The **Definitive** Guide to **Infor WFM Testing**

Table of Contents

A Letter from Our Co-Founder

Over twenty years ago, my co-founder Marce and I began helping enterprise customers with their Workforce Management (WFM) system go-lives.

Back then, technology moved slower, so you would only have to upgrade your WFM system every few years. When it was finally time to upgrade, we'd see many customers spin up new project teams. Unfortunately, those teams would struggle to understand how their WFM system worked or where to begin their testing efforts.

Many of those customers looked to us to develop a Quality Assurance (QA) plan. Paying their employees incorrectly wasn't an option, so they'd spend significant time, effort, and money to mitigate their risks. However, as soon as the project was finished, project team members would move onto their next assignments, and their QA documentation would start collecting dust. Worst of all, business stakeholders would find themselves in the same position again, i.e., afraid to make any changes to a WFM system that they didn't understand, until it was time for the next upgrade.

Around 2012, as Agile methodologies became more prevalent, enterprise organizations began to prioritize the speed of testing over the cost. The need to become more responsive to changing market and consumer demands created a need for faster releases, regular updates, and more frequent testing cycles. Unfortunately, most customers still didn't have the processes and technology to release more frequently without extensive and costly testing cycles (WFM was unique in this regard).

By 2014, because of our decade of experience seeing immense success with a structured testing approach, Marce and I knew that we were uniquely positioned to help these WFM customers. We had a vision of empowering our customers to deliver updates and make production changes faster while ensuring that absolutely no WFM mistakes would impact people's pay or create compliance issues.

Some years later, as customer's embraced cloud technology for their HR Solutions, the need for more frequent testing became even more critical. Not only did WFM customers now need to validate multiple pay-impacting software releases every year, but they also had to keep pace with the social and legislative changes that were coming their way. Fortunately, TestAssure.ai (our automated testing platform) was ready to help Infor customers successfully manage multiple releases per year while ensuring their employees were paid correctly.

Fast forward to this moment, where we feel it's more critical than ever to provide all WFM customers with the foundations for success. We work side by side with some of the top Fortune 100 brands and understand the challenges of managing WFM systems in the most complex and fast-moving enterprise environments. Best of all, we're proud to have worked with both our customers and WFM providers to lower the costs of automated testing technology and reduce the burden of testing for all types of customers – large and small.

The purpose of the Definitive Guide to Infor WFM Testing is to:

- **Help you understand a proven and comprehensive approach to doing structured testing of your WFM system.**
- **Give you the information you need to plan and carry out your crucial testing activities with confidence.**
- **Let you know that there is an automated testing solution out there to help you test your WFM system, should you be interested.**

Armed with this information, you will be able to make better decisions based on your distinct organization, budget, and risk profile. We hope that this guide provides assistance, relief, and a helping hand that enables you to move faster and with more confidence.

We're humbled to support you in your journey.
Sincerely,

Heiko Roth
COO and Co-Founder

heiko.roth@testassure.com

Chapter One
Introduction

Implementing Infor Workforce Management (WFM) is a considerable effort that has a broad impact throughout an organization. Failing to test your Infor WFM solution adequately can result in severe production issues that can impact your workforce and brand and even potentially give rise to serious legal issues.

Unfortunately, many Infor Workforce Management customers are often overwhelmed by their pressing workload and do not realize that testing is THEIR responsibility until it is too late. This oversight leads customers to cut testing short to meet a timeline or a budget resulting in inadequate testing and them taking on unnecessary risks.

A proven strategy for protecting your organization from the risks of poor testing is to adopt a structured testing approach. Structured testing provides a framework that helps you to sufficiently plan your testing activities and guide your team. The best practice is to outline this structured testing framework by creating and distributing a WFM Test Plan for your initiative that serves to define the testing approach and scope, identify key roles and responsibilities, and ensure that everyone on the project is aligned from the outset.

TestAssure has worked closely with Fortune 100 Infor Workforce Management customers for over 20 years. It is through our partnership with these customers – and their need to move faster and with higher quality – that the TestAssure Automated Testing Platform was developed. This Definitive Guide to Infor WFM Testing, therefore, has been compiled based on over 20 years of expertise and experience testing Infor WFM solutions specifically.

This guide will act as the ultimate blueprint for developing your organization's Test Plan document and for building confidence in your Infor WFM solution using a structured testing approach. We will provide extensive information on the different types of testing needed, key testing activities, and best practices for success.

TestAssure®

Chapter Two
Testing Methodology

Various types of testing should be performed to test your Infor WFM solution effectively. Each type of testing serves a specific purpose and evaluates a different aspect of your system to ensure that it is working as expected. It's essential to understand not only the various types of testing but also which ones apply to your specific situation, which will depend entirely on the complexity of your business, your scope of work, accepted risk tolerance, and unique budget and timeline. This understanding will help you to determine a testing approach that takes into account your distinct circumstances.
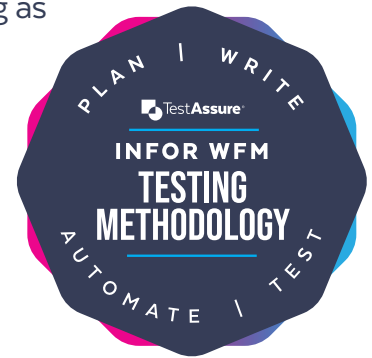
To assist you, TestAssure's Infor WFM Testing Methodology is defined by three distinct Testing Stages (Plan, Write, and Test) and can be seamlessly integrated into any implementation framework.

For example, our methodology can be integrated with the standard waterfall deployment (plan, design, build, test, and deploy) to effectively test your project before go-live implementation.

Our Infor WFM Testing Methodology has also been successfully used with iterative and agile implementation methodologies, which benefited tremendously from the introduction of test automation to reduce testing cycles.

