

## CUSTOMER SUCCESS STORY



## ABOUT THE CLIENT

HEALTHCARE SUBSIDIARY  
of a leading insurance and  
health services company,  
delivering health care  
products and insurance  
services to approximately  
100 million people.



The client needed to deliver a new product on time and ensure defects were eliminated for a combination of a web interface and mainframe-based application. The key for them was to leverage their existing mainframe based data while enhancing the application usability so it could be deployed in pharmacies and used online directly by customers. They selected Conformiq Creator and met their deadline with no surprises and eliminated defects with full visually verified complete test coverage.

## PROJECT OVERVIEW

Mainframe modernization is a broad industry push to upgrade legacy applications by replacing them with new “modern” distributed server systems.

The reasons are many, including difficulty maintaining and enhancing applications, often written in COBOL long ago, where no one knows the code details so enhancements aren't even considered because it's almost impossible to ensure system quality and reliability.

One option is a wholesale replacement but, as you can imagine, the cost can be prohibitive. Another less disruptive and less costly approach is to interface web services and computer interfaces to the mainframe's back-end operations and use advanced testing software to ensure the updated system operations perform as expected.

By using advanced testing technology and methodologies these mission critical, legacy applications can be upgraded with minimal cost and high quality to run on currently available hardware saving millions of dollars in operational costs.

## MODERNIZING LEGACY SYSTEMS

The user interfaces for these mission critical applications can be updated to improve the user experience with modern user interfaces (graphical, mobile, ...) while keeping mainframe back ends.

This means that legacy systems don't need to be decommissioned and replaced in total but more effectively combined for seamless operation. These new interfaces and APIs, plus the mainframe operation with data, need to be tested effectively so full coverage is known and the quality is maintained.

Conformiq's Creator automated test design software does this, thus enabling companies to not abandon their mainframes but modernize them with confidence the new system operation will work properly.



## BUSINESS OBJECTIVES

The company goal is to improve delivery of services by pharmacies and their mutual customers through advanced prescription services ensuring patients get the right medication at the right time at the best cost.

Their current application was designed to enable the pharmacy's customers to “Manage their Medicine Cabinet” anywhere. It is an easy-to-use tool to manage their medications, claims, and orders on any device, whether at home or on the go, send medications right to their home, and get the needed information to find the right drug and pricing options for each user.

The front end of this application is a web-based interface/tool for comprehensive formulary management based on creation and maintenance of formularies using a Medi-Span based product file and other data located on mainframe data servers.

A formulary consists of lists and rule sets with rules set up for various qualifiers. This application uses a rules-based approach to streamline formulary construction and maintenance. Drug data is obtained from an external mainframe-based database and kept current in local application lists using a list expansion process that updates the product information whenever changes are made to the lists and during report generation. These lists can then be queried through an API interface by the external application to determine formulary information for a given drug.

A user can create highly-configurable rules and lists using show history, copy/save as, used in, and other GUI action buttons. Rules automatically handle new products and product changes. A building block concept allows use across formularies as the same rule set can be used in multiple lists/formularies.

The information about this application just serves to highlight that it is a complex application and one that must have complete accuracy of operation against the requirements as well as being defect free.



## CLIENT CHALLENGES

This new application was being developed to provide an improved customer experience in partnership with a major pharmacy chain. The application deployment and planning was decided and the remaining concern was on-time product deployment to the country-wide pharmacy chain. The application used a web-based interface to allow customers access to their prescription information and was linked to the customer's data residing on a mainframe computer. The application had to be released on time and it was critically important that test coverage was known to eliminate defects in all stages of the end to end operation.

Manual test design was recognized as a bottleneck that had to be eliminated to enable successful on time delivery. Testing by hand was the baseline used to assess efficiency gain delivered by tools being considered for use in deployment. Even with this understanding, the testing team tried many test design tools without finding one capable enough to a) handle the application's complexity, b) provide known coverage of the system's operation, c) handle both GUI and mainframe testing, and d) automatically generate executable scripts to shorten their testing time using their custom automation platform.

## OUR APPROACH

This health care testing group was part of the team building the application for both the front end operation plus the mainframe operation to ensure correct data was accessed including prescription history, drug interactions, and prescription costs were accessible to the application user. The project used a modified agile approach to continuously create application capabilities that satisfied the needs of the pharmacy as well as the end users. Fast comprehensive testing with known coverage was imperative to meet their schedule and not have surprise errors at the later stages of delivery. This was especially true for the interaction of the mainframe code with the graphical user interface application.

