Events/Notifications

**Events that should generate an event and notification**

**Artifacts**:

1. Evaluator is notified when an Artifact is submitted as evidence.
2. Evaluatee is notified when the Evaluator rejects an Artifact.
3. Evaluator is notified when the Evaluatee responds to an Artifact rejection communication.
4. Evaluatee is notified when the Evaluator responds to an Artifact rejection communication.
5. Evaluatee is notified when the Evaluator finalizes the evaluation of an Artifact.
6. Evaluator is notified when the Evaluatee finalizes the evaluation of an Artifact and has shared the evaluation with his Evaluator.

**Observations**:

1. Evaluatee is notified when a new observation is created.
2. Evaluatee is notified when the Evaluator finalizes conference prompts.
3. Evaluatee is notified when the observation notes are shared.
4. Evaluatee is notified when the observation scoring is shared.
5. Evaluatee is notified when the observation transitions to the “Teacher Review” work state.
6. Evaluatee is notified when the observation transitions to the “Locked” work state.
7. Evaluatee is notified when the Evaluator requests an “Unlock”.
8. Evaluator is notified when the Evaluatee requests an “Unlock”.
9. Evaluator is notified when the Evaluatee approves an “Unlock”.
10. Evaluatee is notified when the Evaluator approves an “Unlock”.

**Student Growth Goals**:

1. Evaluator is notified when the Evaluatee submits a goal.
2. Evaluator is notified when the Evaluatee attaches an artifact to a submitted goal.
3. Evaluatee is notified when the Evaluator finalizes the evaluation of a goal.

Recent Activity Tracking

We will display the last ten items that the user has modified. The item types that we will track are:

1. SEEvalSession
2. SEArtifactBundle
3. SEStudentGrowthGoalBundle

Consider implementing as a LastModified table with the following fields:

* ModifiedByUserID
* ObjectID
* ObjectTypeID (can use LinkedItemType FK)
* TimeStamp

We need to display the following information for each item

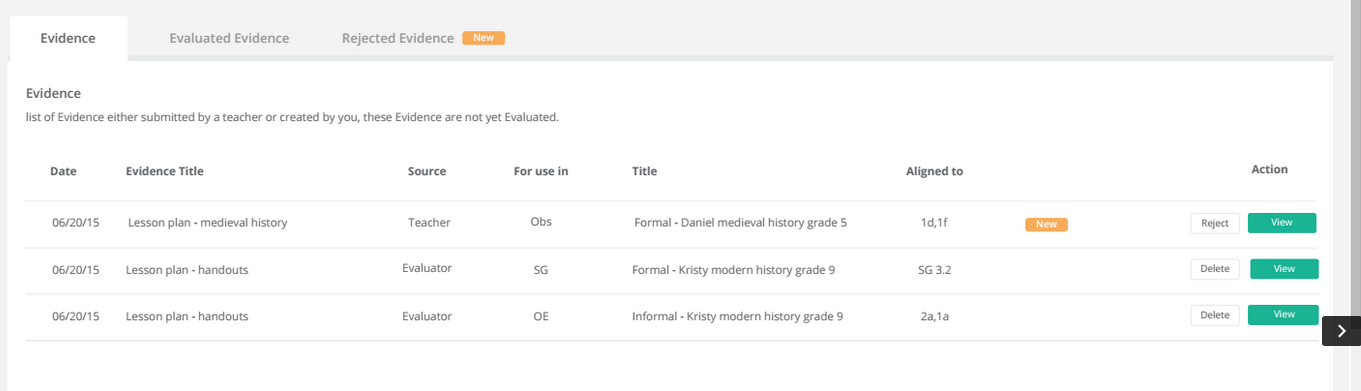
* TimeStamp
* Owner Name
* Object Type
* Object Title
* Url to view item

In order to implement this, we will need to pass the current UserID in the header to the WEB-API with each call.

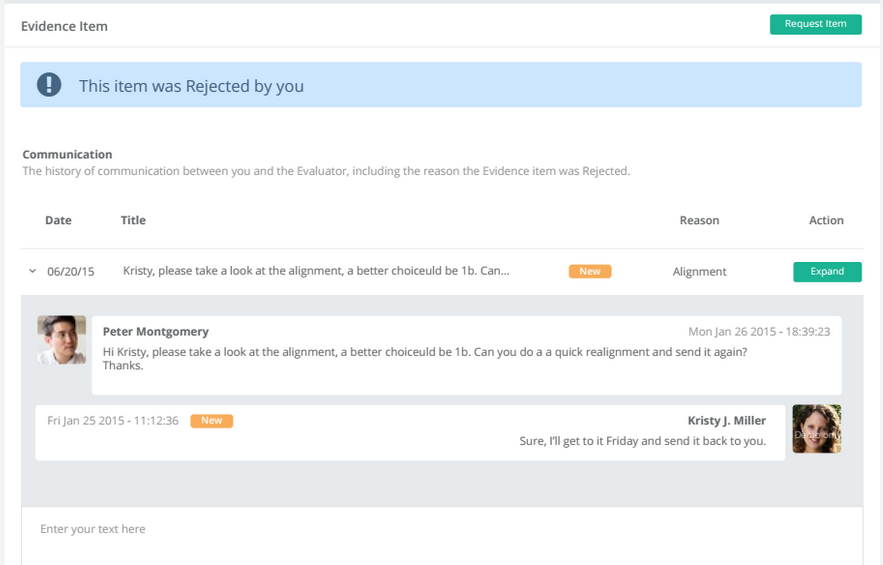
Visited New Item Tracking

Users will receive a notification for changes to an object. But we also want to track that they have navigated to view the object referred to in the notification. Until they visit the page, the item should continue to show a “New” label in the lists that display the item.

For example, a principal will receive a notification when a teacher submits an Artifact as Evidence. But he should also see a “New” label next to that Evidence in his Evidence list until he has viewed that Evidence page.



Same with notifications for new communication messages for an Artifact rejection.



A possible implementation could be to add the column SENotification.ObjectViewed. This is in addition to the column SENotification.IsViewed, which is tracking whether the notification has been viewed.