Testing Framework

Approval

Business Owner	Approved/Not Approved	Signature	Title	Date
John Parry			Associate Director, Enterprise Systems & Project Delivery	

Approval

Approved: ❑ Rejected: ❑ --- Peter Nikoletatos Chief Information Officer Date:	Comments:

Author: Nalini Nair, Team Lead, Business Analysis & Testing, ITS
Version: 1.0
Date: 20 November 2013

Contents

Introduction

Testing is an activity undertaken to assess the quality of a product. It is essentially the verification and validation process to check whether the designed product meets the customers' specification. Testing starts with the exploration of the requirements and what the customer really wants by elaborating on the user stories from different perspectives. Testing becomes a continuous and integrated process where all parties in the project are involved.

Purpose of this document

The purpose of this testing framework document is to create a shared understanding of the overall targets, approach, tools and timing of test activities and service management function for the University's Enterprise Systems. ITS's testing and service management objective is to achieve higher quality and shorter lead times with minimum overhead, frequent deliveries, close teamwork with team and stakeholders, short feedback loops and extensive collaboration.

Enterprise Systems

Enterprise systems are software packages that support business processes; information flows; reporting, and data analytics across the University. The University ensures appropriate data management standards are in place for enterprise system information. ITS provide support and service management function to ensure that governance, security and risks are addressed appropriately to maintain system integrity, reliability and enable continuous functionality.

Enterprise System is defined as having the following attributes:

- Authoritative source of information across the ANU
- Critical to business process and business continuity
- Enables ANU to meet mandatory statutory reporting requirements
- Supports a high risk activity which is core to the ANU, e.g. compliance, teaching

Product Induction

The business owner (subject matter expert) will provide ITS Business Analysts and Test Analysts with comprehensive product induction of the product to be tested. The approach and timeline for the induction exercise will be discussed and agreed by ITS and the business owner prior to commencement of testing activities. It is critical for ITS to undertake this induction process to obtain sufficient knowledge of the product to enable the provision of quality testing, optimal support and quality service management.

Testing Approach

Testing is essentially a risk-based activity. The Test Analyst will design a test approach that utilizes a balance of testing techniques to cover a representative sample of the system in order to minimize risk while still delivering a quality application to production in a timely manner.

The test team will plan, prepare and execute the tests. All testing related plans, schedule, identified risks, tasks effort estimates etc. will be discussed and provided to key stakeholders, as part of the

deliverables. Prior to commencement of testing, testing artefacts such as test plan, test cases, traceability matrix, etc. will be distributed to all relevant parties for acceptance.

Key Stakeholders Participation

Key stakeholders are expected to make themselves available for required participation in the testing process. Commitment and timely responses from stakeholders on clarification and confirmation of testing issues as and when required are essential to the efficient and effective progress and completion of the testing phase.

Time Management

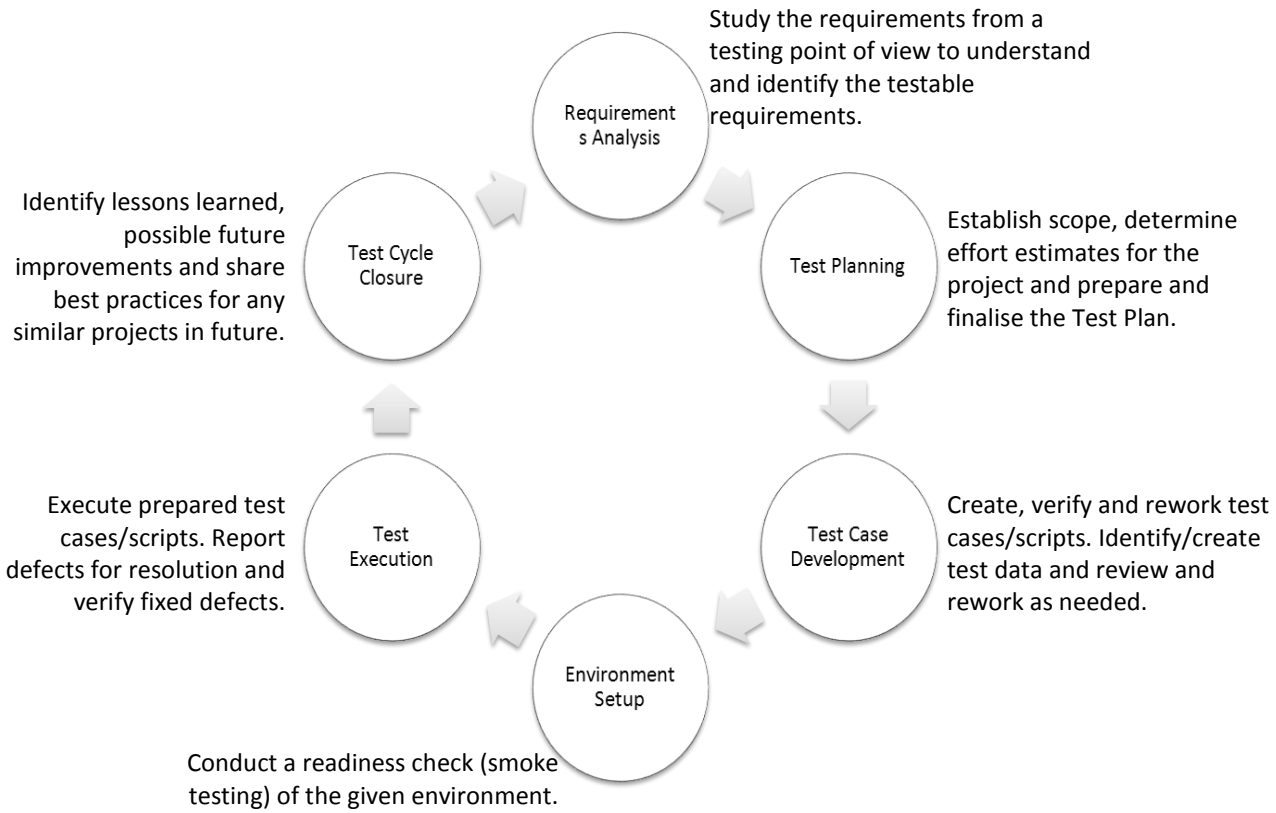
In the interest of all parties and the requirement for ITS to meet committed goals, effective time management is essential. The testing timeline will be established and agreed to prior to commencement of any testing. The test team will provide their time estimates for testing.

Critical Success Factors

The following requirements are critical to the success of achieving effective and efficient testing outcomes and ensure that the end product meets the expected quality.

- Testing considerations must begin in the early phases of an initiative/project.
- Risk-based testing must commence early, identifying risks to system quality and using that knowledge of risk to guide test planning, preparation and execution.
- Test case development must be based on key initiative/project outputs.
- Testing must be objective and must be performed independent of the developers responsible for the application.
- The defect management process must be functional as soon as testing begins, and must ensure that only valid and non-duplicated defects are processed.
- Planning for the systems integration test should start early, as it will involve multiple initiatives/projects, systems, and business solutions groups/colleges
- The scope of regression test should be well defined and where possible, an automated tool should be used to perform regression testing.
- Specification components/releases should be categorized by their relative importance to the business for defect prioritization and performance testing.

