

# UAT Customer Success

A process of verifying that the solution works for the customer’s expected business outcomes. This is also the last step that we can discover and correct any issues prior to the solution going live. It is vital that each customer understands the importance of UAT, since the cost of resolving issues post go-live in production is greater both financially and in the overall adoption of the new solution.

## Summary

The goal of User Acceptance Testing is to assess if the system can support day-to-day business scenarios and ensure the system is sufficient and correct for business usage.



User Testing serves as another level of validation & quality assurance before bringing the solution online for our Client’s Customers. The purpose of this document is to provide observations of Customer-driven best practices during User Test of the Genesys Solution, including risk mitigation strategies & proposed communications.

During an implementation phase Genesys will execute quality assurance testing before the user acceptance testing (UAT) session.

Customer will be responsible for providing a UAT plan suitable to meet the business objectives of the solution. Genesys will provide input through resources such as this guide with tips and suggestions for creating an effective plan. This guide will contain statements of how best to formulate cases that will not only test system functionality but most importantly how the solution will meet the business needs.

## Testing Methodology Overview

* Review the Testing Methodology Plan document
	+ System Testing performed by Genesys
	+ User Acceptance Testing performed by customer
* Customer to assemble a testing team
	+ Testing Lead – responsible for collecting issues and reporting to Genesys
	+ Testers – combination of agents and Supervisor, Quality Managers, Business Unit members that will perform the test cases
* Customer to write test cases
	+ Test Cases must be based upon the approved requirements documentation
* Customer to execute test cases & submit defects

## UAT Timeline and Defect Resolution Calls

* UAT Timeline will be determined by Project Team
* Typically, the timeline can be extended if the Genesys resources are only partially allocated each day, therefore the proper expectations need to be agreed on for defect resolution turnaround
* Defect resolution call frequency (daily, twice a week, etc.) will be determined by the above items
* Defects (or updates to existing defects) should be entered into the Defect Log prior to the resolution calls
* Some issues that are base product issues (not from configuration or customization) will be raised with the Genesys CARE team and a support incident will be created. These issues may not have an immediate fix.

## Items needed for testing

* Test Workstations (with the clients loaded including patches)
* Test Phones (sample of each type if multiple phone types are used)
* Determine if testers will be using their own User IDs, or generic testing IDs.
* Environment: Determine if testing in an environment that is not the production-to-be environment how/when the promotion will occur and what is tested (to what level) in each environment
* Test Phone Numbers (toll free, DIDs)
* Test Data (if applicable for database dips, web services)
* Test Email addresses (if customer is already using live email addresses, need temporary test email addresses)
* Test chat methodology (test from production customer facing web pages with hidden link, or other)

## Testing Lead Responsibilities

* Testing Lead will collect issues/defects and report in the Defect Log and discuss on the Defect Resolution calls
* Testing Lead will be responsible for reporting percent complete
* Determine if re-testing will occur after issue is resolved or held until multiple issues in a scenario are resolved
* Testing Leads will coordinate with your system administrator(s) for changing test phone numbers to point to different users, routing profiles, make open/closed schedule changes, etc.

## OBSERVATIONS & BEST PRACTICES

The following is provided as a reference & variances are to be expected for each Customer’s unique Professional Services engagement. The Client should:

* Based on Business Requirements, their Severity Levels & Impact, prioritize User Test Cases.
* Develop a Staffing Management Plan for Testers, including allocation. For instance, if Testers are assigned to support the existing Production environment & move through User Test Cases, a risk to the schedule should be recognized as Production issues will always take precedence.
* Ensure that Testers are familiar with authorized collateral, such as Business Requirements & Technical Design. In addition to the publication of these documents, conduct a live walkthrough for question & answer.
* Have a clear definition of report categories for Testers.

|  |  |
| --- | --- |
| Defect | A flaw within the solution that does not conform to requirements or makes it unfit for use, e.g. blue screen, that results in resolution via a design, configuration, or product change. |
| Enhancement | A new feature or function that was not identified within the Business Requirements, Technical Design, or Genesys Technical Documentation for the product. These would result in negotiations amongst the Project Leadership Team to determine appropriate next steps within the Change Management process. |
| Issue | A conflict that arises due to a delta in what the user expects vs. a business requirement. For instance, the business requirement requests the widget be green, the user expects it to be Kelly green & the consultant configures it to be forest green. These would result in negotiations amongst the Project Leadership Team to determine the appropriate next steps within the Change Management process. |

Have a clear definition of defect severity levels.

