305CDE

Developing the Modern Web 2
April 2017 (Part-time Cohort)

Assignment Brief

You need to develop an API plus a web app that will interact with it. The exact topic you choose has to be ethics related.

Examples of ethical topics but not limited to:

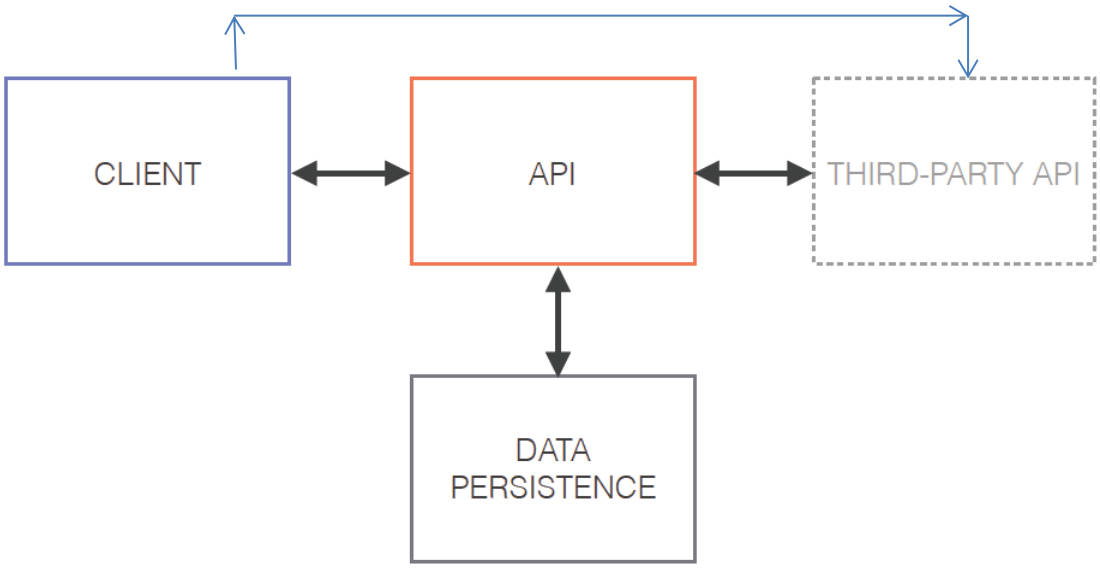
1. Drug taking
2. Personal data privacy
3. AIDS
4. Fraud
5. Sexual health
6. Teenage pregnancy
7. Risk taking behaviour
8. Unhealthy relationships

However, you should ensure that there is data available for you to use via a public API. Here are some examples to get you started. However, you are encouraged to try to find something different. For the higher grades, you will need to combine multiple APIs to derive your data (mashup).

• **Books**: https://developers.google.com/books/
• **Weather**: http://openweathermap.org/api
• **Recipes**: http://api.bigoven.com
• **Travel**: https://developers.google.com/maps/documentation/directions/
• **Tourism**: https://developers.google.com/places/webservice/search

Architecture

The diagram below may help you understand the requirements. Notice that the client interacts with your API **OR** directly interact with the third-party API(s)s and with your data persistence solution.



Features

There is a minimum feature set that you should implement in order to pass the Assignment. However, for a higher grade you would be expected to go far beyond this:

1. Users should be able to search for items without the need to register or log in to access any further features. They should be able to register for an account with appropriate validation such as email confirmation.
2. Logged in users should be able to choose from their search results to create a 'favourites' list
3. They should be able to 'organise' this list (e.g. add notes, delete items, etc.)

No matter the API calls are from your designed API(s) or third-party API(s), the relevant data, however any results, should be persisted. How you persist this data is entirely up to you. However, you will need to justify your choice(s) based on the features available. Options include persisting the data to the filesystem or using an appropriate database (SQL, document or graph). You should carefully select which data you will be saving from the third-party API(s) and only display and persist this. You will lose marks if you persist everything you get back from the third-party API(s).

Programming Language

This assignment is designed to expose you to a range of tools and frameworks used by industry. On the server, you should be using the NodeJS framework and may use any appropriate packages. However, you will need to justify your choices. In the client, you are required to use the AngularJS framework plus any supported modules, justifying your choices. Please note that you will get a zero grade if you choose other languages. You may compile everything locally (i.e. http://127.0.0.1) or lively on public hosting like Heroku. It is encouraged to submit your work with LIVE demonstration for lecturers to test (see below).

Submission

This assignment requires you to demonstrate the range of skills and knowledge required by industry and this is reflected in the coursework submission. You are required to submit two major components (2 links).

1. Source code link to your Github
2. A link to your reflective video (hosted on YouTube)

Source Code

Managing source code is a vital skill if you are to become a successful developer. For this assignment, you are required to track your API code and your client code in separate Git repositories and you will be marked on how efficiently you organise this.

You will be required to submit links to both your Git remotes hosted on Github. Both repositories should include full documentation available through the home page.

To ensure the code can be seen by your lecturers, make sure you give them **reporter** permission. Their usernames are:
`alexng88`

Video

Once you have completed the API and the client, you need to explain how your API and client work. Rather than writing a report, you are required to record a short screencast of 8 min or less. This should cover all the points in the grading criteria and demonstrate your skills and knowledge of the subject. After uploading, you must change the video permissions from private to **unlisted**. This will allow the lecturers to access the video via the link but prevent it being publicly searchable.

Make sure you separately demonstrate the API (using cURL or Postman) and the client and ensure you justify your choices of language constructs and architecture.

1. Demonstrate your API showing the requests and responses (headers/body)
2. Demonstrate the back-end persistence showing how the data is stored
3. Run the unit and acceptance tests on the API explaining code coverage
4. Demonstrate the features of the web client
5. Run the unit and acceptance tests on the web client explaining code coverage
6. Video narration must be done by **yourself.** For any suspected voice noticed, marks will be deducted.

You may deploy your latest API release on a cloud service such as **Heroku** so your lecturers can test its functionalities. It would help to understand and experience in LIVE of your works.

Grading

The marks distribution is as follows:

1. Server side coding (nodeJS + appropriate packages) (30%)
2. Client side coding (AngularJS framework plus any supported modules) (30%)
3. Mashup 3rd party APIs with clear justification of data selection (15%)
4. Video presentation (breakdown is as follows) (25%)
   a) Clear narration (5%)
   b) Content matches source code functions (10%)
   c) Clear API / back-end persistence illustration (10%)

Marks will be deducted on late submission.

1 week	Your marks x 90%
2 weeks	Your marks x 80%
More than 2 weeks	Your marks x 0%

Deadline

26 July 2017