Revision History

Version	Change	Author	Date
0.1	Initial draft taken from OpenUP template	Sompop	5/20/11

Detailed Design Template

This template describes how the design can be organized to be understood from multiple perspectives. It also provides suggestions for how patterns and descriptions of small, reusable interactions can be used to minimize redundancy.

It is important not to think of design as "a document." Design information that is worth keeping for some duration must have a long-lived form. But that form might be as a repository in a visual modeling tool, or as subdirectories of whiteboard diagrams captured with a digital camera, or as an actual document that provides structure for images taken from a myriad of sources.

Designs are often organized into requirement realizations. A requirements realization is a part of the design that shows how one or more requirements is implemented.

This template describes the information that should be conveyed. Typically, it works best to convey the information graphically (either with UML or another unambiguous notation), or at least in words, at an abstract level. You can enhance this with code examples, but best not to render the design solely at the code level.

The structure of the design is suggested in this template.

Design structure

[Describe the design from the highest level. This is commonly done with a diagram that shows a layered architecture.]

Backlog Item

[Backlog Item1]

[Describe the design of a portion of the system (a package or component, for instance). The design should capture both static and dynamic perspectives.

When capturing dynamic descriptions of behavior, look for reusable chunks of behavior that you can reference to simplify the design of the requirement realizations.

You can break this section down into lower-level subsections to describe lower-level, encapsulated subsystems.]

Patterns

[Pattern1]

Overview

[Provide an overview of the pattern in words in some consistent form. The overview of a pattern can include the intent, motivation, and applicability.]

Structure

[Describe the pattern from a static perspective. Include all of the participants and how they relate to one another, and call out the relevant data and behavior.]

Behavior

[Describe the pattern from a dynamic perspective. Walk the reader through how the participants collaborate to support various scenarios.]

Example

[Often, you can convey the nature of the pattern better with an additional concrete example.]

Requirement/Architecture realizations

[Realization1]

View of participants

[Describe the participating design elements from a static perspective, giving details such as behavior, relationships, and attributes relevant to this realization.]

Basic scenario

[For the main flow, describe how instances of the design elements collaborate to realize the requirements. When using UML, this can be done with collaboration diagrams (sequence or communication).]

Additional scenarios

[For other scenarios that must be described to convey an appropriate amount of information about how the requirement behavior will be realized, describe how instances of the design elements collaborate to realize the requirement. When using UML, you can do this with collaboration diagrams (sequence or communication).]

Design Decision

[Describe design decision]

Architecture Driver	
Design Decision	
Alternatives	
Rationale	
Concerns	
Design Response	