Evidence for Project Unit

Hamish Hoad
Cohort E-21

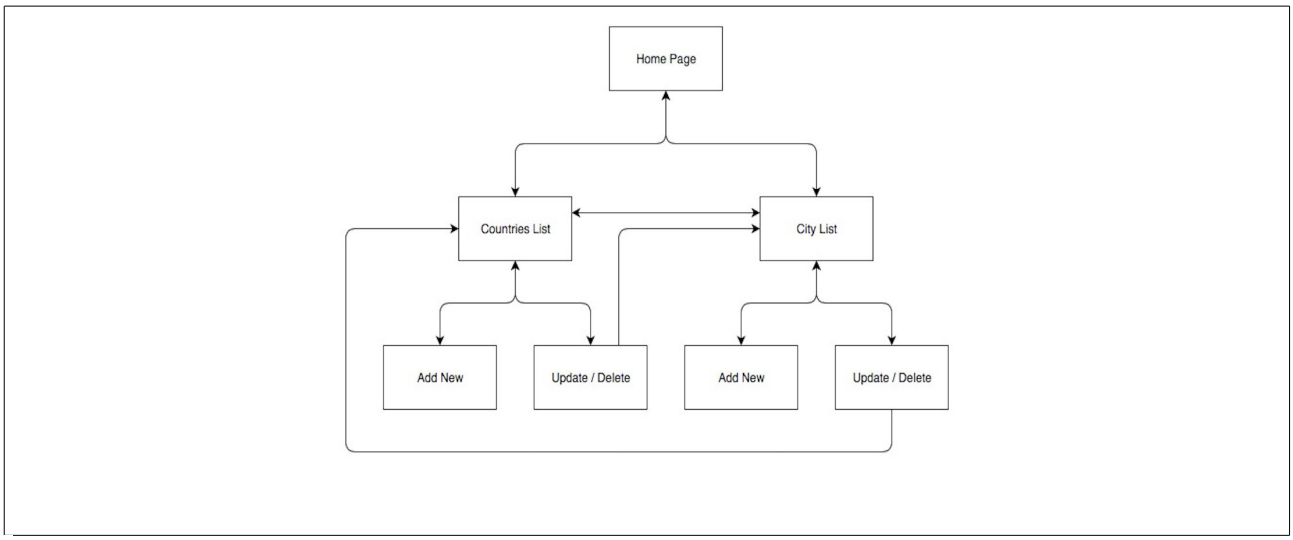
P. 1 Github Contributors page
Evidence for unit

