

# Evidence for Project Unit

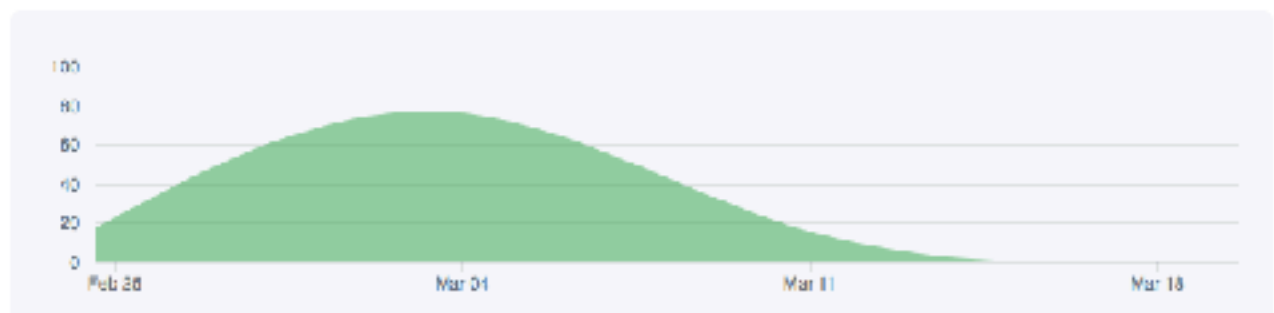
David Rawson  
Cohort E18

P.1 Github contributors page.

Feb 25, 2018 – Mar 20, 2018

Contributions: **Commits** ▾

Contributions to master, excluding merge commits



## P.2 Project brief

# Multi-Language Travel Phrase Web App - High Level Design

## MVP

### Welcome page

- on 1st visit of app allows user to enter their name and home country

### Translation View

- Provide a drop down list of destination countries.
- On country selection displays the country's default language and top 10 travel phrases.
- Provide an option to save favourite country/language pairing & phrases for later use.

