

# Документация протокола запросов клиентских приложений HI на сервер

*от 30.01.15*

## Содержание:

<a href="#"><u>Информация о сервере</u></a>	3
<a href="#"><u>Регистрация пользователя (ad – запрос)</u></a>	4
<a href="#"><u>Проверка на существование пользователя (cu – запрос)</u></a>	5
<a href="#"><u>Обновление координат (uc – запрос)</u></a>	6
<a href="#"><u>Считывание координат (fc – запрос)</u></a>	7
<a href="#"><u>Добавление друга (af – запрос)</u></a>	8
<a href="#"><u>Удаление друга (df – запрос)</u></a>	9
<a href="#"><u>Список всех друзей пользователя (lf – запрос)</u></a>	10
<a href="#"><u>Восстановление пароля пользователя (gr – запрос)</u></a>	11
<a href="#"><u>&lt;bd&gt; ответ</u></a>	12
<a href="#"><u>Фото-сервер (загрузка фотографии)</u></a>	13

## Информация о сервере

<b>Основной сервер:</b>	109.120.164.212:32000
<b>Фото-сервер:</b>	109.120.164.212:5000
<b>Фотографии пользователей:</b>	<a href="https://109.120.164.212/photos">https://109.120.164.212/photos</a>

## Регистрация пользователя (ad – запрос)

Запрос на добавление нового пользователя в систему.

**Структура запроса:** <ad/user/password/e-mail>

<	Начало запроса
/	Разделитель
ad	Идентификатор запроса
user	Логин пользователя
password	Пароль пользователя
e-mail	E-mail пользователя
>	Конец запроса

**Пример запроса:** <ad/ilya/19081994/ilya@mail.ru>

Запрос позволяет зарегистрировать пользователя с логином *ilya*, паролем 19081994 и e-mail *ilya@mail.ru*

**Ответы сервера:**

<ad/ok>	Пользователь добавлен успешно
<ad/bad>	Пользователь с таким именем уже существует

## Проверка на существование пользователя (cu – запрос)

Запрос проверяет, зарегистрирован ли пользователь в системе и верный ли он вводит пароль. Запрос удобен для использования при входе в приложение.

**Структура запроса:** <cu/user/password>

<	Начало запроса
/	Разделитель
cu	Идентификатор запроса
user	Логин пользователя
password	Пароль пользователя
>	Конец запроса

**Пример запроса:** <cu/ilya/19081994>

Данный запрос даёт команду серверу проверить, зарегистрирован ли пользователь ilya в системе и, если пользователь зарегистрирован, верен ли пароль 19081994.

**Ответы сервера:**

<cu/ok>	Ответ означает, что логин и пароль верны
<cu/not>	Ответ означает, что пользователь не зарегистрирован
<cu/bad>	Ответ означает, что введён неверный пароль

## Обновление координат (uc – запрос)

Этот запрос посылает на сервер свежие координаты пользователя. После обработки запроса сервером, стираются старые данные о местоположении пользователя на стороне сервера. Вместо них сохраняются новые координаты. Запрос не защищён и нуждается в переработке. Для написания прототипа подходит.

**Структура запроса:** <uc/user/latitude/longitude>

<	Начало запроса
/	Разделитель
uc	Идентификатор запроса
user	Логин пользователя, обновляющего координаты
latitude	Широта
longitude	Долгота
>	Конец запроса

**Пример запроса:** <uc/ilya/56.34829341/34.89013456>

Такой запрос означает для сервера, что пользователю с логином `ilya` нужно записать новые координаты: широта - 56.34829341 и долгота - 34.89013456

**Ответы сервера:**

<uc/ok>	Ответ означает, что координаты пользователя обновлены успешно
<uc/not>	Ответ означает, что такой пользователь не зарегистрирован

## Считывание координат (fc - запрос)

Этот запрос позволяет узнать текущие координаты другого пользователя (например, друга). Запрос небезопасен и требует доработки для защиты личных данных пользователей. Для прототипа его использование разрешено.

**Структура запроса:** <fc/user>

<	Начало запроса
/	Разделитель
fc	Идентификатор запроса
user	Имя пользователя, чьи координаты следует получить
>	Конец запроса

**Пример запроса:** <fc/ilya>

Такой запрос даёт команду серверу сформировать ответ, в котором будут координаты пользователя ilya, дата и время последнего обновления геопозиции.