P. 2 Project Brief
Evidence for unit

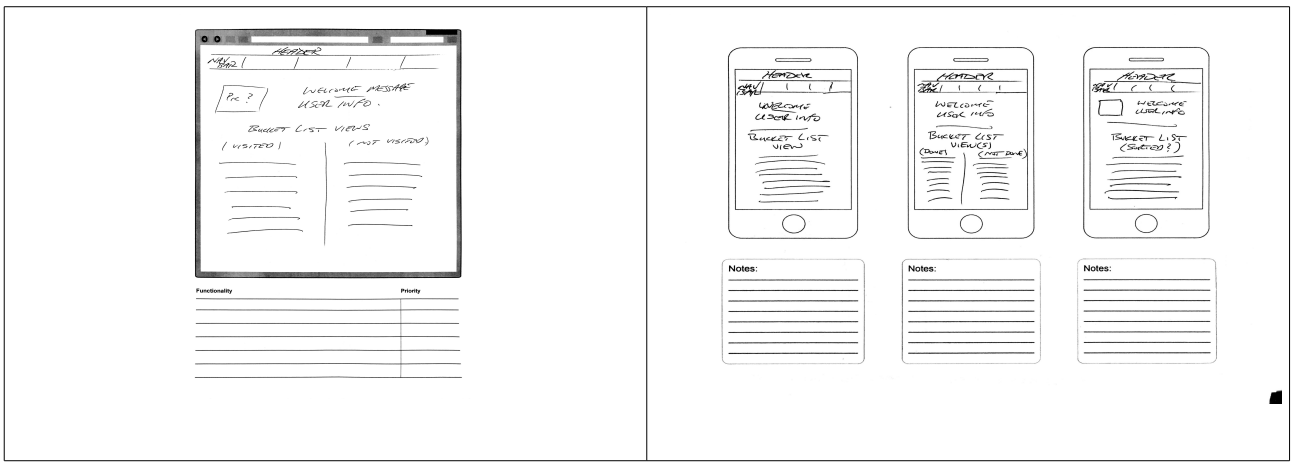
P. 3 Use of Trello
Evidence for unit

P. 4 Acceptance Criteria
Evidence for unit

P. 5 User sitemap



P. 6 Wireframes designs



P. 7 System interactions diagrams

Evidence for unit	Evidence for unit

P. 8 Two Object Diagrams

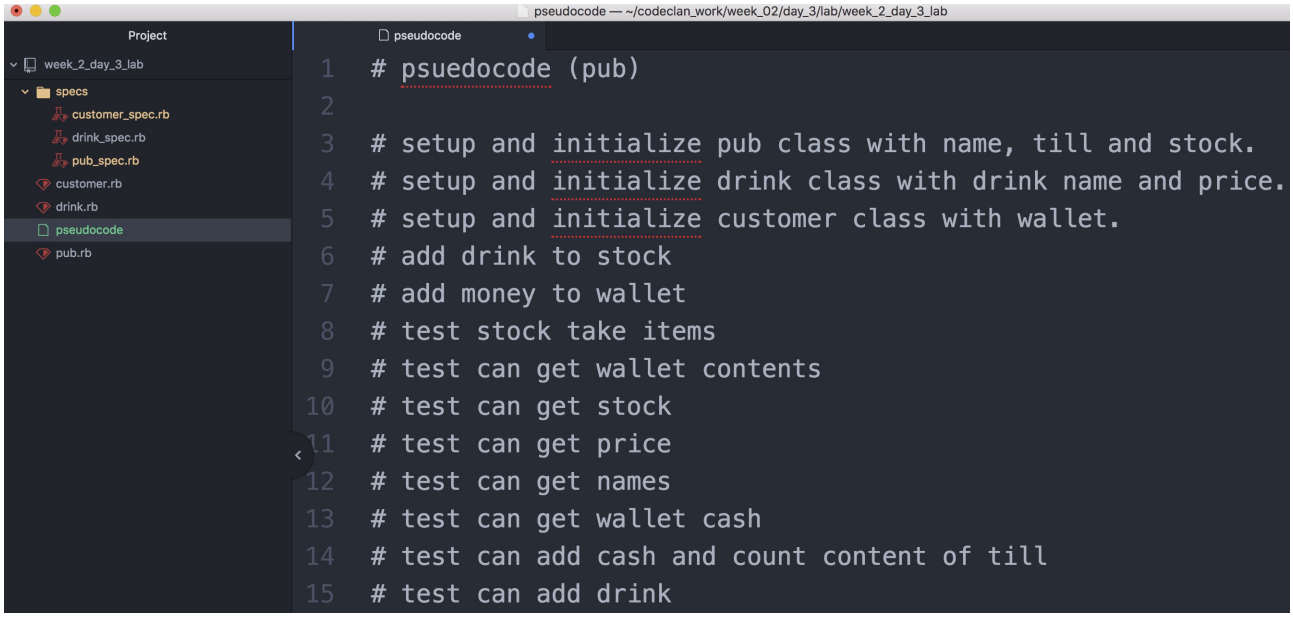
Evidence for unit Evidence for unit

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.

