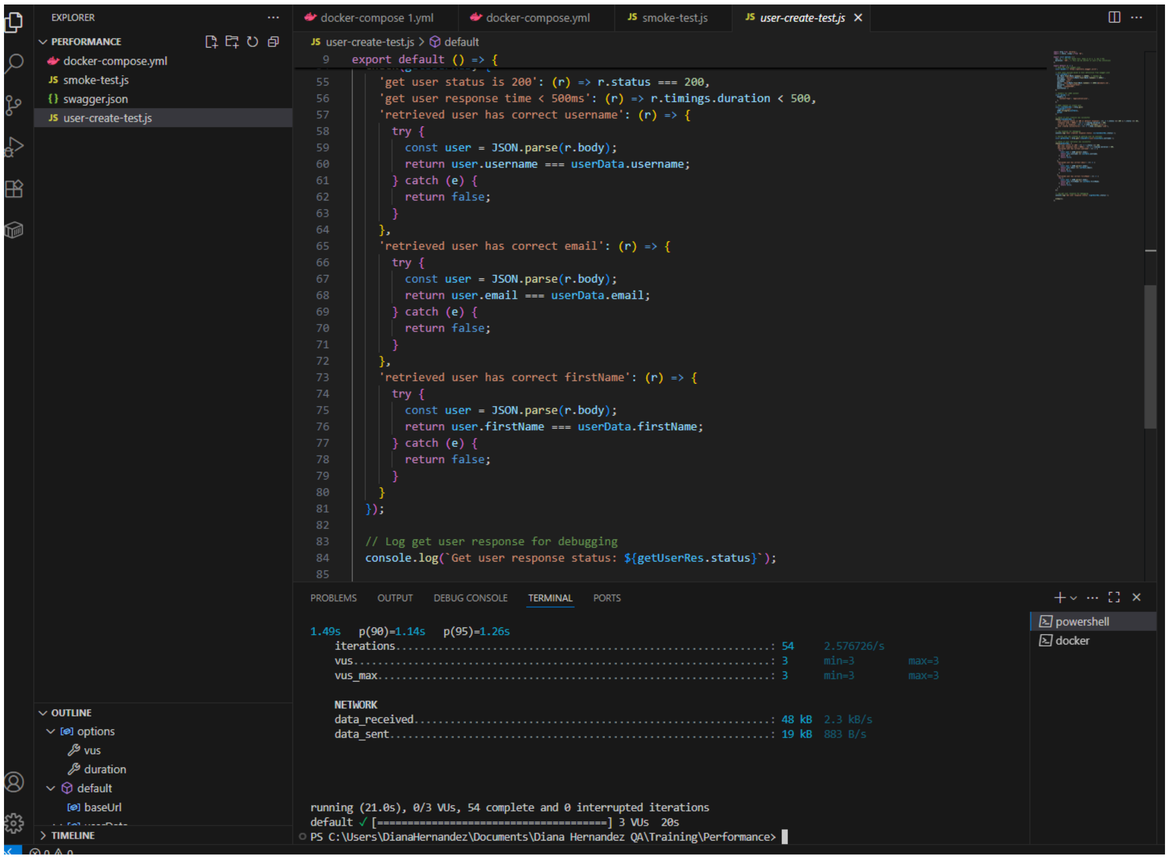
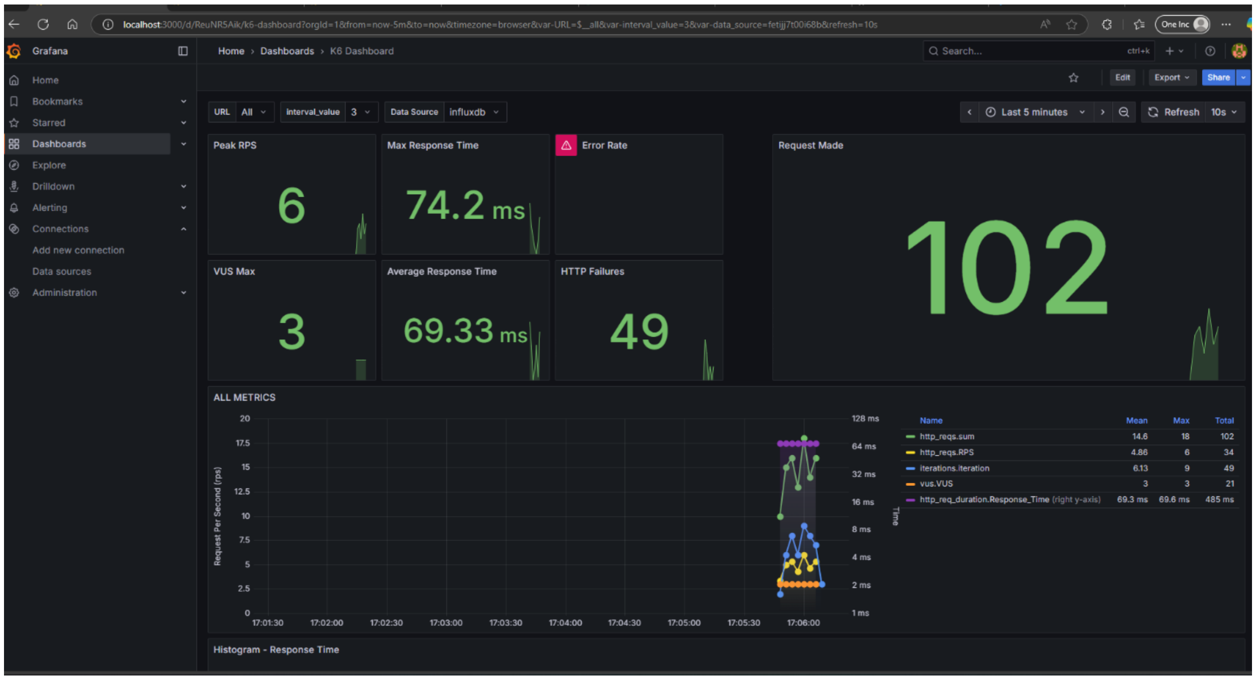
POST / user

User-create-test.js



```javascript
export default () => {
    'get user status is 200': (r) => r.status === 200,
    'get user response time < 500ms': (r) => r.timings.duration < 500,
    'retrieved user has correct username': (r) => {
      try {
        const user = JSON.parse(r.body);
        return user.username === userData.username;
      } catch (e) {
        return false;
      }
    },
    'retrieved user has correct email': (r) => {
      try {
        const user = JSON.parse(r.body);
        return user.email === userData.email;
      } catch (e) {
        return false;
      }
    },
    'retrieved user has correct firstName': (r) => {
      try {
        const user = JSON.parse(r.body);
        return user.firstName === userData.firstName;
      } catch (e) {
        return false;
      }
    }
  });

  // Log get user response for debugging
  console.log(`Get user response status: ${getUserRes.status}`);
```

