

Evidence for Project Unit

Jo Malo, E21

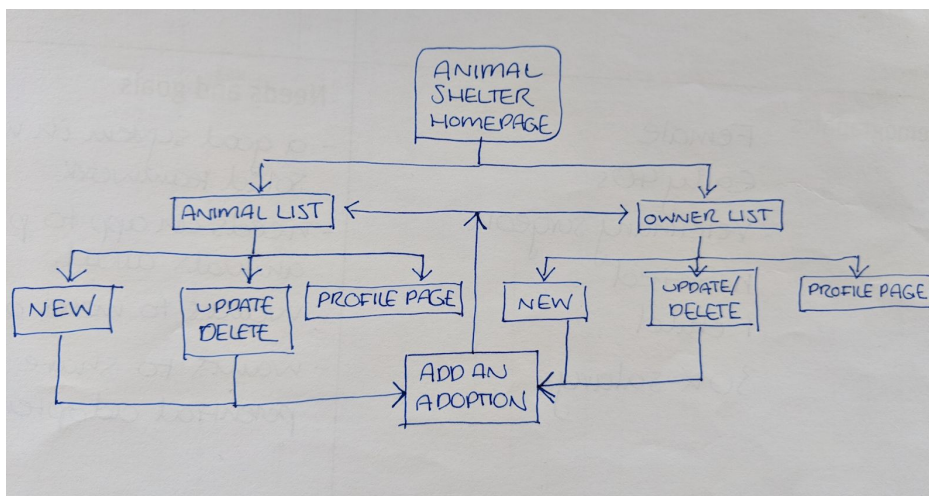
P. 1 Github Contributors page

P. 2 Project Brief

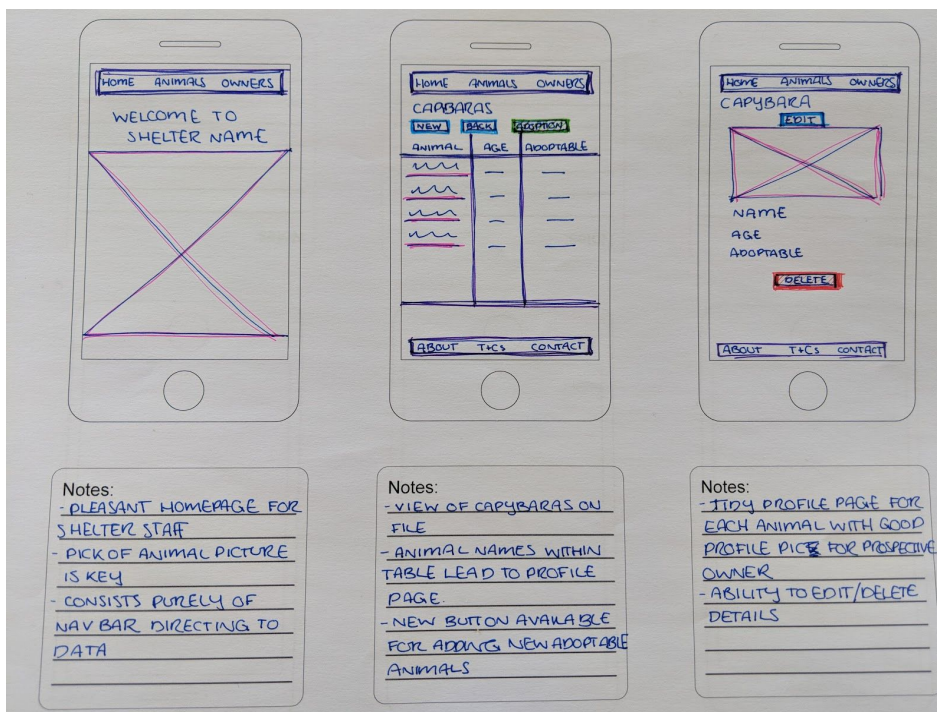
P. 3 Use of Trello

P. 4 Acceptance Criteria

P. 5 User sitemap



P. 6 Wireframes designs



P. 7 System interactions diagrams

P. 8 Two Object Diagrams

P. 9 Choice of two algorithms (find the algorithms on a program you might have written, show the code you have used.) On this example please take a screenshot and write what it is doing and why you decided to use it.

