

PostIt - Phase 1

Description:

The project will be a render that will solve the problem of over complicated, over saturated education spaces online. Currently, students at WLU are using Desire2Learn's MyLearningSpace (MyLS). The audience that uses MyLS have frustrations of how there are too many links and too many buttons. Often it is hard to find the exact links that students or professors actually need. However since MyLS has a lot of features, which are all needed in different courses, we have decided to not proceed with another application for this audience. Our render will instead be a better, simpler version targeted to a younger audience that will better use the simpler version. We have decided focus our implementation strategy for high school students and teachers. Our design will strip all the unnecessary features and buttons that MyLS has, this way students/teachers only see what they really need for their class. The website will have an initial login screen where only three types of users can login; the school IT administrator, a teacher, and students. Upon login, the user will see a list of the courses they are registered in. The administrator will see a list of all courses and flags for when there are concerns they need to review.

Once a user selects which course they want to go to, a new page will appear for that specific course. For the students, they will see four buttons for; announcements, course content, dropbox (or similar assignment submission software), and course calendar. For teachers, they will see three buttons for; announcements, course content, dropbox (or similar assignment retrieval software), and their course calendar. The difference between the two will be that the teachers can write to the pages and post content but students can only view it (except for dropbox pages). For the dropbox page, only students can submit documents (which is assumed to be assignments). Administrators are the last user, who have privileges to suspend content/hide content and suspend users. They will see a list of all the courses and if inappropriate content is posted by either teachers or students, warnings will appear. The system administrator will click on the warnings and it will show what the deemed inappropriate content was or whether there was an inappropriate chat. The system administrator will have the privileges to only view users and the content they posted if its been caught as sensitive content or the messages/content was reported by the students or teachers.

Milestone

1. Create web page mockups using wireframing.
 - a. This will create an image for what we want our application to look like from each use case.
2. Create website template using HTML/CSS/javascript.
3. Implement a client-server web application that allows you to choose what happens with different GET requests after a user has logged in.
4. Research and implement a basic chat messaging application.
 - a. Add features like report/delete chat.
 - b. Add ability to download chat transcripts as a pdf.
 - c. Enable chat history deletion off account after 48 hours (if no warnings are present or reported to system administrator).
5. Add ability for students to submit documents through the assignments portal.
 - a. Will need complex checking that tests submission (idea is to prevent students from submitting malware).
6. Polish final site implementation and ensure browser support for Chrome and Firefox (Edge as well, not important though because IE is consulting google to make a new google-type microsoft edge).
7. Add calendar feature (time permitting)

Timeline

Milestones	Due Date:
1	Feb 9, 2019
2	Feb 13, 2019
3	Feb 20, 2019
4	March 3, 2019
5	March 10, 2019
6	March 17, 2019
7	March 24, 2019

