



Terminal app Shopping Lists

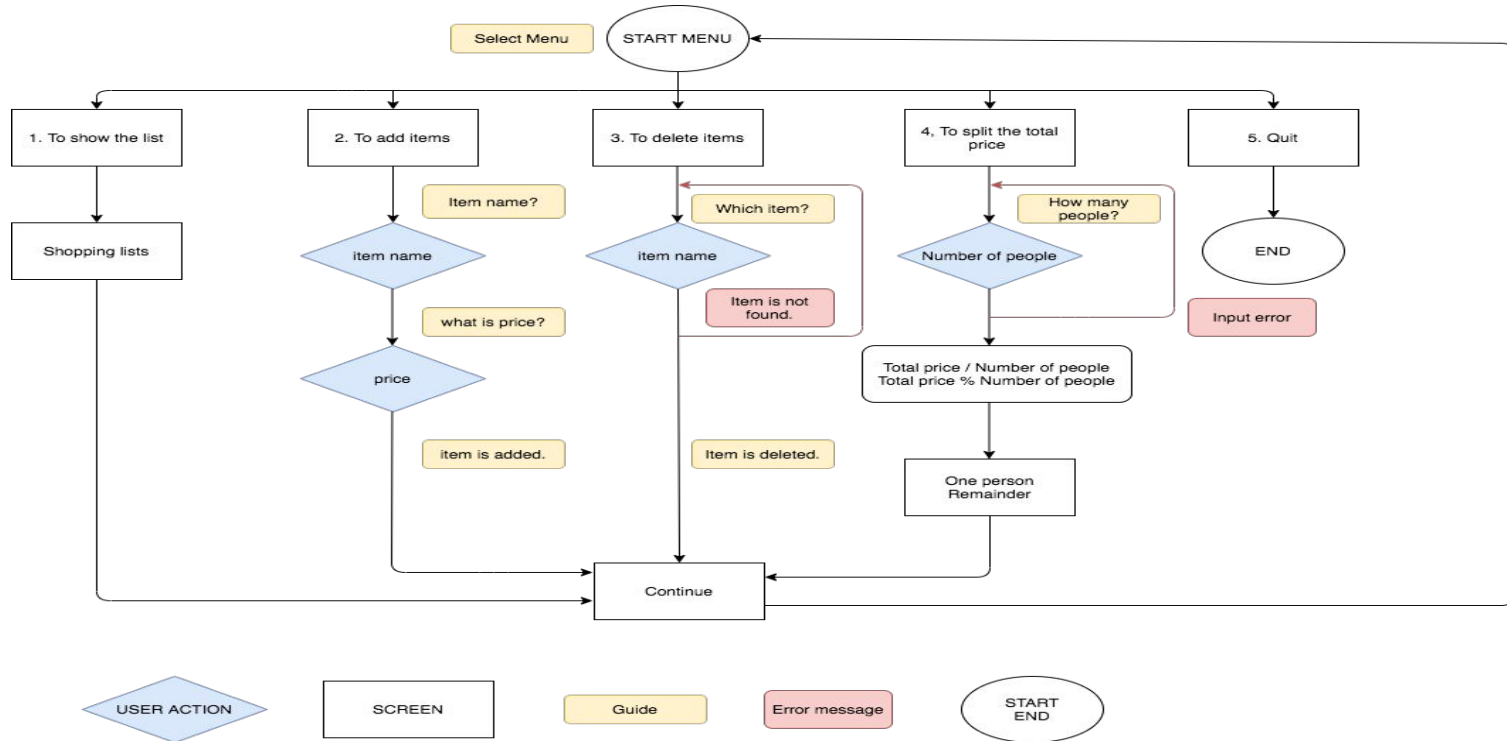
Yuri Ikawa



Features / Functions

1. Check the list
2. Add items to the list
3. Delete items from the list
4. Split the price
5. Exit
6. See total items and total price
7. Loop menu

Overall structure





Code

- Loop (Main menu)
- Error handling
- Input and output
- Calculation

MENU (Loop) ---- *while*

```
while true

  puts "    Press 1) To show the list"
  puts "    Press 2) To add items"
  puts "    Press 3) To delete items"
  puts "    Press 4) To split the total price"
  puts "    Press 5) To quit"

  menu_input = gets.chomp.to_i
  system "clear"

  case menu_input
  when 1
    # Feature 1
  when 2
    # Feature 2
  when 3
    # Feature 3
  when 4
    # Feature 4
  when 5
    # Feature 5
  else
    puts "Choose 1 to 5."
  end

  puts
  puts "🔍" + "Press any key to go back to Top menu"
  gets
  system "clear"
end
```

Error handling ---- *begin, rescue*

Calculation ---- `/` Division, `%` Modulus

```
begin
  puts "Q. How many people?"
  person = gets.to_f
  puts
  puts "|| One person: " + "$#{@number.to_i / person.to_i}".green.italic
  puts "%% Remainder: " + "$#{@number.to_i % person.to_i}".green.italic
  rescue ZeroDivisionError
    puts "ERROR: Please put a number.".red
  retry
end
```

```
def total
  if @number = @lists.values.inject(:+)
    puts " Total items: #{@lists.length}" + " / " + "Total Price: $#{@number}"

  else @number == 0 or @lists.length == 0
    puts " Total items: #{@lists.length}" + " / " + "Total Price: $0"
  end
end
```

Input and output ---- *gets, puts*

Example) Menu 2

```
puts "Q. What do you want to add?"
key = gets.chomp

puts
puts "Q. How much is the price?"
val = gets.to_f

puts
puts "#{key}.green.italic + " / " + "#{val}.green.italic + " is added to the list."

@lists.store(key, val)
```


Development process

1. Decide the theme
2. Make a development plan
3. Research Gems (colorize, artii, tty-prompt(?))
4. Rspec or test code
5. Write a base code
6. Use def method
7. Calculations
8. Design (colorize and artii)

Challenges

- def method
- variable and variable scope
- Test code
- tty-prompt

Future improvement

- Rspec
- calculation for remainder (How to calculate float to two decimal places)

Thank you!