Software Engineering - Using JSON Web APIs in Dart

L.EIC-ES-2021-22

J. Pascoal Faria, Ademar Aguiar

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Agenda

- Motivation
- JSON
- Parsing JSON in Dart
- JSON Web APIs
- Accessing JSON Web APIs in Dart
- Accessing protected data
- Further links

Motivation

- The UNI app interacts with two types of SIGARRA resources:
  - Web Pages in HTML
  - Web Services (Web APIs) in JSON
- The UNI4ALL backend services that are being developed by master students (in Software Systems Architecture) also return data in JSON
- Whenever feasible and needed, you are expected to replace your current mocks by calls to that backend
- So, it is important to understand how to consume JSON Web APIs in Dart/Flutter

JSON

- JSON stands for **J**ava**S**cript **O**bject **N**otation
- JSON is a **text format** for storing and transporting data
- JSON is "self-describing" and easy to understand
- Example:
  - {"name":"John", "age":30, "car":null}
  - It defines an object with 3 properties (name, age, car) with corresponding values
- Further info:
  - https://www.w3schools.com/js/js_json_intro.asp

Parsing JSON in Dart

- The conversion of a JSON string to objects in Dart is a process called ***deserialization*** (the opposite is *serialization)*
- **dart:convert** - provides JSON (de)serialization features
  - **json.decode(*str*)** – deserializes a JSON string into **List**s and **Map**s

json_demo1.dart

```
import 'dart:convert';

void main() {
  var jsonString = '''[{"name": "john", "score": 40},
                       {"name": "mary", "score": 80}]''';
  var scores = json.decode(jsonString); // List
  for (var score in scores) { // Map
    print('The score of ${score['name']} is ${score['score']}');
  }
}
```

```
The score of john is 40
The score of mary is 80
```

JSON Web APIs

- Many Web APIs retrieve data in JSON format
- E.g., occupancy information of FEUP's parking lots can be accessed via the following url:
  - https://sigarra.up.pt/feup/pt/instalacs_geral.ocupacao_parques
- Example result:
  - {"itdc":[{"versao":"bo-1.1.0","dataresposta":"2022-05-16T12:52:32+02:00","codigoresposta":"SOP1","statementsql":"","linhas":"","erro":"","resposta":{"p1lotacao":525,"p3lotacao":325,"p4lotacao":50,"data":"20220516","P1in":278,"P1out":20,"P3in":356,"P3out":53,"P4in":36,"P4out":1,"p1ocupados":258,"p3ocupados":303,"p4ocupados":35,"p1livres":267,"p3livres":22,"p4livres":15}}]}
  - Hint: Paste and format in VSCode for easier reading

Accessing JSON Web APIs in Dart

json_demo2.dart

```dart
import 'dart:convert';
import 'package:http/http.dart' as http;

const parkingUrl =
    'https://sigarra.up.pt/feup/pt/instalacs_geral.ocupacao_parques';

void main() async {
  var response = await http.get(Uri.parse(parkingUrl));
  if (response.statusCode == 200) {
    var data = json.decode(response.body);
    var parkInfo = data["itdc"][0]["resposta"];
    parkInfo.forEach((k,v) => print('${k}: ${v}'));
  } else {
    throw Exception('Failed to read $parkingUrl');
  }
}
```

```
p1lotacao: 525
p3lotacao: 325
p4lotacao: 50
data: 20220516
P1in: 281
P1out: 32
P3in: 371
P3out: 78
P4in: 37
P4out: 4
p1ocupados: 249
p3ocupados: 293
p4ocupados: 33
p1livres: 276
p3livres: 32
p4livres: 17
```

Accessing protected data

- Some services of Web APIs may require authentication
- E.g., in SIGARRA, the schedule of a student (previously authenticated) may be retrieved in JSON from the following URL:
  - https://sigarra.up.pt/feup/pt/mob_hor_geral.estudante?pv_codigo=NNNNNNNN&pv_semana_ini=AAAAMMDD&pv_semana_fim=AAAAMMDD
- The following URL can be used for login / authentication:
  - https://sigarra.up.pt/feup/pt/mob_val_geral.autentica?pv_login=....&pv_password=...
  - Retrieves a JSON string with error/success information an user code & type
- The **cookie** retrieved on login (in field 'set-cookie' of response header), should be passed (sanitized) in calls to services that require authentication (in field 'cookie' of request header)
- A possible implementation in Dart is shown in the next slides

