GitHub Actions Workflow Maintenance Guide

Overview

This document provides comprehensive guidance for maintaining and monitoring the Audityzer GitHub Actions workflows.

Current Status

Workflow files restored: All 24 workflow files have been successfully restored and updated
Action versions updated: All workflows now use the latest action versions (v4)
Error handling improved: Added fallback mechanisms for missing scripts/files
Package.json fixed: Resolved merge conflicts and dependency issues

Workflow Health Monitoring

Quick Health Check

Use our monitoring script to get a quick overview:

```
# Generate workflow health report
node scripts/monitor-workflows.js

# View the generated report
cat reports/workflow-health.md
```

Manual Monitoring

1. **GitHub Actions Page**: https://github.com/romanchaa997/Audityzer/actions
2. **API Endpoint**: `https://api.github.com/repos/romanchaa997/Audityzer/actions/runs`

Success Rate Targets

- **Critical workflows**: >95% success rate
- **Standard workflows**: >80% success rate
- **Experimental workflows**: >50% success rate

Workflow Categories

1. Core CI/CD Workflows

- `ci-cd.yml` - Main CI/CD pipeline
- `main.yml` - Primary CI workflow
- `test-and-build.yml` - Testing and building

2. Security Workflows

- `security-audit.yml` - Dependency security scanning
- `security-scan.yml` - Security checklist scanning
- `codeql.yml` - Code quality and security analysis

- `audityzer-security.yml` - Custom security analysis

3. Testing Workflows

- `playwright.yml` - End-to-end testing
- `cross-platform-tests.yml` - Multi-platform testing
- `bridge-security-tests.yml` - Bridge security testing
- `protocol-testing.yml` - Protocol testing
- `performance.yml` - Performance benchmarking

4. Automation Workflows

- `auto-add.yml` - Auto-add issues to project
- `auto-archive.yml` - Auto-archive closed items
- `auto-label.yml` - Auto-labeling
- `labeler.yml` - PR labeling

5. Deployment Workflows

- `fly-deploy.yml` - Fly.io deployment
- `website-regression.yml` - Website regression testing

6. Social & Communication

- `social-push.yml` - Social media posting
- `tweet-release.yml` - Release announcements

7. Code Quality

- `lint.yml` - Linting and formatting
- `codecov.yml` - Code coverage

Common Issues and Solutions

1. Missing Scripts Error

Problem: Workflow fails with “script not found”

Solution:

```yaml
- name: Run script (if exists)
  run: |
    if [ -f "scripts/script-name.js" ]; then
      node scripts/script-name.js
    else
      echo "Script not found, skipping"
    fi
```

2. Dependency Installation Failures

Problem: `npm ci` fails due to package conflicts

Solution:

```
- name: Install Dependencies
  run: |
    npm ci --legacy-peer-deps || npm install --legacy-peer-deps
```

3. Action Version Deprecation

Problem: Workflows fail due to deprecated action versions

Solution: Update to latest versions:
- `actions/checkout@v4`
- `actions/setup-node@v4`
- `actions/upload-artifact@v4`
- `actions/download-artifact@v4`

4. Secret Missing Errors

Problem: Workflow fails due to missing secrets

Solution: Add conditional checks:

```
- name: Check for secret
  id: check_secret
  run: |
    if [ -n "${{ secrets.SECRET_NAME }}" ]; then
      echo "secret_exists=true" >> $GITHUB_OUTPUT
    else
      echo "secret_exists=false" >> $GITHUB_OUTPUT
    fi

- name: Use secret
  if: steps.check_secret.outputs.secret_exists == 'true'
  run: echo "Using secret"
```

Required Secrets

Repository Secrets

- `GITHUB_TOKEN` - Automatically provided
- `NPM_TOKEN` - For npm publishing (optional)
- `PROJECT_TOKEN` - For project automation (optional)

Optional Secrets

- `SNYK_TOKEN` - For Snyk security scanning
- `CODECOV_TOKEN` - For code coverage reporting
- `TWITTER_*` - For social media automation

Workflow Triggers

Push Triggers

- `main`, `master`, `develop` branches
- Specific file paths for targeted workflows

Pull Request Triggers

- `main`, `master` branches for most workflows
- All branches for testing workflows

Scheduled Triggers

- Security audits: Weekly (Sundays)
- Performance tests: Weekly (Mondays)
- Bridge security: Daily

Manual Triggers

- All workflows support `workflow_dispatch` for manual execution

Performance Optimization

1. Caching Strategy

```yaml
- name: Cache dependencies
  uses: actions/cache@v3
  with:
    path: ~/.npm
    key: ${{ runner.os }}-node-${{ hashFiles('**/package-lock.json') }}
```

2. Parallel Jobs

- Use job dependencies (`needs:`) strategically
- Run independent tests in parallel
- Separate build and test phases

3. Conditional Execution

```yaml
- name: Run only on changes
  if: contains(github.event.head_commit.message, '[run-tests]')
```

Troubleshooting Guide

Step 1: Identify the Issue

1. Check the Actions tab for failed workflows
2. Click on the failed run to see details
3. Examine the error logs

Step 2: Common Fixes

1. **Syntax errors**: Validate YAML syntax
2. **Missing files**: Add conditional checks
3. **Permission errors**: Check repository settings
4. **Secret errors**: Verify secret configuration

Step 3: Testing Fixes

1. Create a feature branch

2. Make the necessary changes
3. Push and observe the workflow results
4. Merge when successful

Monitoring Setup

1. Automated Monitoring

Add this to your workflow:

```
- name: Monitor Workflow Health
  run: node scripts/monitor-workflows.js
  continue-on-error: true
```

2. Alerts

Set up notifications for:
- Consecutive failures (>3)
- Success rate drops below 80%
- Critical workflow failures

3. Regular Reviews

- Weekly review of workflow performance
- Monthly optimization of slow workflows
- Quarterly update of action versions

Best Practices

1. Workflow Design

- Keep workflows focused and single-purpose
- Use descriptive names and comments
- Implement proper error handling

2. Security

- Never expose secrets in logs
- Use least-privilege permissions
- Regularly audit workflow permissions

3. Maintenance

- Update action versions quarterly
- Review and optimize slow workflows
- Remove unused workflows

4. Documentation

- Document workflow purpose and dependencies
- Maintain this guide with changes
- Include troubleshooting steps

Emergency Procedures

Workflow Outage

1. Check GitHub Status page
2. Disable problematic workflows temporarily
3. Implement hotfixes on critical paths
4. Communicate status to team

Security Incident

1. Immediately disable affected workflows
2. Rotate any exposed secrets
3. Review workflow logs for compromise
4. Implement additional security measures

Contact and Support

- **Repository**: https://github.com/romanchaa997/Audityzer
- **Actions Page**: https://github.com/romanchaa997/Audityzer/actions
- **Issues**: https://github.com/romanchaa997/Audityzer/issues

Changelog

2025-06-14

- Restored all 24 workflow files
- Fixed package.json merge conflicts
- Updated action versions to v4
- Added error handling and fallbacks
- Created monitoring and documentation

Last updated: June 14, 2025