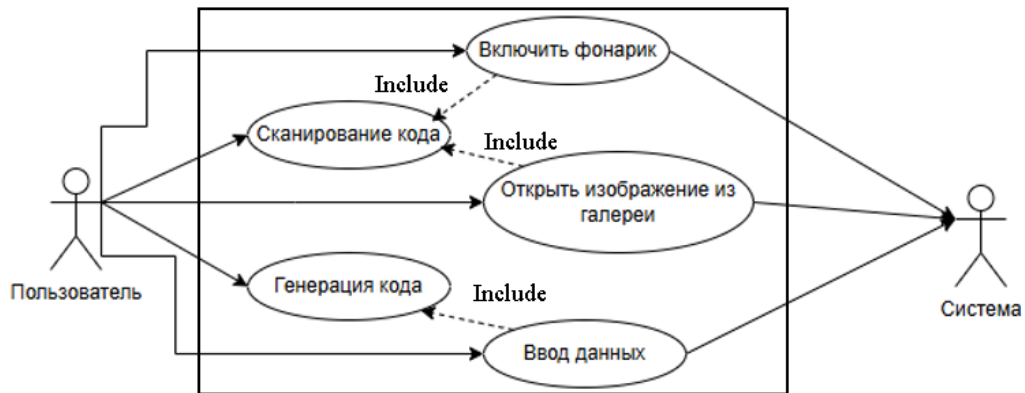


Рисунок 1 – Схема прецедентов

Диаграмма, показывающая основные модули приложения (сканирование, генерация, пользовательский интерфейс, обработка ошибок) и их взаимодействие друг с другом продемонстрирована на рисунке 2.

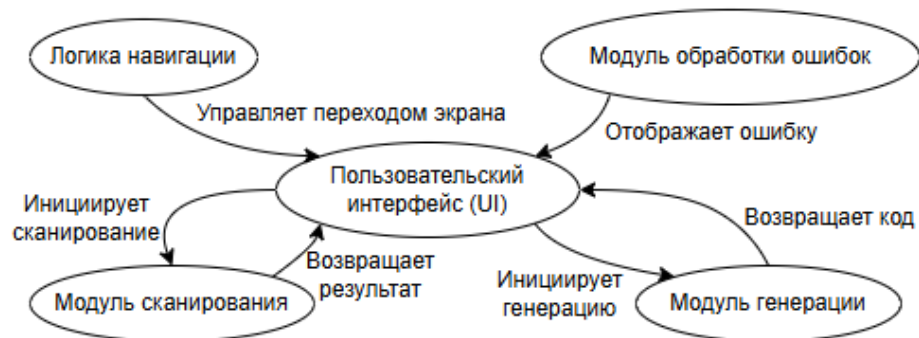


Рисунок 2 – Общая архитектура (декомпозиция) приложения

Навигация организована так, чтобы пользователь мог легко перемещаться между экранами (рисунок 3).

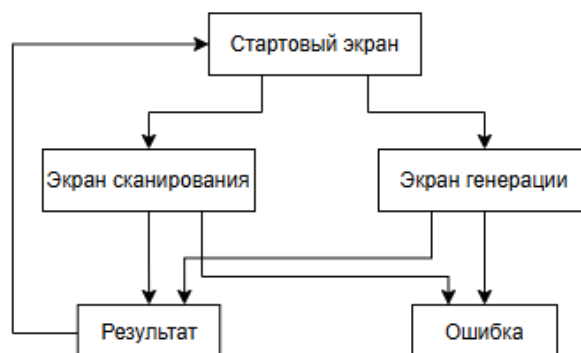


Рисунок 3 – Диаграмма экранов (UI Flow)

Пользовательский интерфейс (UI) приложения предоставляет пользователю доступ ко всем основным функциям. Все действия пользователя происходят через элементы интерфейса, такие как кнопки, поля для ввода и отображение результата (рисунок 4).