**Ответы сервера:**

*При удаче:* <fc/user/latitude/longitude/week month date time year>

<	Начало запроса
/	Основной разделитель
fc	Идентификатор запроса
user	Имя пользователя
latitude	Широта
longitude	Долгота
" " - пробел	Разделитель данных даты и времени
week*	Неделя – 3 символа
month*	Месяц – 3 символа
date*	День
time*	Время в формате 00:00:00 (чч:мм:сс)
year*	Год – 4 символа
>	Конец запроса
<ul style="list-style-type: none"> <li>• в формате Unix системы</li> </ul>	

Пример удачного ответа на запрос, получения координат пользователя с логином ilya:

<fc/ilya//56.34829341/34.89013456/Sat Nov 22 12:34:18 2014>

*При неудаче* <fc/not>

Ответ означает, что пользователь с таким логином не зарегистрирован.

## Добавление друга (af – запрос)

Запрос уведомляет сервер о желании одного пользователя добавить в друзья другого пользователя.

**Структура запроса:** <af/user/password/friend>

<	Начало запроса
/	Разделитель
af	Идентификатор запроса
user	Логин пользователя
password	Пароль пользователя
friend	Логин пользователя, которого нужно добавить в друзья
>	Конец запроса

**Пример запроса:** <af/ilya/19081994/nikita>

Запрос оповещает сервер о том, что пользователь с логином ilya и паролем 19081994 желает добавить в друзья пользователя с логином nikita

**Ответы сервера:**

<af/ok>	Пользователи подружились
<af/bad>	Неудача



## Удаление друга (df – запрос)

Запрос уведомляет сервер о желании пользователя удалить из друзей другого пользователя.

**Структура запроса:** <df/user/password/friend>

<	Начало запроса
/	Разделитель
df	Идентификатор запроса
user	Логин пользователя
password	Пароль пользователя
friend	Логин пользователя, которого следует удалить из друзей
>	Конец запроса

**Пример запроса:** <df/ilya/19081994/Nikita>

Запрос говорит серверу, что пользователь с логином *ilya* и паролем *19081994* хочет удалить пользователя *Nikita* из списка друзей.

**Ответы сервера:**

<df/ok>	Пользователь успешно удалён из списка друзей или его нет в друзьях
<df/not>	Не верное имя пользователя
<df/bad>	Не верный пароль

## Список всех друзей пользователя (If – запрос)

Основная цель запроса – получить в одном ответе от сервера список всех друзей пользователя.

**Структура запроса:** <If/user/password>

<	Начало запроса
/	Разделитель
If	Идентификатор запроса
user	Логин пользователя
password	Пароль
>	Конец запроса

**Пример запроса:** <If/ilya/19081994>

Данный запрос говорит серверу, что требуется выслать список всех друзей пользователя ilya с паролем 19081994.

**Ответы сервера:**

*При удаче:* <If/friend\_1|friend\_2|...|friend\_n>

<	Начало запроса
/	Основной разделитель
If	Идентификатор запроса
	Разделитель друзей
friend_1	Первый друг
friend_2	Второй друг
friend_n	n-ый друг
>	Конец запроса

Пример ответа на запрос, приведённый выше: <If/nikita|kirill|vasya>

Ответ показывает, что у пользователя ilya 3 друга: nikita, kirill и vasya

*При неудаче:*

<If/not>	Логин пользователя не верен
<If/bad>	Не правильный пароль пользователя

## Восстановить пароль пользователя (gp – запрос)

Сервер, получив этот запрос, отправляет на почту, указанную пользователем при регистрации, пароль от его аккаунта.

**Структура запроса:** <gp/user>

<	Начало запроса
/	Разделитель
gp	Идентификатор запроса
user	Логин пользователя
>	Конец запроса

**Пример запроса:** <gp/ilya>

Данный запрос говорит серверу, что пользователю с логином *ilya* нужно отослать пароль на его e-mail.

**Ответы сервера:**

*При удаче:* <gp/email>

Сервер вернёт нам e-mail на который ушло сообщение с паролем пользователя.

Если связывать этот запрос с примером, тогда мы получим следующий ответ: <gp/ilya@mail.ru>

Мы помним, что именно с таким e-mail мы создавали пользователя (стр. 4, ad - запрос).

*При удаче:* <gp/bad>

## <bd> ответ

Сервер высылает сообщение **<bd>** в ответ на запрос в двух случаях:

- 1). Не соблюдена структура запроса: **<идентификатор/**
- 2). Если идентификатор запроса не соответствует имеющимся идентификаторам.

## Фото-сервер (загрузка фотографии)

Фотографии на сервер загружаются с помощью трёх последовательных запросов.

- соединяемся с сервером

1). Отправляем на сервер размер файла в байтах

2). Отправляем на сервер название файла

3). Отправляем поток байт (файл целиком)

- разрываем соединение