

The Computing Challenge

Web APIs
J Allen J Edwards



Problem

- A mobile app without **connectivity** would be limited to the resources it has stored within the phone
- Most of the apps on your phone probably exchange information with a server
- E.g. Weather app

Your App

- For your team project you need to access The Challenge server
- To do this, you will use an API
- You will be using an API that has been written by us

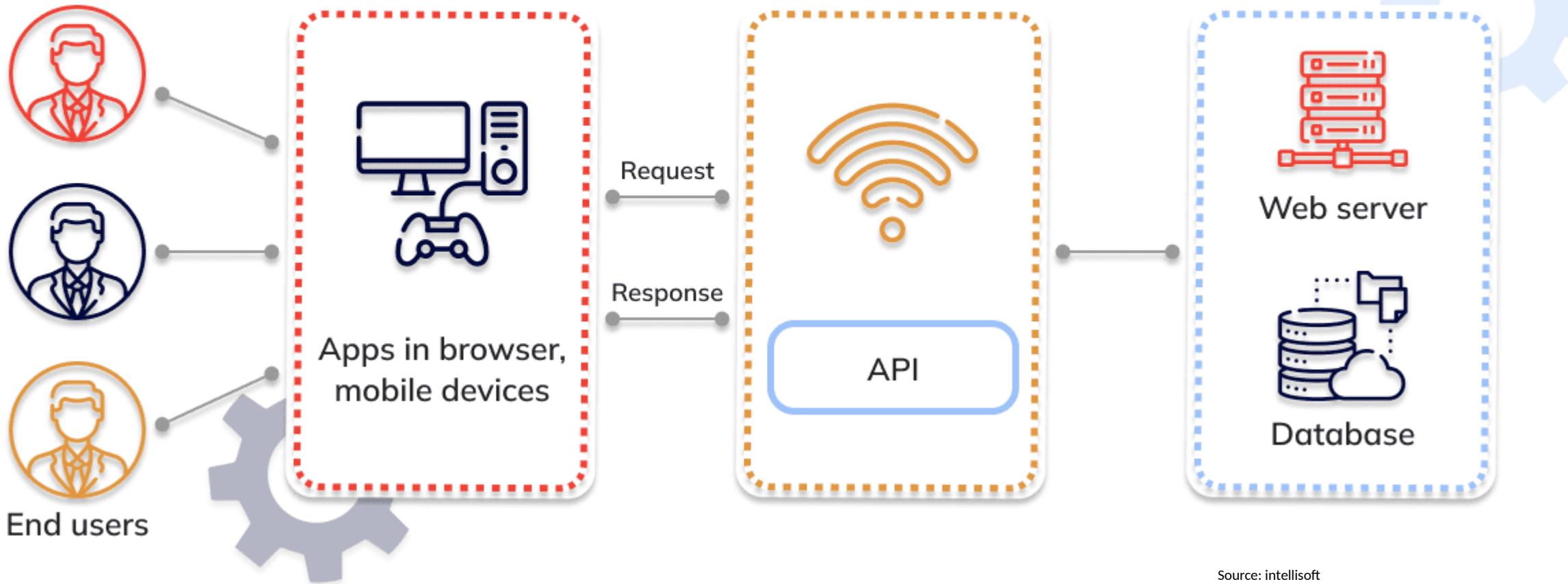




What is an API?

- API stands for Application Programming Interface
- An API is a set of functions allowing us to access the information from another service
- For example, when you use Google maps on your phone the app communicates with the Google API to get the up to date information

What is an **API**?



Source: intellisoft



REST APIs

- The most common type of API on the web is called a REST API
- A REST API uses HTTP methods like GET, PUT, POST, and DELETE to exchange data between client and server
- For the purposes of our apps we only need a simple understanding of making GET requests from Thunkable to our Challenge service.

JSON

- JSON stands for JavaScript Object Notation
- JSON is a format often used when data is sent from a server client
- JSON is an open standard format that is easily human-readable
- JSON uses plain text to store data objects and arrays



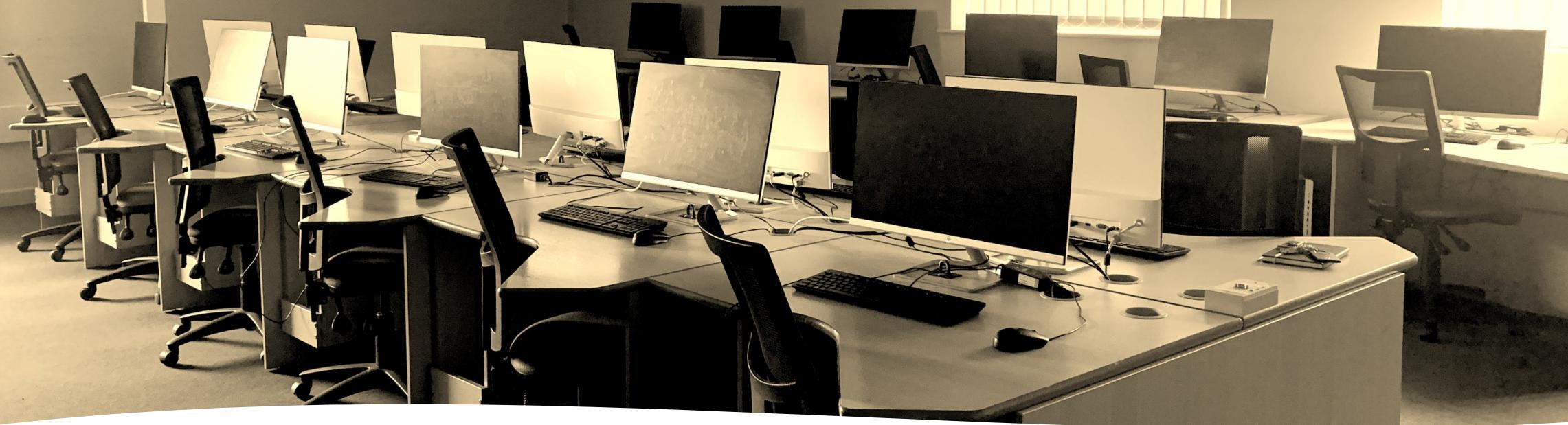
JSON Example

```
{  
  "employees": [  
    {  
      "firstName": "Julie",  
      "lastName": "Allen",  
      "department": "Computing"  
    },  
    {  
      "firstName": "Jonathan",  
      "lastName": "Edwards",  
      "department": "Computing"  
    }  
  ]  
}
```

