



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

О Т Ч Е Т

по лабораторной работе № 1 2

Название:

Добавление модели. ORM. Разработка БД, подключение,
хранение и поиск данных.

Дисциплина: Языки интернет программирования

Студент

ИУ6-31Б

(Группа)

(Подпись, дата)

Т.Е. Старжевский

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

Д. В. Малахов

(И.О. Фамилия)

Москва, 2022

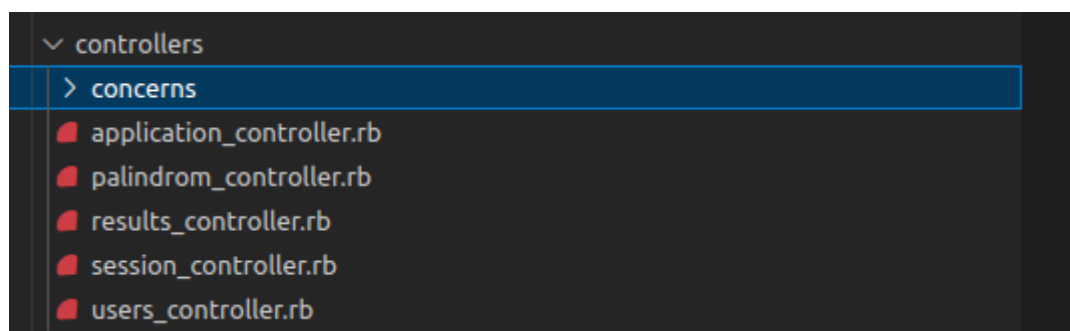
Текст задания:

Модифицировать код приложения ЛР 8 таким образом, чтобы вычисление было невозможно без регистрации пользователя и аутентификации при помощи логина/пароля.

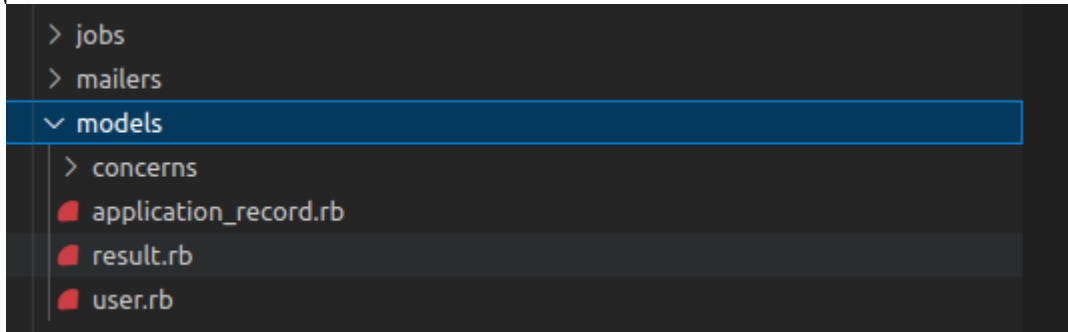
- Сгенерировать при помощи генератора scaffold ресурс для регистрации пользователей.
- Создать БД и выполнить миграцию соответствующим запросом rake.
- Проверить возможность добавления, редактирования информации и получения списка пользователей.
- Удалить отображение поля пароля при просмотре списка пользователей.
- Добавить контроллер сессий.
- Реализовать форму для ввода логина/пароля при обращении по адресу /. Добавить ссылку на регистрацию нового пользователя. При успешном вводе логина/пароля должно осуществляться перенаправление на страницу ввода параметров для вычисления.
- Реализовать при помощи контроллера сессий во всех действиях контроллера проверку о того, прошел ли пользователь аутентификацию или нет (с выдачей соответствующей отладочной информации).
- Вставить фильтры для запроса аутентификации.
- Подготовить интеграционный тест, позволяющий проверить регистрацию нового пользователя, вход под его именем и выполнение вычислений.
- Подготовить интеграционный тест для проверки невозможности выполнения вычислений без ввода логина/пароля.
- Проверить маршруты приложения с помощью rake routes и убрать лишние. Обеспечить доступ при обращении по адресу /.

