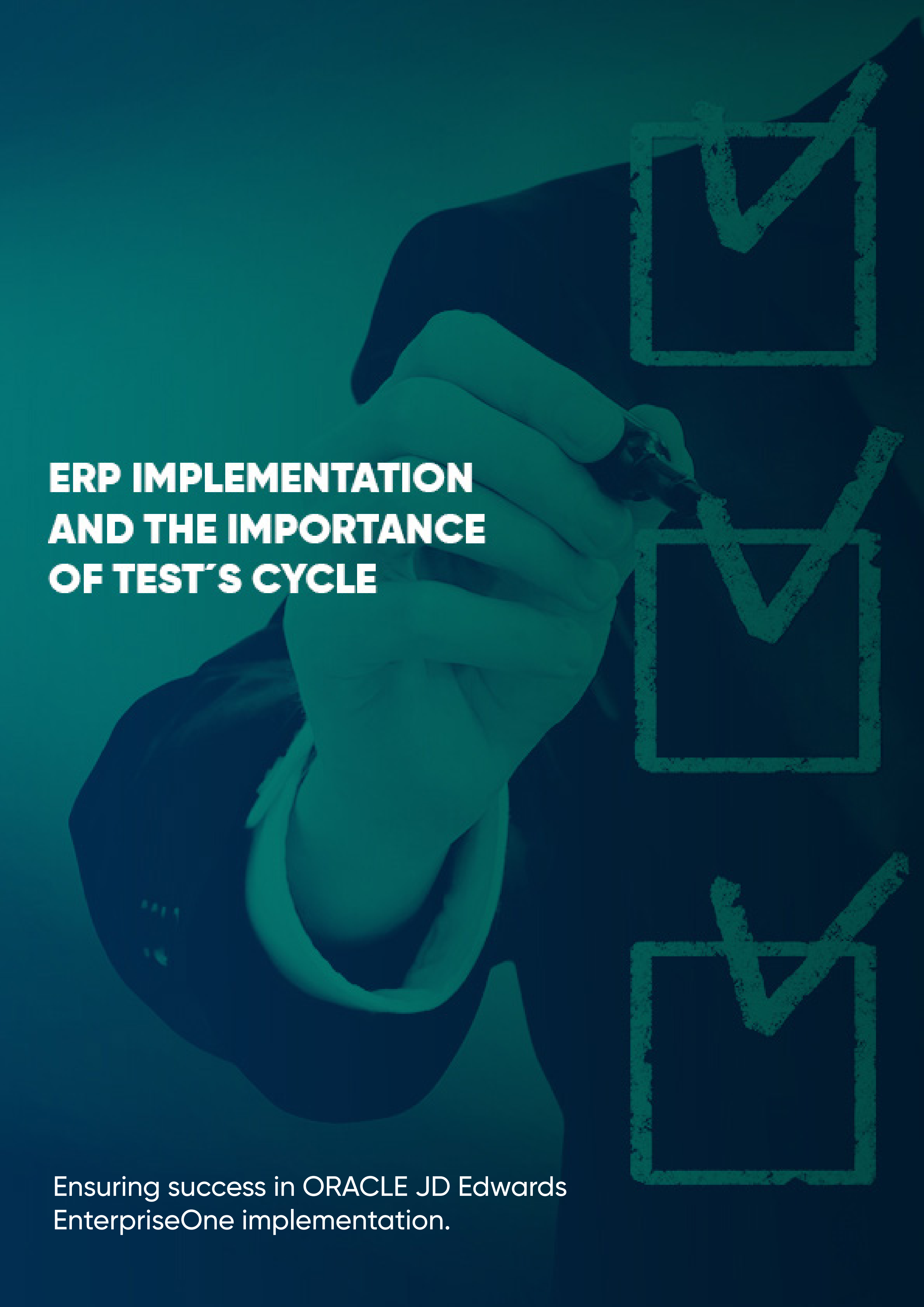


ERP IMPLEMENTATION AND THE IMPORTANCE OF TEST´S CYCLE

Ensuring success in ORACLE JD Edwards EnterpriseOne implementation.

SUMMARY

1. ERP METHODOLOGY IMPLEMENTATION

An ERP Implementation decision should be treated as a project, such as its importance within the organization.

It should be planned in detail and should consider activities that can be tracked, starting with the choice of competitors until the GO LIVE.

After its GO LIVE, another concern must be present in the day-to-day of all C Levels: how to keep the ERP version up to date, incorporating new features and ensuring the intelligent use of all system capacity.

Once your vendor is chosen, the ERP implementation project must go through the planning, execution, and production phase. In all these phases, a subject that can not go unnoticed is the quality of the tests to be performed and possible tools that can assist implementation team in this task.

Planning: At this time, there are no changes to your company's routine. This is a phase of information gathering, where the Project Plan should be developed. Consider at this stage, among other important aspects, what will be the test methodology to be applied, the test period and resources that will be involved in this period, how many test cycles need to be performed and especially if there is a significant productivity gain when choosing some market tool to automate repetitive testing in more than one cycle.

Execution: This is the phase where the previously developed Project Plan is put into practice. It includes software installation, configuration, training and testing of the new system.