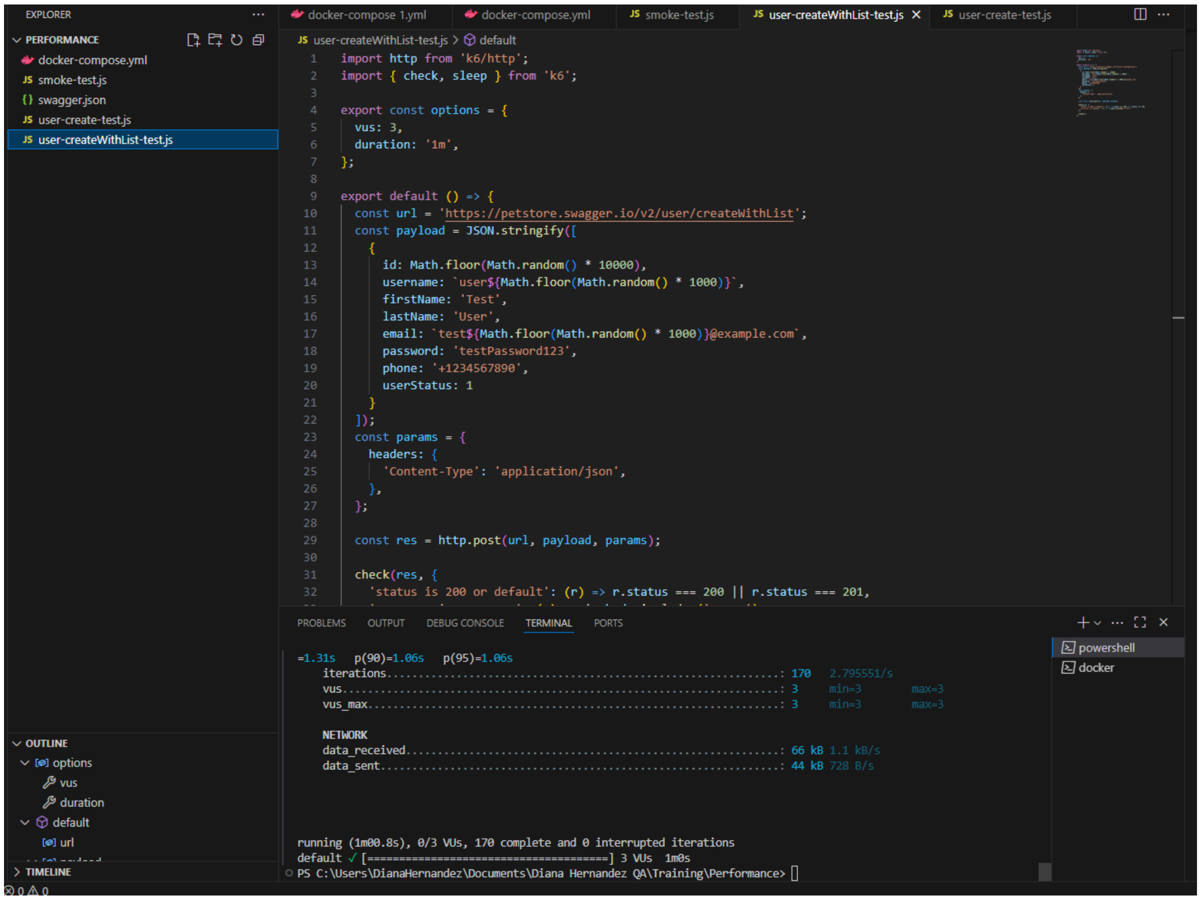
```
1.49s   p(90)=1.14s   p(95)=1.26s
    iterations.............................................................: 54     2.576726/s
    vus....................................................................: 3      min=3        max=3
    vus_max................................................................: 3      min=3        max=3

    NETWORK
    data_received..........................................................: 48 kB  2.3 kB/s
    data_sent..............................................................: 19 kB  883 B/s



running (21.0s), 0/3 VUs, 54 complete and 0 interrupted iterations
default ✓ [====================================] 3 VUs  20s
PS C:\Users\DianaHernandez\Documents\Diana Hernandez QA\Training\Performance>
```



Name	Mean	Max	Total
http_reqs.sum	14.6	18	102
http_reqs.RPS	4.86	6	34
iterations.iteration	6.13	9	49
vus.VUS	3	3	21
http_req_duration.Response_Time (right y-axis)	69.3 ms	69.6 ms	485 ms

