COS Lab TA Planner: Centralized Platform for COS Lab TAs to Manage Their Work Shifts and Notifications

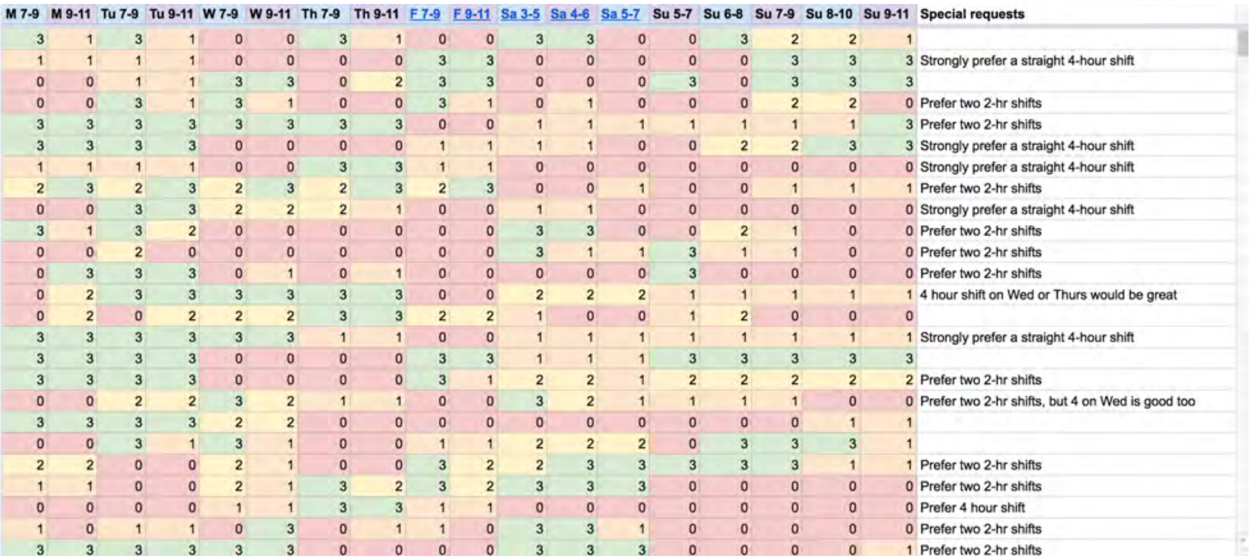
Author: Jonathan Yang
Adviser: Robert Dondero

Abstract

This paper details the design, development, and evaluation of an interactive web platform designed for Princeton COS lab TAs to add, drop, swap and substitute work shifts, create announcements and smartly provide notifications to users. Based on each user's availability, the project supports a streamlined process for customized swap and substitute requests. Allowing administrative COS lab TAs to efficiently create and modify work shifts on a visual calendar interface and post announcements for logistical purposes, COS Lab TA Planner integrates scheduling and notifications into one application. Following a structured software engineering process, COS Lab TA Planner has been developed into a rigorously-tested, rather than feature-based, user product to become a viable application for COS lab TAs.

1. Introduction

In Princeton University, the computer science department hires undergraduates, known as lab TAs, to assist other students taking introductory courses such as COS 109, 126, 217 and 226. With the coordination of the head lab TA and other administrative lab TAs, the lab TAs enroll in shifts throughout the semester. If not a substitute lab TA, a lab TA normally either has one or two shifts per week that last a total of four hours [1]. At the beginning of each semester, the lab TAs schedule permanent shifts for full-time lab TAs at the beginning of each semester via Google spreadsheet. Figure 1 illustrates the current system utilized by lab TAs to enroll in their preferred shifts. After receiving the availability of each lab TA, the head lab TA manually allocates the shifts to the undergraduate assistants.

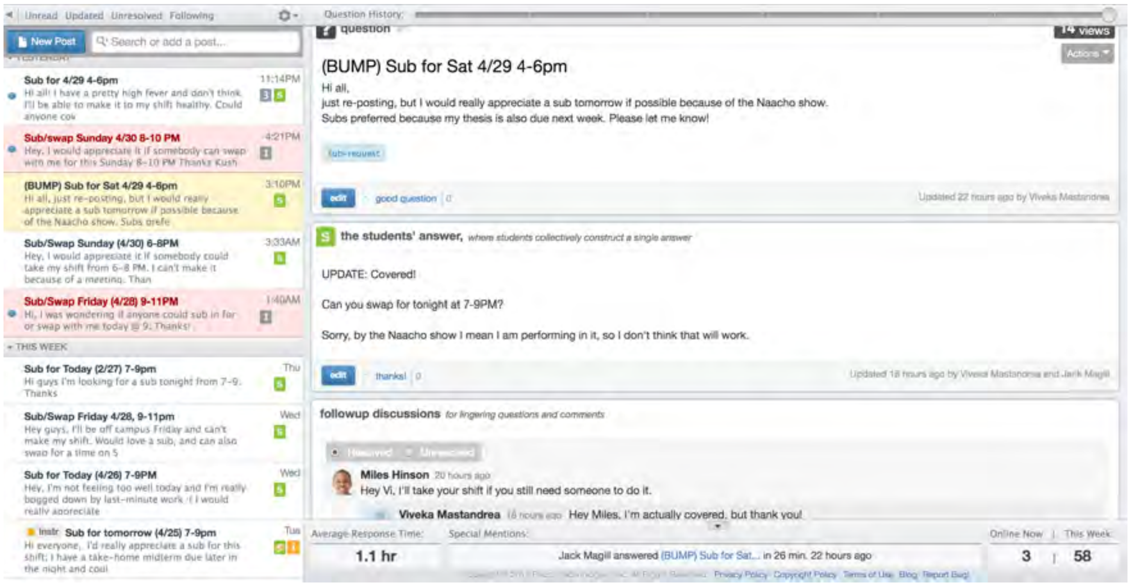


M 7-9	M 9-11	Tu 7-9	Tu 9-11	W 7-9	W 9-11	Th 7-9	Th 9-11	F 7-9	F 9-11	Sa 3-5	Sa 4-6	Sa 5-7	Su 5-7	Su 6-8	Su 7-9	Su 8-10	Su 9-11	Special requests
3	1	3	1	0	0	3	1	0	0	3	3	0	0	3	2	2	1	
1	1	1	1	0	0	0	0	3	3	0	0	0	0	0	3	3	3	Strongly prefer a straight 4-hour shift
0	0	1	1	3	3	0	2	3	3	0	0	0	3	0	3	3	3	
0	0	3	1	3	1	0	0	3	1	0	1	0	0	0	2	2	0	Prefer two 2-hr shifts
3	3	3	3	3	3	3	3	0	0	1	1	1	1	1	1	1	3	Prefer two 2-hr shifts
3	3	3	3	0	0	0	0	1	1	1	1	0	0	2	2	3	3	Strongly prefer a straight 4-hour shift
1	1	1	1	0	0	3	3	1	1	0	0	0	0	0	0	0	0	Strongly prefer a straight 4-hour shift
2	3	2	3	2	3	2	3	2	3	0	0	1	0	0	1	1	1	Prefer two 2-hr shifts
0	0	3	3	2	2	2	1	0	0	1	1	0	0	0	0	0	0	Strongly prefer a straight 4-hour shift
3	1	3	2	0	0	0	0	0	0	3	3	0	0	2	1	0	0	Prefer two 2-hr shifts
0	0	2	0	0	0	0	0	0	0	3	1	1	3	1	1	0	0	Prefer two 2-hr shifts
0	3	3	3	0	1	0	1	0	0	0	0	0	3	0	0	0	0	Prefer two 2-hr shifts
0	2	3	3	3	3	3	3	0	0	2	2	2	1	1	1	1	1	4 hour shift on Wed or Thurs would be great
0	2	0	2	2	2	3	3	2	2	1	0	0	1	2	0	0	0	
3	3	3	3	3	3	1	1	0	0	1	1	1	1	1	1	1	1	Strongly prefer a straight 4-hour shift
3	3	3	3	0	0	0	0	3	3	1	1	1	3	3	3	3	3	
3	3	3	3	0	0	0	0	3	1	2	2	1	2	2	2	2	2	Prefer two 2-hr shifts
0	0	2	2	3	2	1	1	0	0	3	2	1	1	1	1	0	0	Prefer two 2-hr shifts, but 4 on Wed is good too
3	3	3	3	2	2	0	0	0	0	0	0	0	0	0	0	1	1	
0	0	3	1	3	1	0	0	1	1	2	2	2	0	3	3	3	1	
2	2	0	0	2	1	0	0	3	2	2	3	3	3	3	3	1	1	Prefer two 2-hr shifts
1	1	0	0	2	1	3	2	3	2	3	3	3	0	0	0	0	0	Prefer two 2-hr shifts
0	0	0	0	1	1	3	3	1	1	0	0	0	0	0	0	0	0	Prefer 4 hour shift
1	0	1	1	0	3	0	1	1	0	3	3	1	0	0	0	0	0	Prefer two 2-hr shifts
3	3	3	3	3	3	0	0	0	0	3	3	3	0	0	0	0	1	Prefer two 2-hr shifts

Figure 1: Lab TA Shift Scheduling

During the semester, lab TAs send any type of notification generally through email or Piazza. More specifically, as lab TAs inevitably face external circumstances that conflict with their work shifts, the current system for substitute or swap requests is Piazza. When a lab TA posts a substitute or a swap request on Piazza, the entire group is notified either through Piazza or email alerts sent by Piazza. To respond to these requests, other students can compose a new followup discussion on Piazza (Figure 2). Consequently, based on the input of multiple lab TAs, requests for shift changes and announcements via Piazza and email are often ignored due to the overwhelming quantity of notifications (Figure 3).

Conducting a rigorous software engineering process, we performed an user requirement analysis by discussing with the head lab TA and three other lab TAs regarding the current scheduling and notification system (see section 1 of the Appendix to view user input from the head Lab TA). When gathering user input, we discovered the following: (1) lab TAs are advised to report any coding bugs they find on Piazza, but few do because bug posts are often unread. (2) The lab TAs emphasized the current inefficiency of requesting substitutes or swaps on Piazza. (3) Due to the difficulty of managing shifts and notifications with two separate platforms, the lab TAs highlighted the need of an announcements feature and scheduling tool on one application. (4) Because they often schedule extra shifts several weeks in advance throughout the semester, the lab TAs desired a calendar interface where they could schedule shifts directly on. (5) To account for busier weekends for assignments like N-body, the lab TAs wished for an availability interface where they could select which shifts they are available for potential swaps or extra help. (6) As the majority of all notifications are substitute and swap requests, they stressed the need for customized swap and substitute notifications based on their availabilities to reduce email and piazza clutter.



Sub for 4/29 4-6pm 11:14PM
Hi all! I have a pretty high fever and don't think I'll be able to make it to my shift healthy. Could anyone cov

Sub/swap Sunday 4/30 8-10 PM 4:21PM
Hey, I would appreciate it if somebody can swap with me for this Sunday 8-10 PM Thanks Kush

(BUMP) Sub for Sat 4/29 4-6pm 3:10PM
Hi all, just re-posting, but I would really appreciate a sub tomorrow if possible because of the Naacho show. Subs prefe

Sub/Swap Sunday (4/30) 6-8PM 3:33AM
Hey, I would appreciate it if somebody could take my shift from 6-8 PM. I can't make it because of a meeting. Than

Sub/Swap Friday (4/28) 9-11PM 1:40AM
Hi, I was wondering if anyone could sub in for or swap with me today @ 9. Thanks!

Sub for Today (2/27) 7-9pm Thu
Hi guys I'm looking for a sub tonight from 7-9. Thanks

Sub/Swap Friday 4/28, 9-11pm Wed
Hey guys, I'll be off campus Friday and can't make my shift. Would love a sub, and can also swap for a time on 5

Sub for Today (4/26) 7-9PM Wed
Hey, I'm not feeling too well today and I'm really bogged down by last-minute work :(I would really appreciate

Instr Sub for tomorrow (4/25) 7-9pm Tue
Hi everyone, I'd really appreciate a sub for this shift; I have a take-home midterm due later in the night and coul

(BUMP) Sub for Sat 4/29 4-6pm

Hi all,
just re-posting, but I would really appreciate a sub tomorrow if possible because of the Naacho show.
Subs preferred because my thesis is also due next week. Please let me know!

sub-request

Updated 22 hours ago by Viveka Mastandrea

the students' answer, *where students collectively construct a single answer*

UPDATE: Covered!

Can you swap for tonight at 7-9PM?

Sorry, by the Naacho show I mean I am performing in it, so I don't think that will work.

Updated 18 hours ago by Viveka Mastandrea and Jack Magill

followup discussions *for lingering questions and comments*

Resolved Unresolved

Miles Hinson 20 hours ago
Hey Vi, I'll take your shift if you still need someone to do it.

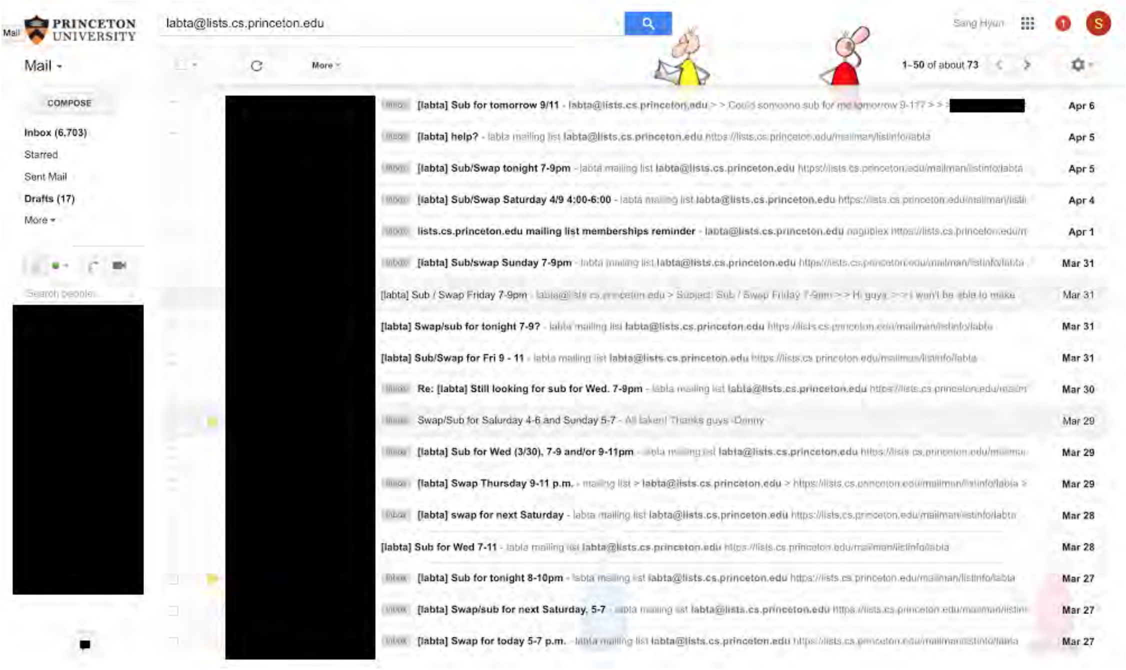
Viveka Mastandrea 18 hours ago Hey Miles, I'm actually covered, but thank you!

Average Response Time: 1.1 hr

Jack Magill answered (BUMP) Sub for Sat... in 26 min. 22 hours ago

Online Now 3 | This Week 58

Figure 2: Example of Sub/Swap Request



labta@lists.cs.princeton.edu

[redacted]

[redacted]

[labta] Sub for tomorrow 9/11 - labta@lists.cs.princeton.edu > > Could someone sub for me tomorrow 9-11? > > [redacted] Apr 6

[labta] help? - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Apr 5

[labta] Sub/Swap tonight 7-9pm - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Apr 5

[labta] Sub/Swap Saturday 4/9 4:00-6:00 - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listi Apr 4

lists.cs.princeton.edu mailing list memberships reminder - labta@lists.cs.princeton.edu nagubiex https://lists.cs.princeton.edu/n Apr 1

[labta] Sub/swap Sunday 7-9pm - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Mar 31

[labta] Sub / Swap Friday 7-9pm - labta@lists.cs.princeton.edu > Subject: Sub / Swap Friday 7-9pm > > Hi guys, > > I won't be able to make Mar 31

[labta] Swap/sub for tonight 7-9? - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Mar 31

[labta] Sub/Swap for Fri 9 - 11 - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Mar 31

Re: [labta] Still looking for sub for Wed. 7-9pm - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailm Mar 30

Swap/Sub for Saturday 4-6 and Sunday 5-7 - All taken! Thanks guys -Denny Mar 29

[labta] Sub for Wed (3/30), 7-9 and/or 9-11pm - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailma Mar 29

[labta] Swap Thursday 9-11 p.m. - mailing list > labta@lists.cs.princeton.edu > https://lists.cs.princeton.edu/mailman/listinfo/labta > Mar 29

[labta] swap for next Saturday - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Mar 28

[labta] Sub for Wed 7-11 - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Mar 28

[labta] Sub for tonight 8-10pm - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Mar 27

[labta] Swap/sub for next Saturday, 5-7 - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listin Mar 27

[labta] Swap for today 5-7 p.m. - labta mailing list labta@lists.cs.princeton.edu https://lists.cs.princeton.edu/mailman/listinfo/labta Mar 27

Figure 3: A Lab TA’s Email Inbox

Thus, the goal of this project is to develop a centralized platform on which COS lab TAs at Princeton University can manage their work shifts and notifications made by the group. Developing into a rigorously tested, rather than a feature based, user product, COS Lab TA Planner is a web application designed to unify customized shift management and announcements.