Выполняем известные ранее команды по созданию приложения

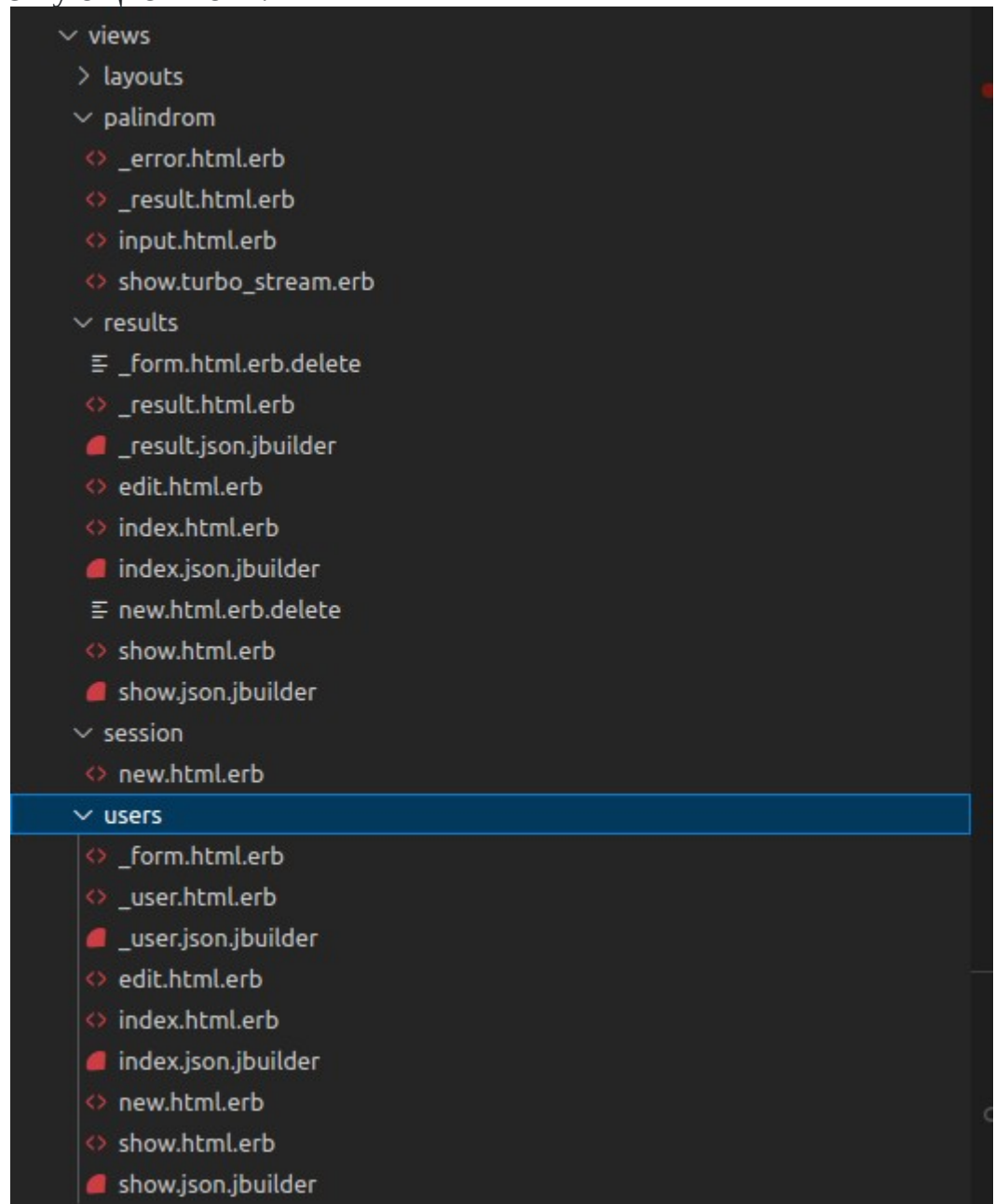
По итогу будет создано четыре контроллера, из которых два сгенерированы командой scaffold, то есть создадут и модели:



Две модели:



Соответствующие вьюхи:



Gemfile adding:

```
# Use Active Model has_secure_password  
[https://guides.rubyonrails.org/active_model_basics.html#securepassword]  
gem 'bcrypt', '~> 3.1.7'  
group :test do
```

```
gem 'capybara'
gem 'rspec-rails'
gem 'selenium-webdriver'
gem 'webdrivers'
end
```

Добавим миграцией поле пароля пользователю:

```
class AddPasswordDigestToUsers < ActiveRecord::Migration[7.0]
  def change
    add_column :users, :password_digest, :string
  end
end
```

