SPEC INNOVATIONS

DEVELOPERS OF INNOSLATE

GUIDE

Verification and Validation for Data-Driven Systems Engineering

JOHN SCHATZ III | JOHN.SCHATZ@SPECINNOVATIONS.COM

Verification and Validation

Verification and Validation (V&V) are independent evaluation processes for determining a system's conformance to requirements and suitability for use. V&V occurs on the right side of the Systems Engineering (SE) V-Model, as shown in Figure 1.

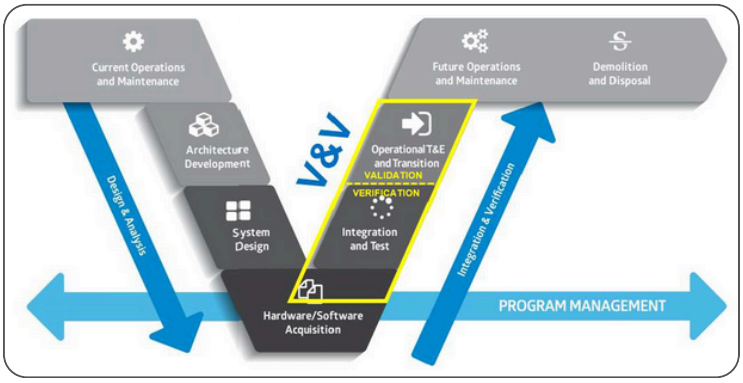


Figure 1. V&V on the Systems Engineering V-Model

Verification

Verification is system-focused, proving that the solution was built according to agreed-upon specification-level requirements. It shows consistency between design decisions and the assumptions underlying requirements. Verification seeks to answer "Are we building the product right?"[1]

Verification occurs prior to validation. It is frequently associated with the terms "Unit Test" and "Developmental Test and Evaluation."

Verification methods are the means by which requirements can be verified. They include:

- **Analysis:** mathematical modeling and analytical techniques used to predict design suitability or performance based on calculated data or data derived from lower-level testing; generally used when other methods are not cost-effective.
- **Demonstration:** system or lower-level operation used to show that a requirement can be achieved; verifies high-level functionality, lacks the detailed data associated with testing.
- **Inspection:** visual examination of design features or identifiable markings.
- **Modeling and Simulation:** certified models and/or simulations used to predict design suitability or performance; can be considered a subcategory of analysis, generally used when other methods are not cost-effective.

- **Testing:** system or lower-level operation used to obtain detailed data to verify performance or to provide sufficient information to verify performance through further analysis; verifies detailed functionality.

The verification methods selected must be executable within program constraints (e.g., time, budget, resources).

Validation

Validation is operationally focused, proving that solution-independent requirements are satisfied. It addresses stakeholder satisfaction and helps to ensure that the system will ultimately be part of the accepted solution within the target environment. Validation seeks to answer, "Are we building the right product?"[2]

Validation occurs after verification, frequently associated with the term "Operational Test and Evaluation."

Validation methods are the means by which stakeholder satisfaction with the system can be validated. They include:

- **Formal and Informal Reviews:** reviews of the system and supporting operational procedures to predict suitability for performing operational concepts.
- **Modeling and Simulation:** certified models and/or simulations used to predict effectiveness and suitability for performing operational concepts.
- **Formal and Informal Demonstrations:** system-level operation in a relevant environment designed to show that the system satisfies stakeholder expectations; lacks the detailed data associated with operational tests.
- **Functional Analysis:** analytical techniques used to predict effectiveness and suitability for performing operational concepts.
- **Operational Tests:** system-level operation in a relevant environment, designed to obtain detailed data necessary to show that the system satisfies stakeholder expectations.

The validation methods selected must be executable within program constraints (e.g., time, budget, resources).