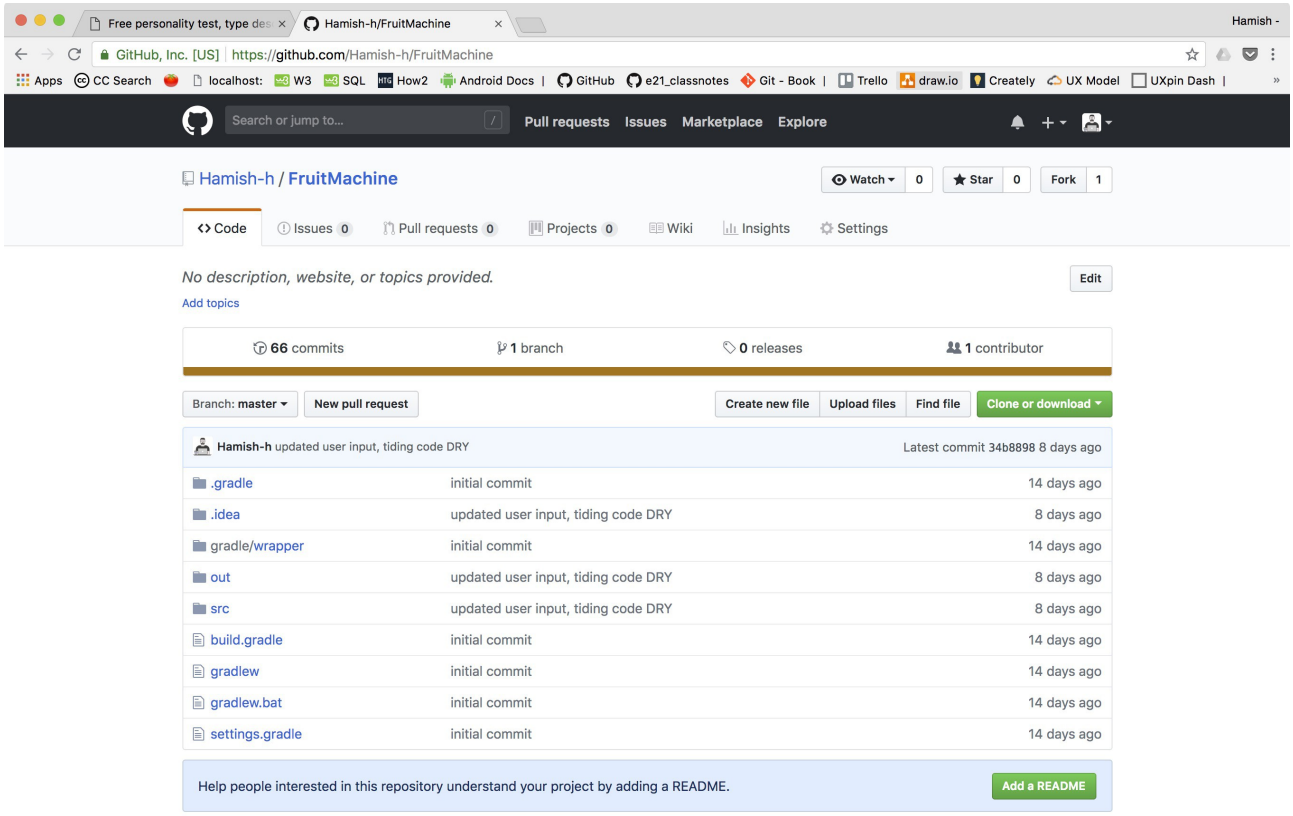
P. 10 Example of Pseudocode

Evidence for unit



P. 11 Github link to one of your projects

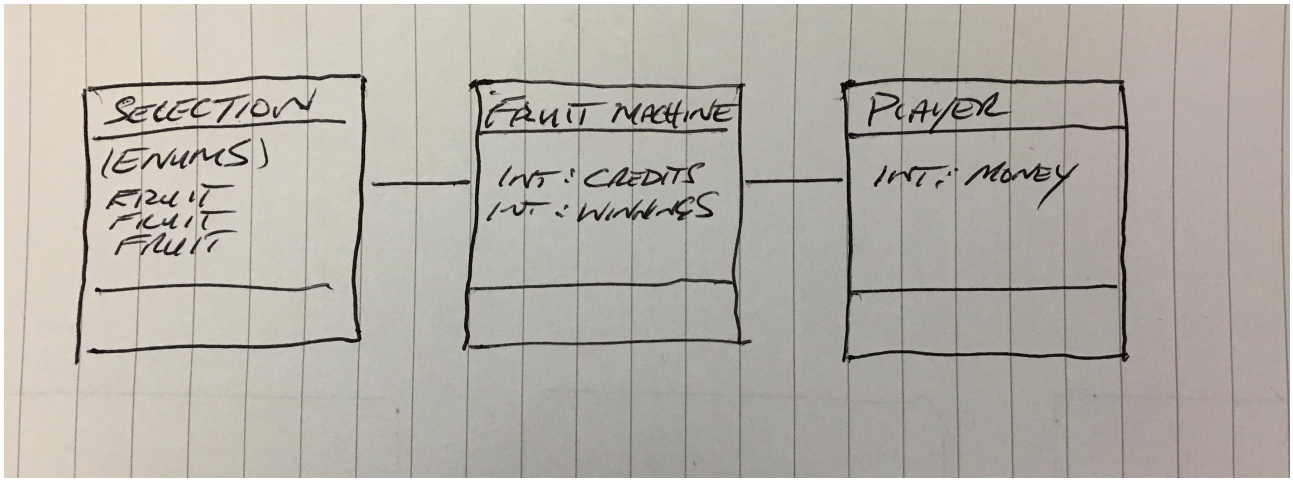
https://github.com/Hamish-h/FruitMachine



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

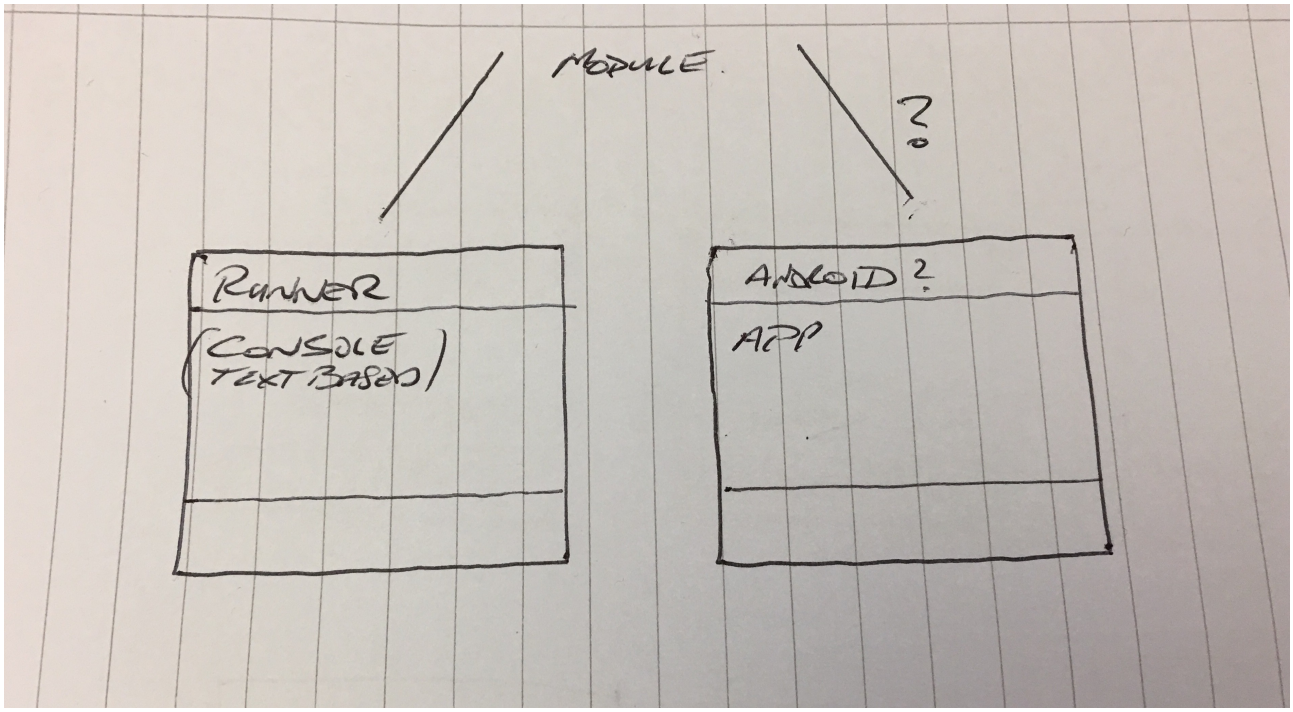
Planning and development – fruit machine