А также уникальный индекс в результатах:

```
class AddUniqIndexForResults < ActiveRecord::Migration[7.0]
  def change
    add_index :results, :upper, :unique => true
  end
end
```

Добавим в модели:

app/models/result.rb

```
class Result < ApplicationRecord
  validates :upper, {
    presence: { message: 'Error: Empty params' },
    numericality: {
      only_integer: true,
      greater_than: 0,
      message: "Error: Incorrect params '%<value>s'"
    }
  }

  before_save :set_palindroms

  def decoded_result
    ActiveSupport::JSON.decode(palindroms)
  end

  private

  def set_palindroms
    self.palindroms = ActiveSupport::JSON.encode(palindroms_before)
  end

  def palindroms_before
    [*1..upper].select { |el| el.to_s.reverse == el.to_s && (el**2).to_s.reverse == (el**2).to_s }
  end
end
```

app/models/user.rb

```
class User < ApplicationRecord
  has_secure_password
```

```
validates :login, presence: true
# validates :password, presence: true
# validates_confirmation_of :password, message: 'Не совпадает пароль'
end
```

Выполним миграцию:

Добавим в контроллеры логику (scaffold не упоминаем, там все автоматически создалось):

app/controllers/session_controller.rb

```
class SessionController < ApplicationController
  skip_before_action :require_login, only: %i[new create]

  def new; end

  def create
    user = User.find_by(login: params[:login])

    if user&.authenticate(params[:password])
      flash[:notice] = 'Successfully log in'
      sign_in user
      redirect_to root_path
    else
      flash[:error] = 'Invalid login or password'
      redirect_to session_new_path
    end
  end

  def destroy
    sign_out
    flash[:notice] = 'Loggin out'
    redirect_to session_new_path
  end
end
```

app/controllers/palindrom_controller.rb

```
class PalindromController < ApplicationController
  def show
    if (query = Result.find_by(upper: params[:n]))
      @notice = 'FROM DB'
      @result = query
    else
      palindrom = Result.new(upper: params[:n])
      if palindrom.save
        @notice = 'SAVE TO DB'
        @result = palindrom
      else
        @error = palindrom.errors.messages[:upper]
      end
    end
  end
end
```

```
skip_before_action :require_login, only: %i[new create]
```

app/views/layouts/application.html.erb

```
<!--DOCTYPE html-->
<html>
<head>
<title>RailsAuthV2</title>
<meta name="viewport" content="width=device-width,initial-scale=1">
<%= csrf_meta_tags %>
<%= csp_meta_tag %>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65 "
crossorigin="anonymous">
<%= stylesheet_link_tag "application", "data-turbo-track": "reload" %>
<%= javascript_importmap_tags %>
</head>

<body>
<header>
<nav>
<div class="navbar">
<div class="brand">
<a href="/"><%= image_tag('apple.png') %></a>
</div>

<div>
<%= link_to 'Results', results_path, class: 'nav-link' %>
</div>
<div>
<%= link_to 'Users', users_path, class: 'nav-link' %>
</div>

<div class="auth">
<%= if current_user.present? %>
<div>
<%= link_to "@#{@current_user.login}", "/users/#{@current_user.id}", class: 'nav-link' %>
</div>
<div>
<%= button_to 'Log out', session_destroy_path, method: :delete, class: 'nav-link button' %>
</div>
<% else %>
<div>
<%= link_to 'Sign Up', new_user_path, class: 'nav-link' %>
</div>
<div>
<%= link_to 'Sign in', session_new_path, class: 'nav-link' %>
</div>
```

```

<% end %>
</div>

<div class="burger">
<div class="line1"></div>
<div class="line2"></div>
<div class="line3"></div>
</div>
</div>
</nav>
</header>

<main>
<div class="container">
<%= yield %>
</div>

<% unless flash[:notice].nil? %>
<div class="container">
<p class="notice-text"><%= flash[:notice] %></p>
</div>
<% end %>
<% unless flash[:error].nil? %>
<div class="container">
<p class="error-text"><%= flash[:error] %></p>
</div>
<% end %>
</main>

<footer>
<div class="navbar">
<div>
<%= link_to 'Privacy policy', '', class: 'nav-link' %>
</div>
<div class="vertical-line"></div>
<div>
<%= link_to 'Address', '', class: 'nav-link' %>
</div>
</div>
</div>
</body>
</html>

```

app/views/palindrom/_error.html.erb

```

<div class="error-text center"><%= @error[0] %></div>

