Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Московский государственный технический университет имени Н.Э. Баумана»

(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ____Информатика и системы управления_______

КАФЕДРА ______Компьютерные системы и сети(ИУ6)_________

Отчет

по зачетной работе

Дисциплина: ___Языки интернет-программирования_________________

Тема зачетной работы: _______Tube BMSTU — сервис для публикации объявления о тубусах ____________

Студент гр. ___**ИУ6-32Б** _________________ ____Д.В.Лабзунова_____

(Подпись, дата) (И.О. Фамилия)

Преподаватель _________________ _________________

(Подпись, дата) (И.О. Фамилия)

Москва, 2020

Задача

Сайт позволяет разместить объявление о ненужном тубусе, что достаточно актуально, так как предметы, такие как начертательная геометрия или инженерная графика, у многих длятся всего пару семестров. И возможность после их окончания отдать их нуждающемуся, например, первокурснику была бы полезной.

На сайте пользователь = объявление, пользователи не могут существовать без него. Поэтому регистрация = добавление объявления. Для тех юзеров, кто хочет найти себе тубус, есть доступ к главной странице с объявлением и без авторизации. Там по нажатию на кнопку «Показать больше» отображаются контакты для связи с автором понравившегося объявления.

Код реализации

application_controller.rb:

```
# frozen_string_literal: true

# Application controller
class ApplicationController < ActionController::Base
  include SessionHelper
  before_action :set_locale

  def current_user
    @_current_user ||= session[:current_user_id] &&
                       User.find_by_id(session[:current_user_id])
  end

  private

  def authenticate
    redirect_to signin_path unless current_user
  end

  def set_locale
    I18n.locale = params[:locale] || I18n.default_locale
  end

  def default_url_ortions(options = {})
    { locale: I18n.locale }.merge option
  end
end
```

main_controller.rb:

```
# frozen_string_literal: true

# Controller for logic
class MainController < ApplicationController
  before_action :set_user, only: %i[index profile edit profile_edit]
  before_action :require_login, only: %i[profile edit profile_edit]

  def index
    @users = User.all
  end

  def profile; end

  def set_user
    @user = User.find_by_id(session[:current_user_id])
  end
```

```ruby
  def profile_edit; end

  def edit
    if !request.get?
      if @user.update(user_params)
        redirect_to main_profile_path
      else
        render :edit
      end
    else redirect_to main_profile_edit_path
    end
  end

  private

  def user_params
    params.permit(:username, :headline, :tube_description, :link, :email, :password, :password_confirmation)
  end

  def require_login
    redirect_to session_new_url unless signed_in?
  end
end
```

session_controller.rb:

```ruby
# frozen_string_literal: true

# Controller for sessions
class SessionController < ApplicationController
  before_action :already_logged, only: %i[login create new]
  def new; end

  def login; end

  def create
    user = User.find_by username: params[:session][:username]

    if user&.authenticate(params[:session][:password])
      p 'Login'
      sign_in user
      redirect_to root_url
    else
      p 'redirect'
      redirect_to session_new_url
    end
  end

  def logout
    sign_out
    redirect_to session_new_url
  end

  def already_logged
    redirect_to main_index_url if signed_in?
  end
end
```

users_controller.rb:

```ruby
# frozen_string_literal: true

# Controller for users
class UsersController < ApplicationController
  before_action :set_user, only: %i[show edit update destroy]
  before_action :already_logged, only: %i[new create]

  def show; end

  def new
    @user = User.new
  end

  def edit; end

  def create
    @user = User.new(user_params)

    respond_to do |format|
      if @user.save
        format.html { redirect_to main_index_path, notice: 'User was successfully created.' }
      else
        format.html { render :new }
        format.json { render json: @user.errors, status: :unprocessable_entity }
      end
    end
  end

  def update; end

  def destroy; end

  private

    def set_user
      @user = User.find(params[:id])
    end

    def user_params
      params.require(:user).permit(:username, :headline, :tube_description, :link, :email, :password, :password_confirmation)
    end

    def already_logged
      redirect_to main_index_url if signed_in?
    end
end
```

session_helper.rb:

```ruby
# frozen_string_literal: true

# helper for auth
module SessionHelper
  def sign_in(user)
    session[:current_user_id] = user.id
    self.current_user = user
  end

  def signed_in?
    !current_user.nil?
  end
  def sign_out
    session[:current_user_id] = nil
    self.current_user = nil
  end

  def current_user=(user)
    @current_user = user
  end
end
```

user.rb модель:

```
class User < ApplicationRecord
  has_secure_password
  validates :username, presence: true, length: { maximum: 30 }, uniqueness: true
  validates :email, presence: true, uniqueness: true
  validates :password_digest, confirmation: true, length: { minimum: 4 }
  validates :headline,  length: { maximum: 30 }
  validates :tube_description, length: { maximum: 500 }
  validates :link,  length: { maximum: 70 }
end
```

schema.rb:

```
define(version: 2020_12_14_140432) do


  create_table "users", force: :cascade do |t|
    t.text "username"
    t.text "password_digest"
    t.text "email"
    t.text "headline"
    t.text "tube_description"
    t.text "link"
    t.datetime "created_at", precision: 6, null: false
    t.datetime "updated_at", precision: 6, null: false
  end

end
```

en.yml:

```
en:
  lables:
    main: 'tube BMSTU'
    slogan: 'Give your tube a new life'
    login: 'Login'
    signup: 'Add an advertisement'
    profile: 'Your advertisement'
    profile_edit: 'Change your advertisement'
    link: 'Link to contact: '
    email: 'Email: '

  fields:
    username: 'Username'
    password: 'Password'
    password_confirmation: 'Password confirmation'
    email: 'Email'
    link: 'Link to contact'
    headline: 'headline'
    description: 'description'

  buttons:
    login: 'Login'
    signup: 'Signup'
    logout: 'Logout'
    edit: 'Edit'
    save: 'Save'
    show: 'Show more'

  links:
    back: 'Back'
    to_registrate: 'New here?'
    to_login: 'I have an account'
```