Test Planning

Test Preparation

Test Plans

Test plans describe the orchestration and intended execution of tests from a program perspective. Test plans should minimally include the following items:

- Test objectives
- Test cost, schedule, and risks
- Resources and test support requirements
- Items to be tested
- Testing approach
- Data collection requirements

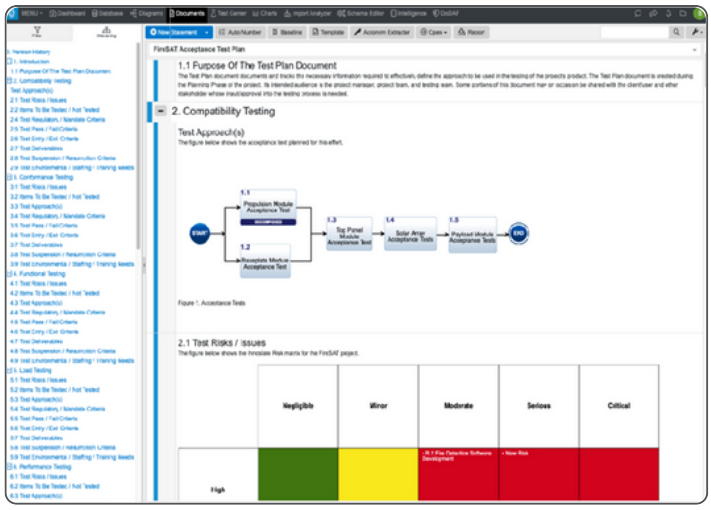


Figure 2. Documents View of Test Plan

Innoslate implements test plans as Innoslate Documents, shown in Figure 2. In Documents View, create a new document with the 'Test Plan Document' type. Select a preloaded template or enter your own format. Populate the test plan with modeled content as described in Sections "Test Plan Models" and "Modeling With Test Cases."

Test Plan Models

Innoslate provides a range of modeling capabilities that can be used to generate diagrams, charts, and tables for incorporation into a test plan. A summary of potential test plan models is shown in Table 1 in the Appendix.

Requirements and Verification

Requirement Verification

A requirement is verifiable when you can "express the expected performance and functional utility so that verification is objective and preferably quantitative."[3] In Innoslate, requirements are verified through the execution of Test Cases.

There are two implementation approaches to make a requirement verifiable, as shown in Figure 3:

- **Approach 1: Ensure Original Requirement Verifiable**: Implemented in Innoslate with Requirement 'verified by' Test Case(s).

- **Approach 2: Create Separate Verification Requirement**: Implemented in Innoslate with Requirement 'verified by' Verification Requirement and Verification Requirement 'verified by' Test Case(s).

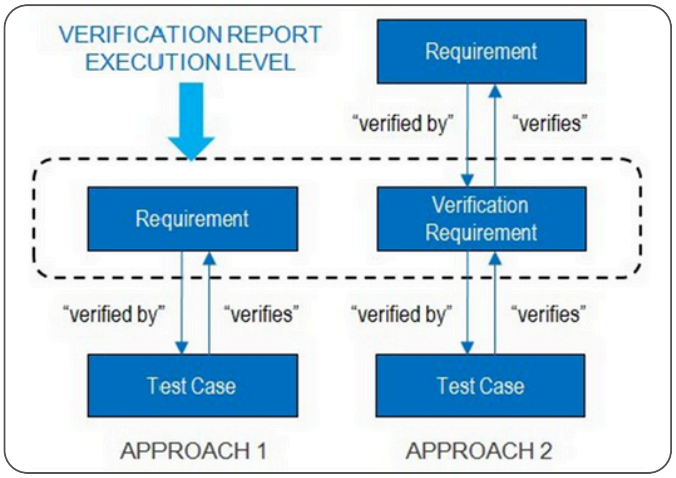


Figure 3. Innoslate Verification Approaches

Verification methods, as described in the "Verification" Section, should be determined as part of the requirements development process. To specify a requirement's verification method, add the corresponding Verification Method Label(s) through the Requirement Metadata sidebar, as shown in Figure 4. For Approach 1, Verification Method Labels are added to Requirement entities. For Approach 2, Verification Method Labels are added to Verification Requirement entities.