Be suspicious of Project Plans with a succinct test phase that do not involve key users or that simplify this phase minimizing their risks. Based on the test´s result implementation partner will seek system´s acceptance: how its functionalities and usability are experienced by the users.

Engaging users in this approval is a critical success factor for your project.

There are usually two phases of testing: the first one where the tests are done individually and the customizations and configurations are tested one by one; and the second one, known as integrated testing, where all process flow - for example, from the entry of a purchase requisition thru the payment to the supplier - is tested. During the integrated tests the boundaries between the different business areas of your company are transposed.

Production: once the system has been accepted, this phase is characterized by the ERP's use in the day to day activities of your company. The problems and challenges faced during this step will be inversely proportional to the quality and comprehensiveness of the tests performed in the execution stage.

Once in production, ensure proper support for ERP usage, so it will continue to be relevant to your business. This support should be used to allow users to ask questions about the use of the system, to promote eventual parameterization corrections in case of internal process changes and to implement improvements to the initial scope of implementation.

In a more advanced degree of support, constant updating of the implemented version should be a goal to be pursued by the company. In these cases, there are tools that can help in the analysis of customized functionalities, pointing out which points have been modified and must be retested again.

Once again, a Project Plan for Updates must be prepared and the testing methodology followed to ensure the continued success of your ERP.

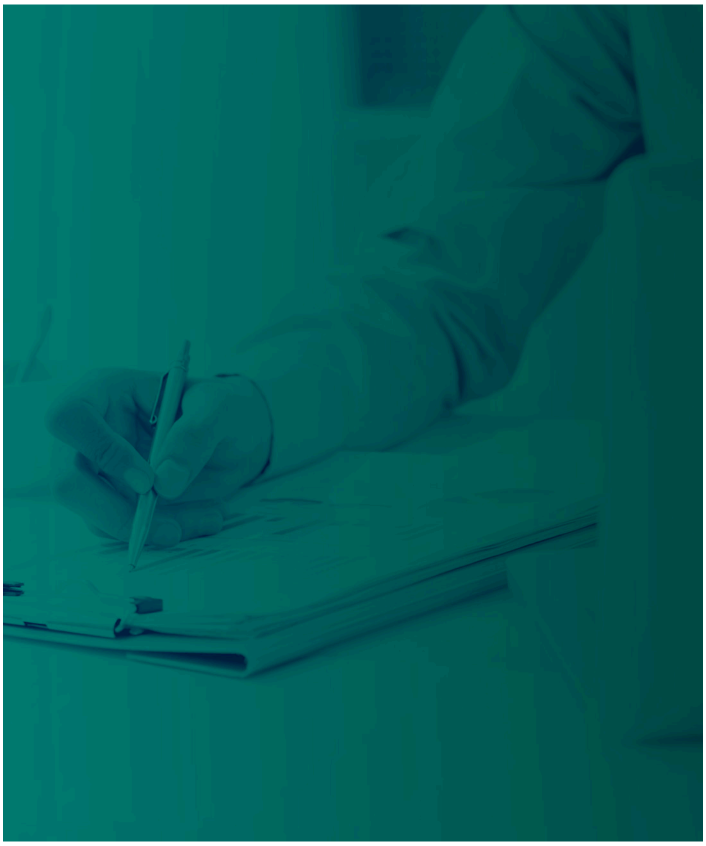
2. ESCAPE FROM THE MOST COMMON ERROS DURING ERP´S IMPLEMENTATION

The implementation of an ERP System requires a large investment both financially and in relation to the human capital that must be allocated during the execution of the project.

While a successful project will translate into streamlining costs and workflow, an unsuccessful one can lead to productivity losses and delays.

Support a risk mitigation policy, where avoiding mistakes will be much less costly than correcting them.

CIO.com, the digital magazine with the largest amount of signatures among CIOs and technology executives, listened to ERP experts and listed some of the most common pitfalls:



Lack of planning

Plan, plan and plan.

During the implementation period, always seek to know the next steps and tasks of your project, ensuring that you will have all the necessary resources at the right time for execution.

Underestimating the time and resources needed

Do not underestimate time, especially in the testing stages. Do not underestimate resources. In order for the work to be well done, it will be necessary to involve knowledgeable resources of the business and in certain phases of the project, this resource must be involved 100% of the time, guaranteed the attention that an implementation project should receive from the organization.

Not having the right people on the team right from the start

Engaging the right amount of resources and the right people right from the start of the project minimizes rework. Since the right people are expected to make the right decisions, guiding the consulting team in the best way to set up the system.

It is usual for each module to have a key user responsible for its implementation. Letting a whole set of modules under the responsibility of a single person will divide your attention.

Underestimate the importance of accurate data

The ERP system is only as good as the data that is in it.

Testing, testing and testing must be done to ensure that the results are in accordance with the data included in the system.

Do not have a test environment

You will not be able to see the true results of changes with a restricted database for testing.

It is important to know the volumes of data, and keep the testing environment up-to-date so that the test performed is significant and actually mirrors the production environment.

3. THE IMPORTANCE OF TESTING DURING ERP IMPLEMENTATION

From now on, we will focus on an essential matter for the guarantee of ERP implementation: Tests.