PLAN	DESIGN	BUILD	TEST	DEPLOY
PLAN	PLAN			
		WRITE		
			TEST	TEST

TestAssure®
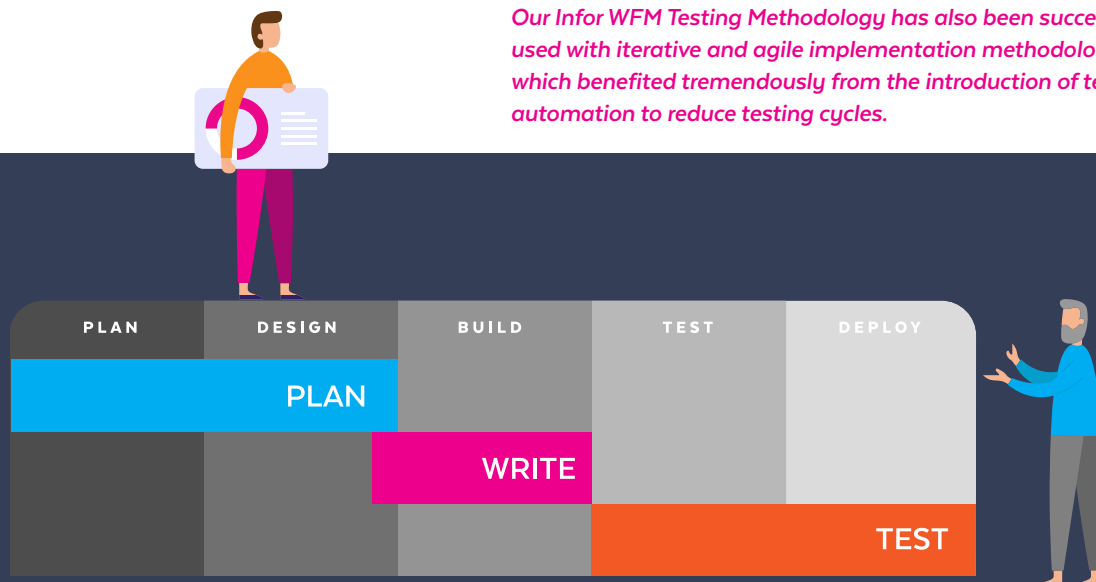
Testing Stages

Lets examine each of these Testing Stages in more detail.

1. PLAN

The key deliverable during the Plan stage is the creation of the WFM Test Plan document, which includes:

- Testing scope, including the type of testing and the functional areas in scope
- High-level testing timelines
- The testing environments and supporting technology to be used
- A plan for defect management
- Specific roles and responsibilities
- Potential risks at each stage

Developing a comprehensive WFM Test Plan is an essential step toward successful project management of your Infor WFM initiative. The WFM Test Plan promotes a shared understanding of scope, high-level timelines, and the roles and responsibilities of everyone involved. And it helps with securing the proper budget, resources, and tools required to deliver a Quality Assurance Program that balances timelines, resources, and risks in a way that is acceptable to leadership. As with any project, setting accurate expectations early is a primary driver of achieving the desired outcome.

In addition to the WFM Test Plan, an initial QA Project Management Plan is created to define and track key tasks, start/end dates, milestones, deliverables, and dependencies. These tasks can either be tracked as part of the overall project plan or separately as part of a stand-alone project plan.

DELIVERABLE(S): *WFM Test Plan, QA Project Management Plan*

2. WRITE

The *Write* testing stage consists of writing *Test Scenarios* to document the expected system behavior. Test scenarios are organized by the functional areas and/or are linked to the business requirements for traceability.

The prioritization of test writing should be based on the risks defined during the planning process with consideration toward the project budget and timeline. The test scenarios should be formally documented based on a business outcome and must clearly identify the expected results for a specific situation. The tests should be written by skilled team members who understand the desired business outcome and should be reviewed by subject matter experts (SMEs).

Finally, and prior to the start of test execution, the QA Lead will create a detailed *Test Execution Plan* that clearly identifies the number of test passes, the tests that need to be run (in prioritized order), who will run them, and when they will be run.

DELIVERABLE(S): **Suite of individual** *Test Scenarios, Test Execution Plan*

3. TEST

The *Test* stage consists of executing the test scenarios according to the *Test Execution Plan.*

Tests that do not pass are reviewed to ensure they are valid test failures. If so, a defect is raised, and the Defect Management Plan is engaged. The Defect Management Plan defines a rigorous, thorough, and collaborative process known as the Defect Triage Process. Each defect is prioritized during the process based on its severity, frequency, risk, and other criteria. Once reviewed, the defect is assigned, resolved, and scheduled for re-test before closure. This cyclical best practice continues until the project team determines that sufficient testing has been completed to release to production.

DELIVERABLE(S): *Test Execution Reports*

> **PRO TIP**
> *To effectively transition to and from each Testing Stage, we recommend formally defining stage gate reviews along with detailed entry and exit criteria for each stage.*

Types of Testing

Our experience has shown us that project teams leading the most successful Infor WFM implementations consider seven distinct types of testing to ensure a smooth go-live.

1. UNIT TESTING

Unit Tests are low-level tests designed to validate each individual configuration in isolation to ensure it functions as expected. Unit testing is completed during the build phase of the project.

RESPONSIBILITY: *Build Team*

2. FUNCTIONAL TESTING

Functional Testing is conducted to evaluate the compliance of a system with specified functional requirements. Functional Testing is designed to ensure a minimum level of software quality and is typically a precursor to further types of testing.

The Functional Testing process validates numerous combinations of inputs and boundary cases within Infor WFM to make sure everything meets the expected output. This type of testing covers several areas. For example, our testing of Infor Time and Attendance includes in-depth coverage of the main building blocks, such as overtime, holiday rules, and premiums zones, to ensure accuracy.

RESPONSIBILITY: *Infor Workforce Management Customer*

3. SYSTEMS INTEGRATION TESTING (SIT)

Systems Integration Testing (SIT) validates interactions between your Infor solutions, including interoperability with other internal and external platforms and hardware. This type of testing is incredibly important because it ensures that all your enterprise systems work together correctly. And it ensures that data moves accurately through your IT ecosystem and fully supports your various business processes.

For example, while there may be several integrations that require testing as part of your Infor Time and Attendance implementation, the Employee Import and the Payroll Export are two critical high-risk integrations that require extensive testing and should be validated with every major release.

RESPONSIBILITY: *Infor Workforce Management Customer*

4. PARALLEL TESTING

Parallel Testing is a comparison between your current Timekeeping system and your new Infor WFM solution. This type of testing helps you to identify and respond to any behavioral differences between the two.

The term Parallel Testing historically implies an apples-to-apples comparison between two types of software. However, in many cases, new features or policies are intentionally added as part of a new initiative, making an apples-to-apples comparison irrelevant. In these cases, your Parallel Testing should validate changes by reviewing and accounting for the differences used in real-life situations.

For example, the TestAssure automated approach processes production punches and schedules from your legacy system and runs this data into your Info WFM Test environment.

We process a subset of your employee population, and then we validate the comparison and provide detailed explanations of any identified variance.

RESPONSIBILITY: *Infor Workforce Management Customer*

> PRO TIP :

Don't take unnecessary risks!
Conducting a parallel test across a representative sample of employees using production data ensures that there are no unexpected increases/decreases in your employees' pay. Even under tight timeline and budget constraints, this type of 'payroll comparison' test is a critical safety check that should be performed with every release to production.