Testing Lifecycle



Testing Tool

All testing will be undertaken using ITS standard testing tool, Hewlett Packard Quality Centre (HPQC). The modules that will be utilised in HPQC are:

- Release Module
- Requirements Module
- Testing Module
- Test Lab Module
- Defects Module

Testing Environments

Prior to commencement of testing, a minimum of 3 environments must be available.

Environment	Description
Development	This is the environment where development of the system is performed. Unit testing will be conducted in this environment before the code is migrated to the test environment.

Environment	Description
Acceptance/Test	This is the environment where the test team will test the code against the completed development of requirements.
Production	This is the live production environment operational for use by the client where verification of the delivered product is conducted.
User Acceptance Testing (Optional)	This is the environment where user acceptance testing will be conducted. It is recommended that a separate environment be established so that internal testing continuity can occur concurrently with user acceptance testing.
Automated Test (where applicable)	This environment is required if automated testing is to be implemented. Automated testing, load testing etc. will need to occur concurrently with acceptance testing, but cannot share the same environment due to data or code version differences.

Quality Assurance Activities

ITS will undertake quality assurance activities to ensure that the development of enhancements, new features/functions and defect resolutions are actioned and delivered according to required acceptance criteria and established key performance indicators.

Smoke Testing

Smoke Testing is performed to ascertain that the critical functionalities of the program are working as expected. It is executed before any detailed tests are executed on the software build. The purpose is to reject a badly broken application, so that the testing team does not waste time testing the software application.

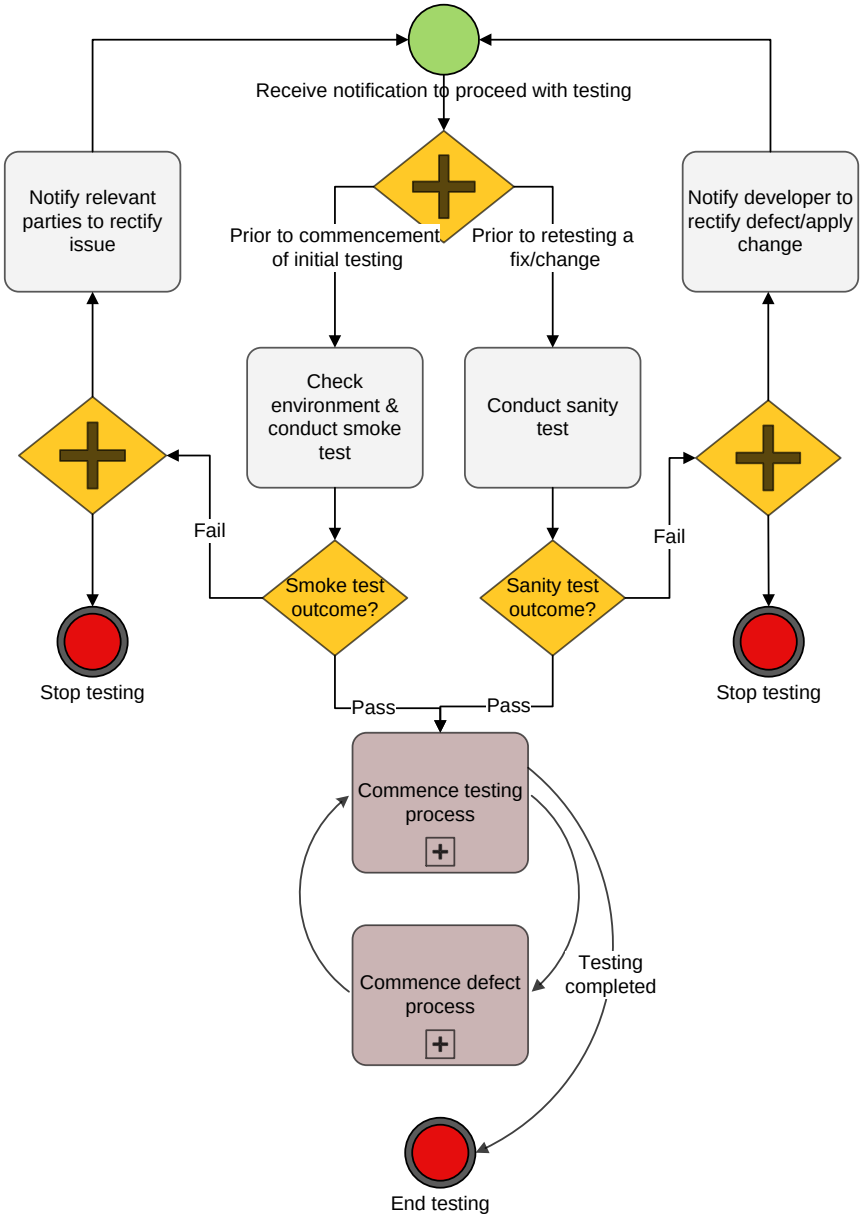
The test cases chosen in the smoke testing activity addresses the most important functionality or component of the system. The objective is not to perform exhaustive testing, but in a cursory way, to verify that the critical functionalities of the system are working. It is verifying the stability and data quality of the system. E.g. A typical smoke test would be to verify that the application launches successfully; check that the user interface is responsive; verify that data loaded in the test environment is reliable and usable, etc.

Smoke testing is like a general health check up.

Sanity Testing

Sanity testing is performed to ascertain that changes and fixed defects have not introduced further issues into the system. The purpose is to determine that the proposed functionality works roughly as expected. If sanity test fails, the build is rejected to save time and costs involved in a more rigorous testing. The objective is not to verify thoroughly the new functionality, but to determine that the developer has applied some rationality in developing the software whilst applying changes and rectifying defects.

To ensure that quality assurance and testing activities are able to proceed without incident, ITS will undertake the following process:



Quality Attributes

The following quality attributes have been identified as essential and are used as a basis for the test approach in terms of priority and test targets.

