



SAP Quality Assurance Services

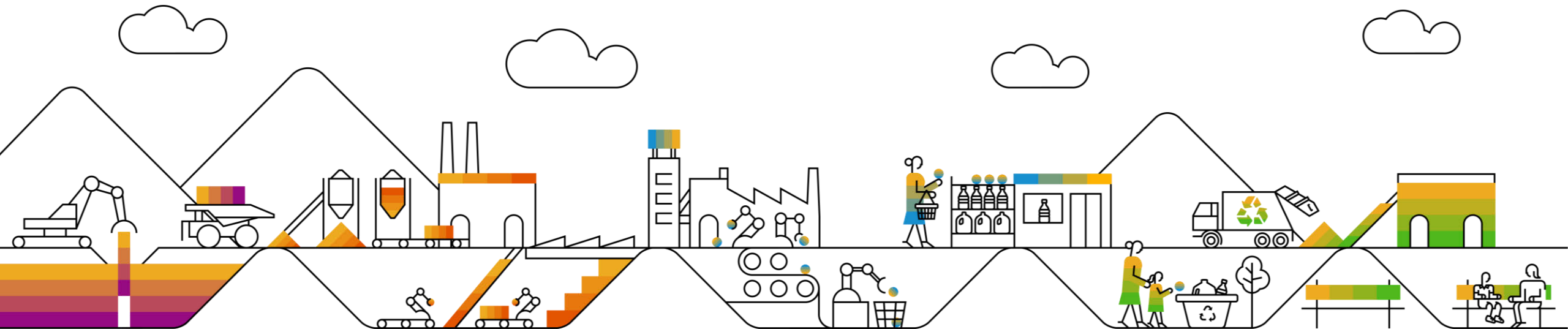
Société de l'assurance automobile du Québec (SAAQ) CASA Project Testing Review – Rev1
November 12, 2020

Contents

- Detailed report
 - Test Management and Strategy
 - Requirements Management
 - UAT Strategy and Current Hotspots
 - Defect Management Lifecycle
 - Test Case Design and Test Execution
 - Common Pain Points in Technical and Development Teams
 - Preparation for Data Governance
 - Test Tool Utilization
 - Common Incidents and Frequency of Incidents During Postproduction
 - Appendix



Executive summary



Overview content

Special notations

This report represents the results of the Quality Assurance Solution Design Review conducted October 26–28, 2020 for the Realize phase of the implementation of SAP ██████████ at Société de l'assurance automobile du Québec (SAAQ).

This report reflects a point-in-time analysis of the current state and future deliverable readiness. This Testing Review focuses on the following areas: Test management and strategy, requirements management, UAT strategy and current hotspots, defect management lifecycle, test case design and test execution, common pain points in technical and development teams, preparation for data governance, test tool utilization, and common incidents and frequency of incidents during postproduction.

The findings and recommendations result from information provided to SAP during Q&A sessions with numerous SAAQ, IBM, and SAP resources.

This report is forward-looking in that it focuses on the actions necessary for future operational success and reserves commentary on project history only for contextual understanding.

As part of this Review, the assessment team was made aware of pain points and concern areas. It was important to understand both in terms of the requirements and functionality needed by the CASA project. This understanding provided additional information and awareness. But as a Design Review, this exercise did not try to resolve all identified pain points and concerns.

This document contains SAP proprietary and confidential information and is provided in respect to SAAQ's implementation of SAP software. This document may not be disclosed or redistributed in whole or in part to any third party without the written consent of SAP.

Overview content

Definitions and indicators

Risk indicator definition



Low: Minor impact on program success and unlikely to adversely affect the implementation or business operations.



Medium: May produce or contribute to a negative outcome during the implementation, including extended Blueprint or Realization lifecycles.



High: A significant likelihood of causing or contributing to a negative impact during implementation or at go-live. A critical topic to be addressed that requires investigation, follow-up, and response.



NA – Topic not reviewed: This topic was either inappropriate for review at this stage of the project or insufficient information was available to review the topic at the proper level of detail.

Executive summary

Overall program risk



Project has **medium** risk profile.