5. USER ACCEPTANCE TESTING (UAT)

User Acceptance Testing (UAT) validates that the end-to-end user workflows for the new Infor WFM solution meet the business requirements. Typically, departmental subject matter experts (SMEs) will perform real-world business workflows to ensure the software is fit for delivery to employees in the field.

Importantly, UAT solicits constructive feedback from participants by guiding them through their day-to-day business functions using production-like data.

User Acceptance Testing allows end-users to understand how the solution will support their daily Workforce Management tasks. Examples include the time off request process, payroll close, and schedule editing and posting.

RESPONSIBILITY: *Infor Workforce Management Customer*

6. REGRESSION TESTING

Regression Testing ensures that a recent fix or change to your WFM system does not negatively affect previously working functionality, wholly shielding your end-users from any impact. This type of testing is critical for any Infor software releases as well as compliance and policy changes that your organization makes to the system. It lets you verify the change and move forward quickly and with confidence.

RESPONSIBILITY: *Infor Workforce Management Customer*

> **PRO TIP**
> *Performing ongoing Regression Testing is critical as business requirements, product releases, and legislation are constantly changing how your Infor WFM solution functions. By doing continuous testing with technology such as TestAssure.ai, you can be confident that new Infor WFM software updates, business or policy changes, updated legislation, and new feature rollouts will not affect people's pay or create compliance issues.*

7. PERFORMANCE TESTING

Performance Testing validates the application's capabilities with respect to response time, stability, reliability, scalability, and resource usage. With the shift to cloud applications, the responsibility for system performance, scalability, and responsiveness rests with the SaaS vendor. But since the risk of underperformance can impact customers directly, most enterprise customers still prefer to also do their own performance testing on the customer-specific configuration of their Infor WFM solution.

RESPONSIBILITY: *Infor Team and Infor Workforce Management Customer*
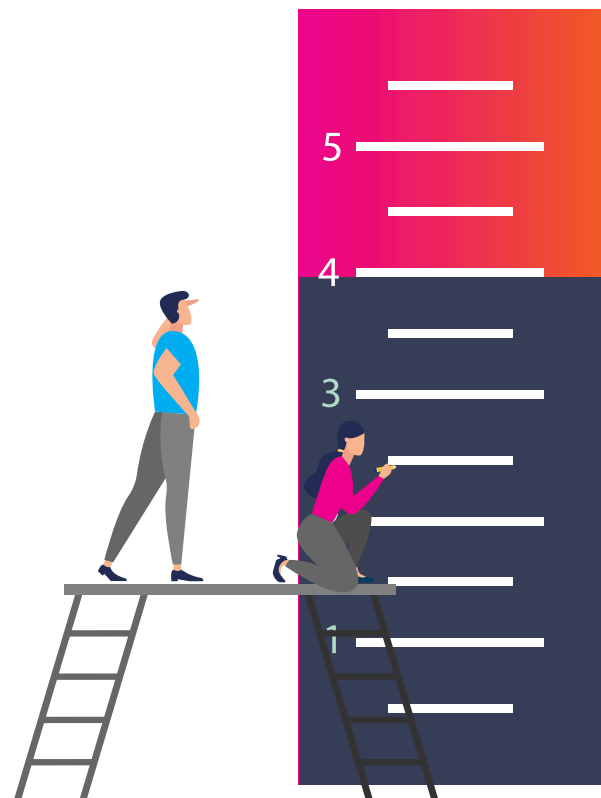
Whether you're rolling out a new Infor product or upgrading an existing platform, testing is a process that will help you mitigate the risk of deployment disruption. Our Infor WFM Testing Methodology of Plan, Write and Test, aligns with any implementation framework and is flexible enough to allow you to find the right balance between time, cost and an acceptable level of risk.

Chapter Three
Test Planning

A high-quality WFM release begins with effective test planning. Effective test planning ensures leadership buy-in and support regarding QA scope, budget, timeline, and the level of risk they are prepared to accept. In this chapter, we will introduce the WFM Test Plan and explain its use during your Infor WFM project.

The WFM Test Plan provides a set of guiding principles that define the testing approach that will be used and ensures that everyone on the project is aligned. It is defined early on as part of the project planning activities.

Key Components of a WFM Test Plan

TESTING SCOPE

An effective WFM Test Plan identifies which types of testing are in scope, the modules, entitlements, or functional areas that will be tested, and the employee populations, business units, locations, geographies, etc. that will be tested.

HIGH-LEVEL TIMELINES

High-level timelines need to be established for each type of testing. The WFM Test Plan defines the key milestones, such as:

- Requirements Approved
- Build Complete
- Solution Design Review (SDR)
- Start and End Dates for Each Testing Activity
  - Functional Test Planning, Writing, and Testing
  - SIT Planning, Writing, and Testing
  - Parallel Test Planning, Writing, and Testing
  - UAT Test Planning, Writing, and Testing
- Go-Live

ENVIRONMENTS AND SUPPORTING TECHNOLOGY

This section specifies the test environments, whether tests will be automated or manual, and the tools you will use to test (e.g. a test automation platform such as TestAssure.ai). It also defines the test data management strategy, as identifying, creating, and managing test data needs to be done meticulously. Each type of testing will have separate data management needs and the protection of personal information and data security needs to be incorporated into the plan. Finally, the *WFM Test Plan* must define what data is required, who will create it, and who will manage it.

DEFECT MANAGEMENT PLAN

The *Defect Management Plan* defines the systematic process used to identify and fix the software bugs discovered during the testing phase of your initiative. This plan defines this process, the tools or technology used, as well as the Defect Manager who will own and manage its execution. The plan will also specify how to communicate, raise, and resolve defects found in the WFM product with Infor.

ROLES AND RESPONSIBILITIES

Clearly defined roles and skillsets that are needed to execute your testing must be included in your *WFM Test Plan*. In addition, you should identify dependencies on other teams as well as external subject matter experts (SMEs) who will play a role in testing. An Infor Workforce Management system implementation requires guidance and support from human resources, payroll, finance, benefits, and IT. These resources will need to be secured well-before you begin testing to ensure they are available and can meet the specified timelines. It is crucial to identify the roles that need to be filled and have an estimated headcount required to meet your testing timelines.

QA PROJECT MANAGEMENT PLAN

All QA tasks and milestones should be managed as part of a formalized *QA Project Management Plan.* This task level project plan includes all individual testing tasks, resource assignments, and timelines. Most importantly, the plan should maintain the upstream and downstream dependencies required for testing to begin. Depending on the size and scope of your QA activities, the *QA Project Management Plan* can be incorporated into the overall project plan or maintained in a separate plan.

