

SUPER- POWERED API Testing

James Messinger



@BigstickCarpet



<http://apitesting.bigstickcarpet.com>



James Messinger



@BigstickCarpet



GitHub.com/BigstickCarpet



SWAGGER



POSTMAN

apitesting.bigstickcarpet.com

- ▶ See the API that we'll be testing!
- ▶ Download sample tests!
- ▶ Links to **even more** exciting downloads!



API Testing Tools

- ▶ Workflow testing vs Endpoint testing
- ▶ CI/CD Integrations
- ▶ Local-First
- ▶ Cross-Platform
- ▶ Free (as in beer 🍺)



create-character.spec.js - Mocha-Chai - [~/Code/Personal/Training/super-powered-api-testing/Mocha-Chai]

```
describe('POST /characters', function () {
  it('should create a single character', function () {
    return Promise.resolve()
      .then(function () {
        // Create a character
        return chai.request(API_ROOT)
          .post('/characters')
          .set('X-API-Key', API_KEY)
          .send({ name: 'Super Coder', powers: ['10x-ing'] });
      })
      .then(function () {
        // Fetch the character
        return chai.request(API_ROOT)
          .get('/characters/supercoder')
          .set('X-API-Key', API_KEY);
      })
      .then(function (response) {
        // The response should be Super Coder
        response.should.be.successful(200);
        response.body.should.be.a.character({
          name: 'Super Coder',
          type: 'hero',
          powers: ['10x-ing'],
          links: {
            self: API_ROOT + '/characters/supercoder'
          }
        });
      })
  });
});
```

File:///Users/james/Code/Personal/Training/super-powered-api-testing/Mocha-Chai

passes: 19 failures: 0 duration: 4.73s 100%

Test Results

- POST /characters
- GET /characters
 - should return an empty array when there are no characters
 - should return a single character
 - should return multiple characters
 - should return heroes, sidekicks, and villains
- GET /characters/{slug}
 - should return the correct character
 - should return links to the character's sidekick and villain
 - should return a 404 if the character doesn't exist
- GET /characters/(with query params)
 - should return an empty array when there are no characters
 - should find characters by type
 - should find characters by name
 - should find characters by name and type
- PUT /characters/{slug}
 - should update the character's properties
 - should clear any properties that aren't updated
 - should change the character's sidekick and nemesis
 - should not allow setting sidekick or nemesis to characters with the wrong type
 - should not allow setting sidekick or nemesis to characters that don't exist
 - should change the character's URL if its name changes
- GET /characters/(with query params)
 - should return an empty array when there are no characters
 - should find characters by type (104ms)
 - should find characters by name (337ms)
 - should find characters by name and type (252ms)
- PUT /characters/{slug}
 - should update the character's properties (116ms)
 - should clear any properties that aren't updated (119ms)
 - should change the character's sidekick and nemesis (334ms)
 - should not allow setting sidekick or nemesis to characters with the wrong type (165ms)
 - should not allow setting sidekick or nemesis to characters that don't exist (71ms)
 - should change the character's URL if its name changes (138ms)

Tests Passed: 19 passed (2 minutes ago)

19 passing (5s)



- ▶ CLI + Browser
- ▶ IDE Integrations
- ▶ Easily automated
- ▶ Browser testing, FTW!



- ▶ Lots of moving parts
- ▶ No GUI
- ▶ Requires JS skills
- ▶ No fourth bullet point

DEMO TIME!



Restlet

The screenshot shows the Restlet Client interface for testing the Super Tech Heroes API. The left sidebar lists various API endpoints under 'MY DRIVE' and 'Super Tech Heroes API'. The selected endpoint is 'Creates a new character' under the '/characters' folder. The main panel displays the scenario details:

- SCENARIO:** Creates a new character
- DESCRIPTION:** This endpoint creates a new character. For heroes, you can also create the sidekick and/or nemesis at the same time by setting the `sidekick` and/or `nemesis` properties to objects rather than strings.
- METHOD:** POST
- SCHEME:** https://\${API_ROOT}/characters
- HEADERS:** Content-Type: application/json, Accept: application/json
- BODY:**

```
1 {
2   "name": "Super Coder",
3   "powers": [
4     "10-xing",
5     "Caffeine"
6   ],
7   "weakness": "Imposter syndrome",
8   "nemesis": {
9     "name": "Space Bugs from Dimension Z"
10 }
11 }
```
- ASSERTIONS:**
 - Status code Equals 201
 - Header location Equals \${API_ROOT}/characters/super-coder
 - JSON Body \$.name Equals Super Coder
 - JSON Body \$.powers[0] Equals 10-xing

Restlet

- ▶ Chrome extension
- ▶ Intuitive GUI
- ▶ Jenkins & GitHub integrations
- ▶ Freemium SaaS features
- ▶ Simple test builder

Restlet

- ▶ Limited test capabilities
- ▶ Limited reusability
- ▶ No conditional requests
- ▶ Limited integrations

DEMO TIME!