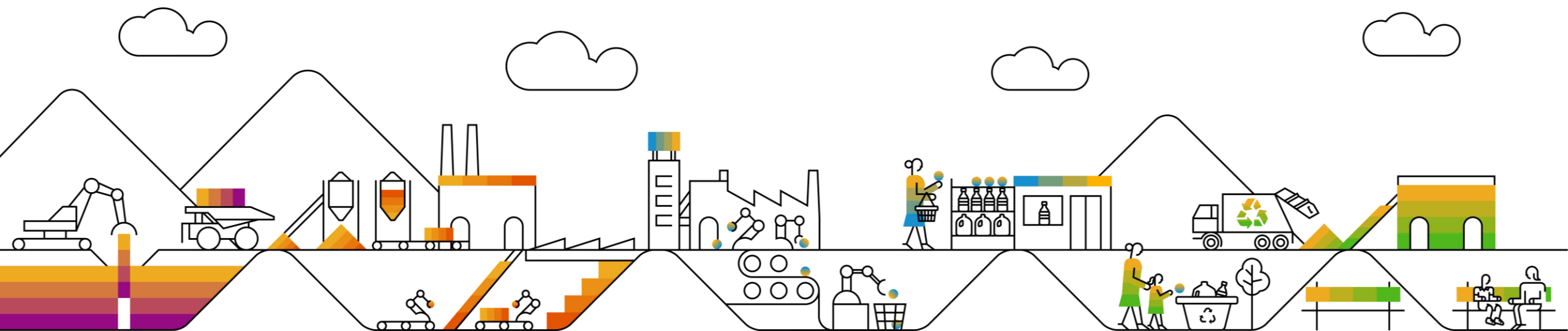
The following are the most important, but not all key findings. It is also important to note that many items identified as medium can easily turn high if not addressed in time.

- Interface readiness – Need a detailed plan for ensuring that the critical interfaces are developed and ready for IT3, IT4, and IT5.
- The performance test plan is incomplete and missing core elements such as concurrent users and hourly KPIs.
- A requirements traceability matrix is missing. It ensures that all the business processes are properly tested, and that no missing requirements are uncovered during Hypercare.
- Security testing is currently not in scope for IT3 and is planned to be executed during the final integration testing phase – IT5. Ensure that business roles are embedded in the test cases and executed earlier in the test cycles.

Executive summary

Solution Area	Risk Indicator	Findings (Low, Medium, and High)			Total Findings
Test management and strategy		0	2	2	4
Requirements management		0	2	0	2
UAT strategy and current hotspots		2	0	0	2
Defect management lifecycle		0	3	0	3
Test case design and test execution		2	0	0	0
Common pain points in technical and development teams		0	1	0	1
Preparation for data governance		0	1	1	1
Test tool utilization		1	1	0	2
Common incidents and frequency of incidents during postproduction		0	1	0	1
Total Findings		5	12	3	19

Detailed report



Overview content

Testing scorecard (1/2)

Review of Testing Items	Status Indicator
Test management and strategy	
Requirements management	
UAT strategy and current hotspots	
Defect management lifecycle	
Test case design and test execution	
Common pain points in technical and development teams	

Overview content

Testing scorecard (2/2)

Review of testing items	Status Indicator
Preparation for data governance	
Test tool utilization	
Common incidents and frequency of incidents during postproduction	

Test management and strategy

(1/4)



Finding, impact, and recommendation

- ❑ **Finding 1:** During the workshop sessions discussions indicated that the interfaces necessary for executing integration testing cycles IT3, IT4, and IT5 will not be ready in time to support each of the testing cycles. The complexity and size of the L2 solution will require that the end-to-end business processes are executed with all integrations in place.
- ❑ **Impact 1:** Without having the interfaces developed and ready for testing as part of the end-to-end business processes will create gaps in the solution and create a risk that the business does not work as expected after the go-live.
- ❑ **Recommendation 1:** Introduce a risk-based testing approach in which the business processes are categorized by the most critical down to the least. Ensure that the interface development team knows what needs to be developed first and build the test plan to front-load the testing cycle with the most-critical processes.