Rough plan

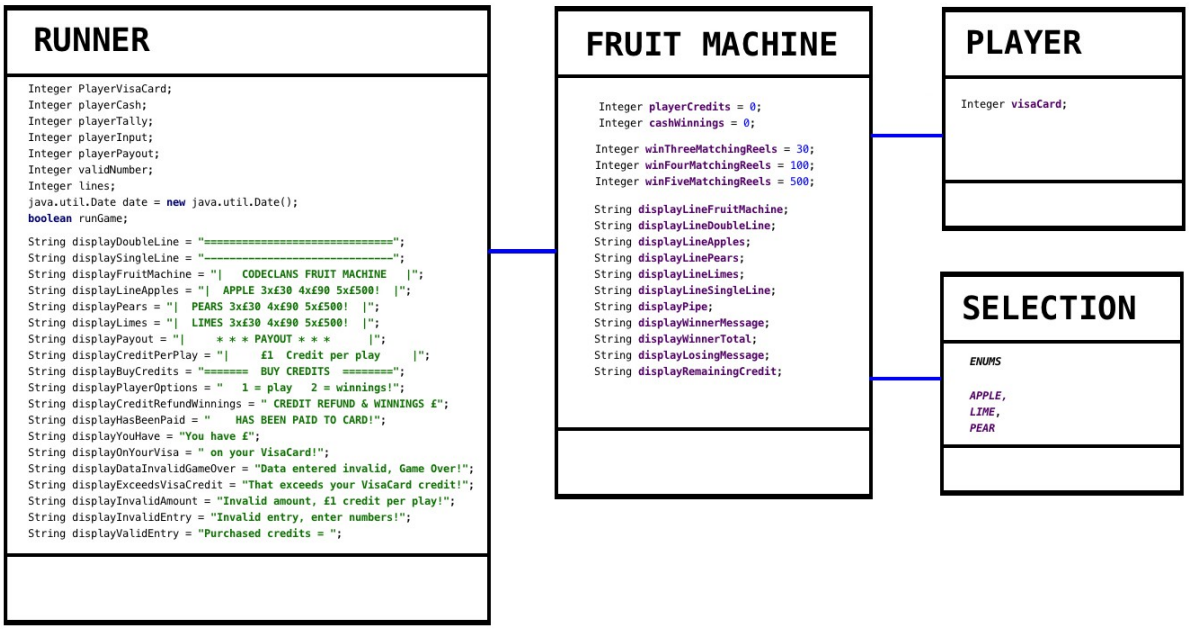


Rough plan for the control function

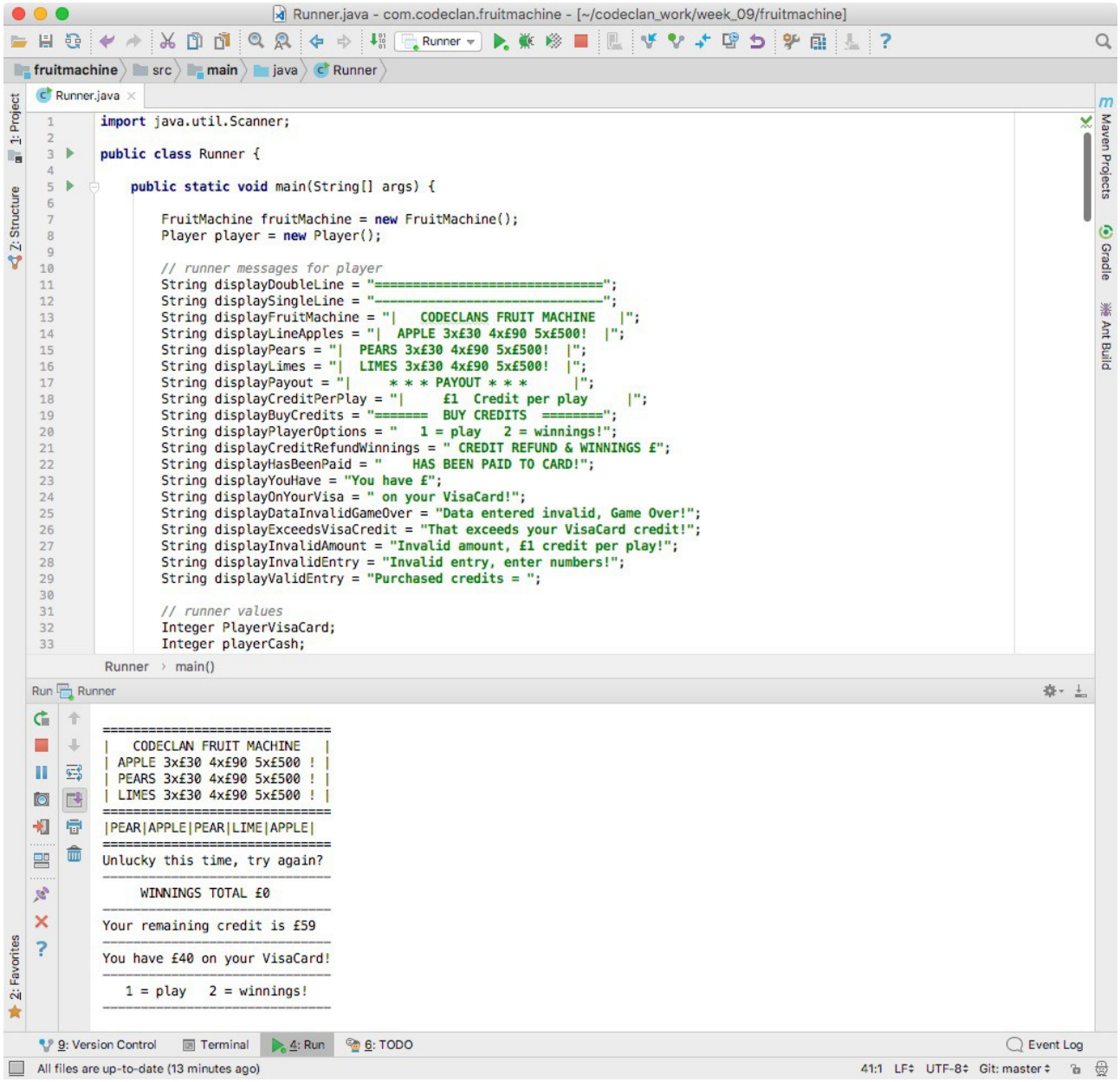
Java runner and Andriod app.



