

# No-code and code based test automation

Lin Tian

# Manage test automation and dashboards

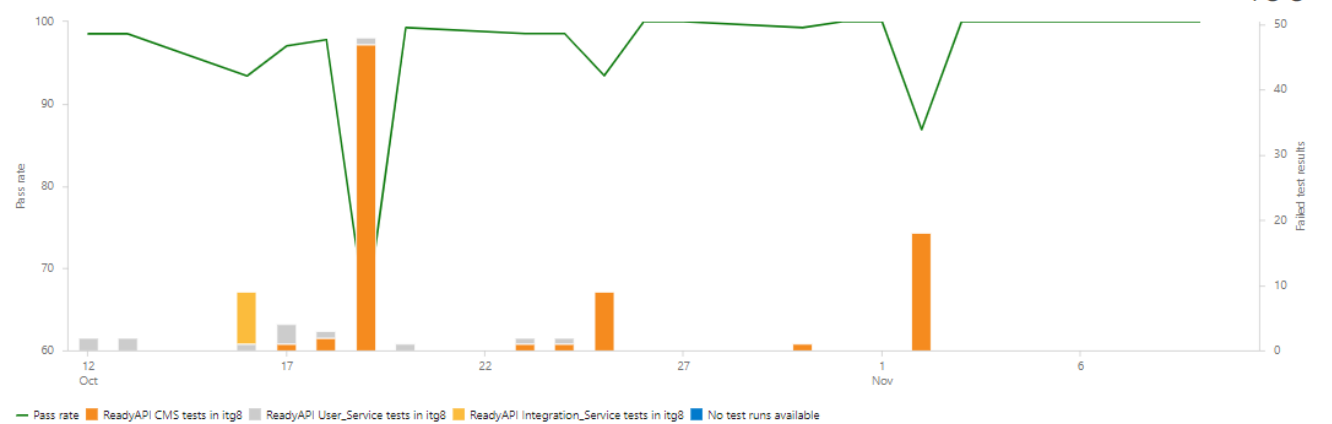
## Ryman Platform - scheduled runs

Target pass rate is 100% - if not, investigate!

These checks are run daily on master in order to provide us with increased confidence that the platform master branch is maintaining a base level of quality

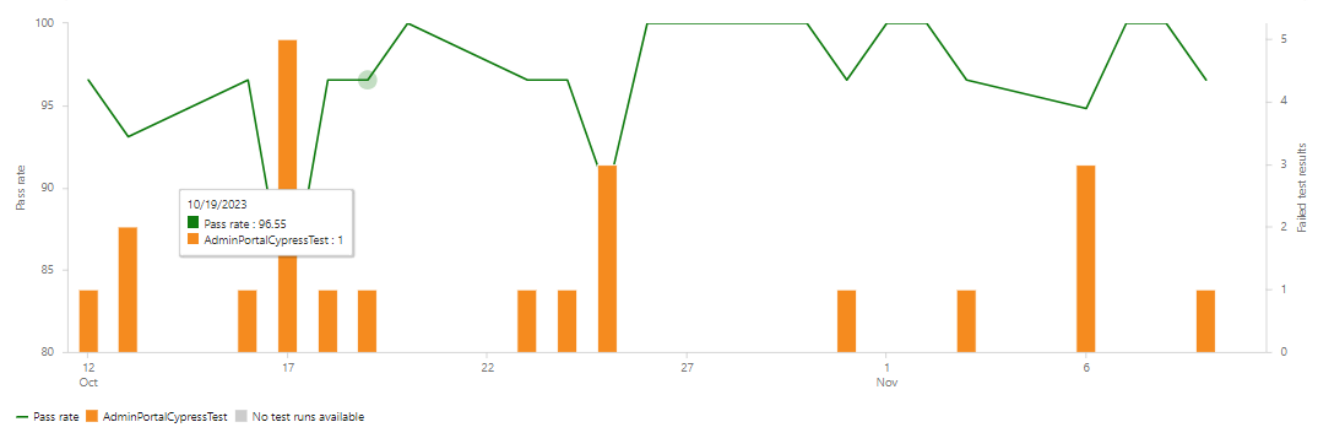
Platform daily ITG8 scheduled run - pass rate and failed test count (ReadyAPI)

