Test Plan

Prepared By: IRWIN Core Team
Last Modification Date: 9/28/2023
API Version: Preparing for v9.0

Document Revisions

Date	Description	Person Responsible
6/5/13	Initial Draft of Plan	Craig Morgan
7/5/14	Update for Year Two	Jennifer Vaughan-Gibson
7/18/15	Update for Year Three	Stacia Taggart
7/24/16	Update for Year Four	Alex Laughton
10/10/16	Revisions to support Initial Response Resources	Jennifer Vaughan-Gibson / Craig Morgan
12/08/16	Review of Business Continuity Procedures	Craig Morgan
12/13/16	Finalizing for V4 Release to Extended Teams	Jennifer Vaughan-Gibson / Alex Laughton
11/14/17	Update for Year Five	Craig Morgan / Ryan Perkl
11/7/18	Update plan to address v5.1 and v6 release plan.	Luci Coleman
5/23/2019	Update plan to address v6 release plan.	Luci Coleman
6/18/2019	Final plan for v6.	Luci Coleman
10/28/2020	Update plan to address v7 release plan.	Mary Koran
11/13/2020	Final plan for v7.	Mary Koran
10/28/2021	Update plan to address v7.1 release plan.	Mary Koran
12/17/2021	Final plan for v7.1.	Mary Koran
09/30/2022	Update plan to address v8.0 release plan.	Mary Koran
10/14/2022	Final plan for v8.0.	Mary Koran
09/18/2023	Update plan for v9.0 release.	Mary Koran
09/28/2023	Final plan for v9.0.	Mary Koran

Contents

1 Introduction

The test plan defines the testing processes that will be used during the development of the IRWIN API. This includes the issue reporting process by which the Extended Teams can report defects identified from their individual testing procedures.

The following testing types are used during an IRWIN release:

- Unit Testing
- Functional API Testing
- Regression Testing, and
- Integration Testing

The purpose of this test plan is to clarify testing efforts to be completed by the IRWIN Core and Extended Teams. Additionally, the purpose is to identify the goals of testing surrounding quality and completeness.

This document is intended for the IRWIN Core Team to define the testing strategy for the Year Seven systems API, Console and related Work Products, and any other IRWIN hosted services, from both the Core Team and Extended Team's perspective. The testing sections covered in this document include:

- Unit / Regression Testing
- User Story / Functional Testing
- Integration Testing
- Business Continuity Testing

This document also aims to help the IRWIN Extended Team understand what testing has been completed, along with how issues are logged in IRWIN Tracker. An understanding of IRWIN Tracker by each partner system will be critical in the logging and communication of bugs across all IRWIN Work Products. This allows IRWIN to identify, track and fix bugs in an efficient manner.

1.1 IRWIN API Environments

IRWIN has three API environments: TEST, Operational Acceptance Testing (OAT), and Production. During any release, the release package is promoted from TEST to OAT to Production. Each promotion only occurs after appropriate testing and acceptance.

In October 2018, the IRWIN Core Team introduced a **next folder** into the TEST and OAT environments in an effort to expose under-development software in advance of Integration Testing.

With the addition of the next folder to these environments, the IRWIN Core Team is able to expose under-development software while still maintaining released software. The following table describes each environment's intended purpose.

