Quark API Documentation

Jazer Barclay

12th May 2022

**Table of Contents**

[1. Overview 1](#_Toc104219637)

[2. Default Routes 2](#_Toc104219638)

[2.1. Signup Route 2](#_Toc104219639)

[2.2. Login Route 3](#_Toc104219640)

[2.3. User routes 4](#_Toc104219641)

[2.4. Unit Route 5](#_Toc104219642)

Overview

This document covers the Quark API publicly available to integrate the core website, apps and external resources with.

All responses are listed in attempt order. So as you check for errors in the request or fail to find the required resources, the errors will occur in the listed order. This will help for development as well as integration testing.

Default Routes

* 1. Signup Route

Route: /register

Method: POST

Request Type: JSON

Request: [email] [username] [password]

Response Type: JSON

Responses:

* Missing Params => (400 Bad request) { error: missing params }
* Invalid Email => (400 Bad request) { error: invalid email address }
* Email Exists => (400 Bad request) { error: email already in use }
* Invalid Username => (400 Bad request) { error: invalid email address }
* Username Exists => (400 Bad request) { error: username already in use }
* Successful Registration => (201 Created) { status: success }
  1. Login Route

Route: /login

Method: POST

Request Type: JSON

Request: [email] [username] [password]

Response Type: JSON

Responses:

* Invalid Token => (400 Bad request) { error: incorrect email or password }
* Missing Params => (400 Bad request) { error: missing params }
* Invalid Email => (400 Bad request) { error: invalid email address }
* Email Exists => (400 Bad request) { error: email already in use }
* Invalid Username => (400 Bad request) { error: invalid email address }
* Username Exists => (400 Bad request) { error: username already in use }
* Invalid Token => (400 Bad request) { error: incorrect email or password }
* Successful Update => (200 OK) { token: [jwt\_token] }
  1. User routes

Route: /user

Method: GET

Request: username=”username”

Response Type: JSON

Responses:

* Missing Params => (400 Bad request) { error: missing params }
* Username Doesn’t Exist => (404 Not Found) { error: user not found }
* Success => (200 OK) { … }

Route: /user

Method: PATCH

Request Type: JSON

Request: [email] [username] [password]

Response Type: JSON

Responses:

* Missing Params => (400 Bad request) { error: missing params }
* Success => (200 OK) { token: [jwt\_token] }
  1. Unit Route

Route: /unit

Method: GET

Request: none

Response Type: JSON

Responses:

* Missing token => (401 Unauthorized) { error: missing token }
* Success => (200 OK) { units }

Route: /unit

Method: GET

Request: [username]

Response Type: JSON

Responses:

* Missing token => (401 Unauthorized) { error: missing token }
* Missing Params => (400 Bad request) { error: missing params }
* Success => (200 OK) { units }

Route: /unit

Method: POST

Request Type: JSON

Request: [label], [start\_time], [end\_time]

Response Type: JSON

Responses:

* Missing token => (401 Unauthorized) { error: missing token }
* Missing Params => (400 Bad request) { error: missing params }
* Success => (201 Created) { }