## HTTP Request Structure

API – интерфейс взаимодействия с веб-приложением.

**Request line:** PUT /hr/ergonomics/posture.doc HTTP/1.1

**Headers:** Host: [www.example.com:8080](http://www.example.com:8080)

Content-Length: 1234

**Empty line**

**Body(optional):** Body must include the number of characters specified in the content length header…

Здесь:

PUT – method

/hr/ergonomics/posture.doc – request-URI

HTTP/1.1 – protocol version

## Main HTTP Methods

PUT – create

GET – read

POST – update

DELETE – delete

HEAD – get status string and header from server

CONNECT – create connection with server

OPTIONS – describe connection properties

TRACE – server location

PATCH – изменяет определенную информацию

## Status Codes

1xx – Informational

2xx – Success

3xx – Redirection

4xx – Client Error

5xx – Server Error

- 100 – Continue – часть запросов получена сервером

- 101 – Switching Protocols – сервер выбирает протокол взаимодействия

- 200 – OK – запрос выполнен

- 201 – Created – запись создана

- 202 – Accepted – запрос принят, но обработка еще не завершена

- 203 – Non-authoritative Information – информация в заголовке получена из локальной или сторонней копии (информация получена из кэша)

- 204 – No Content – данные отсутствуют, например, из-за плохого интернет-соединения

- 205 – Reset Content – браузер должен очистить форму для этой транзакции

- 206 – Partial Content – сервер возвратил часть данных

- 300 – Multiple Choices – пользователь может выбрать ссылку и перейти к локации. Максимальное число ссылок должно быть не больше пяти

- 301 – Moved Permanently – страница перенесена на другой адрес

- 302 – Found – страница временное перенесена на другой адрес

- 303 – See Other – страница может быть найдена по другому адресу, по данному адресу ее нет

- 304 – Not Modified – не обновлено

- 305 – Use Proxy – доступ к странице возможен только через прокси

- 306 – Unused – не используется

- 307 – Temporary Redirect – запрашиваемая страница на время перенесена на другой адрес

- 400 – Bad Request – сервер не понял запрос

- 401 – Unauthorized – запрашиваемая страница требует логин и пароль

- 402 – Payment Required – в данный момент клиент не может использовать данный код

- 403 – Forbidden – доступ к запрашиваемой странице запрещен

- 404 – Not Found – страница не найдена

- 405 – Method Not Allowed – метод является недопустимым

- 406 – Not Acceptable – сервер ответил, но клиент его не понял

- 407 – Proxy Authentication Required – клиент должен авторизоваться через прокси сервер

- 408 – Request Timeout – слишком долгий запрос

- 409 – Conflict – запрос невозможен из-за конфликта

- 410 – Gone – страница более недоступна

- 411 – Length Required – элемент content length не определен

- 412 – Precondition Failed – предусловие запроса ошибочное

- 413 – Request Entity Too Large – запрос не будет принят из-за слишком большого тела запроса

- 414 – Request-url Too Long – url слишком длинный

- 415 – Unsupported Media Type – тип данных медиа-файла не поддерживается

- 416 – Requested Range Not Satisfiable – запрашиваемый диапазон байтов находится за пределами допустимого

- 417 – Expectation Failed – ожидания запроса не могут быть удовлетворены сервером

- 500 – Internal Server Error – запрос не выполнен (сервер не доступен)

- 501 – Not Implemented – запрос не выполнен из-за неподдерживаемого функционала

- 502 – Bad Gateway – сервер получил недоступный запрос от предыдущего сервера. Соединение закрыто

- 503 – Service Unavailable – сервер перегружен или не работает

- 504 – Gateway Timeout – сервер недоступен из-за истечения времени сессии

- 505 – HTTP Version Not Supported – сервер не поддерживает данную версию протокола

Параметры запроса передаются с помощью знака «?», после которого идет перечисление параметров.

Формат запроса контента в Headers:

Key: Accept

Value: application/json

После авторизации сервер отправляет клиенту уникальный ключ – token – использование которого необходимо в последующих запросах для работы с зарегистрированным пользователем.

Для этого нужно в Headers выбрать ключ Authorization и в поле value ввести полученный во время предыдущего запроса token.

Формат переменной для ввода в командную строку: **{{url}}**