ru.yml:

```
ru:
  lables:
    main: 'тубус.МГТУ'
    slogan: 'Подари своему тубусу вторую жизнь'
    login: 'Авторизация'
    signup: 'Регистрация объявления'
    profile: 'Профиль'
    profile_edit: 'Изменить свое объявление'
    link: 'Ссылка для связи: '
    email: 'Электронная почта: '

  fields:
    username: 'Имя пользователя'
    password: 'Пароль'
    password_confirmation: 'Подтверждение пароля'
    email: 'Электронная почта'
    link: 'Сслыка для связи'
    headline: 'Заголовок'
    description: 'Описание'

  buttons:
    login: 'Войти'
    signup: 'Зарегистрироваться'
    logout: 'Выйти'
    edit: 'Редактировать'
    save: 'Сохранить'
    show: 'Показать больше'

  links:
    back: 'Назад'
    to_registrate: 'Впервые тут?'
    to_login: 'У меня уже есть аккаунт'
```

routes.rb:

```
        .application.routes.draw do
  scope "(:locale)", locale: /#{        .available_locales.join("|")}/ do
    get 'users/new'
    get 'session/new'
    get 'session/login'
    post 'session/create'
    root 'main#index'
    get 'main/index'
    get 'main/profile'
    post 'main/edit'
    get 'main/profile_edit'
    get '/signin', to: 'session#new'
    get '/signup', to: 'users#new'
    get '/logout', to: 'session#logout'
    resources :users
  end
end
```

application.rb:

```
require_relative 'boot'

require 'rails/all'

.require(*Rails.groups)

module Project
  class Application < Rails::Application
    config.load_defaults 6.0
    config.i18n.available_locales = [:ru, :en]
    config.i18n.default_locale = :en
  end
end
```

application.html.erb:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Project</title>
    <%= csrf_meta_tags %>
    <%= csp_meta_tag %>

    <%= stylesheet_link_tag 'application', media: 'all', 'data-turbolinks-track': 'reload' %>
    <%= javascript_pack_tag 'application', 'data-turbolinks-track': 'reload' %>
    <%= javascript_pack_tag 'contacts', 'data-turbolinks-track': 'reload' %>
    <%= stylesheet_link_tag 'https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css' %>
    <%= stylesheet_link_tag 'https://fonts.googleapis.com/css2?family=Montserrat:wght@300&display=swap' %>

    <script>
      function show(username)
      {
        document.getElementById(username.id).style.visibility = "visible";
      }
    </script>

    <style>
      body{
        font-family: 'Montserrat', sans-serif;
      }
      .navbar-default {
         color: #bbdefb;
         background-color: #bbdefb;
         border-color: #bbdefb;
       }
      .navbar-brand{
        font-size: 32px;
        color: #212121;
      }

      .inline{
        display: inline-block;
        vertical-align: middle;
      }

      .block{
        display: block;
      }

      .card{
            background-color: #e3e3e3;
        }

        .profile_page{
            margin-left: auto;
            margin-right: auto;
            width: 24em
        }
    </style>
  </head>

  <body>
    <%= yield %>
  </body>
</html>
```

_navbar.html.erb: (main)

```
<nav class="navbar navbar-default navbar-expand-lg">
    <div class="container-fluid">
        <a class="navbar-brand" href="../main/index"><%= t("lables.main") %>
            <img src="/assets/draft.png" style="width:64px;">
        </a>
        <div class="collapse navbar-collapse" id="navbarText">
      <span class="form-inline mt-2 mt-md-0 ml-auto mr-auto" style="color: #212121; font-size:20px">
        <%= t("lables.slogan") %>
      </span>
          <% if @user %>
            <div class ='ml-auto'>
                <img src='/assets/tube2.png' class='inline' style="width: 70px">
                <div class = 'inline'>
                    <div class="navbar-right-links">
                      <div class="inline">
<!--                        <a class="nav-link block profile_box" style="color: #212121; font-size:24px"
href="../main/profile">My tube</a>-->
                        <a class="nav-link block profile_box" style="color: #212121; font-size:24px" href="../main/profile"><%=
user.username %></a>
                      </div>
                      <a class="nav-link inline profile_box" href="../logout"><%= t('buttons.logout') %></a>
                    </div>
                </div>
            </div>
          <% else %>
            <div class ='ml-auto'>
              <a href="../signin"><button type="button" class="btn btn-outline-secondary"  ><%= t("buttons.login")
%></button></a>
              <a href="../signup"> <button type="button" class="btn btn-outline-secondary"><%= t("buttons.signup")
%></button></a>
            </div>
          <% end %>
          <div>

          <div style="margin-left: 32px">
            <li class ="block">
              <%= link_to "Рус", { locale: :ru }, class: "nav-link" %>
            </li>
            <li class ="block">
              <%= link_to "Eng", { locale: :en }, class: "nav-link" %>
            </li>
          </div>
        </div>
      </div>
    </div>
</nav>
```

index.html.erb: (main)

```
<%= render 'navbar', user: @user %>

<% @users.each do |ad| %>
  <% if !@current || @current.id != ad.id %>
        <div class="card" style='margin-top: 20px; margin-left: 32px; margin-right: 64px'>
          <div class="card-body row">
            <div class="col-3">
              <img src='/assets/draft.png' style="width: 80px;">
              <div class='likes block' style='margin-top: 10px'>
                <img src='/assets/user.png' class='inline' style="width: 20px">
                <h6 class='inline'><%= ad.username %></h6>
              </div>
            </div>

