OSP Feature Template

Title:	*(Title of the feature)*
Jira Number:	*(JIRA number as link)*
Related Jira Numbers:	*(JIRA numbers as links)*
Component(s):	*(Sakai / OSP components where the feature would be implemented)*
Author:	*(Name and Institution)*
Date:	*(Date)*
Demo Status and Date (s):	Status: *(Suggested, Pending, Confirmed)* Date: *(Date(s) enhancement demoed)*

- Title of Enhancement (1)
  - Summary (1)
  - Rationale (1)
  - Origin (1)
  - User Stories (1)
    - Actors and Stakeholders
    - Title of 1st Story
  - Functional Details (may be added after community demo) (1)
  - Interaction and Implications (1)
  - Diagrams and Mockups (3)
  - Community Acceptance (4)
- Specification Template (5)
- Behavior
- Interaction
- Quality Metrics
- Assumptions
- Outstanding Issues

Instructions

This template is the vehicle for complying with the OSP Procedure for Feature Requests. Each element of the template is keyed to the step that the element fulfills, in whole or in part. For example, a (1) in the heading indicates that this portion of the template must be filled out in order to complete step 1 of the proposal approval process. The Proposal for Enhancement page you create in Confluence based on this template will provide a locus for community discussion of the enhancement. A primary goal of this process is to capture institutional memory about what decisions were made during the design phase and *why*.

The Proposal for Enhancements template has two parts, the Functional Description and the Technical Specification. The expectation is that the template will be fleshed out over time as plans for the enhancement develop through community discussion.

Part 1: The Functional Description
The Functional Description should be created before the enhancement is first proposed, usually by domain experts such as instructional designers, educators, or project leads. As much as possible, it should paint a picture of what the enhancement is and what it will allow the user to accomplish. Focus should be on functionality rather than implementation. While the Summary, Rationale, Origin, and User Stories are required before the proposal is presented to the community, diagrams, mockups, and the "Functional Details" and "Interactions and Implications" sections may be added later as the proposal matures.

Part 2: The Technical Specification
The second part of the proposal, the Specification, should be filled out once the feature is clearly defined but before coding begins. While the first part of the proposal should usually be filled out by domain experts, the specification would normally be written by a developer or other technical person.

Please remove these instructions from your document.

Part 1: Functional Description

Title of Enhancement (1)

Summary (1)

Give an overview of the envisioned enhancement, providing enough information for a non-technical user to understand what the feature is and what it would provide. Feel free to list specific features of the enhancement, but avoid implementation details and focus on functionality.

Rationale (1)

Explain why this feature would be valuable, and to whom. Include background information about the problem the solution is meant to solve.

Origin (1)

Describe how the need or desire for this enhancement arose, including background information about the problem it is meant to solve. Be specific about institution(s) and people who have played a role in planning the enhancement.

User Stories (1)

The User Stories should paint a picture of what it is like for a user to make use of the enhancement. The actors should be based on real users with definable tasks and goals. Include as many stories as necessary to demonstrate how the enhancement would be used by different types of users.

Actors and Stakeholders

Title of 1st Story

Functional Details (may be added after community demo) (1)

Describe any functionality not fully captured in the User Stories.

Interaction and Implications (1)

Identify and describe potential interactions with existing and planned OSP/Sakai tools and enhancements from a functional perspective.

Diagrams and Mockups (3)

Include any ERDs, flowcharts, sketches, mockups, etc.

Community Acceptance (4)

Indicate how this feature has been discussed by the larger community (e.g., list discussion, specific meetings, etc.). Provide specific records of community acceptance (e.g., list institutions and contacts who also identify this feature as a requirement).

Part 2 of the Proposal for Enhancement Template: The Specification
The specification should be filled out once the feature is clearly defined.

Specification Template (5)

Behavior

Describe each specific behavior of the feature in the present tense as if the feature were implemented perfectly. Use precise, objective language to describe ideal behaviors against which actual behaviors can be evaluated.

In the case of conditions and behaviors that must be evaluated independently, they should be presented in a two-column table as below.

Conditions	Behavior
(Short description of mutually exclusive condition #1)	*(Objective, verifiable behavior in response to condition #1)*
(Short description of mutually exclusive condition #2)	*(Objective, verifiable behavior in response to condition #2)*

When there are workflow behaviors (steps) that must be evaluated in sequence, they should be identified with prerequisite conditions, behavior, and post-behavior conditions as below.

Workflow Steps

(Unique, short, representative name of the step)	
Prerequisite Conditions or Step:	*(Conditions or Step name)*
Behavior:	*(Objective, verifiable behavior)*
Post-step Conditions or Next Step:	*(Conditions or Step name)*

Interaction

List any entities or actors that are used or affected by this feature. Each should link to an entry in the OSP Terminology page.

Quality Metrics

Describe any non-functional requirements of the feature, such as usability, performance, or design. Provide objective and, where possible, quantitative measures against which actual implementations can be evaluated.

Assumptions

Provide any assumptions about implementation path, availability of other required features, schedule concerns or otherwise.

Outstanding Issues

The Outstanding Issues section is a placeholder for the evolution of this specific feature. It should mention any explicit design or implementation decisions that are pending. There must be no outstanding decisions as of the confirmation of the feature as a requirement.