Rationale Management Use Cases
Lauren DiCristofaro
1/21/14

This is further documentation adding to the initial proposal on rationale management. Below are use cases that illustrate how Rationale Management can be incorporated into the system. Further use cases may be added in the future to give the user more control and insight into project rationale.

Level of Effort Estimations:
For most team members, LoE estimations should include some time to become more familiar with the system and technologies and time to handle testing, bug fixes, and unexpected pitfalls.

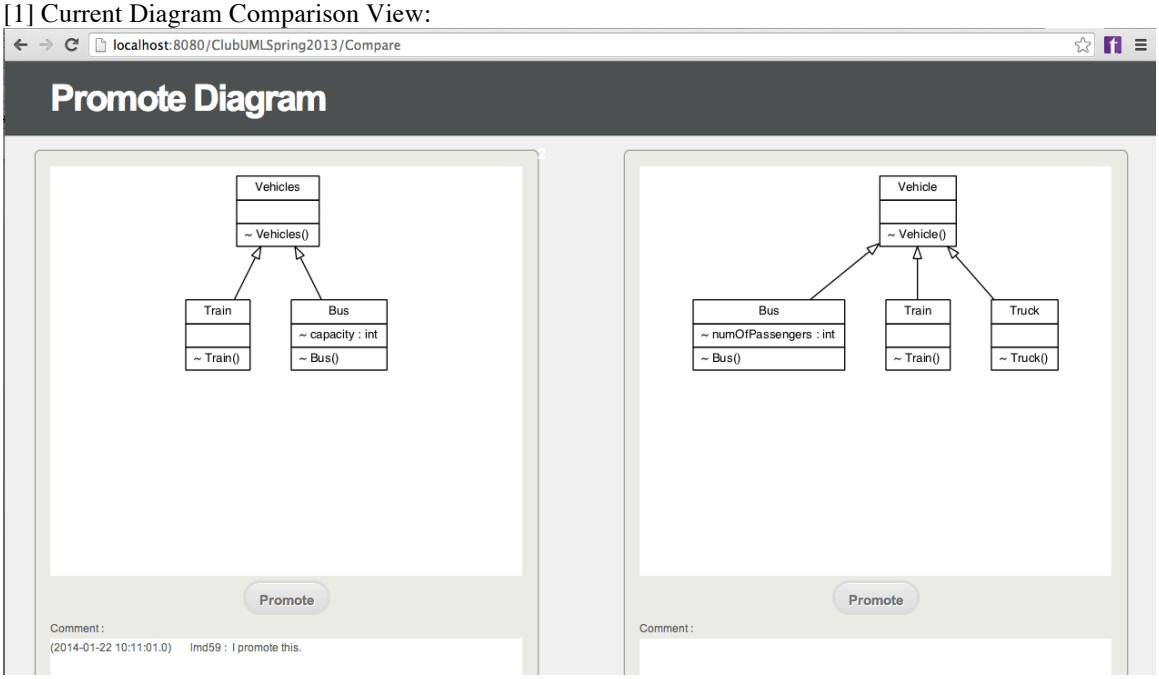
RM1

Use Case Element	Description
Use Case Number	RM1
Application	ClubUML Application
Use Case Name	Create Rationale for Diagram Promotion
Use Case Description	A Team Member wishes to promote a diagram as the best choice to describe a design or requirements. In the process of promoting the diagram, they select rationale parameters to justify their decision.
Primary Actor	Team Member
Precondition	There must be viable diagrams uploaded to the system. The team member must have logged into ClubUML application and navigated to the diagram comparison view[1].
Trigger	The user clicks the promote button.
Basic Flow	1. The user clicks the promote button 2. The application displays a rationale dialog box 3. The rationale dialog box prompts for [none or more] rationale Alternative(s), a Summary [required], Issue, Issue Relationship, Criteria, and Criteria Relationship. 4. The user chooses an Alternative from the other diagrams already uploaded [default to the other diagram in the diagram comparison view]. 5. The user writes in a Summary, Issue, Issue Relationship, Criteria, and Criteria Relationship. 6. The user clicks the OK button. 7. The application adds the promotion and rationale to the database. 8. The application redirects to the diagram comparison view[1] where details of the rationale can be viewed beneath the diagram they promoted.
Alternate Flows	The most significant alternatives and exceptions If in step 5, the user does not write a summary, 1. The user clicks the OK button. 2. An error dialog opens directing the user to input a rationale summary. 3. The user clicks the error dialog OK button

