Use Cases

for

Motherlode

Prepared by SwampCo

V4.0

February 18, 2013

# Guidance for Use Case Template

Document each use case using the template shown in the Appendix. This section provides a description of each section in the use case template.

# Use Case Identification

## Use Case ID

Give each use case a unique integer sequence number identifier. Alternatively, use a hierarchical form: X.Y. Related use cases can be grouped in the hierarchy.

## Use Case Name

State a concise, results-oriented name for the use case. These reflect the tasks the user needs to be able to accomplish using the system. Include an action verb and a noun. Some examples:

* View part number information.
* Manually mark hypertext source and establish link to target.

## Use Case History

### Created By

Supply the name of the person who initially documented this use case.

### Date Created

Enter the date on which the use case was initially documented.

### Last Updated By

Supply the name of the person who performed the most recent update to the use case description.

### Date Last Updated

Enter the date on which the use case was most recently updated.

# Use Case Definition

## Actors

An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor that will be initiating this use case and any other actors who will participate in completing the use case.

## Trigger

Identify the event that initiates the use case. This could be an external business event or system event that causes the use case to begin, or it could be the first step in the normal flow.

## Description

Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case.

## Preconditions

List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each precondition. Examples:

1. User’s identity has been authenticated.
2. User’s computer has sufficient free memory available to launch task.

## Postconditions

Describe the state of the system at the conclusion of the use case execution. Number each postcondition. Examples:

1. Document contains only valid SGML tags.
2. Price of item in database has been updated with new value.

## Normal Flow

Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, “How do I <accomplish the task stated in the use case name>?” This is best done as a numbered list of actions performed by the actor, alternating with responses provided by the system. The normal flow is numbered “X.0”, where “X” is the Use Case ID.

## Alternative Flows

Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative flow, and describe any differences in the sequence of steps that take place. Number each alternative flow in the form “X.Y”, where “X” is the Use Case ID and Y is a sequence number for the alternative flow. For example, “5.3” would indicate the third alternative flow for use case number 5.

## Exceptions

Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use case execution fails for some unanticipated reason. If the use case results in a durable state change in a database or the outside world, state whether the change is rolled back, completed correctly, partially completed with a known state, or left in an undetermined state as a result of the exception. Number each alternative flow in the form “X.Y.E.Z”, where “X” is the Use Case ID, Y indicates the normal (0) or alternative (>0) flow during which this exception could take place, “E” indicates an exception, and “Z” is a sequence number for the exceptions. For example “5.0.E.2” would indicate the second exception for the normal flow for use case number 5.

## Includes

List any other use cases that are included (“called”) by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality.

## Priority

Indicate the relative priority of implementing the functionality required to allow this use case to be executed. The priority scheme used must be the same as that used in the software requirements specification.

## Frequency of Use

Estimate the number of times this use case will be performed by the actors per some appropriate unit of time.

## Business Rules

List any business rules that influence this use case.

## Special Requirements

Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.

## Assumptions

List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.

## Notes and Issues

List any additional comments about this use case or any remaining open issues or TBDs (To Be Determineds) that must be resolved. Identify who will resolve each issue, the due date, and what the resolution ultimately is.

Use Case List

|  |  |  |
| --- | --- | --- |
| Primary Actor | Use Cases | Use Case ID |
| Contributor | Create a nugget | UC.1.CREATE\_NUGGET |
| Content Manager | Moderate nugget | UC.2.MOD\_NUGGET |
| Administrator | Create a user | UC.3.CREATE\_USER |
| Administrator | Edit a user | UC.4.EDIT\_USER |
| Administrator | Delete a user | UC.5.DEL\_USER |
| Administrator | Create a category | UC.6.CREATE\_CAT |
| Administrator | Edit a category | UC.7.EDIT\_CAT |
| Administrator | Delete a category | UC.8.DEL\_CAT |
| Content Manager | Invite contributors | UC.9.INVITE |
| *Authorized* User | Edit a nugget | UC.10.EDIT\_NUGGET |