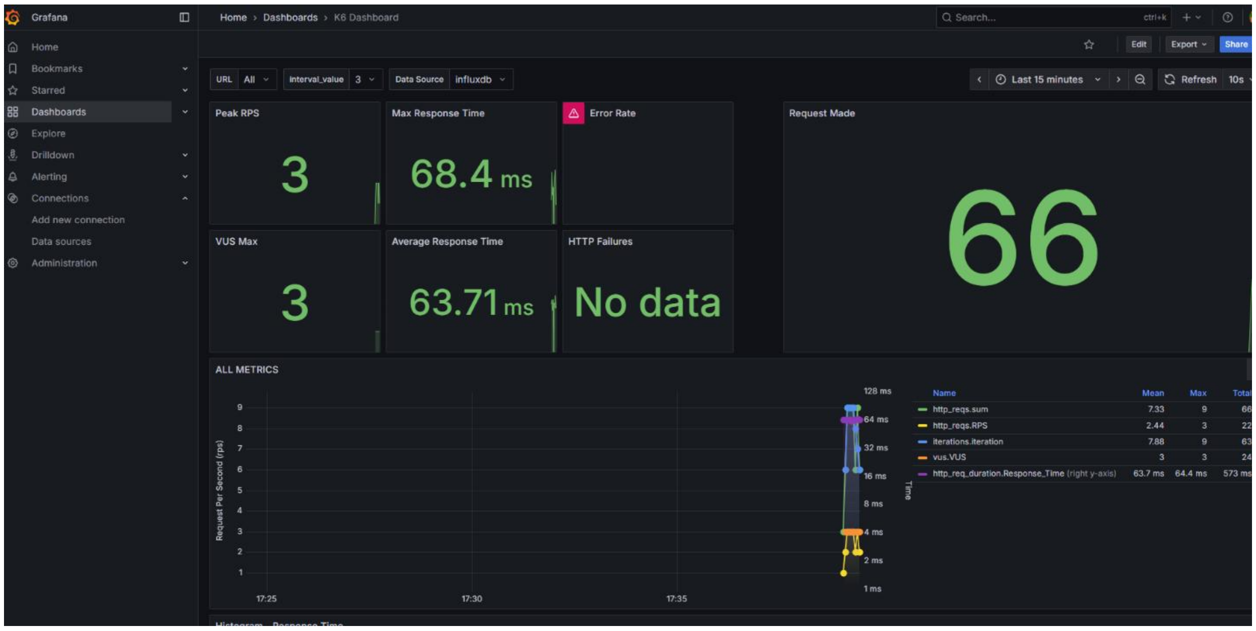
POST / user/createWithList

User-createWithList-test.js



```js
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 3,
  duration: '1m',
};

export default () => {
  const url = 'https://petstore.swagger.io/v2/user/createWithList';
  const payload = JSON.stringify([
    {
      id: Math.floor(Math.random() * 10000),
      username: `user${Math.floor(Math.random() * 1000)}`,
      firstName: 'Test',
      lastName: 'User',
      email: `test${Math.floor(Math.random() * 1000)}@example.com`,
      password: 'testPassword123',
      phone: '+1234567890',
      userStatus: 1
    }
  ]);
  const params = {
    headers: {
      'Content-Type': 'application/json',
    },
  };

  const res = http.post(url, payload, params);

  check(res, {
    'status is 200 or default': (r) => r.status === 200 || r.status === 201,
```

```
=1.31s   p(90)=1.06s    p(95)=1.06s
    iterations.............................................................: 170    2.795551/s
    vus....................................................................: 3      min=3        max=3
    vus_max................................................................: 3      min=3        max=3

    NETWORK
    data_received..........................................................: 66 kB  1.1 kB/s
    data_sent..............................................................: 44 kB  728 B/s


running (1m00.8s), 0/3 VUs, 170 complete and 0 interrupted iterations
default ✓ [======================================] 3 VUs  1m0s
PS C:\Users\DianaHernandez\Documents\Diana Hernandez QA\Training\Performance>
```



Home › Dashboards › K6 Dashboard

URL: All · interval_value: 3 · Data Source: influxdb · Last 15 minutes · Refresh 10s

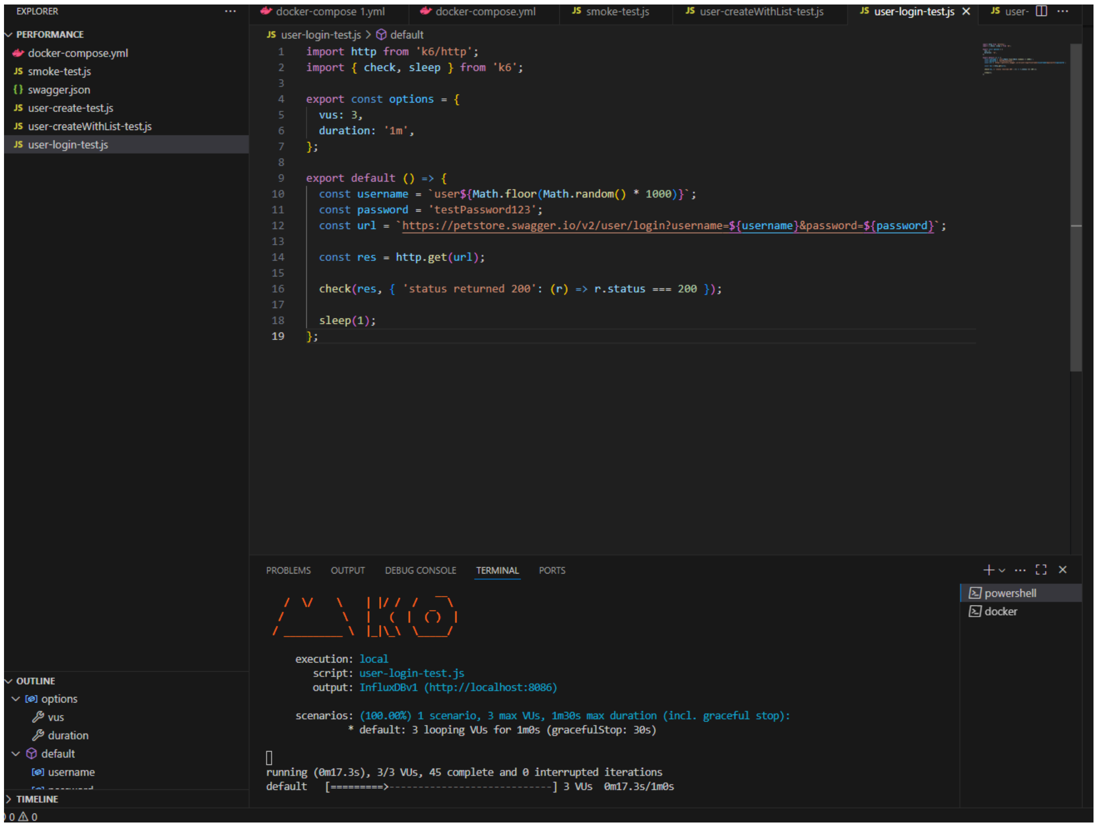
```
Peak RPS                3
Max Response Time       68.4 ms
Error Rate
Request Made            66
VUS Max                 3
Average Response Time   63.71 ms
HTTP Failures           No data
```

ALL METRICS

Name	Mean	Max	Total
http_reqs.sum	7.33	9	66
http_reqs.RPS	2.44	3	22
iterations.iteration	7.88	9	63
vus.VUS	3	3	24
http_req_duration.Response_Time (right y-axis)	63.7 ms	64.4 ms	573 ms

