Working with Remote Data in the Application

Gill Cleeren
CTO XPIRIT BELGIUM

@gillcleeren www.snowball.be

Overview

Accessing remote data

Using HttpClient from shared code

Accessing Remote Data

Data

- Local
- Remote

Remote via service

Security considerations

Accessing Remote Data

ASMX

WCF

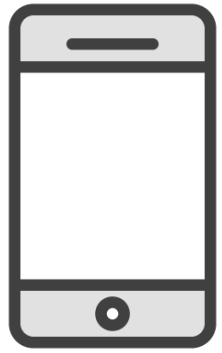
REST

Azure Mobile Apps Service

Working with REST Services

GET

JSON



HTTP Verbs

GET

POST

PUT

DELETE

HTTP Endpoints

Verb	Endpoint
GET	/api/pies
GET	/api/pies/{id}
POST	/api/pies
PUT	/api/pies
DELETE	/api/pies/{id}

Demo

Looking at the REST API

Using HttpClient from Shared Code

HttpClient

- System.Net.HttpClient
- Async
- Receive JSON in HttpResponseMessage
- Used from shared code

```
HttpClient httpClient = new HttpClient();

httpClient.DefaultRequestHeaders.Accept.Add(new
MediaTypeWithQualityHeaderValue("application/json"));
```

HttpClient creation

```
await
httpClient.GetAsync("http://www.bethanyspieshop.com/api/pies");
```

Retrieving Data

Retrieving Data

```
if (!responseMessage.IsSuccessStatusCode) return null;

var jsonResult = await
responseMessage.Content.ReadAsStringAsync().ConfigureAwait(false);

var pies =
JsonConvert.DeserializeObject<IEnumerable<Pie>>(jsonResult);
```

```
await
client.PostAsync("http://www.bethanyspieshop.com/api.pies",
content);
```

Adding Data

Demo

Working with server-side data

Adding support for HttpClient

Summary

Xamarin has rich support for accessing services

REST in combination with HttpClient is preferred way

Up next:
Working with native device features