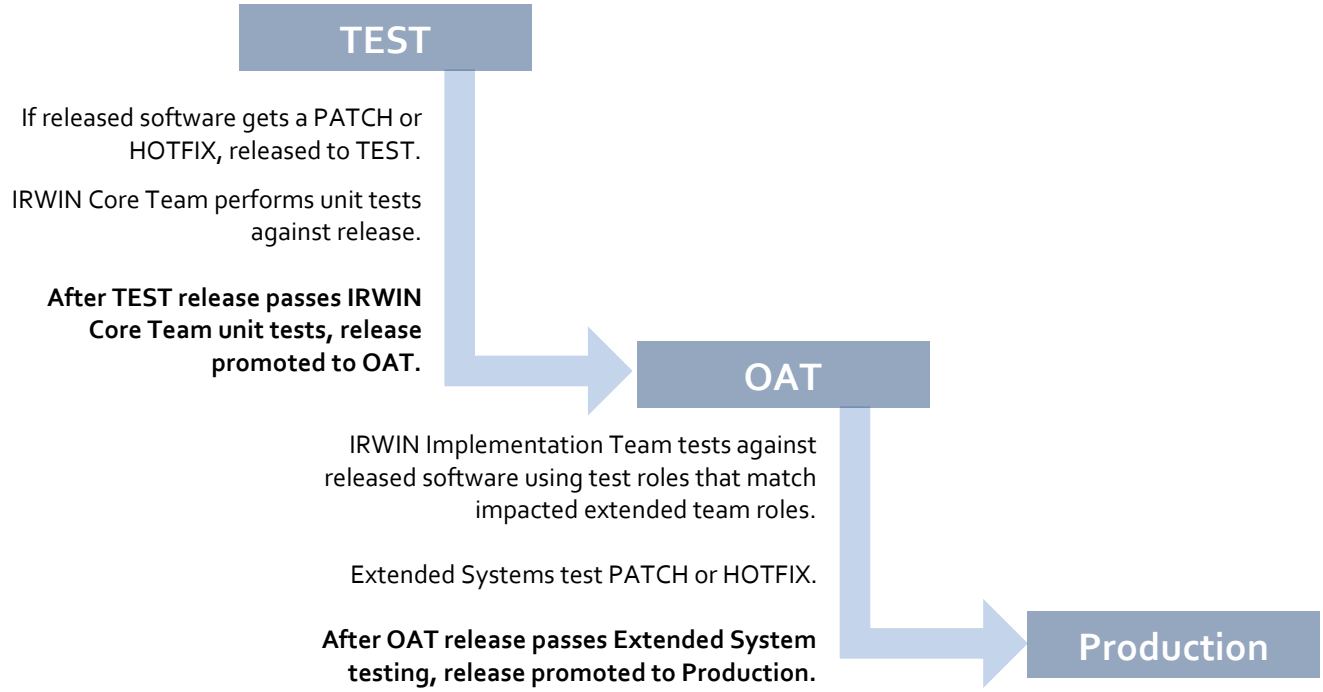
	TEST	OAT	Production
root	Extended Systems testing against released software.	Extended Systems testing & QA against released software.	Extended Systems using the IRWIN API as an integration service.
next	IRWIN Core Team testing against under-development software.	Extended Systems testing against under-development software.	N/A

For more information on the release management procedures, please refer to IRWIN's Release Management Plan on the Wildfire.Gov site.

2 Testing & Release Process Overview

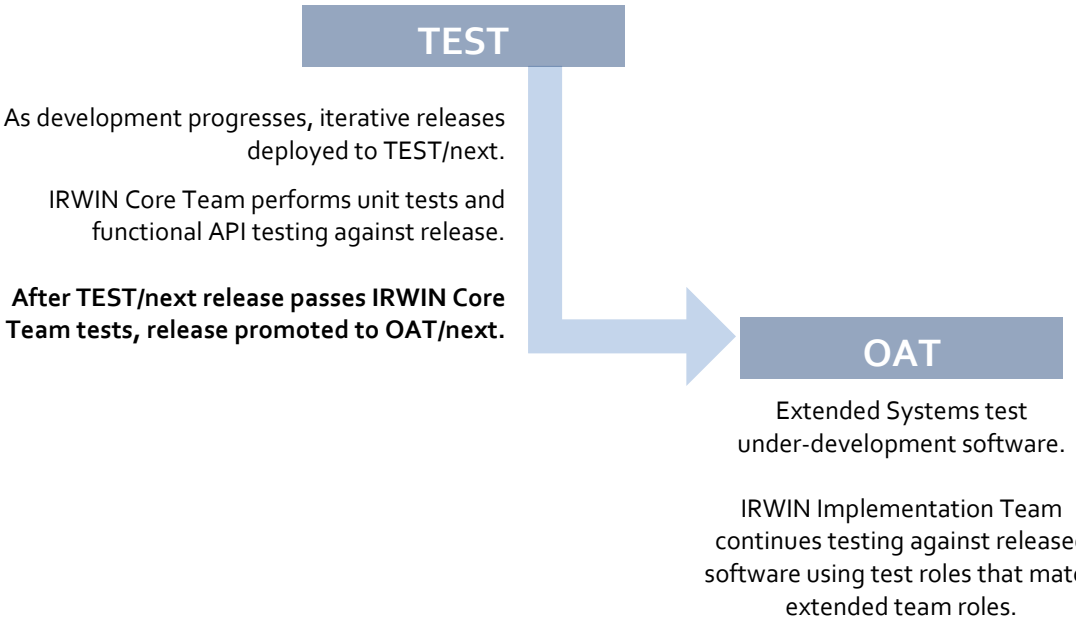
2.1.1 *root*

IRWIN's stable API services, at the root level of the directory, hold currently released software. After release into Production, the root environments undergo the following release deployment to promote a PATCH or HOTFIX deployment.