            <div class="col-5">
              <a class="card-title text-dark" href="#" style="font-size: 32px"><%= ad.headline %></a>
              <p class="card-text" style="font-size: 24px"><%= ad.tube_description %></p>
            </div>

            <div class="col-4">
              <input class='add' type="submit" value="<%= t("buttons.show") %>" onclick='show(<%=ad.username%>)' />
              <div id="<%=ad.username%>" style="visibility: hidden; float: right;">
                <table>
                  <tbody>
                  <tr>
                    <td><%= t("lables.link") %></td>
                    <td><%= ad.link %></td>
                  </tr>
                  <tr>
                    <td><%= t("lables.email") %></td>
                    <td><%= ad.email %></td>
                  </tr>
                  </tbody>
                </table>
              </div>
            </div>

          </div>
        </div>
  <% end %>
<%end %>
```

profile.html.erb: (main)

```
<%= render 'navbar', user: @user %>
<div class="profile_page">
  <h2 class='title inline' colspan='2' style='margin-top: 20px;'><%= t("lables.profile") %></h2>
  <table>
    <tbody>
    <tr>
      <td><strong><%= t("fields.username") %></strong></td>
      <td> <%= @user.username %> </td>
    </tr>
    <tr>
      <td><strong><%= t("fields.email") %></strong></td>
      <td> <%= @user.email %> </td>
    </tr>
    <tr>
      <td><strong><%= t("fields.link") %></strong></td>
      <td> <%= @user.link %> </td>
    </tr>
    <tr>
      <td><strong><%= t("fields.headline") %></strong></td>
      <td> <%= @user.headline %> </td>
    </tr>
    <tr>
      <td><strong><%= t("fields.description") %></strong></td>
      <td> <%= @user.tube_description %> </td>
    </tr>
    </tbody>
  </table>

  <hr>
  <div style="margin-top: 16px;">
    <a href= "../main/profile_edit" class="block" style="margin-bottom: 16px"> <button type="button" class="btn btn-outline-secondary"><%= t("buttons.edit") %></button></a>
    <%= link_to t("links.back"), root_url %>
  </div>
</div>
```

profile_edit.html.erb: (main)

```
<%= render 'navbar', user: @user %>

<%= form_with url: main_edit_url, method: :post, local:true do |form| %>
  <div class="profile_page">
        <h2 class='title inline' colspan='2' style='margin-top: 20px;'><%= t("lables.profile_edit") %></h2>
        <table>
          <tbody>
          <tr>
            <td><%= t("fields.username") %></td>
            <td>
              <%= form.text_field :username %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.email") %></td>
            <td>
              <%= form.text_field :email %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.link") %></td>
            <td>
              <%= form.text_field :link %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.headline") %></td>
            <td>
              <%= form.text_field :headline %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.description") %></td>
            <td>
              <%= form.text_area :tube_description, :cols => "21  ", :rows => "5" %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.password") %></td>
            <td>
              <%= form.password_field :password %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.password_confirmation") %></td>
            <td>
```

```
              <%= form.password_field :password_confirmation %>
            </td>
          </tr>
          <tr>
            <td colspan="2">
              <%= form.submit t("buttons.save") %>
            </td>
          </tr>
          </tbody>
        </table>
        <div style="margin-top: 16px;">
          <%= link_to t("links.back"), main_profile_url %>
        </div>
  </div>

<% end %>
```

new.html.erb: (session)

```
<nav class="navbar navbar-default navbar-expand-lg">
  <div class="container-fluid">
    <a class="navbar-brand" href="../main/index"><%= t("lables.main") %>
      <img src="/assets/draft.png" style="width:64px;">
    </a>
    <div class="collapse navbar-collapse" id="navbarText">
      <span class="form-inline mt-2 mt-md-0 ml-auto mr-auto" style="color: #212121; font-size:20px">
        <%= t("lables.slogan") %>
      </span>
    </div>
    <div style="margin-left: 32px">
      <li class ="block">
        <%= link_to "Рус", { locale: :ru }, class: "nav-link" %>
      </li>
      <li class ="block">
        <%= link_to "Eng", { locale: :en }, class: "nav-link" %>
      </li>
    </div>
  </div>
</nav>

<%= form_for(:session, url: session_create_url, method: :post, local:true) do |f| %>
<main role="main" class="flex-shrink-0" >
  <div class="container">
    <div class="row">
      <div class="col-9" style='position: absolute; left: 80px;'>
        <h2 class='title inline' colspan='2' style='margin-top: 20px;'><%= t("lables.login") %></h2>
        <table>
          <tbody>
          <tr>
            <td><%= t("fields.username") %></td>
            <td>
              <%= f.text_field :username, required: true, label: 'username' %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.password") %></td>
            <td>
              <%= f.password_field :password, required: true, label: 'password' %>
            </td>
          </tr>
          <tr>
            <td colspan="2">
              <%= f.submit t("buttons.login") %>
            </td>
          </tr>
          </tbody>
        </table>
        <div style="margin-top: 16px;">
          <%= link_to t("links.to_registrate"), new_user_path %>
        </div>

      </div>
      <div class="col-3 tags" style='position: absolute; right: 0;'>
<!--        hello-->
      </div>
    </div>
  </div>
</main>
<% end %>
```

_form.html.erb: (users)

```
<%= form_with(model: user, local: true) do |form| %>
  <% if user.errors.any? %>
    <div id="error_explanation">
      <h2><%= pluralize(user.errors.count, "error") %> prohibited this user from being saved:</h2>