POSTMAN

```
Postman: newman run postman-collection.json -e postman-environment.json
newman

Super Tech Heroes API

characters
  Find characters
    GET http://localhost:8080/characters?name=((name))&type=((type)) [400 Bad Request, 6538, 23ms]
      ✓ Response time is less than 200ms
      ✓ Status code is 200
      ✓ 12 characters were returned

  Creates a new character
    POST http://localhost:8080/characters [201 Created, 8438, 119ms]
      ✓ Response time is less than 200ms
      ✓ Status code is 201
      ✓ The Location header is set
      ✓ The new character was returned

  Deletes all characters
    DELETE http://localhost:8080/characters [200 0]
      ✓ Response time is less than 200ms
      ✓ Status code is 204

  Get a specific character
    GET http://localhost:8080/characters/supercode
      ✓ Response time is less than 200ms
      ✓ Status code is 200
      ✓ The Location header is set
      ✓ The correct character was returned

  Updates a character
    PUT http://localhost:8080/characters/supercode
      ✓ Response time is less than 200ms
      ✓ Status code is 200
      ✓ The Location header is set
      ✓ The updated character was returned

schema
  Get API definition schema
    GET http://localhost:8080/schema?accept=((accept))
      ✓ Response time is less than 200ms
      ✓ Status code is 200
      ✓ The schema is valid

  Get API info
    GET http://localhost:8080/ [200 OK, 880B, 3ms]
      ✓ Response time is less than 200ms
      ✓ Status code is 200
      ✓ The version number is correct



|                    | executed | failed |
|--------------------|----------|--------|
| iterations         | 1        | 0      |
| requests           | 7        | 0      |
| test-scripts       | 7        | 0      |
| prerequest-scripts | 0        | 0      |
| assertions         | 23       | 0      |


total run duration: 532ms
total data received: 15.63KB (approx)
average response time: 40ms

Postman: ┌
```

Creates a new character

Creates a new character

This endpoint creates a new character with a `nemesis` property.

POST

Authorization

form-data

1 {
2 "name":
3 "powers":
4 "weaknesses":
5 "CaffeineSensitive":
6 }
7 "weaknesses":
8 "nemesis":
9 "name":
10 }
11 }

All Passed

PASS Response time

(name))&type={{type}} [400 Bad Request, 6538, 23ms]

Created, 843B, 119ms]

New Import Runner

Super Tech Heroes

History Collections

All Me Team

Super Tech Heroes API
7 requests

characters

Find characters

Creates a new character

Deletes all characters

Get a specific character

Updates a character

schema

Get API definition schema

Get API Info

Creates a new character

This endpoint creates a new character object. It expects `nemesis` properties to objects.

POST {{API_ROOT}}/characters

Authorization Headers (2)

form-data x-www-form-urlencoded

```
1: {  
2:   "name": "Super Xing",  
3:   "powers": [  
4:     "10-xing",  
5:     "Caffeine"  
6:   ],  
7:   "weakness": "Invincibility",  
8:   "nemesis": {  
9:     "name": "Space Invader"  
10:    }  
11: }
```

Body Cookies Header

All Passed Skipped

PASS Response time is less than 100ms

PASS Status code is 201

PASS The Location header is present

PASS The new character was created

The screenshot shows the Postman Collection Runner interface. At the top, there are tabs for "Collection Runner" and "Run Results". On the right side of the header are buttons for "Add Monitors", "Run In Command Line", and "Docs". Below the header, a summary shows 23 PASSED and 0 FAILED tests. The main area displays the "Super Tech Heroes API" collection, which was run "just now". The "Run Summary" shows 400 Bad Request, 30 ms, and 101 B. There are three sections of test results: Iteration 1, Iteration 2, and Iteration 3. Each iteration contains several test cases with their status (e.g., PASS), description, and response details.