## Extensions

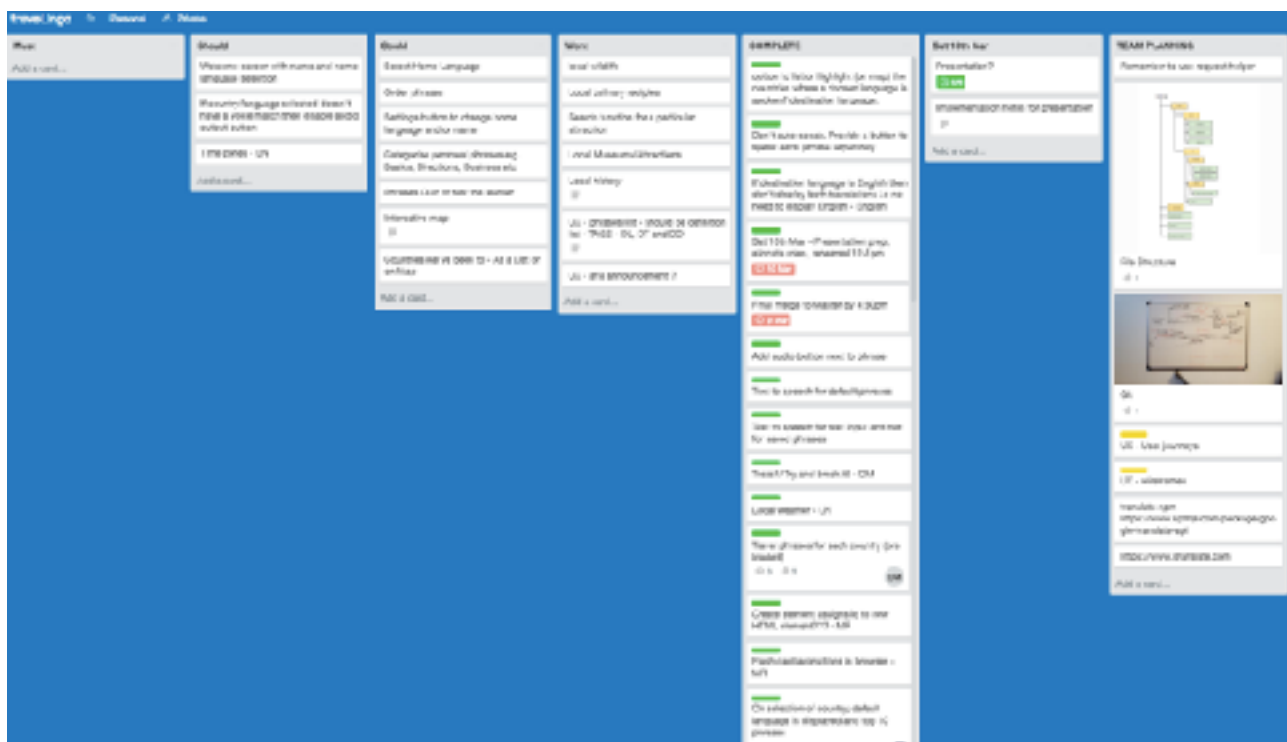
### Priority

- On subsequent visits to web app displays welcome greeting in home language e.g. "Bonjour Jean-Marie"
- Create a settings option to allow user to change name and home country
- User can enter their own phrase to translate and save if desired
- Map view showing visited countries / planned journeys
  - this could also be shown in the text version with a simple flag

### Consider

- User can set the display order of phrases
- Map interaction to show country information

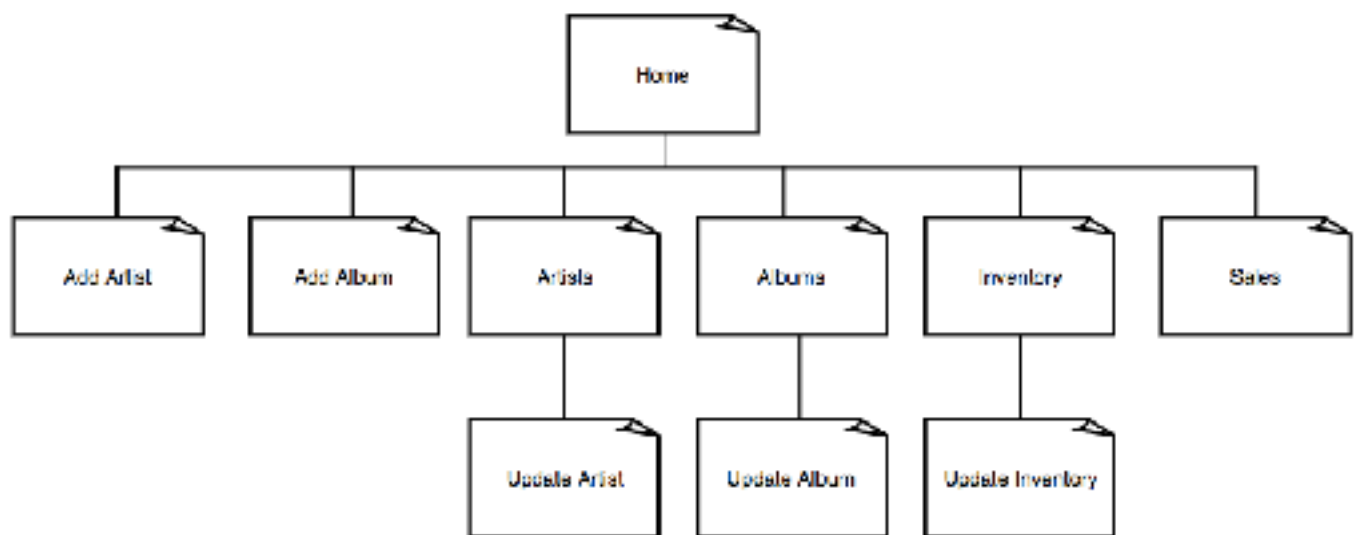
## P.3 Planning - Trello Board



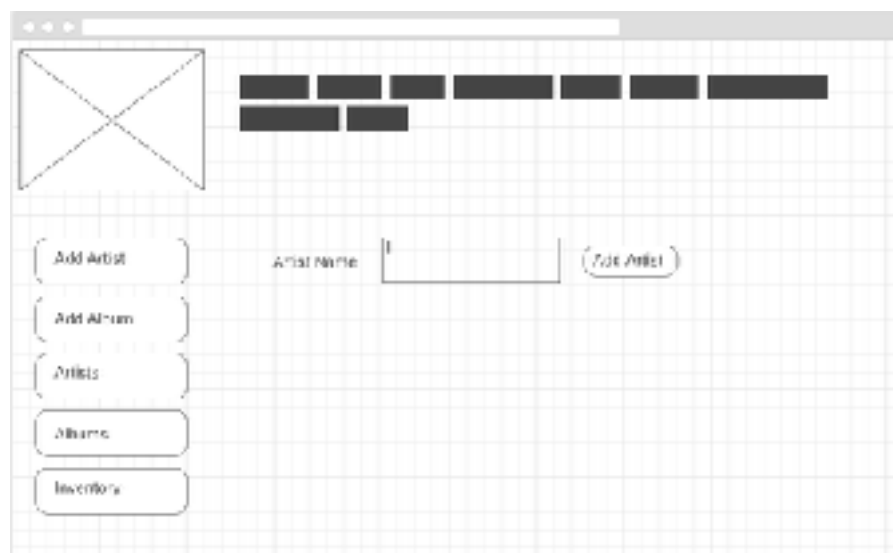
## P. 4 Acceptance Criteria and Test Plan

