

# CI/CD pipelines for UiPath RPA projects

Semantic Versioning

Semantic Versioning 2.0.0 | Semver.org

https://semver.org/spec/v2.0.0.html

العربية (ar) Български (bg) català (ca) čeština (cs) Dansk (da) Deutsch (de) Ελληνικά (el)  
English (en) español (es) فارسی (fa) français (fr) עברית (he) हिन्दी (hin) hrvatski (hr)  
magyar (hu) Հայերեն (hy) Bahasa Indonesia (id) italiano (it) 日本語 (ja) ქართული (ka)  
taqbaylit (kab) 한국어 (ko) Nederlands (nl) polski (pl) português brasileiro (pt-BR) Română (ro)  
русский (ru) slovensky (sk) slovenščina (sl) srpski (sr) svenska (sv) Türkçe (tr)  
українська (uk) Tiếng Việt (vi) 简体中文 (zh-CN) 繁體中文 (zh-TW)

2.0.0 2.0.0-rc.2 2.0.0-rc.1 1.0.0 1.0.0-beta

# Semantic Versioning 2.0.0

## Summary

Given a version number MAJOR.MINOR.PATCH, increment the:

1. MAJOR version when you make incompatible API changes
2. MINOR version when you add functionality in a backward compatible manner
3. PATCH version when you make backward compatible bug fixes

Additional labels for pre-release and build metadata are available as extensions to the MAJOR.MINOR.PATCH format.

## Semantic Versioning Specification (SemVer)

4. Major version zero (0.y.z) is for initial development. Anything MAY change at any time. The public API SHOULD NOT be considered stable.
11. Precedence refers to how versions are compared to each other when ordered.

Example: 1.0.0-alpha < 1.0.0-alpha.1 < 1.0.0-alpha.beta < 1.0.0-beta < 1.0.0-beta.2 < 1.0.0-beta.11 < 1.0.0-rc.1 < 1.0.0.

## Backus–Naur Form Grammar for Valid SemVer Versions

```
<valid semver> ::= <version core>
  | <version core> "-" <pre-release>
  | <version core> "+" <build>
```

HOME DESIGN DEBUG

Cut Undo Copy Paste New Save Export As File Manage Packages Manage Entities Test Manager Recording Screen Scraping Scraping User Events UI Explorer Remove Unused Analyze File Export to Excel Publish

# FrozenChlorine - UiPath Studio Community

# UiPath Studio implementation

Project StandardCalculator StandardCalculator StandardCalculator Sequence

Description:  
This UiPath workflow is designed to perform basic arithmetic operations using a virtual calculator interface. It primarily handles addition but is structured to be extended to other operations. The workflow takes two terms and an operation type as input, processes them using the calculator, and returns the result.

**Publish Process**

Package properties  
Publish options  
Certificate signing

Package Name \* FrozenChlorine

Version  
Current Version 0.0.1  
New Version \* 0.0.2-alpha  
 Is Prerelease

Package Icon Optional Package Icon

Project tags Start typing to get a list of possible matches

Release Notes

Cancel Back Next Publish

Import Arguments Open Workflow

Test Explorer

Search files (Ctrl+Alt+T)