The remainder of this paper is organized as follows. Section 2 briefly discusses the previous iteration of COS Lab TA Planner. Section 3 describes COS Lab TA Planner's key differentiating features through use cases of the application, accompanied with screenshots. Section 4 details the implementation of the project as well as the methodology through which it was built. Section 5 evaluates the results of COS Lab TA Planner. Section 6 concludes by outlining future improvements and key learnings from developing COS Lab TA Planner.

2. Related Work

The previous iteration of COS Lab TA Planner was ShiftMap, a COS333 project. Designed as a general platform for Princeton's student employment groups, ShiftMap attempted to simplify the process of shift planning on Princeton campus. Served to be applicable for dining hall staff, McGraw tutors, COS lab TAs and other organizations, ShiftMap shares certain features with COS Lab TA Planner such as a calendar interface that displays user's shifts and a notification system. While ShiftMap is a functioning prototype, there are numerous implementation issues and lack of features, specific for lab TAs, that currently limit the potential for ShiftMap to be a viable application. Ultimately, ShiftMap failed to engage users in the development process, leading to it be an unused product. In section 3, we will discuss in detail the key differences between the functionality of ShiftMap and that of COS Lab TA Planner. Not only will this section establish clear boundaries from work that the prior COS333 team accomplish, it will also illustrate how COS Lab TA Planner builds upon ShiftMap.

3. Functionality

The previous section briefly described the previous iteration of COS Lab TA Planner that attempted to help student organizations manage their work shifts. By describing the primary use cases that a user may perform when using COS Lab TA Planner, this section highlights four key features in which COS Lab TA Planner differentiates itself and improves upon ShiftMap: (1) integration of an announcement feed, (2) customized swap and substitute requests, (3) creation of an availability interface, and (4) reimplementation of the calendar shift-scheduling interface. These use cases illustrate how lab TAs might take advantage of the core functionalities to alleviate the problems as previously described. Furthermore, these four core functionalities allow COS Lab TA Planner to be a one-stop, integrated platform for COS lab TAs. In each use case, this section provides references to accompanying screenshots located in section 2 of the Appendix.

3.1. Use Case 1: Head Lab TA Creates Group

In order to use COS Lab TA Planner, the user, specifically the head lab TA in these scenarios, must have a Princeton NetID. When viewing the log-in page (Figure 6), a user enters his typical log-in information via Princeton University's Central Authentication Service (CAS) as the user normally would in Blackboard. If the user has already logged onto CAS, the user is redirected to the home page of COS Lab TA Planner.

After securely logging on COS Lab TA Planner, the user is directed to a "New Employment Group" page (Figure 7). Additionally, the header of the website contains: "Home," "Help," "About Us," "No Groups," "Notifications" and the user's NetID. At any point, if the user wishes to logout, the user may hover his NetID. In the "Create New Employment Page," the user inputs the name and description of the group. The user needs to fill out the forms carefully as the web application does not allow editing them in the future. By submitting the filled-out employment page, the user will now automatically become an employer. Once creating the group, the user is directed to the homepage of COS Lab TA Planner, the calendar to manage work shifts (Figure 8).

From the user's perspective, ShiftMap and COS Lab Planner have no differences in functionality for this scenario.

3.2. Use Case 2: Homepage of the Head Lab TA

Now on the homepage of the newly created group, the user can view an empty weekly-based calendar with the "Delete Group" button and an "Update Group" button on the right-side. Although the calendar is initially empty, the user can toggle through the previous and future weeks as well as reverting back to the current week. The yellow tint on the calendar indicates the current day.

If the user wishes to access his group, the user can click on the group name in the header. If the user wishes to create another group besides presumably COS Lab TAs, the user can press "+ New Group." If the user has more than one group, the other groups appear as a dropdown.

As the user is the head lab TA, or the employer of the group, the user can press the "delete group" button to delete group. Deleting the group once clicking the button is an irreversible action.

From the user's perspective, ShiftMap and COS Lab Planner have no differences in functionality for this scenario.

3.3. Use Case 3: Adding and Removing Employers and Employees as Head Lab TA

As head lab TA or employer of the group, the user can add employees and other employers by clicking on the "update group" button. An "Update Employment Group" form will now appear with sections labeled: "New Employer," "New Employee," and "Shifts" (Figure 9). In the "New Employer" section, the user's NetID will automatically appear as the user is, by default, an employer when creating the group. If the head lab TA desires to add another lab TA with administrative privileges, the user would add that person as an employer. To add more employers, the user can enter another user's NetID and press enter (Figure 10). During this action, an alert will appear, notifying the user that he has successfully added the employer. The new user's NetID will now appear within the employers of that group.

To delete an employer, the user can press the "X" button besides the NetID and the user will delete that employer from the group, but that deleted user's NetID will now appear in the "Employees" section (Figure 11). During this action, an alert will appear, notifying the user that he has successfully deleted the other employer. Because a user might not wish to an employer, but still be part of the group as an employee, we designed making the user's NetID appear in the "Employees" section when removing the user from the "Employer's" section. Although the current user's NetID is displayed, the user cannot delete himself as an employer.

Likewise as adding an employer, in the "New Employee" section, the user can enter another user's NetID to add an employee. During this action, an alert will appear, notifying the user that he has successfully added the employee.

To remove an employee, the user can press the "X" button besides the NetID and user will delete that employee from the group. During this action, an alert will appear, notifying the user that he has successfully deleted the employee.

Unlike ShiftMap, COS Lab Planner offers UI feedback by notifying the user that he has successfully added or removed any employer and employee through an alert.

3.4. Use Case 4: Creating, Removing and Editing Shifts as Head Lab TA

As head lab TA or employer of the group, the user can create and delete shifts in two ways: (1) by accessing the "Update Group" button or (2) clicking directly on the calendar interface. In (1), the user will view a form to select a start-time, end-time, date, capacity, and repeat weekly option (not required). To create a shift, the user needs to fill out the entire form. To correctly select a start-time and end-time entry, the user needs to ensure the start-time is not before the end-time. When pressing the date entry, a min-calendar will appear for the user to select a date. The current day is highlighted in yellow-tint. To correctly fill out the capacity, the user needs to type in an integer number. After completing the forms, the user has the option to make the shift repeat every week, and then press "add shift" to create the shift. The shift will now be displayed on the calendar and on the form (Figure 12). An alert will appear, notifying the user has successfully added a shift. If the user incorrectly selects the start-time and end-time or incorrectly inputs the capacity, an error alert will appear (Figure 13).

To remove a shift in (1), the employer can press the "X" button to delete the shift permanently. If the shift repeats on all weekly basis, all of those shifts will be deleted. An alert will appear, notifying the user has successfully deleted a shift (Figure 14).

In (2), the user will view a form to input a start-time, end-time, capacity and to select a repeat weekly option (not required). By clicking on a day on the calendar, COS Lab TA Planner automatically knows the date of the shift. The user will need to fill out the entire form in order to create a shift. To fill out the start-time and end-time entry, the user needs to follow the example of "XX:XXam or pm" (Ex: 10:00am, 1:00pm). To correctly input a start-time and end-time entry, the user needs to ensure the start-time is not before the end-time. After completing the forms, the user has the option to make the shift repeat every week, and then press "add shift" to create the shift (Figure 15). The shift will now be displayed on the calendar. An alert will appear, notifying the user has successfully added a shift. If the user incorrectly inputs the forms, an error alert will appear (Figure 16).

To remove a shift in (2), the employer can click on the existing shift. On the appearing modal, the user can press the "Delete Shift" button to permanently delete the shift (Figure 17). An alert will appear, notifying the user has successfully deleted a shift.

If the head lab TA or an employer needs to adjust the capacity or time of an existing shift, the user can click on an existing shift to edit it (Figure 17). Similar to creating a shift in (2), the user needs to correctly input the start-time, end-time or capacity. The user does not need to modify all three forms. If the user successfully edits the shift, an alert will appear, notifying the user has successfully edited a shift (Figure 18).

Unlike ShiftMap, COS Lab Planner offers UI feedback by notifying the user that he has successfully created or removed any shift through an alert. Error alerts are also present whenever a user incorrectly creates a shift.

ShiftMap's implementation of inputting the start-time, end-time and the capacity is too inefficient as the employer has to manually enter the shift's pertinent info. By highlighting the current day when selecting a date and allowing the user to select as well as input a start-time or end-time, COS

Lab Planner's mechanism for adding shifts is more streamlined in (1).

In (2), COS Lab Planner provides a more intuitive layout to create shifts on the calendar itself. Since the calendar is one of the core components, the "Add Shift" feature in the calendar allows the user to visually interact and determine when to create a shift. Moreover, COS Lab Planner offers the functionality for a user to modify an existing shift. As our user requirement analysis indicates the head lab TA may need to update the capacity of an existing shift whenever assignments like N-Body are due, this feature fulfills a need that ShiftMap lacked.

3.5. Use Case 5: Setting Up The Availability Calendar as Head Lab TA

The availability calendar is designed for a user to select the times of the shifts that he is available to swap or substitute in for another lab TA. Additionally, the availability calendar is designed for the user to indicate the shifts he is willing to swap out of when completing a swap request. For the users in the group to select their personal availabilities, the head lab TA or any other employer must first create these "availability shifts" in the availability calendar. Underneath "Notifications" in the header, the head lab TA can click on "Availability" to access the availability calendar.

Upon viewing the empty availability calendar, the user cannot toggle through the previous and future weeks of the calendar because this interface represents all the permanent shifts that repeat weekly throughout the entire semester. To create an "availability" shift, the head lab TA can click on a day of the week. A modal will appear for the head lab TA to input the start-time and end-time of the permanent, weekly-occurring shift. (Figure 19). To fill out the start-time and end-time entry, the user needs to follow the example of "XX:XXam or pm" (Ex: 10:00am, 1:00pm). To correctly input a start-time and end-time entry, the user needs to ensure the start-time is not before the end-time. The shift will now be displayed on the calendar. An alert will appear, notifying the user has successfully created an availability shift. If the user incorrectly inputs the forms, an error alert will appear.

To delete an availability shift, the head lab TA can click on the existing shift. On the appearing modal, the user can press the "Delete Shift" button to permanently delete the shift (Figure 20). An alert will appear, notifying the user has successfully deleted an availability shift.

In addition to adding and deleting an availability shift, the user can edit an existing availability shift. If the head lab TA or an employer needs to adjust the capacity or time of an existing availability shift, the user can click on an existing shift to edit it. Similar to creating an availability shift, the user needs to correctly input the start-time or end-time. The user does not need to modify either time if he wishes. If the user successfully edits the availability shift, an alert will appear, notifying the user has successfully edited a shift.

The availability interface is completely new feature that ShiftMap lacked. Enabling the head lab TA to create availability shifts will allow all users to manage their personal availabilities for potential swap and substitute requests, which we will discuss in the next section.

3.6. Use Case 6: Selecting Availability in Availability Calendar for All Users

After the head lab TA initially sets up the availability calendar, any user, including the head lab TA and normal lab TA, may select the times of the permanent shifts that he is available to swap or substitute for another lab TA.

When viewing the availability calendar, the shifts created by the head lab TA will be labeled "unavailable" by default. To be available for potential substitute or swap requests the user can click on an availability shift and press "I am available" (Figure 21). Once completed, the availability shift will be updated to "available" and an alert will appear, notifying the user has successfully indicated that he is available for substitute and swap requests for that shift.

To indicate the user is no longer available for substitute or swap requests for that given shift, the user can press "I am not available" (Figure 22). An alert will appear to notify the user has successfully indicated he is no longer available for substitute and swap requests for that given shift.

Having users select their availabilities in the availability calendar allows COS Lab TA Planner to optimally perform filtered substitute and swap requests. Filtering requests based on each user's availability eliminates unnecessary notifications and smartly provides requests that the user is actually interested in. We will detail the mechanisms of substitute and swap requests in use cases 8 and 9.

3.7. Use Case 7: Using the Scheduling Calendar to Enroll and Drop from Shifts for All Users

After the head lab TA initially generates shifts on the scheduling calendar, any user, including the head lab TA and normal lab TA, may view shifts labeled as either "open" or "full" (Figure 23). If the shift is open, then the user has not yet enrolled in it. If the shift is full, then no capacity remains in the shift.

To enroll into an open shift, the user can click on the shift on the calendar and press enroll (Figure 24). Once enrolled, the shift will be updated to "enrolled" and an alert will appear, notifying the user has successfully enrolled into a shift. To drop from the enrolled shift, the user can click on the shift and press drop (Figure 25). An alert will appear to notify the user has successfully dropped from a shift.

COS Lab TA Planner's enrolling and dropping from shift functionality has been redesigned from ShiftMap's previous implementation. While ShiftMap displays a popover to enroll and drop from a shift, this design is limited if there are multiple users interacting with the calendar at the same time: whenever one user interacts with a popover, another user will have his popover automatically closed while attempting to enroll or drop from a shift. Thus, COS Lab Planner addresses this issue through the design of a modal, which smartly closes for one user after enrolling or dropping from a shift without conflicting with another user's modal. Furthermore, COS Lab TA Planner provides alerts to a user after enrolling or dropping from a shift.

3.8. Use Case 8: Completing Substitute Requests for All Users

In the middle of a semester, a user may need to perform a substitute request due to extenuating circumstances. While there are substitute lab TAs specifically hired to only fulfill substitute requests for full-time lab TAs, full-time lab TAs often complete each other's substitute requests. To request a substitute for one of his enrolled shifts, a user will click on his enrolled shift and press "Request sub" (Figure 26) The shift will now be labeled "Sub/Swap pending" and an alert will appear to notify the user has created a successful substitute request.

If a user decides that he no longer needs a substitute before another user accepts the substitute request, a user can cancel it in two ways: (1) by navigating to "Active Requests" within the header's "Notifications" or (2) clicking directly on the substitute pending shift. In (1), the user simply needs to press "Cancel Request" to cancel the sub (Figure 27). If there are no existing substitute requests, the page will simply read, "You currently have no outstanding requests" (Figure 28). In (2), the user can press "Cancel pending sub/swap" to cancel the substitute request (Figure 29). An alert will appear, notifying the user that the request has been successfully canceled.

Only users who have indicated that they are available for a substitute or swap request at the sender's time of the shift will receive the substitute request in the Notifications section of the header (Figure 30). If the user is also enrolled in the same shift of sender, he will not receive the request. Once accepting the substitute request for that shift, a notification will appear for both users and all employers (Figure 31). For the person who sent the substitute request, the shift will now be labelled as "Subbed/ Swapped In" (Figure 32). Similarly, for the person who accepted the substitute request, the shift will now be labeled as "Subbed/Swapped Out." Either party can cancel the accepted substitute request prior to one day of the shift by clicking on the shift in the calendar (Figure 33). Both users will receive a notification regarding the canceled substitute request (Figure 34).

ShiftMap increases the number of unnecessary substitute notifications by sending the substitute request to all users; however, COS Lab TA Planner smartly provides substitute notifications by filtering substitute requests based on each user's availability. Additionally, COS Lab TA Planner allows users to cancel the pending substitute request via the calendar interface, which ShiftMap lacks. Moreover, COS Lab TA Planner provides alerts to a user after creating or canceling the substitute request.

3.9. Use Case 9: Completing Swap Requests for All Users

In the middle of a semester, a user may need to perform a swap request due to extenuating circumstances. Unlike a substitute request, swap requests are designed for full-time lab TAs to exchange their shifts. To request a swap request, a user will click on his enrolled shift and press "Request swap." The shift will now be labeled "Sub/Swap pending" And an alert will appear to notify the user has created a successful swap request.

If a user decides that he no longer needs a swap before another user accepts the swap request, a user can cancel it in two ways: (1) by navigating to "Active Requests" within the header's "Notifications" or (2) clicking directly on the pending swap shift. In (1), the user simply needs to press "Cancel Request" to cancel the sub. If there are no existing swap requests, the page will simply read, "You currently have no outstanding requests." In (2), the user can press "Cancel pending sub/swap" to cancel the swap request. An alert will appear, notifying the user that the request has been successfully canceled.

The mechanism of a swap request is the following: 1) Only users who are available for a substitute or swap request for the time of the shift will receive the swap notification. 2) The users will see a notification that requests to exchange for one of their shifts (Figure 35). 3) Only the enrolled shifts of the user, who accepted the swap request, that match with the times of the sender's availability will appear for the user to select (Figure 36). If there are no shifts that match the availability of the sender, the user will see a page reading, "There are currently no available shifts for you to swap out" (Figure 37). At this point, a user can simply decline the request.

