|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ: **09.03.01 Информатика и Вычислительная техника**

**Отчет**

**по лабораторной работе 8**

**Дисциплина: языки интернет-программирования**

**Вариант №15**

Студент гр. **ИУ6-33Б \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** В.К. Залыгин

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** В.Д. Шульман

(Подпись, дата) (И.О. Фамилия)

Москва, 2023

**Цель**

Углубление теоретических сведений о принципах проектирования Model-View-Controller и получениe практических навыков создания веб-приложения с использованием средств Ruby on Rails, построения простейших форм и выполнения вычислений на стороне серверной части приложения.

**Задание**

Создание каркаса RoR-приложения. Настройка базовых параметров. Создание форм. Передача данных от клиента к серверу.

В ходе выполнения лабораторной работы необходимо разработать Rails-приложение, сформировать контроллер, наполнить его соответствующей логикой и сформировать представления. Результаты проверить функциональным тестом контроллера.

**Выполнение**

Созданный контроллер compute\_controller.rb:

# frozen\_string\_literal: true

class ComputeController < ApplicationController

  def input; end

  def is\_palindrome(x)

    x = x.to\_s

    x == x.reverse

  end

  def view

    n = params[:n].to\_i

    @result = (0..n)

              .select { |i| is\_palindrome(i \* i) }

  end

end

input.html.erb:

<h1>Compute#input</h1>

<%= form\_tag("/compute/view", :method => "get") do %>

    <%= label\_tag("N:") %>

    <%= text\_field\_tag(:n) %> <br/>

    <br/>

    <%= submit\_tag("Compute result") %>

<% end %>

view.html.erb:

<h1>Number less N with their palindome squeries:</h1>

<pre>

+-----------------+-----------------+

|       n         |       n\*n       |

+-----------------+-----------------+

<% @result.each do |n| %><div class="row">| <div class="column"></div> | <div class="column"></div> |</div><div class="row">| <div class="column"><%= n %></div> | <div class="column"><%= n\*n %></div> |</div><% end %></div>+-----------------+-----------------+

<div class="row">| <div class="column">count:</div> | <div class="column"><%= @result.length() %></div> |</div>+-----------------+-----------------+

</pre>

Routes.rb:

# frozen\_string\_literal: true

Rails.application.routes.draw do

  get 'compute/input'

  get 'compute/view'

  # Define your application routes per the DSL in https://guides.rubyonrails.org/routing.html

  # Reveal health status on /up that returns 200 if the app boots with no exceptions, otherwise 500.

  # Can be used by load balancers and uptime monitors to verify that the app is live.

  get 'up' => 'rails/health#show', as: :rails\_health\_check

  # Defines the root path route ("/")

  root 'main#index'

end

compute\_controller\_test.rb:

# frozen\_string\_literal: true

require 'test\_helper'

class ComputeControllerTest < ActionDispatch::IntegrationTest

  test "func test" do

    get "/compute/view", params: { n: 10 }

    assert\_response :success

  end

end

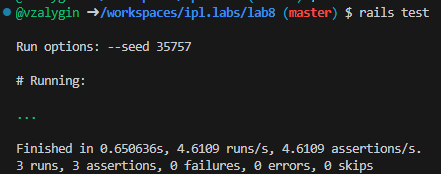


Рисунок 1 - тестран

Rubocop:

@vzalygin ➜ /workspaces/ipl.labs/lab8 (master) $ rubocop

Inspecting 39 files

.....CCCCC.CCC.....C..C..............C.