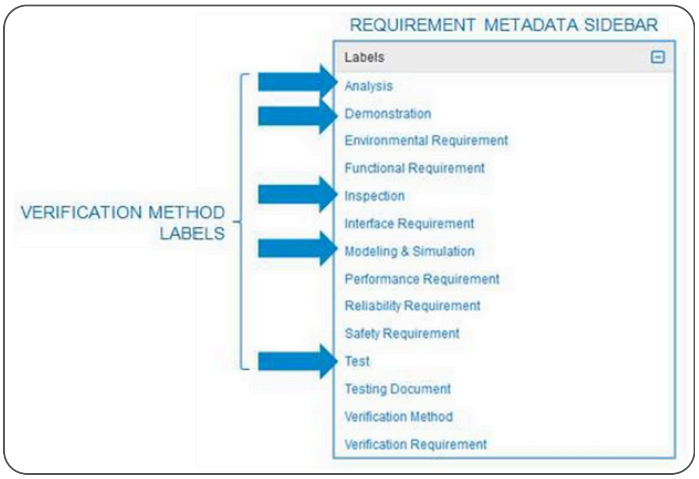


Figure 4. Verification Method Labels in Sidebar

Requirement Verification Reports

Requirement verification reports should be run from the verification approach appropriate execution level (i.e., Requirements document for Approach 1, Verification Requirements document for Approach 2), as shown in Figure 3.

The Verification Cross Reference Matrix (VCRM) shows the verification method that is associated with each requirement. From within Documents View, filter the document by 'Only Requirements.' Select 'Reports' and report type 'VCRM Output (XLSX).' Choose the desired VCRM column options, enter the file name, and select 'Create.' Table 2 in the Appendix shows the VCRM report.

The Requirements Verification Traceability Matrix (RVTM) shows the verification method and associated test case(s) for each requirement. From within Documents View, filter the document by 'Only Requirements'. Select 'Reports' and report type 'RVTM Output (XLSX)'. Choose the desired RVTM column options, enter the file name, and select 'Create'. The RVTM report is shown in Table 3 in the Appendix.

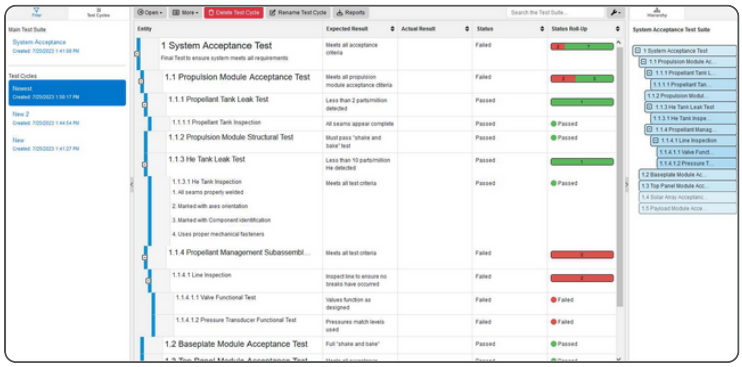


Figure 5. Test Suite View

Developing Tests

Test Management

Test management within Innoslate occurs in the Test Center. The Test Center Dashboard is used to create and manage Test Suites. A Test Suite is an Artifact entity representing a test event. Separate Test Suites should be created for different V&V events (e.g., Unit Tests, Component Tests, System Tests, Operational Test and Evaluation).

The Test Suite View renders the hierarchical collection of related Test Cases comprising the Test Suite, as shown in Figure 5. It supports test management, the visualization of test results, and progress tracking. Test Suite View allows the creation of Test Cycles, which persist a snapshot of the cycle's test results.

Test Cases

Test Cases are tests that occur within a test event. Test Case is a subclass of Action with the additional attributes of Expected Results, Actual Results, Status, and Setup. The Test Case 'Status' attribute is visualized and aggregated within the Test Suite View, as shown in Figure 6.