Chapter Four
Functional Testing

The purpose of Functional Testing is to validate that the solution is configured correctly and meets the unique functional requirements of the customer. The process involves the comprehensive testing of inputs and boundary conditions to ensure that you have correctly configured the application to achieve the desired business results.

TestAssure
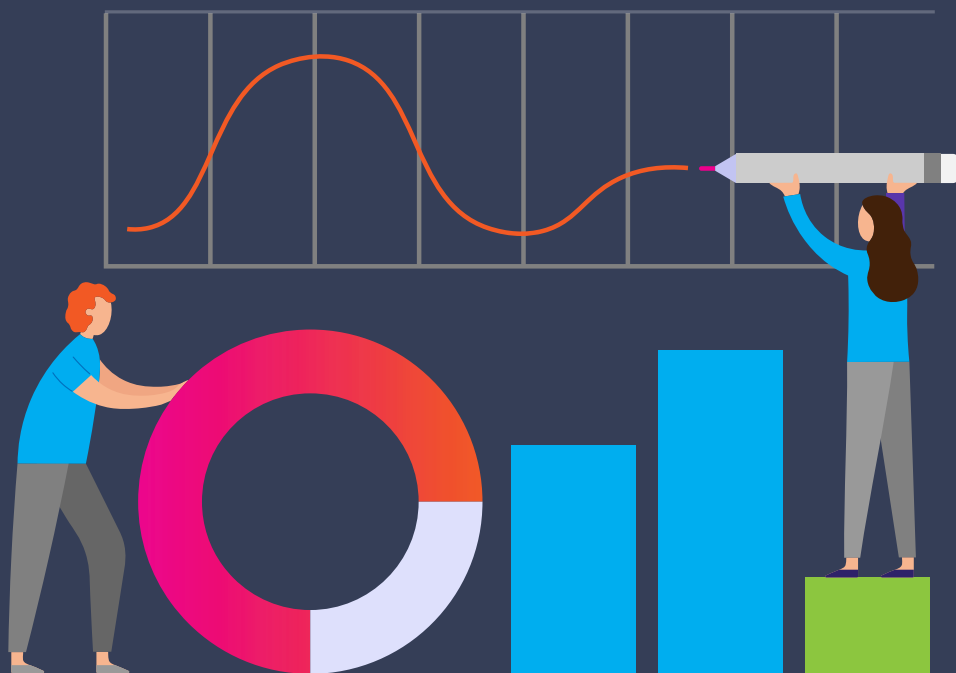
RESOURCES

The members of the project team are responsible for writing and executing the Functional Tests. This may include the vendor, Systems Integrator (SI) partner, or customer (i.e., the IT team or a dedicated Quality Assurance (QA) team). A best practice is to have Functional Testing conducted by independent testers that have not played a role in configuring the solution (or, at a minimum, at least have not configured the particular functionality of the test). This yields an objective third-party perspective that ultimately reduces the risk of misinterpreting or misunderstanding the requirements

ENVIRONMENTS

Functional Testing is performed in a separate QA environment.

TIMELINES

Functional Testing occurs after Build Complete (ie. the configuration of the WFM solution is completed.) If there are delays with the completion of the build, the testing of specific functional areas can still begin with proper planning and communication. In the event that this is necessary, a formal and structured conversation to clarify the outstanding deliverables and establish new delivery dates is needed, and analysis should be done to determine which testing can occur in the interim.

KEY TESTING ACTIVITIES

The table below outlines the key testing activities for Functional Testing at each Testing Stage:

PLAN

- Identify the entitlements (Timekeeping (Salaried/Hourly), Accruals, Scheduling, etc.), geography (country, state, etc.), locations or types of locations, and the key business processes that are included in the scope of testing.

- Confirm the source for the requirements, which may include formal requirements documentation, design documents, consultation with SMEs, or pre-existing configurations/legacy systems.

- Identify key build dependencies, dependencies on external teams, vendors, and SMEs, and secure access to the resources required to begin Test Case Writing.

WRITE

- Tests should be organized by functional areas (daily overtime, weekly overtime, shift premiums, holiday pay, etc.) and/or personas (hourly employee, salaried employee, manager, etc.) to ensure they are easy to reference.

- Write Functional Tests that detail a specific business situation with clearly defined inputs, actions, and expected results. Functional Test scenarios should validate the business logic within the system.

- Functional Tests should be exhaustive, testing all possible inputs, combinations, and boundary cases.

- Tests should be linked to business requirements to provide clear test case traceability and test case coverage analysis.

- Prepare for test execution by preparing the QA test environment, setting up test data, and procuring any required testing tools.

TEST

- At a minimum, all tests should be run at least once. TestAssure recommends planning for a 2 or even a 3-pass testing strategy to account for the re-testing of defects and/or to validate missed or changing requirements.

- As defects are resolved, and tests are re-validated, a subset of tests should be re-executed to ensure defect fixes did not introduce new issues.

Chapter Five
System Integration Testing (SIT)

The purpose of System Integration Testing (SIT) is to validate that the solution correctly interacts with integrated systems and transmits (both sends and receives) the data required to support business processes. SIT involves in-depth testing of the interfaces with external systems and any external hardware. The SIT process also validates the data flowing in and out of system modules for end-to-end testing. This is important as these integrations have a direct impact on how the systems will function as a whole.

TestAssure®

RESOURCES

System Integration Testing is performed by the project team, with end-to-end system interfaces and hardware testing done in close collaboration with the following teams:

- **Customer IT** helps facilitate data transfers, e.g., data processing, security credentials, etc.
- **Source System Subject Matter Experts (SMEs)** from upstream systems work closely with project teams to ensure the correct use cases are identified, the exact test scenarios are entered, and an export is triggered.
- **Destination System SMEs** from downstream systems help to ensure the identification of the correct use cases, ensure the exported data is processed correctly, and that the exported data is validated through processing.

ENVIRONMENTS

SIT is performed in a QA environment with integrated hardware and software. All upstream and downstream systems should be included in this integrated QA environment.

Testing import interfaces requires that test data is keyed into source systems and imported into Infor WFM. Testing export interfaces requires that test data is entered into your Infor WFM solution, exported to target systems, and validated in the QA environment of these destination systems.

To test external hardware, you should connect dedicated non-production hardware to the SIT environment. In addition, with modern systems having a greater emphasis on mobile, you should also test on various mobile platforms and configurations.

