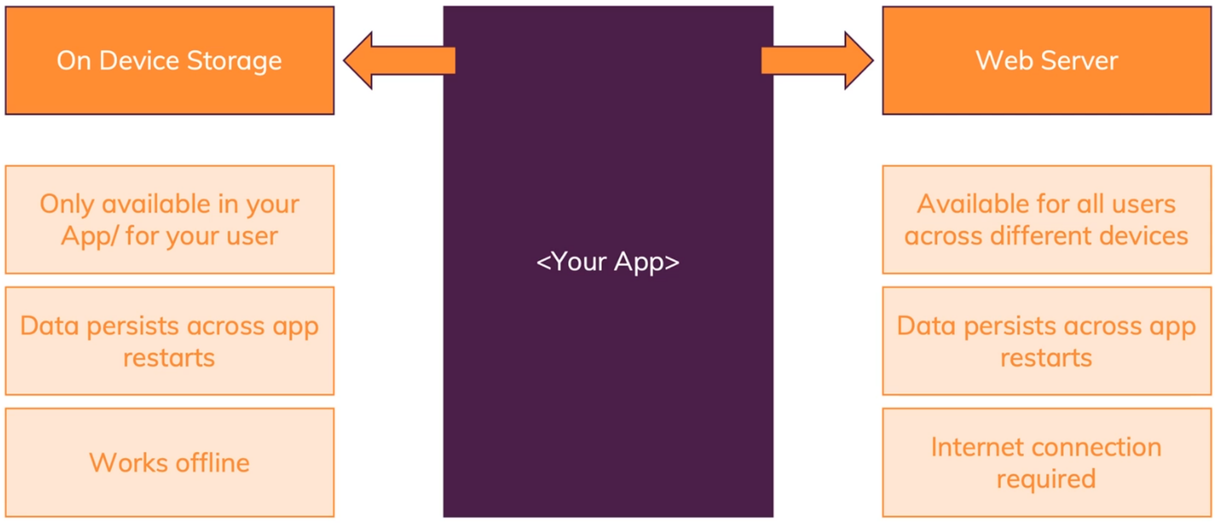
Chapter 9: HTTP Requests

9.1. On-Device vs Web Storage

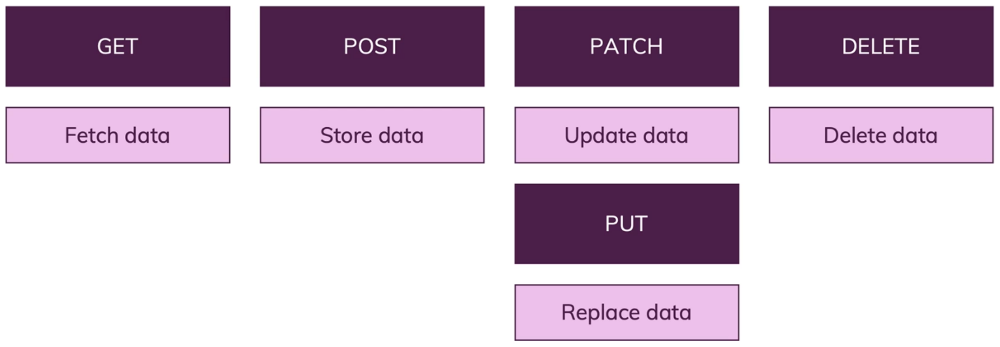

On Device Storage
<Your App>
Web Server
Only available in your App/ for your user
Data persists across app restarts
Works offline
Available for all users across different devices
Data persists across app restarts
Internet connection required

9.2. Http Requests

The internet boasts a vast array of resources hosted on different servers. For you to access these resources, your browser needs to be able to send a request to the servers and display the resources for you. HTTP (Hypertext Transfer Protocol), is the underlying format that is used to structure requests and responses for effective communication between a client and a server. The message that is sent by a client to a server is what is known as an HTTP request. When these requests are being sent, clients can use various methods.

Therefore, HTTP request methods are the assets that indicate the specific desired action to be performed on a given resource. Each method implements a distinct semantic, but there are some standard features shared by the various HTTP request methods.

Http Methods



9.3. Using the Http Package

Nowadays the use of API in an app is very common, almost every app we use in daily life uses API to display the user's data. We will use the http package, which provides advanced methods to perform various operations. REST API uses simple http calls to communicate with JSON data. Each request returns a **Future<Response>.**

Core Methods in http package

1. **post**: This method uses a URL and POST method to post the data and return a Future<Response>. It is used to send the data on the web.
2. **get**: get method uses a URL and fetches the data in JSON format.
3. **delete**: This method is used to delete the data using the DELETE method.
4. **head**: Uses the URL to request the data using the HEAD method and returns the response as Future<Response>.
5. **read**: It is similar to the get method, used to read the data.
6. **put**: It uses the specified URL and updates the data specified by the user using the PUT method.
7. **patch**: It is very similar to the put method, used to update the data.

Setting up the Project

- Install and import the **http** dependency
  - ➔ Add the following dependency in pubspec.yaml file

```
dependencies:
  http:
```

  - ➔ Import the package as http

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

- Creating a Request

```
Future<List<Fruit>> fetchFruit() async {
  final response = await http.get(url);
}
String url = "Your_URL";
```

This is the basic request using get method to fetch the data from the specified URL in the **JSON** format.

