Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Лабораторная работа №2

«Планирование путешествий»

Студент М.А. Бекетова

Проверил О.М. Внук

МИНСК 2024

**1 ЦЕЛЬ РАБОТЫ**

Целью работы является:

– определение и разработка адаптеров (android.widget.AdapterView), которые будут использоваться в приложении;

– реализация страницы Авторизации в мобильном приложении.

**2 ИСХОДНЫЕ ДАННЫЕ К РАБОТЕ**

Среда разработки Android Studio.

Язык программирования Kotlin

Источник, содержащий исходный код: https://github.com/byaketava/Points

# 3 ТЕОРИТИЧЕСКИЕ СВЕДЕНИЯ

Android Studio — это официальная интегрированная среда разработки (IDE) для создания приложений на платформе Android. Она основана на IntelliJ IDEA и предоставляет широкий набор инструментов для разработки, тестирования и отладки приложений. Основные компоненты Android Studio включают редактор кода, который поддерживает синтаксис Java и Kotlin и предлагает автозаполнение, рефакторинг и подсветку синтаксиса. Дизайнер интерфейсов позволяет визуально разрабатывать пользовательские интерфейсы с использованием XML, а эмулятор Android дает возможность тестировать приложения на виртуальных устройствах, эмулируя различные конфигурации и версии Android. Кроме того, Android Studio включает инструменты для анализа производительности, которые помогают выявлять проблемы и оптимизировать приложения.

Kotlin — это современный язык программирования, который стал официальным языком разработки Android-приложений. Разработанный компанией JetBrains, он полностью совместим с Java, что позволяет использовать существующие библиотеки и фреймворки. Kotlin предлагает более строгую проверку типов, что уменьшает количество ошибок во время выполнения, а его лаконичный синтаксис позволяет писать меньше кода для выполнения тех же задач. Кроме того, язык поддерживает функциональные концепции, такие как высшие функции и лямбда-выражения, что делает его более гибким и мощным инструментом для разработчиков.

Мобильная разработка охватывает процесс создания программного обеспечения для мобильных устройств, таких как смартфоны и планшеты. Основные платформы включают Android и iOS, каждая из которых требует знания специфических языков и инструментов. Важными аспектами мобильной разработки являются создание интуитивно понятного интерфейса, работа с различными источниками данных, такими как базы данных и API, а также тестирование на разных устройствах и версиях операционных систем. Оптимизация производительности приложений для работы на устройствах с ограниченными ресурсами также является ключевым моментом.

Работа со списками является неотъемлемой частью мобильной разработки. В Android для отображения списков используются компоненты, такие как ListView и RecyclerView. ListView позволяет отображать вертикальный список элементов и использует адаптер для связывания данных с представлением, в то время как RecyclerView является более гибким и производительным компонентом, поддерживающим различные макеты и улучшенную производительность благодаря переработанному механизму повторного использования элементов.

Адаптеры играют важную роль в связывании данных с представлением, управляя созданием и отображением элементов списка. Существует несколько типов адаптеров, включая ArrayAdapter для простых списков и Custom Adapter, который создается для специфических требований к представлению элементов списка.

**4 РАЗРАБОТКА ПРОЕКТА**

**4.1 Страница авторизации**

Страница авторизации является ключевым компонентом в мобильных приложениях, так как она предоставляет пользователям возможность входа в систему и доступа к защищённому контенту.

В центре страницы находится заголовок, представленный с помощью элемента TextView, который информирует пользователя о необходимости ввести свои данные для входа. Это помогает пользователю ориентироваться в интерфейсе. Два поля ввода, реализованные через EditText, позволяют пользователю ввести логин и пароль. Для поля пароля используется свойство `inputType="numberPassword"`, что скрывает вводимые символы, обеспечивая дополнительную безопасность.

Кнопка "Войти", представленная элементом Button, инициирует процесс аутентификации, отправляя введённые данные на сервер для проверки. Она является важным элементом, так как именно с её помощью пользователь завершает процедуру авторизации.

Кроме того, имеется интерактивный TextView для регистрации, который направляет пользователей на страницу регистрации, что особенно важно для новых пользователей. Наличие такого элемента повышает удобство использования приложения, облегчая процесс создания учётной записи.

Логотип приложения, представленный в виде ImageView, помогает пользователям визуально идентифицировать его, что способствует формированию бренда и улучшает общее восприятие приложения.