Above, on this document, we have already emphasized the importance of the quality of the tests, as well as the involvement of the client's team to guarantee its comprehensiveness and results.

Involving the users since the initial definition of the tests and their execution, makes them feel responsible and committed with final result, guaranteeing the system´s homologation and mitigating risks when the system is implemented in the Production environment.

Minimizing the testing stage is a common mistake that some companies make in their quest to minimize project´s costs. Surveys point out that almost 50% of the total budget for ERP implementation is "spent" on test activities, which in itself is a temptation to reduce costs.

Do not fall into this trap, because the quality of the tests and the time spent on this task will directly impacts the success of your system implementation.

The Test Phase must happen after the system´s set up according to the reality of the company.

Initially, individual tests of each system functionality should be performed. Preferably, tests must be performed by key users with the follow-up of the partner´s implementation team. This step provides the customer with a closer contact with the new system such as an on-the-job training in its functionality. These individual tests will also serve to identify possible corrections in the made set up.

Next, integrated tests must be run. These kind of tests are of the utmost importance as they transcend boundaries, passing through various modules of the ERP system and business areas of the company.

During the integrated testing stage, 100% involvement of users in all areas of the company is required. They will be responsible for testing and gathering evidence, also closely monitored by the partner´s implementation team.

On this occasion users will identify how certain information included in specific module will affect the congener modules.

At the end of the Integrated Test, the approval of the system to be implanted is expected. Without this approval, the system will not be taken to the Production environment.

Any failures identified during the Integrated Test, once documented, and if they are not a very serious problem, can be treated after the GO LIVE date. This decision should be make together (customer and partner) after analyzing pros and cons of postponing the GO LIVE date in relation to the identified failure.

To ensure the right approach to testing an ERP, plan, prepare, execute, monitor, and collect test results.

Considering the reality of JDEdwards Implementation, Upgrade and Enhancement projects, here are some tips to improve delivery quality:

Planning

Define the test strategy: the types of tests, the roles of each one involved, the responsibilities, tools and criteria that will be used during test´s execution.

Depending on the reality of your project, you will need not only a functional test, but also plan stress and contingency tests.

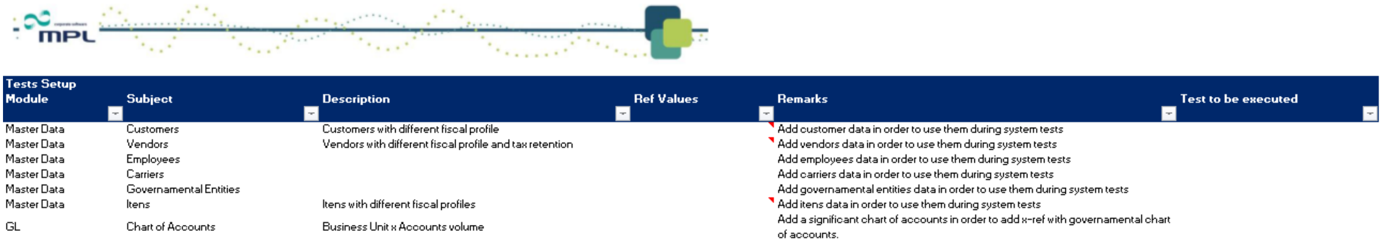
Identify processes and data relevant to testing.

Implement a practice that will allow you to know the critical areas and volumes of data to be tested. When starting a project together with the client, seek answers to some relevant questions. For example, how many accounting entries per month are expected, how much invoices are issued daily, which incoming and outgoing transactions should be tested.

Do not forget to mention and document in the scope of the test any external interface to the ERP.

In this way, both the implementer and the client will have a slightly clearer view on the tests´ data that should be included and treated.

Use tools to identify and record processes and volume of data to be tested. As an example, below is a worksheet that must be completed in conjunction with the customer, focused on detailing the client's data mass:



Tests Setup Module	Subject	Description	Ref Values	Remarks	Test to be executed
Master Data	Customers	Customers with different fiscal profile		Add customer data in order to use them during system tests	
Master Data	Vendors	Vendors with different fiscal profile and tax retention		Add vendors data in order to use them during system tests	
Master Data	Employees			Add employees data in order to use them during system tests	
Master Data	Carriers			Add carriers data in order to use them during system tests	
Master Data	Governamental Entities			Add governamental entities data in order to use them during system tests	
Master Data	Itens	Itens with different fiscal profiles		Add itens data in order to use them during system tests	
GL	Chart of Accounts	Business Unit x Accounts volume		Add a significant chart of accounts in order to add x-ref with governamental chart of accounts.	

In JDEdwards Implementation projects, these information should be colected soon after the requirements analysis. Since this should be the first contact of users with the JDEdwards System, it is recommended to provide a simplified training covering the topics of master data (Address Book and Item Master) inclusion and basic configurations, so that the client can be responsible for add its tests´ data.

In a JDEdwards Upgrade project, while the data migration is done, you must validate the master data by choosing items, customers, and vendors that should be part of the tests.

In JDEdwards Enhancement projects, we suggest that data collection should be done soon after the approval of the Functional Specifications containing the changes that are to be implemented. Knowing, in detail, the changes to be implemented, it is possible to treat all the variations that a test data must present.