- Handling the **error** code

```
final response = await http.get(
      'url');
  if (response.statusCode == 200) {
    //displaY UI
    } else {
    //SHOW ERROR MESSAGE
    }
}
```

If the response status code is 200 then we can display the requested data otherwise we can show the error message to the users

- Decoding the **JSON** data

```
List<Fruit> decodeFruit(String responseBody) {
  final parsed = json.decode(responseBody).cast<Map<String,
dynamic>>();
  return parsed.map<Fruit>((json) => Fruit.fromMap(json)).toList();
}
```

- Creating a Fruit class

```
class Fruit {
  final int id;
  final String title;
  final String imageUrl;
  final int quantity;

  Fruit(
    this.id,
    this.title,
    this.imageUrl,
    this.quantity,
  );
  factory Fruit.fromMap(Map<String, dynamic> json) {
    return Fruit(json['id'], json['title'], json['imageUrl'],
json['quantity']);
  }
  factory Fruit.fromJson(Map<String, dynamic> json) {
    return Fruit(json['id'], json['title'], json['imageUrl'],
json['quantity']);
  }
}
```

- Creating a FruitList class

```
import 'package:flutter/material.dart';
import 'fruit.dart';
import 'fruitItem.dart';

class FruitList extends StatelessWidget {
  final List<Fruit> items;

  FruitList({Key key, this.items});

  @override
  Widget build(BuildContext context) {
    return ListView.builder(
      itemCount: items.length,
      itemBuilder: (context, index) {
        return FruitItem(item: items[index]);
      },
    );
  }
}
```

- Creating the FruitItem

```
import 'package:flutter/material.dart';

import 'fruit.dart';

class FruitItem extends StatelessWidget {
  FruitItem({this.item});

  final Fruit item;

  Widget build(BuildContext context) {
    return Container(
        padding: EdgeInsets.all(2),
        height: 140,
        child: Card(
          elevation: 5,
          child: Row(
              mainAxisAlignment: MainAxisAlignment.spaceEvenly,
              children: <Widget>[
                Image.network(
                  this.item.imageUrl,
                  width: 200,
                ),
                Expanded(
                    child: Container(
                        padding: EdgeInsets.all(5),
                        child: Column(
                         padding: EdgeInsets.all(5),
                         child: Column(
                           mainAxisAlignment:
MainAxisAlignment.spaceEvenly,
                           children: <Widget>[
                             Text(this.item.title,
                                 style: TextStyle(fontWeight:
FontWeight.bold)),
                             Text("id:${this.item.id}"),
                             Text("quantity:${this.item.quantity}"),
                           ],
                         )))
              ]),
        ));
  }
}
```

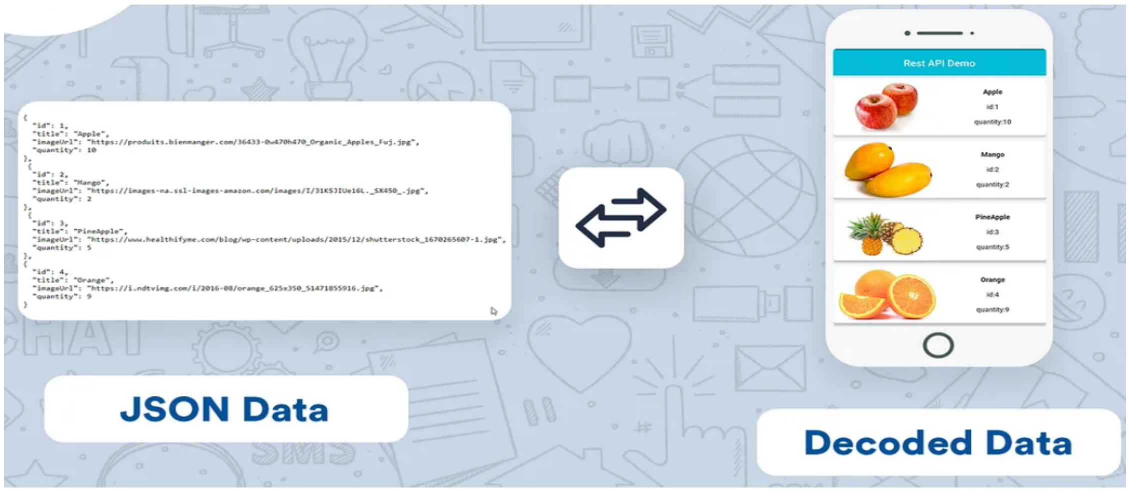
- **Displaying** the response on the screen

```
class MyHomePage extends StatelessWidget {
  final String title;
  final Future<List<Fruit>> products;

  MyHomePage({Key key, this.title, this.products}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(title: Text("Product Navigation"),
         ),
        body: Center(
          child: FutureBuilder<List<Fruit>>(
            future: products,
            builder: (context, snapshot) {
              if (snapshot.hasError) print(snapshot.error);
              return snapshot.hasData
                  ? FruitList(items: snapshot.data)
                  : Center(child: CircularProgressIndicator());
            },
          ),
        ));
  }
}
```

- Result



```
{
  "id": 1,
  "title": "Apple",
  "imageUrl": "https://produits.bienmanger.com/36433-0w470h470_Organic_Apples_Fuj.jpg",
  "quantity": 10
},
 {
  "id": 2,
  "title": "Mango",
  "imageUrl": "https://images-na.ssl-images-amazon.com/images/I/31KSJIUe16L._SX450_.jpg",
  "quantity": 2
},
 {
  "id": 3,
  "title": "PineApple",
  "imageUrl": "https://www.healthifyme.com/blog/wp-content/uploads/2015/12/shutterstock_1670265607-1.jpg",
  "quantity": 5
},
{
  "id": 4,
  "title": "Orange",
  "imageUrl": "https://i.ndtvimg.com/i/2016-08/orange_625x350_51471855916.jpg",
  "quantity": 9
}
```

Rest API Demo

Apple
id:1
quantity:10

Mango
id:2
quantity:2

PineApple
id:3
quantity:5

Orange
id:4
quantity:9