Attribute	Description	Measure and Target
Correctness	Features and functions work as intended	1. 100% completion of agreed features 2. Severity 1 defects = 0 3. Severity 2 defects = 0 4. Severity 3 defects < 5

Attribute	Description	Measure and Target
		5. Severity 4 defects < 10
Integrity	Ability to prevent unauthorised access, prevent information loss, protect from viruses infection, protect privacy of data entered	1. All access will be via HTTPS over a secured connection. 2. User passwords and session tokens are encrypted.
Availability	Percentage of planned up-time that the system is required to be operational.	System is available for 99.99%
Cross Device Compatibility	User interface should render and function properly on different devices	1. Personal computer with Microsoft Windows, Mac OS, Linux, etc. 2. Mobile devices: • Smartphones (iPhone, Android, Blackberry, etc.) • Tablet Computer (iPad, Samsung, Asus, etc.)
Cross Browser Compatibility	User interface should renders and functions properly on the most common Internet browsers	1. IE version >= 8.0 2. Firefox version > = 3.5 3. Safari version >= 5.0 4. Opera 5. Chrome version 11.0
Interoperability	Ease with which the system can exchange information with other systems.	System should be able to exchange information with other systems. E.g. HORUS, ISIS, Wattle, IT Service Desk or StudyAt, etc.
Performance, Load, Stress	Measures the responsiveness of the system under a given load and the ability to scale to meet growing demand.	System should be able to respond within an acceptable timeframe without crashing or breaking down under overloaded conditions.

Levels of Testing Execution

Unit Testing

Unit testing is undertaken by the developer at the development stage of the project. Each smallest piece of testable software in the application is tested in isolation as it is developed. The developer will test the integration of their component in the development environment, against the requirements, prior to moving the completed development into the code repository. Once unit testing has been confirmed, the developer will move the completed development into the test environment and conduct a cursory check to make sure the change has successfully migrated.

Systems Integration Testing

Systems integration testing is undertaken by the test team in collaboration with developers, DBAs or owners of external systems, at the testing stage of the project. This testing verifies that a system is integrated to any external or third-party systems defined in the system requirements, e.g. integration between researchers and IMIS interface or researchers and LDAP.

Solution Requirements Testing

Solution requirements testing are undertaken by the test team at the testing stage of the project. This testing verifies that the system is functioning in accordance with the required functional and non-functional specifications documented by the Business Analyst. Usability testing forms part of the requirements testing to address how usable the product is in providing a good user experience.

Regression Testing

Regression Testing is undertaken by the test team to uncover new errors in existing functionality, after changes have been made to a system, such as functional enhancements, patches or configuration changes. The intent of regression testing is to assure that a change (such as a bug fix) did not introduce new faults.

Load Testing

Load testing is undertaken by the test team in collaboration with the development team. It is the process of putting demand on a system and measuring its response. It tests the capability of the application to function properly under expected normal production conditions and measure the response times for critical transactions or processes to determine if they are within the limits specified in the requirements.

Performance Testing

Performance testing is undertaken by the test team in collaboration with the development team. It is conducted to evaluate and understand the application's scalability when, for example, more users are added or the volume of data increases. This is particularly important for identifying bottlenecks in high usage applications.

Stress/Stability Testing

Stress/stability testing is undertaken by the test team in collaboration with the development team. It is performance testing at a higher than normal simulated loads. Stressing runs the system or application beyond the limits of its specified requirement to determine the load under which it fails and how it fails. A gradual performance slow-down leading to a non-catastrophic system halt is the desired result.

Data Conversion Testing

Data conversion testing is undertaken by the test team in collaboration with the development team. It is performed to verify the correctness of automated or manual conversions and/or loads of data in preparation for implementing the new system.

Production Verification Testing

Production verification testing is conducted by the project team, testing team and users in order to run a cursory check and verify that the changes have been successfully implemented in Production. It is done following the move of the system into Production. Tests conducted in Production should not affect any live processing or interfere with production integrity.

Testing Deliverables

Testing activities will be documented to support communication and understanding of the testing methodology, approach and test coverage. The Test Analyst will prepare the test artefacts for distribution to all relevant parties at various stages of testing.

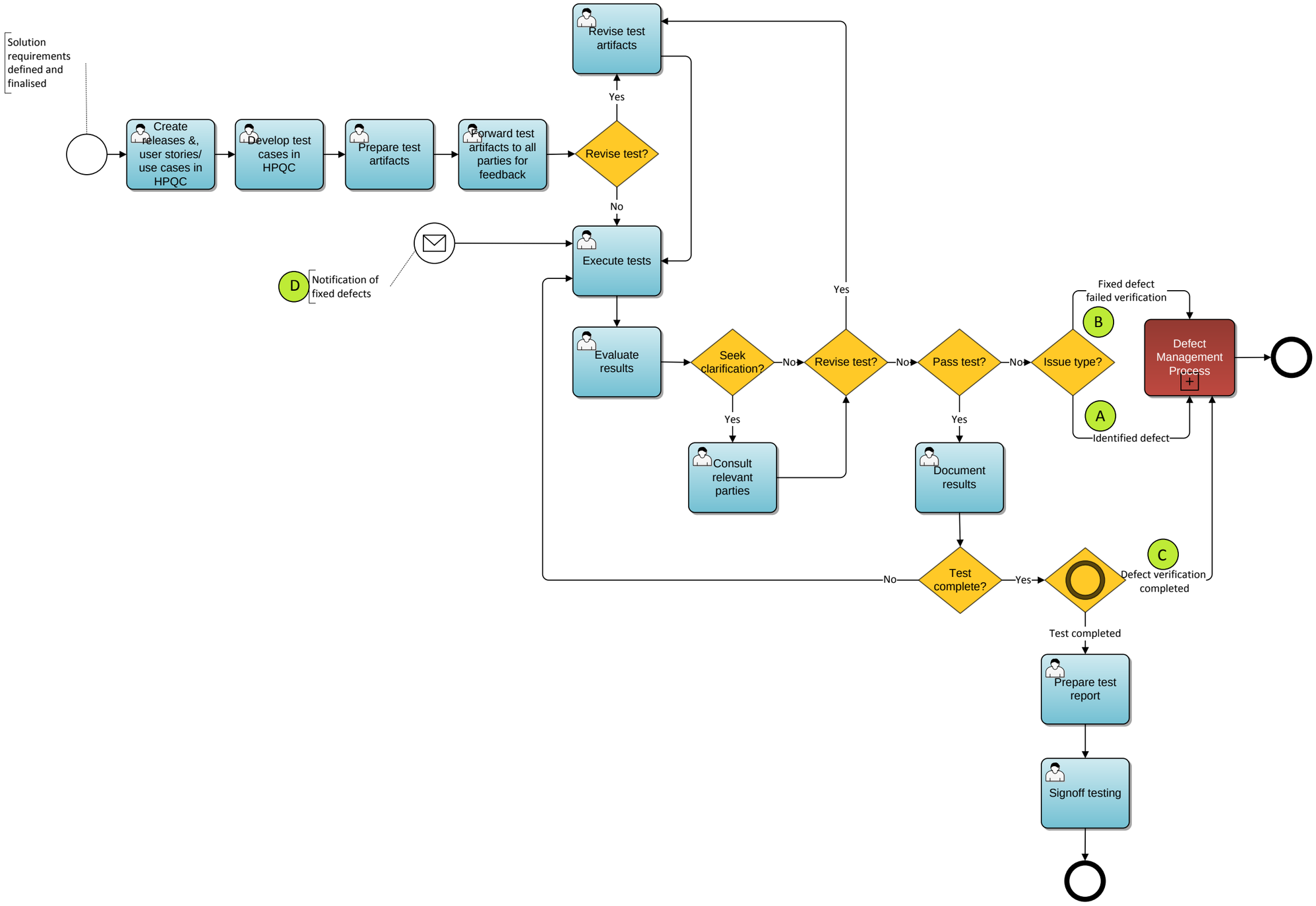
Formalised test results will be provided by the Test Analyst indicating completeness and confirmation that the developed product is in a state that is ready for user acceptance testing.