      <ul>
        <% user.errors.full_messages.each do |message| %>
          <li><%= message %></li>
        <% end %>
      </ul>
    </div>
  <% end %>



  <div class="container">
    <div class="row">
      <div class="col-9" style='position: absolute; left: 80px;'>
        <h2 class='title inline' colspan='2' style='margin-top: 20px;'><%= t("lables.signup") %></h2>
        <table>
          <tbody>
          <tr>
            <td><%= t("fields.username") %></td>
            <td>
              <%= form.text_field :username %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.email") %></td>
            <td>
              <%= form.text_field :email %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.link") %></td>
            <td>
              <%= form.text_field :link %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.headline") %></td>
            <td>
              <%= form.text_field :headline %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.description") %></td>
            <td>
              <%= form.text_area :tube_description, :cols => "21  ", :rows => "5" %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.password") %></td>
            <td>
              <%= form.password_field :password %>
            </td>
          </tr>
          <tr>
            <td><%= t("fields.password_confirmation") %></td>
            <td>
              <%= form.password_field :password_confirmation %>
            </td>
          </tr>
          <tr>
            <td colspan="2">
              <%= form.submit t("buttons.signup") %>
            </td>
          </tr>
          </tbody>
        </table>
        <div style="margin-top: 16px;">
          <%= link_to t("links.to_login"), session_new_url %>
        </div>
      </div>
      <div class="col-3 tags" style='position: absolute; right: 0;'>
        <!--        hello-->
      </div>
    </div>
  </div>

<% end %>
```

new.html.erb: (users)

```
<nav class="navbar navbar-default navbar-expand-lg">
  <div class="container-fluid">
    <a class="navbar-brand" href="../main/index"><%= t("lables.main") %>
      <img src="/assets/draft.png" style="width:64px;">
    </a>
    <div class="collapse navbar-collapse" id="navbarText">
      <span class="form-inline mt-2 mt-md-0 ml-auto mr-auto" style="color: #212121; font-size:20px">
        <%= t("lables.slogan") %>
      </span>
    </div>

    <div style="margin-left: 32px">
      <li class ="block">
        <%= link_to "Рус", { locale: :ru }, class: "nav-link" %>
      </li>
      <li class ="block">
        <%= link_to "Eng", { locale: :en }, class: "nav-link" %>
      </li>
    </div>
  </div>
</nav>

<%= render 'form', user: @user %>
```

styles.css:

```
.inline{
    display: inline-block;
    vertical-align: middle;
}

.block{
    display: block;
}

.navbar-right-links{
    margin-top: -16px;
}

main{
    margin-top: 100px;
    font-family: 'Montserrat', sans-serif;
}

.navbar-default {
    color: #FBE9E7;
    background-color: #FBE9E7;
    border-color: #4FC3F7;
}

.navbar-default .navbar-brand:hover,
.navbar-default .navbar-brand:focus {
    color: #880E4F;
}

.navbar-brand{
    font-size: 32px;
    color: #212121;
}

.navbar-user{
    text-align: center;
    height: 100;
}

.profile_box{
    color: #212121;
    text-align: center;
}

.border_bar_header{
    color: #4E342E
}

.tags{
    text-align: center;
}

.card-tags{
    float: right;
}


ul.pagination {
    display: inline-block;
    padding: 0;
    margin: 0;
}

ul.pagination li {display: inline;}

ul.pagination li a {
    color: black;
    float: left;
    padding: 8px 16px;
    text-decoration: none;
}

table {
    border-spacing: 8px 20px;
}

.title{
    font-size: 48px;
}

.input_text{
    font-size: 24px;
    margin-right: 20px;
    margin-top: 20px;
}

.text_edit{
    width: 400px;
}

.add-question-button{
    height: 80px;
    width: 144px;
    color:  #212121;
    font-size: 24px;
    background-color: #FBE9E7;
    margin-top: 32px;
}

.add-answer-button{
    height: 40px;
    width: 80px;
    color:  #212121;
    font-size: 16px;
    background-color: #FBE9E7;
    margin-top: 32px;
    margin-bottom: 32px;
}

table{
    border-collapse: separate;
    border-spacing: 8px 16px;
}
```

Код тестов

main_controller_test.rb:

```
require 'test_helper'

class MainControllerTest < ActionDispatch::IntegrationTest
  test "should get index" do # можно получить главную страницу без регистрации
    get main_index_url
    assert_response :success
  end

  test "should redirect from profile without auth" do # нельзя зайти в профиль без регистрации
    get main_profile_url
    assert_redirected_to session_new_url
  end
  test 'should redirect from profile edit without auth' do # нельзя редактировать профиль без регистрации
    get main_profile_edit_url
    assert_redirected_to session_new_url
  end
end
```

session_controller_test.rb:

```
require 'test_helper'

class SessionControllerTest < ActionDispatch::IntegrationTest
  test "should get new" do # получение страницы авторизации
    get session_new_url
    assert_response :success
  end

  test "can't go to profile without auth" do # нельзя зайти в профиль без регистрации
    get main_profile_url
    assert_redirected_to session_new_url
  end
end
```

users_controller_test:

```
require 'test_helper'

class UsersControllerTest < ActionDispatch::IntegrationTest
  test "should get new" do # получение страницы регистрации
    get users_new_url
    assert_response :success
  end

  test "can't go to profile without auth" do # нельзя зайти в профиль без регистрации
    get main_profile_url
    assert_redirected_to session_new_url
  end

end
```

main.test.rb:

```
require 'test_helper'

class MainTest < ActionDispatch::IntegrationTest
  setup do # регистрируем объявление для тестов
    post users_path, params: { user: { email: 'test@mail.ru', password: '1111', username:'test', headline:'Tubus' } }
  end