Test management and strategy

(2/4)



Finding, impact, and recommendation

- ❑ **Finding 2:** The performance test strategy does not list all the necessary elements and KPIs that are important for executing proper performance tests. It is missing elements such as identification of the critical end-to-end business processes, hourly document throughputs, critical batch job timings, and service center locations that may be prone to network latency problems.
- ❑ **Impact 2:** Without a clearly defined performance test strategy and KPIs, the performance test findings will not reflect reality, and critical bottlenecks may not be addressed before going live.
- ❑ **Recommendation 2:** Identify elements such as:
 - The size of the performance testing environment compared to the production system (necessary for proper extrapolation of results)
 - The critical batch jobs with expected runtimes
 - The hourly throughput of documents created (average and peak) for the critical transactions
 - A plan for ensuring that the testing is representative of the day-in-the-life business processes and not simply unit-testing transaction codes with concurrent users

Test management and strategy

(3/4)



Finding, impact, and recommendation

- ❑ **Finding 3:** The current data strategy is unclear about the identification of master and transactional data that will drive all the test scenarios. The availability and quality of test data drives the test cases, and if the testing team is unclear about which data set to use, confusion will result when attempting to run the test cases.
- ❑ **Impact 3:** Test velocity will be negatively affected if testers are unclear about which dataset to use or if incorrect data is used for testing and creates erroneous defects.
- ❑ **Recommendation 3:** Document the data strategy to identify the specific datasets that will drive manual testing, functional automation testing, and performance testing. This strategy becomes a part of the entrance criteria for the integration testing (IT3, IT4, and IT5) cycles. Publish the golden datasheet to all members of the test teams to ensure that cross-teams do not consume data reserved for other process teams.

Test management and strategy

(4/4)



Finding, impact, and recommendation

- ❑ **Finding 4:** Security testing is not currently scheduled to be conducted until the final phase of integration testing, test cycle IT5. Workshop discussions also indicated that many security-related issues were uncovered during hypercare of the previous L1 release.
- ❑ **Impact 4:** Improper and/or incomplete testing of security roles and permissions will have a detrimental outcome during the go-live and lead to show-stopper defects.
- ❑ **Recommendation 4:** Plan to execute security testing in IT4 focusing on the critical and high-priority business processes. Embed the necessary security roles into the test cases. For example, step 1 of a test case can have the following terminology: "Login to the CRM system as a customer service supervisor." During IT5, security testing can be completed for the medium- and low-priority business processes.

Requirements management

(1/2)



Finding, impact, and recommendation

- ❑ **Finding 1:** The requirements traceability matrix that contains the mapping of the high-level business processes to the lower-level end-to-end test cases and RICEFW objects is incomplete.
- ❑ **Impact 1:** Without a clearly documented requirements traceability matrix, a gap in understanding all the business processes and their linkage to the lower-level test cases will develop. In such a situation, a business requirement can be missed and uncovered only after the solution has gone live. Stakeholder sign-off cannot be achieved if certain business processes have not been tested and validated.
- ❑ **Recommendation 1:** Engage with key stakeholders to finalize the requirements traceability matrix before the start of integration testing using a top-down approach to identify the business processes first. Then map them to the unit-test cases and finally, the RICEFW objects. This approach ensures proper coverage of the end-to-end business processes and that all business scenarios were properly tested before going live.

Requirements management

(2/2)



Finding, impact, and recommendation

- ❑ **Finding 2:** The approach to handling new business requirements that may be introduced during the execution of the testing cycles is unclear. When a new requirement is introduced while testing is executed in a test cycle, the new functionality and code can break the test cases that have already been executed.
- ❑ **Impact 2:** Introducing new requirements during a test cycle will create issues in terms of fundamental changes to processes that have already been tested and passed and a need for deeper regression testing to ensure that new functionality does not break existing functionality.
- ❑ **Recommendation 2:** Create a formalized committee, such as a change control board (CCB), to review all new changes being introduced to the solution after testing has initiated. This approach helps teams determine the stability of the solution and strengthens governance by giving stakeholders an opportunity to weigh in on the introduction of the new changes.

UAT strategy and current hotspots

(1/2)



Finding, impact, and recommendation

- ❑ **Finding 1:** Workshop sessions discussed plans to execute user-acceptance testing (UAT) in parallel with Final Integration Testing (FIT). The FIT cycle occurs approximately 10 weeks after the final phase of integration testing which is IT5. Additionally, there will be a Day in the Life test cycle which occurs after IT5 which is planned to be executed prior to UAT.
- ❑ **Impact 1:** None.
- ❑ **Recommendation 1:** Continue with the current approach. Typically, executing the user-acceptance testing in parallel with an integration test cycle creates a risk in which end users will be uncovering anomalies in the system. A first pass of the solution needs to be executed by the testing team prior to handing it over to the testers in UAT. Since SAAQ will be executing IT5 and a Day in the Life test cycle prior to entering UAT, the probability that the end users will discover new anomalies is greatly minimized.