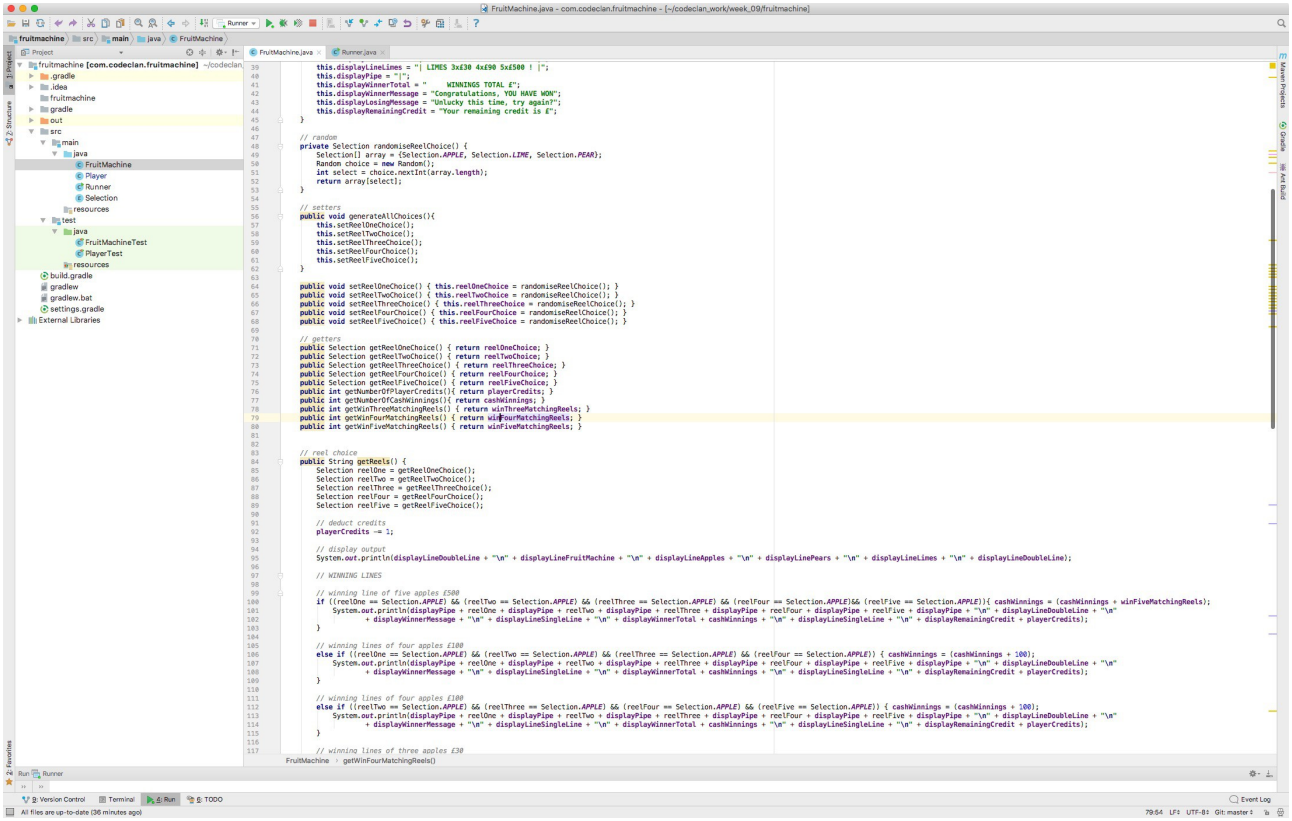
Detailed plan



Coding steps



Coding Steps



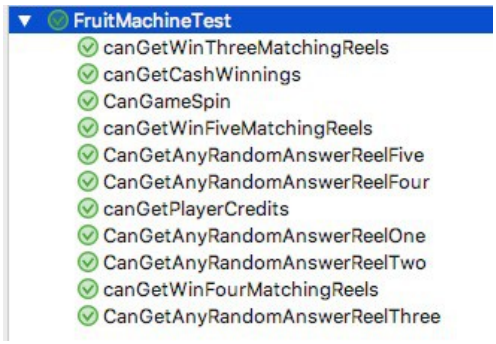
Coding Steps

```
// WINNING LINES

// winning line of five apples £500
if ((reelOne == Selection.APPLE) && (reelTwo == Selection.APPLE) && (reelThree == Selection.APPLE) && (reelFour == Selection.APPLE)&& (reelFive == Selection.APPLE)){ cashWinnings = (cashWinnings + winFiveMatchingReels);
    System.out.println(displayPipe + reelOne + displayPipe + reelTwo + displayPipe + reelThree + displayPipe + reelFour + displayPipe + reelFive + displayPipe + "\n" + displayLineDoubleLine + "\n"
            + displayWinnerMessage + "\n" + displayLineSingleLine + "\n" + displayWinnerTotal + cashWinnings + "\n" + displayLineSingleLine + "\n" + displayRemainingCredit + playerCredits);
}

// winning lines of four apples £100
else if ((reelOne == Selection.APPLE) && (reelTwo == Selection.APPLE) && (reelThree == Selection.APPLE) && (reelFour == Selection.APPLE)) { cashWinnings = (cashWinnings + 100);
    System.out.println(displayPipe + reelOne + displayPipe + reelTwo + displayPipe + reelThree + displayPipe + reelFour + displayPipe + reelFive + displayPipe + "\n" + displayLineDoubleLine + "\n"
            + displayWinnerMessage + "\n" + displayLineSingleLine + "\n" + displayWinnerTotal + cashWinnings + "\n" + displayLineSingleLine + "\n" + displayRemainingCredit + playerCredits);
}

// winning lines of four apples £100
else if ((reelTwo == Selection.APPLE) && (reelThree == Selection.APPLE) && (reelFour == Selection.APPLE) && (reelFive == Selection.APPLE)) { cashWinnings = (cashWinnings + 100);
    System.out.println(displayPipe + reelOne + displayPipe + reelTwo + displayPipe + reelThree + displayPipe + reelFour + displayPipe + reelFive + displayPipe + "\n" + displayLineDoubleLine + "\n"
            + displayWinnerMessage + "\n" + displayLineSingleLine + "\n" + displayWinnerTotal + cashWinnings + "\n" + displayLineSingleLine + "\n" + displayRemainingCredit + playerCredits);
}

// winning lines of three apples £30
else if ((reelOne == Selection.APPLE) && (reelTwo == Selection.APPLE) && (reelThree == Selection.APPLE)) { cashWinnings = (cashWinnings + winThreeMatchingReels);
