

white paper

IMPLEMENTATION TESTING IS OVER Now What?

written by
Greg Linaker



The Problem

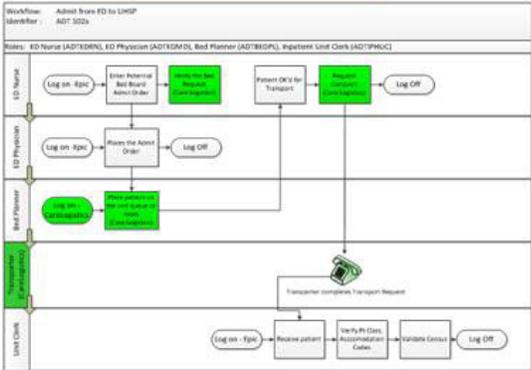
You have just released an EMR upgrade that fixes several issues that have been a problem for your organization for quite some time. In the beginning, all seems well and good, but then the Help Desk starts to get flooded with issues that you were not even aware were part of the upgrade. Then you start getting calls from angry Directors and Vice Presidents asking “did you even test the system before you added the new fixes?” You are thinking “where did we go wrong?” Let us examine some practices from the categories below:



The Solution

Planning. In order to conduct good testing, you need to make sure you plan well in advance. During this planning phase, you need to **analyze the upgrade notes** and any new scripts that the vendor may provide. When analyzing the new upgrade, identify the existing workflows/processes and other external systems that will be impacted and the timeline for the testing phase. You will also want to **start developing test cases and scripts** ensuring that you are testing the entire workflow along with any external systems. If you just develop the test cases and scripts to address the new functionality, chances are you are not going to catch something that could impact the end user in a big way. You then want to **communicate** these new scripts to your IT organization, the training team and the end-users that are impacted. It is good practice to communicate with the same set of documentation, for instance a workflow diagram, the test script and scenario and a document which explains it in a more narrative way to the end user. This is illustrated in the following diagram.

Diagram



Script

Step	Type	Go To	Application	Step No.	Step Name	Step Description	Expected Results	Actual Results	Pass/Fail
1	Test	Start	EMR	1	Log on - Epic	Log on to Epic	Successful login		
2	Test	Start	EMR	2	Enter Patient Bed Board Admit Order	Enter patient information and bed board order	Order is entered		
3	Test	Start	EMR	3	Verify the Bed	Verify the bed availability	Bed is available		
4	Test	Start	EMR	4	Review the Admit Order	Review the admit order	Order is reviewed		
5	Test	Start	EMR	5	Verify the Bed	Verify the bed availability	Bed is available		
6	Test	Start	EMR	6	Place Transport Request	Place transport request	Request is placed		
7	Test	Start	EMR	7	Receive Patient	Receive patient	Patient is received		
8	Test	Start	EMR	8	Verify Pt Class	Verify patient class	Class is verified		
9	Test	Start	EMR	9	Verify Census	Verify census	Census is verified		
10	Test	Start	EMR	10	Log Off	Log off	Successful log off		

Document

SOUTHWESTERN
University, Hospital & Clinics

Workflow: Admit from ED to LIMP
Version: 1.0
Created: 01/15/2013

OBJECTIVE: This workflow describes the steps used to admit an emergency room patient from the LIMP to the ED to the LIMP.

SCOPE:

- 1. Log on - Epic
- 2. Enter a patient's bed board admission order
- 3. Log on - Epic
- 4. Review the admit order
- 5. Log off

PREREQUISITES:

- 1. Log on - Epic
- 2. Review the patient's bed board admission order
- 3. Log on - Epic
- 4. Review the patient's bed board admission order
- 5. Log off

STEPS:

1. Log on - Epic
2. Enter a patient's bed board admission order
3. Log on - Epic
4. Review the admit order
5. Log off

APPENDIX:

- 1. Log on - Epic
- 2. Review the patient's bed board admission order
- 3. Log on - Epic
- 4. Review the patient's bed board admission order
- 5. Log off

Property and Confidentiality: 01/15/2013

SOUTHWESTERN
University, Hospital & Clinics

Workflow: Admit from ED to LIMP
Version: 1.0
Created: 01/15/2013

OBJECTIVE: This workflow describes the steps used to admit an emergency room patient from the LIMP to the ED to the LIMP.