There may be times where testing will need to be qualified due to circumstances beyond the Test Analyst's control (i.e. significant constraints to testing is encountered) resulting in an outcome of incomplete coverage of testing. In such cases, the Test Analyst will include a qualification report detailing reasons for limited testing coverage and highlight the possible risks and impact to the end users.

Templates, forms and samples of the following testing deliverables are attached in Appendix 1.

Testing Stage	Deliverables
Initiation	High Level Estimates
Planning	Test Plan
Development	Test Cases (extracted from HPQC)
	Traceability Matrix (extracted from HPQC)
Execution	Testing Progress Report
	Defect Tracking Report
	UAT Defect Reporting Form
Close	Test Results & Evaluation Report
Throughout the testing stage	Testing Risk Register

Testing Process [Illustration without roles & detail]

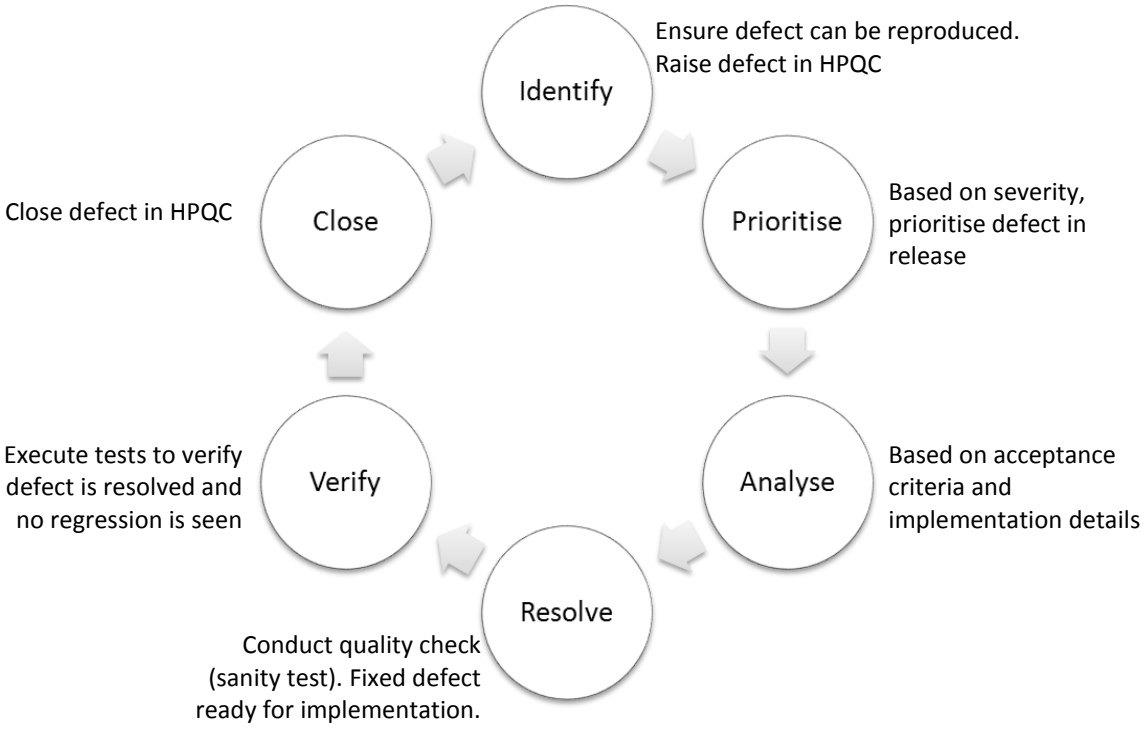


Testing Process [Illustration with roles & detail]

Defect Management

Defects are inevitable but we can minimize the number of defects and their impact on the system. To minimize defects as much as possible we need to assess the critical risks associated with the system. As risk is inherent in all development projects, defining critical risks enables the business analysis and testing team to identify the types of defects that are most likely to occur and the ones that will have the greatest impact on the system. With this information, strategies can then be developed to prevent them and testing can be focused on areas of high risk and high impact.