Accessing protected data - Session state

- The behavior for passing cookies (session state) between HTTP requests can be encapsulated in a “Session” object.

json_demo4.dart

```dart
// Allows passing cookies between HTTP requests.
class Session {
  Map<String, String> _headers = {}; // stores the cookie

  // Posts data to a url and retrieves the deserialized JSON response.
  // Uses a previously retrieved cookie in the request header, if existent.
  // Saves the cookie retrieved in the response header, if provided.
  Future<dynamic> post(String url, dynamic data) async {...}

  // Retrieves the (deserialized) JSON response from a url.
  // Uses a previously retrieved cookie in the request header, if existent.
  Future<dynamic> get(String url) async {...}

  // Saves in "headers" the cookie retrieved in the response header.
  void _updateCookie(http.Response response) {...}
}
```

Accessing protected data – Session usage

json_demo4.dart

```dart
Session s = new Session();
User? user = await login(s, username, password);
if (user != null) {
  var schedule = await getUserSchedule(s, user, "20220515", "20220521");
  print(schedule);
}
```

```dart
Future<User?> login(Session session, String username, String password) async {
  var data = await session.post(loginUrl,
      {'pv_login': username, 'pv_password': password});
  if (data["authenticated"]) {
    return User.fromJson(data);
  } else {
    return null;
  }
}
```

Example response data:
{"authenticated":true,
"codigo":"210006",
"tipo":"F"}

```dart
Future<dynamic> getUserSchedule(Session session, User user,
    String dataIni, String dataFim) async {
  var url = (user.type == "F"? teacherScheduleUrl : studentScheduleUrl)
      + "?pv_codigo=${user.code}&pv_semana_ini=$dataIni&pv_semana_fim=$dataFim";
  return await session.get(url);
}
```

Accessing protected data – Session impl.1

json_demo4.dart

```dart
// Allows passing cookies between HTTP requests.
class Session {
  Map<String, String> _headers = {}; // stores the cookie

  // Posts data to a url and retrieves the deserialized JSON response.
  // Uses a previously retrieved cookie in the request header, if existent.
  // Saves the cookie retrieved in the response header, if provided.
  Future<dynamic> post(String url, dynamic data) async {
    var response = await http.post(Uri.parse(url), body: data, headers: _headers);
    if (response.statusCode == 200) {
      _updateCookie(response);
      return json.decode(response.body);
    }
    else {
      throw Exception('Failed to access $url');
    }
  }
```

1

Accessing protected data – Session impl.2

json_demo4.dart

```dart
// Retrieves the (deserialized) JSON response from a url.
// Uses a previously retrieved cookie in the request header, if existent.
Future<dynamic> get(String url) async {
  var response = await http.get(Uri.parse(url), headers: _headers);
  if (response.statusCode == 200) {
    _updateCookie(response);
    return json.decode(response.body);
  }
  else {
    throw Exception('Failed to access $url');
  }
}

// Saves in "headers" the cookie retrieved in the response header.
void _updateCookie(http.Response response) {
  String? rawCookie = response.headers['set-cookie'];
  if (rawCookie != null) {
    var sanitizedCookie = rawCookie.replaceAll(",", ";"); //needed with SIGARRA
    _headers['cookie'] = sanitizedCookie;
  }
}
}
```

4

2

3

Further links

- Obtain canteens info (no authentication required):
  - https://sigarra.up.pt/feup/pt/mob_eme_geral.cantinas
- Obtain the schedule of an occurrence of a curricular unit (authentication required):
  - https://sigarra.up.pt/feup/pt/mob_hor_geral.ucurr?pv_ocorrencia_id=484425&pv_semana_ini=20220227&pv_semana_fim=20220611
  - Example for ES 2021-22
- Obtain a student's checking account (authentication required):
  - https://sigarra.up.pt/feup/pt/mob_ccorrente_geral.conta_corrente?pv_codigo=NNNNNNNNN
- Other backend services to be provided soon by M.EIC's students (Software Systems Architecture)