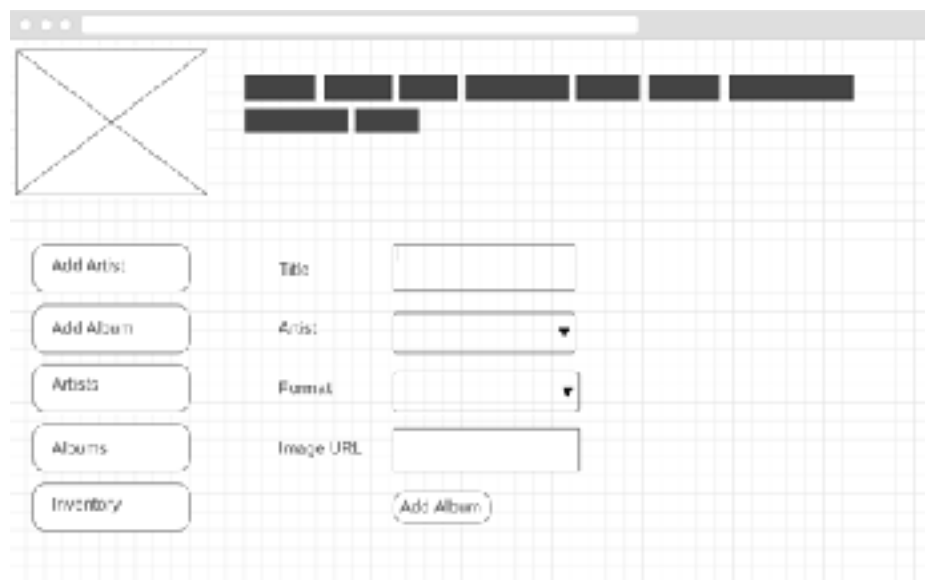
Acceptance Criteria	Expected Result / Output	Pass / Fail
A user can see all countries and their respective languages	Drop down list of countries and their languages displayed	Pass
A user can select a destination country	Common phrases translated to the language of the destination country are displayed.	Pass
A user can input their own phrase to be saved to the database	User's phrase is displayed when the language is selected	Pass