P. 10 Example of Pseudocode

```
def self.all()
  sql = "SELECT * FROM owners"
  results = SqlRunner.run( sql )
  return results.map { |hash| Owner.new( hash ) }
end

def self.find(id)
  sql = "SELECT * FROM owners
  WHERE id = $1"
  values = [id]
  results = SqlRunner.run( sql, values )
  return Owner.new( results.first )
end

def self.delete_all
  sql = "DELETE FROM owners"
  SqlRunner.run( sql )
end

def to map all items (hash of owners)
  get a result = map the hash of owners {hash| new owner (hash)}
  get that result
end

end
```

P. 11 Github link to one of your projects

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

P. 13 User input Make sure you show the input being added.

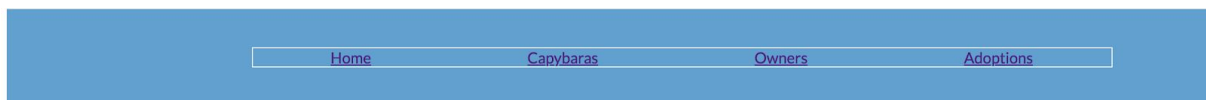
The screenshot shows a code editor with a file explorer on the left and a terminal window on the right. The file explorer shows a project structure for 'ruby-codeclan-project1' with files like 'owners_controller.rb', 'capybaras.sql', 'seeds.rb', 'sql_runner.rb', 'models', 'adoption.rb', 'capybara.rb', 'owner.rb', 'public', 'images', 'capy_head_logo.png', 'capybara_bag.jpg', 'capybara_sleep.jpg', 'styles', 'forms.css', 'structure.css', 'tables.css', 'views', 'adoptions', 'index.erb', 'new.erb', 'capybaras', 'edit.erb', 'homeless_show.erb', 'index.erb', 'new.erb', 'rehomed_show.erb', and 'show.erb'. The main editor shows the 'capybara.rb' file with the following code:

```
3 class Capybara
4
5   attr_reader( :admission, :age, :id )
6   attr_accessor( :name, :available )
7
8   def initialize( options )
9     @id = options['id'].to_i if options['id']
10    @name = options['name']
11    @admission = options['admission']
12    @age = options['age']
13    @available = options['available'].to_i
14  end
15
16  def save()
17    sql = "INSERT INTO capybaras
18    (
19      name,
20      admission,
21      age,
22      available
23    )
24    VALUES
25    (
26      $1, $2, $3, $4
27    )
28    RETURNING id"
29    values = [@name, @admission, @age, @available]
30    results = SqlRunner.run(sql, values)
31    @id = results.first()['id'].to_i
32  end
33
34  def update()
35    sql = "UPDATE capybaras SET (
36    name,
```

The terminal window shows the following output:

```
psql:db/capybaras.sql:1: ERROR: table "adoptions" does not exist
psql:db/capybaras.sql:2: ERROR: table "capybaras" does not exist
psql:db/capybaras.sql:3: ERROR: table "owners" does not exist
CREATE TABLE
CREATE TABLE
CREATE TABLE
-> ruby-codeclan-project1 git:(master) psql -d capybaras -f db/capybaras.sql
DROP TABLE
DROP TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
-> ruby-codeclan-project1 git:(master) ruby db/seeds.rb
From: /Users/jomalo/codeclan_work/week_05/ruby-codeclan-project1/db/seeds.rb @ line 83 :
78:   "owner_id" => owner1.id
79:   })
80:   adoption1.save()
81:
82:   binding.pry
=> 83: nil
```

- User inputting data in app:



Add a new capybara!

