

WebView

Червяков Алексей



Наш план

- Что такое webview
- Зачем нам может понадобиться
- Как настраивать
- Где используется

Организационные моменты

Напоминание
отметиться на
портале

Домашнее задание

Тест для проверки материала

- сдать до 27 декабря
- 15 минут
- 5 баллов

WebView

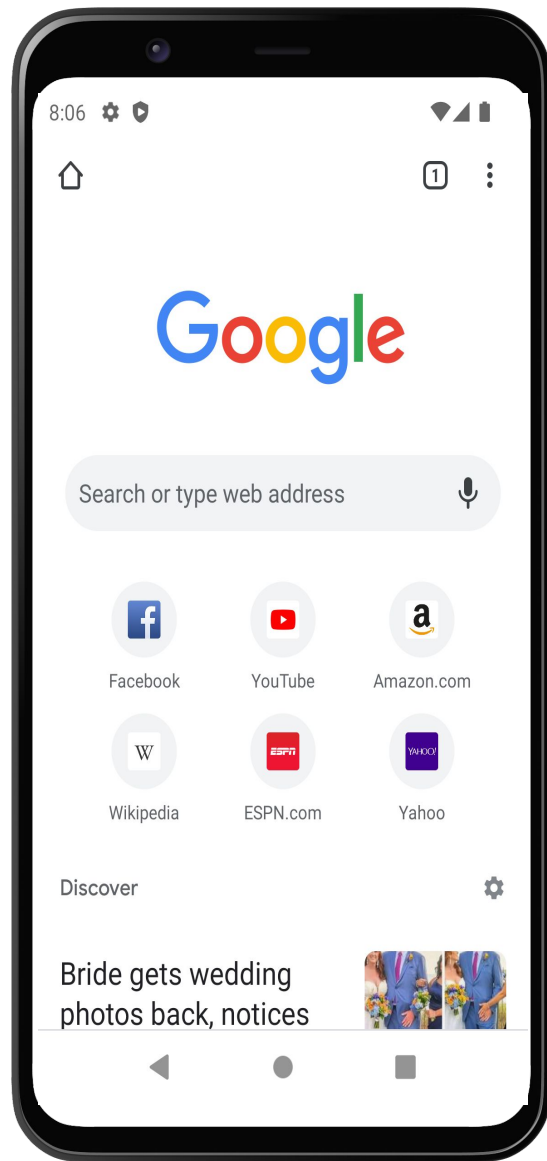
WebView

браузер внутри приложения



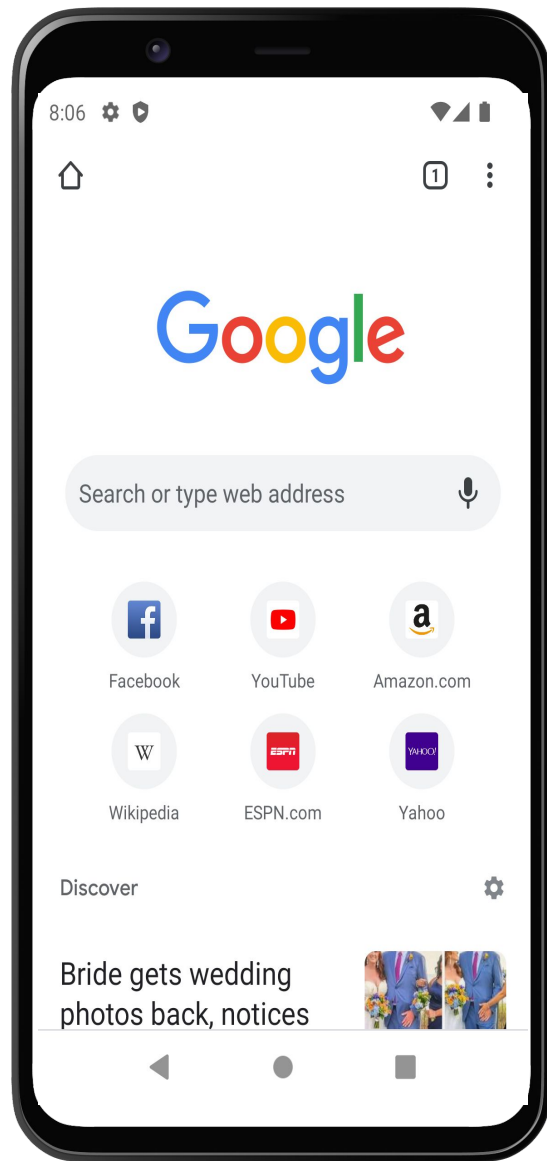
WebView

Это компонент, который позволяет встраивать в наше приложение web страницы



WebView

- Отобразить какую-то дополнительную (второстепенную) информацию
- Отображать информацию, которая должна быть изменяемой
- Имеющийся опросник
- Форма оплаты (?)
- Делаем приложение на основе адаптивного сайта (?)
- Общаться с JS (?)



Из чего состоит webView

```
// manifest
<uses-permission android:name="android.permission.INTERNET" />

// layout
<WebView
    android:id="@+id/webview"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
/>

// load page
webview.loadUrl(url)
```

Из чего состоит WebView

В коде можно еще донстроить WebView под свои нужды (включить JavaScript или сделать ручную обработку переходов)

Дополнительно

```
// advanced
webview.settings.apply {
    javaScriptEnabled = true
}
```

Настройка webView



Собственная обработка переходов

```
webView.apply {  
    webViewClient = SandboxWebClient()  
}  
  
class SandboxWebClient: WebViewClient() {  
    override fun shouldOverrideUrlLoading(view: WebView?, url: String?): Boolean {  
        if (isForbiddenUrl(url)) {  
            Toast.makeText(view!!.context, "Forbidden Url: $url", Toast.LENGTH_LONG).show()  
            return true  
        }  
  
        return false  
    }  
  
    protected fun isForbiddenUrl(url: String?): Boolean {  
        return !url.isNullOrEmpty() && url.startsWith("http")  
    }  
}
```

Safe Browsing

```
<manifest>
    <meta-data
        android:name="android.webkit.WebView.EnableSafeBrowsing"
        android:value="true" />
    . . .
    <application> . . . </application>
</manifest>
```

Использование