UAT strategy and current hotspots

(2/2)



Finding, impact, and recommendation

- ❑ **Finding 2:** Beginning in January 2021, a selected group of superusers will join the project with the intention of being trained in SAP software. The plan calls for the superusers to participate in the first cycle of integration testing (IT3), which is currently scheduled to begin April 2021.
- ❑ **Impact 2:** By engaging with the superusers in January, the team has sufficient time to receive the proper training to participate in the subsequent integration testing cycles and UAT.
- ❑ **Recommendation 2:** Employ a buddy-system approach for UAT in which the business users that don't have much expertise with SAP software can be paired with a superuser to ensure smooth testing and minimal defects. The role of the super user is key for an implementation of this size and complexity. During integration testing, these users can assist with business process validation as well as test case execution and defect reporting.

Defect management lifecycle

(1/3)



Finding, impact, and recommendation

- ❑ **Finding 1:** The defect lifecycle process does not define a specific service level agreement (SLA) for a resolution time of defects. During previous test cycles and releases, the due date of defects was the primary indicator that the project used to ensure that development teams closed defects on time.
- ❑ **Impact 1:** By not establishing clearly defined SLAs for critical P1 and P2 defects, the development teams may not close the defects in a timely manner, which will then block testing and jeopardize the timely exit of integration testing on the agreed-upon project schedule.
- ❑ **Recommendation 1:** Institute a formalized expected resolution time with development teams based on defect priority. For example, a critical P1 defect has a target resolution time of eight hours. This proactive approach keeps development teams focused on resolving true show-stoppers so that test velocity is maintained.

Defect management lifecycle

(2/3)



Finding, impact, and recommendation

- ❑ **Finding 2:** As part of the current defect strategy, no defect-blocking report displays the number of test cases blocked by certain defects.
- ❑ **Impact 2:** Without the ability to generate a defect-blocking report, test velocity will be negatively affected if the number of open defects blocks the execution of many test cases. The development team will need to focus its efforts on resolving the defects that affect the highest number of test cases so that testing can continue.
- ❑ **Recommendation 2:** Generate this type of report manually, given that the current Focused Build for SAP Solution Manager solution cannot generate it. The ability to identify which defects block the greatest amount of test cases is an excellent way to ensure that the pace of testing is maintained and that the development teams focus their efforts on true blockers for the testing team.

Defect management lifecycle

(3/3)



Finding, impact, and recommendation

- ❑ **Finding 3:** No dedicated defect manager is assigned to the project. The L2 solution is extremely large and complex, and as test cases are executed, an opportunity exists to identify defects. A dedicated defect manager can oversee every aspect of the defect to ensure that the issues are addressed and resolved and allow the testers to execute their test cases without delay.
- ❑ **Impact 3:** Without a dedicated defect manager to review and prioritize the defect backlog, it will be very difficult to manage and track the timely completion of each integration test cycle.
- ❑ **Recommendation 3:** Establish a dedicated resource to manage the defects across all the tribes. This resource will escalate defects that have expired SLAs and work with the development teams to prioritize resolution of defects that block the greatest amount of testing. This resource will also oversee the retesting of defects to ensure that the development teams are providing quality fixes to the problems being identified during testing.

Test case design and test execution

(1/2)



Finding, impact, and recommendation

- ❑ **Finding 1:** The workshops saw a good mix of positive and negative test cases written to support the integration testing cycles. The quality of the solution is greatly increased when all aspects of test cases are executed.
- ❑ **Impact 1:** None.
- ❑ **Recommendation 1:** Continue to use negative testing in all cycles of integration testing (IT3, IT4, and IT5) to ensure that a particular test step fails when incorrect data is used.

Test case design and test execution

(2/2)



Finding, impact, and recommendation

- ❑ **Finding 2:** The workshop sessions observed that the testing team members are knowledgeable about the differences between master and transactional data. This distinction is important because certain elements of test cases cannot be executed unless the proper master data is set up and configured as a prerequisite. During end-to-end testing, the transactional data created in the beginning of a process is used by several teams downstream to ensure that the entire business process works as expected.
- ❑ **Impact 2:** None.
- ❑ **Recommendation 2:** Continue with the current approach. As new test team members are introduced, ensure that they are trained in the importance of the types of test data needed for properly executing the test cases. Also ensure that new team members are not allowed to create defects unless they have been reviewed by senior members of the tribe.