```

app/views/palindrom/_result.html.erb

```

<h1 class="color-text"> Palindroms from 1 to <%= params[:n] %> </h1>
<p class="notice-text"><%= @notice %></p>
<div class="table">
<div class="row header">

```

```

<div class="col-4"> # </div>
<div class="col-3"> Number </div>
<div class="col-4"> Number ** 2</div>
</div>
<% @result.decoded_result.each_with_index do |el, idx| %>
<div class="row">
<div class="col-4"> <%= idx+1 %>.</div>
<div class="col-3"> <%= el %> </div>
<div class="col-4"> <%= el**2 %> </div>
</div>
<% end %>
</div>

```

app/views/palindrom/input.html.erb

```

<h1 class="color-text center">Palindroms</h1>
<p class="center">For example 202 and 202 ** 2 = 40804</p>
<%= form_with url: palindrom_show_path, class: "center",
id: "palindrom_form" do |form| %>
<%= form.label :n, "Input end of palindrom`s range:", class: "center" %>
<%= form.number_field :n, class: "text-field", value: "1568" %>
<%= form.submit "Ok", class:"button", id: 'ok' %>
<% end %>
<div id="result" class="container"></div>

```

app/views/palindrom/show.turbo_stream.erb

```

<% if @error.nil?%>
<%= turbo_stream.update 'result', partial: 'result' %>
<% else %>
<%= turbo_stream.update 'result', partial: 'error' %>
<% end %>

```

app/views/session/new.html.erb

```

<div class="container form">
<h2 class="color-text">Authorization</h2>
<%= form_with url: session_create_path, method: :post do |form| %>
<div class="line">
<%= form.label :login, 'Login' %>
<%= form.text_field :login, required: true %>
</div>
<div class="line">
<%= form.label :password, 'Password' %>
<%= form.password_field :password, required: true %>
</div>
<div class="submit">
<%= form.submit :password, value: "Log in", class: 'button', required: true %>
</div>
<% end %>
<div class="line">
<div>Don`t have any account?</div>
<%= link_to 'Sign Up', new_user_path %>
</div>

```


</div>

В пользователях в форме добавим ввод пароля, или уберем его демонстрацию из app/views/users/_user.html.erb если он был сгенерирован автоматически.

config/routes.rb

```
Rails.application.routes.draw do
  root 'palindrom#input'
  post 'palindrom/show'

  get 'session/new'
  post 'session/create'
  delete 'session/destroy'
  resources :results
  resources :users
  # Define your application routes per the DSL in https://guides.rubyonrails.org/routing.html

  # Defines the root path route ("/")
  # root "articles#index"
end
```

Добавим стили и картинку логотипа в images:

app/assets/stylesheets/application.css

```
/*
 * This is a manifest file that'll be compiled into application.css, which will include all the files
 * listed below.
 *
 * Any CSS (and SCSS, if configured) file within this directory, lib/assets/stylesheets, or any
 * plugin's
 * vendor/assets/stylesheets directory can be referenced here using a relative path.
 *
 * You're free to add application-wide styles to this file and they'll appear at the bottom of the
 * compiled file so the styles you add here take precedence over styles defined in any other
 * CSS
 * files in this directory. Styles in this file should be added after the last require_* statement.
 * It is generally better to create a new file per style scope.
 *
 *= require_tree .
 *= require_self
 */
html, body {
  height: 100%;
}
body {
  background: #000;
  color: #b6aeb6;
  font-family: system-ui, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica
  Neue', sans-serif;
  font-weight: 600;
  letter-spacing: -0.022em;
  display: flex;
```

```
flex-direction: column;
}
main {
flex: 1 0 auto;
}
footer {
flex: 0 0 auto;
}
.navbar {
height: 45px;
color: #fff;
margin-bottom: 20px;
padding: 0px;
border-bottom: 1px solid #b6aeb6;
display: flex;
justify-content: space-between;
align-items: center;
}
.navbar .brand img{
height: 30px;
}
.navbar .nav-link {
text-decoration: none;
padding: 4px;
color: #b6aeb6;
}
.navbar .nav-link:hover {
color:#fff;
border-bottom: 1px solid #fff;
}
.navbar .nav-link.button {
color: #b6aeb6;
border: 1px solid transparent;
transition: 0.5s;
border-radius: 6px;
font-size: 16px;
font-weight: 600;
padding: 6px 12px;
background: #000;
}
.navbar .nav-link.button:hover,
.button:hover {
border: 1px solid #fff;
color: #fff;
background: #21262d;
cursor: pointer;
}
.navbar .auth {
display: flex;
align-items: center;
justify-content: space-around;
flex: 0.05 0 auto;
```

```
}

footer .navbar {
  justify-content: center;
}

footer .navbar .vertical-line{
  width: 1px;
  height: 15px;
  margin: 0px 12px;
  background-color: #fff;
}

/* Burger */
.burger {
  display: none;
  cursor: pointer;
}

.burger *{
  content: ' ';
  width: 25px;
  height: 2px;
  margin: 5px;
  background-color: #fff;
  transition: all .4s ease;
}

/* Content */
.container {
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
}

.container a {
  padding: 6px 12px;
  transition: 0.5s;
  border-radius: 6px;
  text-decoration: none;
  color: #fff;
}

.container a:hover {
  background: #21262d;
  cursor: pointer;
}

.container .links {
  padding: 12px;
}

.container.form .line{
  display: flex;
  justify-content: space-between;
  margin: 12px;
}

.container.form .line *{
  padding: 4px 8px;
```