Use Case Templates

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.1.CREATE\_NUGGET | | |
| Use Case Name: | Create a nugget | | |
| Created By: | Matt Walston | Last Updated By: | Geoff Wark |
| Date Created: | October 5, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Administrator (Current), Approved Contributor\*, Unapproved Contributor | | |
| Description: | This use case defines the creation of content, or ‘nugget’. Ultimately, a user with the ‘contributor’ role will submit a nugget creation form (including text, images, videos, .doc files, etc.). This nugget then enters a moderation queue to be dealt with in UC.2.MOD\_NUGGET. | | |
| Trigger: | * *User* clicks on nugget submission button | | |
| Preconditions: | 1. *User* is logged in 2. User wants to contribute a nugget to a particular category | | |
| Postconditions: | 1. Nugget enters moderation queue | | |
| Normal Flow: | 1. *User* fills out desired information (text content, pictures, videos) 2. *User* submits form 3. System validates submission 4. Nugget is placed in moderation queue / System displays feedback 5. *User* is returned to previously viewed category level 6. Use case ends | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (5). Use case ends.  ---  A2.1 Form submission is invalid (blank entries, bad formats, etc.)  A2.2 Error message is displayed  A2.3 Use case continues at (1).  –-  A3.1 *Unapproved contributor* has gone over their submission limit.  A3.2 Error message is displayed  A3.3 Use case continues at (6). Use case ends. | | |
| Exceptions: | N/A | | |
| Includes: | N/A | | |
| Priority: | 10 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: | High | | |
| Business Rules: | Contributor are labeled internally by a binary attribute that represents an 'approved' or 'unapproved' status. Contributors who are unapproved may only post a set number of nuggets (expected value of 5). | | |
| Special Requirements: | Submission should be possible in fewer than two (2) screens. | | |
| Assumptions: | *User* already has existing account with Motherlode | | |
| Notes and Issues: | Contributor role added and expanded. | | |