Defect Lifecycle



Defect Classification

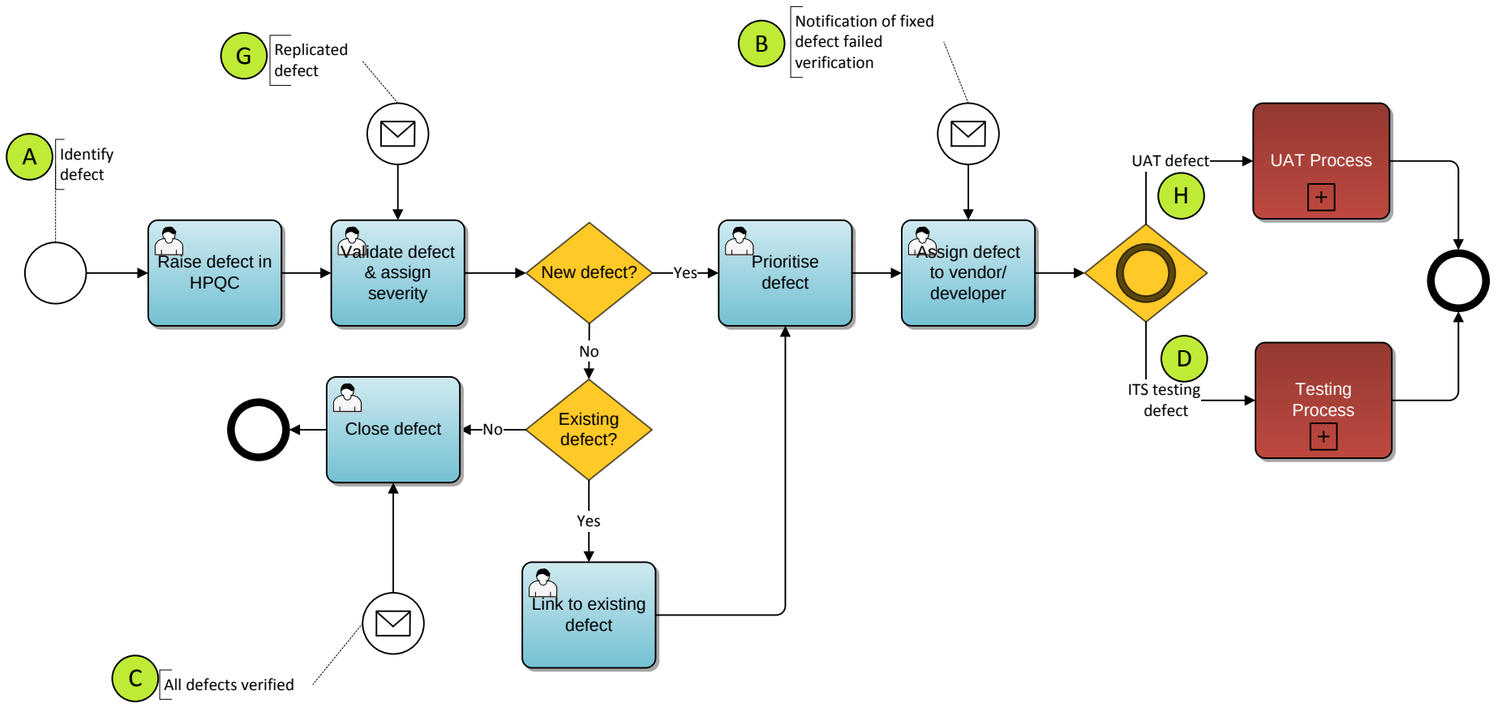
Severity

Severity	Description
1 – Critical	Defect is a show stopper or causing critical loss of business functionality.
2 – Major	Defect is causing major impact to business functionality with no possible interim workaround.
3 – Minor	Defect is causing minor impact to business functionality but is tolerable with possible interim workaround.
4 – Trivial	Defect is causing minor inconvenience only and functionality is not impacted.
5 – Exception	Defect is the result of non-conformance to a standard, is related to the aesthetics of the system, or is a request for an enhancement. Defects at this level may be deferred or even ignored.

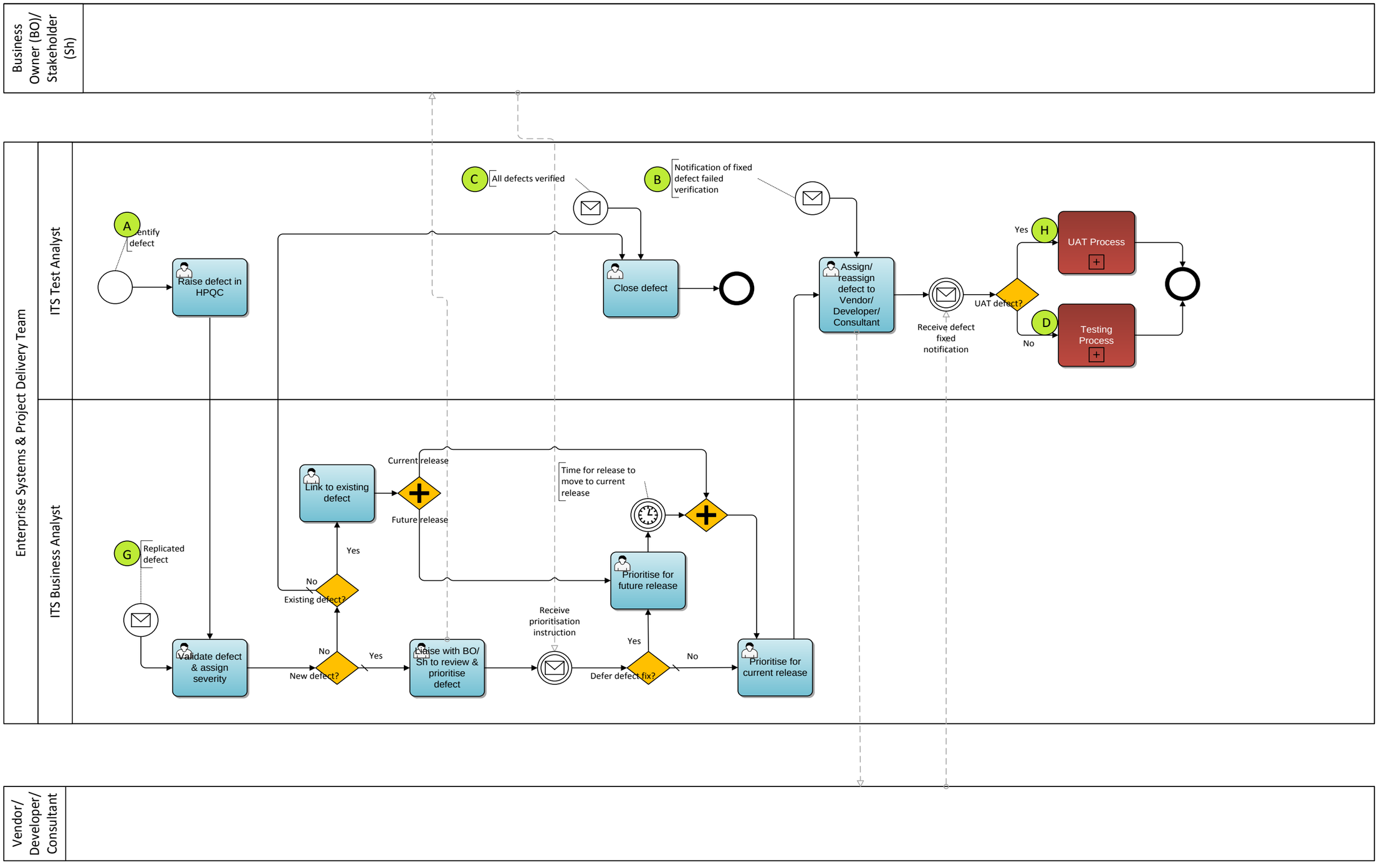
Priority

Priority	Description
1-Resolve Immediately	Further testing cannot occur until the defect has been fixed. The system cannot be used until the fix has been effected.
2-Give High Attention	The defect must be resolved as soon as possible because it is impairing testing activities. System use will be severely affected until the defect is fixed.
3-Normal Queue	The defect should be resolved in the normal course of development activities. It can wait until a new build or version is created.
4-Low Priority	The defect is an irritant which should be fixed but can be fixed after more serious defects have been fixed.
5-Defer	The defect fix can be put off indefinitely. It can be resolved in a future major system revision or not resolved at all.

Defect Management Process [Illustration without roles & detail]



Defect Management Process [Illustration with roles & detail]



User Acceptance Testing (UAT) & Defect Reporting

User Acceptance testing is undertaken by the users following the completion of testing undertaken by the ITS test team. This is an integral part of the quality control as it assists in making sure that the functionality and specifications of the developed product meets users' expectations. It is recommended that users create their own test cases/scenarios based on typical\expected usage of the system. The test analyst can provide assistance such as training and guidance, but should not influence the users as the objective of UAT is to allow for serendipity.