Once accepting the swap request for that shift, a notification will appear for both users and all employers (Figure 38). For the person who accepted the swap request, the shift will now be labeled as "Subbed/Swapped Out" (Figure 39). Similarly, for the person who sent the swap request, the shift will now be labelled as "Subbed/ Swapped In." Either party can cancel the accepted swap request prior to one day of the shift by clicking on the shift in the calendar. Both users will receive a notification regarding the canceled swap request.

The design of a swap request in COS Lab TA Planner is significantly different from that of ShiftMap. In ShiftMap, the mechanism of a swap request is one-to-one: a user can only choose another shift he would like to swap with when creating a swap request. However, practically, a user who creates a swap request should not be limited to choosing one shift to swap with. As long as user has indicated he is available for a swap at the time of that shift, the sender of the swap request should be content with whichever shift the user, who accepts the swap requests, selects. Furthermore, in

ShiftMap, a user could not cancel an accepted swap request. COS Lab TA Planner offers the feature of canceling an accepted swap request prior to one day of the shift. Thus, COS Lab TA Planner's swap request feature filters based on the availabilities of users and smartly provides shifts to swap out and into.

3.10. Use Case 10: Creating and Modifying Announcement for All Users

Throughout the semester, the head lab TA may post important logistical announcements such as "how to use the queue" or "how to enter hours." Additionally, other lab TAs may create announcements to share notable coding bugs they have discovered. Underneath "Notifications" in the header, any user can click on "Announcements" to create an announcement.

After pressing "New Announcement," the user will now view the title, type and description of a new announcement (Figure 40). An announcement is organized by (1) Logistic, (2) Bug and (3) Miscellaneous. Once the user creates an announcement, the announcement will appear for all users (Figure 41). Only the creator of the announcement can edit or delete it. When editing the announcement, the user can modify the title, type or description of it (Figure 42). Deleting the announcement is permanent.

The announcement feature is completely new to COS Lab TA Planner. The feature allows the head lab TA to post important administrative announcements to the group. Furthermore, other lab TAs can report coding bugs or important miscellaneous announcements to the group.

4. Implementation

This section describes the implementation of COS Lab TA Planner, including a discussion of its architecture and technologies used within its architecture.

The general architecture of COS Lab TA Planner consists of a back-end server, a database, and front-end UI. Figure 42 illustrates how each of these tiers interact with one another. The following subsections describe each of these tiers in greater detail.

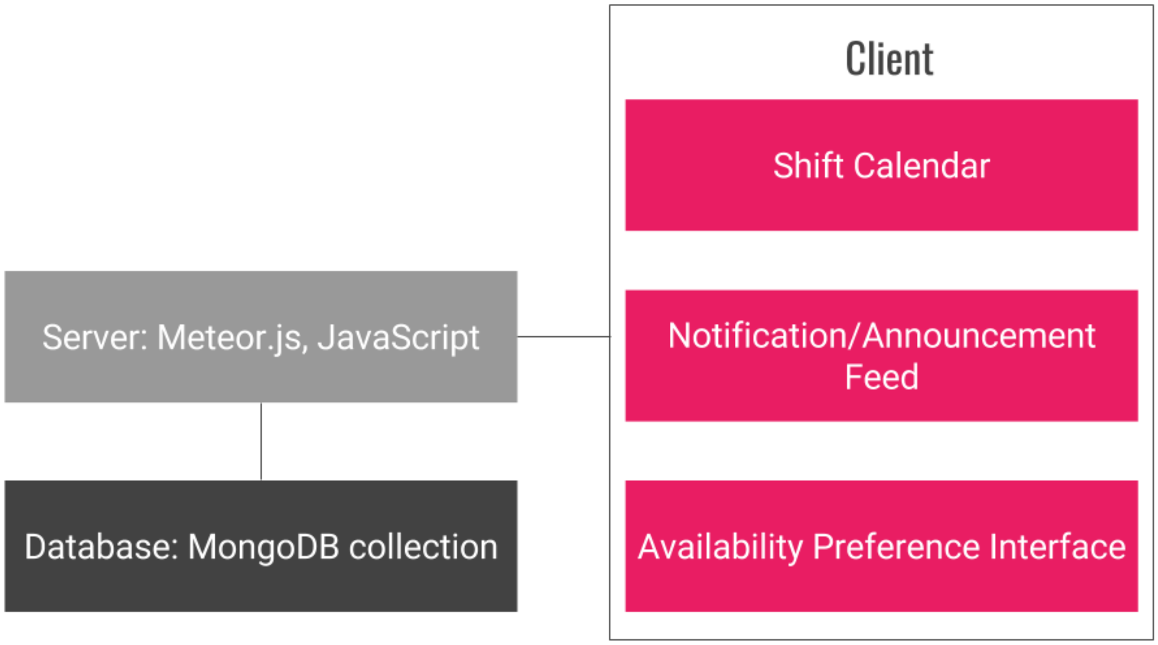

Client
Shift Calendar
Server: Meteor.js, JavaScript
Notification/Announcement Feed
Database: MongoDB collection
Availability Preference Interface

Figure 4: General Architecture

4.1. Meteor

Cos Lab TA Planner is a web-platform built upon Meteor (https://www.meteor.com/). Meteor, or MeteorJS, is a free and open-source Javascript framework written in Node.js. It facilitates the communication between the client and the server. Because of the interactive, backend-driven nature of the product, we decided that Meteor would simplify the process of both developing and maintaining the website. Furthermore, Meteor enables developers to use Javascript for the backend and frontend of the web application [2].

The core problem Meteor solves is a data synchronization problem. Because COS Lab TA Planner contains data on both client and server, Meteor synchronizes the data instantly between the two and also synchronize all clients with each other. Due to Meteor's handling of data, it should appear to all users that they are interacting directly with a single, shared store of data [2].

4.2. Client/Server Collections

COS Lab TA Planner uses Meteor's implementation of MongoDB collections to store information about Users, Groups, Shifts, Notifications, Available Shifts and Announcements. Collections are

Meteor's way of storing persistent data. Items inside collections are called documents. These server-side collections are defined in a schema format to constrain the contents of the collections to conform to a known format. This enables COS Lab TA Planner to confirm the structure of the data as it comes out of the database, instead of when it goes into the database. Thus, whenever incorrect data is detected, COS Lab TA Planner provides an error alert to the user. The client must be granted permission from the server-side to update, insert, and remove information from these collections. To make any changes to the collections from the client-side, the document must be referenced by its unique ID.

The structure of the collections are the following:

The Users collection contains the following attributes: (1) a unique ID that is automatically generated by MongoDB, (2) the NetID of the user, (3) the groups that the user is part of, (4) the shifts the user is enrolled in, (5) the shifts the user is available to swap or substitute into and (6) the ID of the most recently looked at group.

The Groups collection contains the following attributes: (1) a unique ID that is automatically generated by MongoDB, (2) the group's name, (3) the group's description, (4) the employers by their ID, and (5) the users (employees and employers).

The Shifts collection contains the following attributes: (1) a unique ID that is automatically generated by MongoDB, (2) date of the shift if the shift is one-time, (3) weekday of the shift if the shift repeats every week, (4) the start-time of the shift, (5) the end-time of the shift, (6) the users enrolled in the shift, (7) the group the shift exists in and (8) the swaps involving this specific shift. In (8), a swap contains its ID, the user who has swapped in, the user who has swapped out, and the date of the swapped shift.

The Notifications collection contains the following attributes: (1) a unique ID that is automatically generated by MongoDB, (2) the group the notification exists in, (3) the sender of the notification, (4) the ID of the shift, (5) the date of the shift, (6) the start-time of the shift, (7) the end-time of the shift, (8) the type of the notification (swap, sub, add, drop, swapcancel, or subcancel), (9) the users who have declined the request, and (10) the users who have clicked ok for the notification.

The Available Shifts collection contains the following attributes: (1) a unique ID that is automatically generated by MongoDB, (2) the weekday of the shift, (3) the start-time of the shift, (4) the end-time of the shift, (5) the users who are available in that shift, and (6) the group the available shifts exist in.

The Announcement collection contains the following attributes: (1) a unique ID that is automatically generated by MongoDB, (2) the title of the announcement, (3) the user who has created the announcement, (4) the group the announcement exists in, (5) the type of the announcement (logistic, bug or miscellaneous), (6) the description of the announcement, and (7) the date when the announcement is created.

Because all these collections are self-regulated by the activity of the users of COS Lab TA Planner, there is no reason to manually change them. However, the Notifications collection must be manually deleted after a certain amount of time. Because the head lab TA or another employer may choose to keep certain notifications for an extended period of time, this project does not remove any notifications. Additionally, if a new employer is added to the group, the new employer receives all the previous notifications. We suggest that the removal of any notifications be done after the end of each semester, as this is when employees and employers usually change for most on-campus employments.

4.3. Heroku and MongoDB

COS Lab TA Planner is currently hosted on Heroku at: http://labtaplanner.herokuapp.com. It uses MongoDB Sandbox for its backend to store the server-side collections described above.

4.4. Frontend

COS Lab TA Planner mainly uses Blaze, a Meteor library for creating user interfaces by writing reactive HTML templates. Using traditional templates and jQuery, Blaze automatically updates the Document Object Modal (DOM) at runtime [3]. As a result, COS Lab TA Planner has realtime updating with changes in the server-side collection. Additionally, COS Lab TA Planner incorporates various bootstrap elements and jQuery plugins for its frontend. We will now discuss COS Lab TA's core features.

4.5. Scheduling Calendar

For the scheduling calendar, COS Lab TA Planner uses a jQuery plugin called FullCalendar API. FullCalendar is a Javascript event calendar that displays events. More importantly, FullCalendar is a reactive calendar that supports realtime updating.

In ShiftMap's scheduling calendar, we did not fully support realtime updating for each user. Ideally, any changes to the calendar are reflected uniquely for each users. As a result, COS Lab TA Planner has reimplemented the scheduling calendar.

To ensure that our calendar data always updates, we add a method called *events* that will retrieve the shift information from the Shifts Collection. Then, the *events* function finds all the shifts associated with the specific group and checks the status of each shift with respect to the user. The status could be one of the following: open and not filled to capacity, open but filled to capacity, enrolled, waiting for substitute or swap, enrolled but swapped out and swapped in. More importantly, we utilize a Meteor library called Tracker, which is a simple dependency tracking system. Whenever our collection updates, the Tracker forces the calendar to repopulate itself.

To add a new shift or edit existing shifts, we use the *dayClick* and *eventClick* methods of FullCalendar. Previously, ShiftMap generated a popover to provide the users with appropriate options they can choose on a shift, but as previously described, ShiftMap fails to properly handle popovers for multiple users. As a result, we use Bootstrap's modal component to address ShiftMap's limited popover implementation. When creating the modal, we create a dual modal where a user can either add a new shift or edit an existing shift. We can combine these features into a single place that are toggled by reactive data sources like Session variables. Session variables store reactive data by using key and value pairs.

Initially, when editing an existing shift, the times and capacity of the shift was not properly displayed to the user via the modal. Although, these informations were properly stored in the Shift collection, we failed to pass the unique ID of the shift whenever the user opened the modal. To address this, we eventually used a Session variable to pass the ID to the modal. Additionally, COS Lab TA Planner properly closes the bootstrap modals by generating unique IDs for each modal, which is passed from the unique IDs of the shifts.

4.6. Availability Calendar

Similar to the scheduling calendar, COS Lab TA Planner uses FullCalendar API to create the availability calendar. A Bootstrap modal is fired whenever a user selects their availability for a permanent shift that an employer has created. When the user presses the "I am available" button, COS Lab TA Planner adds the user to the availability shift's array of users. Similarly, if the user presses the "I am not available" button, COS Lab TA Planner removes the user from the availability shift's array of users.

4.7. Swap/Substitute Request

When creating a swap or substitute request, COS Lab TA Planner requires the group ID, shift ID, date of the shift, start time of the shift, end time of the shift, and type of request. These values are stored within the Notification collection. Originally for a swap request, COS Lab TA Planner attempted to store all the IDs of the shifts the creator of the request could swap into, but this implementation was later changed because the user could potentially update their availabilities after creating the swap request. Rather than storing a copy of this information, examining the availabilities when a user accepts the swap request is more practical.

When filtering a swap or substitute request, COS Lab TA Planner only sends the request to users who have indicated that they are available at the time of the sender's shift. When creating the notification for a swap or substitute request, COS Lab TA Planner stores the ID of the user's scheduling shift. As a scheduling shift is independent from an availability shift, we match the start-time, end-time and date of the scheduling shift to those of the availability shift to determine if a user is available at the time of the sender's shift. More specifically, if the user exists among the availability shift's array of the users, the user will receive the request. Additionally, COS Lab

Planner ensures the user who receives the request is not enrolled in the same shift of the person who requested the swap or substitute.

When the user accepts a swap notification that requests to exchange for one of his requests, COS Lab TA Planner provides a list of shifts for the user to swap out of for the shift of the sender. COS Lab TA Planner first compares the times of the user's enrolled shifts and times of the sender's availability shifts. If there is a match, we store those shifts. However, COS Lab TA Planner also filters out shifts that the user may have already swapped out because we do not want to display a shift that a user is enrolled in, but already swapped out. To do so, we validate if a user is in the swaps array of a given shift in the Shifts collection. Afterwards, we display the list of shifts for the user to select. Currently, if any of those shifts are a weekly repeating shift, we assume the swap of the shift is on the same week as that of the sender's shift. If the time of the shift has already passed, COS Lab TA Planner moves the date of the shift to that of next week. If not, the date of the shift is on the same week as that of the sender's shift. This design implementation may be a limitation, as ideally, we should not limit the swap of both shifts to the same week.

When implementing the swap and substitute request, we initially could not properly compare the date of an availability shift and that of a scheduling shift. In COS Lab TA Planner, we have the date stored in a string format like so: *2017-03-15*. To convert the date to a more user readable format like *March 15, 2017*, we used a Javascript Date object; however, when using the value to create a new Date object, the date was consistently outputted one day off. Upon experimenting with the format of our string value, we discovered Javascript is inconsistent in converting a string to a Date object. For instance, a date like *2017-03-15* returns *March 14 2017*, but a date like 03-15-2017 returns *March 15 2017*.

Thus, Javascript's Date class does not represent a date, it represents a timestamp. To make it a date, it uses a time zone. It parses it with the GMT/UTC timezone (*March 15 2017, 00:00 UTC*) and then outputs it with a different timezone of four hours (*March 14 2017, 20:00 GMT-0400*). As a result, we had to normalize the date and eliminate the unwanted offset by specifying the correct time zone of Coordinated Universal Time *UTC*.

4.8. Announcements

COS Lab TA Planner uses AutoForm, a Meteor package that adds UI components and helpers to easily create basic forms with automatic insert and update events and automatic reactive validation, to implement announcements. Forming the Announcement collection as a schema is required to utilize AutoForm.

To create a new announcement, we utilize AutoForm's basic insert form. AutoForm's basic insert form is an autogenerated form that uses Bootstrap3 classes and contains the appropriate HTML5 fields for all keys in the Announcements collection. The basic insert form forms validations based on the Announcements collection. If any input is invalid, the form is continually re-validated as the user fixes the inputs.

AutoForm's basic update form is similar to the insert form, except we pass the unique ID of the announcement to update it.

5. Evaluation

To measure COS Lab TA Planner's potential viability, we evaluated the system by inviting current lab TAs to perform the use cases previously described in the Functionality section. In the first round of evaluation, a lab TA, who previously used ShiftMap, completed the scenarios along with the author. In the second round of evaluation, the head lab TA and another lab TA performed the scenarios together. For the purpose of gaining as much feedback as possible, all users performed tasks as an employer. In both evaluations, we used the usability testing evaluation technique.

Usability testing determines whether COS Lab TA Planner "is usable by the intended user population to achieve the tasks for which it was designed" [4]. The process involves observing and questioning the user as he attempts to perform the tasks. While the users were performing the tasks, they were asked to verbally articulate their initial reactions and opinions regarding the system. As the users performed each task, we recorded their actions and feedback. Following each task, we asked the user if they clearly understood how to complete the task by using the application. At the end of the evaluation, we asked the user to share any other comments and suggestions for the application.

When evaluating the application with the three lab TAs according to the usability testing technique, we discovered the following key insights for each assigned task.

5.1. Creating the Group

All users thought creating a group was intuitive. They had no suggestions to improve this functionality.

5.2. Adding and Removing Employers and Employees

To add an employer or a employee, a user needs to input the employer's or employee's NetID. However, two users added an employee or employee by the person's name rather than NetID. One user thought adding an employee or employer was not fully intuitive. Currently, the instruction to add an employer is "Type to add new employer." Similarly, the instruction to add an employee is "Type to add new employee." The users suggested modifying the instructions to read "Type to add new employer (or employee) and press enter." The user, who previously used ShiftMap, enjoyed the alerts appearing after adding or removing an employer or employee.