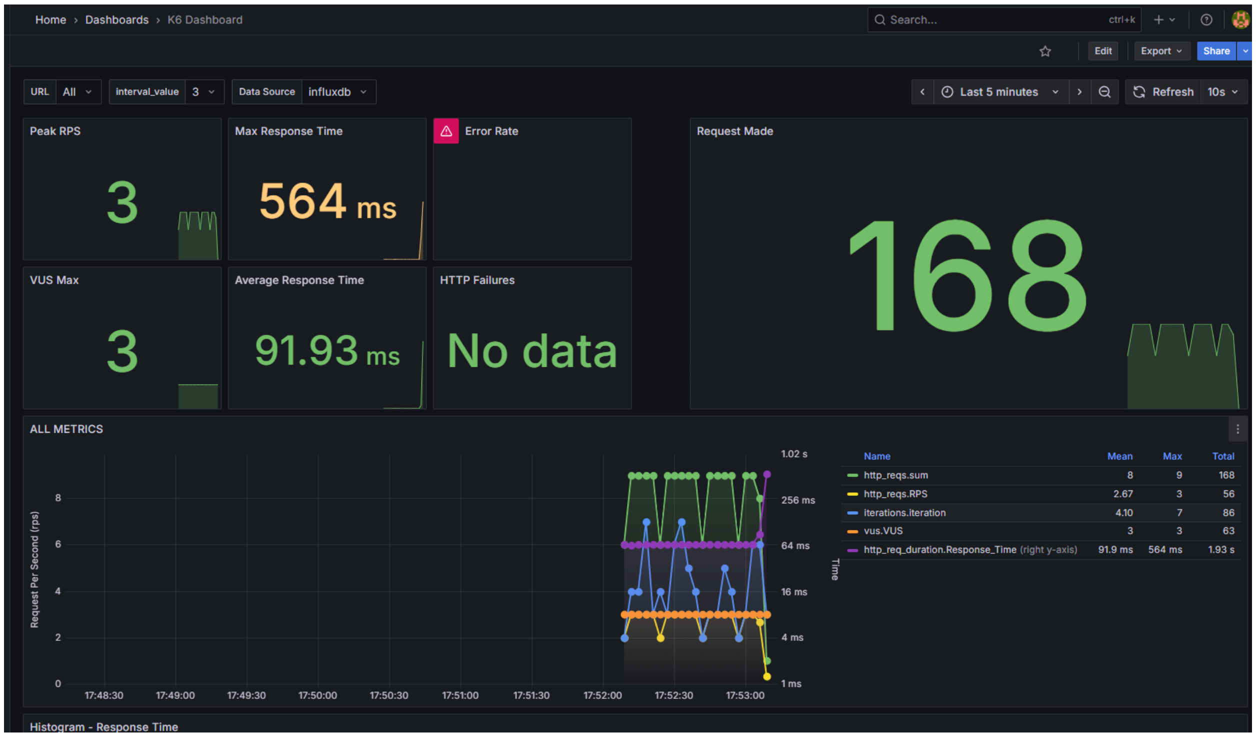
GET user/login

User-login-test.json



```javascript
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 3,
  duration: '1m',
};

export default () => {
  const username = `user${Math.floor(Math.random() * 1000)}`;
  const password = 'testPassword123';
  const url = `https://petstore.swagger.io/v2/user/login?username=${username}&password=${password}`;

  const res = http.get(url);

  check(res, { 'status returned 200': (r) => r.status === 200 });

  sleep(1);
};
```

```
     execution: local
        script: user-login-test.js
        output: InfluxDBv1 (http://localhost:8086)

     scenarios: (100.00%) 1 scenario, 3 max VUs, 1m30s max duration (incl. graceful stop):
              * default: 3 looping VUs for 1m0s (gracefulStop: 30s)


running (0m17.3s), 3/3 VUs, 45 complete and 0 interrupted iterations
default   [=========>--------------------------] 3 VUs  0m17.3s/1m0s
```



Home › Dashboards › K6 Dashboard

URL: All | interval_value: 3 | Data Source: influxdb | Last 5 minutes | Refresh 10s

Peak RPS: 3

Max Response Time: 564 ms

Error Rate

Request Made: 168

VUS Max: 3

Average Response Time: 91.93 ms

HTTP Failures: No data

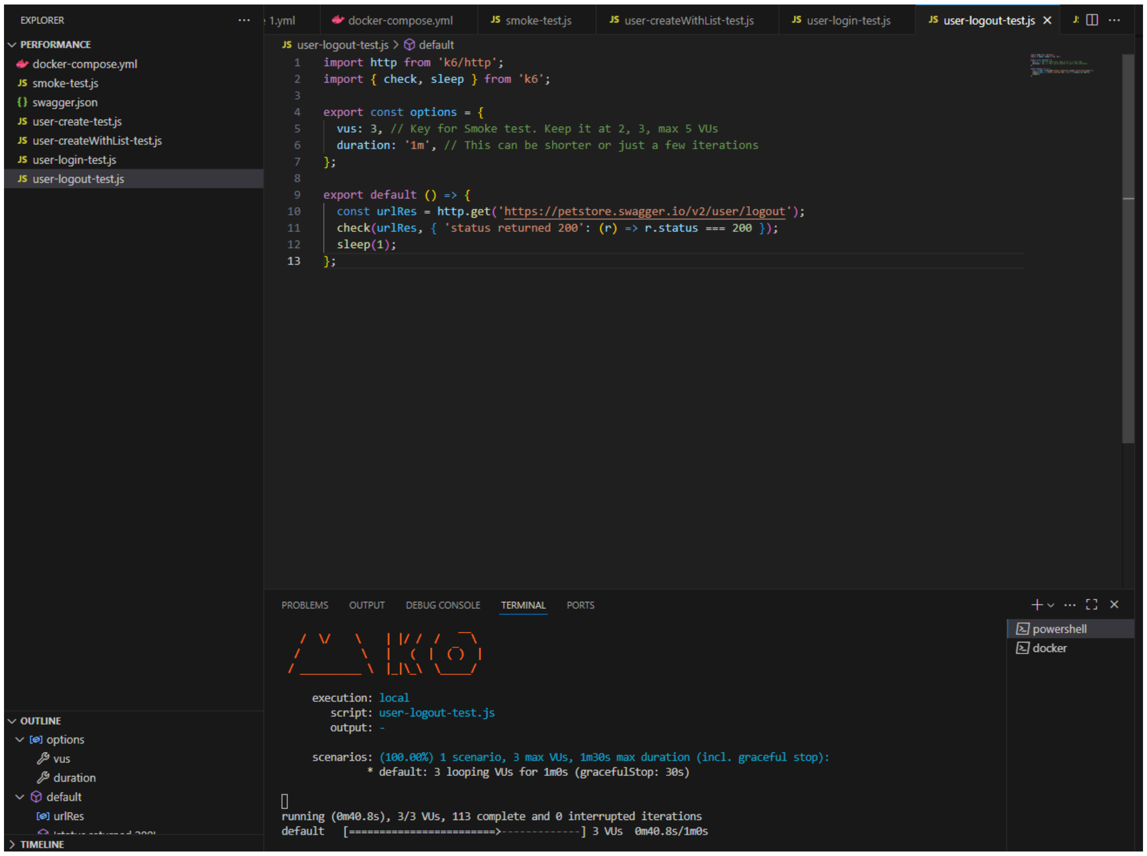
ALL METRICS

	Name	Mean	Max	Total
🟩	http_reqs.sum	8	9	168
🟨	http_reqs.RPS	2.67	3	56
🟦	iterations.iteration	4.10	7	86
🟧	vus.VUS	3	3	63
🟪	http_req_duration.Response_Time (right y-axis)	91.9 ms	564 ms	1.93 s