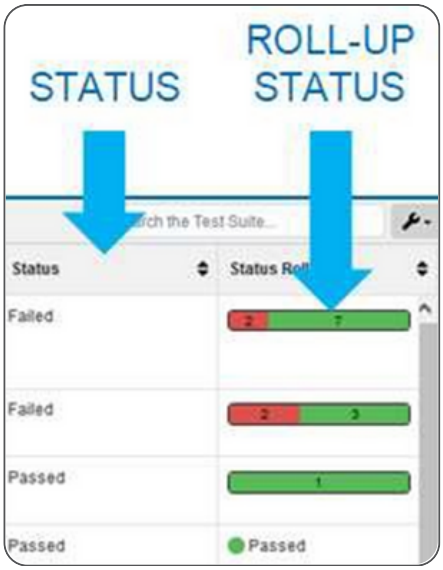
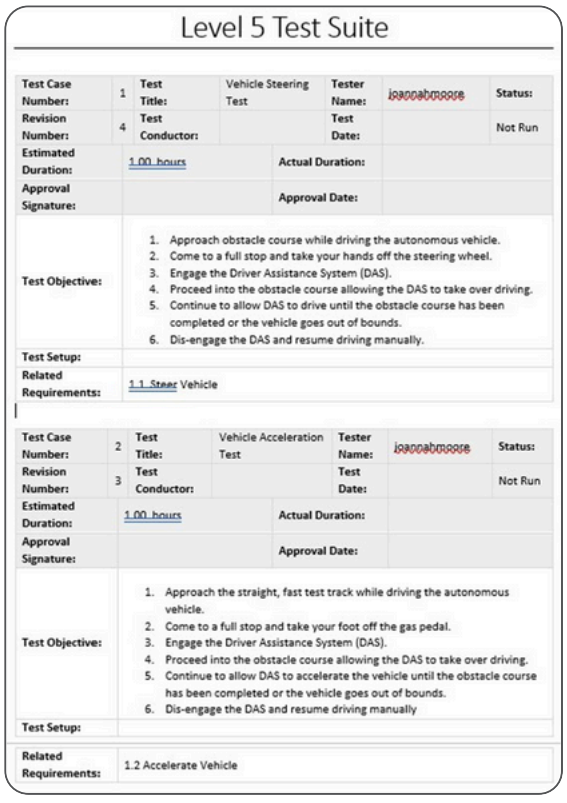


Figure 6. Test Case 'Status'

Requirements are verified through the execution of Test Cases. Priority should be given to ensuring that a Test Case's description, setup information, and expected results are created during V&V planning in coordination with requirement developers.

The Test Cases Output (DOCX) report provides a comprehensive overview of the Test Cases contained within a Test Suite. From within Test Suite View, select 'Reports' and report type 'Test Cases Output (DOCX).' Choose the desired Test Case attribute options, enter the file name, and select 'Create.' The Test Cases Output (DOCX) report is shown in Figure 7.



Level 5 Test Suite

Test Case Number:	1	Test Title:	Vehicle Steering Test	Tester Name:	joannabmoore
Revision Number:	4	Test Conductor:		Test Date:	
Status:	Not Run				
Estimated Duration:	1.00 hours		Actual Duration:		
Approval Signature:			Approval Date:		
Test Objective:	1. Approach obstacle course while driving the autonomous vehicle. 2. Come to a full stop and take your hands off the steering wheel. 3. Engage the Driver Assistance System (DAS). 4. Proceed into the obstacle course allowing the DAS to take over driving. 5. Continue to allow DAS to drive until the obstacle course has been completed or the vehicle goes out of bounds. 6. Dis-engage the DAS and resume driving manually.				
Test Setup:					
Related Requirements:	1.1 Steer Vehicle				

