OSEHRA

Open Source Electronic Health Record Agent

Procedures for Contributing Code and Performing Code Reviews

1. Purpose

This document defines the concrete steps necessary for a user of the OSEHRA EHR system to contribute code back to the OSEHRA code base, or to perform a review in support of code submitted by another individual. The document is divided into two sections corresponding to those divisions.

Code developed for inclusion into, or evaluation for interoperability with the OSEHRA code base can come in three forms:

- Bug fixes or minor code modifications
- Formal OSEHRA Code Releases
- New module contributions or major refactoring

Depending upon the nature of the code change being proposed or reviewed, the code submission process and the review tool will change; however, the goal remains of verifying software quality to ensure code contributions are Safe, Functional, and Compliant as defined in the OSEHRA Software Quality Certification Plan http://www.osehra.org/page/plans-and-white-papers.

2. Submitting Code to the OSEHRA Code Base

As noted in the previous section, code developed for inclusion into or evaluation for interoperability with the OSEHRA code base can come in three forms, bug fixes; formal OSEHRA Code Releases; and new module contributions or major refactoring. These three code submission types are handled by two different code submission processes. If you are submitting:

- Bug fixes or minor code modifications
- Formal OSEHRA Code Releases

Please refer to Section 2.1, “Submitting to the Gerrit Code Review System”. If you are submitting:

- New module contributions or major refactoring

Please refer to Section 2.2, “Submitting to the OSEHRA Technical Journal”.

2.1 Submitting to the Gerrit Code Review System

Bug fixes or minor code modifications and formal OSEHRA releases both require submission to the Gerrit Code Review System. The basic procedures of code submission, including initialization, checkout of the OSEHRA code base, and pushing to Gerrit, remain the same and are covered in “Contributor Git Instructions”, http://www.osehra.org/page/contributor-git-instructions. In this section, we discuss the differences between submissions designed to modify the code base and submissions designed to initiate the code release process.

Both types of submissions, code modifications and release version initiation, begin with an entry in the Jira issue tracker, with the exception that bug reports and requests for code modifications can be submitted to Jira by any person; whereas, the initiation of new releases must be made by a trusted member of the community as part of a formal release process. For release version initiation, the Jira ticket should be titled “Initiation of release #(SHA1)”, where SHA1 corresponds to the unique git tag for the current review target. Non-release Jira tickets should be given a descriptive name and should be given sufficient details in the description so as to allow the developer to understand and replicate the issue.

The specific procedures for setting up a Gerrit environment and for performing the mechanics of submitting to the repository are covered at: http://www.osehra.org/page/contributor-git-instructions. Please refer to that site for additional supplemental information.

2.1.1 Submitting code in response to a standard Jira ticket

To submit code in response to a standard Jira ticket, first obtain a current release from the OSEHRA code repository and follow the directions to set up a testing environment and execute the tests. Download the “OSEHRA M-Code Primary Developer Checklist” and follow the procedures referenced in that document to verify code operation prior to and after the code modifications. Once the code modification is ready, complete the “M-Code Primary Developer Checklist”, attach it to the Jira ticket, and proceed to Section **Erreur ! Source du renvoi introuvable.** to complete the push of the code into Gerrit, referencing the Jira ticket number in the description of the code change. Note that if the code requires modifications or additions to the testing repository, the Gerrit submission process should be executed twice, first in the testing checkout and then in the OSEHRA code checkout. In this case the description provided during the Gerrit code push should refer to the testing checkin as well as the Jira issue.

2.1.2 Submitting code in response to a release version initiation

Code submitted in response to a release version initiation should consist of a single push consisting of a change to the upper level OSEHRA Code Base ATTESTATION file. The change should consist of the addition of a single line as the topmost attestation. The line will contain the SHA1 key for the repository to be released, the date, and the name of the attester. Figure 1 shows an example ATTESTATION file after the addition of the first release attestation. Again complete the “Primary Developer Checklist” and attach it to the Jira ticket prior to performing the push to Gerrit; although, for this specific case, most of

the entries will be N/A. For the short description, the message should replicate the Jira title “Initiation of release #(SHA1)”. The long description can contain release notes along with the reference to the correct Jira ticket.