SCOPE:

- 1. Log on - Epic
- 2. Enter a patient's bed board admission order
- 3. Log on - Epic
- 4. Review the admit order
- 5. Log off

PREREQUISITES:

- 1. Log on - Epic
- 2. Review the patient's bed board admission order
- 3. Log on - Epic
- 4. Review the patient's bed board admission order
- 5. Log off

STEPS:

1. Log on - Epic
2. Enter a patient's bed board admission order
3. Log on - Epic
4. Review the admit order
5. Log off

APPENDIX:

- 1. Log on - Epic
- 2. Review the patient's bed board admission order
- 3. Log on - Epic
- 4. Review the patient's bed board admission order
- 5. Log off

Property and Confidentiality: 01/15/2013

SOUTHWESTERN
University, Hospital & Clinics

Workflow: Admit from ED to LIMP
Version: 1.0
Created: 01/15/2013

OBJECTIVE: This workflow describes the steps used to admit an emergency room patient from the LIMP to the ED to the LIMP.

SCOPE:

- 1. Log on - Epic
- 2. Enter a patient's bed board admission order
- 3. Log on - Epic
- 4. Review the admit order
- 5. Log off

PREREQUISITES:

- 1. Log on - Epic
- 2. Review the patient's bed board admission order
- 3. Log on - Epic
- 4. Review the patient's bed board admission order
- 5. Log off

STEPS:

1. Log on - Epic
2. Enter a patient's bed board admission order
3. Log on - Epic
4. Review the admit order
5. Log off

APPENDIX:

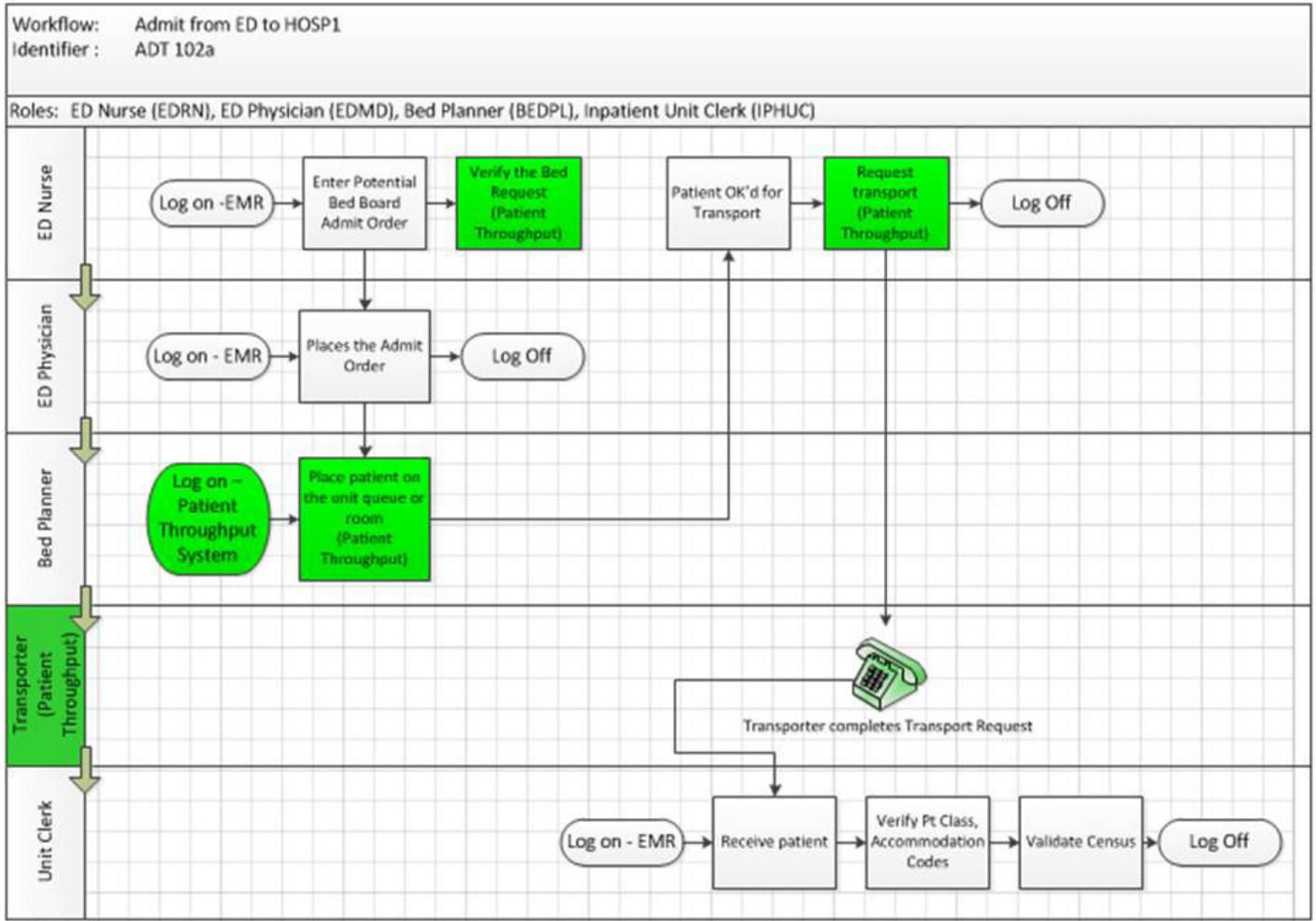
- 1. Log on - Epic
- 2. Review the patient's bed board admission order
- 3. Log on - Epic
- 4. Review the patient's bed board admission order
- 5. Log off

