

# Lawrence Berkeley National Laboratory

## Recent Work

### Title

Use Case Driven Requirements for Reagent Tracking at the JGI

### Permalink

<https://escholarship.org/uc/item/1fp9h5wr>

### Authors

Fazo, Joni  
Alcivare, Dana  
Israni, Sanjay  
[et al.](#)

### Publication Date

2005-09-27



# Use Case Driven Requirements for Reagent Tracking at the JGI

Joni Fazo<sup>1,3</sup>, Dana Alcivare<sup>3</sup>, Sanjay Israni<sup>2,3</sup>, Damon Tighe<sup>2,3</sup>, Christine Lansang Naca<sup>3</sup>, René Perrier<sup>3</sup>, Arkady Voloshin<sup>3</sup> and David Pletcher<sup>1,3</sup>

<sup>1</sup> EEBI Division, CAR  
LLNL

<sup>2</sup>Biosciences Dicatorate  
LLNL

<sup>3</sup>DOE Joint Genome Institute  
(JGI)

*Venomoth is the product name for the new JGI high-throughput DNA sequencing laboratory information management system (LIMS). Once complete, Venomoth will include several subsystems to support, monitor and track the DNA sequencing workflow process. One of the initial subsystems under development is the Reagent Tracking subsystem. This subsystem will allow Production Sequencing staff to receive reagent lots into the LIMS, and provide the Quality Control group with a mechanism for controlling which reagent lots are active on the production line. The Reagent Tracking subsystem also will provide a central repository for recording and tracking reagent quality control activities.*

## Introduction:

Documenting requirements for the Reagent Tracking subsystem involved many challenges:

- How can the subsystem best meet the needs of the JGI Production Sequencing staff?
- How should the subsystem interface with legacy and future subsystems of our LIMS?
- What is the best approach for discovering new requirements and managing changing requirements?
- What is the best approach for creating a requirements specification that is readable by users, yet contains sufficient detail to serve as a basis for software development and verification activities?

## Results:

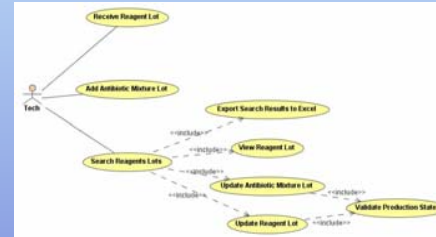
We chose an approach that blends the quality focus of IEEE recommendations for requirements management, with the user focus of use case modeling. The final approved Software Requirements Specification (SRS) documents functional requirements via use cases and non-functional requirements (such as performance and usability requirements) following the traditional IEEE recommendations.

## Methods:

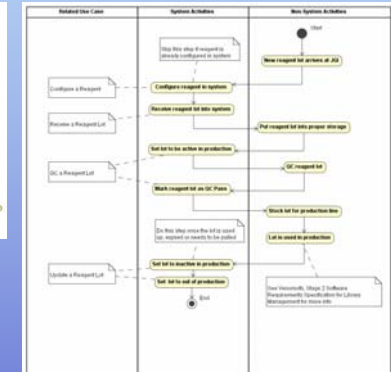
Use case driven requirements using written descriptions, UML diagrams, as well as user-interface mock-ups. Iterative formal review sessions of the SRS that included all stakeholders:

- Users: verified that software will meet their needs
- Management: verified that software fits into project plans
- Software Development: verified that software could be implemented
- Software QA: verified that requirements are testable

## UML Diagrams:



Above: Use case model of laboratory technician activities, Right: Simplified life cycle of reagent lot



## Discussion:

### Pros to our approach:

- Use cases provide a communication tool for stakeholders, users, developers and management, that facilitates agreement on project scope prior to sign-off on investment of development resources.
- Use cases allow for project planning that is modular and iterative.

### Cons to our approach:

- Some requirements are best discovered and communicated via prototyping rather than the written word of an SRS.
- The final SRS was many pages long, very detailed and potentially difficult to maintain.