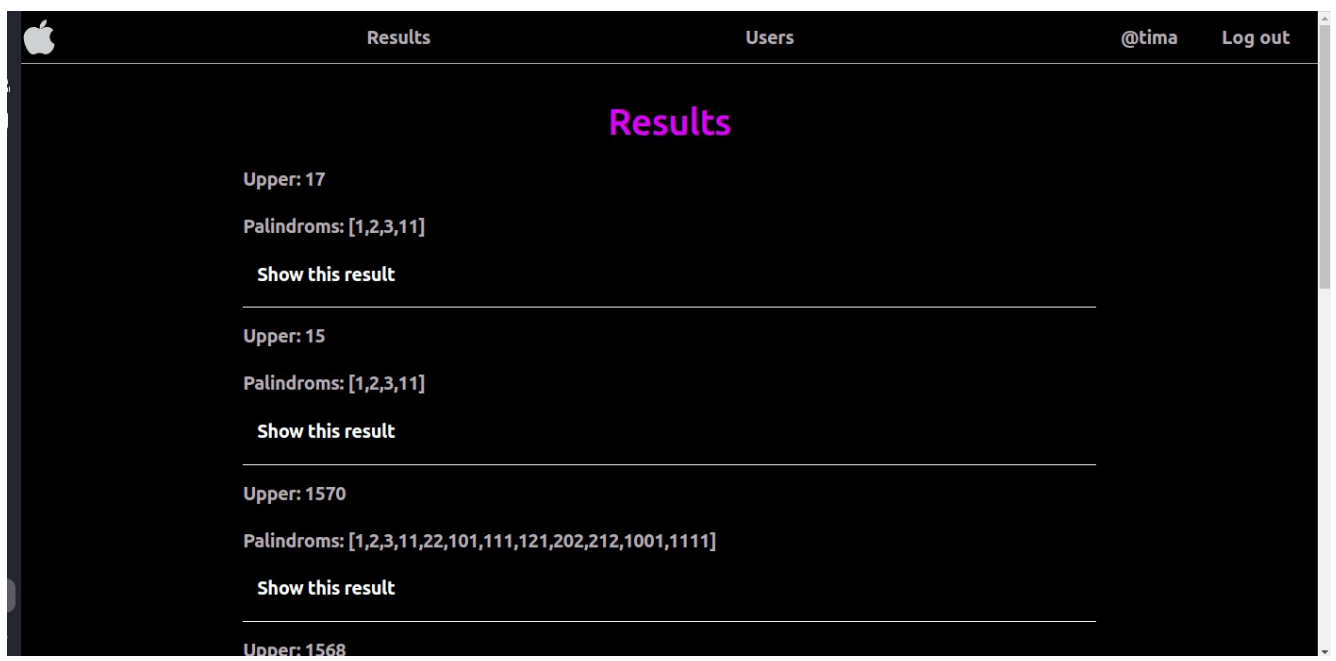
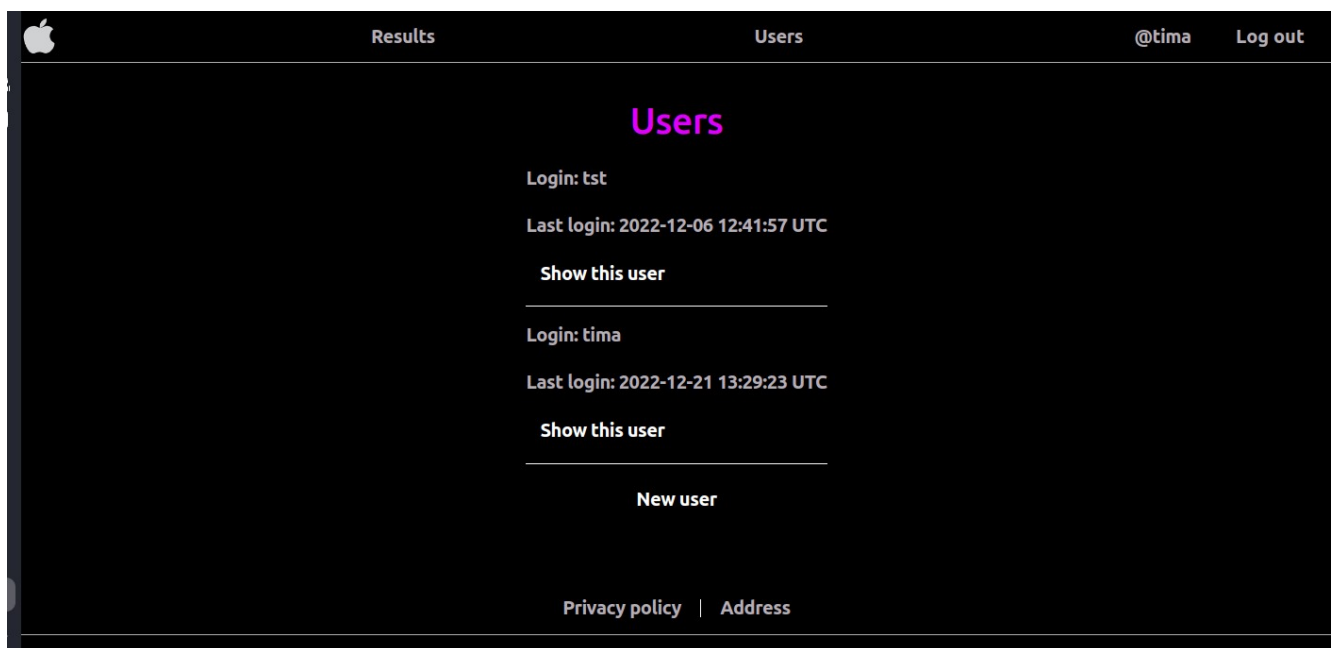
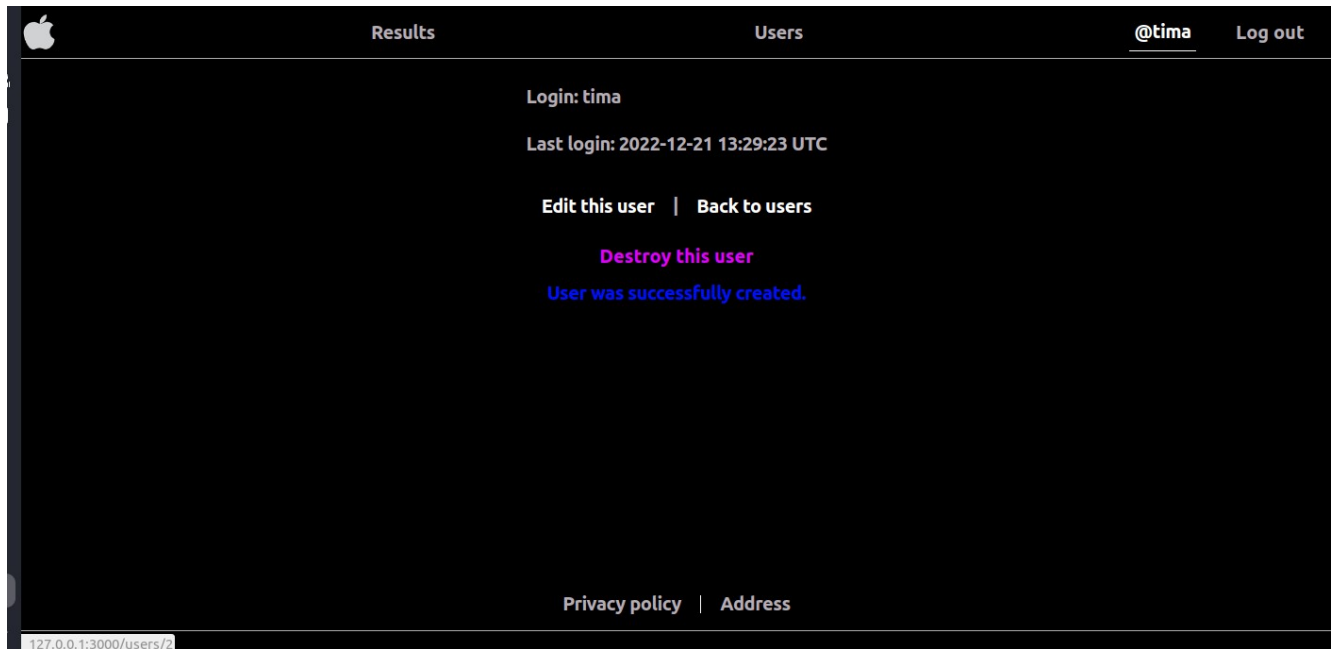
```

}
.container.form .submit{
display: flex;
justify-content: flex-end;
}
.container .element {
border-bottom: 1px solid #fff;
margin-bottom: 12px;
}
/* Content button */
.button {
color: #e100ff;
border: 1px solid transparent;
transition: 0.5s;
border-radius: 6px;
font-size: 16px;
font-weight: 600;
padding: 6px 12px;
background: #000;
}
.button:hover {
border: 1px solid #fff;
background: #21262d;
cursor: pointer;
}
/* Table */
.container .table {
font-size: 20px;
color: #b6aeb6;
text-align: center;
}
.container .table .header{
color: #fff;
}
.container .color-text {
padding: 10px;
color: #e100ff;
}
.container .notice-text {
color: rgb(0, 17, 250);
}
.container .error-text {
color: rgb(250, 0, 0);
}