Property and Confidentiality: 01/15/2013

This will help to obtain buy-in from each area and prove to the end users that you are testing the new upgrade. The following images, are illustrations of the three documents with an explanation of the purpose of each.

Workflow Diagram

The diagram employs a role-based swimlane structure and standardized color coding. The **green** color on the diagram indicates interaction with other external systems. You could use the color **yellow** on the diagram to indicate a manual step or process. The diagram will display a high-level overview of the workflow being changed and tested.



Workflow Test Script

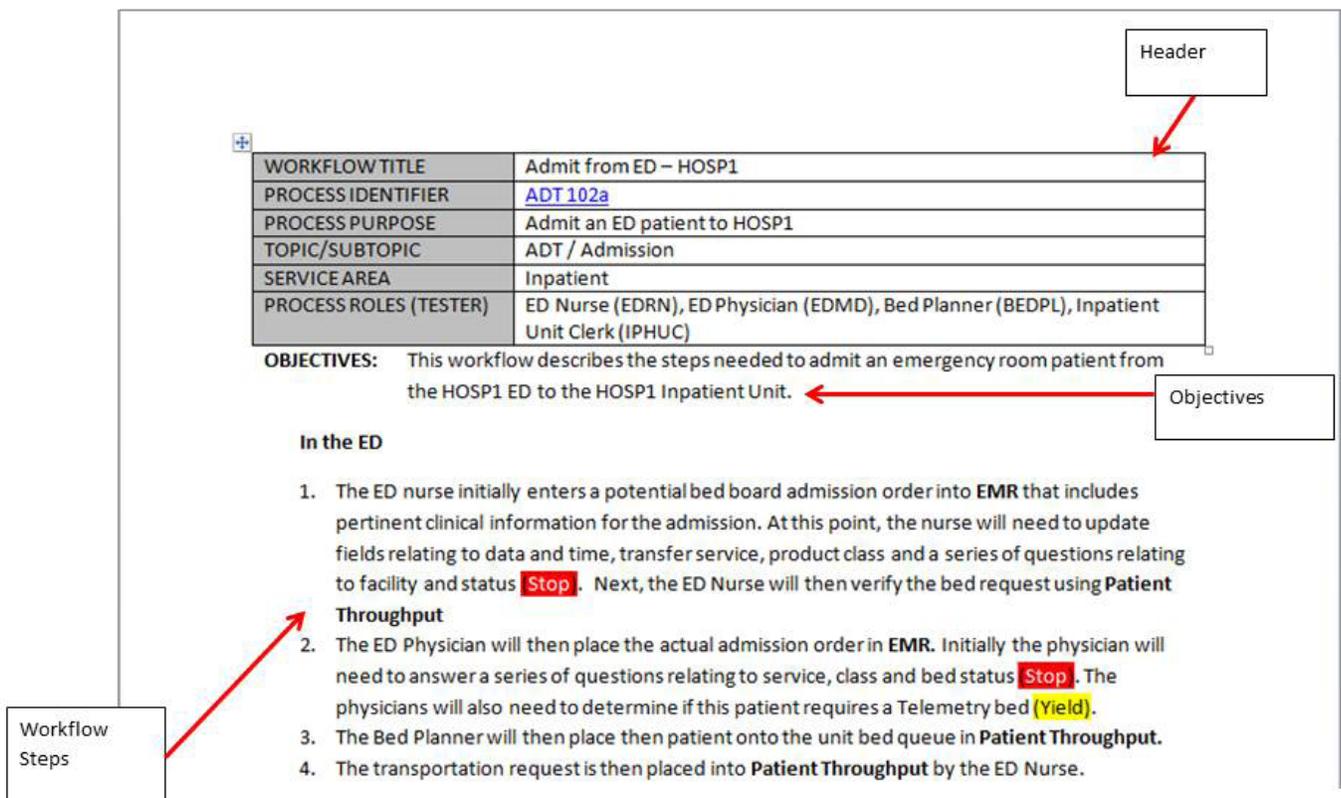
The Workflow Test Script lists the detailed step-by-step actions that need to be done to complete the workflow and which test user needs to perform that step. Notice the variables column. Variables are items that plug into a scenario. For instance, if you need to enter an order for the script, the variable would be the specific order and the user can specify what specific order they want the script tested with. This makes the script more flexible. Notice the Meta Tags column. **Meta tags** are tied to each script so the script can be pulled from a database. For instance, if you want to pull all the scripts pertaining to the ED, as long as the script has a meta tag of ED, it would be listed.