	4. The application removes the error dialog and continues to display the rationale dialog. If the user clicks the CANCEL button during any time in which the rationale dialog is open, 1. The application closes the rationale dialog. 2. No changes are made to the comparison screen or the underlying database

Potential Future: link issues and criteria to a project management tool.

[1] Current Diagram Comparison View:



Level of Effort

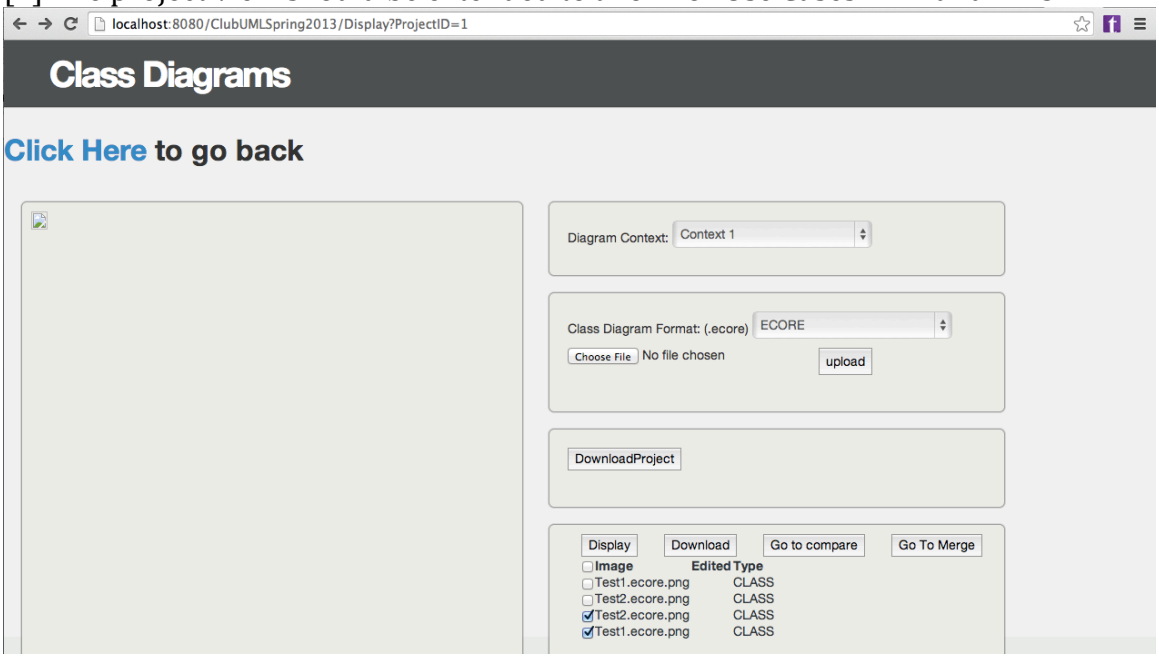
RM1 involves creating a new dialog box, which involves doing the design and functionality behind displaying it. The dialog must display information from the database and collect user input data to store in the database. The database must be extended to include Rationale objects which have a relationship with existing database items (diagrams, promotions, project...). This use case also involves limited validation.

