Evidence for Project Unit

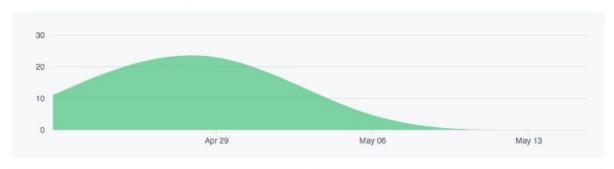
Jane Flucker E19

P. 1 Github contributors page

Apr 22, 2018 - May 16, 2018

Contributions: Commits ▼

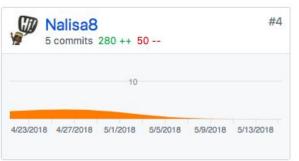
Contributions to master, excluding merge commits











P. 2 Project brief

Route Planner

Visit Scotland are look for ways to encourage people to walk and cycle. Your task is to create an app that allows users to search for cycling and hiking routes, view routes on a map, save routes to a wishlist and mark a route done.

You could use GoogleMaps Directions API:

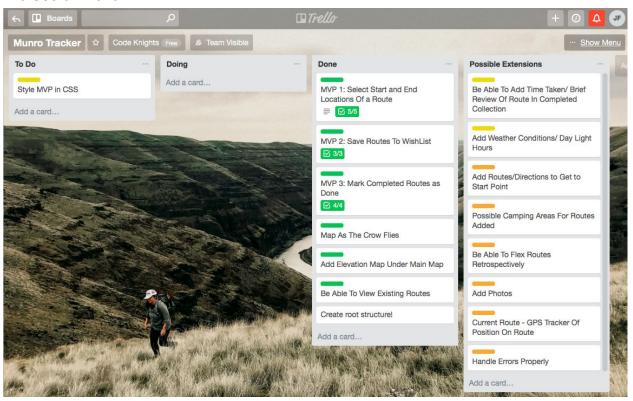
• https://developers.google.com/maps/documentation/directions/

MVP

Users should be able to:

- · Select start and finish locations for their route
- · Save routes to a wishlist
- · Mark completed routes as 'done'

P. 3 Use of Trello



P. 4 Acceptance criteria

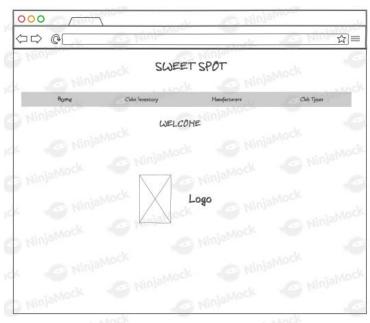
Acceptance Criteria	Expected Result/Output	Pass / Fail
User can see the map of Scotland on app.	On the app starting, a map is loaded and displayed.	PASS
User can select start and end points.	Using the mouse click to select the start and end points on the map shown by marker.	PASS
User can save the route into the wish-list	Once the two markers on map click the save button and this route details are saved to and displayed in the wish-list table on screen.	PASS
User can choose to mark a route in wish-list as complete.	Once a route is marked complete it should be displayed in completed list table on screen.	PASS