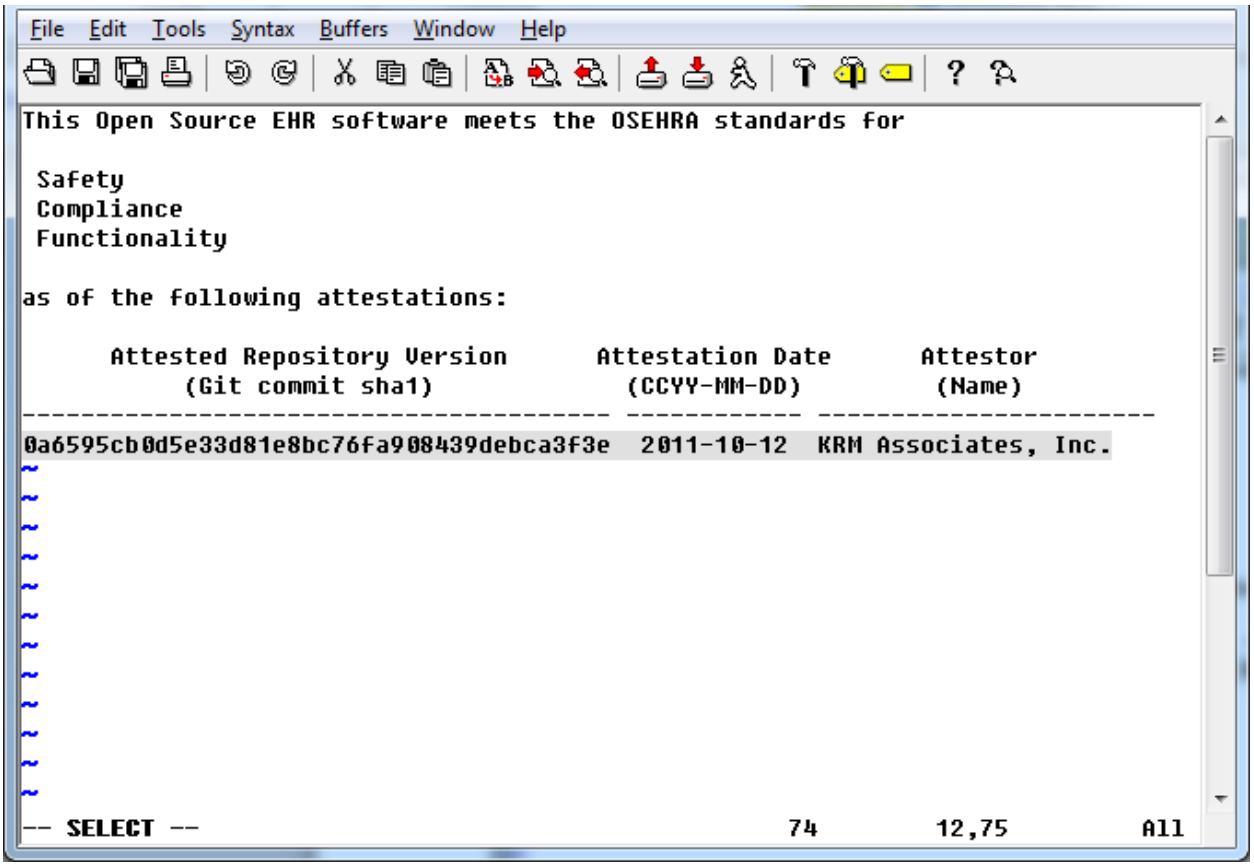

Figure 1 - Structure of the ATTESTATION file. The highlighted line shows the structure of the new addition containing the SHA1 key, date and attester name.

2.2 Submitting to the OSEHRA Technical Journal

Substantial code contributions such as new VistA modules or major refactorings of the existing code base require a submission to the OSEHRA Technical Journal (OTJ). OTJ submissions allow for a more thorough description of the submitted code; allow community members to download, use, try, and maintain the submitted code prior to and independently of its eventual inclusion into the OSEHRA code base; and allow for persistence of the submission.

