

# The Challenge

To build a simple CRUD web application which can run on mobile or desktop browsers and a backend to accompany it.

This challenge is an opportunity to demonstrate how you would do this given a few loose constraints both around the source of data and the tooling to make this happen.

Due to the many frameworks, libraries and tooling options in the wild, there is no hard and fast rule as to what you can cobble together - the overall objective of this exercise is to establish some subject matter, a context to discuss some technical aspects around the choices and approaches taken!

## Here we go...

Ok, as mentioned above there are no hard and fast rules, however the general format of what we're looking for covers the various facets encountered when developing web apps in general. These broadly cover:

- Source control (git, svn etc)
- Build and package tooling (webpack?, package management etc)
- Framework choices (none [vanilla js]? angular? react? vue?)
- Deployment (getting the app & API deployed to hosting services, CORS etc)
- Resiliency (handling network failures / server errors, input validation)
- Backend technologies
- Knowledge of an agile/iterative process.

Modern frameworks can enable scaffolding of simple applications very quickly. Can you build on that kind of foundation to demonstrate your abilities across these facets?

# The app

It's all about cakes! create a very simple app that can allow a user to view and add to a list of favorite cakes.

If we were to put these into a simple agile format, then the stories would look like:

**S1** - As a cake lover, I can view all cakes that have been submitted so I can drool at their awe- some tastiness.

## *Acceptance Criteria*

- A simple list showing the image and name of the cake and nothing else

**S2** - As a cake lover, I can submit cakes that I like so everyone can drool at my tasty suggestions.

## *Acceptance Criteria*

- I should be able add a cake from the list of all cakes view
- I should be taken to a view where I am able to specify the name a comment and a yum factor between 1 and 5.
- I should be returned back to the list of cakes after submitting.
- I should not be able to successfully submit an invalid form. Instead a relevant error message should be displayed below all invalid form fields.

**S3** - As a cake lover I can view details about a single cake so that I can see the comment made against it.

## *Acceptance Criteria*

- I can select/click or tap any cake in the list and be taken to a view where I can see the comment/review made.
- I should be able to close or navigate back to the list of cakes once I've read the cake details.

# Technical aspects / constraints and goals

Build a simple RESTful API to provide the following operations to the app:

- GET /cakes
- GET /cakes/{id}
- POST /cakes
- PUT /cakes/{id}
- DELETE /cakes/{id}

A cake payload is in the form of:

```
{  
  id: <number>,  
  name: <string>,  
  comment: <string>,  
  imageUrl: <string>,  
  yumFactor: <number>  
}
```

## Guidance

- The app must be maintained within a source controlled repo
- Any framework can be used on front / back
- The app should be responsive across mobile/desktop browsers
- The imageUrl can just be a plain text field (no image picking is required)
- The inclusion of Progressive Web App features is encouraged
- Validation criteria:
  - Name: Required
  - Comment: Required, min length 5, max length 200
  - imageUrl: Required
  - YumFactor: Required
  - Ensure cakes are unique by name. I.E cannot add a cake with a name that matches a previously added cake