In the last two cases above, probably it won´t be necessary to train users, since they are already familiar with the System. In this case, the customer can review the records without supervision of the partner.

Preparation

Define the test plan and test schedule. Since the initial project timeline, there should be activities regarding both individual and integrated testing.

In case of individual tests, for each implemented customization it is important to elaborate Test Scenarios, which indicates the steps for the execution of a specific test, considering all possible variations related to the database.

This scenario, if elaborated by the partner team, should be forwarded to the client. So key user can validates it and, if necessary, add new scenarios.

At last, this scenario, should contain:

- the basic settings that must be made before the test is started. In terms of the JDEdwards System, these basic configurations may involve completing processing options, adding UDCs, and so on.
- the data required to perform the tests.
- the necessary instructions - step by step - so that the user can, following these instructions execute the test from start to finish.
- a description of the expected result at each step contained in the test scenario
- the date the test was run, observations regarding the test result and its final status (if approved or rejected).

The latter information must be completed by the person responsible for the test during its execution.

If your company is using previously approved templates for ERP Implementation, standard functionalities have often been tested. In these cases, create test scenarios when you need to change preconfigured operations or configure new operations.

But remember that all features - even standards ones - must be tested invariably during integrated testing.

For integrated tests, build test scripts considering processes from start to finish. Send the scripts to the customer's team for validation.

This script, similar to the Individual Test Scenarios, should contain:

- the identification of the process being tested
- the steps in the tests.

In this case, list possible integrations to be tested at this time, if they have not yet been tested during the individual tests.

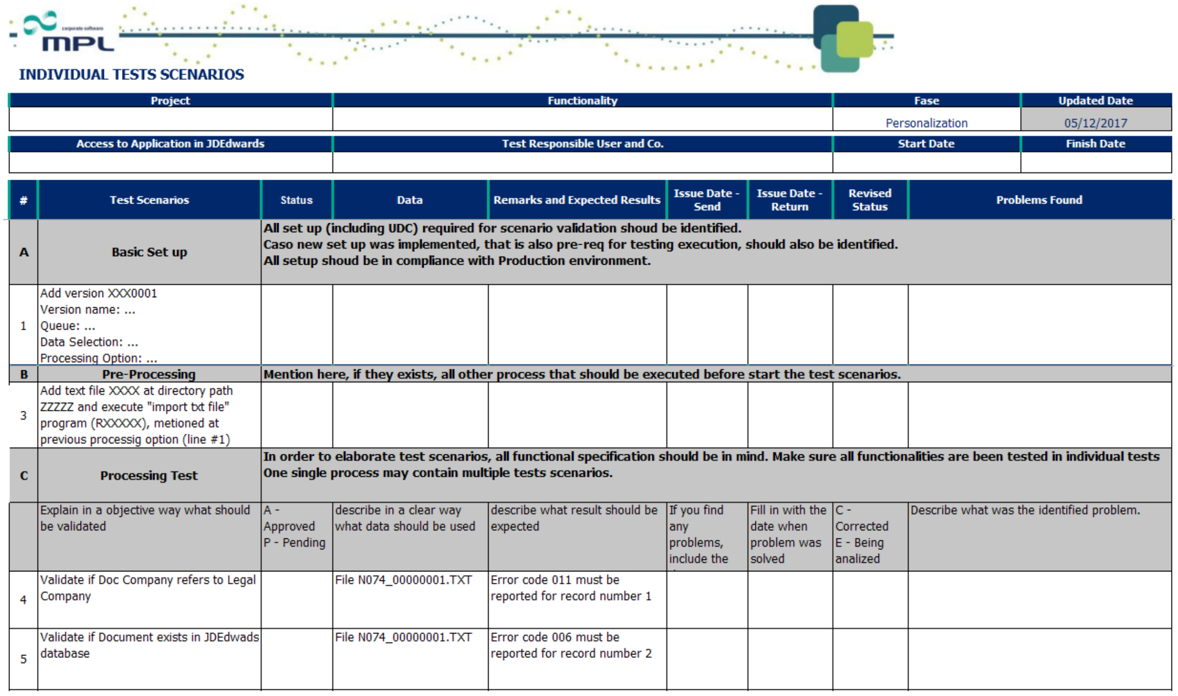
- necessary instructions for performing the tests.
- expected outcome at the end of each stage to be performed
- information that is prerequisite for performing subsequent testing steps.

For example, if we are going to receive a purchase order, we need the Purchase Order Number. This information should be recorded at a previous testing stepa of "Creating a Purchase Order".

This should be a common practice since the user responsible for receiving the order may be different from the user responsible for including it.

- identification of the user responsible for each test step within the process
- date, result achieved, and final test status

This information must be completed during the execution of the Integrated Tests.