Working with a REST API

- The web addresses that make up an API are called endpoints
- To work with an API you must know what endpoints are available?
- What does the endpoint do?
- Does the endpoint require input?





Working with a REST API

- All information sent to the API will be contained within the web address itself
- Example below sending my name:
- www.example.com/awesome/jonathan

Working with a REST API

- What is the expected output at each endpoint?
- Here the example gives us *firstname*, *lastname*, and *department* for each *employee*
- You need to know what errors may be generated if the server cannot provide the response anticipated
- Luckily The Challenge API is well documented. 🎉

```
{  
  "employees": [  
    {  
      "firstName": "Julie",  
      "lastName": "Allen",  
      "department": "Computing"  
    },  
    {  
      "firstName": "Jonathan",  
      "lastName": "Edwards",  
      "department": "Computing"  
    }  
  ]  
}
```

A Real Example



- The challenge server has various endpoints, one being:
<http://www.jedw.co.uk/challenge/getleaderboard>
- This is real you can try it on your phone right now!
- The '**getleaderboard**' endpoint requires no input
- The '**getleaderboard**' endpoint returns id, team code, team name, and score for each team in the Challenge

Leaderboard JSON Sample

```
[  
  {  
    "id": "1",  
    "teamcode": "A1",  
    "teamname": "Team Awesome",  
    "score": "10"  
  },  
  {  
    "id": "2",  
    "teamcode": "B1",  
    "teamname": "Champions",  
    "score": "4"  
  },  
  {  
    "id": "3",  
    "teamcode": "C1",  
    "teamname": "Winners",  
    "score": "6"  
  },  
]
```

*Output is abridged to fit + teams made up

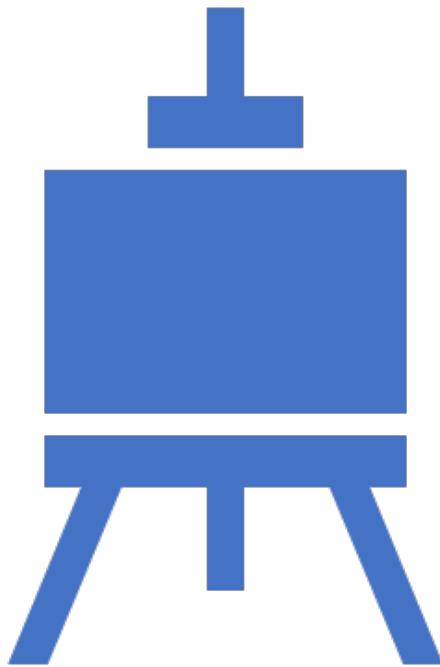


Have a **break**, have a 

{

Demo Time;

}



The Friday Showcase

- The stand-up meeting focuses on the **team**
- The Friday Showcases primary focus is your **product**
- The showcase will take place within your Friday lab
- You will have 10 minutes to present your app to the other teams
- And earn points for a place on the leaderboard



The Friday Showcase

- A good quality presentation will show off the app in it's best light
- It is not a requirement that all team members speak in the presentation but all should contribute to what is presented
- You should be prepared to answer questions from the room
- Your presentation can take any format...
 - PowerPoint
 - Video
 - Live demonstration

The Friday Showcase

Week 1 Prototype Requirements:

- The app should present the user with a series of riddles to be solved (minimum of 3 for demo purposes)
- The user should be able to enter an answer and proceed to the next riddle
- The riddles should be presented one at a time, in a fixed order
- The app should show the user their score (based on the number of correct answers) at the end of the quest

The Friday Showcase

Minimum Feature-Set

- Facility to enter a players name
- Facility to display riddles
- Facility to submit answers to riddles
- Facility to display the leaderboard (from labsheet 3)

Homework

- Make a start on your prototype app for this weeks team challenge
- Prepare for the Friday Showcase

Reminder: Your stand-up meeting will take place in Thursdays lab



Thursday's Lab Work

- In Thursday's lab you will work on an individual Thunkable task relating to todays lecture topic.
- You will also conduct a stand-up meeting.