Workflow:		Workflow: Admit from ED to HOSP1									
Identifier:		Identifier: ADT 102a									
Section A: Overview											
Testing Cycle:											
Testing Date:											
Section B: Preparation											
Preparation Steps:		The patient must be an existing registered ED patient.									
Section C: Testing Information											
Departments Used:											
Test Patient:											
Pass/Fail		Untested									
		Variables Column		Metatags Column							
Step	Team	User / Dept	Workflow Step	Testing Step / Process	Expected Outcome	Variables	Metatags	Actual Outcome	Pass / Fail	Com	
1	ED App	EDRN / HOSP1 ED	Log on	Log in to Hyperspace.	The ED Track Board appears		ED, HOSP1				
2	ED App	EDRN / HOSP1 ED	Enter Potential Bed Board Admit Order	Click on the All Patients button from the buttons available in the header.	The screen will now show all patients		Admt, Admission				
3	ED App	EDRN / HOSP1 ED		Select on the desired patient from the tracking board	The ED Narrator window should appear						
4	ED App	EDRN / HOSP1 ED		Click the Orders Tab and enter the and	The preference list						
		User ID in Test System									

Workflow Document - Core Components

The workflow document is meant to summarize the diagram and test script in a more narrative format. This is what should be communicated to everyone when a workflow changes or a new one is created. The core components of this document are listed below along with an example of the components.

- **Header** Identifies the workflow, processes, intent, and the users
- **Objectives** Provides a brief summary statement followed by a user-centric description of the steps detailed in the script for this individual
- **Workflow Steps** Lists the steps each user role needs to perform
- **Required Items** Details any mandatory steps (Stops and Yields in EMR) that are seen in the scripts. This is followed by the Workflow diagram
- **Prerequisites** As workflows may build upon previous actions, this section informs the Tester of what Workflows or actions must occur prior to this workflow being tested
- **Workflow Diagram** Displays the workflow diagram for the workflow
- **Software Applications** Lists the various EMR modules in use as well as any other non-EMR applications
- **Technical Analysis** Informs the Tester of what authorizations must be in place (users roles as well as software access to the EMR and other systems)
- **Manual Steps** Actions such as telephone calls, patient transportation and similar processes are noted here as well as who performs these steps
- **Meta tags** These are key words from within the workflow that can be used to define primary processes and/or services within the workflow (i.e. lab test, patient registration, etc.)



Throughput

- The ED Physician will then place the actual admission order in **EMR**. Initially the physician will need to answer a series of questions relating to service, class and bed status **Stop**. The physicians will also need to determine if this patient requires a Telemetry bed **Yield**.
- The Bed Planner will then place then patient onto the unit bed queue in **Patient Throughput**.
- The transportation request is then placed into **Patient Throughput** by the ED Nurse.
- The patient may be transported by a Transporter, or by the nurse as a self-transport. The nurse will acknowledge the transport within Epic however the transport is a manual step. This then completes the ED portion.

At the HOSP1 inpatient unit

- The inpatient unit clerk receives the patient and accepts the new patient. The unit clerk then verifies class and accommodation codes. The unit clerk will then validate the unit census which completes this workflow.



