

MarkUs Usability and Interface Design Plan

By Victoria Mui

Section A: Project-wide

Topic A1: Upload/Download CSV file – Replace Accordion Menu with Button + Modal Dialogue

Description: To replace the current accordion menu upload/download options with upload and download buttons running along the top of the view. Let's call these "Action Buttons".

Next Steps:

- These action buttons will spawn a modal dialogue (similar to the interface we currently have for adding a new student to a group in the *Groups & Graders* view), where the user can specify the location of the CSV file to upload or download.
- To avoid confusion and maintain consistency, will have to apply this migration throughout the entire project.

Rationale: Having the accordion menu on the side will greatly affect the readability of the 2-pane Assign Graders view. Also, the accordion menu has caused confusion amongst users in the past, so switching to modal dialogues may improve the usability of MarkUs as a whole.

Topic A2: Reorganize Action Links & Action Buttons

Note: "Action Links" refer to the action items to the right of the page header located at the top of every view.

Description: Current prototype inherits action links from the existing release of MarkUs. However, if we were to replace the accordion menu with action buttons running along the top of the view (right-to-left), these may interfere with the existing action links we already have running from the left to the right.

Next Steps: Compile a list of action links and buttons that we will have for each view. Analyze the list and see how we can categorize, trim down, or reorganize the layout of these options.

Rationale: As mentioned in the description, the left-to-right action links may potentially interfere with the right-to-left action buttons.

Section B: Assign Graders Tab

Topic B1: Do Not Need a Review/Read-Only Mode

Description: In the prototype, users are automatically put into a read-only mode so they can view existing grader-to-group assignments.

Next Steps: Users will now be automatically put into the edit mode where they can create new grader-to-group assignments and make changes to the existing ones.

Rationale: Typically, users only use the Assign Graders feature during the initial setup/creation of a new assignment. And if they were to revisit this view, it would probably be to make changes to the current grader-to-group assignments.

Topic B2: Allow for Multiple Grader Selection

Description: In the current prototype, more than one grader can be selected in the *Graders* pane. But do we have use cases where more than one grader would be assigned to the same group of students? Should these checkboxes be radio buttons instead?

Next Steps: Keep the checkboxes and allow for multiple users to be selected!

Rationale: It's very possible that more than one grader would be assigned to the same group of students. Also, we will be implementing a feature that helps the user evenly distribute selected groups amongst selected graders.

Topic B3: Introduce "Select All" and "Clear All" Buttons

Description: These were not introduced in the current prototype.

Next Steps: Need to add "Select All" and "Clear All" options for both the *Graders* pane and the *Groups* pane.

Rationale: User(s) most likely wouldn't want to manually select all the graders and/or groups.

Topic B4: Evenly Distribute Selected Groups Amongst Selected Graders

Description: As mentioned in Topic B2, the current prototype does not support this feature. We will be adding this feature, but how should this be implemented?

Next Steps: Possibilities:

- Have a button that allows users to select the next n groups that they would like to assign graders to.
- Select a set of graders and evenly distribute the (selected) groups sequentially or randomly amongst them.

Rationale: A useful feature that is implemented in the current release of MarkUs, and is one that we will definitely want to keep.

Topic B5: Include Graders' Full Names (In Addition to User IDs)

Description: (Feature Request) The existing release of MarkUs only lists the user IDs of the graders in the *Groups & Graders* view. With a large number of graders, it becomes hard to differentiate between graders solely by their user IDs.

Next Steps: Add full names of graders to the *Graders* pane.

Rationale: To make it easier for users (in particular, instructors) to differentiate between graders.

Section C: Group Formation Tab**Topic C1: Provide Student List for Group Creation**

Note: *This change will be made in the grader and instructor interfaces only. Due to privacy issues, students will not have access to the class list when they are creating the groups themselves. Students will be expected to know their group members' user IDs.*

Description: The user currently has to draw on their own records for the user IDs of the students that they would like to put into a group. This is mainly because the list of (unassigned) students is not currently made available to the user at the time of group creation.

Next Steps: Rework the existing *Group* formation tab interface so that users have access to the list of (unassigned) students when they are creating a new group. One possible option is to use a layout similar to the prototyped *Assign Graders* view where the list of (unassigned) students will be placed in a pane where users can make multiple selections.

Rationale: To prevent the user from having to draw on their own records for the current list of (unassigned) students.

Topic C2: Provide a List of Unassigned Students When Adding Members to an Existing Group

Note: *This change will be made in the grader and instructor interfaces only. Students will not be allowed to make changes to their groups. Only graders and instructors have permission to do so.*

Description: The user currently has to draw on their own records for the user IDs of the students that they would like to add to an existing group. A read-only list of unassigned students is provided to the user in the current release of MarkUs. However, the user loses access to this list when they actually have to enter the user IDs of these students.

Next Steps: There are various ways of addressing this problem. However, we need to come up with a way of visualizing large class lists (e.g., possibly 500+ students) such that graders/instructors can easily find the student(s) that they are looking for. Some ideas:

- Drop down menu
 - However, finding a specific student in a large list of unassigned students may be extremely difficult (with a lot of scrolling involved).
- Drop down menu + search feature

Rationale: To prevent the user from having to draw on their own records for the current list of unassigned students.

Others

Research/Field Work:

- Ask users about their most wished for shortcuts/features (e.g., shortcut to evenly distribute groups amongst graders)
- Go through different views and tally the number of items for the accordion menu in that view.
 - Purpose: To see if replacing accordion menu with buttons is a feasible. This would also give us a sense of how big the change will be if we apply this to the entire project. We may also discover existing options in the accordion menu that works best in an accordion menu.