P. 5 User sitemap

Clubs Inventory Add new Add new Add new Sitemaps Club Types Add new Add new

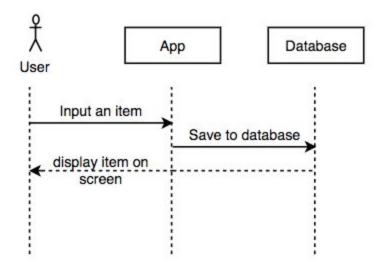
P. 6 Wireframe designs



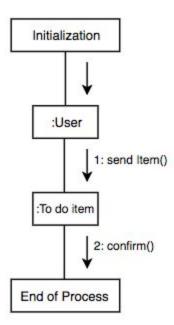


P. 7 System interactions designs

Sequence Diagrams

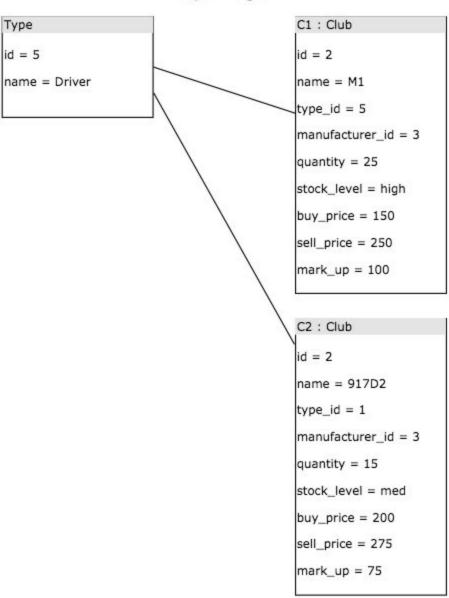


Collaboration Diagram



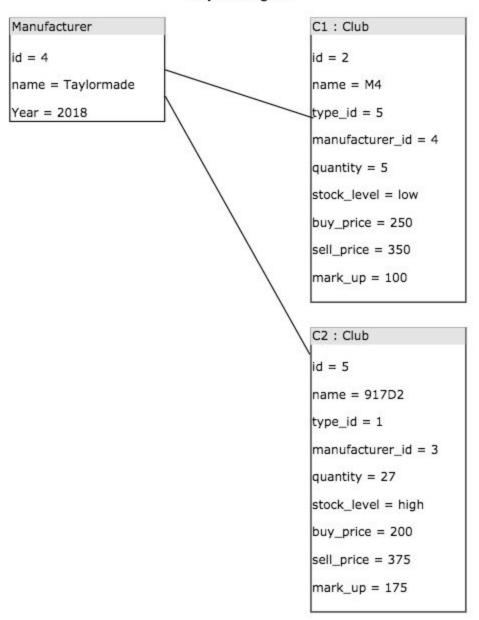
P. 8 Two object diagrams

Object Diagram



P. 8 Two object diagrams...cont

Object Diagram



P. 9 Choice of two algorithms

I have chosen this code as it brings together the three database tables for displaying all the information for the user to display on app.

```
def level_indicator()
  if @quantity > 0 && @quantity <= 10
    return "low"
  elsif @quantity > 10 && @quantity <= 20
    return "med"
  else
    return "high"
  end
end</pre>
```

I have chosen this code as it is dependant on the quantity as to the information it holds in table for a display on app that is at a glance for user.

P.10 Pseudocode

```
Pseudocode

Pseudocode

Start function level indicator

if quantity passed in is between 0 and 10

show stock level as low

if quantity passed in is between 11 and 20

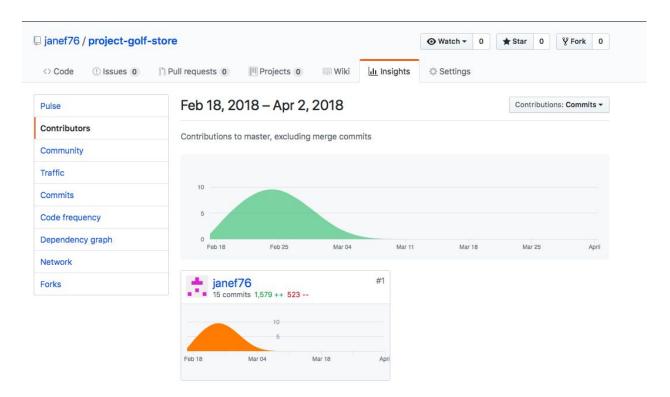
show stock level as med

otherwise

show stock level as high

End function
```

P.11 Github link to one of your projects



P.12 Screenshot of your planning and the different stages of development to show changes

Golf Store	
Must have '-	
- Show inventory page listing clubs details and manutar	ctures
- Show inventory page listing clubs details and manutary - Add in new stock with all details including quantity	
- Add in new manufactures gutomacically - use a calculated method to show stock levels eg high fi	nedflow
Nice to have: A byy and sell price for each saturally in stock	la.
- they builded nice calculate marking and allowy on "	n entry
- Add new table "Type" to add a drop down rown for quick - CRUND functionality to all tables	
	25000

	Clubs	Manufacturer
	id -int Pk	id-int Ph
	name-string	name -sking
(MVP)	tupe-string Fu	
	quantity-int	
	stock_level-string	
Type	manufacturer-INTF4	
id-int pu	tealc-stechlevel	
name-shina	[buy-price-int]	
1	sell-price-int	
	markey-int	