Histogram - Response Time

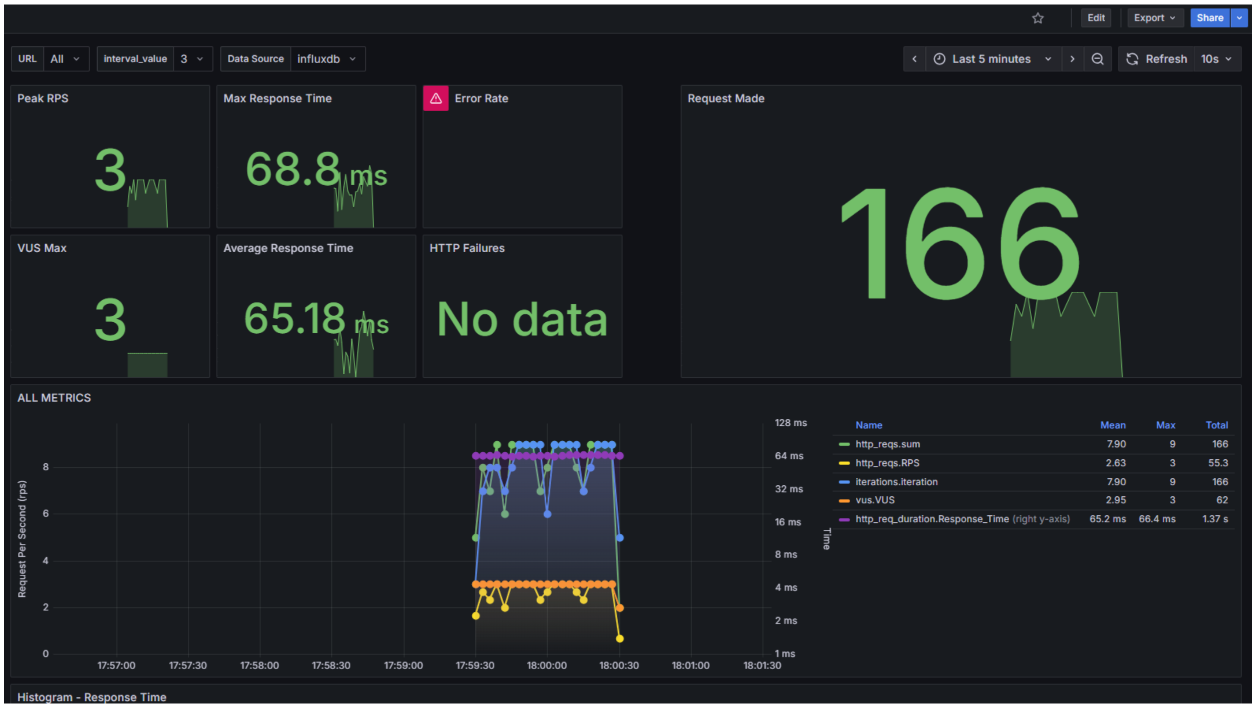
GET /user/logout

User-logout-test.js



```javascript
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 3, // Key for Smoke test. Keep it at 2, 3, max 5 VUs
  duration: '1m', // This can be shorter or just a few iterations
};

export default () => {
  const urlRes = http.get('https://petstore.swagger.io/v2/user/logout');
  check(urlRes, { 'status returned 200': (r) => r.status === 200 });
  sleep(1);
};
```

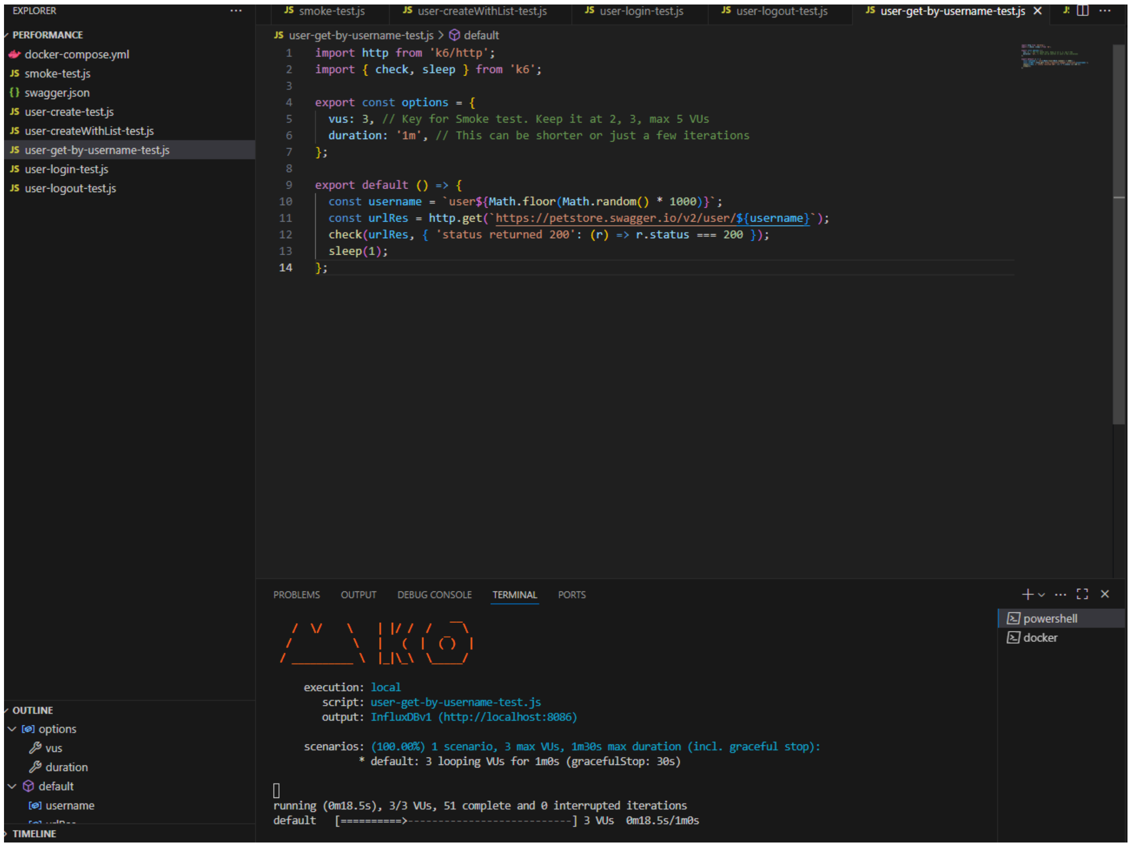
```
     execution: local
        script: user-logout-test.js
        output: -

     scenarios: (100.00%) 1 scenario, 3 max VUs, 1m30s max duration (incl. graceful stop):
              * default: 3 looping VUs for 1m0s (gracefulStop: 30s)


running (0m40.8s), 3/3 VUs, 113 complete and 0 interrupted iterations
default   [=======================>------------] 3 VUs  0m40.8s/1m0s
```