To submit code to the OSEHRA Technical Journal, first obtain the “OSEHRA M-Code Primary Developers Checklist” from the OSEHRA web site (http://www.osehra.org), or from the OTJ site. The “OSEHRA M-Code Primary Developers Checklist” provides a set of steps, procedures and documentation requirements that should form part of any submission. As each step of the checklist is met, mark it as complete and save the document. Along with developing the code, also generate a set of automated tests that execute using the OSEHRA Code Testing framework, a description of any additional functional tests that should be carried out manually to fully test the system, and a Technical Article that describes the functional goals of the system, the use of the system and any additional details that may help a user

of the package and for the subsequent developers who will maintain the package. Technical Articles are expected to follow the style of a technical report, with particular focus on providing guidance for the future use and maintenance of the new code contribution.

Use an archival tool (for example, zip or tar.zip) to generate several contribution packets consisting of:

- The code to be submitted to the OTJ
- The automated tests and data to be submitted in support of the code
- The completed "OSEHRA M-Code Primary Developers Checklist" along with any required supporting documents

When combined with the Technical Article, this results in four files that need to be prepared. Once the submission is ready, go to the OTJ and click on Submit as shown in Figure 2. The submission process will walk through the required steps of the submission including:

- Choosing a submission target
- Agreeing to the open source license
- Filling in the contact and general information of the submission
- Uploading the:
    - Technical Article
    - Source Code
    - Test Code and data
    - OSEHRA M-Code Primary Developers Checklist and supporting documents
- An optional developer specific logo

At the end of the process the article and code is uploaded to the OTJ and becomes available for download, review, comments and eventual inclusion into the OSEHRA code base.

Figure 2 - OSEHRA Technical Journal home page with the submit button indicated by the red arrow.

3. Reviewing Code after Submission

All code review has as its goal the certification of code quality. The steps and attestations of the review process are similar for all review processes; however, the specific procedures depend upon the review system chosen by the contributor. To review a code submission to:

- The Gerrit code review system

Please refer to Section 3.1, Reviewing Submissions to the Gerrit Code Review System.

To review a code submission to

- The OSEHRA Technical Journal

Please refer to Section 3.2, Reviewing Submissions to the OSEHRA Technical Journal.

3.1 Reviewing Submissions to the Gerrit Code Review System

Gerrit code reviews are intended to ensure that contributed code modifications and release versions are of high quality and suitable for integration into the OSEHRA code base. There is no technical differentiation between the two types of review on the Gerrit Code Review site, but we do differentiate them procedurally.

Bug fixes and minor code modifications are considered to be local changes. The reviewer is intended to review the specific code submitted for review and to compare unit and regression tests from before the inclusion of the submission in the code base with unit and regression tests after the inclusion of the submission.

Release versions contain no code changes beyond the change of version number. Instead, the reviewer is expected to look at the corpus of changes that were added since the last release version. This code review is manual and should only be undertaken by someone skilled in Git and knowledgeable with the VistA system. Since release reviews can encompass a large number of code changes, they also represent a substantial investment in time to complete.

For both types of code submissions, the review is in two stages.

- The first stage is a **Peer Review** to establish code quality
- The second is a **Software Quality Assurance (Final) review** which occurs just prior to formal inclusion of the contribution into the OSEHRA code base.

Each of these reviews is covered below.

3.1.1 Peer Review

A peer review is a necessary confirmation that the submitted code is of sufficiently high quality so as to be eligible for inclusion in the OSEHRA code base. Peer reviews can be made by anyone, and multiple peer reviews are allowed and encouraged; however, at least one passing peer review must be made by a trusted individual if the code is to be considered for adoption.

Gerrit peer review is based on the "OSEHRA Peer Review Checklist". This should be downloaded from the OSEHRA web site, http://www.osehra.org prior to beginning the review process.

To perform a peer review, go to the OSERHA Gerrit review site at http://review.code.osehra.org and log in. Find the article you want to review. Click on the article to bring up the publication page, and then click the one of the diff buttons (blue arrows) to bring up a code review tool (Figure 3). Verify that the code looks to fix the corresponding Jira issue and that the code is appears to be compliant with the OSEHRA SAC. Walk through the "OSEHRA Peer Review Checklist" executing all the appropriate tests for ***Safe, Compliant, and Functional*** and marking all items Pass or Fail. Once the status of the code with respect to the checklist has been determined, press the review button to bring up the attestation page (Figure 4). For each of Safe, Compliant, and Functional; mark the section +1 if all the items in the checklist have a pass for the section. If all three sections have a pass and the visual code review looked

good, mark the Code Review Attestation +1 (+2 if you are a trusted reviewer). Press Publish Comments to end the review and submit the results to the OSEHRA Dashboard. Note that any sections marked -1 when the "Submit Review" button is clicked will show up as a failing test on the dashboard. Finally complete the review by uploading the completed checklist document to the Jira issue tracker, attaching it to the corresponding Jira issue.