|  |  |
| --- | --- |
| 1-Critical | Causes a Severe impact on business operations of end Customer, e.g. calls cannot be completed or critical business processes are disabled. Alternatively, causes a severe impact on business operations due to the accumulated impact on multiple Customers. Continuous or near continuous interruption of service. No workaround available. |
| 2-High | Causes a non-critical impact on business operations of end Customer, e.g. call processing altered in such a way as to degrade service quality or handling of business data. Alternatively, causes a non-critical impact on business operations due to the accumulated impact on multiple Customers. Intermittent disruption of service. No stable workaround available. |
| 3-Medium | Causes a minor impact on business operations of end Customer, e.g. minimal degradation of call processing or handling of call data. Alternatively, causes a minor impact on business due to the accumulated impact on multiple Customers. |
| 4-Low | Causes little or no impact on business operations of end user. Alternatively, causes little or no impact on business operations due to the accumulated impact on multiple Customers. |

* Based on the Environment, identify User Test Cases where modifications to configurations are required in order to isolate those instances. This will help avoid False Failures being reported.
* Create Roles & Responsibilities Chart (RACI), e.g. identify who is responsible for making the changes in each environment.
* Determine how best to segment User Test Cases. For example, do not conduct “happy path” testing in parallel to failover testing.
* Establish a testing schedule to reduce the risk of testers stepping on each other creating False Failures.
* Customer environments with 24 hour operations will require a follow-the-sun type staffing model, shift management, technical leads, open operational technical conference bridges, & shift transition handoffs for continuity.
* Automate where possible. Contact your Genesys Account or Project Team for more information on Empirix Testing as a Service (GTaaS).
* Ensure the process & responsible parties for making changes in each environment is known & communicated. Manual change control environment requires a change document, a specific document owner, & tight document version control. Contact your Genesys Account or Project Team for more information on Spotlight powered by blackchair to improve change management. Automated change management reduces workload & mistakes.
* Leverage a centralized User Test Case, Issue & Defect Tracking Tool
* For efficiencies in both resolution & communications, identify team members who can act as a “human workflow” to aide in:
	+ Logging the appropriate symptoms & adequate call or workflow details to facilitate troubleshooting
	+ Routing to the appropriate subject matter expert (SME)
	+ Timely follow-ups
	+ Tracking
	+ Various communication levels, e.g. Executive vs. Project Team vs. SMEs
	+ Root cause documentation
* Prioritize Defect Resolution for the Consultants or default to the Use Test Case priority.
* Establish environment change windows; on the adverse, determine if Environment Moratoriums are required for certain time periods. Between the Test & Deploy activities, an Environment Moratorium including Change Management, is highly recommended to preserve the quality of the solution:
	+ Ensure system integrity & maintain the system assured/certified via System Integration Testing (SIT), User Test, & Performance Testing, if applicable
	+ Reduce the risk of negative impacts to performance or functionality assured/certified via SIT, User Test, & Performance Testing
	+ Support both Performance Testing & Production for readiness & stabilization
	+ Reduce the risk of rework implications, including monetary, resource, & system impacts
	+ Manual change control environment requires a change document, a specific document owner, & tight document version control.

Sample template available upon request.

* Determine the level of rigor for Environment Change Control, including ad hoc changes, regression testing & rollback plans.
* Identify non-Project sanctioned environment changes that could have a downstream impact, e.g. Corporate/Information Technology deployed operating system patch updates.
* Develop a cadence for communicating progress to identified recipients & quantify where possible, e.g. percent complete, number of impacted User Test Cases, start & finish variances in business days, estimate to complete.
* For Governance, identify User Test & Go-Live approvers including proxies for out of office schedules.
* Do not reduce required testing time in an attempt to reduce the overall project timeline. This will result in overlapping efforts, conflicting activities, general inefficiencies, quality concerns, & additional remediation time, which will directly offset the perceived timeline improvement.
* Take the challenge to “break it”, but be solution-focused to resolve issues before Go-Live. And have a little bit of fun in the process!

## How to test with the software

* Review the full training PPT
* Place Intercom/Internal test calls to get familiar with behavior before executing test cases
* For testing queue calls, place test calls to the test numbers associated with the routing profile and queue you are testing.
* If limited test phone numbers, your Administrator may have to move a number between routing profiles before that profile can be tested
* Ensure emails and/or chats are distributed to the appropriate queues.

## UAT Test Cases

* Test cases should cover all functionality of the system:
	+ IVR/prompts and flow, open hours closed hours
	+ ACD queue behavior
	+ Queue settings
	+ Agent settings
	+ Agent’s Desktop
	+ Reporting (out of the box reports, custom reports if applicable)
	+ Phone device
	+ Call quality
	+ Multiple locations/sites
	+ Integrations
	+ Emails
	+ Chats
	+ Business scenarios that agents may be performing outside of/in addition to telephony related functionality
	+ Decide if testers will be divided up to test each component separately, or if testers will test a business unit/departments scenario from start to finish, then move to the next scenario.

# Example Business Outcome (Use Case) to Test Case

The Business Outcome (Use Case) defines desired system behavior from the perspective of a user. That user may be the end user (the person calling into an IVR application), an Agent (the person receiving a routed call), or a Supervisor (someone monitoring the activities of the Agents as they provide service to the customers), or some other actor interacting with the system under test.