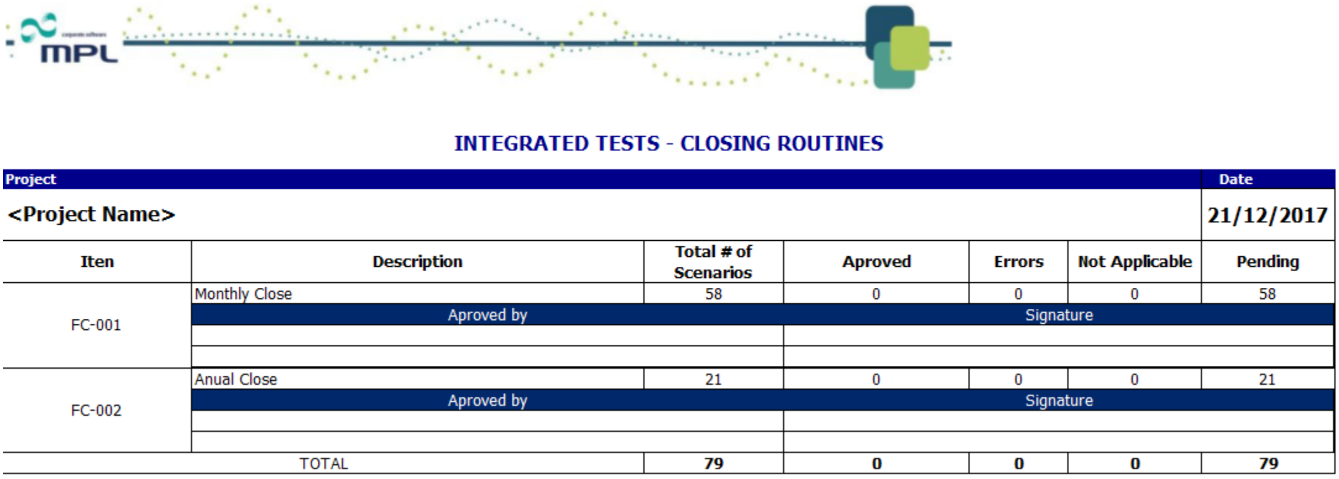
INDIVIDUAL TESTS SCENARIOS

Project	Functionality	Fase	Updated Date
		Personalization	05/12/2017

Access to Application in JDEdwards	Test Responsible User and Co.	Start Date	Finish Date

#	Test Scenarios	Status	Data	Remarks and Expected Results	Issue Date - Send	Issue Date - Return	Revised Status	Problems Found
A	Basic Set up	All set up (including UDC) required for scenario validation shoud be identified. Caso new set up was implemented, that is also pre-req for testing execution, should also be identified. All setup shoud be in compliance with Production environment.						
1	Add version XXX0001 Version name: ... Queue: ... Data Selection: ... Processing Option: ...							
B	Pre-Processing	Mention here, if they exists, all other process that should be executed before start the test scenarios.						
3	Add text file XXXX at directory path ZZZZZ and execute "import txt file" program (RXXXXX), metioned at previous processig option (line #1)							
C	Processing Test	In order to elaborate test scenarios, all functional specification should be in mind. Make sure all functionalities are been tested in individual tests One single process may contain multiple tests scenarios.						
	Explain in a objective way what should be validated	A - Approved P - Pending	describe in a clear way what data should be used	describe what result should be expected	If you find any problems, include the	Fill in with the date when problem was solved	C - Corrected E - Being analized	Describe what was the identified problem.
4	Validate if Doc Company refers to Legal Company		File N074_00000001.TXT	Error code 011 must be reported for record number 1				
5	Validate if Document exists in JDEdwads database		File N074_00000001.TXT	Error code 006 must be reported for record number 2				