5.3. Creating, Removing and Editing Shifts

When creating a shift on the calendar or in "Update Group," each user had no difficulty inputting the required forms. All users commented on the intuitiveness of creating and viewing shifts on a calendar, as opposed to a spreadsheet. However, one user suggested when creating a shift on the calendar, the modal should display the date. Even though the user knew which day they clicked on the calendar, the user suggested displaying the date regardless. Another user suggested modifying the label of "Repeat Weekly" to "Repeat Every Week." Rather than inputting *7:00pm* for a start time or end time, one user suggested modifying the input to *7*. Because no shifts occur before 3:00pm, the user believed abbreviating the input to *8* rather than *8:00pm* would make the mechanism of adding or editing shifts quicker. Also, because no shift starts at a 30 minute increment, one user suggested removing all half-hour starting shifts in "Update Group." Finally, all users suggested truncating the calendar’s appearance from 3:00pm to 11:00pm.

5.4. Setting Up The Availability Calendar

While understanding the availability calendar’s function to filter swap or substitute requests, two users were confused why the shifts created from the scheduling calendar did not appear in the availability calendar. They did not expect the shifts in the scheduling calendar to be separate from those in the availability calendar. Furthermore, they expected the ability to toggle through the previous and future weeks as well as reverting back to the current week. Interestingly, while working by themselves in the second evaluation, the two users misinterpreted the availability calendar’s functionality. Rather than viewing the availability calendar as a fixed week for all permanent shifts (since all shifts repeat on a weekly basis), they attempted to create availability shifts for future weeks. When creating an availability shift, one user suggested changing the title of the modal to "Add a Time When I am available" rather than "Adding a Shift." Finally, the users discovered a minor frontend bug: when hot reloading the page, the availability shifts would not appear until the users explicitly clicked on "Availability" in the header.

5.5. Selecting Availability in Availability Calendar

Initially, the head lab TA intuited that selecting "I am not Available" would prohibit all users from requesting substitute or swap requests. After explaining the functionality, the users did not face complications when selecting their availability. The first suggestion they had was changing the labels on the availability calendar from "Unavailable" to "I am currently not Available" and "Available" to "I am currently Available." The second suggestion was providing a video tutorial to the user when he or she first uses the availability calendar.

5.6. Enrolling and Dropping Shifts in Scheduling Calendar

One of the users found a minor frontend bug when using the modal to enroll or drop from shifts in the scheduling calendar: After going to another page and coming back to the scheduling calendar, a modal's enroll and drop buttons initially disappear until a user clicks on another shift. After clicking on another shift, the two buttons function as normal. Besides this issue, the users had no difficulty enrolling and dropping from their shifts.

5.7. Completing Substitute or Swap Requests

While substitute or swap requests functioned properly for all three users, the head lab TA proposed changing the mechanism of canceling a substitute or swap request: if a user accepts a substitute or swap request, the user should not be able to cancel it. If the user who accepts the substitute or swap request cannot make the shift, he should now send a substitute or swap request. Additionally, the users suggested listing the substitute and swap requests at the top of an employer's notification feed. Currently, the substitute and swap requests are viewed below the notifications of users enrolling or dropping from shifts. Interestingly, one user believed mutual acceptance is needed for a swap request. When a lab TA accepts a swap request, the request is completed. However, one user believed the person who sends the request should complete the swap request by confirming the swapped shifts of both users. Finally, a user discovered COS Lab TA Planner does not display the time for the swapped out shift of the user, who accepts the swap request, after a swap request is completed.

5.8. Creating and Modifying Announcement

All three users properly created and modified an announcement. One user attempted to pin an announcement similar to Piazza's implementation of posts, but was confused when she could not find the functionality. Even though we told the users that the number of announcements would be limited due to the swap and substitute requests being in another section, all users suggested the functionality to pin an announcement. Additionally, a user attempted to edit an announcement even though he was not the creator of the announcement. Because he was an employer, the user expected to modify the announcement. He suggested allowing the employers to delete or modify all existing announcements. Finally, the head lab TA suggested a functionality for people to post new threads on an existing announcement. On Piazza, people currently "bump" their substitute or swap requests on existing posts, which is especially helpful when people are usually too busy to pay attention to notifications the first time they receive them.

6. Conclusion

COS Lab TA Planner is designed to serve as a centralized platform for Cos lab TAs to manage their work shifts and notifications. The feedback about COS Lab TA Planner's improved features are overall positive. The core functionalities are all implemented into the system with minor frontend bugs, but further refinement and polishing will turn COS Lab TA Planner into a deployable project. The head lab TA suggested the following improvements: (1) Creating a web-interactive tutorial as lab TAs would need to transition to a new system. (2) Linking COS Lab TA Planner to Princeton's lab queue website to integrate both applications. (3) Creating a monitoring feature to prevent lab TAs from skipping their shifts without informing the head lab TA. Overall, the head lab TA enjoyed the system as notifications are reduced, shifts can be viewed on an intuitive calendar and customized substitute and swap requests are provided. Ultimately, it needs to incorporate the iterative updates that the evaluators have stated, but after incorporating these features and performing a more thorough evaluation in the future, COS Lab TA can potentially adopt the project.

By following a rigorous software engineering process composed of user requirement analysis and user-involved evaluation, we have developed a rigorously tested, rather than feature based, user-driven product. Developing COS Lab TA Planner has illustrated the importance of conducting a structured engineering and design process.

7. Acknowledgement

I would like to thank my advisor, Dr. Dondero, for providing guidance and feedback throughout this entire semester. I would also like to thank Sally Jiao, the head lab TA, for her cooperation and participation in the testing session. Finally, I would like to thank Danny Kim, a lab TA and former ShiftMap developer, for his helpful advice and encouragement to complete this project.

8. Honor Code

This paper represents my own work in accordance with University regulations.

X Jonathan Yang

References

[1] Princeton University, "Undergraduate Grader/Lab TA Application" [Online]. Available: http://tas.cs.princeton.edu/apply.php

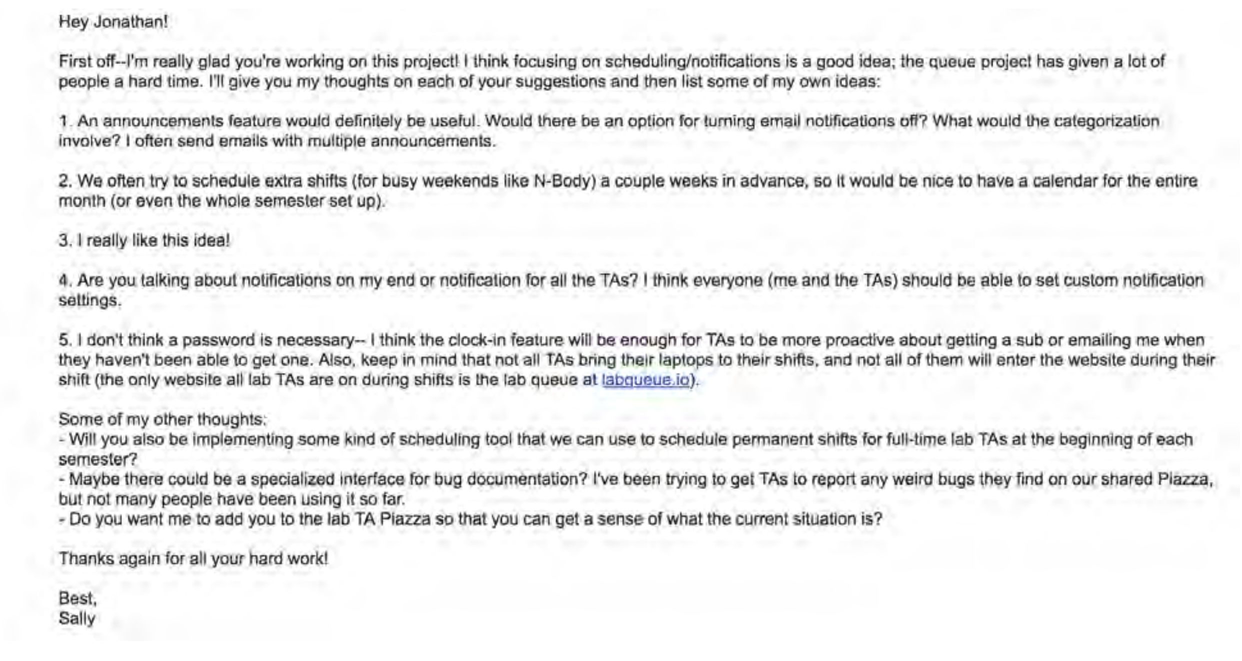
[2] Meteor, "Meteor Developers" [Online]. Available: https://guide.meteor.com/

[3] Blaze, "Blaze Developers" [Online]. Available: http://blazejs.org/

[4] Y. Rogers, H. Sharp, and J. Preece, *Interaction Design: Beyond Human-Computer Interaction*, 3rd ed. John Wiley and Sons, 2011, ch. 14.

9. Appendix

9.1. User Input from Head Lab TA



Hey Jonathan!

First off--I'm really glad you're working on this project! I think focusing on scheduling/notifications is a good idea; the queue project has given a lot of people a hard time. I'll give you my thoughts on each of your suggestions and then list some of my own ideas:

1. An announcements feature would definitely be useful. Would there be an option for turning email notifications off? What would the categorization involve? I often send emails with multiple announcements.

2. We often try to schedule extra shifts (for busy weekends like N-Body) a couple weeks in advance, so it would be nice to have a calendar for the entire month (or even the whole semester set up).

3. I really like this idea!

4. Are you talking about notifications on my end or notification for all the TAs? I think everyone (me and the TAs) should be able to set custom notification settings.

5. I don't think a password is necessary-- I think the clock-in feature will be enough for TAs to be more proactive about getting a sub or emailing me when they haven't been able to get one. Also, keep in mind that not all TAs bring their laptops to their shifts, and not all of them will enter the website during their shift (the only website all lab TAs are on during shifts is the lab queue at labqueue.io).

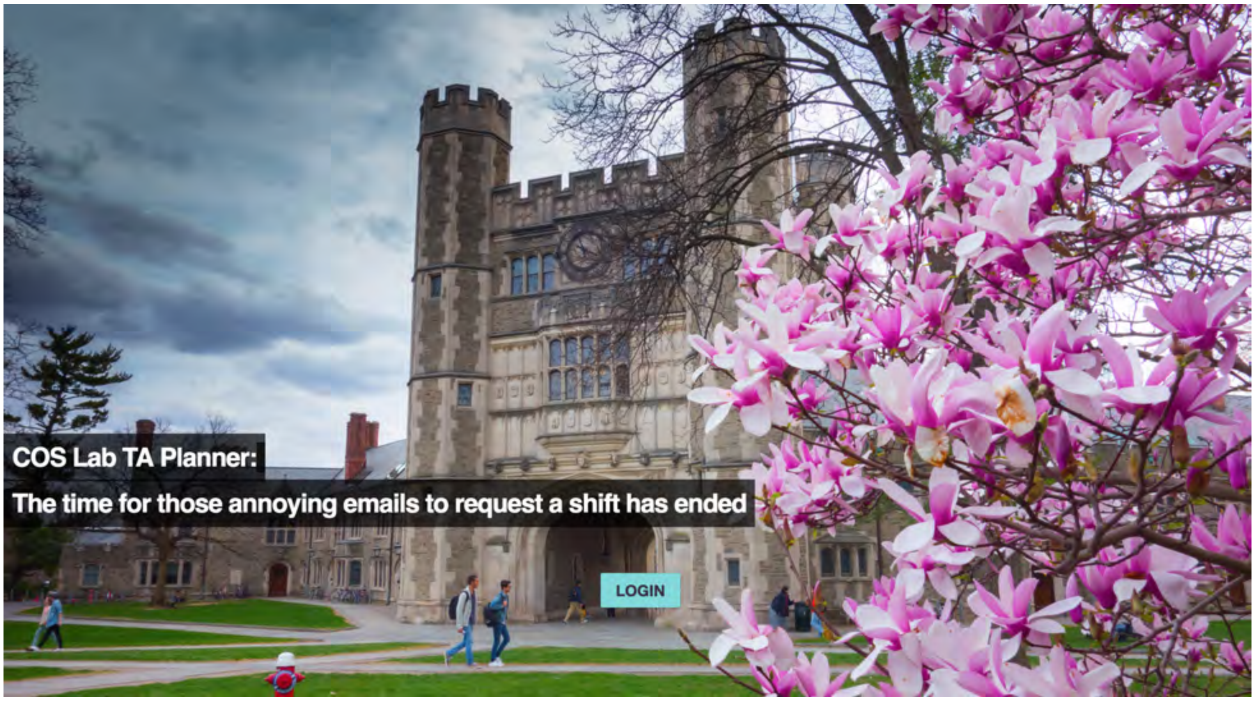
Some of my other thoughts:
- Will you also be implementing some kind of scheduling tool that we can use to schedule permanent shifts for full-time lab TAs at the beginning of each semester?
- Maybe there could be a specialized interface for bug documentation? I've been trying to get TAs to report any weird bugs they find on our shared Piazza, but not many people have been using it so far.
- Do you want me to add you to the lab TA Piazza so that you can get a sense of what the current situation is?

Thanks again for all your hard work!

Best,
Sally

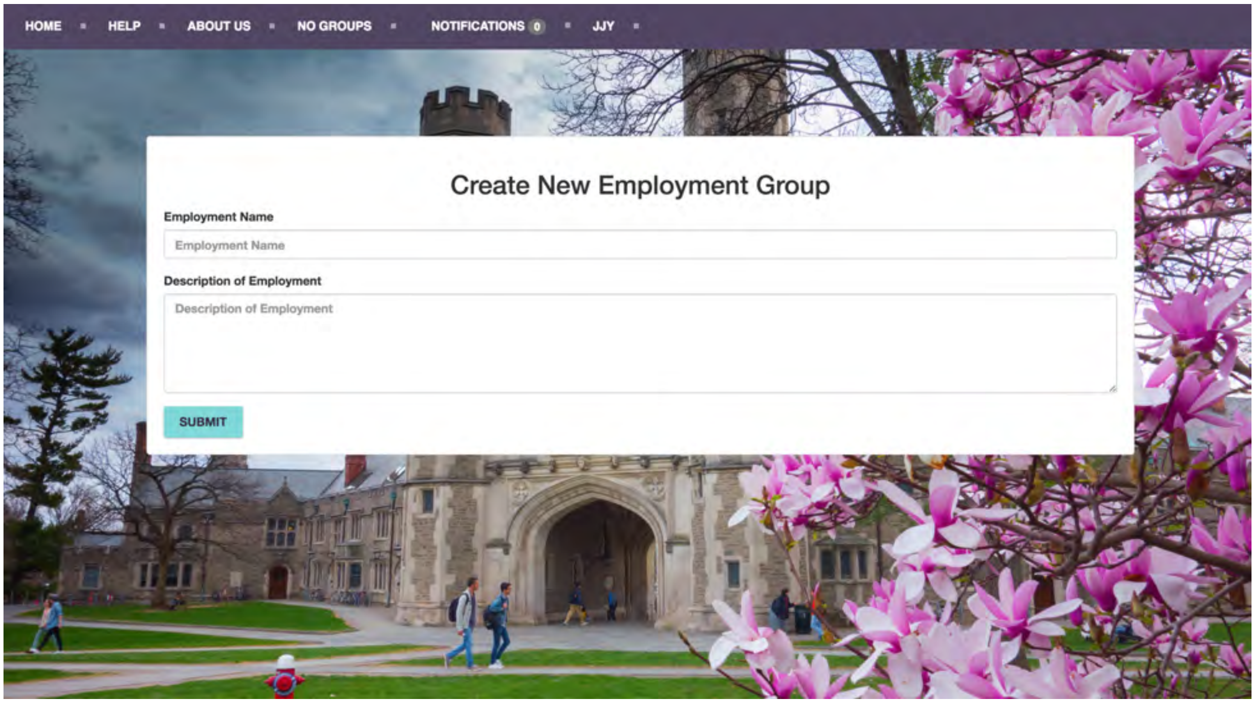
Figure 5: User Input from Head Lab TA

9.2. Screenshots of COS Lab TA Planner


COS Lab TA Planner:

The time for those annoying emails to request a shift has ended

Figure 6: Log-In Page



Create New Employment Group

Employment Name

Employment Name

Description of Employment

Description of Employment

Figure 7: Creating New Group

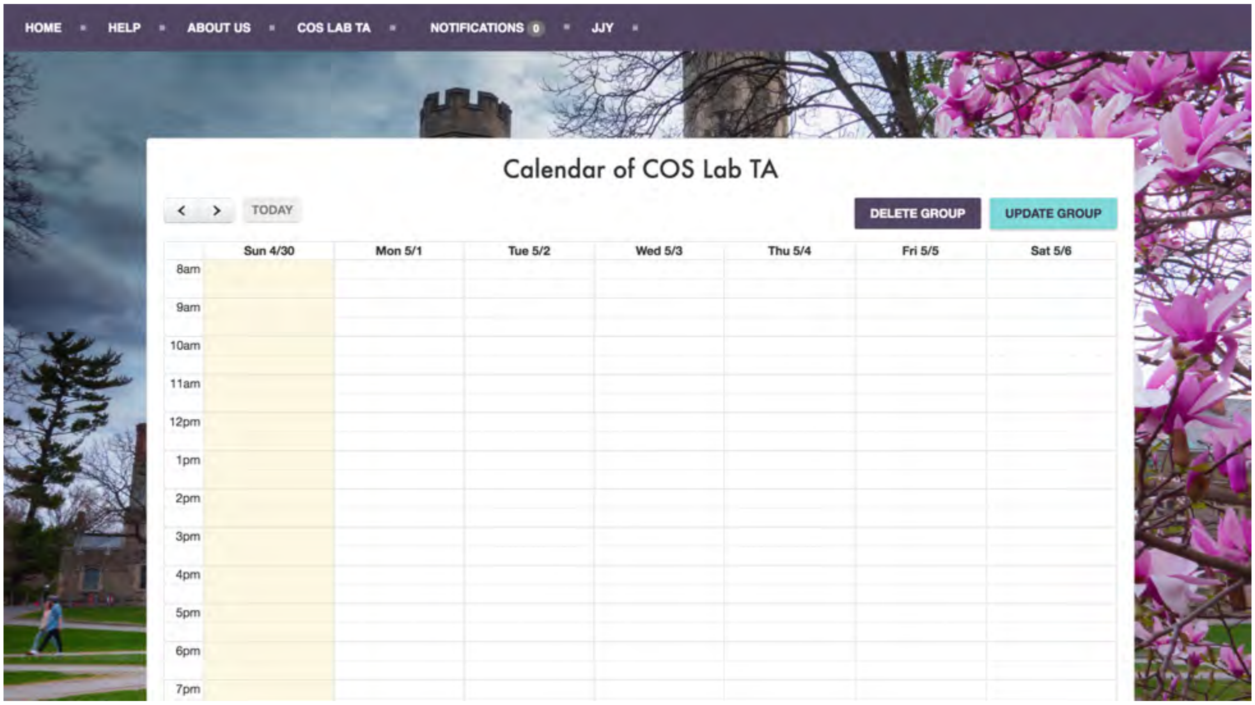


Figure 8: Homepage

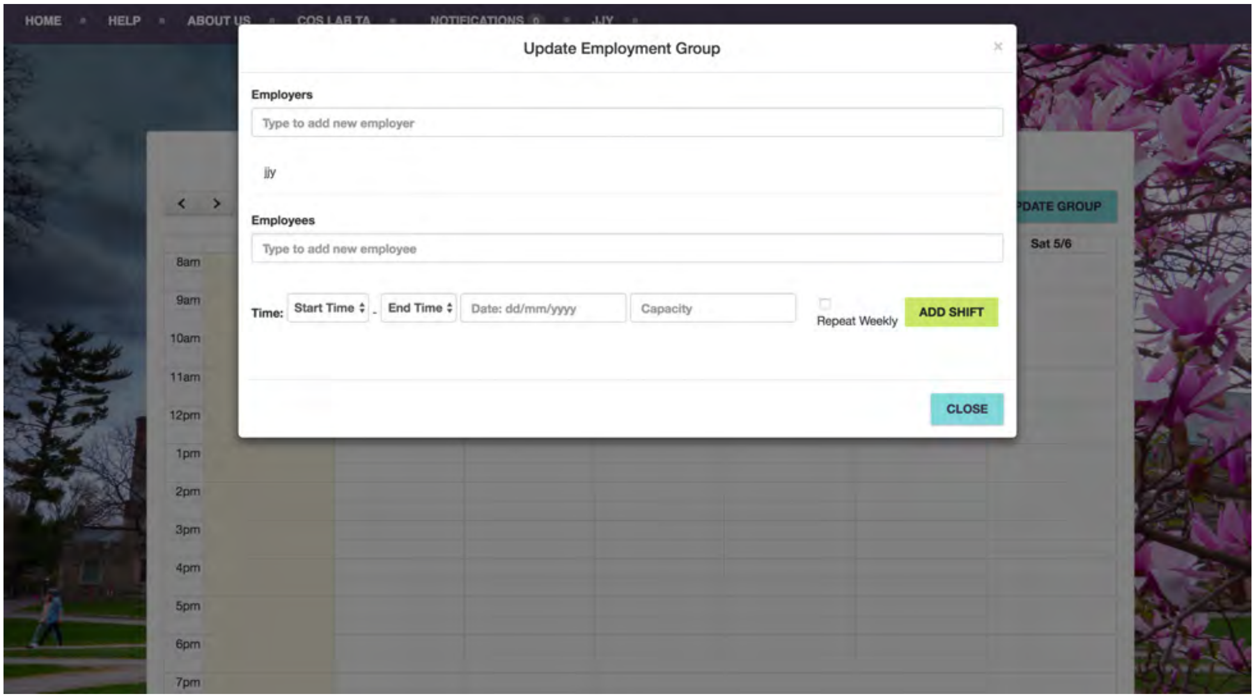
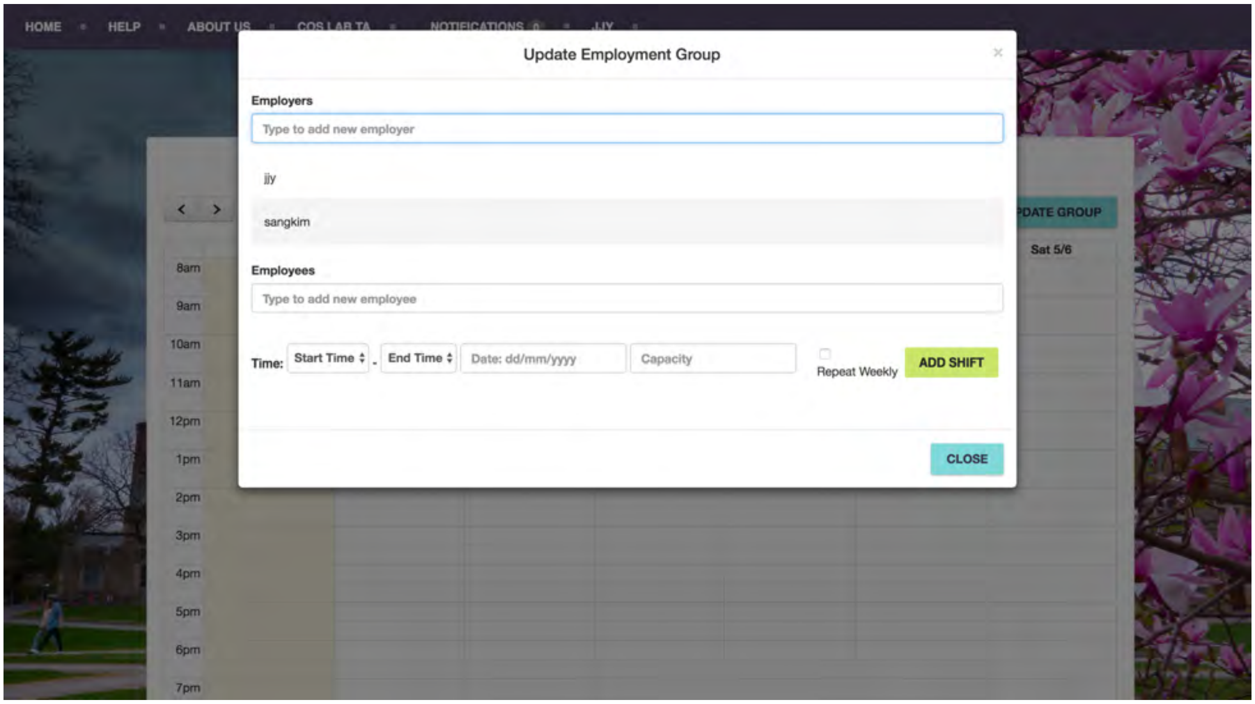


Figure 9: New Employment Group



Update Employment Group

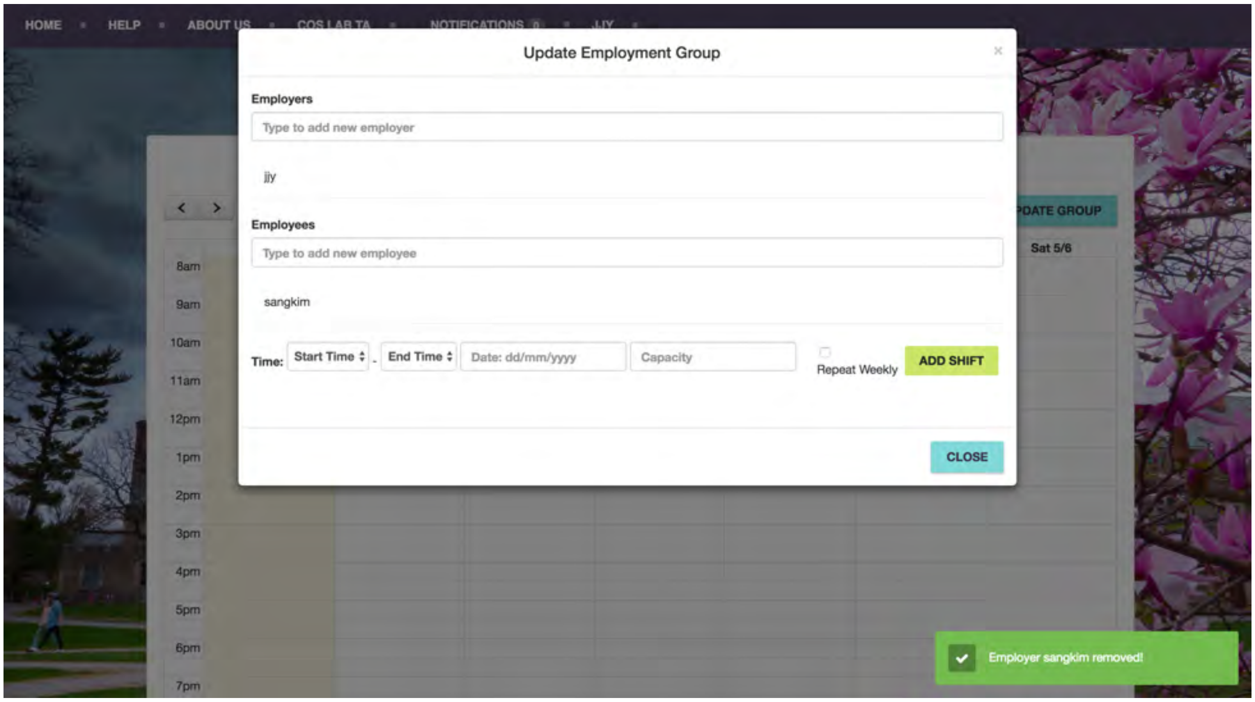
Employers: Type to add new employer

jiy

sangkim

Employees: Type to add new employee

Figure 10: Adding Employer



Update Employment Group

Employers: Type to add new employer

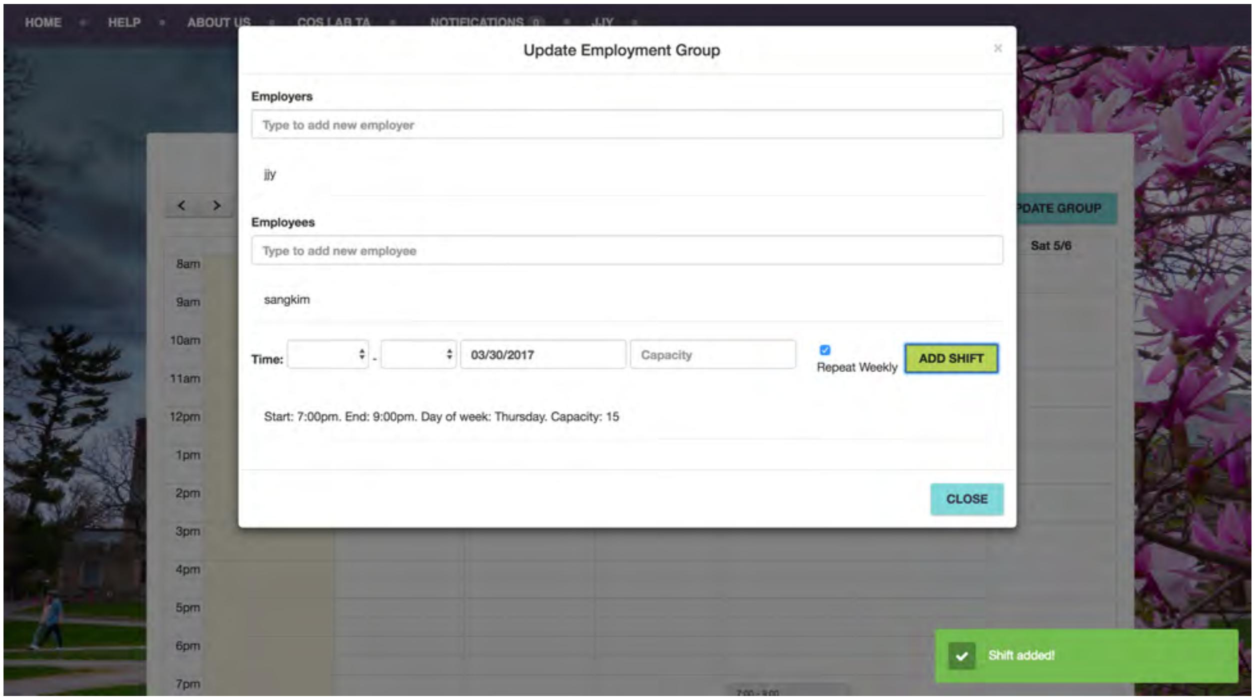
jiy

Employees: Type to add new employee

sangkim

✓ Employer sangkim removed!

Figure 11: Removing Employer



Update Employment Group

Employers

Type to add new employer

jjy

Employees

Type to add new employee

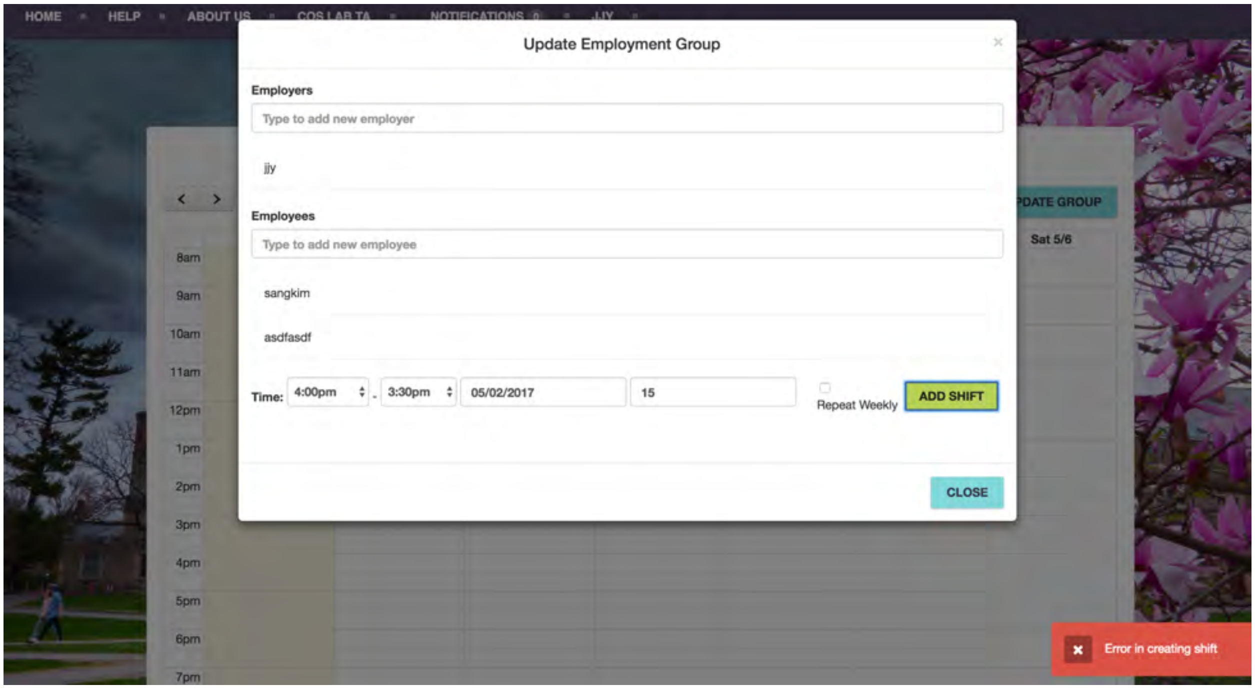
sangkim

Time: - 03/30/2017 Capacity ☑ Repeat Weekly

Start: 7:00pm. End: 9:00pm. Day of week: Thursday. Capacity: 15

✓ Shift added!