Таким образом, страница авторизации, включает в себя все необходимые элементы для создания удобного и функционального интерфейса. Использование таких компонентов, как TextView, EditText и Button, обеспечивает интуитивное взаимодействие пользователя с приложением. Внедрение списков, таких как выпадающие списки для выбора языка, улучшает взаимодействие пользователя с приложением и повышает его доступность, что особенно важно для создания приложений, использующихся людьми из разных стран и культур. На рисунке 4.1 изображена страница авторизации.

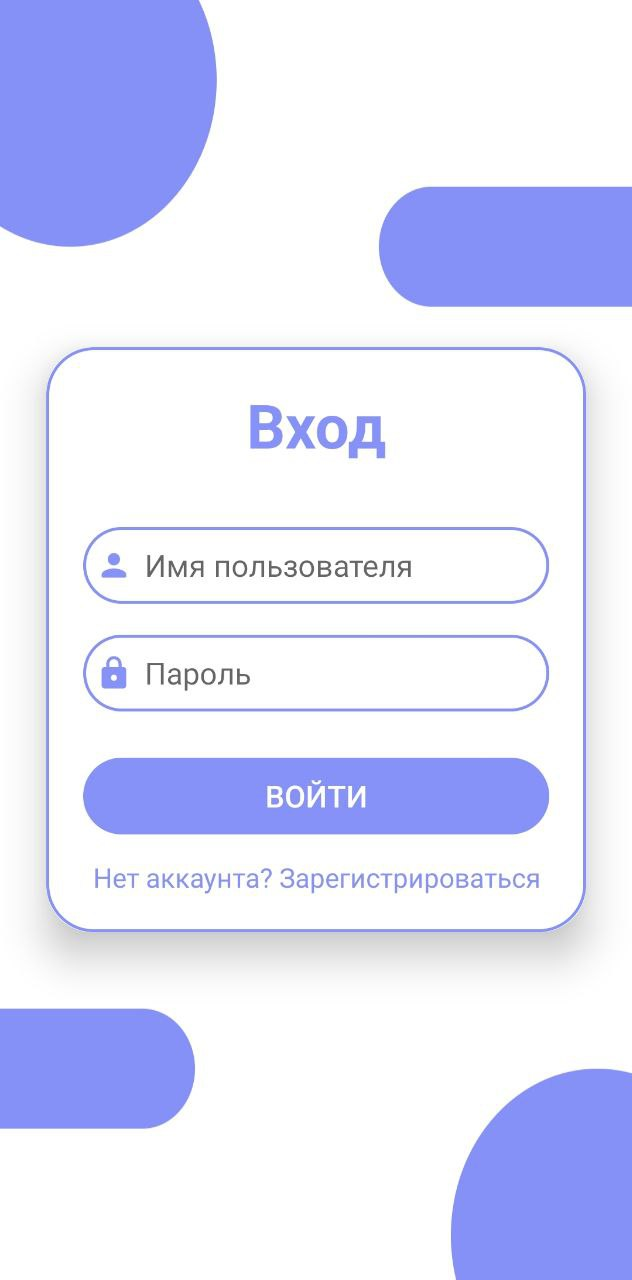


Рисунок 4.1 – Страница авторизации

**4.2 Страница регистрации**

Страница регистрации также является важной частью мобильного приложения, так как она предоставляет пользователям возможность создать новую учётную запись и получить доступ к функционалу приложения.

В центре страницы находится заголовок, который отображается с помощью элемента TextView и информирует пользователей о том, что они находятся на странице регистрации. Это помогает сразу понять, что именно нужно сделать. Поле ввода имени, реализованное через EditText, позволяет пользователям ввести своё имя и содержит подсказку, облегчающую понимание, что необходимо ввести. Аналогично, поле для логина или адреса электронной почты также представлено через EditText, что делает процесс регистрации более понятным и удобным.

Два поля для ввода пароля, одно для его создания и одно для подтверждения, обеспечивают безопасность, позволяя пользователю вводить пароль дважды для избежания ошибок. Эти поля настроены так, чтобы скрывать вводимые символы, что важно для защиты личной информации. Кнопка "Создать учётную запись" инициирует процесс регистрации, отправляя введённые данные на сервер. Она выделяется благодаря стилизованному фону и заметному тексту, что делает её привлекательной для пользователя.

Таким образом, страница регистрации включает в себя все необходимые элементы для создания удобного и функционального интерфейса. Каждый компонент, от заголовка до полей ввода и кнопок, работает вместе, чтобы обеспечить пользователю интуитивно понятный процесс регистрации. Использование таких элементов, как TextView, EditText и Button, а также взаимодействие с элементами, такими как ImageView, создаёт привлекательный и эффективный интерфейс. Это особенно важно для обеспечения положительного пользовательского опыта и повышения вероятности того, что новые пользователи завершат процесс регистрации и начнут использовать приложение. На рисунке 4.2 изображена страница регистрации.