P.12 Screenshot of your planning and the different stages of development to show changes...cont

As a	I want to	So that
Store owner	Check stock list quickly	I can optimize sales
Store owner	Check potential profit margins	Change price it reeded
Employee	Update stock	The owner has an accu
		Stock inventory
implayee	Answer queries quickly	I'm able to do other too
•		

u	Go to home page	User Jone Navigale to Inventory page	Select an item	The date quantity of itam
9.	Display home page	Access do read and display content	Access of b road display content	Access db write updated value
	User actions Senech stock level indication	Add new stock item	Thick inventory screen for new litem	8
	System Response Access db display high, med or low	Access db write	Access of b Display, show contact	

P.13 User input

SWEET SPOT

HOME PA	AGE	CLUBS INVENT	ORY	N	IANUFACTURE	RS		CLUB TYPES
ALL CLUBS								
CLUB NAME	CLUB TYPE	MANUFACTURER NAME	QUANTITY	STOCK INDICATOR	BUY PRICE	SELL PRICE	MARK UP	UPDATE/DELETE
M1 DRIVER	DRIVER	TAYLORMADE	15	MED	300	450	150	CLICK HERE
917 3 WOOD	FAIRWAY WOODS	TITLEIST	2	LOW	200	250	50	CLICK HERE
XJ20 3-PW	IRONS	CALLAWAY	25	HIGH	400	600	200	CLICK HERE
FX 2	PUTTERS	BEN SAYERS	11	MED	20	35	15	CLICK HERE
V FORGED 60	WEDGES	NIKE	21	HIGH	40	60	20	CLICK HERE

SWEET SPOT

HOME PAGE	CLUBS INVENTORY	MANUFACTURERS	CLUB TYPES
DELETE CLUB:			
CLUB: V FORGED 60			
Delete Club			
UPDATE CLUB INFORMATION:			
NAME: V forged 60			
TYPE: wedges			
MANUFACTURER: Nike			
QUANTITY: 15			
BUY PRICE: 45			
SELL PRICE: 70			
Update			

P.13 User input...cont



P.14 Interaction with data persistence



P.14 Interaction with data persistence...cont

HOME PAGE CLUBS INVENTORY MANUFACTURERS CLUB TYPES ADD NEW CLUB: NAME: M4 TYPE: Griver MANUFACTURER: Taylormade © QUANTITY: 15 BUY PRICE: 200 SELL PRICE: 300

SWEET SPOT HOME PAGE **CLUBS INVENTORY** MANUFACTURERS **CLUB TYPES ALL CLUBS CLUB NAME CLUB TYPE** MANUFACTURER NAME QUANTITY STOCK INDICATOR BUY PRICE SELL PRICE MARK UP UPDATE/DELETE M1 DRIVER DRIVER TAYLORMADE 15 300 450 150 917 3 WOOD FAIRWAY WOODS TITLEIST 2 LOW 200 50 XJ20 3-PW IRONS CALLAWAY 25 HIGH 400 200 **PUTTERS** BEN SAYERS 11 MED 20 35 15 V FORGED 60 WEDGES 15 MED DRIVER **TAYLORMADE** 15 MED 200

ADD NEW CLUB

P.15 User output result

SWEET SPOT

			ALL	CLUBS				
CLUB NAME	CLUB TYPE	MANUFACTURER NAME	QUANTITY	STOCK INDICATOR	BUY PRICE	SELL PRICE	MARK UP	UPDATE/DELETE
M1 DRIVER	DRIVER	TAYLORMADE	15	MED	300	450	150	CLICK HERE
917 3 WOOD	FAIRWAY WOODS	TITLEIST		LOW	200	250	50	CLICK HERE
XJ20 3-PW	IRONS	CALLAWAY	25	HIGH	400	600	200	CLICK HERE
FX 2	PUTTERS	BEN SAYERS	11	MED	20	35	15	CLICK HERE
V FORGED 60	WEDGES	NIKE	15	MED	45	70	25	CLICK HERE
M4	DRIVER	TAYLORMADE	15	MED	200	300	100	CLICK HERE