Test Case Number:	2	Test Title:	Vehicle Acceleration Test	Tester Name:	joannabmoore
Revision Number:	3	Test Conductor:		Test Date:	
Status:	Not Run				
Estimated Duration:	1.00 hours		Actual Duration:		
Approval Signature:			Approval Date:		
Test Objective:	1. Approach the straight, fast test track while driving the autonomous vehicle. 2. Come to a full stop and take your foot off the gas pedal. 3. Engage the Driver Assistance System (DAS). 4. Proceed into the obstacle course allowing the DAS to take over driving. 5. Continue to allow DAS to accelerate the vehicle until the obstacle course has been completed or the vehicle goes out of bounds. 6. Dis-engage the DAS and resume driving manually				
Test Setup:					
Related Requirements:	1.2 Accelerate Vehicle				

Figure 7. Test Cases Output (DOCX) Report

Modeling With Test Cases

Since Test Cases are a subclass of Action, they can be viewed and manipulated in any diagram available to Actions. Test procedures can be created from a Test Suite's Test Cases. Test procedure modeling can be used to:

- Establish detailed test procedure schedules.
- Estimate test procedure resource utilization and cost.

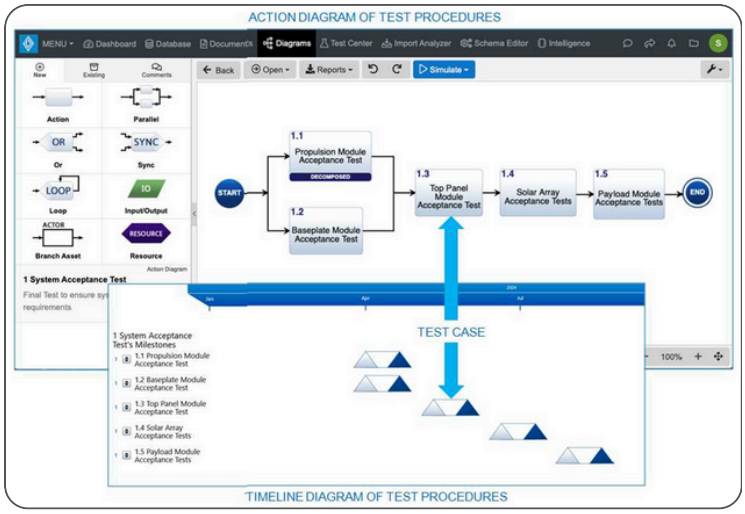


Figure 8. Test Procedure Schedule Model

To establish detailed test procedure schedules, first open any decomposed Test Case as an Action Diagram. Then, arrange the decomposing child Test Cases to orchestrate the test process. Finally, open the decomposed Test Case as a Timeline Diagram to view and adjust the corresponding test

schedule. Figure 8 shows a test procedure schedule model.

To estimate test procedure resource utilization and cost, add resource and cost information to the modeled test procedures. Then run a Monte Carlo Simulation from within the test procedure's Action Diagram.

Expected resource utilization and cost can be determined from the values displayed on simulator panels (e.g., Resource (Radar), Cost Bar Chart, Status) and within simulation reports (e.g., Monte Carlo Resource Report, Monte Carlo Cost Report).

Test Execution

Facilitating Tests

The Innoslate project dashboard is the first page displayed when a user accesses a project. Project dashboard widgets can be customized to facilitate V&V by providing testers with situational awareness and one-click access to test resources. The suggested dashboard uses in support of V&V and their corresponding widgets are shown in Table 4 in the Appendix.

For all users to see the customized widgets, a Project Owner should create the dashboard layout. Each widget added should have the 'Save to Project' switch turned on in its settings. After the customized widgets have been added, the Project Owner should select the 'Save Dashboard Layout' button.

Running Tests

Test Cases are executed in accordance with the Test Procedure Schedule. A Test Case's 'Actual Results' and 'Status' attributes should be filled in during V&V testing. 'Status' values will be rolled up within the Test Suite hierarchy as shown in Figure 6.