TIMELINES

SIT should occur after the Unit and Functional Testing of individual system components. This is because executing SIT tests requires significant time and effort, thus performing unit and functional testing before SIT ensures the efficient use of time and resources. It also guarantees that the system exhibits a basic level of stability and correct business logic before engaging others.

SIT should also be completed before User Acceptance Testing begins. (Note: Accommodations can be made for lingering low to medium priority issues that have minimal impact on UAT participants.)
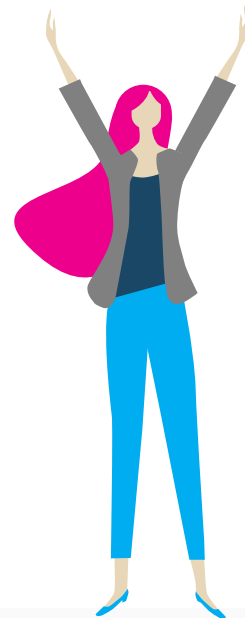
KEY TESTING ACTIVITIES

The table below outlines the key testing activities for System Integration Testing at each Testing Stage:

📋 PLAN

- Identify the integration points that should be tested. This includes a listing of interfaces (both imports and exports), hardware (clocks, POS, etc.), and scheduled processes (archiving, schedule generation, etc.).

- Confirm the source for the requirements, which includes Cross Reference Table (CRT) documentation, technical designs, and business process requirement documents.

- Identify source system SMEs (and secure their time!) that will be required to identify use cases, key in test data, and trigger data exports to Infor WFM. Do this early!

- Identify destination system SMEs (and secure their time!) that will be required to identify use cases, process exports from Infor WFM, and validate test cases. Do this early!

✍️ WRITE

- Write integration tests that validate the flow of data in specific business situations, or use cases. For example, in the case of an Employee Import, specific tests should be written for all Personas that cover a: new hire, promotion, demotion, job change, rate change, re-hire, location change, termination, etc.

- Tests should be organized by integration and business use case to ensure they are easy to reference and provide business context.

- Tests should be written in priority order based on risk and should confirm all possible combinations, mappings, and boundary cases.

  - The employee import and payroll export are two critical (and complex!) integrations that should be included with any Infor WFM testing project.

- Prepare for SIT test execution by specifying, allocating, and setting up the required test users and data, identifying shared locations, and securing the required credentials to exchange the required data between systems.

📋 TEST

- Work in collaboration with upstream system SME's to enter test data and trigger exports into Infor WFM.

- Validate imported data in Infor WFM.

- Work in collaboration with downstream system SME's to process exports from Infor WFM and validate each test case.

Chapter Six
Parallel Testing

Parallel Testing serves an essential purpose as it helps your entire organization to build confidence in a new system release, upgrade, or major feature release by identifying and accounting for the behavioral differences between two new versions of the software.

In Infor WFM, specifically, Parallel Testing is a comparison between your current Timekeeping system and your new Infor WFM solution. Being able to comprehensively explain those key differences, assess your organization's impacts, and gain buy-in from key customer subject matter experts (SMEs) is vital to secure a smooth go-live.

TestAssure®

RESOURCES

The project team works with Payroll SMEs to determine the scope of Parallel Testing and validation to be performed. While the project team performs most of the preparation, processing, analysis, and documentation for Parallel Testing, your identified Payroll SMEs are accountable for understanding, validating, and ultimately signing off on any variances discovered. This often requires a case-by-case walk-through to validate that each identified difference is indeed legitimate and expected.

When discrepancies between the TEST and PROD outputs are discovered, the expertise of additional SMEs may be required, including:

- **Legal** to provide the organization's interpretation of laws and CBAs, if clarification is needed
- **Human Resources** to clarify company policies, help you understand the impact on employees, and communicate those changes to the field staff
- **Finance** to understand and approve of any budgetary impacts or profit and loss reporting changes
- **Operations** to approve of any changes to scheduling or labor demand

ENVIRONMENTS

An efficient Parallel Testing process requires a stable QA environment where the business rules have been tested and verified. It also requires functioning interfaces that can import real-world data to support Parallel Testing in the QA environment with data loaded from production.

TIMELINES

Parallel Testing should take place after at least two passes of Functional and System Integration Testing.

> **PRO TIP**
> *Parallel Testing provides an effective way to complete SOX compliance audits before your go-live. For example, you can use the TestAssure.ai Payroll Compare tool to achieve a meticulous accounting of your pay changes and quickly proceed through your SOX audit process. The Payroll Compare tool can then automate compliance for every subsequent Infor WFM software update or patch.*

KEY TESTING ACTIVITIES

The table below outlines the key testing activities for Parallel Testing at each Testing Stage:

PLAN

- Identify the Business SME(s) that will review the Parallel Test Validation Report and walk them through the validation effort that will be required to secure their sign off.

- Work with your Corporate IT and Data Security teams to gain approval to pull the selected employees' production data and import the data into your WFM QA environment.

  – Check your organization's policy on using personally identifiable information (PII) in QA environments – masking of PII may be required.

WRITE

- Identify a representative sample of employees that will be selected for review. The key is that the employees selected should cover multiple Personas and be representative of the full population.

- Confirm the historical dates that will be selected (TestAssure recommends validating a week with a holiday and one without).

TEST

- Extract the required historical data (punches, schedules, payroll export, etc.) for the selected employee population from the existing production system.

- Process the extracted historical data into the QA test environment. This may require manual manipulation of the exported production files for import into Infor WFM and/or the masking of data based on your organization's security policies.

- Run the Payroll Export in Infor WFM.

- Conduct a thorough comparison of the test payroll export versus the production output, analyzing each day for each identified employee for the selected time periods. For each discrepancy, carefully document the reason for each difference (i.e. is this caused by a requirement or policy change?).

- Review the root of each difference with the identified Business SME to gain signoff approvals from whoever will ultimately sign off on the final results.

Chapter Seven
User Acceptance Testing (UAT)

User Acceptance Testing (UAT) validates that the end-to-end workflows will in fact allow end-users to perform their day-to-day business functions. As such, UAT is sometimes referred to as "week-in-the-life" testing because it uses PROD-like data to mimic the real-world business processes performed by critical roles in the organization.

These 'user journeys' are best validated by hands-on, proven leaders from the field that expertly understand their business and are motivated to provide constructive feedback on how the system supports each workflow. This dialogue with UAT participants is a critical aspect of UAT and provides early and important feedback on whether the delivered solution will be enthusiastically adopted by the field and deliver the projected return on investment (ROI).

Unfortunately, in our experience, this feedback is often missed, which is why we prefer to think of User Acceptance Testing as more of a 'survey' than a 'test'. Done right, a successful UAT is a good barometer for change management, training, and ultimately user adoption.