    System.out.println(displayPipe + reelOne + displayPipe + reelTwo + displayPipe + reelThree + displayPipe + reelFour + displayPipe + reelFive + displayPipe + "\n" + displayLineDoubleLine + "\n"
            + displayWinnerMessage + "\n" + displayLineSingleLine + "\n" + displayWinnerTotal + cashWinnings + "\n" + displayLineSingleLine + "\n" + displayRemainingCredit + playerCredits);
}

// winning lines of three apples £30
else if ((reelTwo == Selection.APPLE) && (reelThree == Selection.APPLE) && (reelFour == Selection.APPLE)){ cashWinnings = (cashWinnings + winThreeMatchingReels);
    System.out.println(displayPipe + reelOne + displayPipe + reelTwo + displayPipe + reelThree + displayPipe + reelFour + displayPipe + reelFive + displayPipe + "\n" + displayLineDoubleLine + "\n"
            + displayWinnerMessage + "\n" + displayLineSingleLine + "\n" + displayWinnerTotal + cashWinnings + "\n" + displayLineSingleLine + "\n" + displayRemainingCredit + playerCredits);
}

// winning lines of three apples £30
else if ((reelThree == Selection.APPLE) && (reelFour == Selection.APPLE) && (reelFive == Selection.APPLE)) { cashWinnings = (cashWinnings + winThreeMatchingReels);
    System.out.println(displayPipe + reelOne + displayPipe + reelTwo + displayPipe + reelThree + displayPipe + reelFour + displayPipe + reelFive + displayPipe + "\n" + displayLineDoubleLine + "\n"
            + displayWinnerMessage + "\n" + displayLineSingleLine + "\n" + displayWinnerTotal + cashWinnings + "\n" + displayLineSingleLine + "\n" + displayRemainingCredit + playerCredits);
}
```

Output

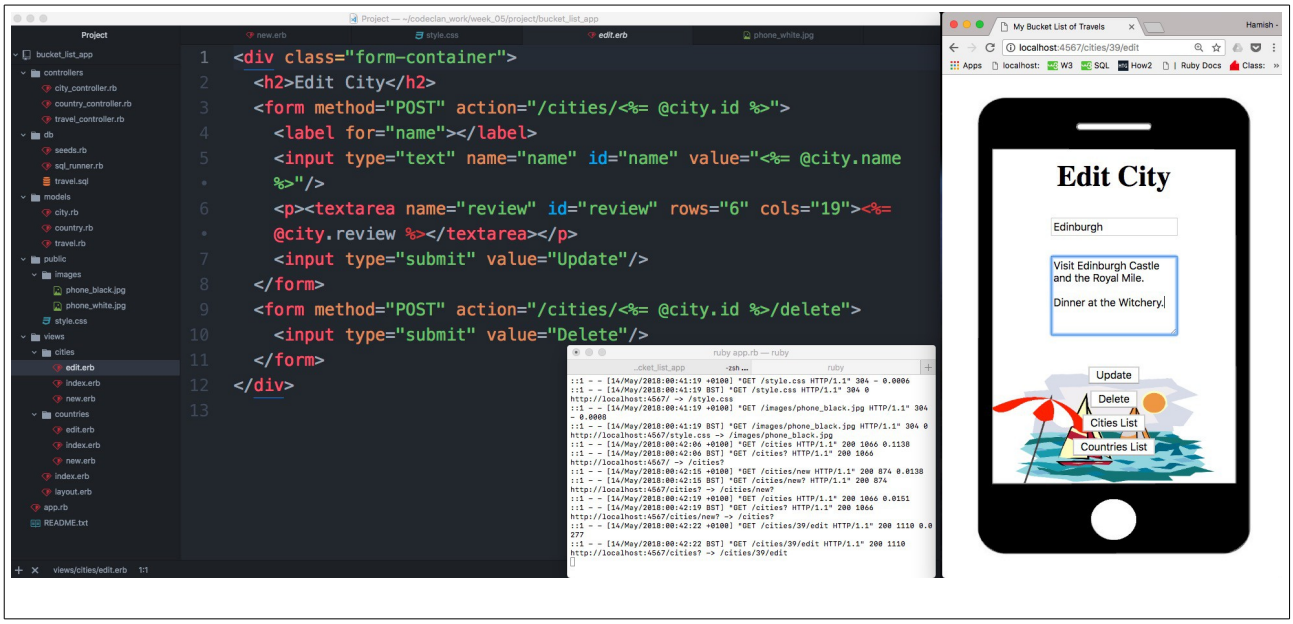
```
==============================
|   CODECLAN FRUIT MACHINE   |
| APPLE 3x£30 4x£90 5x£500 ! |
| PEARS 3x£30 4x£90 5x£500 ! |
| LIMES 3x£30 4x£90 5x£500 ! |
==============================
|PEAR|APPLE|PEAR|LIME|APPLE|
==============================
Unlucky this time, try again?
------------------------------
     WINNINGS TOTAL £0
------------------------------
Your remaining credit is £59
------------------------------
You have £40 on your VisaCard!
------------------------------
   1 = play   2 = winnings!
------------------------------
```