\* , indicates *proposed* primary actor

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.2.MOD\_NUGGET | | |
| Use Case Name: | Moderate a nugget | | |
| Created By: | Matt Walston / Nicole Deroin / Geoff Wark | Last Updated By: | Geoff Wark |
| Date Created: | October 5, 2013 | Date Last Updated: | December 11, 2013 |
|  | | | |
| Actors: | Content manager\*, Administrator | | |
| Description: | Defines how nugget moderation from the queue is executed. Default behavior would be to accept a nugget as ‘valuable’. Denying a nugget will either delete it from OR move it out of the queue (to be discussed at a later date). | | |
| Trigger: | * Content manager decides to moderate (approve or deny) a specific nugget | | |
| Preconditions: | 1. Nugget(s) exist in the queue to be moderated. | | |
| Postconditions: | 1. A nugget is moderated, and moved out of the queue. | | |
| Normal Flow: | 1. Content Manager logs into Motherlode system. 2. Content Manager navigates to desired category level 3. Content Manager selects nugget 4. Content Manager approves nugget 5. Systems moves approved nugget to ***[TBD]*** / Systems displays feedback 6. System points Content Manager back to previously desired category level. 7. Use case ends. | | |
| Alternative Flows: | A1.1 From (3), Content Manager may choose to deny nugget.  A1.1a Denied nuggets will be moved to ***[TDB]***  **---**  A2.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A2.2 Use case continues at (6). Use case ends. | | |
| Exceptions: |  | | |
| Includes: | UC.1.CREATE\_NUGGET | | |
| Priority: | 9 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: | High | | |
| Business Rules: | 1. A content manager is assigned to a specific category(ies)    1. If none is, an effective manager would be another content manager higher up the hierarchy (up to Admin-level) 2. Some level of ‘quality control’ exists. To be determined later. (May not be driven by the system.) | | |
| Special Requirements: | Moderation should be possible within two (2) screens. | | |
| Assumptions: |  | | |
| Notes and Issues: | Basic implementation complete. Pushing authorized nuggets to a final front-end to be worked on in a later, related project. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.3.CREATE\_USER | | |
| Use Case Name: | Create a user | | |
| Created By: | David King, Jr. | Last Updated By: | Geoff Wark |
| Date Created: | October 1, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | \*Non-user | | |
| Description: | This use case describes the user creation process. | | |
| Trigger: | * An ‘authorized’ user desires to add a new user to Motherlode | | |
| Preconditions: | 1. Non-user wishes to create account | | |
| Postconditions: |  | | |
| Normal Flow: | 1. Non-user logs into Motherlode 2. Non-user clicks 'create account' button 3. Non-user submits new user information (including name, e-mail, phone, ROLE, etc.) 4. System adds new *Unauthorized Contributor* to database. / displays feedback 5. System points new *Unauthorized Contributor* (previous Non-user) to homepage. 6. End of use case | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (7). Use case ends.  ---  A2.1 Form submission is invalid (blank entries, bad formats, etc.)  A2.2 Error message is displayed  A2.3 Use case continues at (4). | | |
| Exceptions: |  | | |
| Includes: |  | | |
| Priority: | 8 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: | Moderate | | |
| Business Rules: | 1. An ‘authorized’ user stems from the user hierarchy (Admin > Content Manager > Contributor; subject to change). A role of equal, or above, value cannot be assigned. | | |
| Special Requirements: |  | | |
| Assumptions: | 1. | | |
| Notes and Issues: | User creates their own account now. Admin/”Authorized user” is not required. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.4.EDIT\_USER | | |
| Use Case Name: | Edit a user | | |
| Created By: | David King, Jr. | Last Updated By: | Geoff Wark |
| Date Created: | October 1, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Administrator\*, Content Manager | | |
| Description: | Describes the process of editing a user’s profile. This can include changing name, username, contact information or promoting/demoting their role. | | |
| Trigger: | * A user of Motherlode needs to have some aspect of their profile changed. | | |
| Preconditions: | 1. A user profile is incorrect, out-of-date, needs revision, etc. | | |
| Postconditions: | 1. Desired user profile is updated. | | |
| Normal Flow: | 1. ‘Authorized’ user logs into Motherlode 2. ‘Authorized’ user navigates dashboard to list of users 3. ‘Authorized’ user clicks an “edit user” button 4. ‘Authorized’ user makes changes to user information (including name, e-mail, phone, ROLE, etc.) 5. System applies changes to database. / displays feedback 6. System points ‘authorized’ user back to list of users page 7. End of use case | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (7). Use case ends.  ---  A2.1 Form submission is invalid (blank entries, bad formats, etc.)  A2.2 Error message is displayed  A2.3 Use case continues at (4). | | |
| Exceptions: |  | | |
| Includes: | UC.3.CREATE\_USER | | |
| Priority: | 7 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: |  | | |
| Business Rules: | 1. An ‘authorized’ user stems from the user hierarchy (Admin > Content Manager > Contributor; subject to change). A role of equal, or above, value cannot be modified. | | |
| Special Requirements: |  | | |
| Assumptions: | 1. | | |
| Notes and Issues: | Basic implementation completed. | | |