After significant evaluation of all market tools for automated test generation, Conformiq Creator was chosen because it could test the operation of this heterogeneous technology application plus delivered directly executable test scripts for their Selenium web execution platform plus their proprietary mainframe execution platform.



### THE SOLUTION

Conformiq Creator was used to graphically capture the mainframe application's operational logic by manually modeling the screen operation. This information was obtained in part from the 3270 terminal screen operation in which the entry fields were shown and data values and strings for each entry were noted and modeled. The operation of each of the different screens was graphically captured in this way.

From mainframe screen capture like is shown below in Figure 1, the interface available for testing the functionality to be tested was identified and the links to the test data needed to generate data driven test cases were included in a graphical model of the application's operation. A sample model illustrating the captured interactions via the mainframe in a Structure Diagram (SD) and an Activity Diagram (AD) using these interactions is shown in a Creator screen shot in Figure 2 below.

The structure diagram is the starting point and foundation for building a model for testing as it captures the interface available for interacting with the application under test via mainframe or other interfaces such as GUI, mobile, API, or the back-end. To streamline this part of the model creation for mainframe applications, modelers used the Excel mainframe importer in Creator to efficiently create these mainframe custom action libraries. This cuts SD specification effort drastically - nearly zero - and gives the direct path to effective test automation.

Flows describing the functionality to be tested in the activity diagram can be specified to link already existing test data by using Excel spreadsheets, e.g., created from queries on SQL databases or production data. A second and unique option is to allow Creator to automatically generate test data needed to test each path based on the modelled application logic. This data generation capability saves additional time as testers don't need to wait for approved data to create test cases. Creator does this for them. In this particular case study, modelers decided to link existing test data to the models themselves.

The Activity Diagram shown in Figure 2 was created by creating flows of the application logic, specifying activity nodes based on interactions with the application under test, i.e., by dragging and dropping input or verification actions from the imported mainframe interface, and splitting flows based on data injected by input actions. The Creator test generation engine then understood and analyzed the flow and actions against the linked test data then used the user selected test optimization method to generate the optimized number of tests that cover all the application included in the model - both the positive and the negative test case flows. This ensured complete test coverage and, with the reporting tools within Creator, the coverage of each test case was shown as a highlighted path through the activity diagram. Also shown was the traceability matrix so each test case was linked with the requirement(s) it covered.