Estimation: Best case = 80 person hours

RM2

Use Case Element	Description
Use Case Number	RM2
Application	ClubUML Application
Use Case Name	Select Decision Diagram
Use Case Description	The team lead wishes to replace (change decision for) a diagram as the ground truth that will reflect current development goals and intentions.
Primary Actor	Team Lead
Precondition	There are currently decision diagrams and alternative diagrams uploaded to the system. The team lead has logged in and navigated to the project tab.
Trigger	The team lead clicks on a project link to the project for which they wish to select a decision diagram.
Basic Flow	1. The application directs the team lead to the project view[2]. 2. The project view shows a list of current decision diagrams. 3. The user selects one current decision diagram and clicks the UPDATE DECISION button. 4. The application presents an update decision dialog. 5. The update decision dialog prompts the user to pick a diagram from the currently uploaded diagrams (excluding the currently selected decision diagram). 6. The user selects a diagram as the new decision. 7. The decision dialog prompts the user to select [none or more] rationales from the existing rationales associated with the newly selected diagram. 8. The user selects all relevant rationales and clicks OK. 9. The application saves the new decision without deleting the old decision that was selected in step 3. The application instead saves a reference to the predecessor decision. 10. The application redirects to a project view page that shows the updated decision.
Alternate Flows	The most significant alternatives and exceptions If the decision diagram is new and not replacing an old decision diagram, 1. In the project view, the team lead clicks the NEW DECISION button. 2. The application shows a NEW DECISION dialog. 3. The new decision dialog prompts the user to pick a diagram from the currently uploaded diagrams 4. See basic flow steps 6-8 5. The application saves the decision without a predecessor decision and redirects to a project view page that shows the new decision. If the user clicks the CANCEL button during any time in which a decision dialog is open, 1. The application closes the decision dialog. 2. No changes are made to the project view screen or the underlying database. If in step 3, the user selects multiple decision diagrams to update, 1. 1. The application opens an error dialog directing the user to select a single decision to update. 2. The user clicks the error dialog OK button 3. The application removes the error dialog and continues to display the project view.

[2] The project view should be extended to allow for Use Cases RM2 and RM3



Level of Effort:

RM2 involves adding both Rationale and Decision objects to the database with a relationship between each of these and existing database items (diagrams, promotions, etc.), and there's also relationships between Decision items and Rationale. This use case involves two new (but related) dialogs which need to be designed and integrated into the current application. The dialogs show information from the database and store user input to the database. This also involves a limited amount of validation. RM2 implies some functionality from RM1 which allows users to create Rationales, but it could potentially be done with the current Promotions instead of Rationales.

Estimation: Best case = 80 person hours (given RM1 completed or using Promotions in place of Rationales)

RM3

Use Case Element	Description
Use Case Number	RM3
Application	ClubUML Application
Use Case Name	Generate Rationale History Documentation
Use Case Description	A Team Member wishes to generate documentation of the history of decisions that led to the current ground truth including the relevant rationale that went into each of those decisions.
Primary Actor	Team Member
Precondition	The team member has logged in and navigated to the project tab.
Trigger	The team member clicks on a project link to the project for which they wish to generate decision history documentation.
Basic Flow	1. The application directs the team member to the project view[2]. 2. The project view shows a list of current decision diagrams. 3. The team member clicks the GENERATE DOCUMENTATION button. 4. The application generates an html document that shows the rationale details of all decisions made and indicates the predecessor decision (if it exists). 5. The application directs to a decision history page which shows a DOWNLOAD HTML button, a BACK TO PROJECT button, and the generated html document within a subview 6. The team member clicks the DOWNLOAD HTML button. 7. The application downloads the html file to the users computer. 8. The team member clicks the BACK TO PROJECT button. 9. The application redirects the team member back to the project view.
Alternate Flows	If there are not yet any decisions, 1. The decision history page shows a blank html file in the subview.

Potential Future: Show decision history in a friendly format (not text).

Level of Effort:

RM3 assumes functionality from RM2 and RM1 which allow users to create Rationales and Decisions. RM1 and RM2 add data to the database, whereas this use case retrieves data that's been added. It involves creating a new page (designing and integrating with current system), and it involves developing and populating a document format for the existing data and adding the ability to download that document.

Estimation: Best case = 60 hours (assuming database has already been extended)