Common pain points in technical and development teams

(1/1)



Finding, impact, and recommendation

- ❑ **Finding 1:** Transport requests into the systems are not harmonized with and between other quality assurance and testing systems, and large volumes of errors (such as Return Code 12) are continually reported.
- ❑ **Impact 1:** Consistent errors in transports result in object integrity and dependency failures along with delays and rework and indicate development quality issues. A lack of understanding the root cause of the issues will create lost time and effort among the different team members in their attempts to triage and resolve the problems as they are manifested.
- ❑ **Recommendation 1:** Engage with the SAP MaxAttention team to understand all the errors being encountered and create an action plan to address each of the error codes.

Preparation for data governance

(1/2)



Finding, impact, and recommendation

- ❑ **Finding 1:** The workshop discovered that unplanned, ad hoc requests are initiated from the test team, requesting data to be generated to support the execution of the test cases.
- ❑ **Impact 1:** Unplanned requests to the data team mean that the team cannot create the test data necessary to support the test cases. This situation will negatively affect testing velocity because testers will be unable to achieve the daily test execution targets.
- ❑ **Recommendation 1:** Create a weekly test plan that identifies the test scenarios and test cases. Publish the plan to the data team to ensure that it is aware of what scenarios are being tested ahead of time. This proactive approach will ensure that the data is prepared before executing the test cases and that the data team can manage its workload more effectively.

Preparation for data governance

(2/2)



Finding, impact, and recommendation

- ❑ **Finding 2:** A lack of centralization was noted among the different tribes regarding the parameters needed to create the proper test data. Certain tribes are extremely knowledgeable and can proactively request the data that is necessary for their tests. Other tribes are unclear about the specific data needed for their scenarios.
- ❑ **Impact 2:** A lack of centralization among the tribes will create a situation in which the data team is continually attempting to create data multiple times throughout the day because certain tribes do not know the specific parameters. This situation results in a loss of productivity for all the teams.
- ❑ **Recommendation 2:** Create a data playbook that contains all the parameters needed for testing each of the testing scenarios. Publish the playbook to all the tribes so that they are aware of the correct type of data needed for each of the scenarios. This proactive approach will ensure that the data team creates the correct data the first time that the request is made to the team.

Test tool utilization

(1/2)



Finding, impact, and recommendation

- ❑ **Finding 1:** The data team is using a functional test automation tool, [REDACTED] Certify, to create the data for the test scenarios.
- ❑ **Impact 1:** Creating large volumes of test data using a functional test automation tool severely limits the amount of data that can be generated. The [REDACTED] Certify tool is not a load-generation tool.
- ❑ **Recommendation 1:** Plan to use the SAP LoadRunner application by Micro Focus or an equivalent dedicated performance-testing tool. By using the data playbook, several automated scripts can be created and executed in parallel, which will create hundreds of different types of data objects that can then be consumed by the various tribes in their test cases.

Test tool utilization

(2/2)



Finding, impact, and recommendation

- ❑ **Finding 2:** The project is using SAP Solution Manager and Focused Build for SAP Solution Manager to manage test case execution and defect tracking.
- ❑ **Impact 2:** None.
- ❑ **Recommendation 2:** Continue to use SAP Solution Manager as the source of truth to document all business requirements and solution documentation. Then use Focused Build for SAP Solution Manager so that the project can create a detailed set of work packages that link to all the business requirements. This approach ensures that all business requirements will be fully tested as part of integration testing. Managing test cases and defects in one centralized tool will increase test velocity and ensure that defects are correctly cataloged.