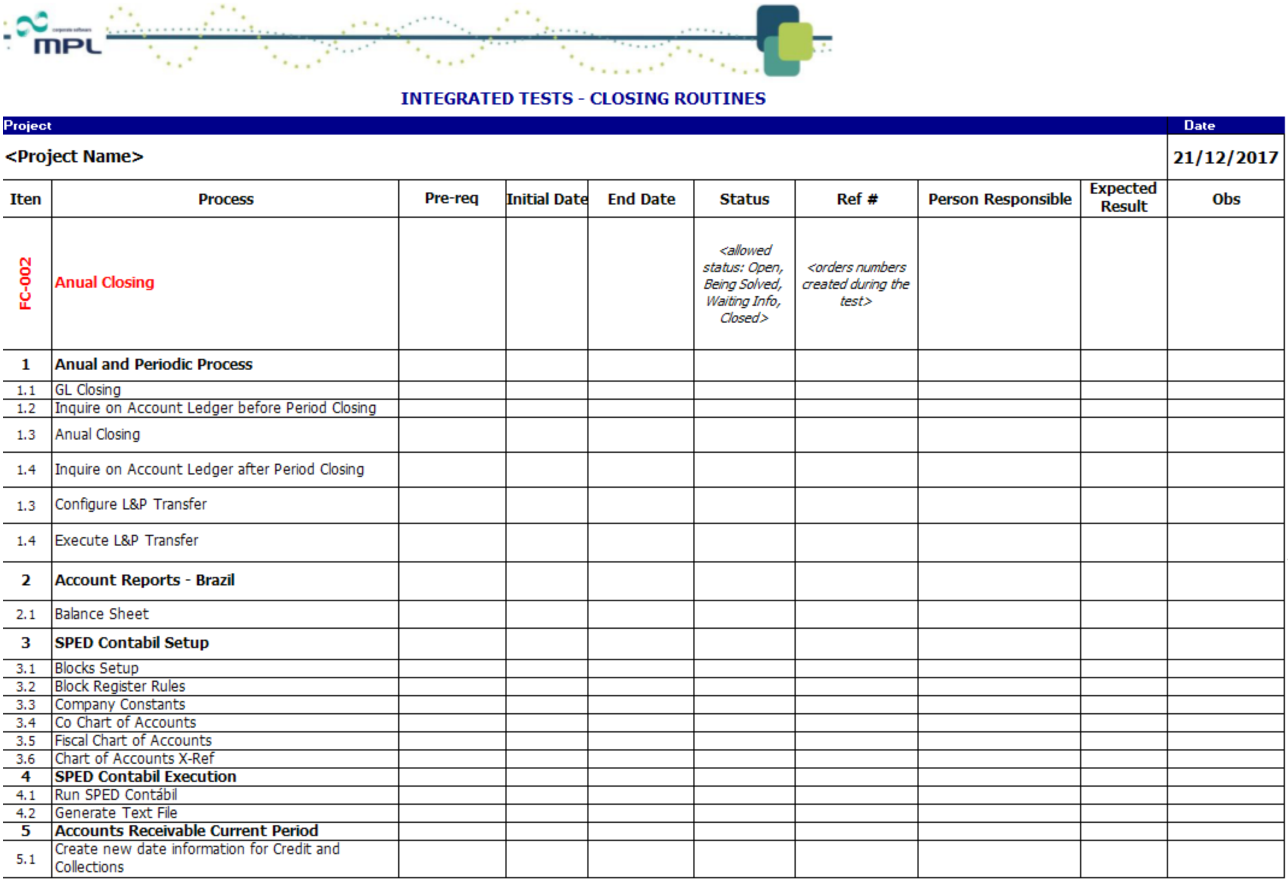
The Integrated Test Scripts should be done by processes, containing a cover sheet listing all the subprocesses to be tested, their approver and the total number of processes involved, whether or not they are successfully tested and still pending testing processes:



INTEGRATED TESTS - CLOSING ROUTINES

Project						Date
<Project Name>						21/12/2017
Iten	Description	Total # of Scenarios	Aproved	Errors	Not Applicable	Pending
FC-001	Monthly Close	58	0	0	0	58
	Aproved by		Signature			
FC-002	Anual Close	21	0	0	0	21
	Aproved by		Signature			
	TOTAL	79	0	0	0	79

Each subprocess listed on the cover page should have its own test scenario, identifying each of the following steps:



INTEGRATED TESTS - CLOSING ROUTINES

Project									Date
<Project Name>									21/12/2017
Iten	Process	Pre-req	Initial Date	End Date	Status	Ref #	Person Responsible	Expected Result	Obs
FC-002	Anual Closing				<allowed status: Open, Being Solved, Waiting Info, Closed>	<orders numbers created during the test>			
1	**Anual and Periodic Process**								
1.1	GL Closing								
1.2	Inquire on Account Ledger before Period Closing								
1.3	Anual Closing								
1.4	Inquire on Account Ledger after Period Closing								
1.3	Configure L&P Transfer								
1.4	Execute L&P Transfer								
2	**Account Reports - Brazil**								
2.1	Balance Sheet								
3	**SPED Contabil Setup**								
3.1	Blocks Setup								
3.2	Block Register Rules								
3.3	Company Constants								
3.4	Co Chart of Accounts								
3.5	Fiscal Chart of Accounts								
3.6	Chart of Accounts X-Ref								
4	**SPED Contabil Execution**								
4.1	Run SPED Contábil								
4.2	Generate Text File								
5	**Accounts Receivable Current Period**								
5.1	Create new date information for Credit and Collections								

It is important that both the Individual and Integrated Test Scenario are developed together with the key users. If this is not possible, they must be approved by the key user before the tests are started.

Prepare the test environment. Consider a significant volume of master data and operations to be performed so that the tests mirror, as closely as possible, the reality of the production environment.

In upgrade or improvement projects, plan a data refresh of the test environment with the production data.

During the tests - at the latest, during the integrated tests - it is important that the access profiles are already established so that they can be tested by each of the users.

Conduct a Test Planning meeting, involving all key users and project managers. At this meeting emphasize the importance of the tests, involve them to look for real examples of the operations that should be tested. To do this, refer to the first stage of test planning, where volumetry, masses of tests and operations were listed.

Make clear the dates and expected commitment of each of the users during the tests, based on the schedule that was approved by the client.

Execution

Run the unit tests according to the test scenarios built and approved by the client.

