

()

-

- HCCP. , , HCCP, , .

(POST /login) , API HCCP. Ruby:

```
def login
  request_params = {
    url: URI.join(config.url, '/api/login').to_s,
    method: :post,
    payload: {
      params: {
        login: config.login,
        password: config.password
      }
    }.to_json
  }

  render json: RestClient::Request.execute(request_params)
end
```

API HCCP json.

GET . POST hccp_auth_token. :

```

        around_filter :proxy_error

def proxy
  method = request.method.downcase.to_sym
  result = request_widget_backend(params[:path], method, permitted_params)

  if method == :put
    if result
      head :no_content
    else
      head :bad_request
    end
  else
    render json: result
  end
end

def request_widget_backend(path, method = :get, parameters = {})
  request_params = if method == :get
    {
      method: :get,
      url: URI.join(config.url, path).to_s,
      headers: { params: parameters }
    }
  else
    {
      method: method,
      url: "#{URI.join(config.url, path)}?hccp_auth_token=#{params[:
hccp_auth_token]}",
      payload: parameters.to_json
    }
  end

  RestClient::Request.execute(request_params)
end

def proxy_error
  yield
rescue RestClient::Exception => e
  render json: e.response, status: e.response.code
end

private

def permitted_params
  params.symbolize_keys.except(*service_params)
end

def service_params
  %i[controller action path format]
end

```

```
: , proxy_error, , , , .
```

JS

```
JS CSS HCCP , render().
```

```
, render():
```

| | | |
|---------|--|--|
| | 0 | |
| baseURL | URL , ('http://localhost:3001/'), ('/hccp/'). http://localhost:3001/login(). -http://localhost:3003/hccp/login, http://localhost:3003/-URL, . | |

| | | |
|-------------|---------------------------------|--|
| locale | , ('ru') | |
| customerId | id, | |
| containerId | id, ('customer-widget-id') | |

:-

```

!!!
%html
  %head
    %script{type: "text/javascript", src: "http://localhost:3000/customerWidget.js"}
    %link{rel: 'stylesheet', type: "text/css", href: "http://localhost:3000/customerWidget.css"}
  %body
    %div
      %h1 Welcome
      %p
        #customer-widget-id{style: 'width: 1000px; margin: 0 auto; font-size: 13px; border: 5px dashed
lightgrey;'}

:javascript
  document.addEventListener('DOMContentLoaded', () => {
    HCCPCustomerWidget.render({
      baseUrl:      '/hccp/',
      locale:       'ru',
      customerId:   '50162901',
      containerId:  'customer-widget-id',
    });
  });

```