3.1.2 Software Quality Assurance (Final) Review

A final review is a necessary confirmation that all required procedures have been executed, the submission is complete, and that the code is ready to be included into the OSEHRA code base. Only one passing final review is required for a submission and the code contribution can be merged into the OSEHRA code base as soon as a passing final review has been attested. As such, final reviews can only be made by a trusted individual who possesses sufficient permission to perform the code merge step.

Gerrit peer review is based on the "OSEHRA Final Review Checklist". This should be downloaded from the OSEHRA web site, http://www.osehra.org, prior to beginning the review process.

To perform a final review, go to the OSERHA Gerrit review site at http://review.code.osehra.org and log in. Find the article you want to review. Click on the article to bring up the publication page, and then click the one of the diff buttons (blue arrows) to bring up a code review tool (Figure 3). Verify that the code seems to fix the corresponding Jira issue and that the code appears to be compliant with the OSEHRA SAC. Walk through the "OSEHRA Final Review Checklist" executing all the appropriate tests for ***Safe, Compliant, and Functional*** and marking all items Pass or Fail. Once the status of the code with respect to the checklist has been determined, press the review button to bring up the attestation page (Figure 4). For each of Safe, Compliant, and Functional; mark the section +1 if all the items in the checklist have a pass for the section. If all three sections have a pass and the visual code review looked good, mark the Code Review Attestation +1 (+2 if you are a trusted reviewer). Press Publish Comments to end the review and submit the results to the OSEHRA Dashboard. Note that any sections marked -1 when the "Submit Review" button is clicked will show up as a failing test on the dashboard. Complete the review by uploading the completed checklist document to the Jira issue tracker, attaching it to the corresponding Jira issue, and then merge the change into the OSEHRA code base using the Gerrit merge procedures.