Figure 12: Correctly Creating Shift



Update Employment Group

Employers

Type to add new employer

jjy

Employees

Type to add new employee

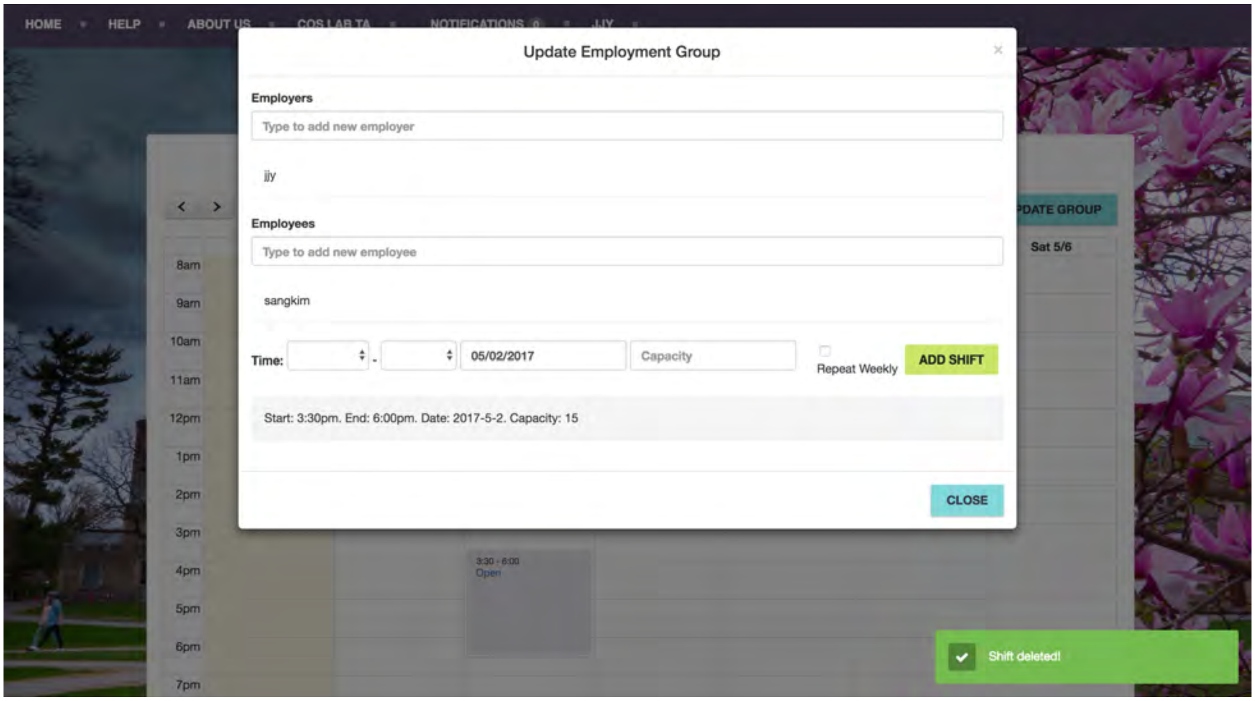
sangkim

asdfasdf

Time: 4:00pm - 3:30pm 05/02/2017 15 □ Repeat Weekly

✖ Error in creating shift

Figure 13: Incorrectly Creating Shift



Update Employment Group

Employers

Type to add new employer

jjy

Employees

Type to add new employee

sangkim

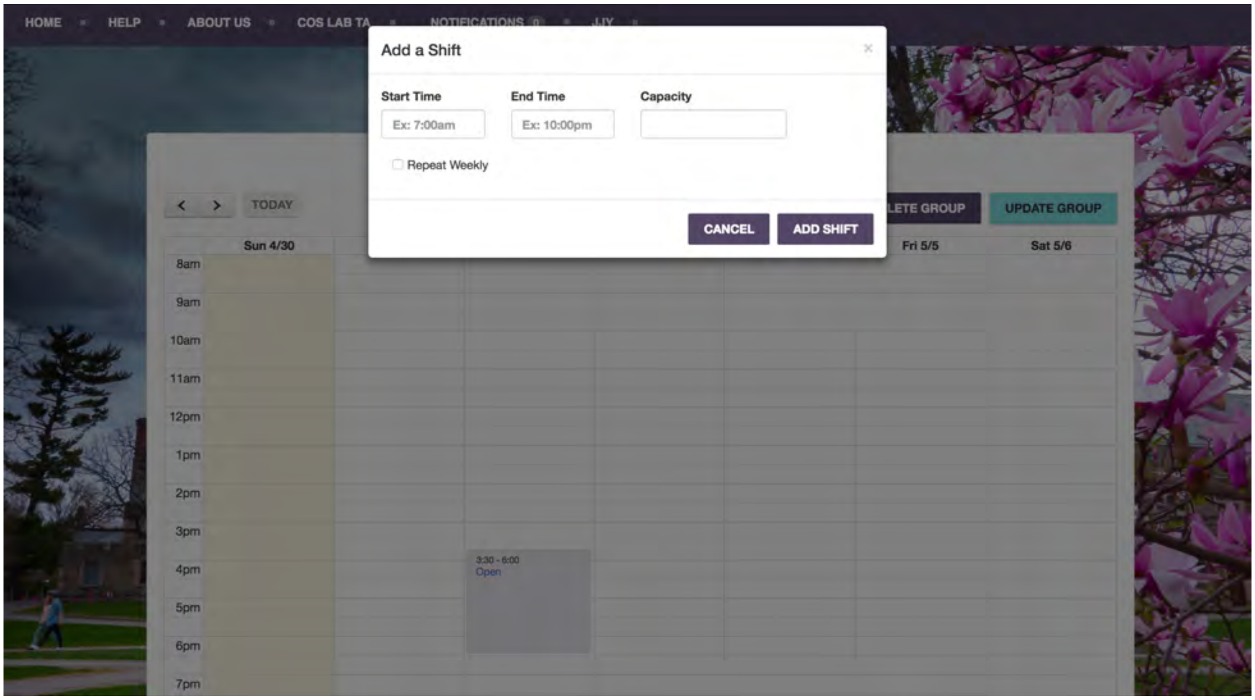
Time: 05/02/2017 Capacity □ Repeat Weekly ADD SHIFT

Start: 3:30pm. End: 6:00pm. Date: 2017-5-2. Capacity: 15

CLOSE

✓ Shift deleted!

Figure 14: Deleting Shift



Add a Shift

Start Time: Ex: 7:00am

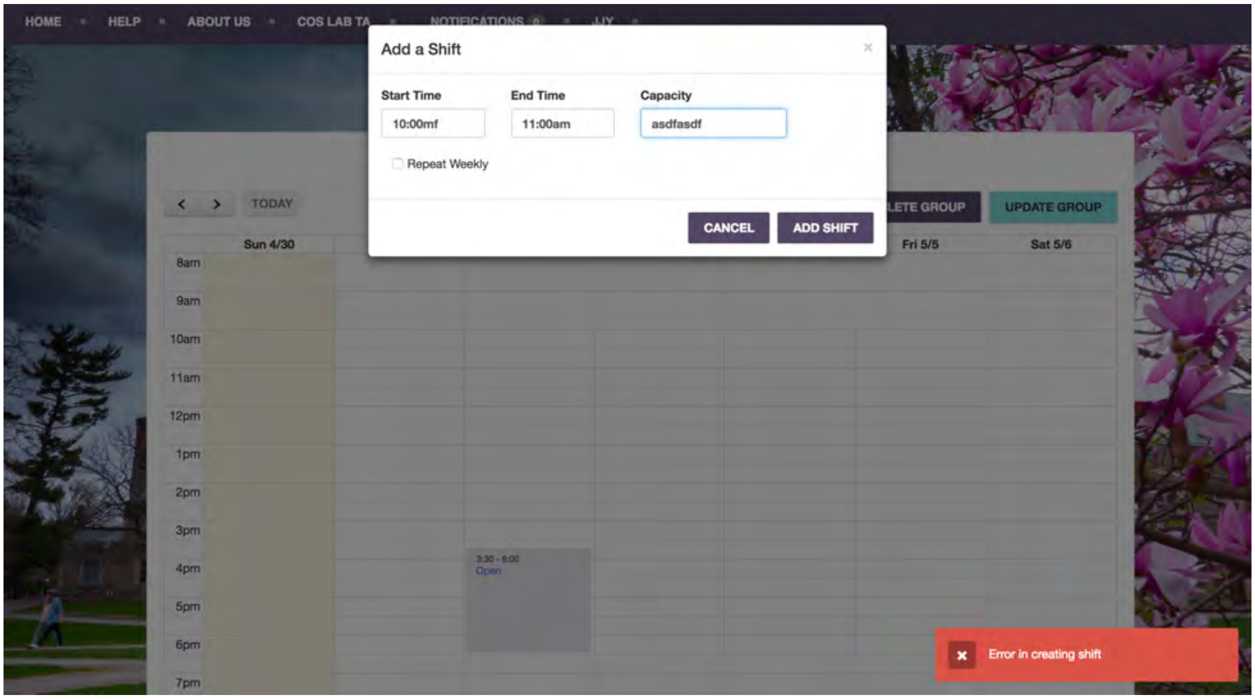
End Time: Ex: 10:00pm

Capacity

□ Repeat Weekly

CANCEL ADD SHIFT

Figure 15: Adding Shift on Scheduling Calendar



Add a Shift

Start Time: 10:00mf

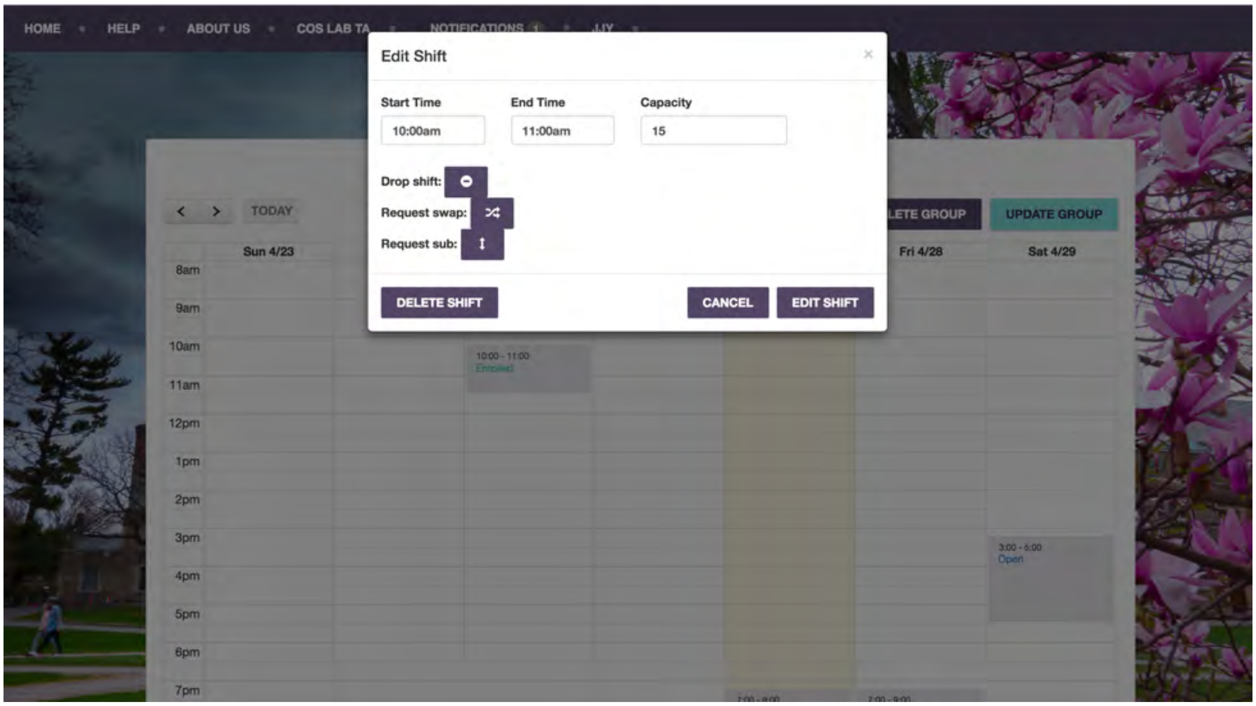
End Time: 11:00am

Capacity: asdfasdf

□ Repeat Weekly

Error in creating shift

Figure 16: Incorrectly Adding Shift on Scheduling Calendar



Edit Shift

Start Time: 10:00am

End Time: 11:00am

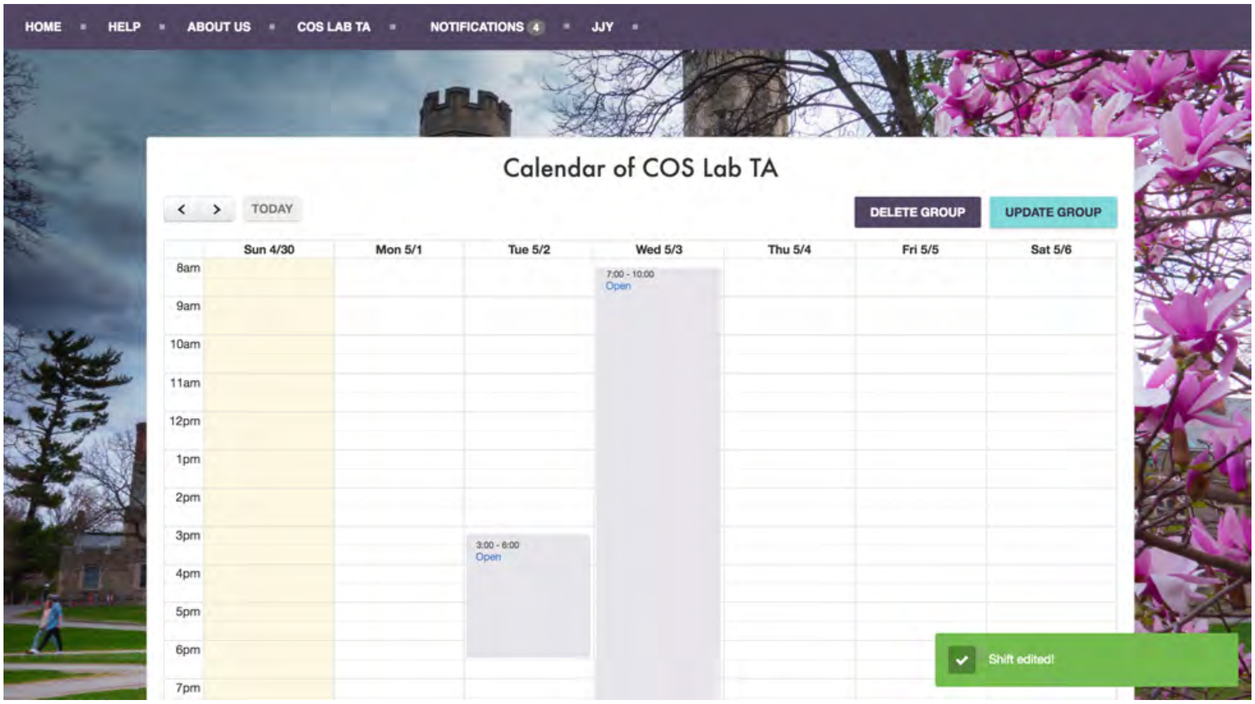
Capacity: 15

Drop shift:

Request swap:

Request sub:

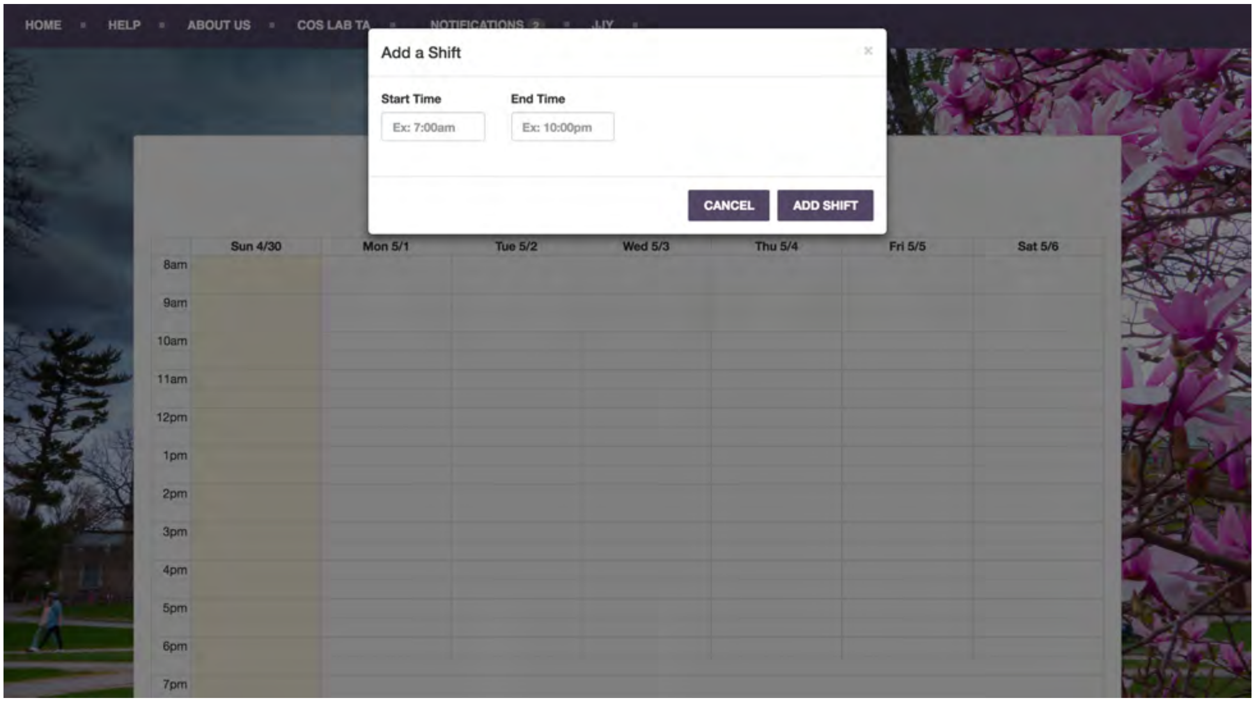
Figure 17: Editing Shift on Scheduling Calendar



Calendar of COS Lab TA

Shift edited!

