

Automation & Performance Testing Instructions

This document includes all instructions required to run the Cypress automated functional tests and the k6 performance test for the Neighborly QA Sandbox assessment. It contains folder structure, commands, installation steps, and execution procedures.

1. Cypress Automated Test Instructions

The Cypress suite validates:

- Program Staff user can create a program.
- The created program appears in the Programs List.
- The Program Detail page displays all expected information.

1.1 File Structure

```
cypress/
  e2e/
    programManagement.cy.js
  support/
    commands.js
    e2e.js
cypress.config.js
```

1.2 Cypress Custom Commands

- cy.signInAsRole(role)
- cy.fillFieldByLabel(label, value)
- cy.fillDateByLabel(label, mm/dd/yyyy)

1.3 Running the Cypress Tests

1. Install dependencies:

```
npm install
```

2. Open Cypress (interactive mode):

```
npx cypress open
```

3. Run Cypress headless:

```
npx cypress run --spec "cypress/e2e/programManagement.cy.js"
```

2. k6 Performance Testing Instructions

The k6 script measures response time and performance of the `/programs` endpoint. It validates response correctness, HTML loading, and timing thresholds.

2.1 File Location

k6/programs-list-test.js

2.2 k6 Script

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

export const options = {
  stages: [
    { duration: '5s', target: 20 },
    { duration: '20s', target: 50 },
    { duration: '5s', target: 0 },
  ],
  thresholds: {
    http_req_duration: ['p(95)<800'],
    http_req_failed: ['rate<0.01'],
  },
};

const BASE_URL = 'https://neighborly-qa-sandbox-test.lovable.app';

const SESSION_COOKIE_NAME = 'session-id';
const SESSION_COOKIE_VALUE = '<REPLACE_WITH_FRESH_SESSION>';

// my session id was 0a2b8a9a-522f-45a8-b5d6-5af52b34795c

const params = {
  cookies: {
    [SESSION_COOKIE_NAME]: SESSION_COOKIE_VALUE,
  },
  headers: {
    'User-Agent': 'k6-neighborly-perf-test',
    'Accept': 'text/html,application/xhtml+xml',
  },
};

export default function () {
  const res = http.get(`#${BASE_URL}/programs` , params);

  check(res, {
    'status is 200': (r) => r.status === 200,
    'landed on /programs': (r) => r.url.includes('/programs'),
    'HTML includes root div': (r) =>
      r.body && String(r.body).includes('<div id="root"></div>'),
  });
}
```

```
    'response time < 800ms': (r) => r.timings.duration < 800,  
  });  
  
  sleep(1);  
}
```

2.3 Running the k6 Test

- 1. Install k6 locally, or upload the script to Grafana Cloud k6.
- 2. Replace `<REPLACE_WITH_FRESH_SESSION>` with an active `session-id` cookie from browser DevTools. (I used my own session ex. 0a2b8a9a-522f-45a8-b5d6-5af52b34795c)
- 3. Run locally:

```
k6 run k6/programs-list-test.js
```