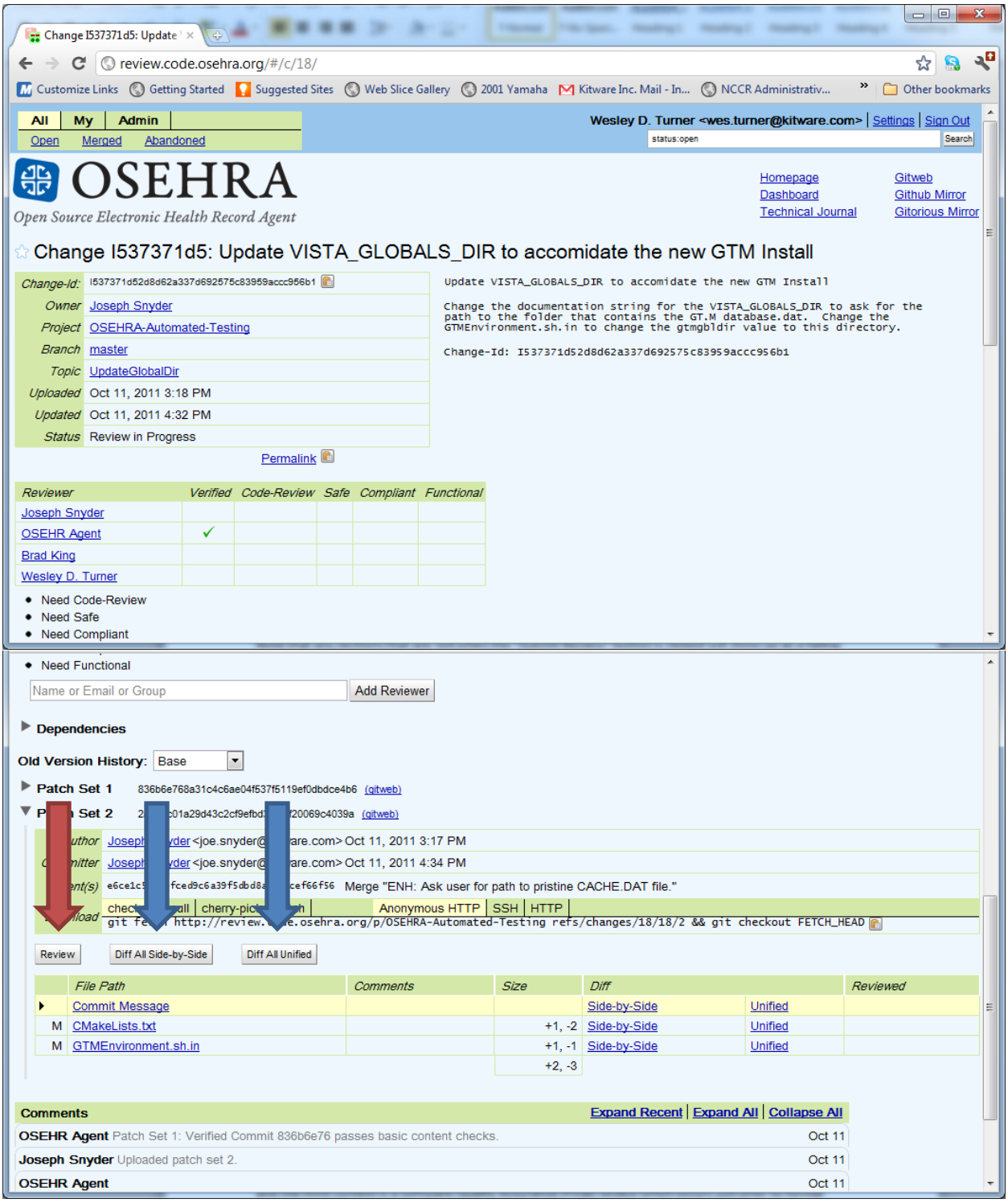

Figure 3 - Gerrit Code Review Site with Review Button (red arrow) and Diff Buttons (blue arrows).

