

Performance Testing Setup Guide

Quick Setup and Test Execution

This guide provides step-by-step instructions to run performance tests on the Events List endpoint and collect results for academic submission.

Prerequisites

1. **Docker Desktop** (Windows/Mac) or Docker Engine (Linux)
2. **k6 Load Testing Tool**: <https://k6.io/docs/getting-started/installation/>
 - Windows: `choco install k6` or `winget install k6`
 - Mac: `brew install k6`
 - Linux: See k6 website for distribution-specific instructions

Step 1: Start the Application Stack

Navigate to the Project directory and start all services:

Linux/Mac:

```
cd Project  
./monitoring-manager.sh start
```

Windows PowerShell:

```
cd Project  
.\\monitoring-manager.ps1 start
```

Windows Command Prompt:

```
cd Project  
monitoring-manager.bat start
```

This starts:

- Backend application (port 8080)
- PostgreSQL database (port 5433)
- Prometheus metrics (port 9090)
- Grafana dashboard (port 3001)
- Container and system monitoring

Step 2: Verify Services Are Running

Check that all services are healthy:

Linux/Mac:

```
./monitoring-manager.sh health
```

Windows PowerShell:

```
.\monitoring-manager.ps1 health
```

Windows Command Prompt:

```
monitoring-manager.bat health
```

You should see:

- Backend (8080): ✓ Healthy
- Frontend (3000): ✓ Healthy
- Prometheus (9090): ✓ Healthy
- Grafana (3001): ✓ Healthy

Step 3: Run the Events List Performance Test

Navigate to the load-tests directory:

```
cd load-tests
```

Run the test using k6:

```
k6 run scripts/events-list-test.js --out json=results/events-list.json --summary-export=results/events-list_summary.json
```

The test will:

- Ramp up to 50 RPS over 30 seconds
- Ramp up to 100 RPS over another 30 seconds
- Sustain 100 RPS for 5 minutes
- Ramp down over 30 seconds

Total test duration: approximately 6.5 minutes

Step 4: Collect Metrics and Results

After the test completes, you'll find:

1. Terminal Output: Shows real-time metrics including:

- Total requests
- Success/error rates
- Response time percentiles (P50, P90, P95, P99)
- Throughput (RPS)

2. JSON Files (in `results/` directory):

- `events-list.json`: Raw metrics data
- `events-list_summary.json`: Summary statistics

3. Grafana Dashboard (optional):

- Open `http://localhost:3001` (admin/admin123)
- View real-time performance metrics during the test

Step 5: Access the Completed Report

The performance report with analyzed results is available at:

- `load-tests/EVENTS_LIST_PERFORMANCE_REPORT.md`

This report includes:

- Test configuration and methodology
- Complete results with actual measurements
- Performance analysis
- SLO compliance evaluation
- Recommendations

Troubleshooting

Services won't start:

```
docker-compose down -v  
docker-compose up -d
```

Backend not healthy:

```
# Check logs  
docker-compose logs backend  
  
# Restart backend  
docker-compose restart backend
```

k6 not found:

- Verify k6 is installed: `k6 version`
- Install from <https://k6.io/docs/getting-started/installation/>

Test fails with authorization errors:

- Ensure the backend is fully started and healthy
- Check that the default admin user exists (admin@aiu.edu / admin123)
- Review backend logs for authentication issues

Viewing Monitoring Dashboards

Grafana (Metrics Visualization):

- URL: `http://localhost:3001`
- Username: admin
- Password: admin123
- Pre-configured dashboard: "AIU Trips & Events - Performance Dashboard"

Prometheus (Raw Metrics):

- URL: `http://localhost:9090`
- Query examples:
 - P95 latency: `histogram_quantile(0.95, sum(rate(http_server_requests_seconds_bucket[1m])) by (1e))`
 - Request rate: `sum(rate(http_server_requests_seconds_count[1m]))`

Stopping the Stack

When finished:

Linux/Mac:

```
cd Project  
./monitoring-manager.sh stop
```

Windows PowerShell:

```
cd Project  
.\\monitoring-manager.ps1 stop
```

Windows Command Prompt:

```
monitoring-manager.bat stop
```

Files for Academic Submission

The following files should be submitted:

1. **EVENTS_LIST_PERFORMANCE_REPORT.md** - Complete performance analysis report
 2. **results/events-list_summary.json** - Test results summary
 3. Screenshots from Grafana (optional but recommended)
-

Test Configuration:

- Endpoint: GET /api/events
- Target: P95 < 200ms @ 100 RPS
- Tool: k6
- Duration: ~6.5 minutes
- Monitoring: Prometheus + Grafana