Test Cases that are 'Blocked' may generate Issue entities. Relate the Test Case with the associated Issue entity using the 'causes' relationship.

A Test Suite can be run multiple times. After executing and recording the results for each Test Case within the Test Suite, save the Test Cycle. In the Test Suite View, select 'More' and then 'New Test Cycle.' Enter a name

for the Test Cycle and select 'Create'. Saving a Test Cycle persists a snapshot of the test results and resets the Test Case statuses to 'Not Run'. Saved Test Cycles can be selected for viewing on the sidebar's 'Test Cycles' tab.

Reporting Test Results

The TVM Output (XLSX) report provides an overview of a Test Suite's Test Cases, including their corresponding status and verified requirements. In the Test Suite View, select 'Reports' and report type, 'TVM Output (XLSX).' Choose the desired Test Case attribute options to include 'Status', enter the file name, and select 'Create'. The TVM Output (XLSX) report is shown in Table 5 in the Appendix.

Upon test completion, a detailed Test Suite status report can be generated by running the Test Cases Output (DOCX) Report, as described in "Test Cases" Section.

Conclusion

Innoslate supports Verification and Validation activities. It provides a versatile toolset for planning, executing, and reporting on test events.

References

[1] "Verification and Validation: Overview." AcqNotes. March 15, 2024. https://acqnotes.com/acqnote/careerfields/verification-validation.

[2] Larson, Wiley J. Applied Space Systems Engineering. Boston, MA: McGraw-Hill, 2009.

Appendix

Test Plan Model	Entity (Class)	Diagram/Chart/Table
Test Equipment and Facilities	Test Equipment (Asset); Facilities (Asset)	Hierarchy Diagram
Test Organization with Roles/Responsibilties	Test Organization (Asset); Roles/Responsibilities (Statement)	Hierarchy Diagram; Entity Table [Asset 'traced from' Statement]
Risks and Mitigations	Risk (Risk); Mitigation Occurrence (Time); Mitigation (Action/Asset)	Risk Diagram; Risk Burn-Down Chart; Entity Table [Risk 'mitigated by' Action/Asset]
Criteria	Criteria (Characteristic/Measure)	Entity Table [Characteristic/Measure 'specifies' Entity]
Test Procedure Resource Utilization	Test Procedure (Action); Resource (Resource)	Action Diagram with Monte Carlo Simulation
Schedule	Timeline/Task (Task)	Timeline Diagram; Gantt Chart
WBS with Cost	WBS (Statement); Cost (Cost)	Entity Table [Statement 'incurs' Cost]
Data Capture Times	Data (Measure); Time (Time)	Entity Table [Measure 'occurs' Time]
Test Equipment Locations	Test Equipment (Asset); Location (Physical)	Entity Table [Asset 'located by' Physical]

Table 1. Potential Test Plan Models

Requirement		Verification Method(s)				
Number	**Name**	**Analysis**	**Demonstration**	**Inspection**	**Modeling & Simulation**	**Test**
CL.SRD.3.2.1.1	Vitals Measurement		X			
CL.SRD.3.2.1.2	Clinical Care		X			
CL.SRD.3.2.1.3	Medication		X			
CL.SRD.3.2.1.4	Financial Assistance Counseling		X			
CL.SRD.3.2.2.1	Patient Data					X
CL.SRD.3.2.2.2	Operational Data					X
CL.SRD.3.2.2.3	Decision Support Materials					X
CL.SRD.3.2.3.1.1	Reception Area			X		
CL.SRD.3.2.3.1.2	Vital Signs Area			X		
CL.SRD.3.2.3.1.3	Examination Rooms			X		
CL.SRD.3.2.3.1.4	Check Out Area			X		
CL.SRD.3.2.3.1.5	Waiting Room			X		
CL.SRD.3.2.3.1.6	Pharmacy			X		

