

Improving Quality Assurance Processes for Implementer Acceptance

Project Summary and Next Steps

September 18, 2020

Background

With support from Digital Square (Notice B), the Bahm and OpenMRS conducted the OpenMRS Improving Quality Assurance Processes project during 2019-20, and was transitioned to OpenMRS oversight in September 2020

The OpenMRS Community contends that quality is a critical factor in establishing a user's trust in a system dedicated to managing highly personal health data. This trust is earned through rigorous, comprehensive, and systematic quality assurance (QA) testing and protocols. Prior to this project, the community provided limited QA on the technical side of the software release, without a means of conducting structured and repeatable in-depth end to end testing of common, community products.

Following software development best practices, the team proposed a new community-wide quality assurance process that includes developer and implementer participation, piloted the OpenHIE automated test framework and tools for building a documented set of OpenMRS-specific test cases, and automated the first few priority test cases on the QA roadmap to establish feasibility and sustainability of using this approach. The team also collaborated with release managers and implementers to pilot the integration of an automated test framework and test management tool into the release processes for our community Reference Application and a squad's module.

The OpenMRS platform is a foundation that OpenMRS implementers build on in response to their clients' needs and requirements. For implementers with constrained development resources, having confidence that the OpenMRS platform functions as intended can lead to greater efficiencies in their own quality assurance processes. Additional efficiencies can be gained through greater access to an automated test framework with a library of test cases that can be adapted or extended for use in implementer's settings. By leveraging community-established quality assurance protocols and with access to a flexible, test management suite that can be integrated with multiple automated test frameworks, countries

and partners implementing OpenMRS can trust community built and maintained open source electronic medical record tools.

Project Journey

- Using the results of a landscape analysis of community quality assurance tools and processes, the Quality Assurance Support Team selected CucumberStudio, an automated test management suite, that could build on existing community QA strengths while addressing areas for improvement.
 - CucumberStudio supports integration with the OpenMRS Community's existing automated test framework, Selenium, as well as other test frameworks used by OpenMRS implementers and squads.
 - As a test management system based on behavior driven development, CucumberStudio offers the OpenMRS community a means to improve documentation of requirements, testing SOPs, and integrate quality assurance into the development process while sharing responsibility for quality assurance across development team members.
- Developed prioritization criteria used to review Reference Application test cases and create a test case roadmap.
- Using prioritized test cases, piloted and refined processes for using CucumberStudio.
- Advanced a community-led and sustainable process by engaging with OpenMRS implementers (Partners in Health, Brown University, DIGI/University of Washington, AMPATH, Mekom Solutions) and Global Goods (OpenELIS, OpenHIE) contributing to community squads (FHIR Squad, OCL for OpenMRS Squad, Microfrontend Squad) and community-led Reference Application release team.
- Established a change management plan to introduce CucumberStudio and promote a culture of testing through open community working sessions as well as outreach and support to community squads and implementers collaborating on product releases.
- Piloted CucumberStudio-Cypress integration as a part of the QA process for the OCL for OpenMRS module, released on 11 September 2020. This effort demonstrated two key points: 1) that squads and implementers can integrate CucumberStudio into their QA process, using an automated test framework other than Selenium, and 2) the level of effort required to fully use CucumberStudio and an automated test framework for end-to-end testing.
- Demonstrated community QA process using CucumberStudio-Selenium integration at OpenMRS Community Design Forums, OpenHIE DevOps Working Group meeting, and IHE Services meeting.

Lessons Learned

- Sustainable, community-led QA processes and tools give implementers and squads the flexibility to use their existing tools alongside or with new community tools. The QA Support Team found that implementers and squads either use different automated test

frameworks or prefer to select an automated test framework themselves. CucumberStudio is an automated test management suite that supports integration with multiple frameworks that allows them to retain this autonomy - while addressing critical gaps, such as poorly documented requirements and lack of SOPs.

- Creating and implementing a community culture change and a change management strategy is essential to adoption of new community QA processes and tools. Through the landscape analysis and automated tool analysis and initial experimentation with CucumberStudio, community members raised critical questions about why testing had to be broader than just unit testing or a small set of Selenium tests. The most common concern was that a full QA testing pipeline might be unsustainable in the current context of the community. These discussions led the QA Support Team to develop and implement a change management strategy as a part of piloting CucumberStudio. Through initial discussions with squads and open community working sessions, the QA Support Team highlighted the benefits of using CucumberStudio and the QA pain points that CucumberStudio's BDD approach could address.
- The OpenMRS squad model increased opportunities to meaningfully engage, evangelize, and pilot new QA processes and tools with a broad range of implementers. The QA Support Team initially reached out to two implementers (Partners in Health, Intellisoft) to gauge their interest in piloting CucumberStudio. However, new opportunities to engage with as many as nine different implementers via squads and a country-level multi-partner project emerged. This led to unexpected opportunities for implementers to use these new QA frameworks for set of test cases in a safe environment, alongside their traditional QA tools (i.e: existing automated test cases, spreadsheets, manual test scripts).