Tests



P. 13 User input

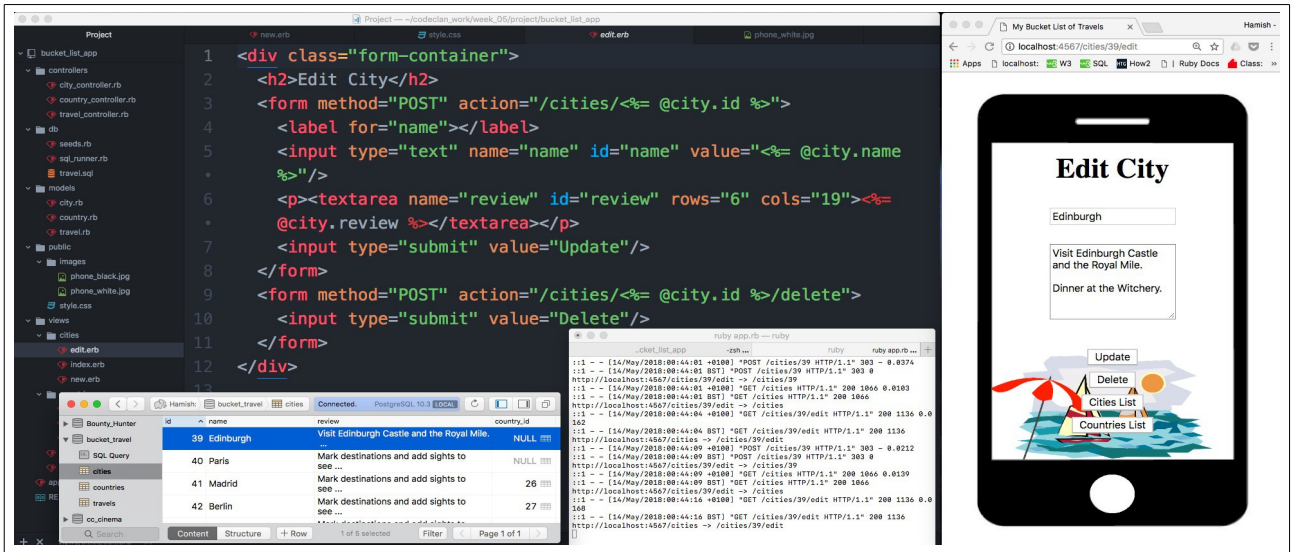
User inputs and edits a City, adding places to visit.



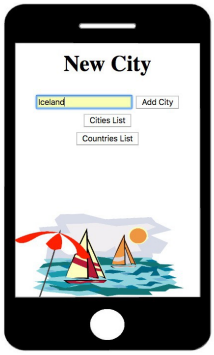
P. 14 Interaction with data persistence

Make sure you show the input being added.

User inputs a City, adds places to visit and saves the information which is then added to the database as shown.



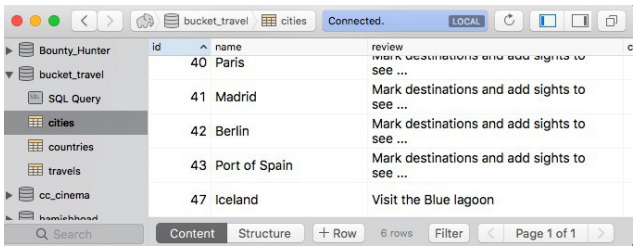
P. 15 User output result



User inputs information and clicks add city which adds information to the database.

The system then returns a list of cities showing the new city has been added to the database.

The user can then update (or delete) city information, such as a place to visit.



The information is then added to the database as shown in the above tables.

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

Evidence for unit

P. 17 Testing your program

Show the test code, the test not passing.....and then the test fixed.

Example of test code

with the test NOT passing

```ruby
class Card
  attr_reader :suit, :value

def initialize(suit, value)
  @suit = suit
  @value = value;
end
end
```