Table 2. VCRM Report

Requirement				Test Case			
Number	**Name**	**Description**	**Verification Method(s)**	**Number**	**Name**	**Description**	**Status**
CL.SRD.3.2.3.2.1	Cleanliness	The clinic shall adhere to Virginia Department of Health (VDH) guidelines for healthcare facility cleanliness.	Inspection	CLIN.SAT.1.4.1	Cleanliness Inspection	Inspector enters clinic; Inspector verifies that clinic meets VDH guidelines for healthcare facility cleanliness	Passed
CL.SRD.3.2.3.2.2	Medical Equipment Disinfection	The clinic shall adhere to Virginia Department of Health (VDH) guidelines for medical equipment disinfection.	Demonstration	CLIN.SAT.1.4.2	Medical Equipment Disinfection Demonstration	Patient led to vitals station; Medical assistant takes patient's vital signs; Vitals station disinfected	Passed
CL.SRD.3.2.3.2.3	Ventilation	The clinic shall adhere to Virginia Department of Health (VDH) guidelines for healthcare facility ventilation.	Inspection	CLIN.SAT.1.4.3	Ventilation Inspection	Inspector enters clinic; Inspector verifies that clinic meets VDH guidelines for healthcare facility ventilation	Blocked
CL.SRD.3.2.3.2.4	Medical Waste Management	The clinic shall adhere to Virginia Department of Health (VDH) guidelines for healthcare facility medical waste management.	Demonstration	CLIN.SAT.1.4.4	Medical Waste Management Demonstration	Patient let to exam room; Provider delivers clinical care; Medical waste disposed of in accordance with clinic policies	Passed

Table 3. RVTM Report

Item	Description	Widget
Activity Countdown	Provide a countdown clock to test specific activities, such as the start of a test phase or the deadline for test completion.	Countdown Clock
External Test Resources	Provide descriptions of and links to externally provided test resources such as facility/range schedules or simulation capabilities.	Wiki
Post Test Reports	Provide access to reports for completed test phases.	Wiki
Test Organization	Provide test organization roles/responsibilities and contact information in tabular format.	Entity Table
Test Overview	Provide a summary of the current test phase with any specific instructions or notifications.	Wiki
Test Plan	Provide a description and link for the test plan. The description should be short and summarize the items covered in the test plan.	Document Card or Wiki
Test Schedule	Display a thumbnail with one-click access to the test schedule.	Diagram Card [Timeline Diagram] or Wiki
Test Suite	Display a thumbnail with one-click access to the Test Suite where the tester will be recording results.	Test Suite Card or Wiki

Table 4. Dashboard Uses Supporting V&V

Test Case				Verified Requirement		
Number	**Name**	**Expected Result**	**Status**	**Number**	**Name**	**Description**
CLIN.SAT.1	System Acceptance Test	Meets all acceptance criteria	Failed			
CLIN.SAT.1.1	Clinic Services	Meets all acceptance criteria	Not Run			
CLIN.SAT.1.1.1	Vitals Measurement Demonstration	Vitals station demonstrates capability to measure patient vitals	Passed	CL.SRD.3.2.1.1	Vitals Measurement	The clinic shall measure patient vitals.
CLIN.SAT.1.1.2	Clinical Care Demonstration	Clinic demonstrates capability to provide clinical care	Passed	CL.SRD.3.2.1.2	Clinical Care	The clinic shall provide patient clinical care.
CLIN.SAT.1.1.3	Medication Demonstration	Clinic demonstrates capability to dispense medication	Not Run	CL.SRD.3.2.1.3	Medication	The clinic shall dispense medication.
CLIN.SAT.1.1.4	Financial Assistance Counseling Demonstration	Clinic demonstrates capability to provide financial assistance counseling	Not Run	CL.SRD.3.2.1.4	Financial Assistance Counseling	The clinic shall provide financial assistance counseling to patients that are uninsured, under insured, or in other need of financial assistance.
CLIN.SAT.1.2	Clinic Information Systems	Meets all acceptance criteria	Failed			

Table 5. TVM Output (XLSC) Report