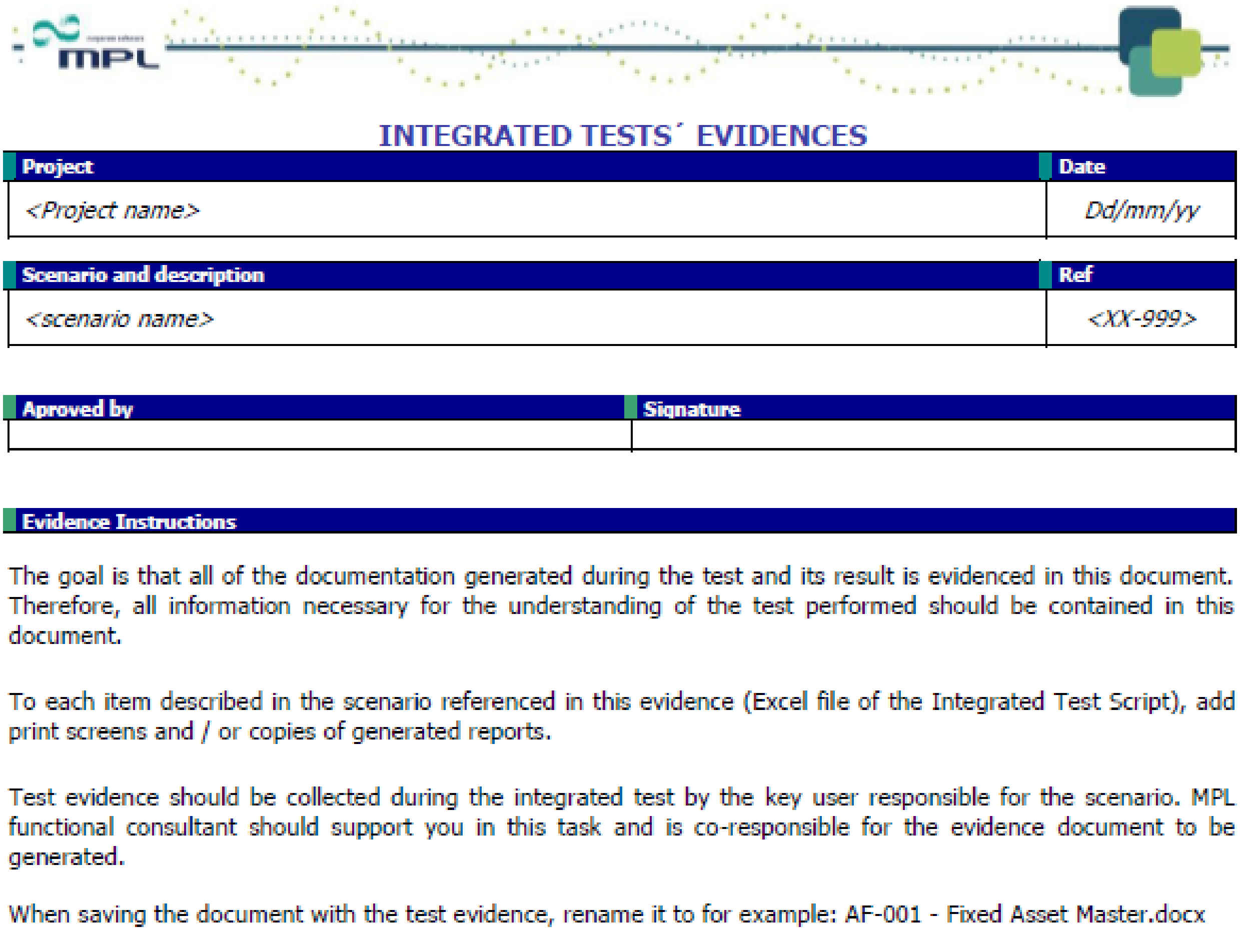
Key Users are responsible for performing these tests, with the third-party consultant´s support and follow-up.

Individual tests can be performed throughout the project as customizations are developed and made available to the end user.

After approval of all individual tests, the Integrated Test must be started.

Integrated testing should follow previously approved scripts. At this moment, key users are responsible, always oriented by the outsourced consulting company, to collect tests´ evidences.

These collected material must contain print-screens, generated reports, or any other information that evidences each of the steps contained in the script. Identify the tests´ evidence using the identification of the process being tested, so it will be easy to correlate the test performed and its evidence.



INTEGRATED TESTS´ EVIDENCES

Project	Date
<Project name>	*Dd/mm/yy*

Scenario and description	Ref
<scenario name>	*<XX-999>*

Aproved by	Signature

Evidence Instructions

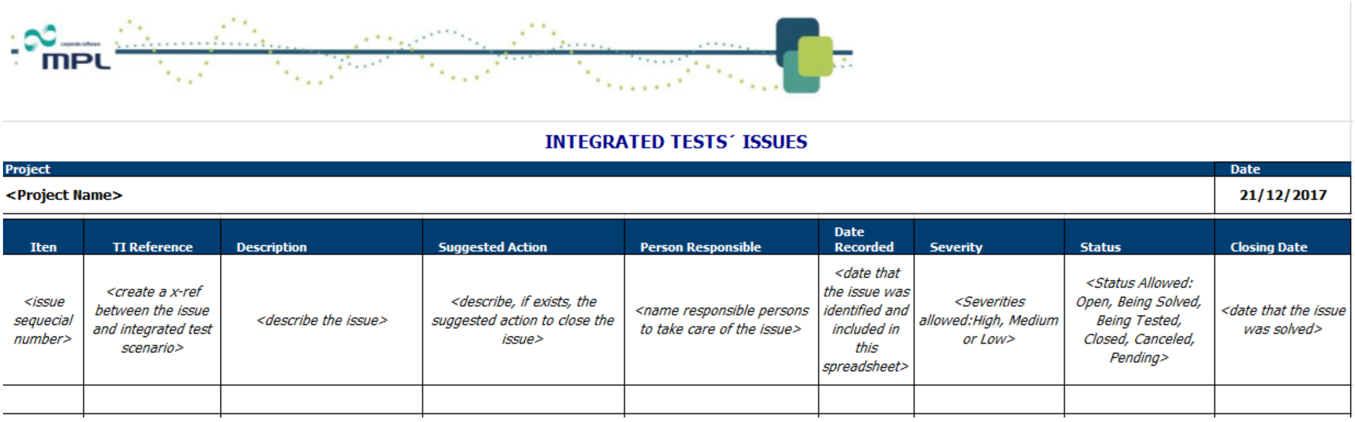
The goal is that all of the documentation generated during the test and its result is evidenced in this document. Therefore, all information necessary for the understanding of the test performed should be contained in this document.