|  |  | | |
| Use Case ID: | UC.5.DEL\_USER | | |
| Use Case Name: | Delete a user | | |
| Created By: | David King, Jr. | Last Updated By: | Geoff Wark |
| Date Created: | October 1, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Administrator\*, Content Manager | | |
| Description: | Explains how the deletion of users takes place. | | |
| Trigger: | * A user account requires deletion | | |
| Preconditions: | 1. A user account is determined to require deletion (for whatever reason? ***TDB***) | | |
| Postconditions: | 1. A user account is permanently deleted from the database | | |
| Normal Flow: | 1. ‘Authorized’ user logs into Motherlode 2. ‘Authorized’ user navigates dashboard to list of users 3. ‘Authorized’ user clicks a “delete user” button 4. ‘Authorized’ user confirms deletion 5. ‘Authorized’ user confirms deletion for a second (2nd) time 6. System applies changes to database. / displays feedback 7. System points ‘authorized’ user back to list of users page 8. End of use case | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (8). Use case ends. | | |
| Exceptions: |  | | |
| Includes: | UC.3.CREATE\_USER | | |
| Priority: | 7 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: |  | | |
| Business Rules: | 1. An ‘authorized’ user stems from the user hierarchy (Admin > Content Manager > Contributor; subject to change). A role of equal, or above, value cannot be modified. | | |
| Special Requirements: |  | | |
| Assumptions: | 1. | | |
| Notes and Issues: | Basic implementation completed. | | |
|  |  | | |
| Use Case ID: | UC.6.CREATE\_CAT | | |
| Use Case Name: | Create a catagory | | |
| Created By: | Geoff Wark | Last Updated By: | Geoff Wark |
| Date Created: | October 7, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Administrator\*, content manager(?) | | |
| Description: | This use case describes the process of creating/adding categories to the hierarchy | | |
| Trigger: | * A new category (product or lower level) is chosen to be added by [someone]. | | |
| Preconditions: | 1. There exists a deficiency in the category hierarchy such that a new category must be created | | |
| Postconditions: | 1. The desired category has been created/added to the hierarchy | | |
| Normal Flow: | 1. ‘Authorized’ user logs into Motherlode 2. ‘Authorized’ user navigates dashboard to list of categories (or a point in the hierarchy) **TBD** 3. ‘Authorized’ user clicks an “add category” button 4. ‘Authorized’ user submits new category information (including name, location, etc..) 5. System adds new category to hierarchy. / displays feedback 6. System points ‘authorized’ user back to [previously viewed page during (3)] 7. End of use case | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (7). Use case ends.  ---  A2.1 Form submission is invalid (blank entries, bad formats, etc.)  A2.2 Error message is displayed  A2.3 Use case continues at (4). | | |
| Exceptions: |  | | |
| Includes: | N/A? | | |
| Priority: | 7 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: | Rarely | | |
| Business Rules: | 1. An ‘authorized’ user stems from the user hierarchy (Admin > Content Manager > Contributor; subject to change) and if the Content Manager has permissions over [what categories]. A category of higher value than permissions granted cannot be created. *(this needs to be discussed in detail later…)* | | |
| Special Requirements: | ? | | |
| Assumptions: | 1. | | |
| Notes and Issues: | Basic implementation completed. | | |
|  |  | | |
| Use Case ID: | UC.7.EDIT\_CAT | | |
| Use Case Name: | Edit a catagory | | |
| Created By: | Geoff Wark | Last Updated By: | Geoff Wark |
| Date Created: | October 7, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Content Manager\* (?), Administrator | | |
| Description: | Describes the process of editing an existing category. | | |
| Trigger: | * [Someone] makes the determination that a category must be edited in some way (title, placement in hierarchy perhaps?, etc.) | | |
| Preconditions: | 1. Existing category must be changed in some way… | | |
| Postconditions: | 1. Category information has been updated | | |
| Normal Flow: | 1. ‘Authorized’ user logs into Motherlode 2. ‘Authorized’ user navigates dashboard to list of categories (or a point in the hierarchy) **TBD** 3. ‘Authorized’ user clicks an “edit category” button 4. ‘Authorized’ user submits updated category information (including name, location, etc..) 5. System updates category. / displays feedback 6. System points ‘authorized’ user back to [previously viewed page during (3)] 7. End of use case | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (8). Use case ends.  ---  A2.1 Form submission is invalid (blank entries, bad formats, etc.)  A2.2 Error message is displayed  A2.3 Use case continues at (4). | | |
| Exceptions: |  | | |
| Includes: | UC.6.CREATE\_CAT | | |
| Priority: | 6 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: | Rarely | | |
| Business Rules: | 1. An ‘authorized’ user stems from the user hierarchy (Admin > Content Manager > Contributor; subject to change) and if the Content Manager has permissions over [what categories]. A category of higher value than permissions granted cannot be edited. *(this needs to be discussed in detail later…convoluted)* | | |
| Special Requirements: | ? | | |
| Assumptions: | 1. Category to be edited already exists. | | |
| Notes and Issues: | Basic implementation completed. | | |
|  |  | | |
| Use Case ID: | UC.8.DEL\_CAT | | |
| Use Case Name: | Delete a catagory | | |
