OSP Procedure for Feature Requests

Actors:

- **Community** - Anyone active on the portfolio@collab.sakaiproject.org email list and/or weekly OSP teleconferences
- **Developer/Technical Liaison** - One or more technical developers responsible for facilitating the functional and design requirements, as well as the actual coding.
- **Functional Team** - Responsible for steering functional development of OSP (see OSP Functional Team Kickoff Meeting October 2008)
- **Developer Community** - Anyone responsible for any portion of OSP code development
- **Functional Analyst** - Person primarily responsible for advocating a specific new feature and shepherding the feature through this procedure (e.g. Functional Team member, UI designer).

Assumptions:

- The **Community** shall develop a requirements timeline for each subsequent release, which shall include a deadline for publishing proposed requirements on confluence and a deadline for community response to published requirements.

- Unless otherwise agreed upon by the **Community**, all enhancements must adhere to this procedure. Bug fixes and small-scale changes may omit some or all of the Enhancement/Specification Template based on community consensus.

- This procedure shall not contradict the Sakai Release Practices Proposals

Procedure

1. A **Functional Analyst** shall publish (in confluence) a proposed enhancement using the Enhancement/Specification Template by the aforementioned deadline for publishing proposed requirements. The **Functional Team** shall review and discuss the proposal prior to wider community involvement.
2. The **Community** shall review and discuss the proposed enhancement by the aforementioned response deadline, culminating with a decision on the weekly OSP teleconference (or face-to-face meeting). Those reviewing the proposal shall consider conditional release, priority, potential impact and ripple effects, and architectural review.
    - If the proposal is rejected and the proposing institution decides to move forward with development, the resulting code should not be added to trunk without further review.
3. The **Functional Analyst** shall develop UI Mockups and present a walkthrough to the community, and shall revise the proposal based on community input.
4. The **Community** shall review the UI Mockups (e.g. comments in confluence). Approval shall be made on the weekly OSP teleconference (or face-to-face meeting).
5. The **Developer/Technical Liaison** shall publish a design specification which *may* include class diagrams, algorithms, data modeling, entity relationship diagrams, logic diagrams, etc.
6. The **Developer Community** shall review the design specification and approve or request further details and/or revision.
7. The **Developer/Technical Liaison** shall begin code development within a *branch* of the subversion source repository.
8. The **Developer/Technical Liaison** shall periodically provide demonstrations of progress at Monday meetings and incorporate feedback from the **Functional Team** and **Community**.
9. The **Developer/Technical Liason** shall configure the osp-nightly server for review (build with the specified subversion *branch*).
10. The **Community** and **Functional Analyst** shall begin QA testing as soon as possible. Those involved in testing shall verify compliance with original UI mockups and user stories. The **Functional Analyst** and **Developer** shall work together to refine documentation based on community input.
11. The **Developer/Technical Liason** shall request approval from the Sakai community to merge changes to the trunk. The osp-nightly shall be returned to the trunk build.