Common incidents and frequency of incidents during postproduction

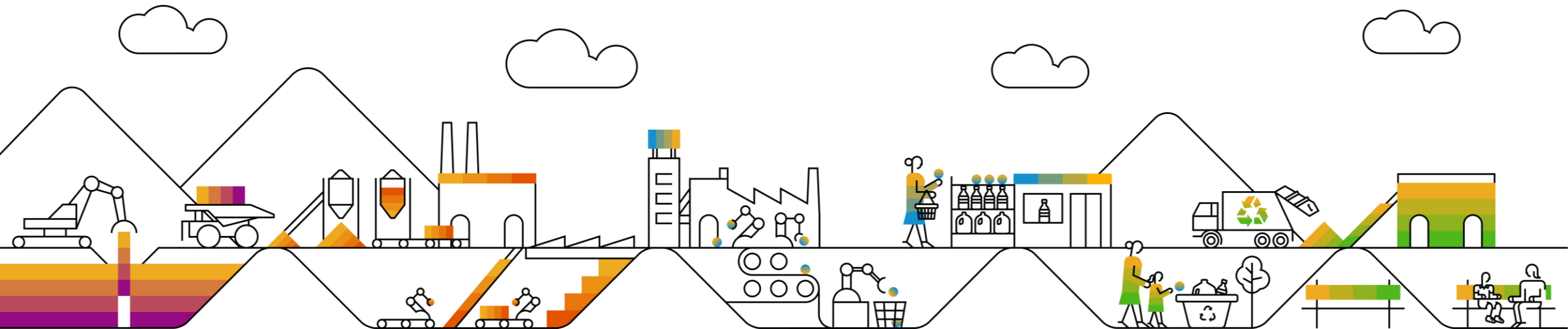
(1/1)



Finding, impact, and recommendation

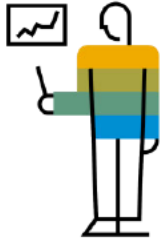
- ❑ **Finding 1:** At this time, no regression testing plan is defined. The L2 solution will contain heavy RICEF developments and integrations, and a risk exists that the L2 solution will conflict with the existing business processes at a code and configuration level without a clearly defined regression testing plan.
- ❑ **Impact 1:** The lack of a clearly defined regression testing plan will have a negative affect on the project because the opportunity exists for new code to break existing code. At go-live, existing business processes will not function as expected in some situations.
- ❑ **Recommendation 1:** Introduce a risk-based testing approach in which the critical business processes are identified and cataloged. Test cases can then be built and eventually automated using ██████████ Certify. By executing the critical processes regularly, the project can have confidence that RICEF development does not negatively affect core business processes.

Appendix



Appendix

Next steps



Follow-Up Action Items

Action Item

- Response Plan Due Date: Nov 27, 2020

Appendix

Documents reviewed

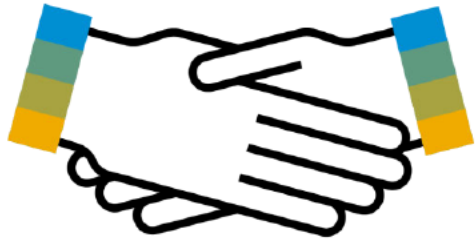
Name	Name
Guide – Suivi des anomalies de tests.pptx	TI 2 - Rencontre de lancement V3.pptx
Guide – Suivi du statut de l'exécution des essais .pptx	TI2 - Rapport de suivi - exécution - 2020-05-27.pptx
■■■■■FOR_■■■■■-gestion des anomalies_EN.pptx	Rédaction TU (1).pptx
L2 - Strategie des Essais_V6.docx	Exécution TU (2).pptx
Présentation suivi et gestion des anomalies.pptx	Environnements, livraison et essais.pptx
Processus de réalisation bout-en-bout - EN.pptx	CSV2.600_TU001_Créer nouvelle demande remb_création.xlsx
Stratégie - Type essais V2.pptx	GRU9.4_TU008_Créer Plage RDV_Ressource Classe 6D GRU244.xlsx

Appendix

Interview participants

Interviewee	Role
████████████████████	SAAQ Test Lead
████████████████████	Alliance Testing Lead
████████████████████	SAAQ Strategic Adviser
████████████████████	Deployment Coordinator
████████████████████	SU Training and Integration
████████████████████	SME Control and Monitoring of People
████████████████████	SME Vehicle Control and Monitoring
████████████████████	Alliance Cutover Lead
████████████████████	Testing Team – Usage of Testing Tools
████████████████████	Testing Team – Usage of Testing Tools
████████████████████	Testing Team – Automation of Data Creation Using ██████████
████████████████████	SAAQ Leader SAP Business Solution Support and Maintenance

Thank you



Contact information:

Jim Orfanakos
Quality Director

Customer Success | Quality Management



Follow us



www.sap.com/contactsap

© 2020 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platforms, directions, and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

See www.sap.com/copyright for additional trademark information and notices.