

Разработка учебного прототипа погодного приложения для ОС Android

Павлов Артем Дмитриевич

Приамурский государственный университет им. Шолом-Алейхема

Студент

Андриенко Иван Сергеевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В статье рассматривается процесс разработки базового прототипа погодного приложения для операционной системы Android. Приложение реализует выбор города из предустановленного списка и отображение фиксированных погодных данных. Для хранения последнего выбранного пользователем города используется механизм SharedPreferences. В качестве дополнительного элемента интерфейса реализовано всплывающее информационное окно (AlertDialog). Разработка выполнена в среде Android Studio с использованием языка Java.

Ключевые слова: Android, погодное приложение, SharedPreferences, AlertDialog, пользовательский интерфейс, прототип.

Development of a Weather Application Prototype for Android OS

Pavlov Artem Dmitrievich

Sholom-Aleichem Priamursky State University

Student

Andrienko Ivan Sergeevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article describes the development of a basic prototype of a weather application for the Android operating system. The application allows users to select a city from a predefined list and displays fixed weather data. The SharedPreferences mechanism is used to store the last selected city. An informational pop-up window (AlertDialog) is implemented as an additional user interface element. The development was carried out in Android Studio using the Java programming language.

Keywords: Android, weather application, SharedPreferences, AlertDialog, user interface, prototype.

1 Введение

1.1 Актуальность

Мобильные приложения, предоставляющие информацию о погоде, являются одними из самых востребованных среди пользователей. Они обеспечивают быстрый доступ к актуальным метеорологическим данным, что важно для планирования повседневной деятельности. Разработка подобного приложения представляет собой практическую задачу, позволяющую изучить основы Android-разработки: работу с интерфейсом, управление настройками и хранение данных.

1.2 Обзор исследований

Многие современные исследования посвящены созданию и улучшению мобильных приложений для операционной системы Android.

М. О. Демушкин рассматривает интегрированную среду разработки Android Studio, её преимущества и недостатки при создании мобильных приложений. Android Studio обеспечивает удобные инструменты для отладки, визуального проектирования интерфейса и оптимизации кода, что делает её эффективной средой для обучения и практической разработки [1].

Т. А. Крамаренко, Е. В. Фешина и Т. В. Лукьяненко подчеркивают важность создания интеллектуальных мобильных приложений с адаптивным интерфейсом, способных учитывать предпочтения пользователя и обеспечивать комфортное взаимодействие с системой. Роль технологий машинного обучения и удобных интерфейсов имеет большое значение для повышения эффективности мобильных приложений [2].

Н. Р. Рабимов анализирует особенности разработки приложений в Android Studio, описывая архитектуру проектов, структуру кода и принципы взаимодействия с компонентами системы. Отдельное внимание уделяется вопросам адаптации интерфейса под различные версии Android и оптимизации производительности [3].

А. В. Ряхов акцентирует внимание на значении мобильных приложений в корпоративной среде, где они способствуют автоматизации рабочих процессов и улучшению коммуникации внутри предприятий. Особое внимание уделяется необходимости продуманного интерфейса и интеграции приложений с внутренними системами организации [4].

И. Т. Темнов рассматривает разработку концепции мобильных приложений для планирования досуга, отмечая важность визуализации данных и интерактивных элементов для повышения вовлеченности пользователей. Исследование подчеркивает роль дизайна и эргономики интерфейса в формировании положительного пользовательского опыта [5].

Таким образом, проведенный анализ показывает, что ключевыми направлениями современных исследований в области Android-разработки являются совершенствование пользовательского интерфейса, адаптивность, повышение функциональности и использование Android Studio как основной среды программирования и тестирования мобильных приложений.

1.3 Цель исследования

Целью исследования является разработка учебного прототипа погодного приложения для операционной системы Android. Проект демонстрирует основные принципы работы с пользовательским интерфейсом, хранением данных и взаимодействием с пользователем в среде Android Studio.

2 Материалы и методы

Разработка проводилась в среде Android Studio с использованием языка Java. Минимальная версия SDK установлена на API 21 (Android 5.0 Lollipop).

Основные компоненты реализации:

- Spinner — выбор города из списка;
- TextView и ImageView — отображение погодных данных и иконок;
- SharedPreferences — сохранение последнего выбранного города;
- AlertDialog.Builder — создание всплывающего информационного окна.