# с неверными данными нельзя войти
  test 'user with incorrect credentials will be redirected to login page' do
    post session_create_url, params: { session: {  username:'hello', password: '1111' } }
    assert_redirected_to session_new_url
  end

# если данные авторизации правильные — редирект на главную
  test 'user with correct credentials will be redirected to main page' do
    post session_create_url, params: { session: {  username:'test', password: '1111' } }
    assert_redirected_to "/"
  end

# редирект на авторизация при попытке зайти в профиль без авторизации
  test "should redirect from profile without auth" do
    get main_profile_url
    assert_redirected_to session_new_url
  end
```

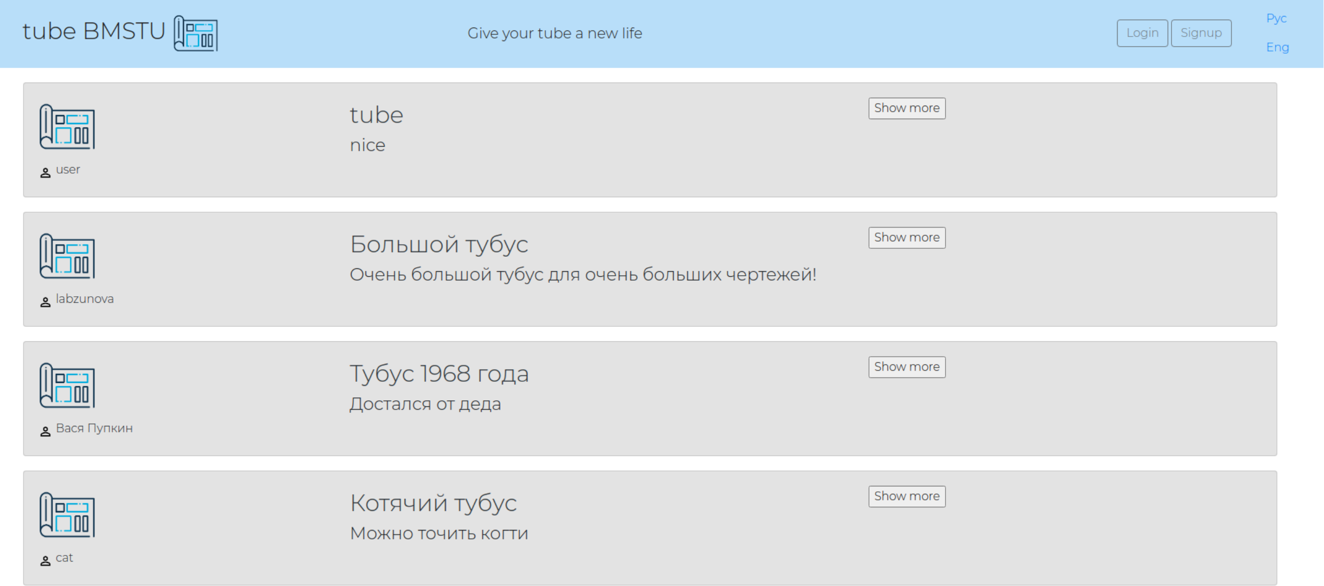
```
# если авторизован — можно зайти в профиль
  test 'should do to profile if authorized' do
    post session_create_url, params: { session: {  username:'test', password: '1111' } }
    get main_profile_url, params: { session: {  username:'test', password: '1111' } }
    assert_response :success
  end

# если авторизован — можно зайти в редактирование профиля
  test 'should do to profile edit if authorized' do
    post session_create_url, params: { session: {  username:'test', password: '1111' } }
    get main_profile_url, params: { session: {  username:'test', password: '1111' } }
    assert_response :success
  end

# можно редактировать страницу
  test 'should do to edit' do
    post session_create_url, params: { session: {  username:'test', password: '1111' } }
    get main_profile_edit_url
    assert_response :success
    post main_edit_url, params: { user: { email: 'new_test@mail.ru', password: '1111', username:'new_test', 
headline:'new_tubus'  } }
    assert_response :redirect
    assert_redirected_to main_profile_url
  end

# можно выйти
  test 'can logout'  do
    post session_create_url, params: { session: {  username:'test', password: '1111' } }
    get main_profile_edit_url
    post main_edit_url, params: { user: { email: 'new_test@mail.ru', password: '1111', username:'new_test', 
headline:'new_tubus'  } }
    get '/logout'
    assert_redirected_to session_new_url
  end
end
```

user_test.rb:

```
require 'test_helper'

class UserTest < ActiveSupport::TestCase
# сохраняется в бд
  test 'should save to db' do
    user = User.new(email: 'test@mail.ru', password: '1111', username: 'test' )
    assert user.save
    user.delete
  end

# нельзя зарегистрироваться с такой же почтой
  test 'check adding not unique email' do
    first = User.create email: 'test@mail.ru', password: '1111', username: 'test1'
    first.save
    second = User.create email: 'test@mail.ru', password: '1111', username: 'test2'
    second.validate
    assert_equal second.errors[:email], ['has already been taken']
  end

# нельзя зарегистрироваться с таким же логином
  test 'check adding not unique username' do
    first = User.create email: 'test1@mail.ru', password: '1111', username: 'test'
    first.save
    second = User.create email: 'test2@mail.ru', password: '1111', username: 'test'
    second.validate
    assert_equal second.errors[:username], ['has already been taken']
  end
```

```
# нельзя зарегистрироваться с пустым логином
  test 'username cant be empty' do
    instance =        .new(email: "test1@mail.ru", password: '1111' )
    instance.validate
    assert_equal instance.errors[:username], ["can't be blank"]
  end