2.1.2 *next*

IRWIN's next folder holds under-development software. As the IRWIN Core Team continues to develop the upcoming versioned release (v9.0), the next folder undergoes the following release deployment process. The Development Team will send a notification to the community before each release to TEST/next. There will be a subsequent 48 hour testing window within TEST/next for the IRWIN Core Team to perform tests against the recent promotion. Upon successful testing, the Development Team will promote to OAT/next.



3 Testing Types

The following testing types are used during an IRWIN release:

- **Unit Testing** — The IRWIN Development Team will develop unit tests against the various functions of individual software components. Unit testing ensures that the smallest functions (or units) of software are working as expected.

- **Functional API Testing** — The IRWIN Core Team Business Leads will produce a series of user stories associated with an IRWIN version release. These user stories will be developed into a suite of automated tests. These user stories are commonly referred to as 'testing scenarios.'

  Functional API tests are performed on both under development software and release candidates. Importantly, functional API tests are run in the context of an end-user, utilizing the same access and authorization mechanisms.

- **Regression Testing** — The IRWIN Core Team will conduct regression testing of individual software components to ensure existing functionality still performs as expected. Regression testing will include both the IRWIN Development Team's unit tests and the IRWIN Core Team's function API tests.

- **Integration Testing** — The Extended Systems will verify the software's ability to generate acceptable results for production tasks and are responsible to perform a comprehensive test of the entire system. This testing will look for completeness to ensure that all parts are included and that the whole system will perform as designed. This completeness is expressed via a readiness self-review conducted by the Extended Teams' users that identifies the specific gaps that need to be addressed for a production release.

Type	Technology	Success Criteria	Tester(s)
Unit Testing	Visual Studio	Conditional test matches Pass/Fail criteria	IRWIN Development Team
Functional API Testing	vRest	Test case meet criteria defined in the User Story	IRWIN Core Team
Regression Testing	Visual Studio (Unit Testing) vRest (Functional API Testing)		IRWIN Core Team
Integration Testing	Determined by each Extended System	IRWIN Core Team Business Leads will provide criteria for testing acceptance	Extended Systems

3.1 Unit Testing

The IRWIN Development Team will develop unit tests against the various functions of individual software components. Unit testing ensures that the smallest functions (or units) of software are working as expected.

Unit testing tests for the basic API operations: (1) AddFeatures, (2) UpdateFeatures, and (2) QueryFeatures. Each basic operation has associated validation rules, specific to IRWIN API layers and tables. Unit testing tests field validity and minimum required fields for each IRWIN role and layer.

3.2 Functional API Testing

In preparation for a version release of IRWIN, the IRWIN Core Team Business Leads generate a series of testing scenarios to be met by the upcoming IRWIN version release. These scenarios are a focused set of workflows represented in IRWIN's Workflow Diagrams.

Functional API tests are performed on both under development software and release candidates. Importantly, functional API tests are run in the context of an end-user, utilizing the same access and authorization mechanisms. Functional API tests use end-user credentials to test the implementation of the scenarios, leveraging a combination of basic API operations and various IRWIN roles.

3.3 Regression Testing

Each versioned release of the IRWIN API inherits the functionalities defined and implemented in the previous versions. At each release, regression tests are conducted to ensure individual software components' existing functionality still performs as expected. These regression tests include the IRWIN Development Team's unit tests and the IRWIN Core Team's function API tests.

3.4 Integration Testing

For IRWIN Extended Systems, the ability to replicate real-world situations in a controlled environment is invaluable leading up to fire season. Prior to a Production Release, the IRWIN Core Team hosts an Integration Testing session to facilitate this controlled testing across the Extended Systems.

Note: If held, in-person meetings are mandatory for ReadWrite Extended Systems and optional for ReadOnly Extended Systems. With the status of COVID-19 limiting travel, the IRWIN Core Team may utilize virtual meetings in place of in-person meetings for both ReadWrite and ReadOnly Extended Systems.

Integration Testing is intended to track the efficiency and quality of IRWIN data transfer. During Integration Testing sessions, the testing scenarios developed by the IRWIN Core Team Business Leads are provided to the Extended Systems in an Integration Test Plan.

During the Integration Testing, these scenarios are conducted in live-time by the various Extended Systems. In many cases, a single scenario may involve multiple systems: e.g. one system performs an AddFeature and the next systems perform a series of UpdateFeatures.

4 Unit/Regression Testing

Extended Systems should report any issues with the IRWIN API to the IRWIN Implementation Team to Eric Neyman (eric.r.neyman@saic.com), Stephen Bankston (stephen.bankston@saic.com), or Brent Wood (brent.r.wood@saic.com).