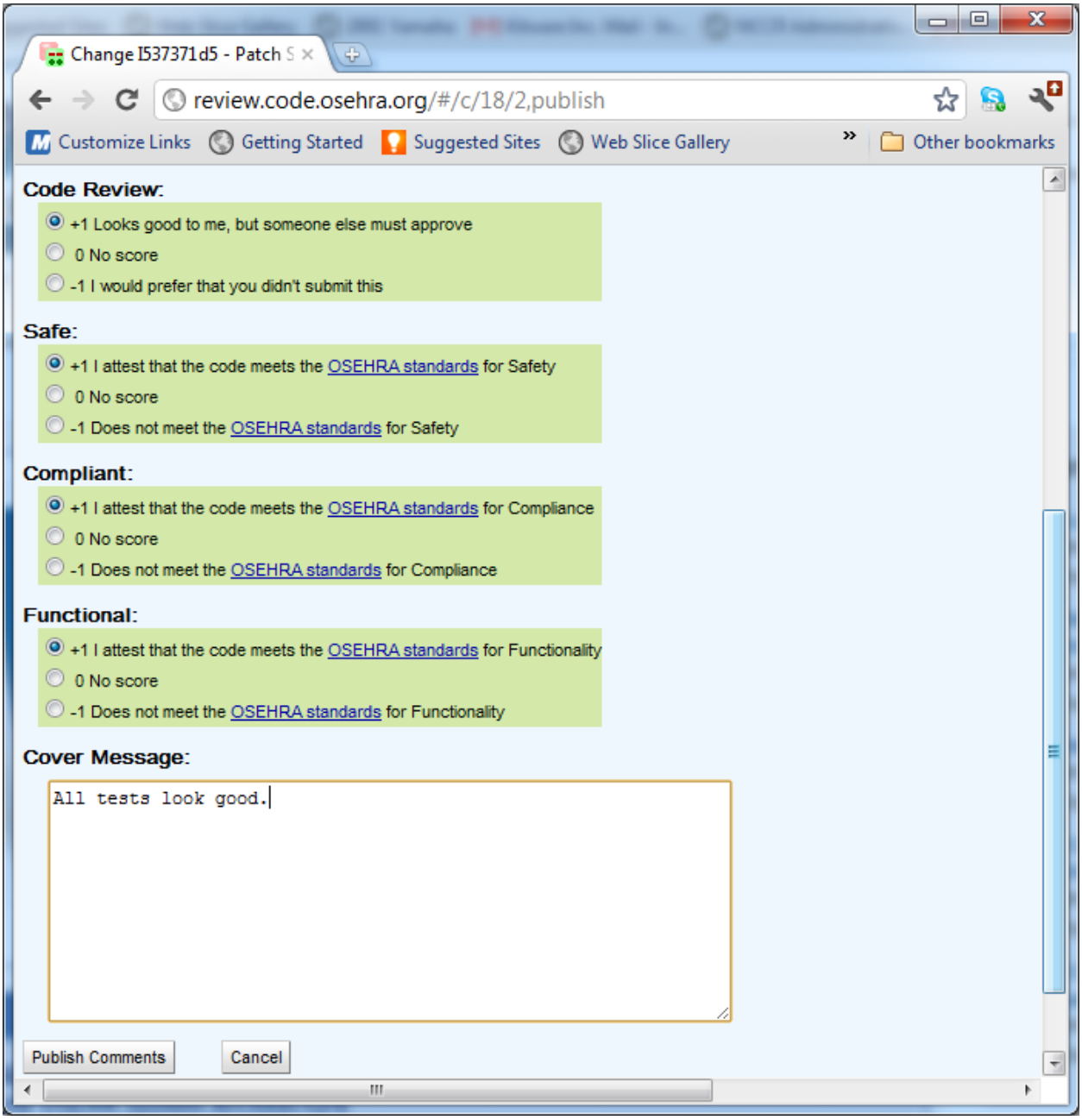

Figure 4 - Code review attestations.

3.2 Reviewing Submissions to the OSEHRA Technical Journal

There are three different contexts for reviews in the OTJ.

- The first and simplest context is to give the code a “thumbs up” indicating it is a positive contribution and should be considered for inclusion into the OSEHRA code base
- The second context is a technical peer review and attestation of code quality; and

- The third context is a Software Quality Assurance (Final) review which occurs just prior to formal inclusion of the contribution into the OSEHRA code base. Each of these contexts is covered below.

3.2.1 Thumbs Up Review

Thumbs up reviews are the simplest reviews, but in some sense can be the most powerful. Thumbs Up reviews are available to the entire OSEHRA community and giving code a "Thumbs Up" or a "Thumbs Down" is voting for, or against that code being brought into the OSHERA code base. The "Thumbs Up" is not a substitute for a full Software Quality Certification, but it is a mechanism to prioritize and establish the desirability and utility of new capability.

To perform a "Thumbs Up" review, go to the OTJ site and log in. Find the article you like (or dislike). Click on the article to bring up the publication page and click on the thumbs up or thumbs down symbol to indicate a positive or negative impression of the paper (Figure 5). Comments on why you voted are welcome and encouraged.

3.2.2 Peer Review

A peer review is a necessary confirmation that the submitted code is of sufficiently high quality so as to be eligible for inclusion in the OSEHRA code base. Peer reviews can be made by anyone, and multiple peer reviews are allowed and encouraged; however, at least one passing peer review must be made by a trusted individual if the code is to be considered for adoption.

To perform a peer review, go to the OTJ site and log in. Find the article you want to review. Click on the article to bring up the publication page and then click the Review tag (Figure 5). The review checklist page article will be displayed (Figure 6). Make sure the "Peer Review" tab is selected and walk through the review process indicated in the peer review checklist. As each test is verified, check it off. When all checks are made in a section, the corresponding upper level bullet will become checked and turn green. Once the review is complete, click the "Submit Review" button to finalize the review and send the attestation to the OSEHRA Code Quality Dashboard. Note that any sections that are red when the "Submit Review" button is clicked will show up as a failing test on the dashboard.

3.2.3 Software Quality Assurance (Final) Review

A final review is a necessary confirmation that all required procedures have been executed, the submission is complete, and that the code is ready to be included into the OSEHRA code base. Only one passing final review is required for a submission and the code contribution can be merged into the OSEHRA code base as soon as a passing final review has been attested. As such, final reviews can only be made by a trusted individual who possesses sufficient permission to perform the code merge step.