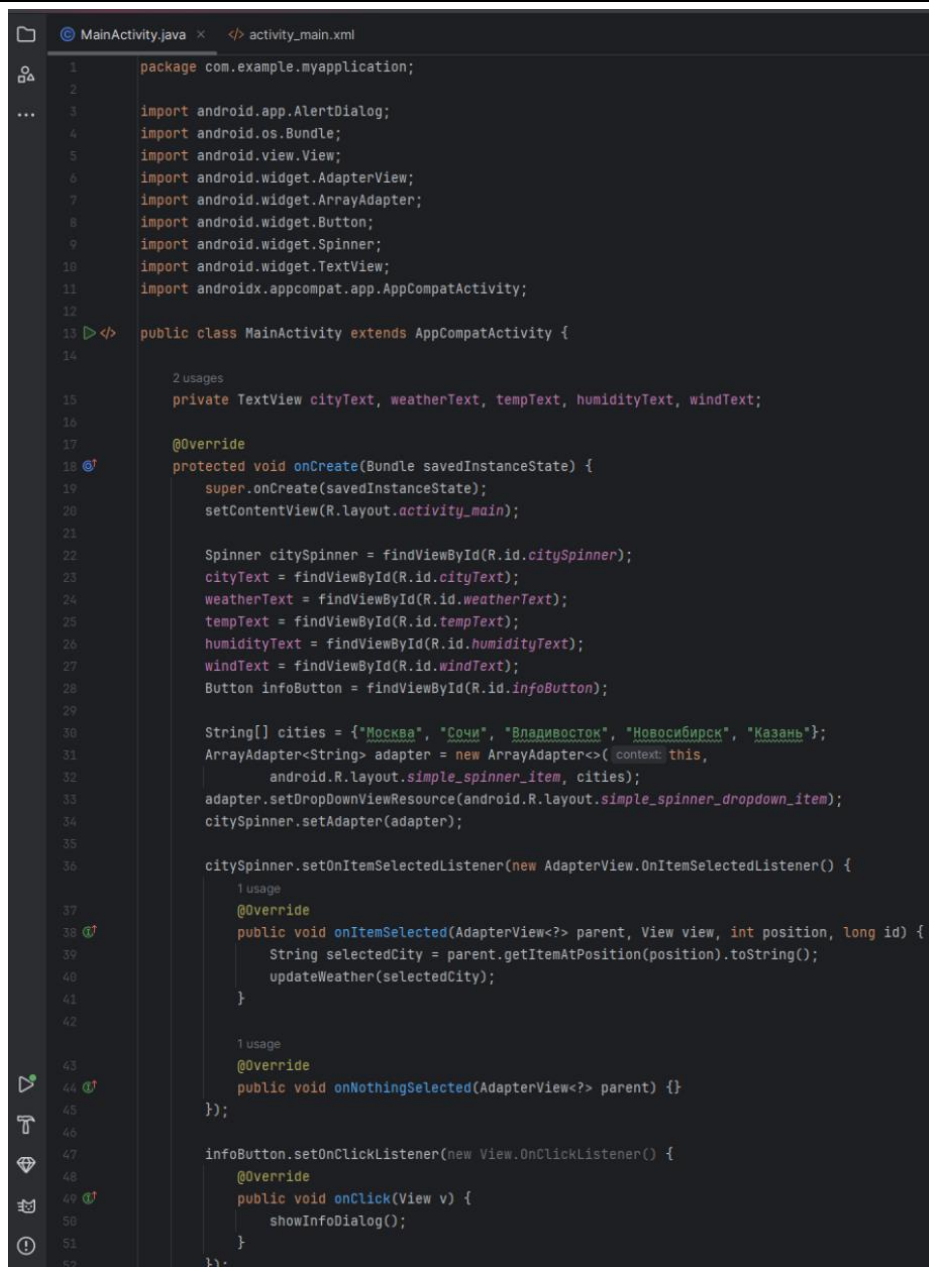
3 Результаты и обсуждения

3.1 Структура проекта и пользовательский интерфейс

Создан новый проект с пустой активностью (Empty Activity). Интерфейс разработан в файле `activity_main.xml` и включает:

- список городов (Spinner),
- кнопки обновления и справки (Button),
- элементы отображения данных (TextView, ImageView).

Макет построен с использованием контейнеров `LinearLayout` и `ConstraintLayout`, обеспечивающих корректное отображение элементов на разных экранах (рис.1).



```
1 package com.example.myapplication;
2
3 import android.app.AlertDialog;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.AdapterView;
7 import android.widget.ArrayAdapter;
8 import android.widget.Button;
9 import android.widget.Spinner;
10 import android.widget.TextView;
11 import androidx.appcompat.app.AppCompatActivity;
12
13 public class MainActivity extends AppCompatActivity {
14
15     2 usages
16     private TextView cityText, weatherText, tempText, humidityText, windText;
17
18     @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21         setContentView(R.layout.activity_main);
22
23         Spinner citySpinner = findViewById(R.id.citySpinner);
24         cityText = findViewById(R.id.cityText);
25         weatherText = findViewById(R.id.weatherText);
26         tempText = findViewById(R.id.tempText);
27         humidityText = findViewById(R.id.humidityText);
28         windText = findViewById(R.id.windText);
29         Button infoButton = findViewById(R.id.infoButton);
30
31         String[] cities = {"Москва", "Сочи", "Владивосток", "Новосибирск", "Казань"};
32         ArrayAdapter<String> adapter = new ArrayAdapter<>(context, this,
33             android.R.layout.simple_spinner_item, cities);
34         adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
35         citySpinner.setAdapter(adapter);
36
37         citySpinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
38             1 usage
39             @Override
40             public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
41                 String selectedCity = parent.getItemAtPosition(position).toString();
42                 updateWeather(selectedCity);
43             }
44
45             1 usage
46             @Override
47             public void onNothingSelected(AdapterView<?> parent) {}
48         });
49
50         infoButton.setOnClickListener(new View.OnClickListener() {
51             @Override
52             public void onClick(View v) {
53                 showInfoDialog();
54             }
55         });
56     }
57 }
```