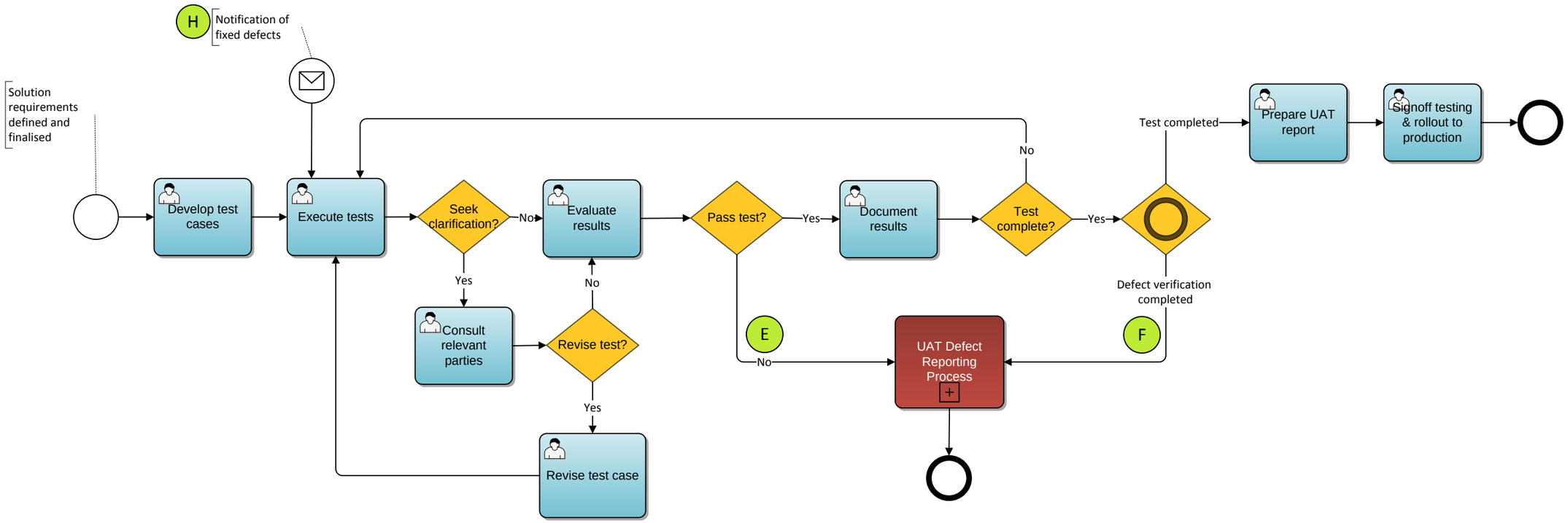
Users will provide daily progress reports on their testing to ensure visibility of how the testing progress is tracking and if any assistance is required.

Defects are to be submitted/raised as soon as they are discovered so that the ITS Test Team can address the defects sooner rather than later and assign the defects to the vendor/developer/consultant for resolution as soon as possible. Users who have access to ITS testing tool Hewlett Packard Quality Center (HPQC) can raise the defects directly in the tool. Alternatively, a UAT defect reporting form (attached in Appendix 1) is to be used and submitted to ITS.

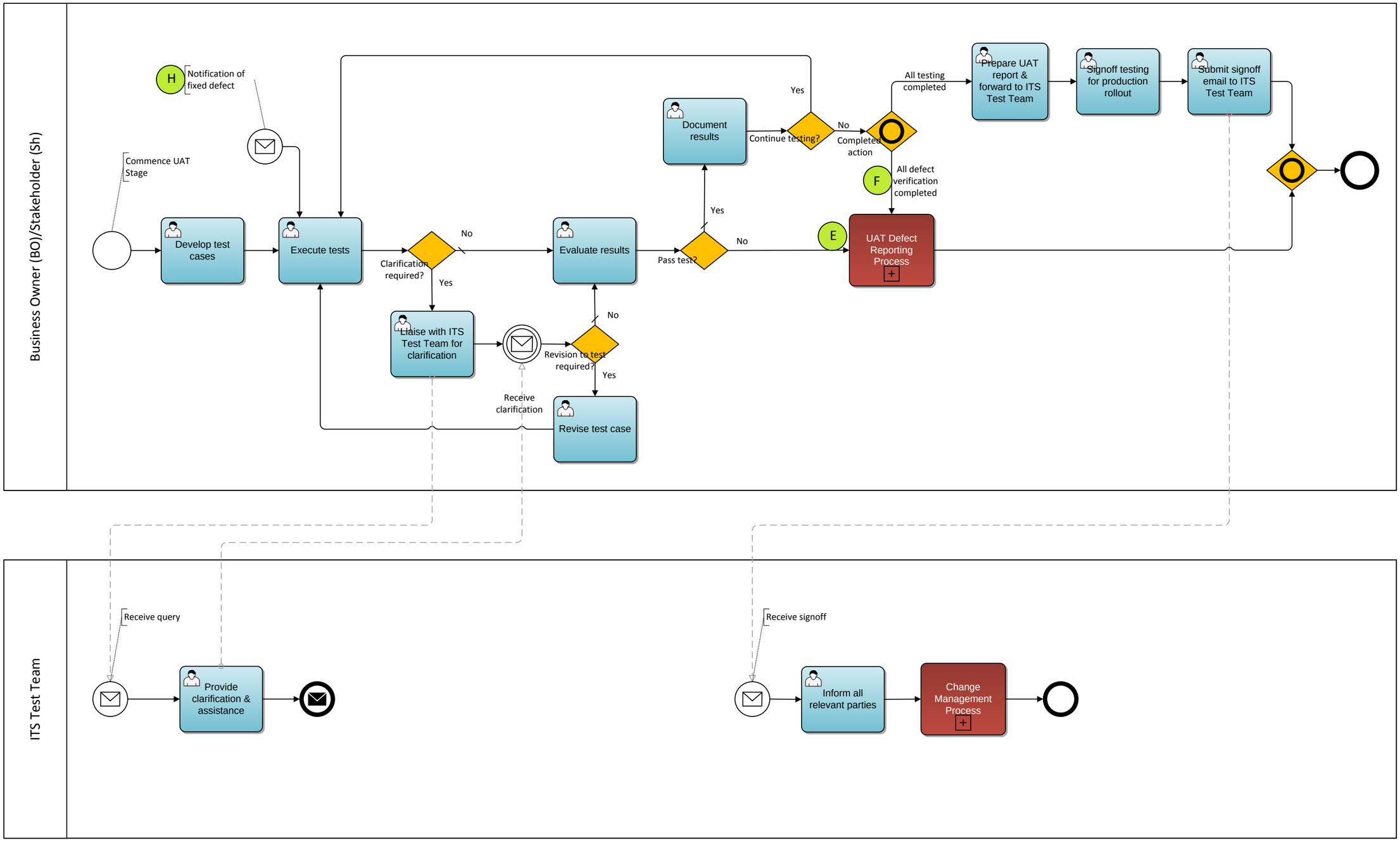
Defects raised, will be replicated, analysed and validated by the ITS Business Analysis and Test Team prior to assigning the defect to the vendor/developer/consultant.