STOP AND YIELDS		
Category	TESTER	DESCRIPTION
STOP	ED Nurse	Update date and time of transfer, service, class and bed status
STOP	ED Physician	Update date and time of transfer, service, class and bed status
YIELD	ED Physician	Determine if the patient will require a telemetry bed

Workflow Steps



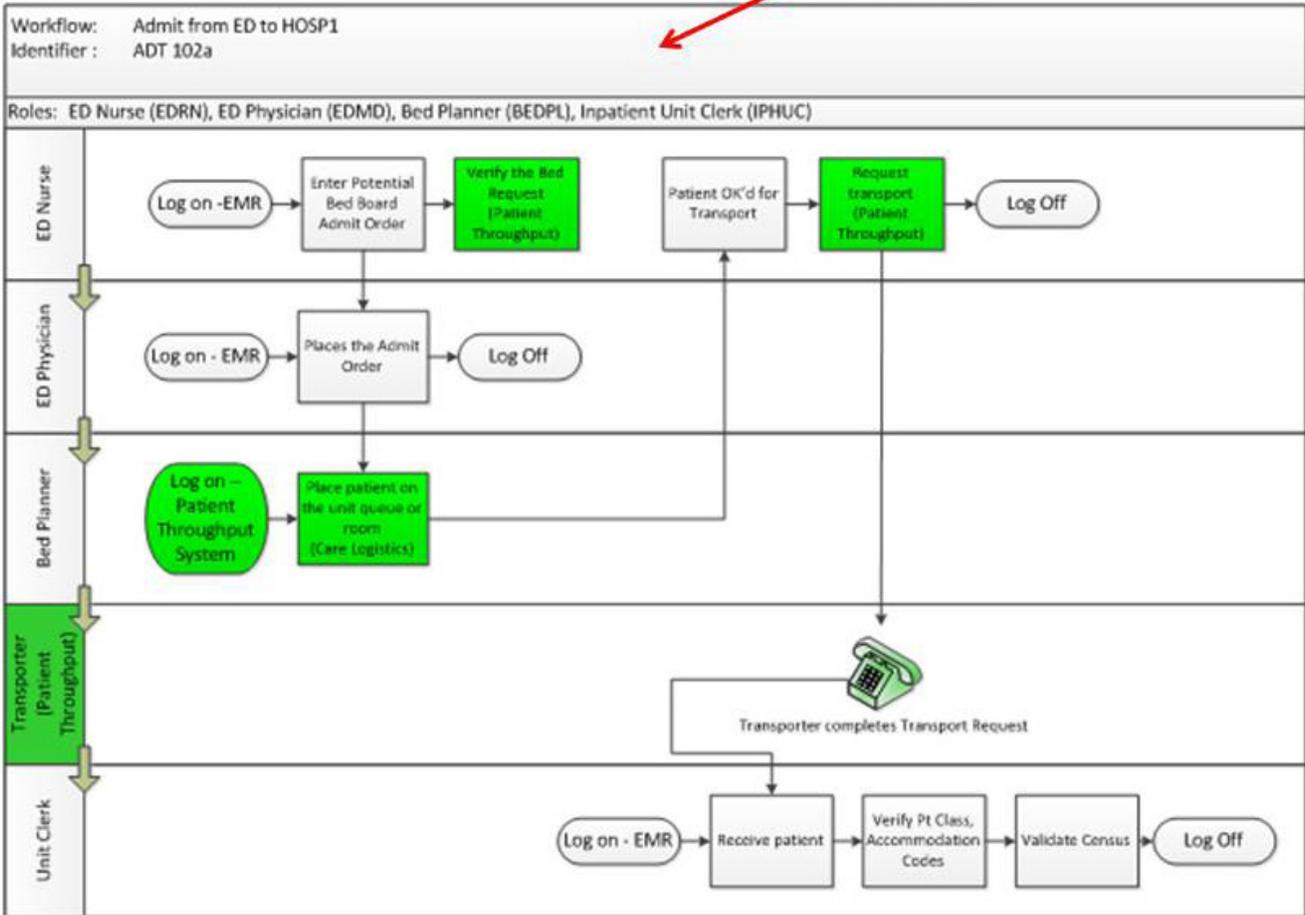
PREREQUISITES: This workflow assumes the following:

- The patient has been registered as an ED patient
- The patient has been admitted as an ED patient



Prerequisites

Workflow Diagram



SOFTWARE APPLICATIONS: This workflow uses both the EMR and Patient Throughput systems

TECHNICAL ANALYSIS: The users have the appropriate credentials to access and the EMR and Patient Throughput systems. The terminals or devices used have access to EMR and Patient Throughput systems.