Iteration	Test Case	Status	Description	Response
Iteration 1	GET Find characters	PASS	http://localhost:8080/cha...s API / characters / Find characters	400 Bad Request 30 ms 101 B
	PASS	Response time is less than 200ms		
	PASS	Status code is 200		
	PASS	12 characters were returned		
Iteration 2	POST Creates a new character	PASS	http://localhost:8080/cha...racters / Creates a new character	201 Created 102 ms 240 B
	PASS	Response time is less than 200ms		
	PASS	Status code is 201		
	PASS	The Location header is set		
	PASS	The new character was returned		
Iteration 3	DELETE Deletes all characters	PASS	http://localhost:8080/cha... characters / Deletes all characters	200 OK 165 ms 49 B
	PASS	Response time is less than 200ms		
	PASS	Status code is 204		
	GET Get a specific character	PASS	http://localhost:8080/cha...haracters / Get a specific character	404 Not Found 13 ms 73 B
PASS		Response time is less than 200ms		
PASS		Status code is 200		



POSTMAN

- ▶ Native app
- ▶ Flow control via script
- ▶ Intuitive GUI
- ▶ Easily automated
- ▶ JavaScript tests
- ▶ Freemium SaaS features

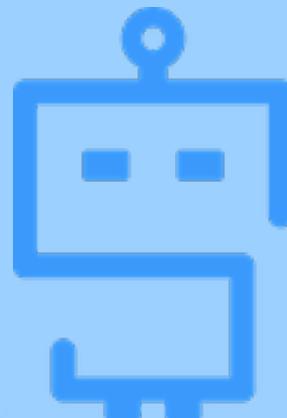


POSTMAN

- ▶ Requires JS skills
- ▶ Limited reusability
- ▶ Limited third-party libs

DEMO TIME!





StopLight

Super Tech Heroes · BigstickCarpet Collections

BigstickCarpet / Collections / Super Tech Heroes

Discussion

+ Save File

Run Scenario Run Collection + New Step

home

BEFORE + None.

SCENARIOS + Super Tech Heroes

settings

steps [6]

- GET Find characters 200
- POST Creates a new character 201
- GET Get a specific character 200
- PUT Updates a character 200
- DELETE Deletes all characters 200
- GET Get API info 200

AFTER + None.

UTILITIES + None.

Normal B

Optional scenario notes...

\$.env ✓

API_ROOT	http://localhost:8080
SLUG	supercoder

Steps + Add Step

	GET Find characters	200	▶	✎	🔗	✖
1	POST Creates a new character	201	▶	✎	🔗	✖
2	GET Get a specific character	200	▶	✎	🔗	✖
3	PUT Updates a character	200	▶	✎	🔗	✖
4	DELETE Deletes all characters	200	▶	✎	🔗	✖
5	GET Get API info	200	▶	✎	🔗	✖

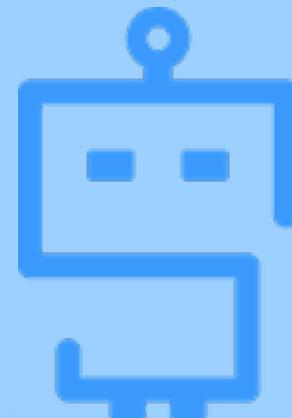
Visual </> Code ✓ Valid

[Conductor] Stoplight API Host is https://api-next.stoplight.io

```
Super Tech Heroes Scenario (1.208s)
  Find characters, GET http://localhost:8080/characters (0.5s)
    ✓ output.status (eq) 200
    ✓ output.body (validate.contract) against JSON Schema
  Creates a new character, POST http://localhost:8080/characters (0.104s)
    ✓ output.status (eq) 201
    ✓ output.body (validate.contract) against JSON Schema
  Get a specific character, GET http://localhost:8080/characters/supercoder (0.025s)
    ✓ output.status (eq) 200
    ✓ output.body (validate.contract) against JSON Schema
  Updates a character, PUT http://localhost:8080/characters/supercoder (0.076s)
    ✓ output.status (eq) 200
  Deletes all characters, DELETE http://localhost:8080/characters (0.495s)
    ✓ output.status (eq) 200
    ✓ output.body (validate.contract) against JSON Schema
  Get API info, GET http://localhost:8080/ (0.008s)
    ✓ output.status (eq) 200
    ✓ output.body (validate.contract) against JSON Schema
```

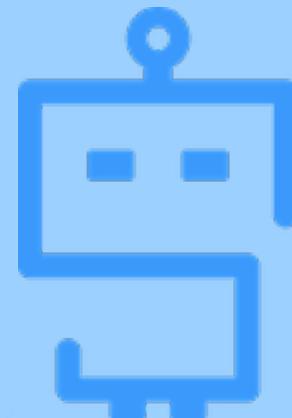
✓ 11 tests passed

Stoplight: []



StopLight

- ▶ Native app
- ▶ Swagger integration
- ▶ GUI test builder
- ▶ Easily automated
- ▶ JavaScript tests
- ▶ Freemium SaaS features



StopLight

- ▶ Complex UI
- ▶ Limited flow control
- ▶ Limited third-party libs



DEMO TIME!