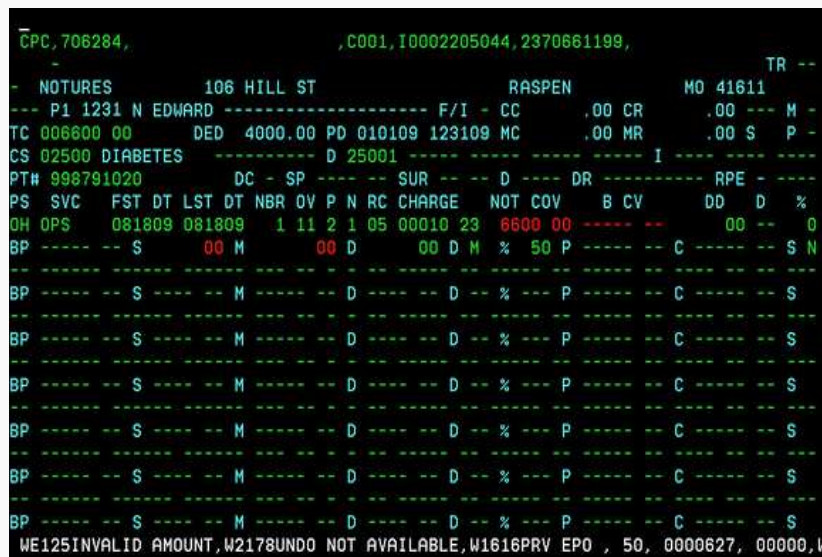


FIG.1 OBFUSCATED EXAMPLE SCREEN OF APPLICATION UNDER TEST

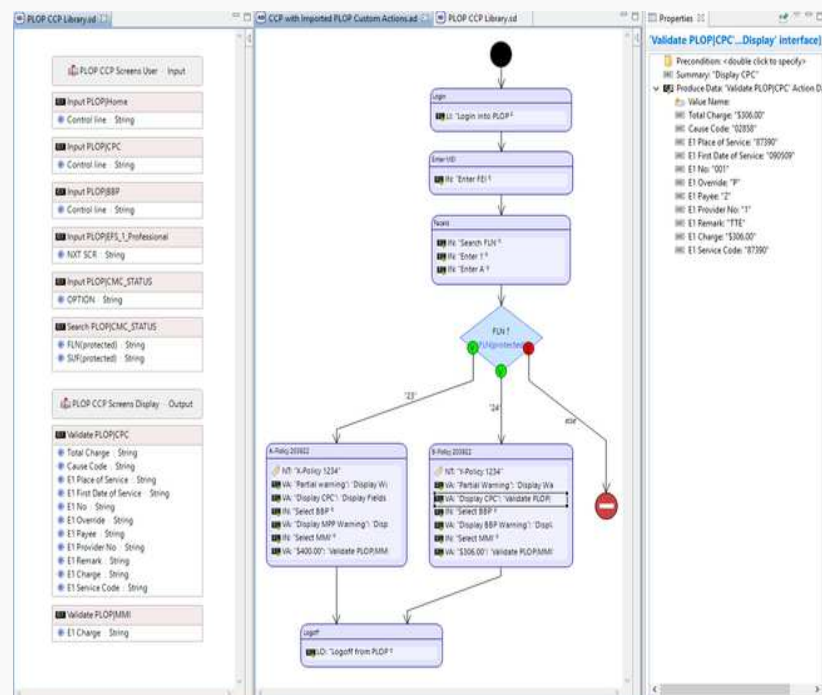


FIG. 2 CREATOR EXAMPLE MODEL



## THE SOLUTION

One key benefit achieved from this approach was that the entire team visualized not only the generated tests but also the overall functionality of the application by having a graphical model at hand. Often as in this case, applications running on mainframes are old and any original author is long gone. Without this understanding of the application logic it was very difficult to know what to test and nearly impossible to know if complete test coverage had been achieved. Additionally, as the model grows with new additions based on new application capabilities, component AND integration testing is delivered.

Once the generated test cases were reviewed, the next step was to automatically generate the test scripts with validations directly for automatic execution with the customer's custom test execution framework. This capability makes Creator unlike any other tool and delivered huge time and efficiency savings (as all other tools only generate the test cases for manual execution and testers must write the code snippets for execution along with determining their test validations). This is where the DevOps tool chain breaks down and Creator is uniquely able to bridge this gap and enable test execution automation fast enough to deliver a continuous CI/CD process. This enablement of direct automated test execution created huge time savings for the project. The tightly integrated testing process solution using Creator is shown in the diagram below (Figure 3).