SWEET SPOT

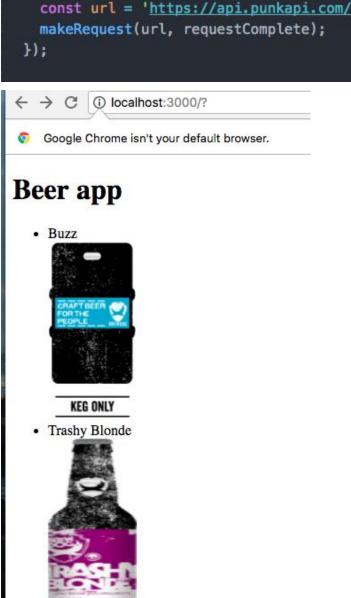
	HOME PAGE	CLUBS INVENTORY	MANUFACTURERS	CLUB TYPES
-DELETE CLUB:-				
CLUB: FX 2				
Delete Club				

SWEET SPOT

HOME PA	AGE	CLUBS INVENT	ORY	N	IANUFACTURE	RS		CLUB TYPES
	ALL CLUBS							
CLUB NAME	CLUB TYPE	MANUFACTURER NAME	QUANTITY	STOCK INDICATOR	BUY PRICE	SELL PRICE	MARK UP	UPDATE/DELETE
M1 DRIVER	DRIVER	TAYLORMADE	15	MED	300	450	150	CLICK HERE
917 3 WOOD	FAIRWAY WOODS	TITLEIST	2	LOW	200	250	50	CLICK HERE
XJ20 3-PW	IRONS	CALLAWAY	25	HIGH	400	600	200	CLICK HERE
V FORGED 60	WEDGES	NIKE	15	MED	45	70	25	CLICK HERE
M4	DRIVER	TAYLORMADE	15	MED	200	300	100	CLICK HERE

P.16 API being used within your program.

```
document.addEventListener('DOMContentLoaded', () => {
  const url = 'https://api.punkapi.com/v2/beers';
  makeRequest(url, requestComplete);
});
```



· Berliner Weisse With Yuzu - B-Sides

P.17 Bug tracking report

Bug Tracking Report

Stock level to display low if quantity 10 or below	Failed	Added in an equals-sign to the less than- sign.	Passed
Name type to be displayed on club inventory	Failed	ID showing first, fixed code to display name	Passed
Mark-up to be displayed on screen and update if price changed	Failed	Stored the value in database after performing calculation	Passed
Be able to delete an item in stock	Failed	Changed to allow deleting by ID so as only 1 item is deleted at a time	Passed
Show stock item in newest first	Failed	Changed SQL statement to order the items newest first	Passed

P.18 Testing in a program

```
card_game_spec.rb
   require("minitest/autorun")
  require_relative("../card_game")
   require_relative("../card")
    class TestCardGame < MiniTest::Test</pre>
      def setup
        @card1 = Card.new("club", 1)
        @card2 = Card.new("heart", 3)
        @card3 = Card.new("spade", 6)
11
        @cardgame = CardGame.new()
      end
      def test_card_for_ace()
        card = @cardgame.checkforAce(@card1)
        assert equal(true, card)
      end
      def test_card_for_not_ace()
        card = @cardgame.checkforAce(@card2)
        assert equal(false, card)
      end
      def test highest card()
        value = @cardgame.highest_card(@card1, @card2)
        assert_equal(@card2, value)
      end
      def test_total_value_of_cards()
        total = CardGame.cards_total([@card1, @card2, @card3])
        assert_equal("You have a total of 10", total)
      end
    end
```

P.18 Testing in a program...cont

```
implementation_and_testing git:(master) × ruby specs/card_game_spec.rb
specs/card_game_spec.rb:2:in `require_relative': /Users/janeflucker/codeclan_work/pda/implementation_and_testing/card_game.rb:20: syntax error, unexpected keyword_end, expecting end-of-input (SyntaxError) end # extra end statement, not ne

from specs/card_game_spec.rb:2:in `<main>'
implementation_and_testing git:(master) × ||

implementation_and_testing git:(master) × ruby specs/card_game_spec.rb
Run options: --seed 34395

# Running:
....
Finished in 0.001254s, 3189.7927 runs/s, 3189.7927 assertions/s.

4 runs, 4 assertions, 0 failures, 0 errors, 0 skips
implementation_and_testing git:(master) × ||
implementatio
```