TestAssure®

There are several UAT delivery models that can be used to solicit this important feedback from UAT Workshop Participants:

1 ON-SITE, INSTRUCTOR-LED
The instructor should provide a brief overview of the solution before the end-users test the real-world workflows by actually logging into the Infor WFM environment..

2 OFF-SITE, INSTRUCTOR-LED
This is similar to the on-site approach, but the sessions are held online in a virtual setting.

3 OFF-SITE, INSTRUCTOR-PERFORMED
This isn't our preferred approach but can be used when field SMEs are resource-constrained and unavailable or when costs are a concern. This variation of an instructor-led workshop is delivered virtually and sees the instructor perform all the persona-based use cases (i.e., clicks through the application). In this format, the field participants (SMEs) follow along and document their results and comments in real-time, providing instantaneous feedback while the instructor performs the tests.

RESOURCES

User Acceptance Tests are written by project team members, such as the Customer or Systems Integration (SI) partner, in collaboration with Business SMEs with intimate knowledge of the business processes and standard operating procedures (SOPs). UAT workshops are led by an instructor, but the end-user participants are ultimately accountable for validating that the solution supports their business needs and for 'accepting' that the solution is ready for release.

ENVIRONMENTS

UAT is best performed in a separate QA environment or instance. If this is cost prohibitive, UAT may also occur on the same QA environment as other testing as long as measured security precautions are taken, such as masking or separating test data, assigning test employees and end-users to each tester, and carefully coordinating the timelines for project team deployments and bug fixes.

TIMELINES

UAT execution occurs after Functional Testing and System Integration Testing (SIT) is complete with no high or showstopper defects.

If for any reason there is a need to begin UAT before Functional Testing and SIT is completed, careful planning and risk management will need to take place. Remember that UAT offers an important outsiders perspective and represents an opportunity for end-users to provide valuable feedback on how the new system supports their day-to-day business needs. If the system in which participants perform UAT is missing key features and/or is full of functional defects, this important opportunity will be missed and the feedback will be understandably negative.

> PRO TIP
UAT is critical for the success of your initiative, but it's a costly endeavor. Removing SMEs from regular business operations could impact your top and bottom lines. While customers often approach us with planned UAT periods of four to six weeks, this is often a reflection of poor functional and integration testing, lack of planning, and/or a simple lack of trust in the vendor or chosen SI. In our experience, with the right approach, UAT can be performed in one to two weeks.

KEY TESTING ACTIVITIES

The table below outlines the key testing activities for
User Acceptance Testing at each Testing Stage:

📋 PLAN

- Identify the in-scope business processes for the key roles or personas that will be using the system. For example:

  - **Employee Persona.** View My Timesheet, Approve Punches, View My Schedule, Request Time Off, etc.

  - **Manager Persona.** Approve Timesheet, Approve/Reject Time Off Approvals, Create Schedule, etc.

  - **Area Manager Persona.** Review Schedule, Approve Schedule, Review Attendance Exceptions, etc.

  - **Payroll.** Close Payroll, Approve Leave, Apply FMLA.

- Define the delivery format of the UAT workshop including when, where, who will participate and who will lead the UAT workshops.

- Identify whether extensive knowledge transfer and/or training will be required for UAT participants, and who will provide the training/KT.

✍️ WRITE

- Review business processes, SOPs, project requirements, design documents, and functional test cases to map and document the identified use cases by persona.

- Prepare training materials to ensure that each UAT participant receives system training before the UAT test case execution.

- Prepare the UAT test environment and the setup of test data for each UAT participant including any data masking (end-users will want to see data that is as similar as possible to real-world scenarios).

- Establish frequent touchpoints with the project team to manage dependencies and timelines, and to ensure any requirements or process changes are communicated to the test case writers and incorporated into the UAT Workshop materials and process flows.

📝 TEST

At the start of the UAT Workshop, each participant will receive a personalized UAT test plan that sets clear expectations and includes:

- Clear identification of the personas that each participant will validate.

- Documented test cases and process flows for each test.

- Test data and unique sample personas allocated and assigned to each participant.

- Timelines for completion each day.

- Details on how to capture and provide feedback.

- Providing project leadership a daily summary of UAT Status, defects, and issues. The build team should be on hand and ready to respond to any concerns raised by the participants.

- Consolidation and review of participant feedback with project stakeholders.

TestAssure®

Chapter Eight
Regression Testing

Regression Testing ensures that a recent update or change to a software product has not negatively affected previously working functionality. Historically, Regression Testing was an infrequent occurrence for WFM Solutions — QA teams would conduct Regression Testing every few years when the business upgraded their Infor WFM Solution or when there was a significant policy, process, or legislative change.

Today, staying agile is the new business imperative. Customers are moving to enterprise cloud solutions, like Infor CloudSuite, so they can respond more quickly to competitive and market influences, as well as maintain compliance with new legislation. In addition, Infor CloudSuite delivers software updates and patches every month that support innovative new features and ensures your system is always up-to-date and secure.

In practical terms, this pace of constant change means you'll need to conduct more testing, more often, and within a limited window of time. This can put pressure on your SMEs and heighten the potential risk to your organization. This raises the importance of establishing an efficient Regression Testing process so that you can instill confidence in both your team and stakeholders that the changes introduced will not negatively impact any existing employees or features. Additionally, proof of this type of testing may also be required for a SOX compliance audit.

That said, building this level of confidence can be quite time consuming as every change to your WFM system could potentially compromise your employees' pay and benefits as well as your organization's bottom line.
Fortunately, this process can be accelerated significantly with the introduction of an automated testing technology such as TestAssure.ai.

RESOURCES

The Infor Workforce Management customer is responsible for verifying that every release is tested in the available maintenance window before going into production. Customers are responsible for testing using their data and their configuration, including extensions and external interfaces. To avoid business interruptions of your unique configuration or employee workflows any new software changes, such as new features, configuration changes, and even bug fixes made by the vendor need to be verified.. During 'business as usual' operations, the business owner or systems administrator is responsible for Regression Testing, but it might be handled by a project team if there is an active initiative underway.

ENVIRONMENTS

Regression Testing should take place in a non-production environment with current production configurations in place.

TIMELINES

It is imperative that Regression Testing occurs before every production change, including:

• New Infor Workforce Management software updates and patches

• Any changes to customer business processes or policies

> **PRO TIP**
> *Customers that implement effective Regression Testing strategies deploy releases to production more frequently and catch more defects before they hit production. This results in faster ROI, fewer incident tickets, and reduced support costs.*

• Implementation of federal or state legislative changes

• When a new feature is configured or implemented

• Before rollouts to new employee populations, locations, business units, or geographies

KEY TESTING ACTIVITIES

There are a few key steps to follow when it comes to Regression Testing.