Рисунок 1. Фрагмент кода MainActivity.java

3.2 Логика приложения

Основная логика реализована в классе MainActivity.java.

При создании активности выполняется привязка элементов интерфейса и настройка адаптера для Spinner. Метод updateWeatherData() обновляет данные о температуре, влажности и ветре, а также меняет иконку в зависимости от выбранного города (рис.2).

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Погода"
    android:textSize="18sp"
    android:textColor="#2C3E50"
    android:textStyle="bold"
    android:layout_above="@id/citySpinner"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="10dp" />

<Spinner
    android:id="@+id/citySpinner"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_above="@id/infoButton"
    android:layout_marginBottom="250dp"
    android:background="@android:color/white"
    android:padding="12dp" />

<Button
    android:id="@+id/infoButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Меню"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="30dp"
    android:backgroundTint="#87CEEB"
    android:textColor="FFFFFF"
    android:paddingLeft="20dp"
    android:paddingRight="20dp"
    android:paddingTop="10dp"
    android:paddingBottom="10dp" />

</RelativeLayout>
```

Рисунок 2. Разметка интерфейса activity_main.xml

3.3 Работа приложения на эмуляторе

Приложение протестировано в эмуляторе Android Studio. При выборе города информация корректно обновляется, а данные сохраняются между сессиями с помощью SharedPreferences, что повышает удобство взаимодействия (рис.3).

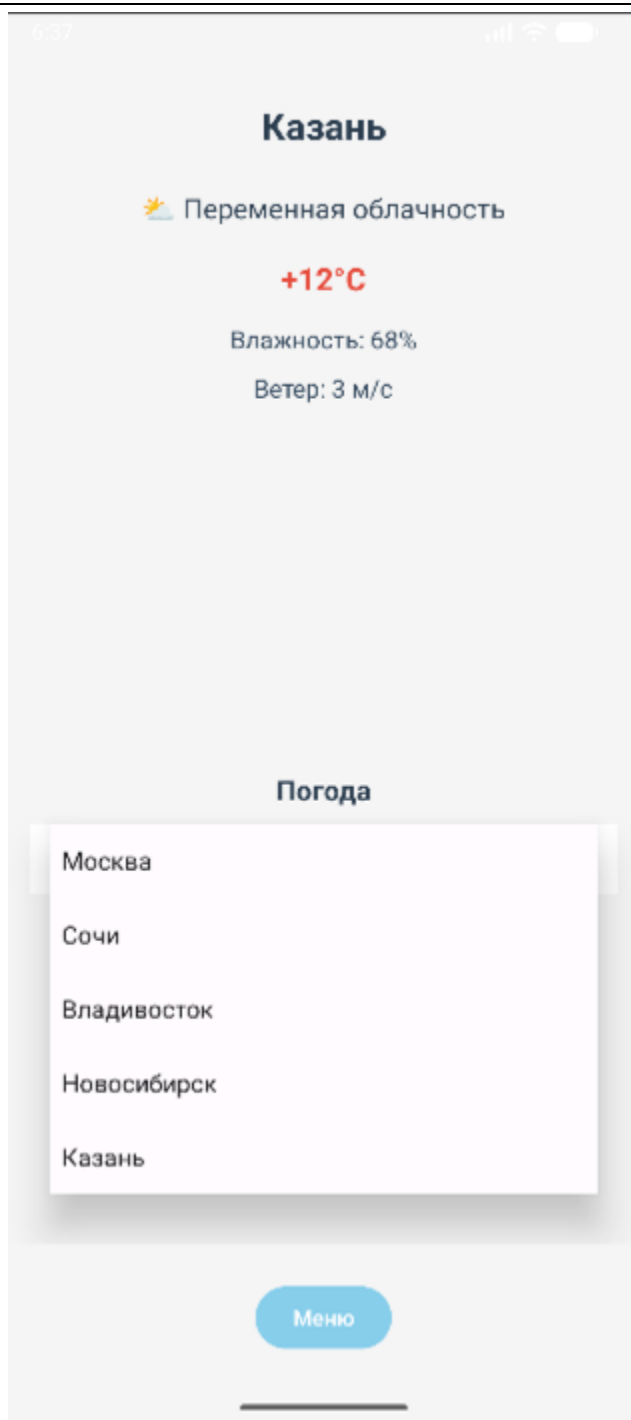


Рисунок 3. Работа приложения на эмуляторе Android Studio

3.4 Визуальная часть интерфейса в Android Studio