## P.5 User Sitemap

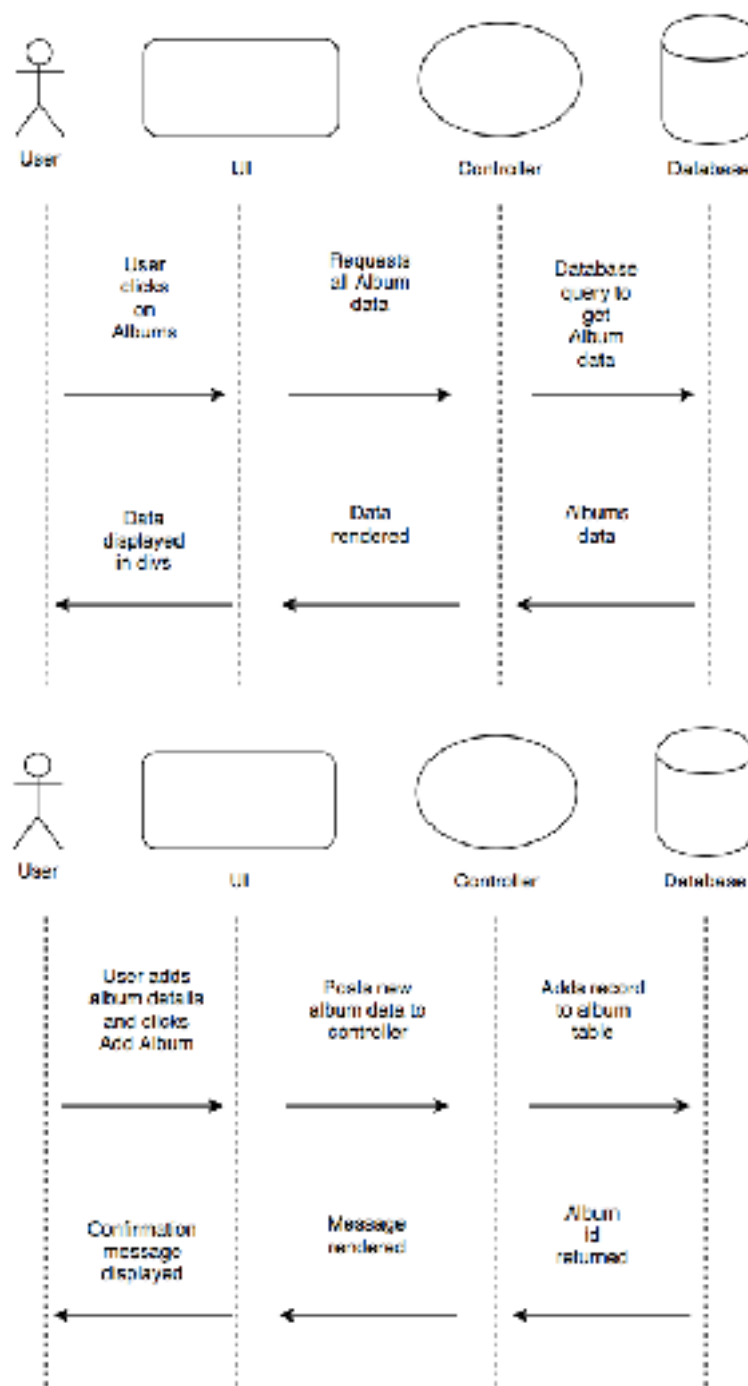


## P.6 Wireframe Diagrams

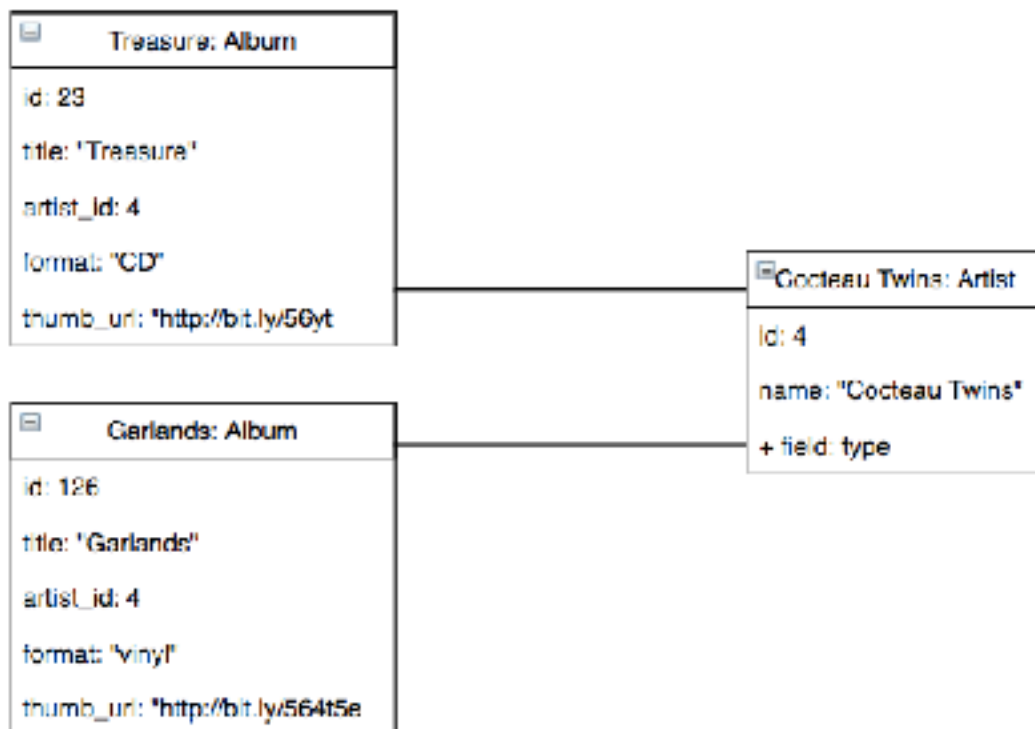
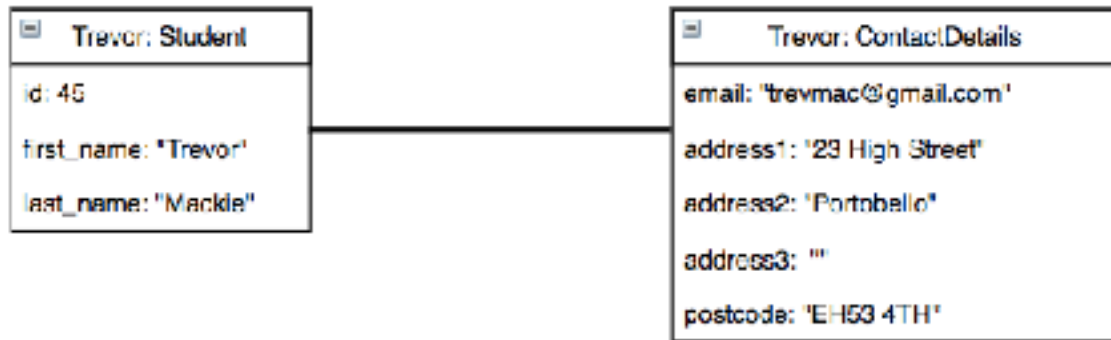