Name	Mean	Max	Total
http_reqs.sum	7.90	9	166
http_reqs.RPS	2.63	3	55.3
iterations.iteration	7.90	9	166
vus.VUS	2.95	3	62
http_req_duration.Response_Time (right y-axis)	65.2 ms	66.4 ms	1.37 s

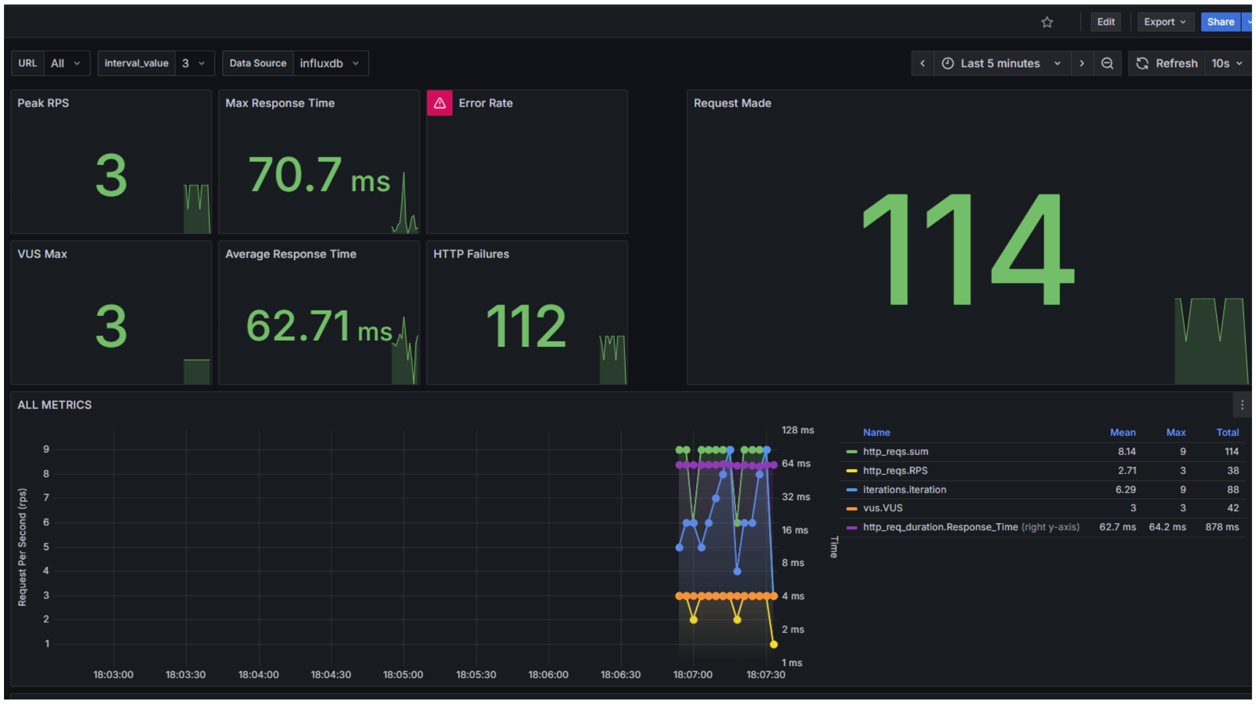
GET /user/{username}

user-get-by-username-test.js



```javascript
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 3, // Key for Smoke test. Keep it at 2, 3, max 5 VUs
  duration: '1m', // This can be shorter or just a few iterations
};

export default () => {
  const username = `user${Math.floor(Math.random() * 1000)}`;
  const urlRes = http.get(`https://petstore.swagger.io/v2/user/${username}`);
  check(urlRes, { 'status returned 200': (r) => r.status === 200 });
  sleep(1);
};
```

```
     execution: local
        script: user-get-by-username-test.js
        output: InfluxDBv1 (http://localhost:8086)

     scenarios: (100.00%) 1 scenario, 3 max VUs, 1m30s max duration (incl. graceful stop):
              * default: 3 looping VUs for 1m0s (gracefulStop: 30s)

running (0m18.5s), 3/3 VUs, 51 complete and 0 interrupted iterations
default   [==========>--------------------------] 3 VUs  0m18.5s/1m0s
```



URL: All · interval_value: 3 · Data Source: influxdb · Last 5 minutes · Refresh 10s