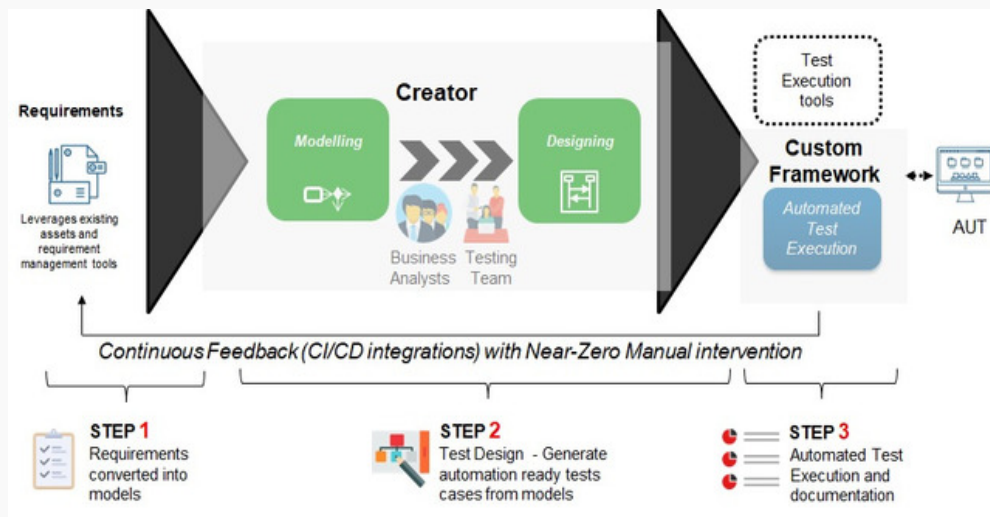
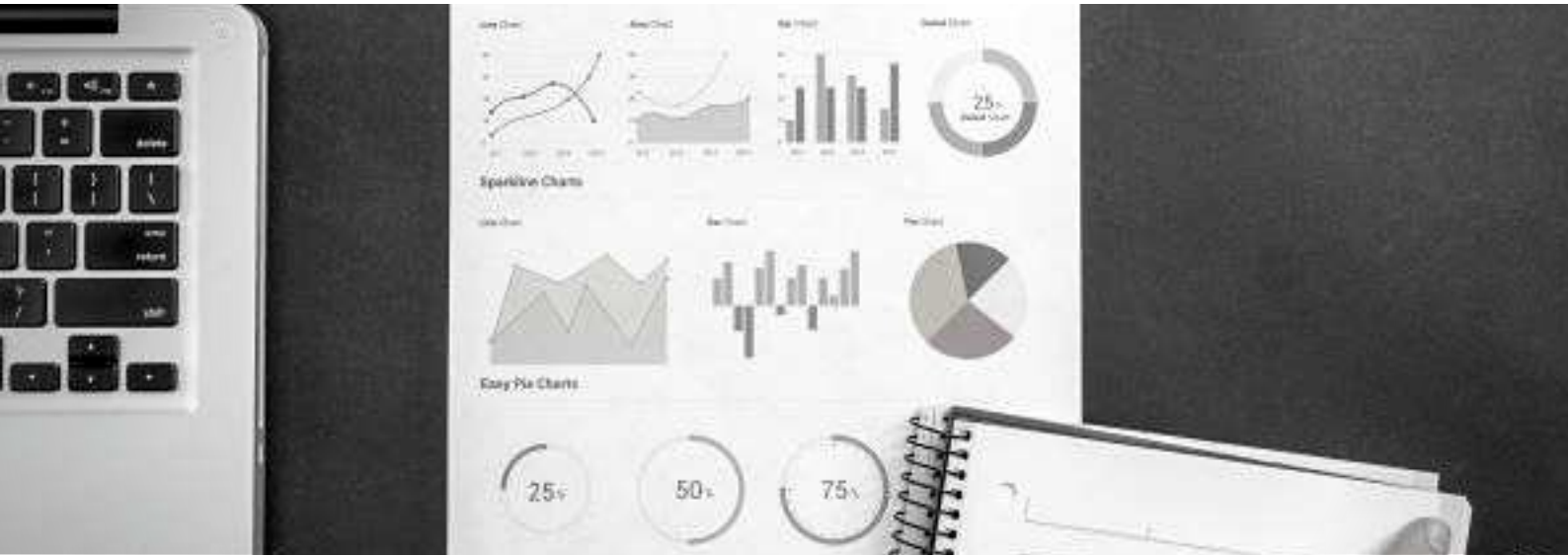


FIG. 3 END2END INTEGRATED TESTING PROCESS WITH FULL AUTOMATION



## CLIENT BENEFITS

The customer was able to meet the target release date that had been scheduled with their pharmacy partner with high confidence the application was fully covered by the generated test cases. This was due to Creator's graphical reporting of test coverage through the application and direct traceability to the product's requirements. They were able to deliver on time fully tested code even when development slipped some of their dates.

Creator is best known for its ability to test GUI based applications but the focus of this case study is the testing of the mainframe based application.

As the test cases were nontrivial, the manual effort took considerable time. Because Creator was used to model the application logic (not the test logic like with other tools) the modeling effort was significantly faster and the test case and executable script generation time almost immediate.

## SUMMARY

Based on the end to end testing capabilities within Creator and its ability to test both through the graphical interface and the mainframe operation, this customer used Creator to test their critical application. It was tested in time to meet the arranged go-to-market schedule and its availability for customer use is now prominently displayed in signs outside their partner pharmacy locations.

The next revision to this application is starting using the previous models so even greater time and cost savings will be achieved.



Based on the initial results showing a 6X productivity gain, this represented \$360K savings each year if the tester's annual burdened cost is just \$60K each.