Figure 18: Alert after Editing Shift on Scheduling Calendar

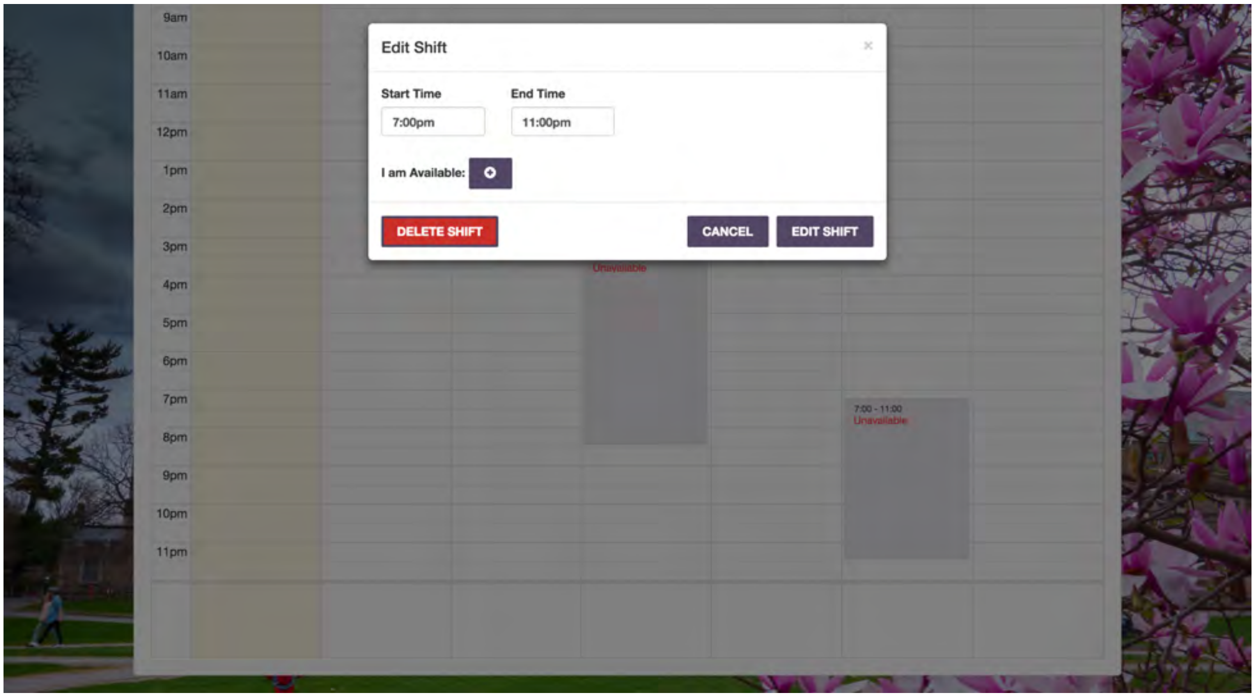


Add a Shift

Start Time: Ex: 7:00am

End Time: Ex: 10:00pm

Figure 19: Creating Shift on Availability Calendar



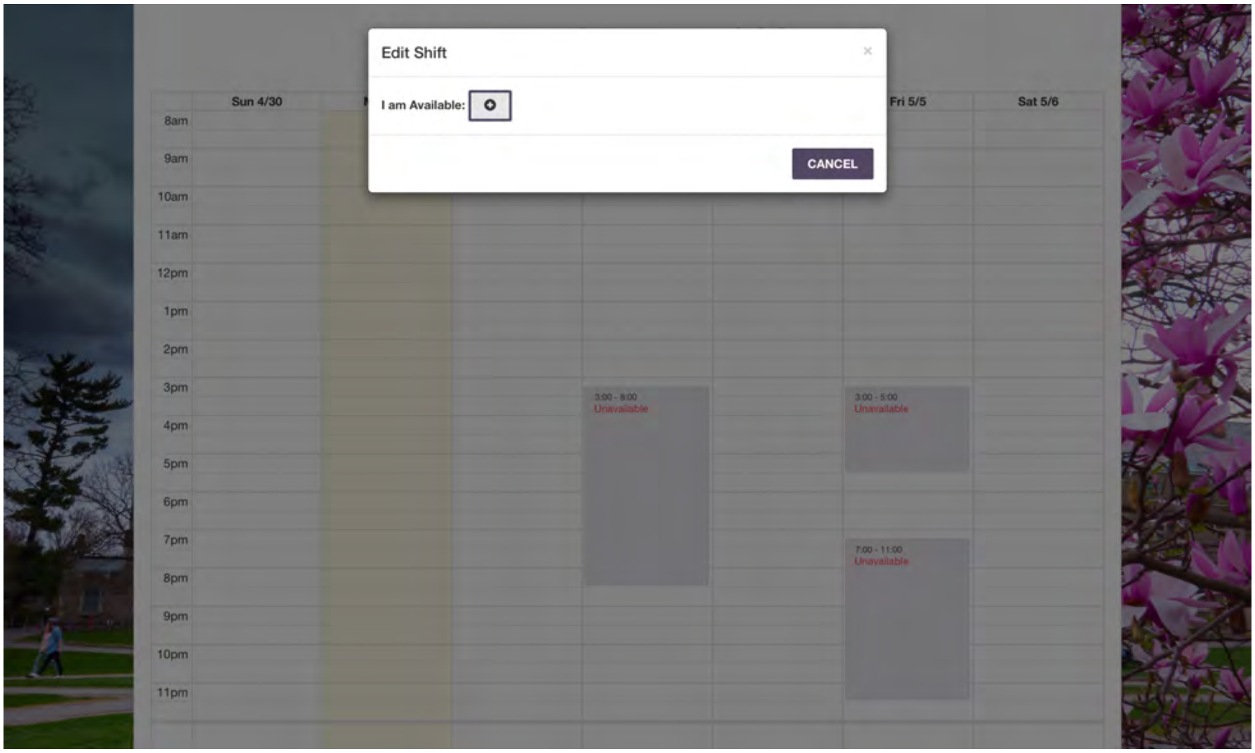
Edit Shift

Start Time: 7:00pm

End Time: 11:00pm

I am Available:

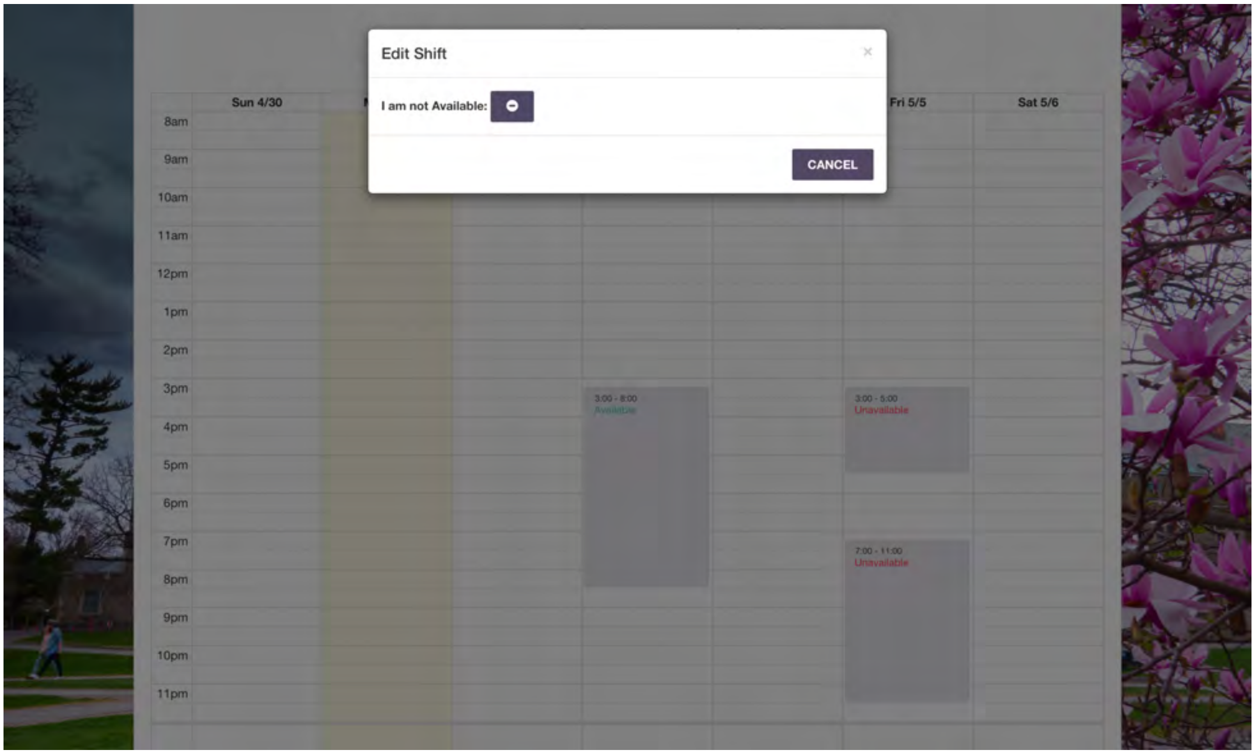
Figure 20: Deleting Shift on Availability Calendar



Edit Shift

I am Available:

Figure 21: I am Available



Edit Shift ×

I am not Available: ⊖

CANCEL

Sun 4/30 | Fri 5/5 | Sat 5/6

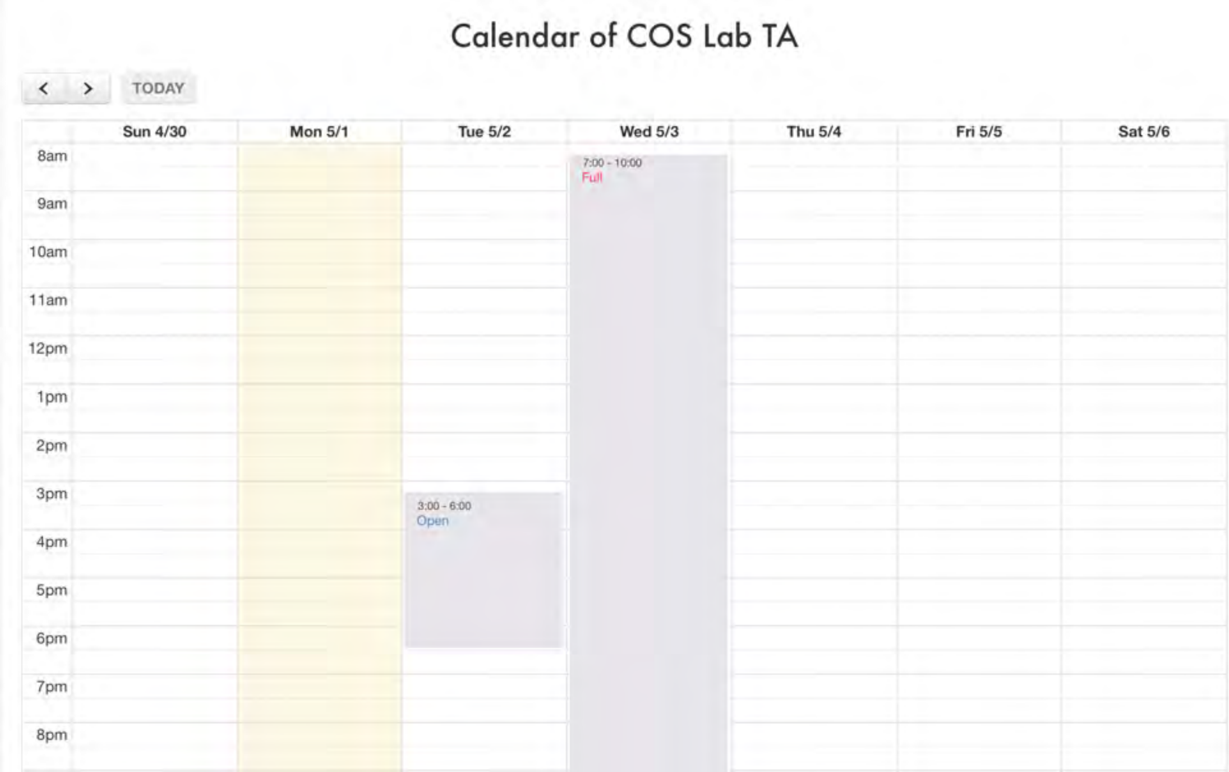
8am
9am
10am
11am
12pm
1pm
2pm
3pm
4pm
5pm
6pm
7pm
8pm
9pm
10pm
11pm