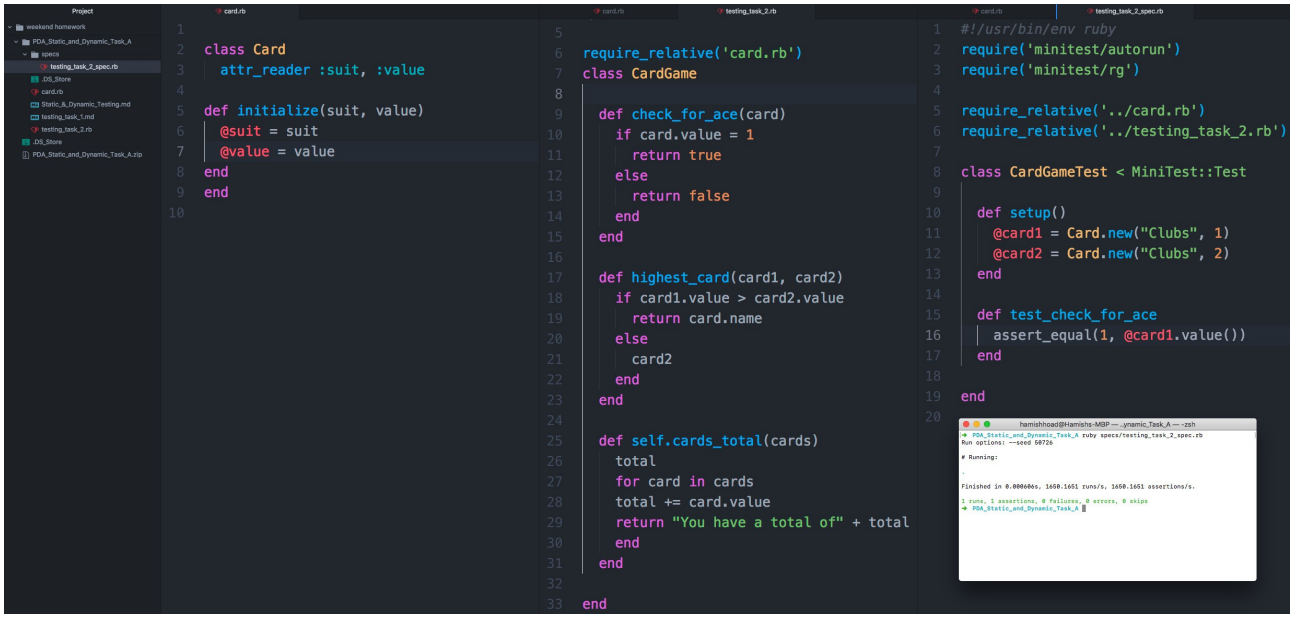
```ruby
require_relative('card.rb')
class CardGame


  def checkforAce(card)
    if card.value = 1
      return true
    else
      return false
    end
  end

  def highest_card(card1, card2)
  if card1.value > card2.value
    return card.name
  else
    card2
  end

  def self.cards_total(cards)
    total
    for card in cards
    total += card.value
    return "You have a total of" + total
  end
end
```

```ruby
class CardGameTest < MiniTest::Test

  def setup()
    @card = CardGame.new("Clubs", 1)
  end

  def test_check_for_ace
    assert_equal("Clubs", @card.value)
  end

end
```

```
hamishhoad@Hamishs-MBP — ...ynamic_Task_A — -zsh
        from specs/testing_task_2_spec.rb:6:in `require_relative'
        from specs/testing_task_2_spec.rb:6:in `<main>'
➜  PDA_Static_and_Dynamic_Task_A ruby specs/testing_task_2_spec.rb
specs/testing_task_2_spec.rb:6:in `require_relative': /Users/hamishhoad/codeclan_work/week_05/weekend homework/PDA_Static_and_Dynamic_Task_A/testing_task_2.rb:18: syntax error, unexpected tIDENTIFIER, expecting '|' (SyntaxError)
  def highest_card(card1, card2)
        from specs/testing_task_2_spec.rb:6:in `<main>'
➜  PDA_Static_and_Dynamic_Task_A ruby specs/testing_task_2_spec.rb
specs/testing_task_2_spec.rb:6:in `require_relative': /Users/hamishhoad/codeclan_work/week_05/weekend homework/PDA_Static_and_Dynamic_Task_A/testing_task_2.rb:32: syntax error, unexpected end-of-input, expecting keyword_end (SyntaxError)
        from specs/testing_task_2_spec.rb:6:in `<main>'
➜  PDA_Static_and_Dynamic_Task_A ruby specs/testing_task_2_spec.rb
specs/testing_task_2_spec.rb:6:in `require_relative': /Users/hamishhoad/codeclan_work/week_05/weekend homework/PDA_Static_and_Dynamic_Task_A/testing_task_2.rb:31: syntax error, unexpected end-of-input, expecting keyword_end (SyntaxError)
        from specs/testing_task_2_spec.rb:6:in `<main>'
➜  PDA_Static_and_Dynamic_Task_A
```

Example of test code

with the test then passing



```ruby
class Card
  attr_reader :suit, :value

def initialize(suit, value)
  @suit = suit
  @value = value
end
end
```

```ruby
require_relative('card.rb')
class CardGame

  def check_for_ace(card)
    if card.value = 1
      return true
    else
      return false
    end
  end

  def highest_card(card1, card2)
    if card1.value > card2.value
      return card.name
    else
      card2
    end
  end

  def self.cards_total(cards)
    total
    for card in cards
    total += card.value
    return "You have a total of" + total
    end
  end

end
```

```ruby
#!/usr/bin/env ruby
require('minitest/autorun')
require('minitest/rg')

require_relative('../card.rb')
require_relative('../testing_task_2.rb')

class CardGameTest < MiniTest::Test

  def setup()
    @card1 = Card.new("Clubs", 1)
    @card2 = Card.new("Clubs", 2)
  end

  def test_check_for_ace
    assert_equal(1, @card1.value())
  end

end
```

```
hamishhoad@Hamishs-MBP — ...ynamic_Task_A — -zsh
➜  PDA_Static_and_Dynamic_Task_A ruby specs/testing_task_2_spec.rb
Run options: --seed 58726

# Running:

.

Finished in 0.000606s, 1650.1651 runs/s, 1650.1651 assertions/s.

1 runs, 1 assertions, 0 failures, 0 errors, 0 skips
➜  PDA_Static_and_Dynamic_Task_A
```