Peak RPS: 3

Max Response Time: 70.7 ms

Error Rate

Request Made: 114

VUS Max: 3

Average Response Time: 62.71 ms

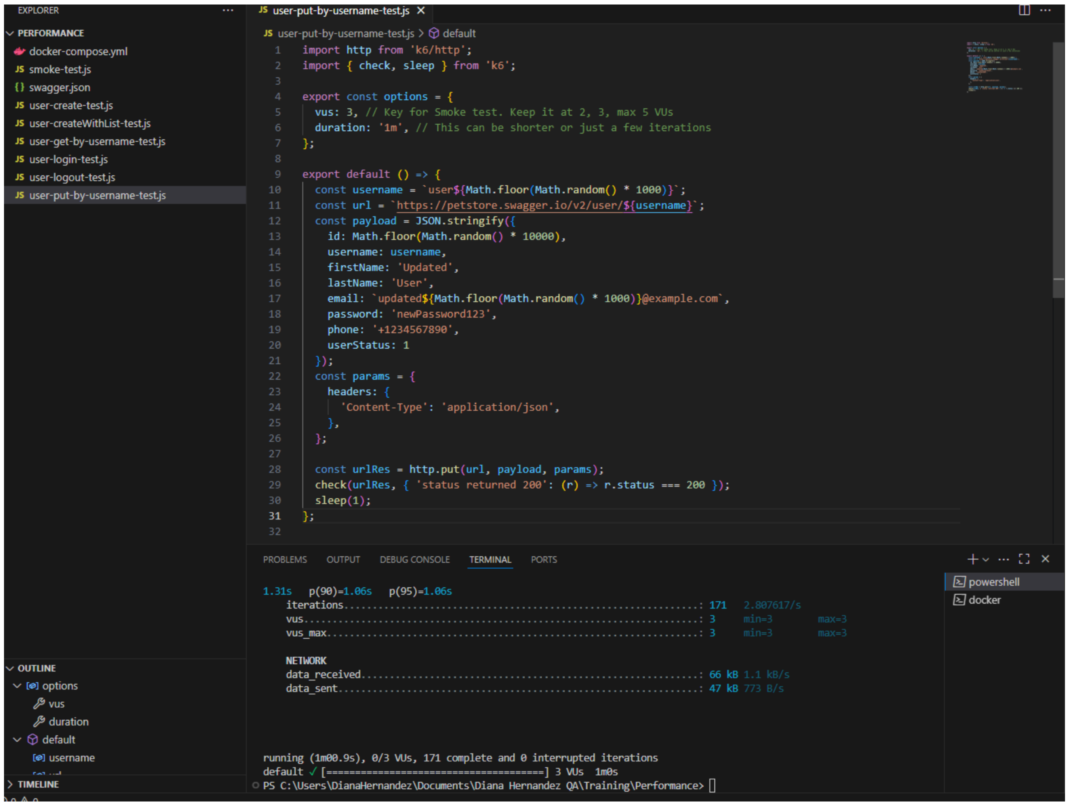
HTTP Failures: 112

ALL METRICS

Name	Mean	Max	Total
http_reqs.sum	8.14	9	114
http_reqs.RPS	2.71	3	38
iterations.iteration	6.29	9	88
vus.VUS	3	3	42
http_req_duration.Response_Time (right y-axis)	62.7 ms	64.2 ms	878 ms

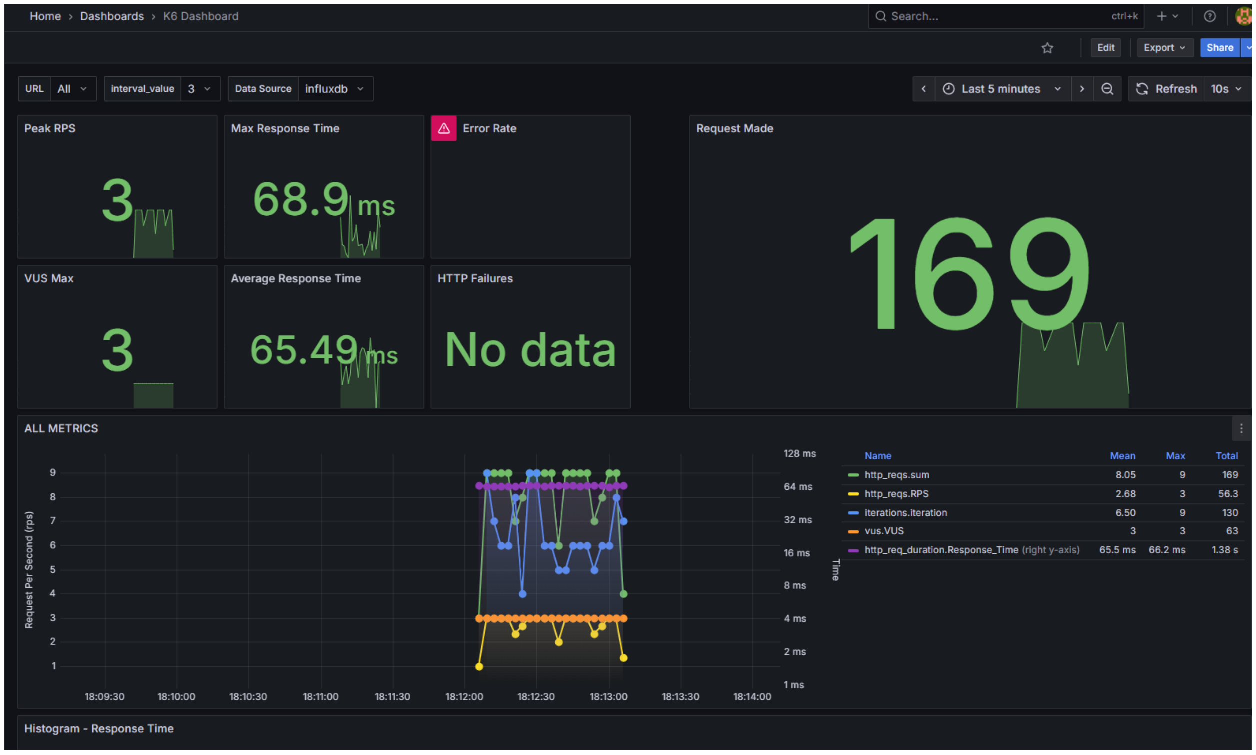
PUT /user/{username}

user-put-by-username-test.js



```javascript
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 3, // Key for Smoke test. Keep it at 2, 3, max 5 VUs
  duration: '1m', // This can be shorter or just a few iterations
};

export default () => {
  const username = `user${Math.floor(Math.random() * 1000)}`;
  const url = `https://petstore.swagger.io/v2/user/${username}`;
  const payload = JSON.stringify({
    id: Math.floor(Math.random() * 10000),
    username: username,
    firstName: 'Updated',
    lastName: 'User',
    email: `updated${Math.floor(Math.random() * 1000)}@example.com`,
    password: 'newPassword123',
    phone: '+1234567890',
    userStatus: 1
  });
  const params = {
    headers: {
      'Content-Type': 'application/json',
    },
  };

  const urlRes = http.put(url, payload, params);
  check(urlRes, { 'status returned 200': (r) => r.status === 200 });
  sleep(1);
};
```