## P.7 System Interaction diagrams



## P.8 Object Diagrams



## P.9 Two Algorithms

```
findIndexesOf: function (arr, itemToFind) {  
  let newArray = [];  
  arr.forEach(function(element, index){  
    if(element === itemToFind){  
      newArray.push(index);  
    }  
  })  
  return newArray;  
},
```

This algorithm takes in an array and an item to find within the array. It will compare each array element to the item. If they match the index of the item is pushed into a new array. This new array is returned.

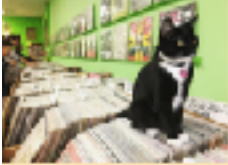
```
findDuplicates: function (arr) {  
  let newArray = [];  
  let indexes = [];  
  arr.forEach(function(element){  
    if(newArray.includes(element)===false){  
      indexes = arrayTasks.findIndexesOf(arr, element);  
      if(indexes.length > 1 ){  
        newArray.push(element);  
      }  
    }  
  })  
  return newArray  
},
```

This algorithm takes an unsorted array and iterates through the elements to see if it has already encountered one the same before. If not this function will employ the function above to get an array of indexes of all occurrences of that element. If the array of indexes has a length greater the one the element is pushed into the newArray. On the next iteration the element will already be included in the newArray so a duplicate will not be added.

## P.10 Pseudocode for a function

```
# Check if the game is over  
# If the round number is the same as the total number of rounds  
# then the game is over.  
# Else, the game is not over.
```

