

Maria Molloy, Tori Spychalski and Daniel Simpson  
11/30/18  
Software Development II

### Limbo User Stories Implementation

For our Limbo project, some of the user stories we focused on changed since our group changed. The user stories we chose to focus on were about finder and owner convenience while entering their lost items. For example, we thought it was very important for a user of our system to be able to easily find any part of the site at any time. Because of this we included the toolbar at the top of every page so that users can access other pages no matter what page they are on. Furthermore, for whichever page is active, that tab on the toolbar changes colors so users will not mix up what page they are on. This is especially important for entering lost or found items, since it would be very unfortunate for a user to enter an item they lost onto the found items page. Because of this, it is easy for a user to make sure which page they are on when entering an item.

Another user story we focused on was the user easily being able to input where they lost/found their item so that it would be cohesive in the database and it would eliminate any vagueness or questions so that the item could be found or returned faster. To address this user story, we added drop down menus for the “location lost” and “location found” entries in the report forms. The drop down has every official building on Marist’s campus so that a user will have every option they need.

We implemented the given user stories as well. For example, one user story is for a owner/finder to know the status of their item. We prioritized this by including a table of recently lost and recently found items on our landing page. Because of this, users do not need to log in or navigate to a different page to quickly check for their item. This is important because users may

need to check for an item quickly and immediately, for example on their phone or on a shared computer. Another example is the admin user being able to add or delete other admin users. We implemented this by creating a boolean value in the database table for users to check if they are an admin or not. When someone creates an account this value defaults to not an admin, however other logged in admins can easily change this by accessing the database.