Last 30 days

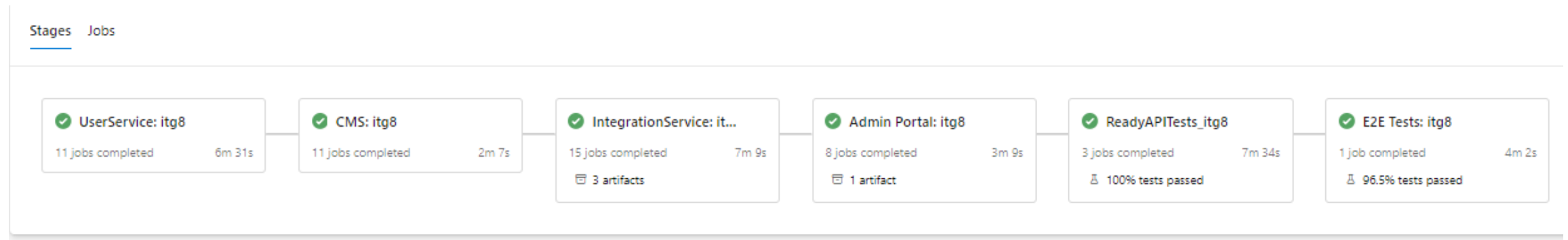


Platform daily ITG8 scheduled run - pass rate and failed test count (Cypress)

Last 30 days



# Manage test runs | scheduled pipelines



# COTS-based low-code / no-code based test automation approach

## ReadyAPI

The screenshot displays the ReadyAPI interface for configuring test suites. On the left, a tree view shows the structure of the test suite, including:

- APIs
- Functional Tests
  - Utility Test Cases
  - Warm Up Environment
    - Start Services
  - IntegrationService BusinessCritical
    - GET - Gets a list of all CodeTypes (enums)
      - Given I get authorised
      - REST Get a list of all the code types
    - GET - Gets a list of all related organisations
      - Given I get authorised
      - REST Get all the related organisations
    - GET - Gets a list of all related organisations by village
    - GET - Gets a list of all related organisations by village group
    - GET - Gets a list of all rooms by village
    - GET - Gets a list of all unit contracts by department
    - GET - Gets a list of all unit prices by village
    - GET - Gets a list of all units by village
    - GET - Gets a list of departments by village
    - GET - Gets a list of related persons by ids
    - GET - Gets a list of related persons by village
    - GET - Gets a list of staff by village
    - GET - Gets a list of unit contracts by ids
    - GET - Get a list of all users by village
    - GET - Get a list of villages
    - GET - Gets a user by id

On the right, a list of test suites is shown, each with a description and associated tags. A yellow arrow points from the 'Warm Up Environment' test suite in the tree view to its corresponding entry in the list:

- Utility Test Cases (disabled)
- Warm Up Environment (Tags: CMS, Integration\_Service, ...)
- IntegrationService BusinessCritical (Tags: IntegrationServiceBusinessCritical)
- Integration Service Regression (Tags: Integration\_Service)
- Integration VCare Delta API Regression (Tags: VCare\_Delta\_API)
- CMS BusinessCritical (Tags: CMSBusinessCritical)
- Event Releases Regression (Tags: CMS)

At the bottom, there are tabs for 'Description', 'Setup Script', 'TearDown Script', and 'Tags'. A link to 'Learn about regression test issues' is also visible.

# Open source / code-based test automation approach

# Cypress + Cucumber

```
@Regression
@BusinessCritical
Open Cypress | Set "@focus"
Scenario: 149138 Different views in admin portal
  When I click on Published button
  Then I see Category listing for event templates
  When I click to expand the template category
  Then I find templates are listed below
```

```
When('I click on Published button', ()=>{
  cy.intercept('GET', '/api/EventTemplateCategory')
  cy.contains("Published").click()
  cy.wait('@templatecategory')
})
```

The screenshot shows the Cypress test runner interface on the left and the application being tested on the right. The Cypress runner displays a file tree with 'Event01.feature' selected. The test log shows the following steps:

```
23 log visible.progress appears
24 get .progress-bar
25 - assert expected <div.progress-bar.progress-bar-
  animated.bg-info.progress-bar-striped> to be visible
26 get .mb-3.visible.progress
```

The application interface on the right is the Ryman Healthcare admin portal. It features a navigation menu with 'Event Catalogues' expanded, showing 'Charles Upham' as the selected item. The user is logged in as 'Independent Residents'.