Rosie Architecture Review
Issues raised in the review meeting on 10/26/12

- **Consistent Vocabulary**
  - Especially between the diagram and the rest of the documents, there are changes in the vocabulary used to describe the components, such as "Build number" vs. "Build ID". The inconsistent vocabulary makes it difficult to tell which components simply have different names in documents, and which really are different components.
    - **Characterization: ISSUE**
    - **SHOULD-FIX**
- **Document: Overview**
  - Putting "both objects" into the database seems to be a typo from a previous incarnation of the architecture, as it is not consistent with the rest of the description.
    - **Characterization: QUESTION**
    - **SHOULD-FIX**
- **Document: Component Analysis**
  - Add analysis of the components – we are confident you can use them, based on your experience, but would be clearer if there was a reason stated why these specific choices were the best ones for the situation. In fact, the MongoDB is analyzed in the "Persisted Objects" document; could move that to this document.
    - **Characterization: ISSUE**
    - **SHOULD-FIX**
- **Document: Interfaces**
  - "Primary" uses were listed, but were there other uses that might be important to the architecture that were left out? Not sure if you are covering all the important interfaces by only talking about the primary ones.
    - **Characterization: QUESTION**
    - **COMMENT**
  - Contradiction with the diagram – in diagram, can build from both the HTML and the CLI; in interfaces, can't build from the HTML. Could use clarification about which is correct.
    - **Characterization: ISSUE**
    - **SHOULD-FIX**
  - CLI takes only one command to get started – didn't understand what this meant. A few more sentences about what this means would go a long way in making the CLI as a whole easier to understand.
    - **Characterization: QUESTION**
    - **COMMENT**
  - Are Rosie API and JSON API the same thing? (From meeting: yes, they are.) This goes along with the consistent vocabulary note at the beginning.
    - **Characterization: QUESTION**
    - **COMMENT**
  - Does the queue hold build objects or just build numbers? (From the meeting: just build numbers.) Clarify this so that the reason the interface is simple is clear, not just lacking description.
    - **Characterization: QUESTION**
    - **COMMENT**
  - Web front end is shown as having only a single part, but it seems there are actually multiple different parts included. Should go into more detail about the interface with the user, etc.

- - **Characterization: ISSUE**
  - **SHOULD-FIX**
- **Document: Issues**
  - In the solved issues, clarify what you talked to Mark about and why that helped you solve them. We didn't understand that the solution was that it didn't matter that it wasn't object-oriented, so we thought the issue was still at large, though approved by Mark.
    - **Characterization: QUESTION**
    - **COMMENT**
  - In the outstanding issues, put more justification of why synchronous was chosen – you actually said it to us during the meeting, so add it to the document! "Worker thread – only one, so synchronous, but asynchronous to the application server" was the comment if I wrote it correctly. There was justification and clarification when we were talking to you, so that would clear up the uncertainty in the document.
    - **Characterization: ISSUE**
    - **SHOULD-FIX**
- **Document: Persisted Objects**
  - In MongoDB, move the explanation to the component analysis. For more info, look at the issue raised in the Component Analysis section.
    - **Characterization: ISSUE**
    - **COMMENT**
- **Document: Rosie Architecture**
  - How do the build results get put into the database? Nowhere in the documentation is it clear that they are actually put there, but it is clear that they need to get there somehow for access later.
    - **Characterization: DEFECT**
    - **MUST-FIX**
- **Document: Runtime Components**
  - Clarification of why the "code" is included. In meeting, explained very clearly that it dictates how the API must respond, so that should go in the paragraph before the code to make it clear why it's there.
    - **Characterization: QUESTION**
    - **COMMENT**
  - Clarification of if the synchronous worker is building IN the builder when it's running in the background? A short clarification of this should suffice.
    - **Characterization: QUESTION**
    - **COMMENT**
  - Clarification/description of what the configuration file actually is and where it should be stored. (During meeting: standalone file that will be figured out during implementation.) Should add this to the description.
    - **Characterization: ISSUE**
    - **SHOULD-FIX**

Overall: Project is APPROVED.
The changes we suggested were understood by the group and they had ideas about how they should approach them. We are confident that they will be able to make the changes and clarifications we suggested and make this into a working product.