To perform a final review, go to the OTJ site and log in. Find the article you want to review and click on it to bring up the publication page. Click on the Review tag (Figure 5). The review checklist page article will be displayed (Figure 6). Make sure the "Final Review" tab is selected and walk through the review process indicated in the final review checklist. As each test is verified, check it off. When all checks are made in a section, the corresponding upper level bullet will become checked and turn green. Once the review is complete, click the "Submit Review" button to finalize the review and send the attestation to

the OSEHRA Code Quality Dashboard. Note that any sections that are red when the "Submit Review" button is clicked will show up as a failing test on the dashboard.

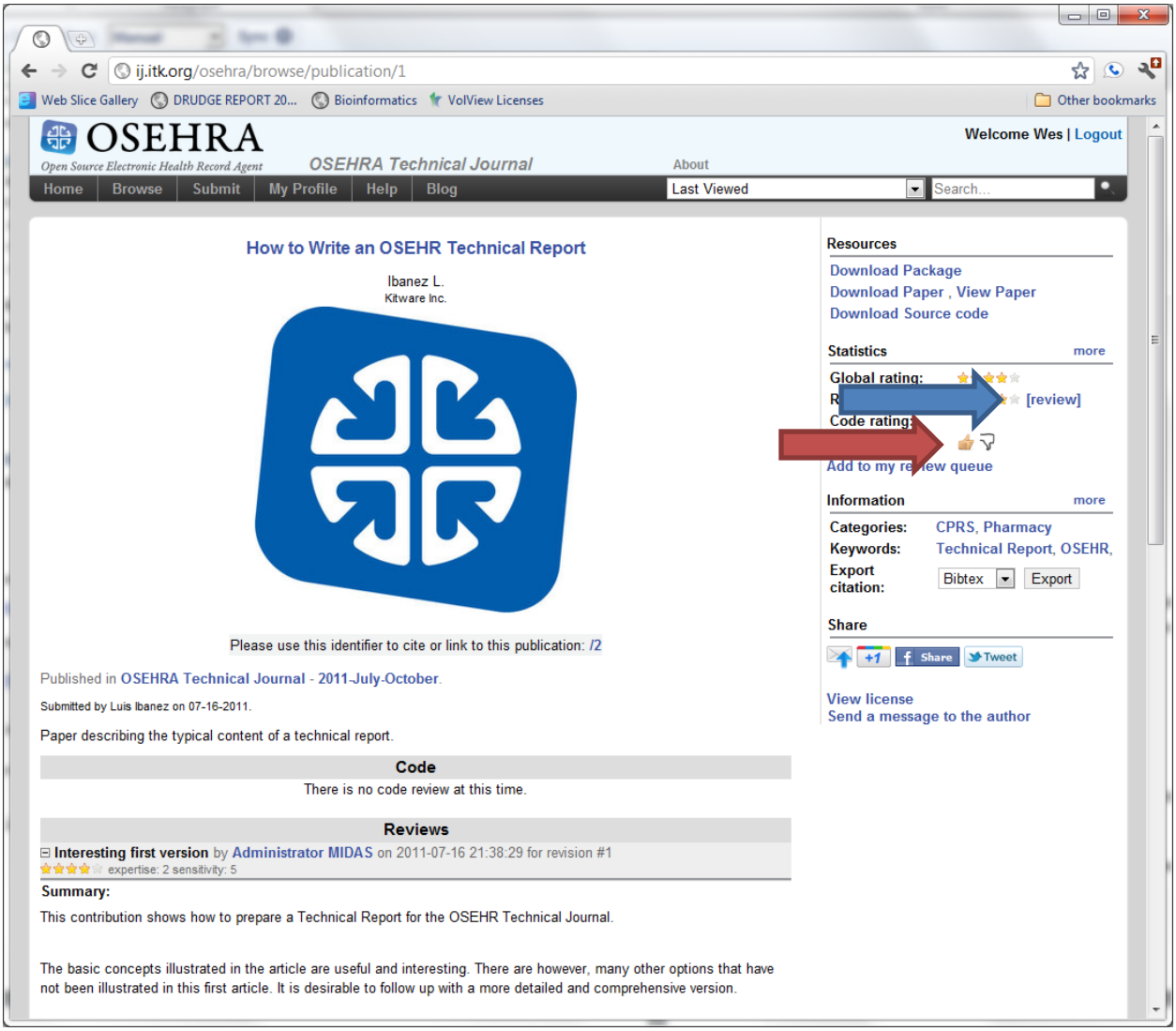

Figure 5 - OTJ Article page with the formal review button (blue arrow) and thumbs up review (red arrow) indicated.