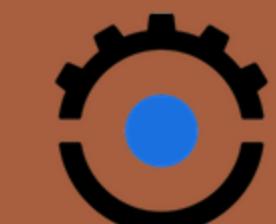
OTHER SUPERHEROES...



APIFORTRESS



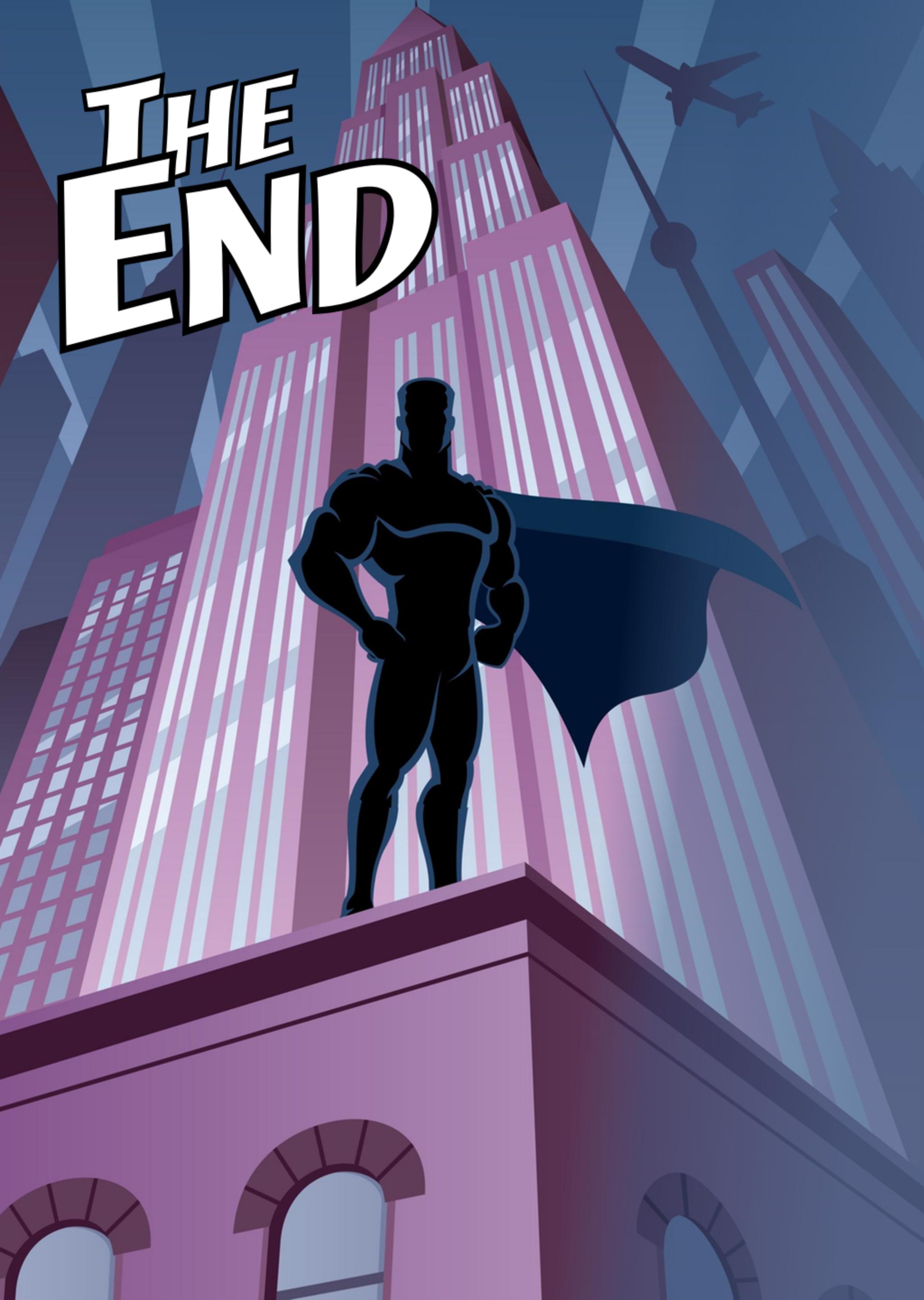
Insomnia



Runscope

Ping-API

THE END

A stylized illustration of a muscular man standing on the edge of a building. He is shirtless, wearing dark pants, and has one hand on his hip. The background features a city skyline with various buildings, a bridge, and an airplane in the sky.

James Messinger



@BigstickCarpet



GitHub.com/BigstickCarpet