CanaryUserInterface

TestCase\_Canary

StandardCalculator

TestCase\_Standard (5) 00:21

(1) ID: "1", Operation: "Addition", T... 00:21

(2) ID: "2", Operation: "Addition", T... 00:21

(3) ID: "3", Operation: "Addition", T... 00:21

(4) ID: "4", Operation: "Addition", T... 00:21

(5) ID: "5", Operation: "Addition", T... 00:21

AdditionOf2Terms 100%

ClickListOfCharacters 75%

CloseAllApplications 100%

InitAllApplications 100%

InitAllSettings 65%

Initialization 90%

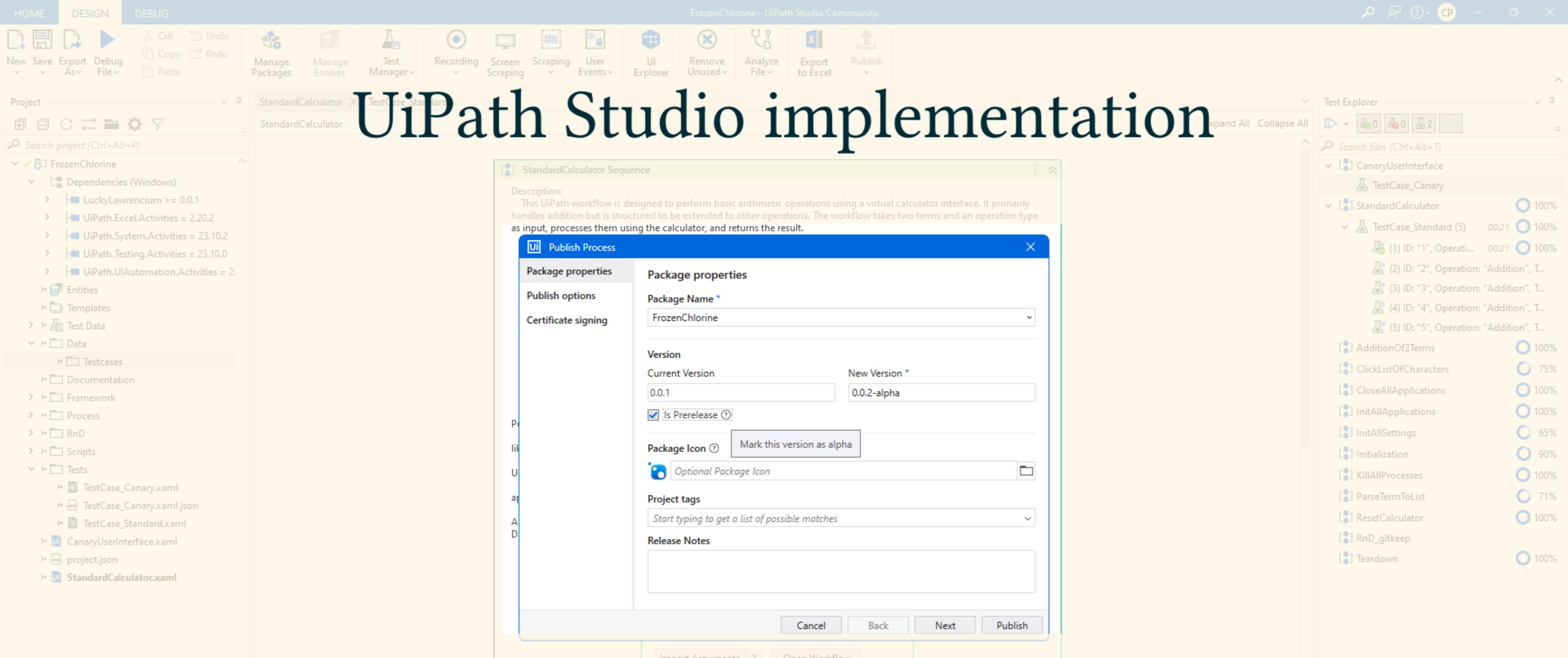
KillAllProcesses 100%

ParseTermToList 71%

ResetCalculator 100%

RnD\_gitkeep 100%

Teardown 100%



projectVersion ::= <version core>  
| <version core> "-" <pre-release>

Open



Open a Local Project  
Navigate and open an existing project

Open



Clone a Local Project  
Create a copy of an existing project for reuse like UI, Variables, etc.

Close

Start

Tools

Templates

Team

Settings

Help

# Templates are not Gospel!

Open Recent



FrozenChlorine  
VB Windows GIT

RPA UiPath process automating the Windows Calculator App

## New Project



Process  
Start with a blank project to design a new automation process.



Library  
Create reusable components and publish them together as a library. Libraries can be added as dependencies to automation processes.



Test Automation  
Start with a blank project to create a new test project



Template  
Create a template that you can use to build automations that have a set of predefined common characteristics

## New from Template



Attended Automation Framework  
This template allows you to create an attended automation with the Attended Framework using the new Forms and Desktop Triggers capabilities.



Background Process  
Create a process that starts in the Background and can run concurrently with other processes for as long as it doesn't use UI interaction.



Mobile Testing  
Create a project for your mobile app automated test cases



Orchestration Process  
Implement a process through service orchestration and human intervention as well as through long-running transactions.



Robotic Enterprise Framework  
Create a transactional business process that follows best practices for large scale deployments.



SAP Testing Project  
Create a project for your SAP test automation which includes SAP Change Impact Mining. IMPORTANT: This package version requires Studio v2021.10.0 or above.



Transactional Process  
Model a process as a flowchart diagram.

[More Templates](#)

In my opinion a template should come with version 0.1.0

# Example git log

```
cpm@host:/mnt/t/example$ git log --all --decorate --oneline --graph
* 36c3d50 (HEAD -> master) Merging development changes
* 23c2383 (tag: v2.0.0) Breaking: new input argument for config file
* 536ff71 (tag: v1.1.0) Feature: Added reporting
* 409121f (tag: v1.0.1) Fix: bug
* 0996c07 (tag: v1.0.0) Major: Release to Production
* d09d6db (tag: v0.2.1) Fix: bug detected during UAT
* 6969130 (tag: v0.2.0) Feature: Implemented requirement foo
* 83a5dac (tag: v0.1.1) Minor: Refactored code
* 3ceda6a (tag: v0.1.0) Initial version
```

commit messages or pull request titles are reflecting the version change

# Next step

Automating the build process  
on the basis of a pull request title!