| Created By: | Geoff Wark | Last Updated By: | Geoff Wark |
| Date Created: | October 7, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Administrator\*, Content Manager (?) | | |
| Description: | Describes the process of deleting a category from the hierarchy | | |
| Trigger: | * [Someone] makes the determination that a category must be deleted | | |
| Preconditions: | 1. Category to be deleted exists | | |
| Postconditions: | 1. Desired category is deleted | | |
| Normal Flow: | ‘Authorized’ user logs into Motherlode   1. ‘Authorized’ user navigates dashboard to list of categories (or a point in the hierarchy) **TBD** 2. ‘Authorized’ user clicks a “delete category” button 3. ‘Authorized’ user confirms deletion 4. System deletes category. / displays feedback 5. System points ‘authorized’ user back to [previously viewed page during (3)] 6. End of use case | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (7). Use case ends. | | |
| Exceptions: |  | | |
| Includes: | UC.6.CREATE\_CAT | | |
| Priority: | 5 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: | Very rarely | | |
| Business Rules: | 1. An ‘authorized’ user stems from the user hierarchy (Admin > Content Manager > Contributor; subject to change) and if the Content Manager has permissions over [what categories]. A category of higher value than permissions granted cannot be deleted. | | |
| Special Requirements: |  | | |
| Assumptions: | 1. | | |
| Notes and Issues: | Basic implementation completed. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.9.INVITE | | |
| Use Case Name: | Invite a contributor | | |
| Created By: | Uday Allika | Last Updated By: | Geoff Wark |
| Date Created: | October 6, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Admin/Content Manager(?) | | |
| Description: | This use case goes over the methods used to invite a contributor to the Motherlode system using an outside/implemented e-mail system (sendgrid? Maybe… ITS will provide a solution; ***TBD***). | | |
| Trigger: | * A non-user wants or join the system and has contacted a user, or… * … *needs to be discussed* | | |
| Preconditions: | 1. Non-user desires to be a user (contributor) | | |
| Postconditions: | 1. Non-user has been invited through email to create a Contributor account | | |
| Normal Flow: | 1. … 2. Email invitation is sent / System provides feedback 3. Use case ends | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (X). Use case ends. | | |
| Exceptions: |  | | |
| Includes: | *UC.3.CREATE\_USER?* | | |
| Priority: | 2 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: |  | | |
| Business Rules: |  | | |
| Special Requirements: |  | | |
| Assumptions: | 1. | | |
| Notes and Issues: | **THIS FUNCTIONALITY HAS BEEN SCRAPPED** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.10.EDIT\_NUGGET | | |
| Use Case Name: | Edit a nugget | | |
| Created By: | Nicole DeRoin | Last Updated By: | Geoff Wark |
| Date Created: | October 5, 2013 | Date Last Updated: | February 18, 2014 |
|  | | | |
| Actors: | Contributor\*, Content Manager, Administrator | | |
| Description: | Describes the process of editing a nugget which has been submitted to the moderation queue. | | |
| Trigger: | * Contributor decides to edit a previously submitted nugget (NOT yet moderated), or… * ‘Authorized’ user decides to edit a submitted nugget | | |
| Preconditions: | 1. Desired nugget is already submitted 2. Desired nugget is still in moderation queue (aka. NOT moderated) | | |
| Postconditions: | 1. Desired nugget has been edited in some way (title, content, etc.) | | |
| Normal Flow: | 1. *User* logs into Motherlode system. 2. *User* navigates to desired category level 3. *User* selects nugget 4. *User* edits nugget (title, content, etc.) 5. Systems applies edits to nugget in moderation queue / Systems displays feedback 6. System points *User* back to previously desired category level. 7. Use case ends. | | |
| Alternative Flows: | A1.1 Cancellation occurs at any point in the normal flow (closes browser, power outage, etc.)  A1.2 Use case continues at (X). Use case ends.  ---  A2.1 Form submission is invalid (blank entries, bad formats, etc.)  A2.2 Error message is displayed  A2.3 Use case continues at (4). | | |
| Exceptions: |  | | |
| Includes: | UC.1.CREATE\_NUGGET, UC.2.MOD\_NUGGET | | |
| Priority: | 4 (10 point scale where 1 = lowest, 10 = highest) | | |
| Frequency of Use: |  | | |
| Business Rules: | *Must be discussed with individual clients to determine who can edit and when/where/why/how/etc.* | | |
| Special Requirements: |  | | |
| Assumptions: | 1. Nugget to be edited already exists. | | |
| Notes and Issues: | Basic implementation. | | |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Initial creation | 1-Oct-13 | Initial creation | 1.0 |
| Update – Uday | 6-Oct-13 | Added Uday’s contributions | 1.1 |
| Update – Nicole/Geoff | 7-Oct-13 | Added Nicole and Geoff’s contributions. Minor revisions to Uday’s work | 1.2 |
| Rework | 7-Oct-13 | Rework begun by David. | 1.3 |
| Rework update | 8-Oct-13 | Rework complete by Geoff | 1.3.1 |
| Editing process | 8-Oct-13 | Checks for typos/minor issues | 1.3.2 |
| Final preparation for sprint 1 | 9-Oct-13 |  | 1.4-FINAL |
| Updates for sprint 2 | 11-Nov-13 | Changes made for professor comments and because of updated functionality | 2.0 |
| Updates for spring 3 | 11-Dec-13 | Changes made to reflect updated functionality at the symposium | 3.0 |
| Updates for spring 4 | 18-Feb-14 | Changes made to reflect updated functionality (specifically involving 'approved' users) | 4.0 |