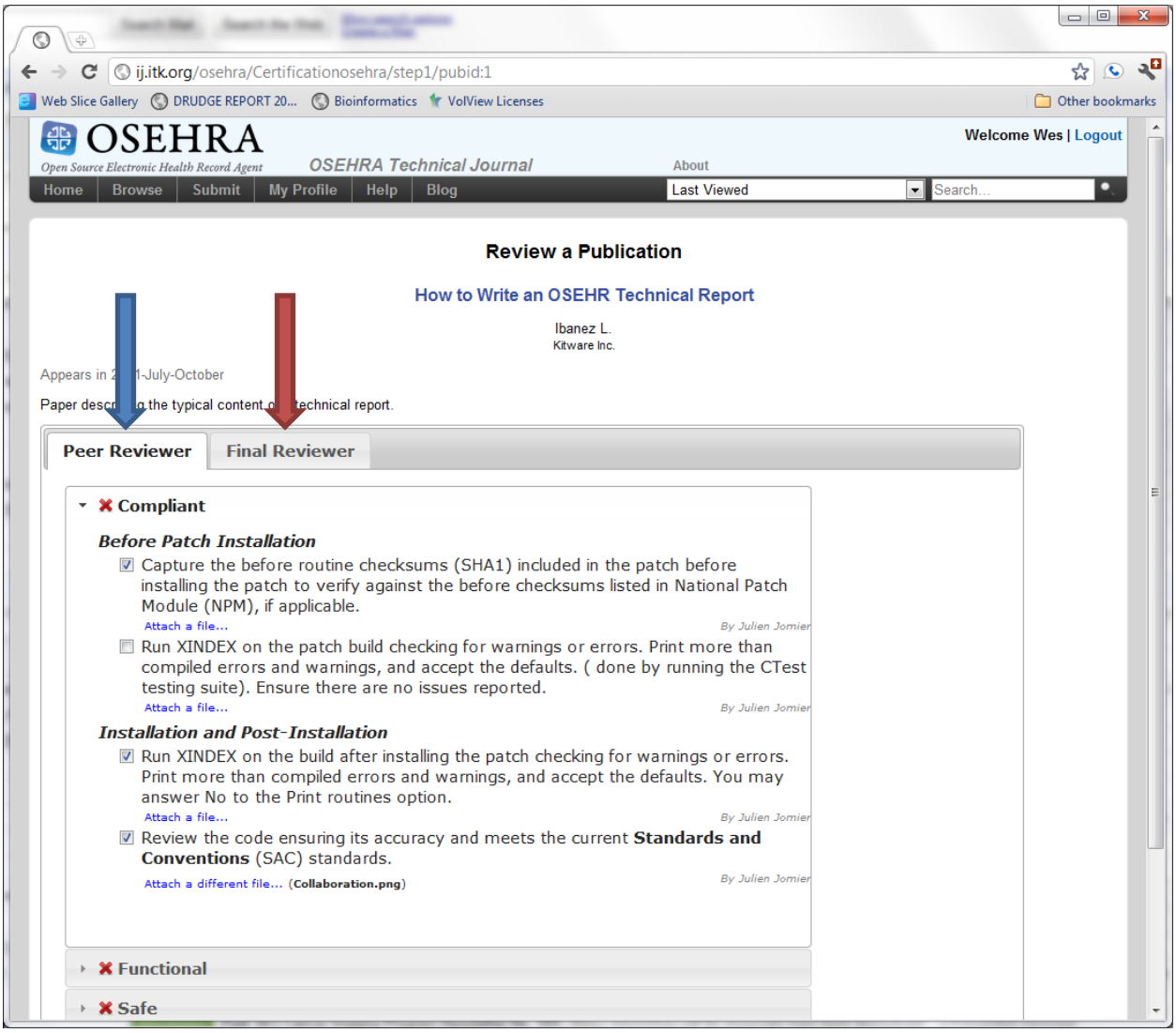

Figure 6 - OSEHRA formal review page with Peer Review (blue arrow) and Final Review tabs (red arrow) indicated.