MANUAL STEPS: Patient transportation to HOSP1 unit

METATAGS:

- Admit
- Admission
- ED
- Emergency Department
- ED Admission
- Potential Admission
- Admit Order
- Patient Throughput
- Bed Management
- Bed Control
- Transport

Preparation

Now that you have a good idea of what the testing will entail, it's important to prepare your environment so that the testing runs smoothly. This includes the following:

- Ensure the testing environment has been cleared out of any previous testing data, physicians, patients, etc. so that you know you are starting with clean testing data.
- Populate the testing environment with physicians, patients, etc. that will fit with your new scripts/scenarios.
- Populate the testing environment with testing IDs that have correct role/security that matches what is currently in production and matches the real world. Due to new functionality being added, security and/or roles may need to be modified.
- Ensure the test interfaces are running to and from external systems and that the testers have the appropriate access to these external systems.
- Identify any external users you want to be involved in the testing and make sure they get any necessary training in order to complete the testing successfully.
- Perform an initial dry-run test with a couple of scripts to ensure the environment is set-up correctly.

It must be emphasized that the training environment and user IDs match the real world; otherwise it's not a real test.

Execution

This is the phase when you perform the testing event with the super users/end users. Send out the schedule early and several times to ensure this communication is being received. You should perform this in a large testing room with a "Manager of the Day" (MOD) to ensure the right people are there at the right time. The testers should document thoroughly what their findings are. From there it needs to be determined if the defect was a build issue, script issue, interface issue or training issue. This information needs to be fed back to the appropriate teams for modifications. It then needs to go through the appropriate Change Control channels so that a formal process is followed for changing the test system. It should be treated just as if it were production. Also, everyone should be made aware of any changes that are being made through Change Control. It should be determined here what needs to be regression tested. You need to repeat the test scripts until they pass without any issues.

Maintenance

Once you are finished with your testing. It is good practice to keep a library of the test scripts so they can be used again as a template with minor modifications. Why let a lot of hard work go to waste. Using taxonomy is a good idea so that you can classify your scripts in ordered categories. For example, you may want an order for all of your Admit, Discharge, Transfer (ADT) scripts into a category with a numbering scheme. You might choose ADT 101 for the first one, ADT 102 for the second and so on. Another good practice for maintaining scripts is to use meta tagging which was illustrated earlier in this document. You can tag your scripts with characteristics of that script. If these are put into a database, you can retrieve needed scripts by searching on their meta tags. For instance, you have a script which covers the workflow of Admitting a Patient to the Hospital (ADT 101). When the script was written, it was tagged with the words Admit, Transport, etc. as

these are all steps in the script. Let's imagine that the transport functionality changes and we need to test all of the scripts that contain the Transport steps. If the scripts are meta tagged correctly, you could simply search through them using the word Transport and see a list of all the scripts that need to be tested. It's the same concept as searching for keywords in a search engine. Many organizations skip the maintenance piece of scripting, which they regret later.

Closing Statement

I hope you can see that I've tried to emphasize the importance of putting time, effort and resources on the front end when it comes to testing with any system at any point. It's better to go through some pain up front, than to have a bunch of fixes after you release the new functionality which also leads to upset and untrusting end users in the end.

ABOUT THE AUTHOR

Greg Linaker is an implementation, Training and Consulting professional with over 18 years experience in the health care industry. Greg's first client in health care was working with VHA and their member hospitals standardizing clinical data across 150 hospitals. He then would assist member hospitals benchmark their data with the other hospitals analyzing various patient populations and outcomes. Greg went on to work with several niche vendors before moving into consulting. Consulting has allowed Greg to work in project management and process/change management for various Epic and Cerner organizations. He also managed several system selection and implementation projects focusing on patient through-put software and was part of one of the first HIE projects in the country.

COMPANY OVERVIEW

Divurgent is not the typical healthcare consulting firm. As a nationally recognized company, we strive to be different, to think outside of the box for innovative healthcare solutions.

Our goal is simple. To transform healthcare to our clients and the communities they serve.

Focused on the business of hospitals, health systems and affiliated providers, Divurgent believes successful outcomes are derived from powerful partnerships.

Recognizing the unique culture that every organization offers, we leverage the depth of our experienced consulting team to create customized solutions that best meet our client's goals. Utilizing best practices and methodologies we help improve our client's operational effectiveness, financial performance and quality of patient care.

OUR COMMITMENT

Divurgent is dedicated to helping our clients improve their operational effectiveness, financial performance and quality of patient care. Through thought leadership, providing value for our services and delivering innovative solutions, we are committed to improving the quality and safety of healthcare delivery for our clients and the communities they serve.

JOIN THE DISCUSSION. CONNECT!