Upon completion of UAT, a UAT completion report and an email indicating signoff of UAT and signoff for production rollout is to be submitted to ITS Test Team for their notification to the relevant parties to commence the required activities for production rollout.

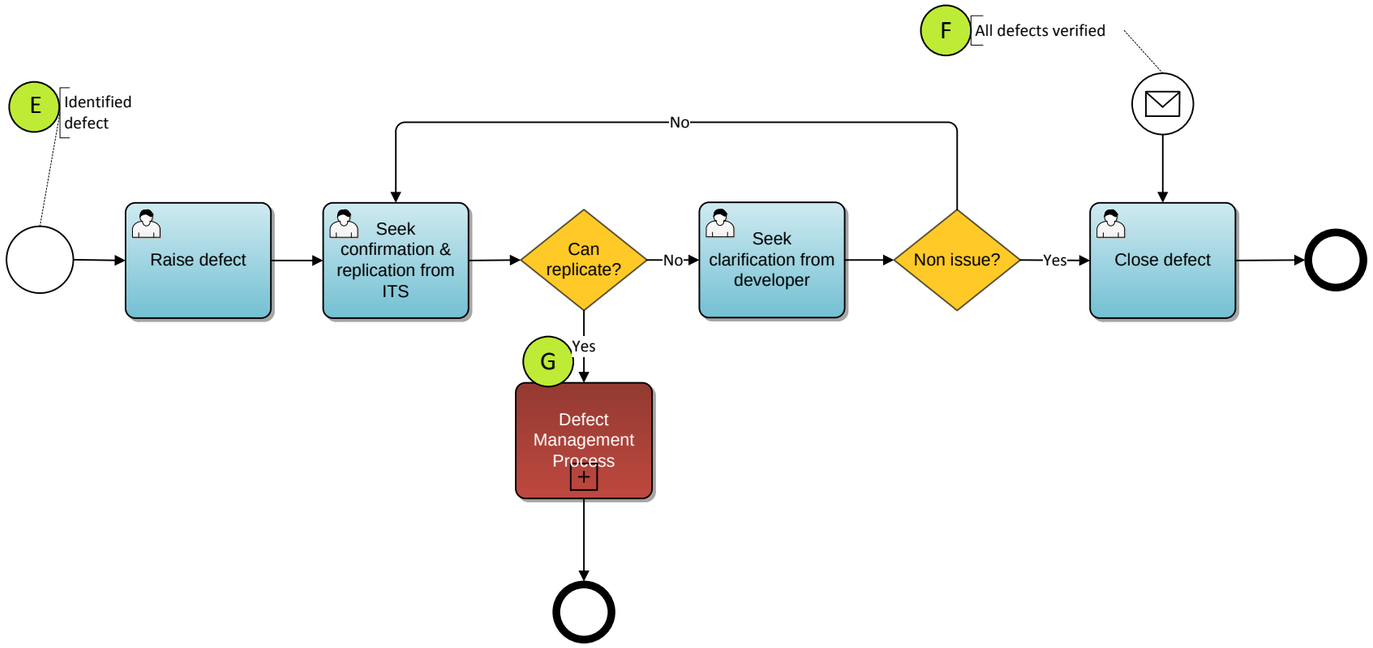
User Acceptance Test (UAT) Process [Illustration without roles & detail]



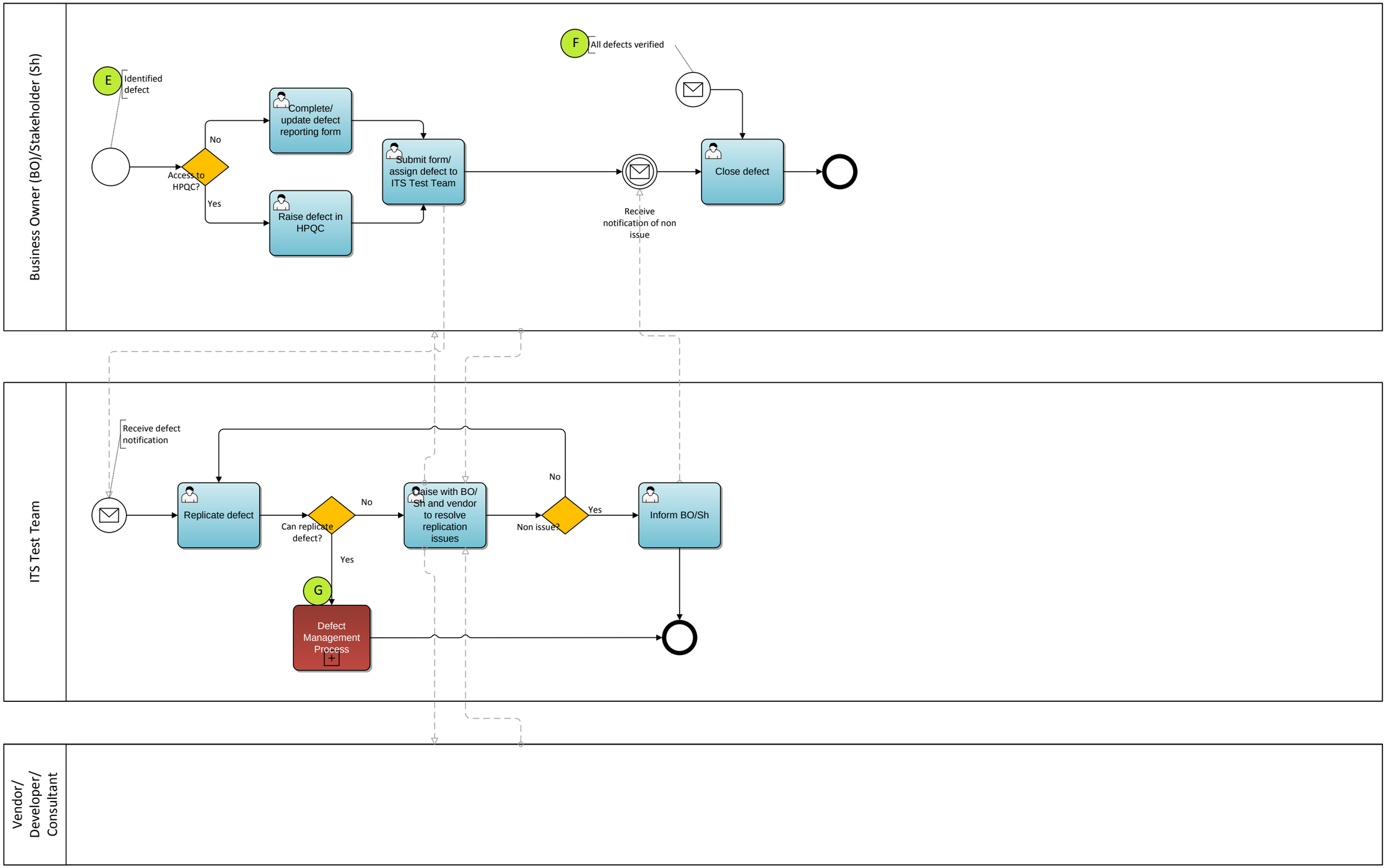
User Acceptance Testing (UAT) Process [Illustration with roles & detail]



UAT Defect Reporting Process [Illustration without roles & detail]



UAT Defect Reporting Process [Illustration with roles & detail]



Communication and Progress Reporting

Communication is a vital component in any activity. Communication keeps all parties involved, informed and aware of progress and, enables quick resolutions and informed decision making on testing issues.

