

STEP 1: Define Workflow Inputs & Triggers

```
name: Playwright Tests
```

```
on:
```

```
  pull_request:
```

```
  branches:
```

```
    - develop
```

```
  schedule:
```

```
    - cron: '0 0 * * *'
```

```
  workflow_dispatch:
```

```
  inputs:
```

```
  shards:
```

```
    description: 'Number of shards'
```

```
    required: true
```

```
    default: '3'
```

```
  workers:
```

```
    description: 'Number of workers'
```

```
    required: false
```

```
  tags:
```

```
    description: 'Tags to filter tests'
```

```
    required: false
```

```
  retries:
```

```
    description: 'Number of retries'
```

```
    required: false
```

```
  environment:
```

```
    description: 'Deployment environment'
```

```
    type: choice
```

```
    default: 'local'
```

```
  options:
```

```
    - local
```

```
    - dev
```

```
    - staging
```

```
    - uat
```

STEP 2: Create a Composite Action for Setting Environment Variables

```
name: 'Set Environment Variables'
description: 'Set environment variables based on inputs'

inputs:
  environment:
    description: 'The environment to set'
    required: true
  workers:
    description: 'Number of workers'
    required: false
  retries:
    description: 'Number of retries'
    required: false

runs:
  using: 'composite'
  steps:
    - name: Set BASE_URL
      shell: bash
      run: |
        if [ "${{ inputs.environment }}" == "dev" ]; then
          echo "BASE_URL=http://dev.example.com" >> $GITHUB_ENV
          echo "API_BASE_URL=https://api.dev.example.com" >> $GITHUB_ENV
        elif [ "${{ inputs.environment }}" == "staging" ]; then
          echo "BASE_URL=https://staging.example.com" >> $GITHUB_ENV
          echo "API_BASE_URL=https://api.staging.example.com" >>
$GITHUB_ENV
        elif [ "${{ inputs.environment }}" == "uat" ]; then
          echo "BASE_URL=https://uat.example.com" >> $GITHUB_ENV
          echo "API_BASE_URL=https://api.uat.example.com" >> $GITHUB_ENV
        else
          echo "BASE_URL=http://localhost:3000" >> $GITHUB_ENV
          echo "API_BASE_URL=https://api.local.example.com" >> $GITHUB_ENV
        fi

    - name: Set WORKERS
      shell: bash
      if: "${{ inputs.workers != '' }}"
      run: echo "WORKERS=${{ inputs.workers }}" >> $GITHUB_ENV

    - name: Set Retries
      shell: bash
      if: "${{ inputs.retries != '' }}"
      run: echo "RETRIES=${{ inputs.retries }}" >> $GITHUB_ENV
```

STEP 3: Define the Setup Job

```
jobs:
  setup:
    runs-on: ubuntu-latest
    outputs:
      matrix: ${ steps.create-matrix.outputs.matrix }
    steps:
      - name: Checkout code
        uses: actions/checkout@v4

      - name: Create dynamic matrix
        id: create-matrix
        run: |
          echo "Creating matrix..."
          shards=${ github.event.inputs.shards }
          if [ -z "$shards" ]; then
            shards=3
          fi
          shardArray=$(seq -s, 1 $shards | sed 's/,/"/g')
          matrix=$(echo "{\"shard\": \"$shardArray\",
            \"total_shards\": \"$shards\"}")
          echo "matrix=$matrix" >> $GITHUB_OUTPUT

      - name: Set Environment Variables
        uses: ./github/actions/set-environment-variables
        with:
          environment: ${ github.event.inputs.environment }
          workers: ${ github.event.inputs.workers }
          retries: ${ github.event.inputs.retries }
```

STEP 4: Define the Test Job

```
jobs:
  test:
    runs-on: ubuntu-latest
    needs: setup
    strategy:
      fail-fast: false
    matrix: ${{ fromJSON(needs.setup.outputs.matrix) }}
    steps:
      - name: Checkout code
        uses: actions/checkout@v4

      - name: Set Environment Variables
        uses: ./github/actions/set-environment-variables
        with:
          environment: ${{ github.event.inputs.environment }}
          workers: ${{ github.event.inputs.workers }}
          retries: ${{ github.event.inputs.retries }}

      - name: Run Playwright tests
        if: ${{ github.event.inputs.tags == '' }}
        run: npx playwright test --shard=${{ matrix.shard }}/${{ matrix.total_shards }} --project=chrome
        env:
          BASE_URL: ${{ env.BASE_URL }}
          WORKERS: ${{ env.WORKERS }}
          RETRIES: ${{ env.RETRIES }}

      - name: Run Playwright tests with tags
        if: ${{ github.event.inputs.tags != '' }}
        run: npx playwright test --shard=${{ matrix.shard }}/${{ matrix.total_shards }} --project=chrome --grep=${{ github.event.inputs.tags }}
        env:
          BASE_URL: ${{ env.BASE_URL }}
          WORKERS: ${{ env.WORKERS }}
          RETRIES: ${{ env.RETRIES }}
```

STEP 5: Configure Playwright

```
export const baseUrl = process.env['BASE_URL'];

const config: PlaywrightTestConfig = {
  ...(process.env.RETRIES ? { retries:
parseInt(process.env.RETRIES) } : {}),
  ...(process.env.WORKERS ? { workers:
process.env.WORKERS } : {}),
  use: {
    baseUrl,
  },
  // Other configurations
};

export default config;
```