# нельзя зарегистрироваться с пустой почтой
  test 'email cant be empty' do
    instance =        .new(username: "test", password: '1111' )
    instance.validate
    assert_equal instance.errors[:email], ["can't be blank"]
  end

  test 'should get added data from db' do
# получение записанных в бд данных
    user =        .new(email: 'evgeniy@mail.ru', password: '1111', username: 'Евгений', headline: 'My tube' )
    assert user.save
    assert_equal  'Евгений',        .find_by_email('evgeniy@mail.ru').username
    id = user.id
    assert_equal  'My tube',        .find_by_id(id).headline
  end
end
```

Результаты выполнения:



user
tube
nice

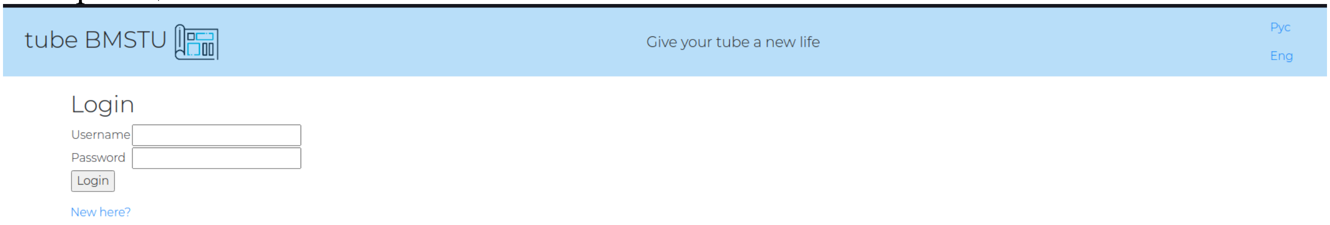
labzunova
Большой тубус
Очень большой тубус для очень больших чертежей!

Вася Пупкин
Тубус 1968 года
Достался от деда

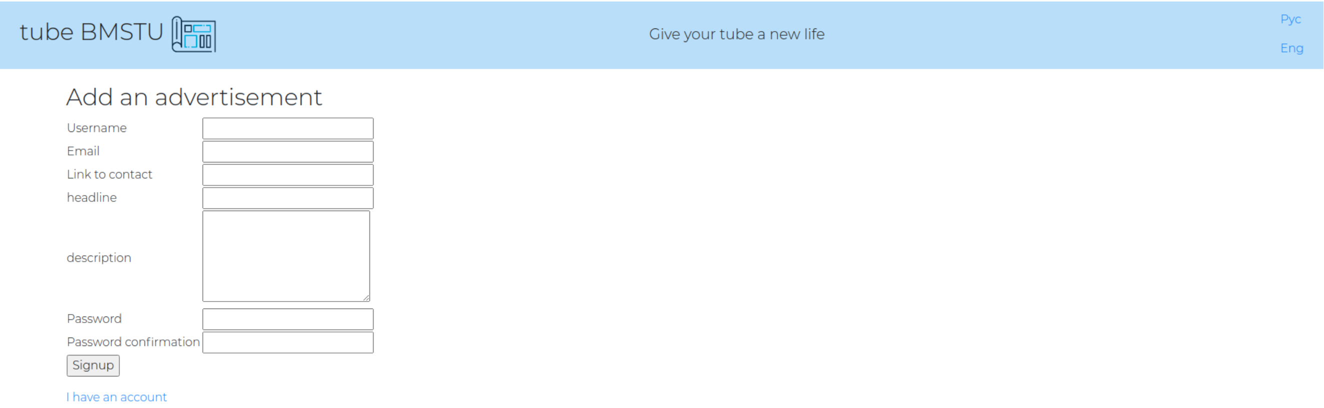
cat
Котячий тубус
Можно точить когти

Главная страница:

Авторизация:



Регистрация:



tube BMSTU

Give your tube a new life

Рус

Eng

Add an advertisement

Username

Email

Link to contact

headline

description