## P.11 Solo Project



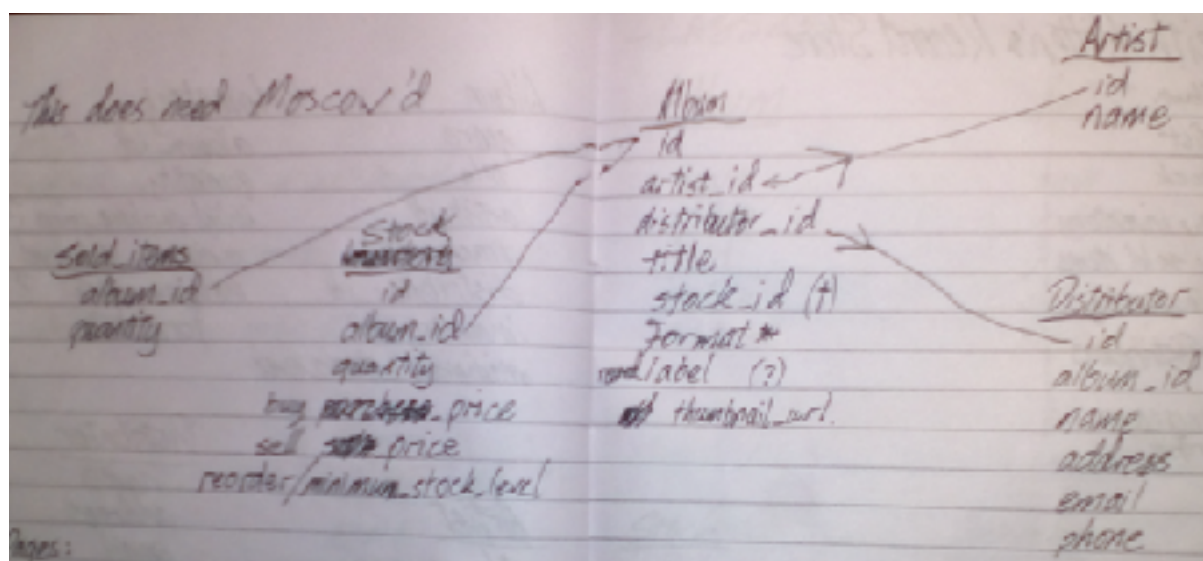
# Keith Kitten's Records

### Albums

	Album Title	Artist	Quantity	Buy Price	Sell Price	Re-order Level	Stock Check		
<a href="#">Add Album</a>	Old News White Sox	Adams and the News	8	4.50	12.00	1	Reorder Item	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
<a href="#">Add Album</a>	Rollin	Brown Ring Don Don Band	9	2.50	8.00	1	Stock OK	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
<a href="#">Albums</a>	Heaven On Las Vegas	Catroux Blues	5	0.50	12.00	4	Stock OK	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
<a href="#">Inventory</a>	4 Feet High And Rising	De La Soul	3	7.50	15.00	3	Stock OK	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
<a href="#">Albums</a>	Garlands	Catroux Blues	2	2.50	15.00	1	Stock OK	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
	Suffer Row	The Plores	4	4.50	8.00	3	Stock OK	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
	Cholitis	The Plores	1	4.50	10.00	0	Reorder Item	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
	Baroque Mende	The Plores	5	4.50	10.00	5	Reorder Item	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>
	Lemovade	Reynold	0	4.50	12.00	0	Reorder Item	<a href="#">Update Inventory</a>	<a href="#">Sell Item</a>

[https://github.com/davidrawson/week05\\_project1](https://github.com/davidrawson/week05_project1)

## P.12 Planning stages of development



Album	Artist	Stock
id	id	id
title	name	album_id
artist_id		quantity
Format		buy price
thumb-url		sell price
		re-order level

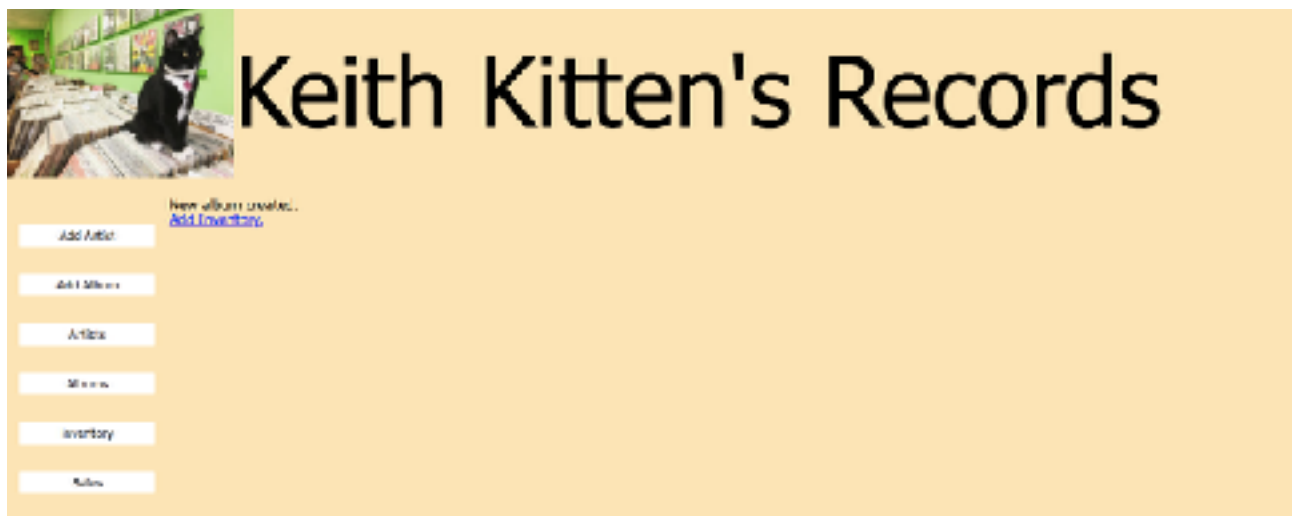
The first photo shows early ideas for the models needed. The second shows the simplified model that I proceeded with.