To each item described in the scenario referenced in this evidence (Excel file of the Integrated Test Script), add print screens and / or copies of generated reports.

Test evidence should be collected during the integrated test by the key user responsible for the scenario. MPL functional consultant should support you in this task and is co-responsible for the evidence document to be generated.

When saving the document with the test evidence, rename it to for example: AF-001 - Fixed Asset Master.docx

Every failure identified during testing should be documented so it can be managed. In most cases, the test should be repeated until it is successfully completed. Otherwise, the decision to abandon a particular scenario must be made in conjunction with project management.



INTEGRATED TESTS´ ISSUES

Project	Date
<Project Name>	21/12/2017

Iten	TI Reference	Description	Suggested Action	Person Responsible	Date Recorded	Severity	Status	Closing Date
<issue sequecial number>	*<create a x-ref between the issue and integrated test scenario>*	*<describe the issue>*	*<describe, if exists, the suggested action to close the issue>*	*<name responsible persons to take care of the issue>*	*<date that the issue was identified and included in this spreadsheet>*	*<Severities allowed:High, Medium or Low>*	*<Status Allowed: Open, Being Solved, Being Tested, Closed, Canceled, Pending>*	*<date that the issue was solved>*

Each successful test script must have a formal approval in its evidence.

In cases of Upgrade or Improvement projects, a regression test should be done. That is: to ensure that the changes applied will not impact the other functionalities already implemented.

Monitoring

To monitor the progress of the tests, consider the number of steps described in each script and the number of steps already successfully tested.

In this way, you will be able to follow the progress of the tests and check if the test will need to be extended or if it will be over before the expected time.

Mitigating possible delays in the testing phase is key to securing the GO LIVE end date.

During the tests´ monitoring, questions about the usability and performance of the new system should be addressed and resolved.

Closing

After running all the Integrated Test scripts, it will be necessary to evaluate whether all the test evidence has been approved and whether all the faults found have already been closed.

At this time, a meeting should be held for the final decision regarding GO LIVE.

If there are no faults to be corrected and all the evidence is approved, there is no doubt and the project can move on to its final stage which corresponds to the preparation of the production environment and GO LIVE.

If there are logged faults, these should be analyzed, considering the pros and cons of a GO LIVE postponement. If it is a small business failure, probably, it can be corrected after GO LIVE, during the stabilization phase of the new ERP.

4. TESTING AID TOOLS

In the market, there are some tools that can be considered to assist in the execution of the tests.

These tools are mainly important if there is an expectation of repeating the tests, for example in more than one cycle during deployment project´s period or if a large volume of data is required.

These tools can be divided into groups classified as:

Quality Center: allows user tests´ automation in corrective or improvement projects, since they allow to save scenarios of previous transactional tests.

Functional Testing: a complete test automation and regression testing solution that helps reduce risk and error.

Load Runner: stress or performance testing tool, which simulates the parallel execution of multiple users.

5. REFERENCES

1. Como Funciona a Metodologia de Implantação de ERP

Brascomm 2.0

Dezembro, 2016

2. As 6 fases da implementação de um sistema ERP

www.hoito.com.br

Fevereiro, 2016

3. 13 Erros Comuns de ERP e Como Evitá-los

Jennifer Lonoff Schiff, CIO/EUA

Março, 2012

4. GUTS-RS Testes em Projetos de Implantação de ERP

Carlos Tristacci e Rafael Machado Eder

Novembro, 2015

5. O Papel dos Testes em Sistemas de ERP

Prime Control

Julho, 2016

corporate software

MPL

ABOUT MPL: MPL has been operating in the Brazilian and South American markets since 1985, having started its operation with JDEdwards in 1993, being the leader of the practice in Brazil.

It has performed hundreds of JDEdwards implementation projects in Brazil and more than 60 rollouts in the territory.

It consolidated a broad knowledge of the legal aspects of Brazil and of all the market best practices that are applicable to Brazilian companies, having developed its own methodology for conducting rollout projects.

It has expert consultants in various industries and areas of competence of JDEdwards with more than 20 years of experience.

It maintains a business unit focused on JDEdwards support, helping more than 3000 end users in various versions of WorldSoftware and Enterprise One, applying its own methodologies and tools focused on the approval and testing of the deployment project.

Developed accelerators for the implementation process that reduce risks and deadlines.