

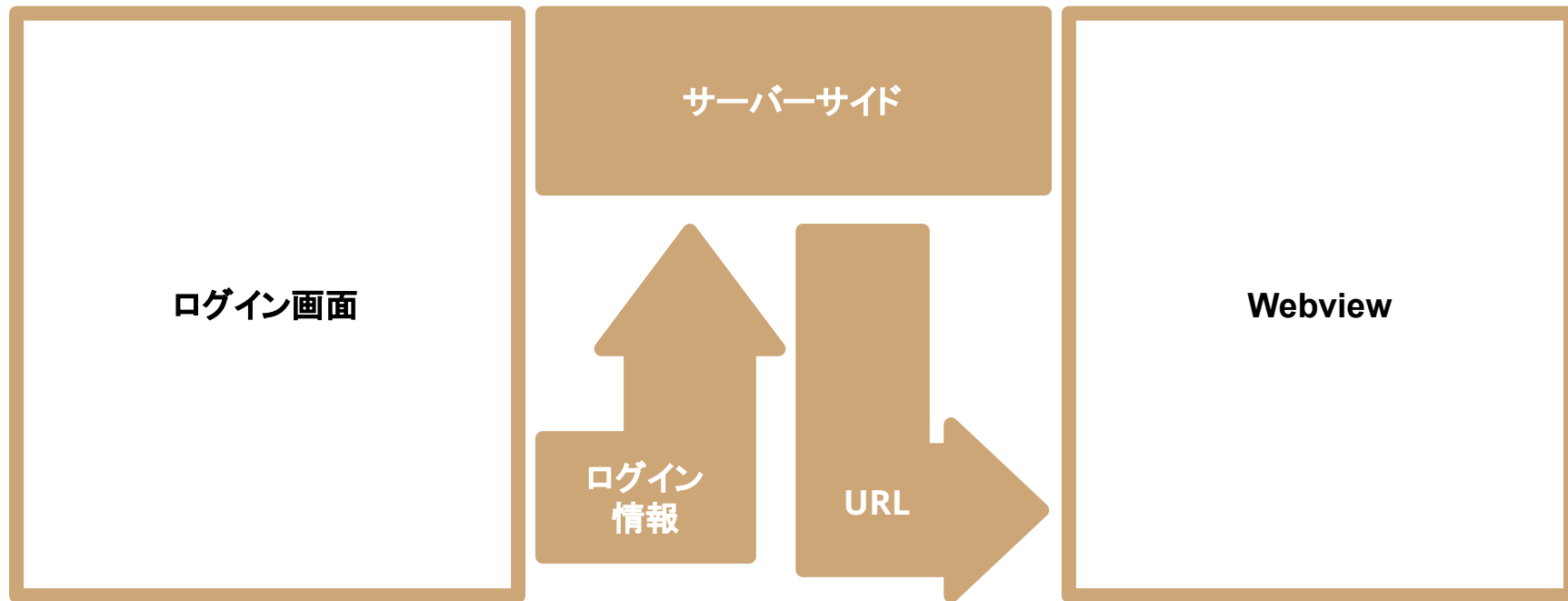
ログイン、 Webview表示だけ のAndroidアプリ (java)を作成する



アプリに携わる前までの知識 (※あまり正しくない情報です)

- 使用する機能の権限宣言が必要
 - manifestファイルに記述していく
- xml ファイルにレイアウトを記述する
 - webview というタグ(?)を追加すれば、webview が配置される
- Activity にメソッドを記述する
 - webview でどの URL を読み込むかとか

作ろうとしたアプリ



Webviewだけのアプリ

レイアウトの xml ファイルに

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

Activity の onCreate() に

```
WebView myWebView = findViewById(R.id.webview);  
myWebView.loadUrl("http://www.example.com");
```

AndroidManifest.xml ファイルに

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

Webview

ログイン画面

ログイン画面

レイアウトの xml ファイルに

```
<EditText android:id="@+id/login" ... />
<EditText android:id="@+id/pass" ... />
<Button android:id="@+id/btn_login" ... />
```

Activity の onCreate() に

```
view.findViewById(R.id.btn_login).setOnClickListener(
    new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            ...
        }
    }
);
```

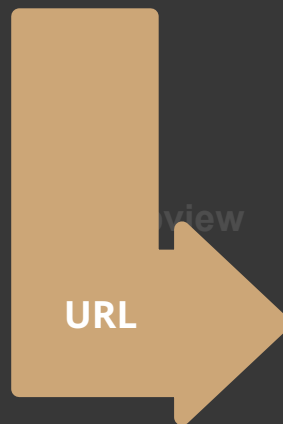
サーバーサイドとの通信部分

```
RequestQueue queue = Volley.newRequestQueue(this);
StringRequest postRequest = new StringRequest(
    Request.Method.POST, "POSTするURL",
    new Response.Listener() { //正常時処理 },
    new Response.ErrorListener(){ //エラー処理 })
{
    @Override
    protected Map<String,String> getParams(){
        Map<String,String> params = new HashMap<>();
        params.put("login","Textviewの値");
        params.put("pass","Textviewの値");
        return params;
    }
};
queue.add(postRequest);
```



サーバーサイドとの通信部分

```
RequestQueue queue = Volley.newRequestQueue(this);
StringRequest postRequest = new StringRequest(
    Request.Method.POST, "POSTするURL",
    new Response.Listener() {
        @Override
        public void onResponse(String response) {
            Intent intent = new Intent(context,
                webViewActivity.class);
            intent.putExtra("url", response);
            startActivity(intent);
        }
    },
    new Response.ErrorListener(){ //エラー処理 }
    { //その他必要な処理 };
    queue.add(postRequest);
```



Webviewだけのアプリ

サーバーサイド

Activity の onCreate() に

```
Intent i = getIntent();  
String url = i.getStringExtra("url");  
WebView myWebView = findViewById(R.id.webview);  
myWebView.loadUrl(url);
```

Webview

情報

URL

まとめ・参考URL

- <https://developer.android.com/codelabs/build-your-first-android-app#0>
 - Android公式のJavaでの最初のアプリを作るチュートリアル。とてもよくできているので、おすすめです。
- <https://developer.android.com/courses?hl=ja>
 - 最初のアプリ以外にもいろいろなチュートリアルが用意されているので、公式サイトだけで勉強ができます。
- <https://developer.android.com/guide/webapps/webview?hl=ja#java>
 - Android公式のWebviewの作り方