When reporting an issue, please include the following details:

- Systems Affected
- Output Error / Result
- Reproducible Steps
- Credentials Used
- Priority Level (see table at right)

Issue Priority Level	Priority Rating Description
1 Crash / Data Loss	Also known as a Showstopper – it crashes the system with data loss or causes the system to freeze
2 Major Issue	These are critical issues that can crash the system but do not cause data loss
3 Minor Issue	Important issues that cause the desired requirement to be non-functional with no work-around
4 Cosmetic	Implemented requirement is non-functional but a workaround exists

Issue Priority Levels

As issues are reported, the IRWIN Implementation Team coordinates with the remainder of the IRWIN Core Team to begin troubleshooting. Each issue is logged into the IRWIN Core Team's Issue Tracker where it is assigned to the appropriate member of the IRWIN Core Team and given a priority.

As each issue is addressed, the IRWIN Implementation Team will maintain regular communication with the individual who reported the issue. Depending on the severity of the issue, a HOTFIX or PATCH may be required. Please refer to IRWIN's Release Management Plan on the Wildfire.Gov site for release procedures.

5 Business Continuity Testing

This plan is applicable to all cloud infrastructure services deployed, configured, and managed by DOI IRWIN on the AWS cloud. This includes hardware or software failure, a network outage, a power outage, physical damage to a building like fire or flooding, human error, or some other significant event.

This plan is specific for High Availability and Disaster Recovery scenarios that the Extended Systems might be affected by. The goal is for the Extended Teams to receive support in meeting response and recovery objectives, listed below:

- Recovery Time Objective (RTO)
  The Recovery Time Objective (RTO) is the time it takes after a disruption to restore a business process to its service level, as defined by the operational level agreement (OLA). The RTO that applies to the IRWIN system are as follows:
  - TEST Environment – 4 to 24 hours
  - OAT Environment – 4 to 24 hours
  - Production Environment – 5 minutes to 4 hours
- Recovery Point Objective (RPO)

The Recovery Point Objective (RPO) is the acceptable amount of data loss measured in time. The RPO that applies to the IRWIN system are as follows:

- Test Environment – 4 to 24 hours
- OAT Environment – 4 to 24 hours
- Production Environment – 5 minutes to 4 hours

5.1 Scenario – High Availability / Within Region

The IRWIN system resides within the US West AWS Region (Primary hosting region). When an incident is determined to have occurred within the primary Availability Zone (AZ) within the US West Region, the system will be restored in a secondary AZ.

The system in the secondary AZ will be restored from the snapshots (backups) within the applicable RTO and RPO timeframes. The secondary AZ (now the new primary AZ) will handle all traffic for the environment effected.

5.1.1 Infrastructure Testing

1. IRWIN support team (Innovate and SAIC) receives a system alert notification. OWF Enterprise Cloud Hosting team will determine if one of the Availability Zones (AZ) went offline and is unavailable.
2. OWF Enterprise Cloud Hosting team informs and alerts the IRWIN Development team (IRWIN_Support@innovateteam.com), IRWIN Core Team, and the IRWIN Product Owner who notifies the IRWIN Extended Team with the expected resolution date/time.
3. DOI IRWIN System Owner declares Disaster Recovery and authorizes the restore of the IRWIN system in a different AZ within the primary region.
4. In an alternate AZ (new primary AZ) that is available within the primary region, SAIC technical team restores the IRWIN system from the most recent snapshots that meet RPO requirements, performs necessary cloud configuration, and validates the infrastructure components to confirm they all are functioning as expected. The SAIC team will also complete the recovery tasks within the defined RTO parameters.
5. Upon completion of IRWIN deployment and verification, the SAIC team documents environmental variables that have been modified (such as IP's), the snapshot date/time (in UTC) and notifies the IRWIN system owner, DOI CISO, DOI/IRWIN government stakeholders, and IRWIN Development team (IRWIN_Support@innovateteam.com).

5.1.2 COTS ArcGIS Application Testing

The IRWIN Development team reviews the status of:

- ArcGIS Server

The COTS software is verified or mitigated of any issues. The IRWIN Development team reviews the server federation and status of IRWIN services. The IRWIN Development team issues are cataloged (such as resourcing the MXD's and SOI config file) and reported to the IRWIN project team.

5.1.3 IRWIN API Testing

The IRWIN project team verifies that both cataloged issues and any other issues encountered are mitigated after smoke testing the system.