Offenses:

app/controllers/compute\_controller.rb:3:1: C: Style/Documentation: Missing top-level documentation comment for class ComputeController.

class ComputeController < ApplicationController

^^^^^^^^^^^^^^^^^^^^^^^

app/controllers/compute\_controller.rb:6:7: C: Naming/PredicateName: Rename is\_palindrome to palindrome?.

  def is\_palindrome(x)

      ^^^^^^^^^^^^^

app/controllers/compute\_controller.rb:6:21: C: Naming/MethodParameterName: Method parameter must be at least 3 characters long.

  def is\_palindrome(x)

                    ^

app/controllers/main\_controller.rb:3:1: C: Style/Documentation: Missing top-level documentation comment for class MainController.

class MainController < ApplicationController

^^^^^^^^^^^^^^^^^^^^

app/helpers/application\_helper.rb:3:1: C: Style/Documentation: Missing top-level documentation comment for module ApplicationHelper.

module ApplicationHelper

^^^^^^^^^^^^^^^^^^^^^^^^

app/helpers/compute\_helper.rb:3:1: C: Style/Documentation: Missing top-level documentation comment for module ComputeHelper.

module ComputeHelper

^^^^^^^^^^^^^^^^^^^^

app/helpers/main\_helper.rb:3:1: C: Style/Documentation: Missing top-level documentation comment for module MainHelper.

module MainHelper

^^^^^^^^^^^^^^^^^

app/mailers/application\_mailer.rb:3:1: C: Style/Documentation: Missing top-level documentation comment for class ApplicationMailer.

class ApplicationMailer < ActionMailer::Base

^^^^^^^^^^^^^^^^^^^^^^^

app/models/application\_record.rb:3:1: C: Style/Documentation: Missing top-level documentation comment for class ApplicationRecord.

class ApplicationRecord < ActiveRecord::Base

^^^^^^^^^^^^^^^^^^^^^^^

bin/bundle:24:3: C: Metrics/CyclomaticComplexity: Cyclomatic complexity for cli\_arg\_version is too high. [9/7]

  def cli\_arg\_version ...

  ^^^^^^^^^^^^^^^^^^^

bin/bundle:24:3: C: Metrics/MethodLength: Method has too many lines. [11/10]

  def cli\_arg\_version ...

  ^^^^^^^^^^^^^^^^^^^

bin/bundle:24:3: C: Metrics/PerceivedComplexity: Perceived complexity for cli\_arg\_version is too high. [9/8]

  def cli\_arg\_version ...

  ^^^^^^^^^^^^^^^^^^^

bin/bundle:86:3: C: Metrics/MethodLength: Method has too many lines. [12/10]

  def activate\_bundler ...

  ^^^^^^^^^^^^^^^^^^^^

bin/bundle:99:121: C: Layout/LineLength: Line is too long. [198/120]

    warn "Activating bundler (#{bundler\_requirement}) failed:\n#{gem\_error.message}\n\nTo install the version of bundler this project requires, run `gem install bundler -v '#{bundler\_requirement}'`"

                                                                                                                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

config/application.rb:12:3: C: Style/Documentation: Missing top-level documentation comment for class Lab8::Application.

  class Application < Rails::Application

  ^^^^^^^^^^^^^^^^^

config/environments/development.rb:5:1: C: Metrics/BlockLength: Block has too many lines. [26/25]

Rails.application.configure do ...

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

test/controllers/main\_controller\_test.rb:6:8: C: [Correctable] Style/StringLiterals: Prefer single-quoted strings when you don't need string interpolation or special symbols.

  test "func test" do

       ^^^^^^^^^^^

test/controllers/main\_controller\_test.rb:7:9: C: [Correctable] Style/StringLiterals: Prefer single-quoted strings when you don't need string interpolation or special symbols.

    get "/compute/view", params: { n: 10 }

        ^^^^^^^^^^^^^^^

39 files inspected, 18 offenses detected, 2 offenses autocorrectable

Tip: Based on detected gems, the following RuboCop extension libraries might be helpful:

  \* rubocop-capybara (https://rubygems.org/gems/rubocop-capybara)

  \* rubocop-rails (https://rubygems.org/gems/rubocop-rails)

You can opt out of this message by adding the following to your config (see https://docs.rubocop.org/rubocop/extensions.html#extension-suggestions for more options):

  AllCops:

    SuggestExtensions: false

Замечания rubocop связаны с невыполненным форматированием самого фреймворка.

**Вывод**

Проведено углубление теоретических сведений о принципах проектирования Model-View-Controller и получены практические навыки создания веб-приложения с использованием средств Ruby on Rails, построения простейших форм и выполнения вычислений на стороне серверной части приложения.