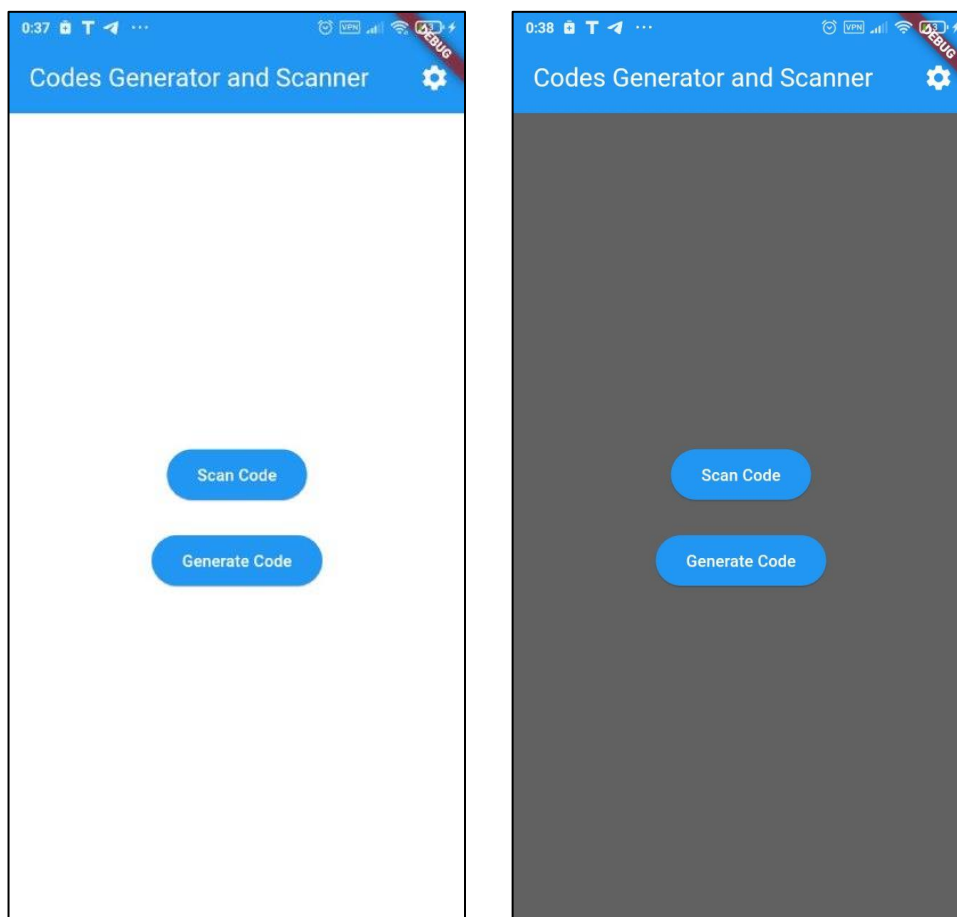


Рисунок 4 – Главный экран: светлая и темная темы

В ходе выполнения проекта было разработано мобильное приложение, способное сканировать и генерировать QR- и штрих-коды, предоставляя удобный пользовательский интерфейс и необходимые функции для работы с кодами. Использование технологии Flutter позволило обеспечить кроссплатформенность приложения, что делает его доступным для пользователей как на устройствах Android, так и на iOS.

### Литература

1. Нативная разработка мобильных приложений / пер. с англ. А. Н. Киселева. – М. : ДМК Пресс, 2020. – 376 с.: ил.
2. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения. Часть 1 : учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова ; Южный федеральный университет. – Ростов-на-Дону ; Таган- рог : Издательство Южного федерального университета, 2019. – 139 с.

3. Васильев, Н. П. Введение в гибридные технологии разработки мобильных приложений : учебное пособие для вузов / Н. П. Васильев, А. М. Заяц. – 3-е изд., стер. – Санкт-Петербург : Лань, 2022. – 160 с.

4. Колисниченко, Д. Н. Программирование для Android. Самоучитель. 3-е изд., перераб. и доп. / Д. Н. Колисниченко. – СПб. : БХВ-Петербург, 2021. – 288 с.

5. Бурнет, Э. Привет, Android! Разработка мобильных приложений / Э. Бурнет. – СПб. : Питер, 2012. – 256 с.