## P.13 & P.14 User Input and Data Persistence



The screenshot shows the 'Keith Kitten's Records' web application. On the left is a sidebar with navigation links: 'Add Artist', 'Add Album', 'Artists', 'Albums', 'Inventory', and 'Home'. The main area has a title 'Keith Kitten's Records' and a form to add a new album. The form fields are: 'Title' (text input with 'I WALK THE LINE'), 'Artist' (dropdown menu with 'Johnny Cash'), 'Format' (dropdown menu with 'CD'), and 'Image URL (optional)' (text input with 'http://img.1000gigs.com'). There is an 'Add Album' button at the bottom of the form.

Here the user is adding a new album, I Walk The Line by Johnny Cash.



The screenshot shows the 'Keith Kitten's Records' web application after the album has been added. The sidebar is the same. The main area now shows a message: 'New album created. [Add Inventory](#)'. The 'Add Album' form is still visible but the 'Add Album' button is no longer present.

It is confirmed that the new album has been created.



The screenshot shows the 'Keith Kitten's Records' web application displaying the album details in the 'Albums' search. The album is 'I Walk The Line' by 'Johnny Cash' in 'CD' format. The image URL is 'http://img.1000gigs.com'. There is an 'Add Album' button at the bottom right.

The album is being displayed in the Albums search.



Thomas Le Mende	The Poles	5	4.90	10.00	5	Re-order item.	Update Inventory	Sell Item
Lemonade	Beyoncé	6	4.90	12.00	8	Re-order item.	Update Inventory	Sell Item
rap	Wham	1	0.00	0.00	0	Stock ok.	Update Inventory	Sell Item
The Power Of Paddy	Bongwater	3	0.00	0.00	0	Stock ok.	Update Inventory	Sell Item
Treasure	Cordaeu Twins	0	0.00	0.00	0	Re-order item.	Update Inventory	Sell Item
I Walk The Line	Johnny Cash	0	0.00	0.00	0	Re-order item.	Update Inventory	Sell Item

It is also present on the inventory table with a quantity of 0.

P.15 Correct output of results, and feedback.

Updating the inventory of the Johnny Cash album.

Quantity:

Buy Price:

Sell Price:

The updated quantity, buy price and sell price are displayed on the inventory page.

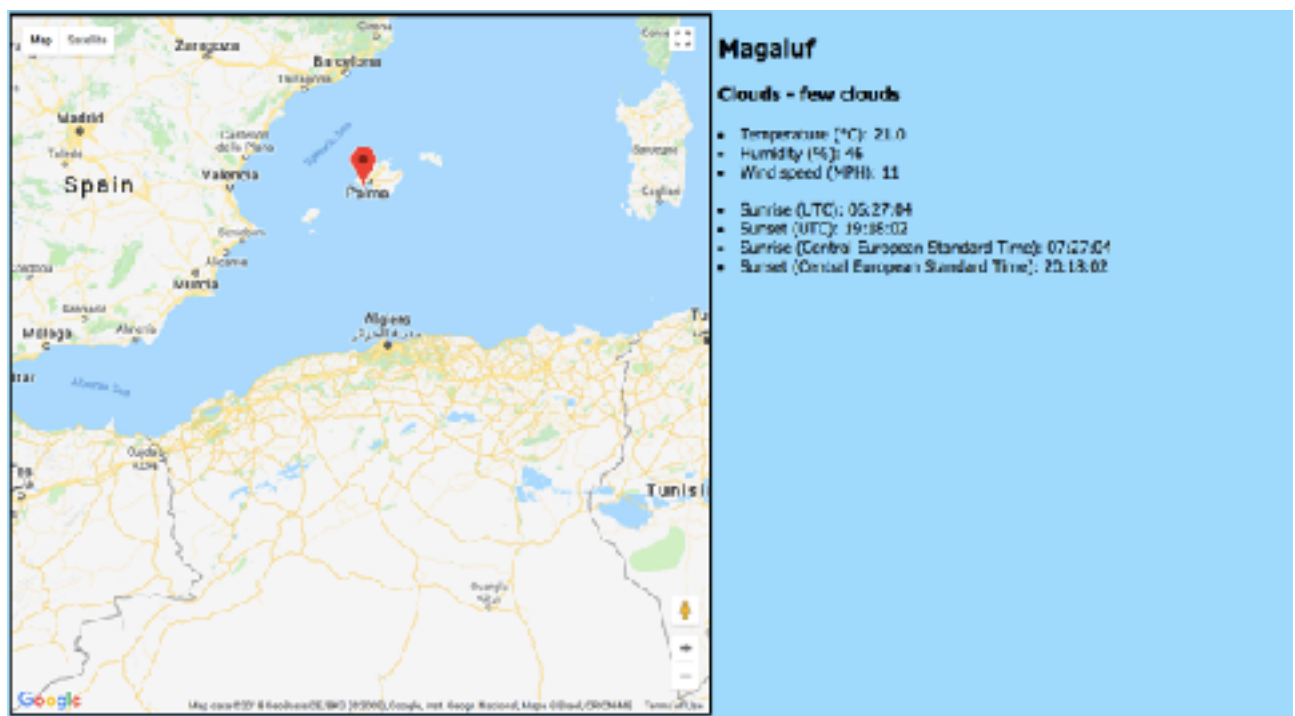
Lemonade	Beyoncé	6	4.90	12.00	8	Re-order item.	Update Inventory	Sell Item
rap	Wham	1	0.00	0.00	0	Stock ok.	Update Inventory	Sell Item
The Power Of Paddy	Bongwater	3	0.00	0.00	0	Stock ok.	Update Inventory	Sell Item
Treasure	Cordaeu Twins	0	0.00	0.00	0	Re-order item.	Update Inventory	Sell Item
I Walk The Line	Johnny Cash	10	6.40	9.90	0	Stock ok.	Update Inventory	Sell Item