```

Запустим приложение:

```
rails s
```



Results

Users

@tima

Log out

Palindroms

For example 202 and 202 ** 2 = 40804

Input end of palindrom's range: 1568

Ok

Palindroms from 1 to 1568

FROM DB

#	Number	Number ** 2
1.	1	1
2.	2	4
3.	3	9
4.	11	121
5.	22	484
6.	101	10201

@tima

Log out

Без логина нельзя открыть другие пути:

Results

Users

Sign Up

Sign in

Authorization

Login

Password

Log in

Don't have any account? Sign Up

Please Sign in

Results

Users

Sign Up

Sign in

Register

Login

Password

Password confirmation

Submit

Already have an account? Sign in

Privacy policy | Address

Тестирование:

spec/system/users_spec.rb

```
require 'rails_helper'

RSpec.describe 'Users', type: :system do
  describe 'GET ' do
    context 'Should return' do
      before do
        User.create(login: 'azam', password: '1111')
      end

      scenario 'success while registration' do
        visit new_user_path
        fill_in :user_login, with: 'tst'
        fill_in :user_password, with: '1111s'
        fill_in :user_password_confirmation, with: '1111s'
        find('.button').click
        sleep(0.1)
        expect(current_path).to eq '/users/2'
      end

      scenario 'results after success login' do
        visit root_path
        fill_in :login, with: 'azam'
        fill_in :password, with: '1111'
        find('.button').click

        sleep(0.1)

        visit root_path
        find('#ok').click
        expect(current_path).to eq(root_path)
      end
    end

    scenario 'shouldn't redirect without auth' do
      routes = [root_path, users_path, '/users/1', session_new_path, results_path, new_result_path]
      routes.each do |route|
        visit route
        expect(current_path).to eq session_new_path
      end
    end
  end
end
```

Запустим тесты:

```
rspec
```

```
● timofey@timofey-ASUS:~/Projects/git bmstu/lab12/Rails_auth$ bundle exec rspec
DEBUGGER: Attaching after process 15947 fork to child process 16009
...

Finished in 3.01 seconds (files took 0.87702 seconds to load)
3 examples, 0 failures

○ timofey@timofey-ASUS:~/Projects/git bmstu/lab12/Rails_auth$
```

arter 🍌 Ln 43, Col 1 (1118 selected) Spaces: 2 UTF-8 LF Ruby Go Live 🔍 🔔

Вывод: Научился работать с сессиями и сохранять их в куки в Rails приложении, шифровать пароли пользователя, а также писать системные тесты используя сарубара & Rspec.