1. Be sure you have elected to receive detailed information on upcoming Infor releases by subscribing to notifications in the Infor CloudSuite Self-Service Portal (CSSP). These important notifications will ensure you have insight into the upcoming release schedule for Infor WFM software updates and patches and will help you determine the timing and frequency of when Regression Testing needs to occur.

2. For every Infor software update and patch, review the changes being introduced to the system and identify what Regression Testing should be executed. The more mature your Regression Test bed is, the more that existing tests can be leveraged.

3. Depending on the impact of what is to be installed, a full backup may be performed by Infor, or by your Systems Integration (SI) partner prior to the implementation in the customer's production environment.

4. Execute the tests, and, as with all other types of testing, always investigate any failures that occur. Raise any defects by logging a support incident via Concierge where Infor Support can work with your or your SI partner to help triage and then retest the fix once it is delivered.

5. As changes occur, update the Regression Test bed accordingly. The project team should continue to modify the Regression Test bed as part of the change control process and grow it over time as new features are introduced. It is especially essential to add test cases that protect against any defects that slip through testing and produce defects in production – *especially for pay impacting defects!*

Test**Assure**®

Chapter Nine
Defect Management

A defect is a situation where the actual system functionality is different than the expected system functionality. The Defect Management Plan identifies the team members responsible for managing and resolving defects, how often they will meet, what tools they will use to support Defect Management, and the Defect Triage Process.

TestAssure®

What is Defect Triage?

Defect Triage is the systematic process used to identify, prioritize, fix, retest, and close defects.

IDENTIFY DEFECT	REVIEW DEFECT	ADDRESS DEFECT	RE-TEST DEFECT	CLOSE DEFECT

The Defect Triage Process is a time and resource-intensive task. So it's critical that the team responsible for writing and running test cases, i.e., the Quality Assurance or QA Team, always verify and double-check that every failed test is a legitimate failure and worthy of further investigation by the Defect Triage Team. When a test fails, the QA Team should first verify that:

- The correct test script was used
- The test data used was set up correctly
- The correct steps were followed
- The results were captured and interpreted correctly
- There are no issues with the automation code (if applicable)
- The test did not fail because the system under test was unavailable or unreachable

It's important to note that not every failed test case maps directly to a unique defect, i.e., a single defect may cause several test cases to fail. So similar failed tests should be grouped for further investigation.

These critical validation steps, taken by the QA Team before raising defects, are proven best practices for improving the efficiency of the Defect Triage Process.

Overview of the Defect Triage Process

IDENTIFY DEFECT

- Once the QA Team identifies a defect, the tester will log the defect into the Defect Tracking System and assign it to the Defect Manager for review.
- All defects should be fully documented and include:
  - Test scenario that was run
  - Steps to reproduce
  - Expected results
  - Actual results
  - Name of the tester/date/time of the test
  - Screenshots

REVIEW DEFECT

- The Defect Manager reviews new defects for clarity, documentation, and duplication to ensure that sufficient information has been provided to the Defect Triage Team to review
- The Defect Triage Team includes representation from the business, Build Team, and QA Team
- The Defect Triage Team meets frequently (often daily) to prioritize each defect based on its severity, frequency, and risk
- Once the Defect Triage Team has reviewed and accepted a defect, it is assigned to a Build Team member for investigation and resolution in prioritized order

ADDRESS DEFECT

- The Build Team member investigates the defects, identifies root cause, and addresses the issues based on their assigned priority:
  - If more information is required, the team member will contact the tester directly
- There are several common reasons why a test may have failed, including:
  - A change in the system requirement
  - An incorrect expected result
  - An incorrect system configuration
  - There is a core product defect with the system under test
- If further clarification is required, the Build Team member will work with stakeholders to clarify the correct system behavior.

RETEST DEFECT

- Once the defect has been addressed, it is assigned back to the testing queue for retesting by the QA Team.
- The original defect is retested to verify if it has been resolved.
- If the test fails, it should go back to the 'Review Defect' stage.

CLOSE DEFECT

- The Defect Triage Team reviews the defect after it passes retest for final review and closure. Once this step is complete, the team notifies all affected parties that the issue is resolved.

Defect Severity and Priority

Defect severity and priority are two terms used in the Defect Triage Process. Severity relates to the system's standards and functionality, whereas priority refers to scheduling the work required to investigate and resolve a defect.

DEFECT SEVERITY

Severity is the degree of impact the defect has on the application component or system. An initial severity can be set by the QA Team when the defect is created and should be reviewed by the Defect Triage Team.

Showstopper	Major components are not functioning. No workaround is available. Testing cannot continue until this issue is addressed.
Serious	Major feature/business process impeded; non-trivial workaround exists.
Moderate	Feature of the system not functioning, a usable workaround exists.
Minor	The defect does not impact the functionality or user workflow.

DEFECT PRIORITY

Priority refers to the order in which the defect should be fixed and is usually driven by the business value of fixing the defect. The defect's priority should not be set by the QA Team, but rather agreed to by the Defect Triage Team upon review of the entire scope of out-standing defects.

Urgent	The defect needs to be addressed ASAP. Emergency release outside of the normal deployment schedule may be warranted.
High	The defect should be fixed in the next release process.
Medium	The defect should be fixed after urgent and high defects are addressed.
Low	The defect does not impact the functionality or user workflow.

Defect Triage Lifecycle and Process Flow

The Defect Management Plan should include a formal process flow depicting the full lifecycle of every defect. This should include the defect status and assignment of ownership at each stage.

Chapter Ten
Reporting

The cornerstone of a successful testing initiative is transparency into test planning, writing, and execution, as well as clear and timely communication of testing results and defect status. The best way to proactively share this information is through a standardized and predictable set of reports.

Key Reports

Our experience has shown us that the following key reports provide the analytics needed to effectively manage QA for the majority of Infor WFM projects.

REPORT	PURPOSE
Testing Status Report	Delivered throughout a project; this report provides all stakeholders a weekly update on all facets of the projects' QA responsibilities. In many projects, the Testing Status Report is incorporated in an overall Project Status Report.
Test Execution Report	Delivered during the Test stage; this daily or weekly report provides stakeholders a snapshot of what tests were executed in the last elapsed time period.
Defect Status Report	Delivered during the Test stage; this daily or weekly report provides stakeholders with a list of all raised defects resulting from the test execution, along with their severity, priority, and latest status.
Final Test Summary Report	Delivered at the end of the Test stage to provide a summary of the testing performed and the resulting final status so that a determination can be made whether or not to exit the Test phase.