Throughout the testing lifecycle, the Test Analyst will communicate collaboratively with all relevant parties to verify any ambiguity. The Business Analyst will be the initial point of contact to clarify requirements/specifications where necessary, with stakeholders and collaborate with the testing team to bring across a consistent understanding of the requirements/specifications.

A testing progress report and a defect tracking report will be provided to all relevant parties at the end of each day to enable visible analytics on the progress of tests and defects.

Testing Signoff

There are six signoff stages in the testing phase which will be submitted via email to ITS Test Team.

1. Unit testing signoff
2. Signoff on development rollout
3. Signoff on test/acceptance rollout
4. ITS testing signoff
5. UAT testing sign off
6. Signoff on production rollout

Vendor/Consultant/Developer Rules of Engagement

The vendor/consultant/developer will operate within the boundaries of the vendor management framework. The principles within the framework that underpin the rules of engagement are:

Principles

- Open and communicative process
- Formal agreed contract
- Quality service
- Appropriate and timely management of risks, issues and communication

Deliverables

- Formal contractual agreement
- Performance and status reports
- Agreed metrics
- Escalation process
- Issue/change management process

Reporting and Analytics – Mandatory Items

- Performance and issues
- Risks
- Financial
- Services (availability) e.g. targets/work packages
- Reporting frequency

Roles and Responsibilities

Roles and responsibilities are defined using a RACI matrix. The RACI matrix describes the roles of those involved in the respective activities. It describes roles as having one or more of the following responsibilities for a given task or deliverable.

RACI Key

[R]esponsible does the work to ensure that the action/activity is completed.

[A]ccountable is ultimately responsible for ensuring completion of a function, activity, or decision. Only one position should be accountable for every action or decision.

[C]onsulted must be consulted prior to the work and gives input.

[I]nformed means that they must be notified of the outcome.

RACI Matrix

Tasks	Roles											
	Vendor/Consultant/Developer	Business Sponsor	Business Owner/Stakeholder	ES Application Development	ITS Test Analyst	ITS Business Analyst	ITS Project Manager	Vendor Project Manager	ITS Infrastructure	ITS Service Delivery & Engagement	Service Divisions	Colleges /Schools
Service Integrity		**A**										
Product Induction												
Preparation & execution			RA		C	C	I	I				
Testing Tool Setup												
Release module setup					C	RA	I					
Requirements module Setup					C	RA	I					
Testing module setup					RA	C	I					
Test lab setup					RA	C	I					
Defects module setup					RA	C	I					
Test Environments												
Setup & access development environment	RA		I	C	I	I	I	I	C			

Setup & access test/acceptance environment	RA		I	C	I	I	I	I	C			
Setup & access production environment	RA		I	C	I	I	I	I	C			
Setup & access user acceptance test (UAT) environment (optional)	RA		I	C	I	I	I	I	C			
Setup & access automated test environment (where applicable)	RA		I	C	I	I	I	I	C			
Rollout to development	RA		I	C	I	I	I	I	C			
Rollout to test/acceptance	RA		I	C	I	I	I	I	C			
Rollout to production	RA		I	C	I	I	I	I	C	I	I	I
Quality Assurance Activities												
Unit testing signoff	RAC		I		I	I	I	I				
Populate initial testing data	RA		I	C	I	I	I	I				
User account setup	RA		I	C	C	I	I	I				
Email account setup	RA		I	C	C	I	I	I				
Smoke testing	I		I		RA	C	I	I				
Sanity testing	I		I		RA	C	I	I				
Types of Testing Execution												
Unit testing	RAC		I		I	I	I					
Systems integration testing			I	C	RA	C	I	I	C		C	C
Solution requirements testing			C		RA	C	I	I				
Regression testing			C		RA	C	I	I				
Load testing	C		C	C	RA	C	I	I	C			
Performance testing	C		C	C	RA	C	I	I	C			
Stress testing	C		C	C	RA	C	I	I	C			
Production verification testing			RA		R	C	I	I				
Testing Process												
Develop user stories/use cases					CI	RA						
Develop test cases					RA	C						
Prepare test plan					RA	C	I					
Prepare test artefacts					RA	C	I					
Acceptance of test plan			RA		C	R	R					
Execute tests	I		I		RA	C	I	I				
Evaluate & clarify results	C		RA		C	R	I	I				
Raise defects	C				RA	C	I	I				
Prepare progress reports	I		I		RA	C	I	I				
Prepare test results & evaluation report			I		RA	C	I	I				
Signoff ITS testing	I		I		RA	C	I	I				
Defect Management												
Identify defects	I				RA	C	I	I				
Validate & prioritise defects	I		RA		C	R	I	I				
Analyse defects	I		RA		C	R	I	I				
Resolve defects	RA		C		C	C	I	C				
Verify fixed defects	I		I		RA	C	I	I				

Close defects	I		I		RA	C	I	I				
User Acceptance Testing												
Develop test cases			RA		C	C	I					
Execute tests			RA		C	C	I					
Raise defects			RA		C	C	I					
Verify fixed defects	I		RA		RC	C	I	I				
Close defects	I		RA		C	C	I	I				
Prepare UAT progress reports	I		RA		I	I	I	I				
Prepare UAT test results & evaluation report (if required)	I		RA		C	C	I	I				
Signoff UAT	I	I	RA		I	I	I	I				
Communication												
Vendor reporting requirements as per formal contract and SLAs	RA	I	C		C	C	C	C				
ITS reporting requirements as per formal contract and SLAs	C	I	C		RA	C	C	C				
Notify ITS test team of fixed defects/changes	RA		I		C	C						
Notify vendor of defects/changes	C		I		RA	C						
Risk Assessment												
Identify & document testing risks	C		C		RA	C	I	I				
Update testing risk register	I		I		RA	C	I	I				
Signoff on Rollout												
Signoff on development rollout	RA		I		I	I	I	I				
Signoff on test/acceptance rollout	RA		I		I	I	I	I				
Signoff on production rollout	I	I	RA	I	C	C	I	I	I	I	I	I

Risks Assessment

Risks will be identified, assessed and documented along with appropriate mitigation action. Risks will also be reviewed during the testing cycle to address any changes and required modifications to the risk indicators and mitigation strategies. Where applicable, these risks will form part of the primary risk register and will be updated throughout the duration of the testing cycle. The testing risk register template (examples included) is attached in Appendix 1.

Templates

Testing templates are available on the ITS Website.