Name: Admission: Age:

- User data being saved:

[Show homeless capybaras](#)

[Show rehomed capybaras](#)

[Add a capybara](#)

Capybaras

Name	Admission Date	Age	Available?
Julian	2018-05-03	5	No
Xavier	2018-02-03	8	Yes
Nacho	2015-05-20	10	Yes
Maria	2017-12-07	2	Yes
Antonio	2018-05-02	4	No
Julio	2016-01-05	9	Yes
Suca	2016-09-27	2	Yes
Antonio	2018-04-14	5	No

P. 14 Interaction with data persistence

- User inputting data:

Add a prospective owner

Full Name:

[Add an owner](#)

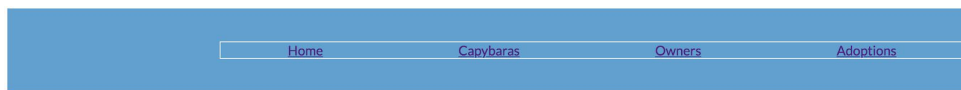
Owners

Owner ID	Name
1	Harry Marshall
2	Jack Pot
3	Mike Hall

```
ruby-codeclan-project1 — ruby db/seeds.rb — ruby db/seeds.rb — 112x24
ruby
[2] pry(main)> Capybara.all
=> [#<Capybara:0x007f9469025800 @admission="2018-05-03", @age="5", @available=0, @id=1, @name="Julian">,
#<Capybara:0x007f9469025800 @admission="2018-02-03", @age="8", @available=1, @id=2, @name="Xavier">,
#<Capybara:0x007f9469024600 @admission="2015-05-20", @age="10", @available=1, @id=3, @name="Nacho">,
#<Capybara:0x007f9469024600 @admission="2017-12-07", @age="2", @available=1, @id=4, @name="Maria">,
#<Capybara:0x007f94694936c8 @admission="2018-05-02", @age="4", @available=0, @id=5, @name="Antonio">,
#<Capybara:0x007f94694936d0 @admission="2016-01-05", @age="9", @available=1, @id=6, @name="Julio">,
#<Capybara:0x007f9469493c20 @admission="2016-09-27", @age="2", @available=1, @id=7, @name="Suca">,
#<Capybara:0x007f9469493ab8 @admission="2018-04-14", @age="5", @available=0, @id=8, @name="Antonio">]
[3] pry(main)> Capybara.Antonio
NoMethodError: undefined method 'Antonio' for Capybara:Class
from (pry):3:in '<main>'
[4] pry(main)> Capybara.find(8)
=> #<Capybara:0x007f9469108510 @admission="2018-04-14", @age="5", @available=0, @id=8, @name="Antonio">
[5] pry(main)> Owner.all
=> [#<Owner:0x007f946945a6c8 @id=1, @name="Harry Marshall">, #<Owner:0x007f946945a628 @id=2, @name="Jack Pot">]
[6] pry(main)> Owner.all
=> [#<Owner:0x007f94688a8820 @id=1, @name="Harry Marshall">,
#<Owner:0x007f94688a86b8 @id=2, @name="Jack Pot">,
#<Owner:0x007f94688a85c8 @id=3, @name="Mike Hall">]
[7] pry(main)> Owner.find(3)
=> #<Owner:0x007f94693b1f50 @id=3, @name="Mike Hall">
[8] pry(main)> ]
```

P. 15 User output result

- User input of new adoption on app:



Match an owner with a capybara!

Select a capybara:

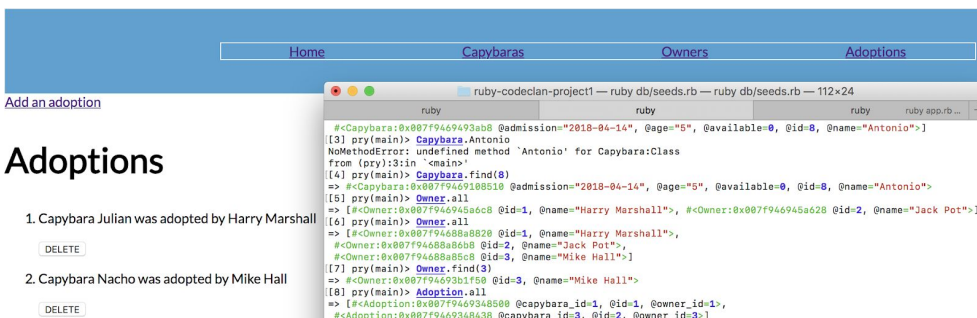
Nacho

Select an owner:

Mike Hall

CLICK TO ADOPT

- User clicks on 'click to adopt' and is directed to the adoptions table view:



P. 16 Bug tracking report showing the errors diagnosed and corrected.

P. 17 Testing your program Show the test code, the test not passing.....and then the test fixed.