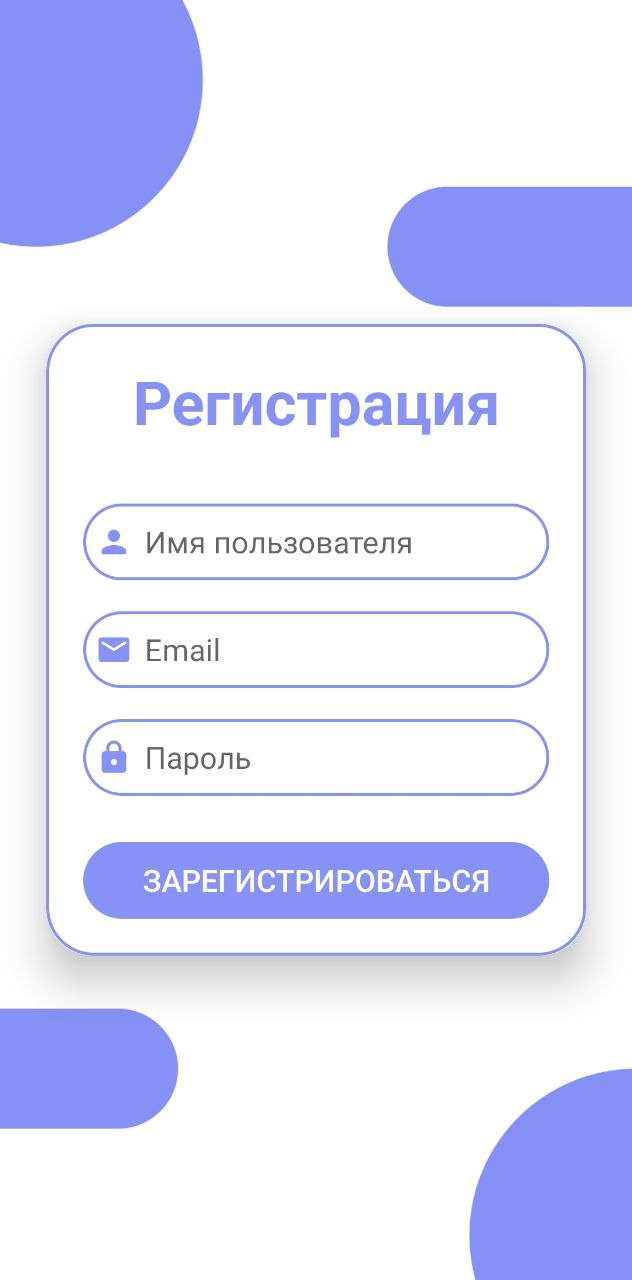


Рисунок 4.2 – Страница регистрации

**4.3 Иконка приложения**

Иконка мобильного приложения — это первое, что видит пользователь при поиске приложения. Она помогает установить визуальную идентичность бренда, привлекает внимание и вызывает интерес. Хорошо продуманная иконка может значительно повысить узнаваемость приложения.

Иконка приложения «Пункты» изображена на рисунке 4.3.



Рисунок 4.3 – Иконка приложения

**5 ВЫВОДЫ**

В ходе разработки страниц авторизации и регистрации, можно сделать несколько важных выводов о проделанной работе.

Во-первых, каждая из страниц была тщательно спроектирована с учетом потребностей пользователей, что позволяет обеспечить интуитивно понятное взаимодействие с приложением. Элементы интерфейса, такие как TextView, EditText и Button, были использованы эффективно, чтобы предоставить пользователям необходимые функции с минимальными усилиями.

Во-вторых, логичное и аккуратное размещение элементов интерфейса в рамках `ConstraintLayout` обеспечивает адаптивность и гибкость дизайна. Это важный аспект, учитывая разнообразие мобильных устройств на рынке.

В-третьих, использование визуальных элементов, таких как ImageView, добавляет эстетическую ценность страниц и способствует формированию бренда приложения. Это улучшает общее восприятие и делает приложение более привлекательным для пользователей.

Наконец, разработка страниц с возможностью добавления настроений, деятельности, заметок и установления целей демонстрирует понимание потребностей целевой аудитории. Это не только делает приложение более функциональным, но и способствует повышению вовлеченности пользователей, что является важным для успеха любого мобильного приложения.