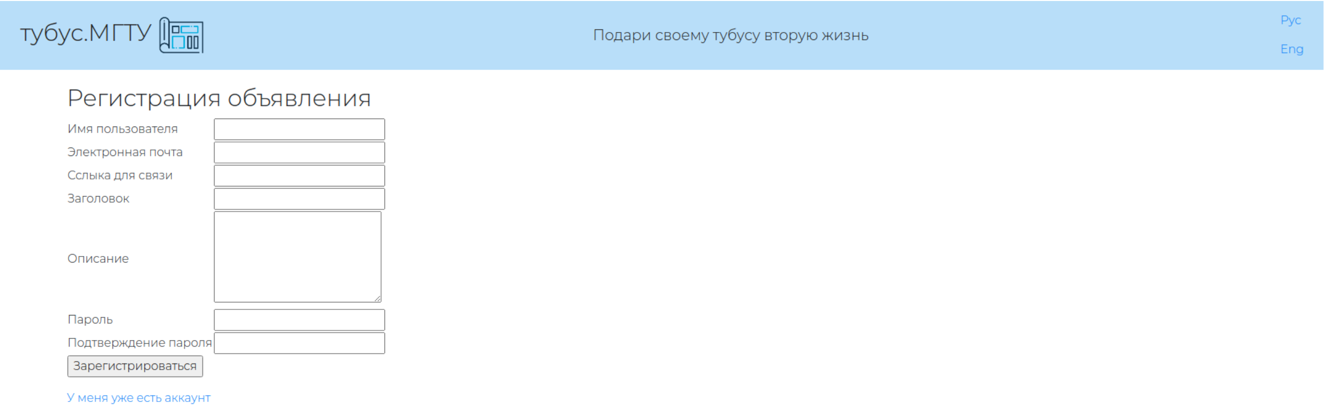
Password

Password confirmation

Signup

I have an account

Работает локализация:



тубус.МГТУ

Подари своему тубусу вторую жизнь

Рус

Eng

Регистрация объявления

Имя пользователя

Электронная почта

Сслыка для связи

Заголовок

Описание

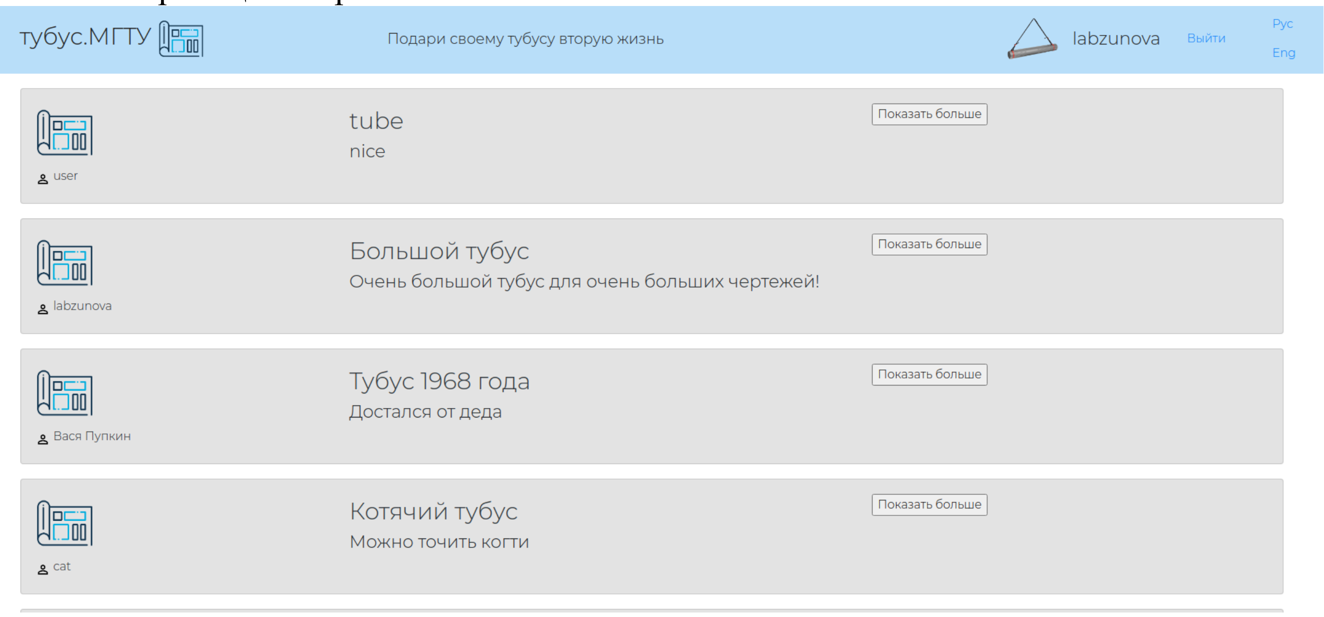
Пароль

Подтверждение пароля

Зарегистрироваться

У меня уже есть аккаунт

Главная страница авторизованного пользователя:



тубус.МГТУ

Подари своему тубусу вторую жизнь

labzunova

Выйти

Рус

Eng

tube

nice

user

Показать больше

Большой тубус

Очень большой тубус для очень больших чертежей!

labzunova

Показать больше

Тубус 1968 года

Достался от деда

Вася Пупкин

Показать больше

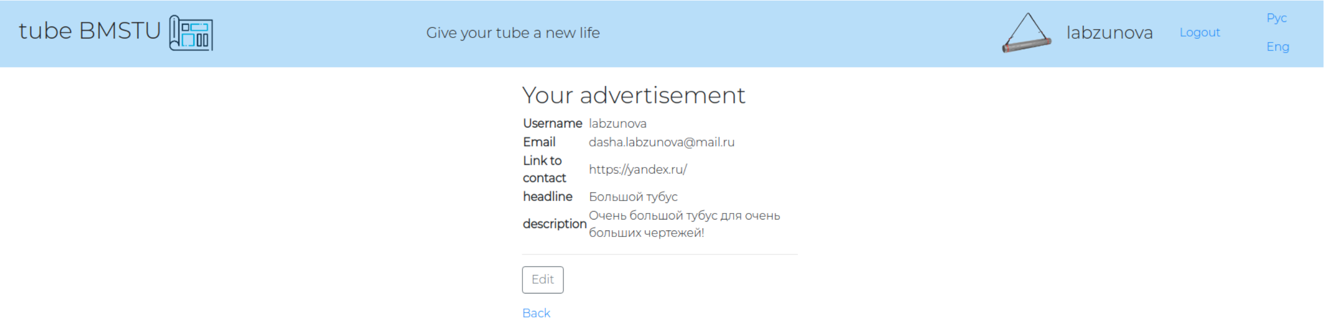
Котячий тубус

Можно точить когти

cat

Показать больше

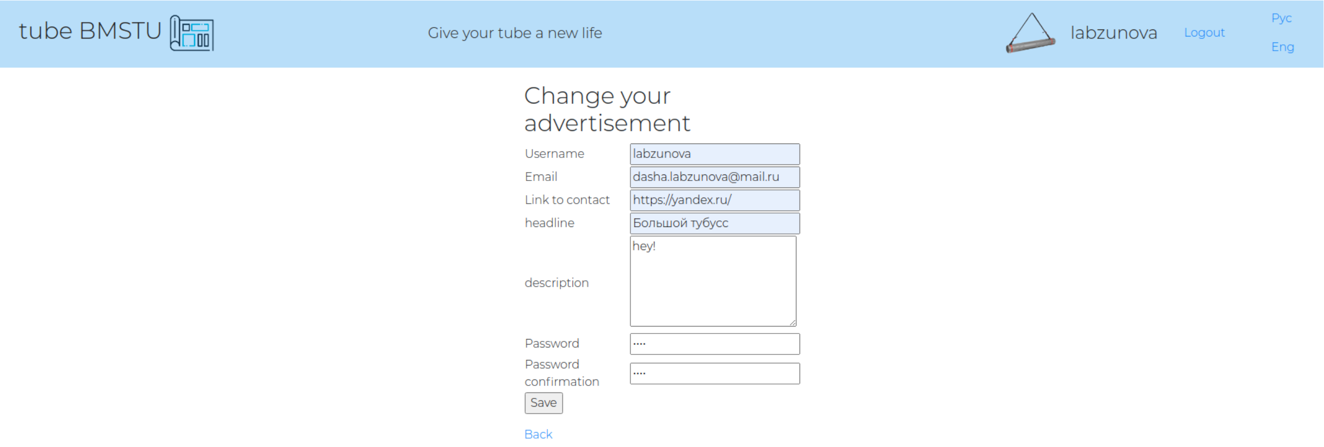
Профиль:



Your advertisement

Username	labzunova
Email	dasha.labzunova@mail.ru
Link to contact	https://yandex.ru/
headline	Большой тубус
description	Очень большой тубус для очень больших чертежей!