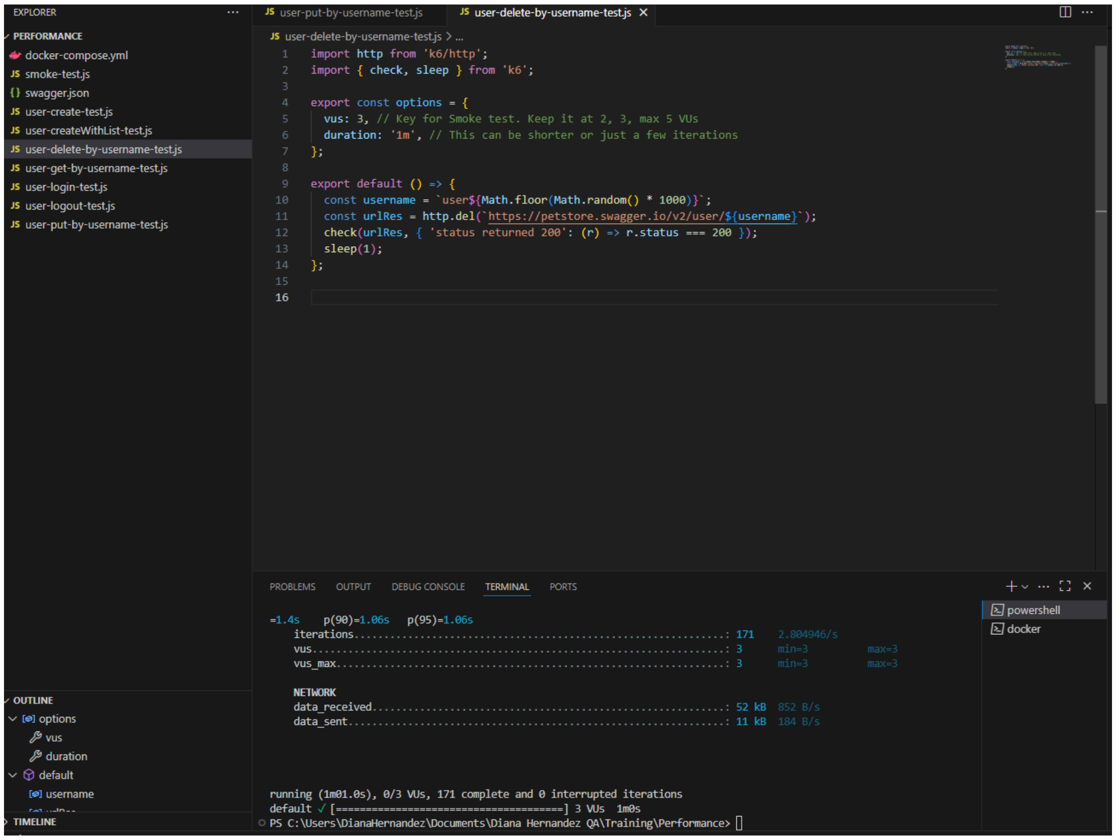
```
1.31s   p(90)=1.06s   p(95)=1.06s
   iterations.........................................................: 171   2.807617/s
   vus................................................................: 3     min=3        max=3
   vus_max............................................................: 3     min=3        max=3

   NETWORK
   data_received......................................................: 66 kB 1.1 kB/s
   data_sent..........................................................: 47 kB 773 B/s




running (1m00.9s), 0/3 VUs, 171 complete and 0 interrupted iterations
default ✓ [======================================] 3 VUs  1m0s
PS C:\Users\DianaHernandez\Documents\Diana Hernandez QA\Training\Performance> 
```



	Name	Mean	Max	Total
🟩	http_reqs.sum	8.05	9	169
🟨	http_reqs.RPS	2.68	3	56.3
🟦	iterations.iteration	6.50	9	130
🟧	vus.VUS	3	3	63
🟪	http_req_duration.Response_Time (right y-axis)	65.5 ms	66.2 ms	1.38 s

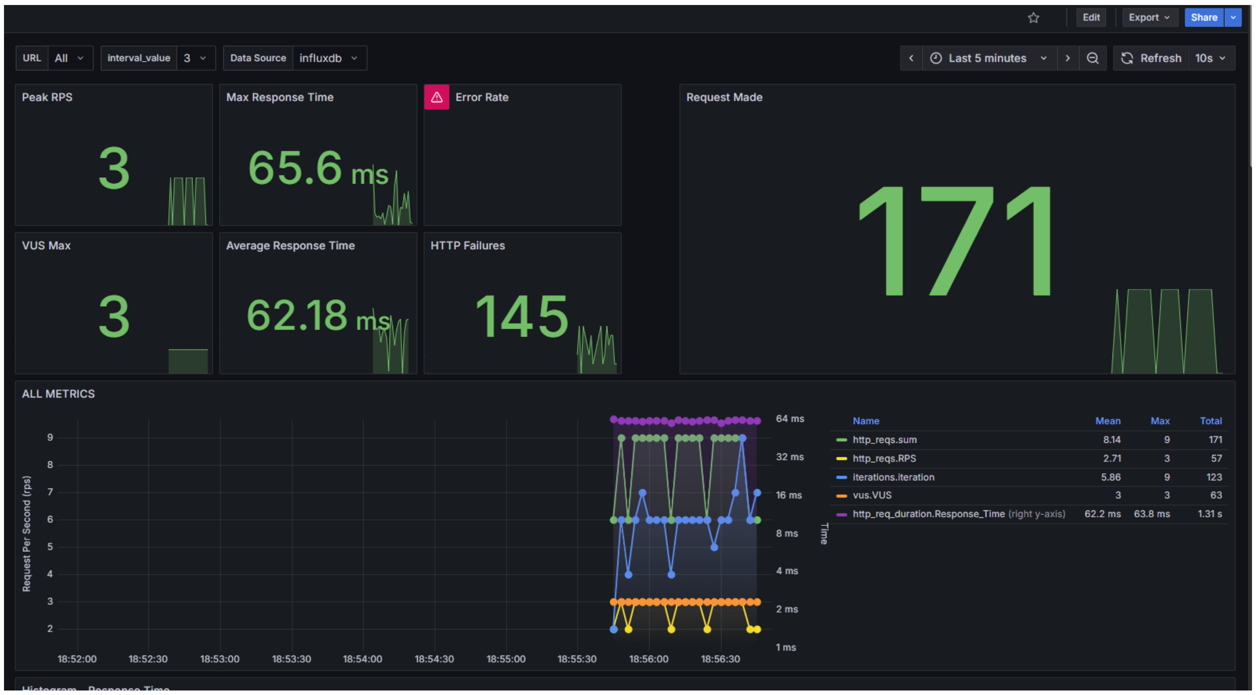
DELETE /user/{username}

User-delete-by-username-test.js



```javascript
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 3, // Key for Smoke test. Keep it at 2, 3, max 5 VUs
  duration: '1m', // This can be shorter or just a few iterations
};

export default () => {
  const username = `user${Math.floor(Math.random() * 1000)}`;
  const urlRes = http.del(`https://petstore.swagger.io/v2/user/${username}`);
  check(urlRes, { 'status returned 200': (r) => r.status === 200 });
  sleep(1);
};
```

```
=1.4s    p(90)=1.06s   p(95)=1.06s
   iterations.............................................................: 171    2.804946/s
   vus....................................................................: 3      min=3        max=3
   vus_max................................................................: 3      min=3        max=3

   NETWORK
   data_received..........................................................: 52 kB  852 B/s
   data_sent..............................................................: 11 kB  184 B/s



running (1m01.0s), 0/3 VUs, 171 complete and 0 interrupted iterations
default ✓ [======================================] 3 VUs  1m0s
PS C:\Users\DianaHernandez\Documents\Diana Hernandez QA\Training\Performance>
```



URL All | interval_value 3 | Data Source influxdb

Peak RPS 3

Max Response Time 65.6 ms

Error Rate

Request Made 171

VUS Max 3

Average Response Time 62.18 ms

HTTP Failures 145

ALL METRICS

	Name	Mean	Max	Total
🟩	http_reqs.sum	8.14	9	171
🟨	http_reqs.RPS	2.71	3	57
🟦	iterations.iteration	5.86	9	123
🟧	vus.VUS	3	3	63
🟪	http_req_duration.Response_Time (right y-axis)	62.2 ms	63.8 ms	1.31 s