# MediKait App Testing

**Manual Testing Document for MediKait Application - By Joseph B**

## Mass and Volume Testing - going through and testing input

```ruby
def volume_conversion(prompt)
volume_choices = ['KL to L', 'L to mL', 'mL to uL', 'uL to mL', 'mL to L',
'L to KL', 'Back to Main Menu', 'Exit Application']
while true
volume_choice = prompt.select('Please Select: ', volume_choices)
choice = volume_choices.find_index(volume_choice) + 1
if choice >= 1 && choice <= 6
value = get_value('Please enter value to convert: ')
if choice == 1
puts value.to_s + ' KL = ' + (value * 1000).to_s + ' L'.colorize(:blue)
elsif choice == 2
puts value.to_s + ' L = ' + (value * 1000).to_s + ' mL'.colorize(:blue)
elsif choice == 3
puts value.to_s + ' mg = ' + (value * 1000).to_s + ' uL'.colorize(:blue)
elsif choice == 4
puts value.to_s + ' uL = ' + (value / 1000).to_s + ' mL'.colorize(:blue)
elsif choice == 5
puts value.to_s + ' mL = ' + (value / 1000).to_s + ' L'.colorize(:blue)
elsif choice == 6
puts value.to_s + ' L = ' + (value / 1000).to_s + ' KL'.colorize(:blue)
end
else
break
end
end
puts "\e[H\e[2J"
loading_screen("T1A3 Ruby Code Terminal APP ".colorize(:blue))
choice
end
```

Using this code - I have input mutliple commands, letters, numbers, mixed, symbols and even entered Ruby clear commands such as `\e[H\e[2J`. It correctly asks you to put in a correct value and will not pass until you do.

I have done this for all conversion rates in both mass and volume. Both are built off the same code sequence and were unbreakable to the best of my knowledge.

## Definition and Supplement Page

```ruby
def  vitamin_use(prompt)
vitamins  = ['Vitamin D- Helps your body absorb calcium',  'Vitamin C- Immune system Boost/gro
'Iron- Increase Energy, Brain Function and Promotes Healthy Red Blood Cells',  'Zinc- Immune S
'Magnesium- Bone Health/Energy Production (also good for cramps)',  'Folate/Folic Acid- Aids F
puts  'List of Useful supplements'.colorize(:blue)
vitamins.each  do  |item|
puts  item
end
menu_choices  = ['Back to Main Menu',  'Exit Application']
menu_choice  =  prompt.select('Please Select: ',  menu_choices)
puts  "\e[H\e[2J"
loading_screen("T1A3 Ruby Code Terminal APP ".colorize(:blue))
menu_choices.find_index(menu_choice) +  1
end
```

As both the Definitions and the Minerals and Supplements page are both built off the same code structure. There is no input option given. This meaning the use cannot break the code sequence unless a file or code sequence is injected into the source code that is gathered and output. Users with malicious intent could do something with this.