P.16 An API being used.

```
const formWeatherApiUrl = function(){
  const coords = JSON.parse(localStorage.getItem('coords'));
  const url = "http://api.openweathermap.org/data/2.5/weather?lat=" + coords.lat +
    "&lon=" + coords.lng + "&APPID=" + APIkey;

  makeRequest(url, requestComplete);
}
```

The API call for retrieving weather data for a particular latitude and longitude.



The weather data for Magaluf being displayed.

## P.17 Bug Tracking report

Bug Description	Pass / Fail	Bug Fix	Pass / Fail
Can sell when the quantity in stock is 0	Fail	Insert check so method returns without sale if quantity < 1 in Stock model	Pass
Not showing most popular item first	Fail	Set order as descending in sql query in Sale model	Pass
Not limited to top 5 sales items	Fail	Set limit to 5 in sql query in Sale model	Pass

## P.18 Testing

```
it('EXTENSION - should find duplicate values in an array, returning a new array of the duplicates', function () {  
  const arr = [1, 2, 4,3, 4, 4, 5, 5, 5]  
  assert.deepStrictEqual(arrayTasks.findDuplicates(arr), [4, 5])  
})
```

Test code.

```
findDuplicates: function (arr) {  
  let newArray = [];  
  let indexes = [];  
  arr.forEach(function(element){  
    if(newArray.includes(element)===false){  
      indexes = newArray.findIndexesOf(arr, element);  
      if(indexes.length > 1 ){  
        newArray.push(element);  
      }  
    }  
  })  
  return newArray  
},
```

The function it is testing.

#### Array tasks

- ✓ should concatenate two arrays, returning a new array
  - ✓ should insert an item in an array at any index position
  - ✓ should square all values in an array, returning a new array
  - ✓ should calculate the sum of all values in an array
  - ✓ should remove all instances of a value from an array, returning a new array
  - ✓ should find all occurrences of a value, returning an array of index positions
  - ✓ should calculate the sum of all of even numbers in an array squared
- 1) EXTENSION - should find duplicate values in an array, returning a new array of the duplicates

7 passing (12ms)  
1 failing

The test failing.

```
findDuplicates: function (arr) {  
  let newArray = [];  
  let indexes = [];  
  arr.forEach(function(element){  
    if(newArray.includes(element)===false){  
      indexes = arrayTasks.findIndexesOf(arr, element);  
      if(indexes.length > 1 ){  
        newArray.push(element);  
      }  
    }  
  })  
  return newArray  
},
```

The amended function.

### Array tasks

- ✓ should concatenate two arrays, returning a new array
- ✓ should insert an item in an array at any index position
- ✓ should square all values in an array, returning a new array
- ✓ should calculate the sum of all values in an array
- ✓ should remove all instances of a value from an array, returning a new array
- ✓ should find all occurrences of a value, returning an array of index positions
- ✓ should calculate the sum of all of even numbers in an array squared
- ✓ EXTENSION – should find duplicate values in an array, returning a new array of the duplicates

8 passing (10ms)

All tests passing.