 An example of Business Outcome might be: End user can change their phone number within the Self-Service IVR

Writing test cases for these requirements involves breaking them down into discreet scenarios that represent different views of the requirement.

Test Case: User changes phone number to international number

* + User calls into IVR
		- Main Menu plays
	+ User selects 1 for account maintenance
		- Account Maintenance menu plays
	+ User selects 3 for personal information
		- Personal Information menu plays
	+ User selects 2 for phone number
		- Phone Number menu plays
	+ User selects 2 to change phone number
	+ User enters +34 454 659134
		- System responds with new number and asks for confirmation
	+ User presses ‘1’ to confirm change
	+ User hangs up

 A number of test case can be derived from this requirement – different countries, valid and invalid phone numbers (does the system check for a 10-digit number for U.S. numbers?), user abandons call after making the change but before confirming, etc.

# PureConnect Example Cases

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Responsible Tester** | **Category**  | **Business Scenario**  | **Steps** | **Expected Results**  | **Actual Result** | **Comment**  | **Interaction ID**  | **Est. Test Duration (mins)** |
| BPT\_01 | John Doe | Billing | Customer canceling subscription | Caller dials billing number during open hours of the day | Call hears proper greeting and routes to queue |   |   |   | 5min |
|   |   |   |   | Billing with subscription skill receives the call | Proper agent can pick-up call and have two-way audio |   |   |   |   |
|   |   |   |   | The Caller explains they are a business partner wanting to cancel | The agent can see transfer button and select for receiving warm transfer option |   |   |   |   |
|   |   |   |   | Agent warm transfers to business partner queue | agent is routed to business partner rep and has two-way audio |   |   |   |   |
|   |   |   |   | Agent completes warm transfer | Agent is dropped from call with caller and business partner rep talking with two-way audio |   |   |   |   |
|   |   |   |   | Agent dispositions call | pop-up for disposition displays for agent and can select proper transfer message |   |   |   |   |
|   |   |   |   | Rep agent accesses CRM to make cancelation request | Rep can navigate between Genesys client and CRM window for input |   |   |   |   |
|   |   |   |   | Rep completes call | pop-up for disposition displays for Rep and can select proper transfer message |   |   |   |   |
|   |   |   |   | Rep continues with after call work | Follow-up status is auto-selected for Rep with 30 seconds of time before becoming available again |   |   |   |   |

# PureEngage Example Cases

| **Test Name/Reference** | **Tester Name and Test login** | **Description** - Describes the purpose of the test, is also often used for any prerequisites for the test | **BRD/CRD Reference**This cross-references directly with a specific requirement from the BRD/CRD | **Test Step Name** | **Description (Step)** - this is the action undertaken in the step to achieve the expected result | **Expected Results** - The results you are trying to achieve within the step | **Report Category:**This determines the category of the results options are:· Defect· Enhancement· Issue | Report **Priority**This determines the priority of the tests options are:· 1-Critical· 2-High· 3-Medium· 4-Low |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 05.19 – Adding Future Exceptions | BillyBobUser1 | To demonstrate the ability to edit shifts e.g.: add exceptions to single & multiple attendees before schedule generation and after generation | Schedule Module - Test 19 | 1 | Log into Web Supervisor | User is logged in |   |   |
|   |   |   |   | 2 | (A) To setup Calendar item - Navigate to <Calendar> from the menu pane, select two agents | User will be presented with the Agents option. |   |   |
|   |   |   |   | 3 | Select each agent, highlight relevant dates, select 'get data' button | The 'add calendar item' icon will now be active and once selected the 'add calendar item' box will present. |   |   |
|   |   |   |   | 4 | Select 'add calendar item' box and choose a 'exception' item leaving the requested state as 'preferred'. Select next | User is presented with exception options |   |   |
|   |   |   |   | 5 | Select an Exception type and select finish | User will be presented with the new calendar items for the 2 individual agents.  |   |   |
|   |   |   |   | 6 | Highlight all exceptions items with a status of 'preferred' and select the 'grant' icon  | The user will note the status change to 'Granted' |   |   |
|   |   |   |   | 7 | Build Check - Navigate to Schedule module, <schedule> Scenario, include the two agents & date range that includes the date that the exception was added to the calendar | Select a scenario with the same 2 teams as above |   |   |
|   |   |   |   | 8 | Select Schedule Build Wizard from 'Actions' menu or top right work pane | Schedule Build Wizard box will present |   |   |
|   |   |   |   | 9 | Select 'next' check sites have been selected, then 'finish'. | Schedule builder will have pulled data from the Calendar and placed events on schedule |   |   |
|   |   |   |   | 10 | (B) Navigate to the Master schedule/intra-day view | Master schedule will open |   |   |
|   |   |   |   | 11 | Select team/agents and activities from the Objects window | Agents schedules populate |   |   |
|   |   |   |   | 12 | Add an exception (non-meeting) to an agent’s schedule | Exception can be added to agent schedule |   |   |