Загрузка страницы

```
WebView webView = findViewById(R.id.webView);  
String htmlText = "<html><body>Percent test: 100% </body></html>";
```

```
webView.loadUrl("www.google.ru");  
webView.loadData(htmlText, "text/html", "en_US");  
webView.loadDataWithBaseURL(null, htmlText, "text/html", "en_US",  
null);
```


Взаимодействие Android с функциями страницы

Проще всего построить взаимодействие при помощи вызова JavaScript функций самой страницы. Код самой функции уже может делать изменения на странице.

Необходимо разрешить использование JavaScript.

Для вызова JavaScript-функции поможет метод **evaluateJavascript**. Где первый параметр - это вызов самой функции. Второй параметр - подписка на результат исполнения

Если поискать еще варианты вызова, для старых Android работал такой вариант **webView.loadUrl("javascript:enable();")**.

```
// !!!!
webView.settings.apply {
    javaScriptEnabled = true
}
```

```
// without result
webView.evaluateJavascript("function1()", null)
```

```
// with result
webView.evaluateJavascript("function2('$param')") {
    // return String
}
```

Взаимодействие страницы с Android

Необходимо разрешить использование JavaScript.

Зарегистрировать свой **JavascriptInterface** при помощи метода **addJavascriptInterface**. Вторым параметром метода, **name**, потребуется для того, чтобы дергать методы именно этого интерфейса со страницы.

JavascriptInterface класс создается без наследования. Но методам, к которым будет обращаться JavaScript, надо будет пометить аннотацией **@JavascriptInterface**.

```
// !!!!
webview.settings.apply {
    javaScriptEnabled = true
}

// set callback
webview.apply {
    addJavascriptInterface(CustomWebInterface(), "Android")
}

// callback example
class CustomWebInterface(val activity: Activity) {
    @JavascriptInterface
    fun showToast(message: String) {
        ...
    }
}

// on page
function showAndroidToast(message) {
    Android.showToast(message);
}
```

Вопросы?

Пишем код

Оставьте отзыв!



Спасибо за внимание!

Червяков Алексей