Artifacts

Artifact	Link
OpenMRS Maturity Model: September 2020	OpenMRS Maturity Model 2020-09
Landscape Analysis	Final QA landscape and ecosystem report
QA Prioritization sheet	QA Prioritization Criteria
QA Communication Strategy	QA Communication Strategy
QA Team Test Case Road Map	QA Team Test Case Road Map
Quarterly reports	
OpenMRSQA_Global Good Quarterly Report_2019-10-15	OpenMRSQA_Global Good Quarterly Report_2019-10-15.docx
OpenMRSQA_Global Good Quarterly	OpenMRSQA_Global Good Quarterly

Report_2020-01-15	Report_2020-01-15
OpenMRSQA_Global Good Quarterly Report_2020-04-15	OpenMRSQA_Global Good Quarterly Report_2020-04-15
OpenMRSQA_Global Good Quarterly Report_2020-07-15	OpenMRSQA_Global Good Quarterly Report_2020-07-15
Meeting Notes	
2019 Meeting Notes	2019 Notes
2020 Meeting Notes	2020 Notes
Presentations	
OpenMRS QA presentation_OMRS19	Christine_OpenMRS QA presentation_OMRS19
Design Forum: OpenMRS QA presentation	Design Forum: OpenMRS QA presentation
OpenMRS QA presentation: OpenHIE DevOps Working Group	OpenHIE:OpenMRS QA presentation
OpenMRS QA presentation: IHE Services	IHE: OpenMRS QA presentation
GDHF Abstract 2020 Individual Presentation: OPENMRS QA Program	Improving Trust in OpenMRS: Towards A Proactive and Extensible Community Software Quality Assurance Framework
Key call recordings	
10th June 2020	https://iu.mediaspace.kaltura.com/media/2020-06-10+QA+CucumberStudio+Design+Meeting/1_aqkiku63
15th June 2020	https://iu.mediaspace.kaltura.com/media/2020-06-15A+QA+CucumberStudio+Design+Meeting/1_s1575zmk
22nd June 2020	https://iu.mediaspace.kaltura.com/media/2020-06-22A+QA+Automation+Design+Meeting/1_py01xesn
Test cases	
Reference Application test cases	Ref App test case
Test case Sample Template	Test case Sample Template

Manual Test Cases	Manual Test Cases
Wiki	
QAFramework Technical documentation (Cucumber-Selenium Set up)	https://wiki.openmrs.org/pages/viewpage.action?pageId=235277297
How-To: Integrate Community QA Processes and Tools into the Development Process	https://wiki.openmrs.org/display/docs/How-to+%3A+Integrate+Community+QA+Processes+and+Tools+into+the+Development+Process
Quick Guide to Creating a Test Case on CucumberStudio	https://wiki.openmrs.org/display/docs/Quick+Guide+to+Creating+a+Test+Case+on+CucumberStudio
Blog Posts	
Digital Square Blog Post: OpenMRS QA Project 2020-02	Digital Square Blog Post: OpenMRS QA Project 2020-02
Five Things We've Learned About OpenMRS and Quality Assurance	Five Things We've Learned About OpenMRS and Quality Assurance
What Quality Assurance Tools Have the Potential to Enhance our QA Practices?	What Quality Assurance Tools Have the Potential to Enhance our QA Practices?
Making Community-Led Quality Assurance Happen	Making Community-Led Quality Assurance Happen
Pilot Documentation	
Protocol: QA Process and Tool Integration	How To: QA Process and Tool Integration
Pilot Report	Pilot Report

Next Steps

Quality Assurance Support Team

- Support community release managers to plan and implement quality assurance for upcoming Platform and Reference Application releases.
- Coordinate with the Reference Application release manager to identify testers to automate additional Reference Application test cases using CucumberStudio-Selenium integration in anticipation of the upcoming Reference Application release

Squads

- Follow up with Microfrontend Squad, Analytics Engine Squad, and FHIR Squad
- Provide guidance on proof of concept testing and validation approaches to support Analytics Engine Squad and the patient level indicator reporting scope.
- Continue to identify opportunities to promote best community QA practices and use of automated tools, including CucumberStudio.

Future directions

- Strengthen the adoption of the improved QA for community-wide utilization and conduct dissemination of a QA model to the broader global goods community.
- Extend OpenMRS automated test portfolio.
- Address QA for OpenMRS interoperability testing.

Impact

The main focus of this project was to strengthen quality assurance processes and the use of automated test frameworks within the OpenMRS community, resulting in high-quality software and increasing trust in our system through rigorous, comprehensive, systematic quality assurance. Our goal was to develop a community-led, sustainable quality assurance process by creating an initial set of automated test cases and piloting an automated test framework to manage and conduct these tests as a part of the QA process.

This project led to a set of prioritized, automated test cases for the Reference Application as well as a set of test cases for the OCL for OpenMRS module, each conducted using a different automated test frameworks (Selenium and Cypress respectively) while managed by the same CucumberStudio automated test management tool. In bringing our community quality assurance specialists together with community release managers and implementers working in squads, this project demonstrated that quality assurance is a responsibility shared across team members and is most effective when integrated into the development cycle from the beginning. Moreover, instead of relying on an external QA team, implementers experienced how squads take the lead with QA, choosing how they want to implement community-led quality assurance processes and use community-tested QA tools.

Over the course of this project, exploratory discussions on quality assurance for OpenMRS FHIR initiatives led to new opportunities for the OpenMRS QA Support Team to experiment with a QA tool, CucumberStudio, that can be used with interoperability test cases for OpenMRS to exchange data with other systems in the OpenHIE architecture. Through working sessions and webinars, the OpenMRS QA Support Team has shared its experience and lessons learned with the OpenMRS Community, the OpenHIE DevOPs working group, and the IHE Services group.

As a result, this project sets a strong foundation that in the midterm will enable future integration with the Instant OpenHIE project as well as leverage its experience with the OpenHIE testing framework and process to serve as an advocate and leader for other global good communities interested in adapting this model and tooling. In the long term, this will lead to high quality, interoperable global goods that countries and users can trust.