Sample Reports for QA Best Practice

Here are some of the details typically found in each of the reports mentioned above:

TESTING STATUS REPORT

This report includes:

- Red/Yellow/Green status on timelines, budget, scope, risks, issues, etc. (for enterprise projects)
- Key activities and dependencies last week/next week
- Testing milestones
- Outline of issues and risks
- Progress on testing activities per plan
    - Applicable data based on the current testing stage:
        – Write: # of tests written
        – Test: # of tests run, # of tests planned, # of tests passed/failed, # of defects
    - Data should be broken down by functional area/geography/location in a way that is meaningful to project leadership

This report should be provided to the Executive Leadership Team, Stakeholders and the Project Team.

TEST EXECUTION REPORT

This report typically includes:

- # of tests run – actual vs. planned
- # of tests - passed vs. failed
- # of defects raised (with any critical defects highlighted)
- Total # of outstanding defects by severity
- Data should be broken down by functional area/geography/location in a way that is meaningful to project leadership

This report should be provided to the Project Team.

DEFECT STATUS REPORT

This report includes:

- # of total defects raised
- # of defects by severity
- # of defects by priority
- # of defects by status (New, Assigned, In Progress, Ready for Re-test, Resolved, Blocked, Closed)
- Burn down chart showing rate of defects being opened vs. closed
- Breakdown of defects by functional area
- Optional: Breakdown of defects by root cause category

This report should be provided to the Executive Leadership Team, Stakeholders and the Project Team (including the Defect Triage Team).

FINAL TEST SUMMARY REPORT

This report includes:

- Full overview of test bed including # of tests by functional area/geography/location and # of automated tests vs manual and coverage
- Total # of test executions
- # of tests run – actual vs. planned
- # of tests - passed vs. failed
- # of total defects raised
- # of defects by severity
- # of defects by priority
- # of defects by status (New, Assigned, In Progress, Ready for Re-test, Resolved, Blocked, Closed)
- Breakdown of defects by functional area
- Optional: Breakdown of defects by root cause category
- Open Issues/Risks

This report should be provided to the Executive Leadership Team, Stakeholders and the Project Team.

Chapter Eleven
Test Automation

Infor WFM solutions are ever-changing, complex integrated solutions designed to fulfill the fundamental contract between the employee and the employer. Continuously testing these solutions is the only way to ensure that you're always in compliance with your employee obligations and commitments. Test automation solutions significantly simplify and accelerate these continuous testing efforts.

TestAssure®

Why Customers Need a Test Automation Solution

The most common reasons that customers look for a test automation solution are to:

- Reduce the manual workload of testing
- Accelerate testing timelines
- Improve testing quality and coverage
- Increase testing capacity and scope
- Reduce the cost of testing business-driven policy changes
- Reduce the cost of manually Regression Testing each WFM release
- Improve business outcomes

Most importantly, a test automation solution helps Infor Workforce Management customers deliver business value faster.

TestAssure.ai - TestAssure's Test Automation Platform

Continuously testing your Infor WFM solution is a significant undertaking that is often underestimated. That's why we built TestAssure.ai, our test automation platform.

TestAssure.ai is a purpose-built testing solution for Infor WFM customers that makes test automation more accessible than ever. The platform delivers pre-built, Infor-specific test automations that require zero code to implement or maintain.

TestAssure.ai expedites the entire test automation process, meaning you can be up and running in a matter of days. This way, you'll waste no time and can quickly run thousands of fully automated test cases in the most high-risk areas of your business.

How TestAssure.ai Can Help You

Here are the common testing challenges that our TestAssure.ai platform can help with:

TESTING CHALLENGE	TESTASSURE.AI
I'm not sure where to start testing. I thought Infor did it for us?	• Testing must be your responsibility as the only subject matter expert (SME) of your company's HR requirements and policies.. • Our Infor WFM Testing Methodology provides a comprehensive QA roadmap including a sample WFM Test Plan, best practices, and test case templates. • Our Infor WFM testing library comes with pre-built automated tests that were created specifically for Infor products.
Managing our test cases in spreadsheets is very confusing and difficult.	• Our test automation platform provides an easy to use interface and searchable repository of all your tests and testing results.
I don't have the capacity to test; we didn't think it would be this much work.	• Our test automation and test library help you significantly lower the demands of testing on your resources. • Our Professional Services Team can deliver a variety of packaged services to lead and/or augment your WFM QA initiatives. And we seamlessly transition back to your team once your capacity constraints are removed. • Our test automation platform offers full support for Functional, Systems Integration, Parallel, UAT, and Regression Testing. • Our Infor WFM Testing Methodology is backed by years of Workforce Management (WFM) and highly-relevant testing experience.
Test writing takes too long.	• Our test automation platform offers global search & replace, mass edit, and templating functionality to help you write, edit and copy tests quickly and easily. These features typically speed up the test writing and maintenance of tests by 10x. • Our Infor WFM testing library means you don't need to create tests from scratch, it comes with pre-built automated tests that were created specifically for Infor products. • Our modern approach to test writing means our tests are easy to understand, maintain, and update. Best of all, they are fully automated.

TESTING CHALLENGE	TESTASSURE.AI
Manual Test execution is very time consuming and error prone.	• The time required to run tests manually is a common bottle-neck in WFM initiatives that often results in project delays. • With manual testing, each test case takes anywhere from 15 min to 60 min to execute. • It takes an average of 4 seconds to run an automated test using our platform. • Human effort is reduced to a button click to launch a series of tests which can run in the background without effecting the system under test. • Test data is automatically managed (created and cleaned up) through the automation engine. • Your team is able to spend their time investigating and resolving defects vs. running tests.
Limited time for multiple testing cycles or re-testing.	• With our test automation platform, re-running tests can be done easily with the click of a button. • Unfortunately, due to the time it takes to run tests manually, defects often mean only running a limited set of tests. Our platform gives you the confidence to run your full suite of Regression Tests with every change. • There is no limit to how many times a test can be automatically re-run.

TestAssure.ai can help if you are implementing a new Workforce Management system, upgrading to a new version, or releasing business-driven changes through-out the year. Our test automation platform will help you minimize the impact of system changes, reduce your risks, and help you move faster with confidence.

Want to
Learn more
about
TestAssure.ai?

Move faster with confidence – discover the benefits of Infor WFM test automation and continuous testing.

Contact us today for a test drive – email us at automation@testassure.com or visit TestAssure.com.

TestAssure®