Встроенный Layout Editor Android Studio использовался для визуальной проверки интерфейса. Элементы расположены корректно, дизайн адаптирован под разные разрешения экранов (рис. 4).

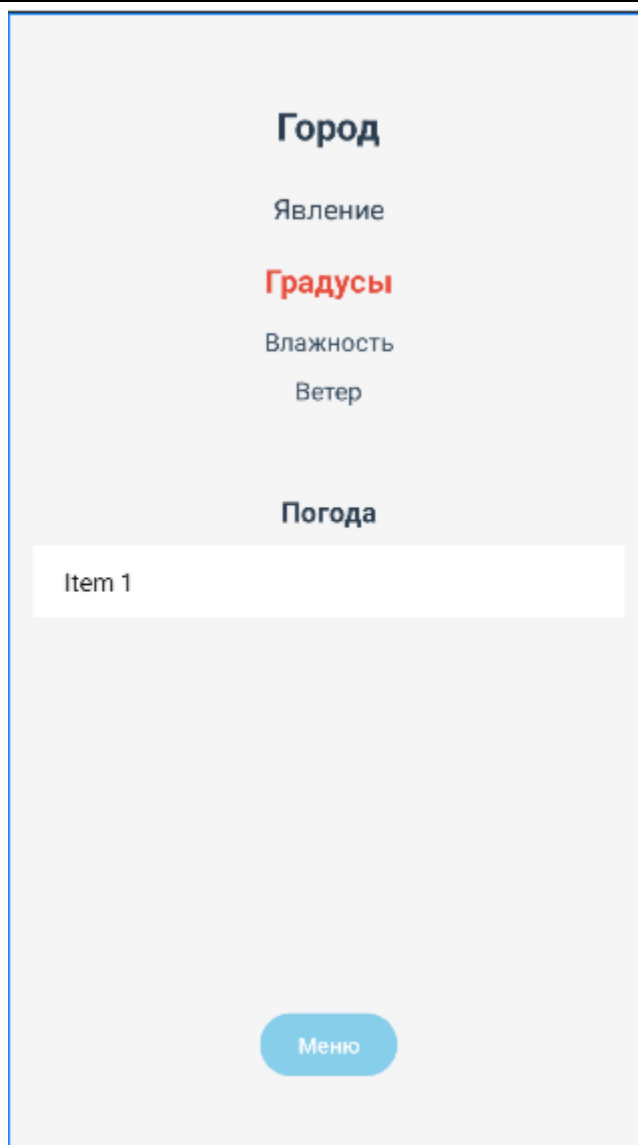


Рисунок 4. Визуальная часть интерфейса Android Studio

4. Вывод

В результате разработки создан учебный прототип погодного приложения для Android, реализующий выбор города, отображение погодных данных и сохранение пользовательских настроек. Приложение демонстрирует стабильную работу интерфейса и может служить основой для дальнейшего расширения — подключения реальных погодных API, добавления геолокации и динамического обновления данных.

Библиографический список

1. Демушкин М. О. Интегрированная среда разработки Android Studio достоинства и недостатки применения // Инновационное развитие науки и образования. 2019. №2(1). С. 119–121. URL: <https://www.elibrary.ru/item.asp?id=38142057>
2. Крамаренко Т. А., Фешина Е. В., Лукьяненко Т.В. Разработка интеллектуальных мобильных приложений // Известия Южного

- федерального университета. Технические науки. 2022. № 2. С. 249–259. URL: <https://cyberleninka.ru/article/n/razrabotka-intellektualnyh-mobilnyh-prilozheniy>
3. Рабимов Н. Р. Мобильные приложения в Android Studio // Академическая публицистика. 2019. № 7. С. 17–22. URL: <https://www.elibrary.ru/item.asp?id=39178464>
 4. Ряхов А. В. Роль разработки мобильных приложений в предприятии // Вестник магистратуры. 2014. №5(32). С. 7–10. URL: <https://cyberleninka.ru/article/n/rol-razrabotki-mobilnyh-prilozheniy-v-predpriyatii>
 5. Темнов И. Т. Разработка концепции приложения для планирования досуга // Шаг в науку. 2025. №2. С. 45–49. URL: <https://cyberleninka.ru/article/n/razrabotka-kontseptsii-prilozheniya-dlya-planirovaniya-dosuga>