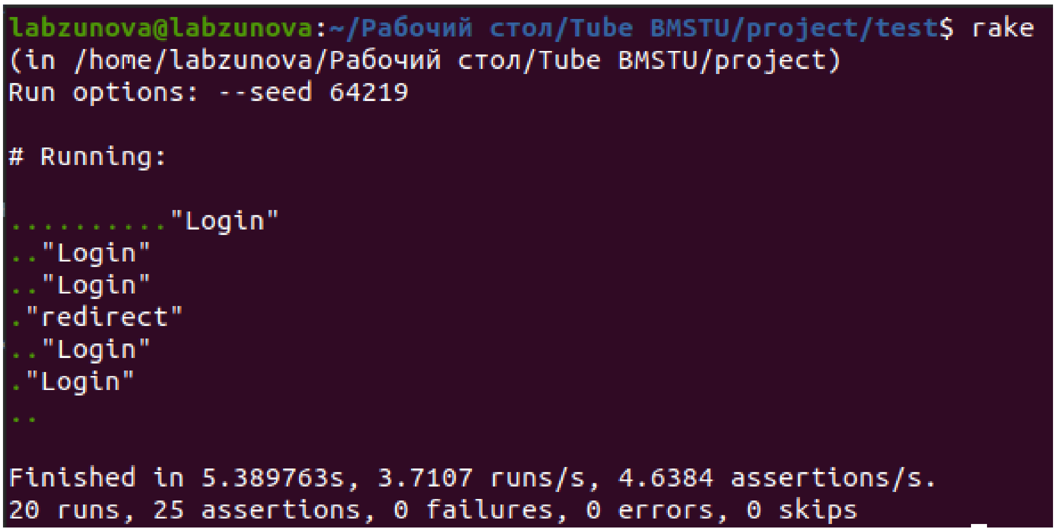
Настройки своего аккаунта-объявления:



Change your advertisement

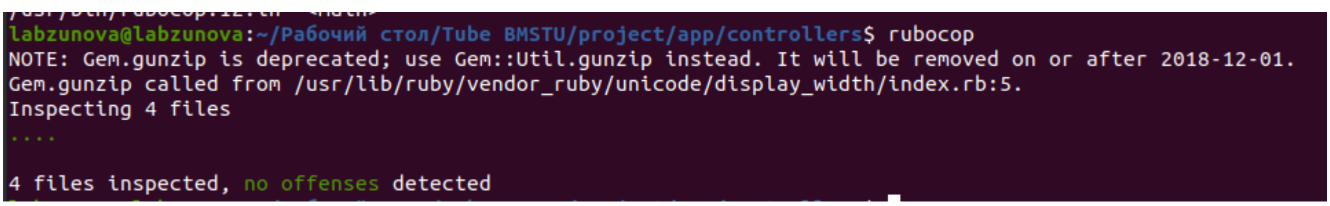
Username	labzunova
Email	dasha.labzunova@mail.ru
Link to contact	https://yandex.ru/
headline	Большой тубусс
description	hey!
Password	••••
Password confirmation	••••

Тесты:



```
labzunova@labzunova:~/Рабочий стол/Tube BMSTU/project/test$ rake
(in /home/labzunova/Рабочий стол/Tube BMSTU/project)
Run options: --seed 64219

# Running:

.........."Login"
.."Login"
.."Login"
."redirect"
.."Login"
."Login"
..

Finished in 5.389763s, 3.7107 runs/s, 4.6384 assertions/s.
20 runs, 25 assertions, 0 failures, 0 errors, 0 skips
```

Проверка rubocop:



```
/usr/bin/rubocop:12:in `<main>'
labzunova@labzunova:~/Рабочий стол/Tube BMSTU/project/app/controllers$ rubocop
NOTE: Gem.gunzip is deprecated; use Gem::Util.gunzip instead. It will be removed on or after 2018-12-01.
Gem.gunzip called from /usr/lib/ruby/vendor_ruby/unicode/display_width/index.rb:5.
Inspecting 4 files
....

4 files inspected, no offenses detected
```