3:00 - 8:00
Available

3:00 - 5:00
Unavailable

7:00 - 11:00
Unavailable

Figure 22: I am not Available



Calendar of COS Lab TA

< > TODAY

Sun 4/30 | Mon 5/1 | Tue 5/2 | Wed 5/3 | Thu 5/4 | Fri 5/5 | Sat 5/6

8am
9am
10am
11am
12pm
1pm
2pm
3pm
4pm
5pm
6pm
7pm
8pm

7:00 - 10:00
Full

3:00 - 6:00
Open

Figure 23: Open or Full

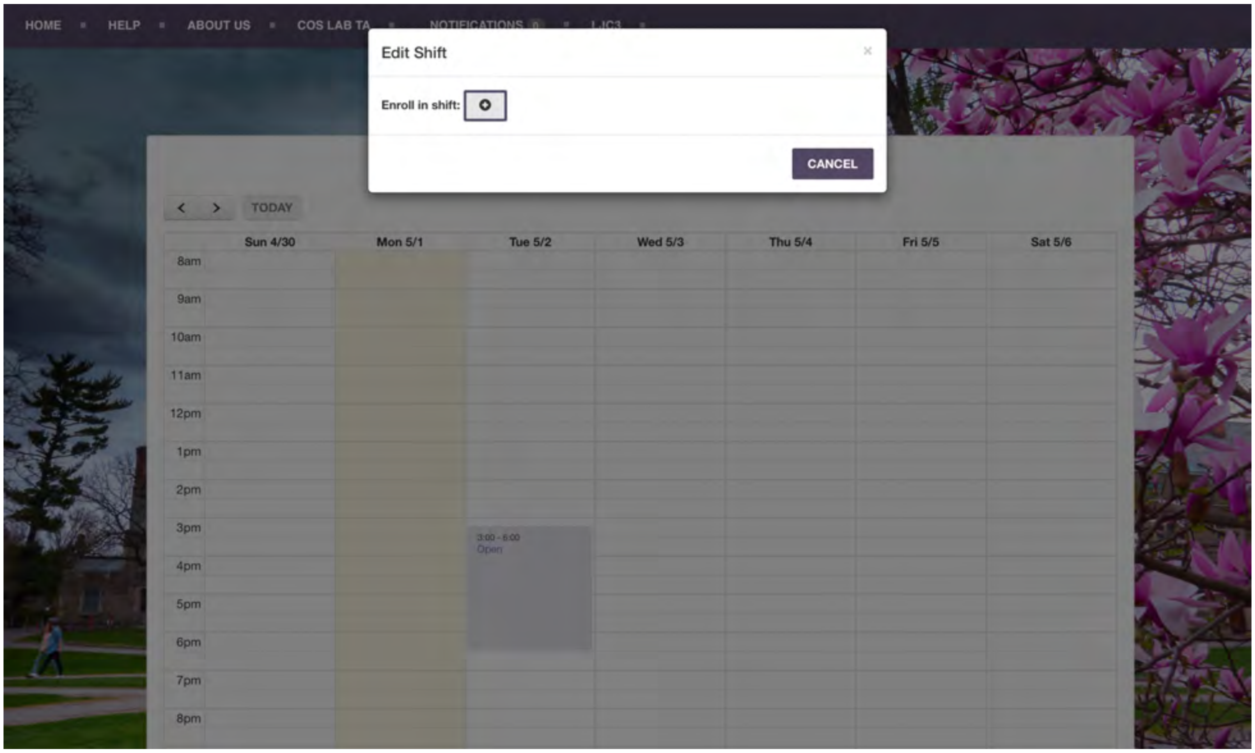


Figure 24: Enroll in Shift

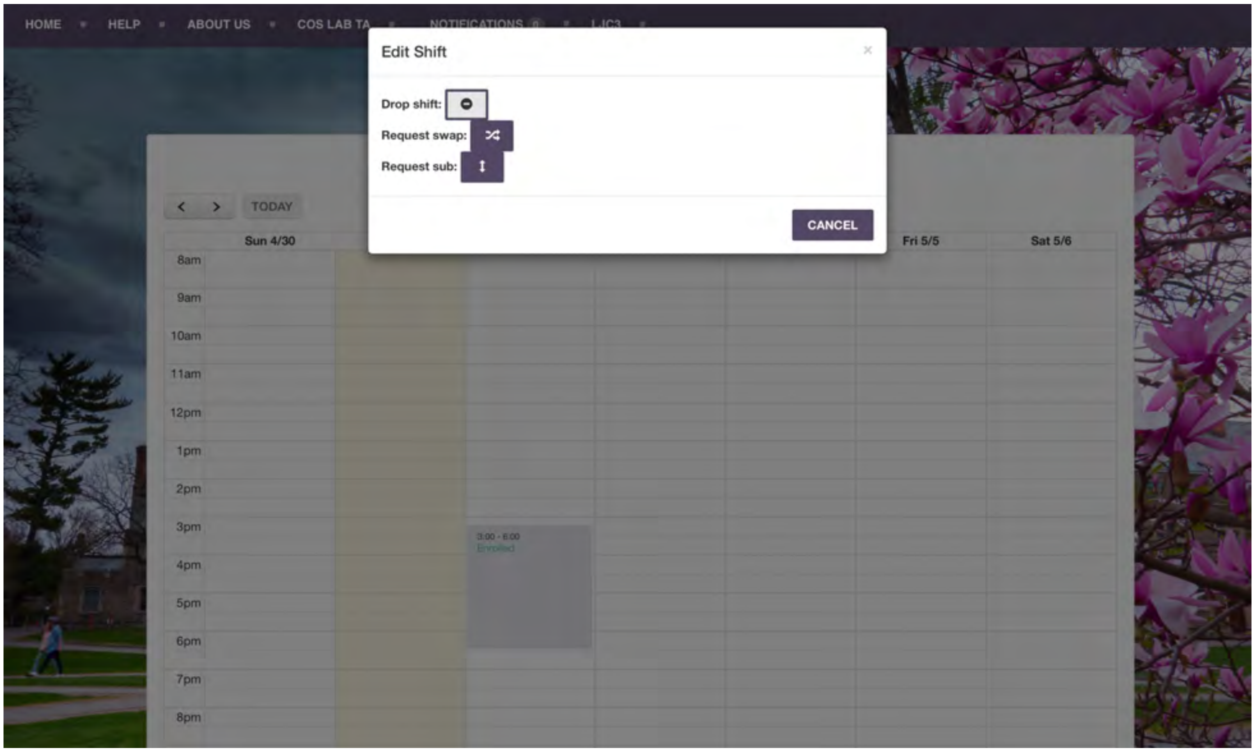
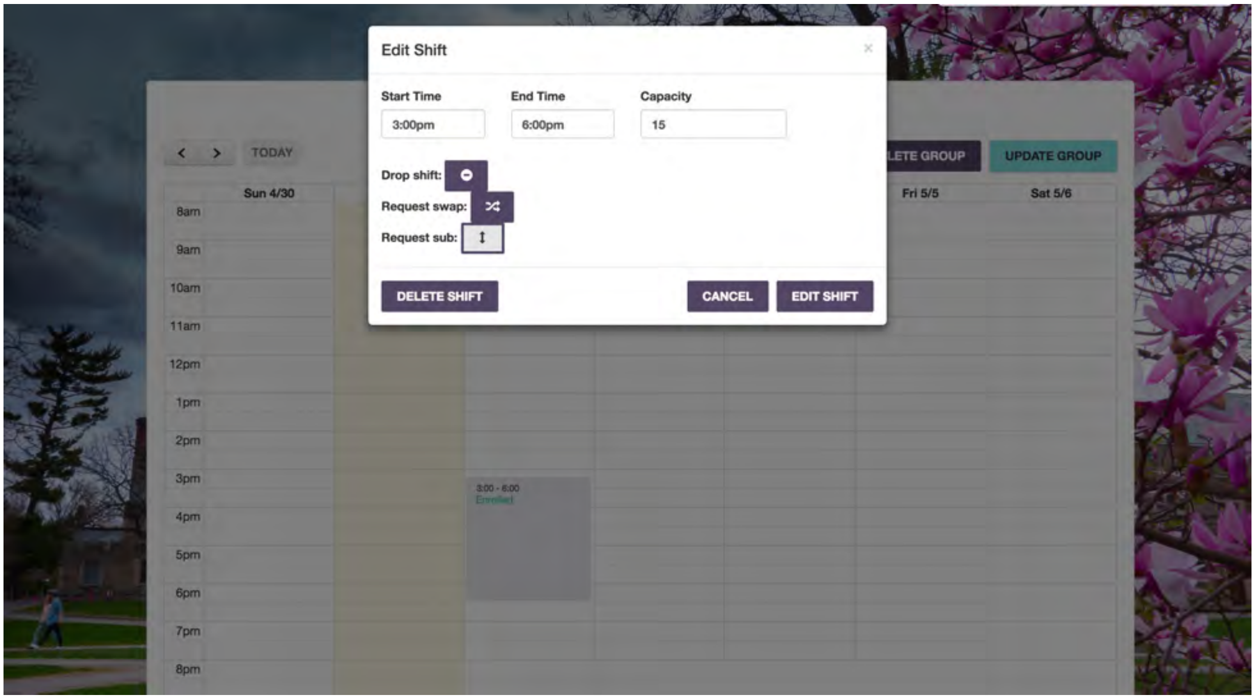


Figure 25: Drop from Shift



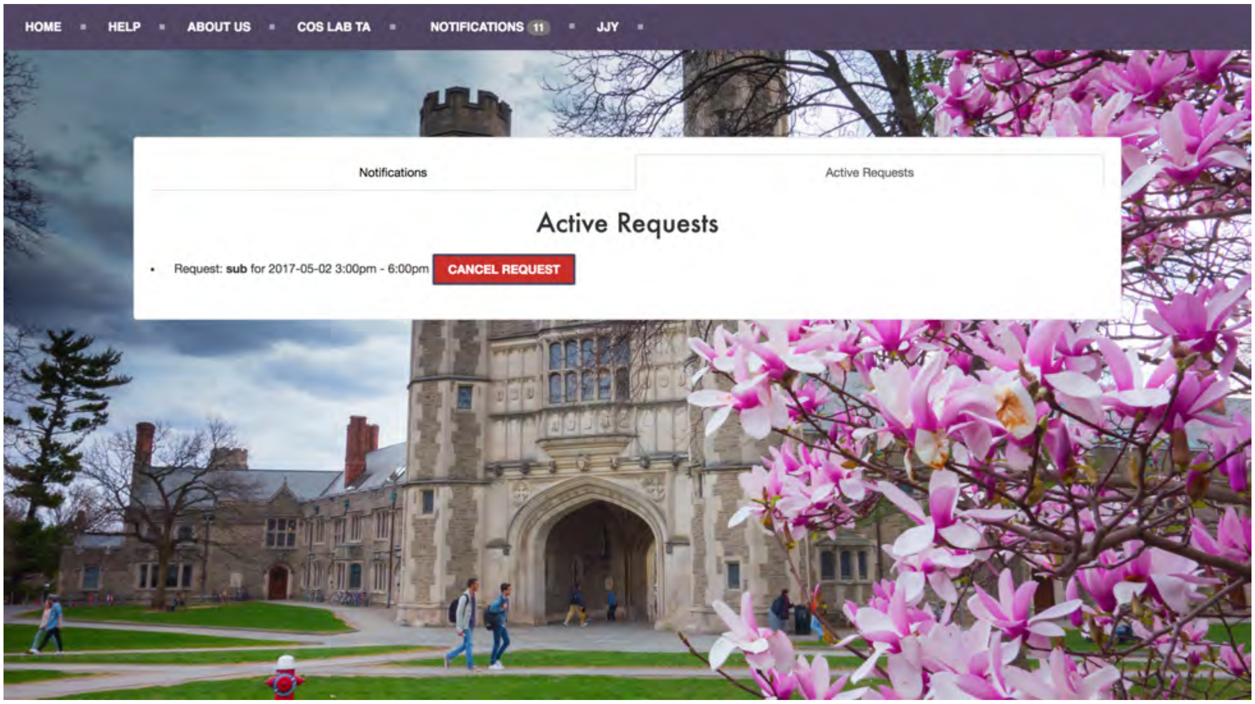
Edit Shift

Start Time: 3:00pm

End Time: 6:00pm

Capacity: 15

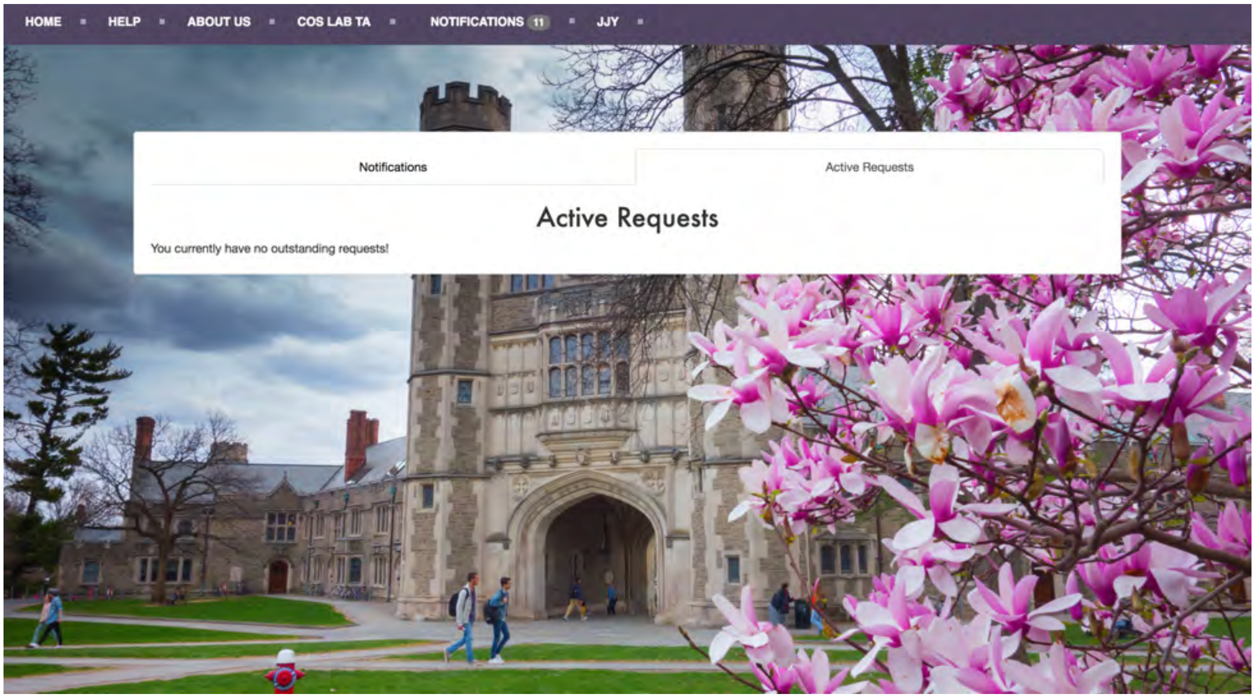
Figure 26: Request Sub for Shift



Active Requests

- Request: **sub** for 2017-05-02 3:00pm - 6:00pm

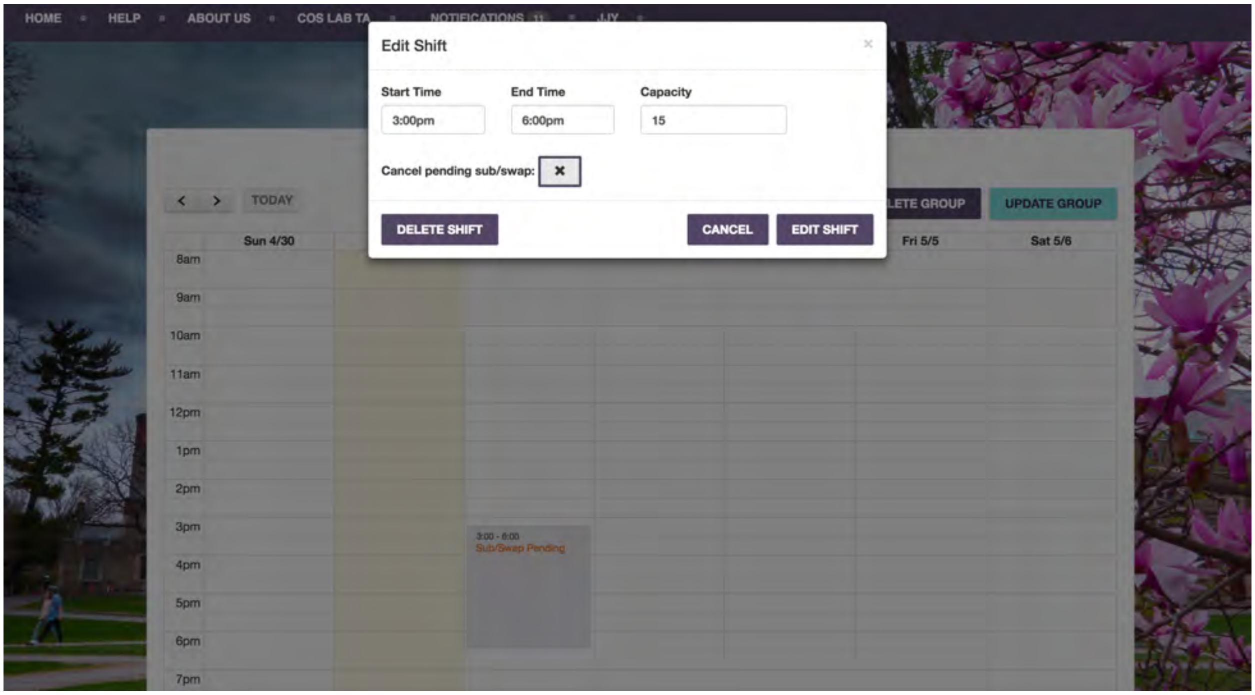
Figure 27: Cancel Sub



Active Requests

You currently have no outstanding requests!

Figure 28: No requests



Edit Shift

Start Time: 3:00pm

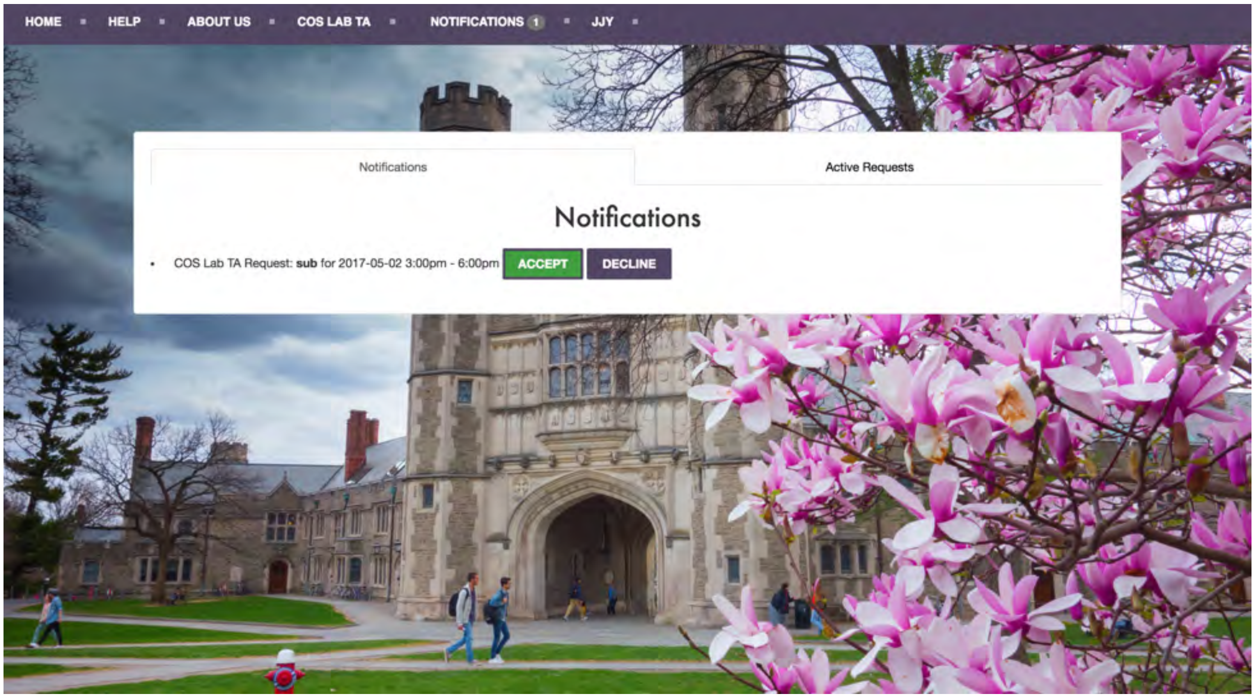
End Time: 6:00pm

Capacity: 15

Cancel pending sub/swap: ☒

3:00 - 6:00
Sub/Swap Pending

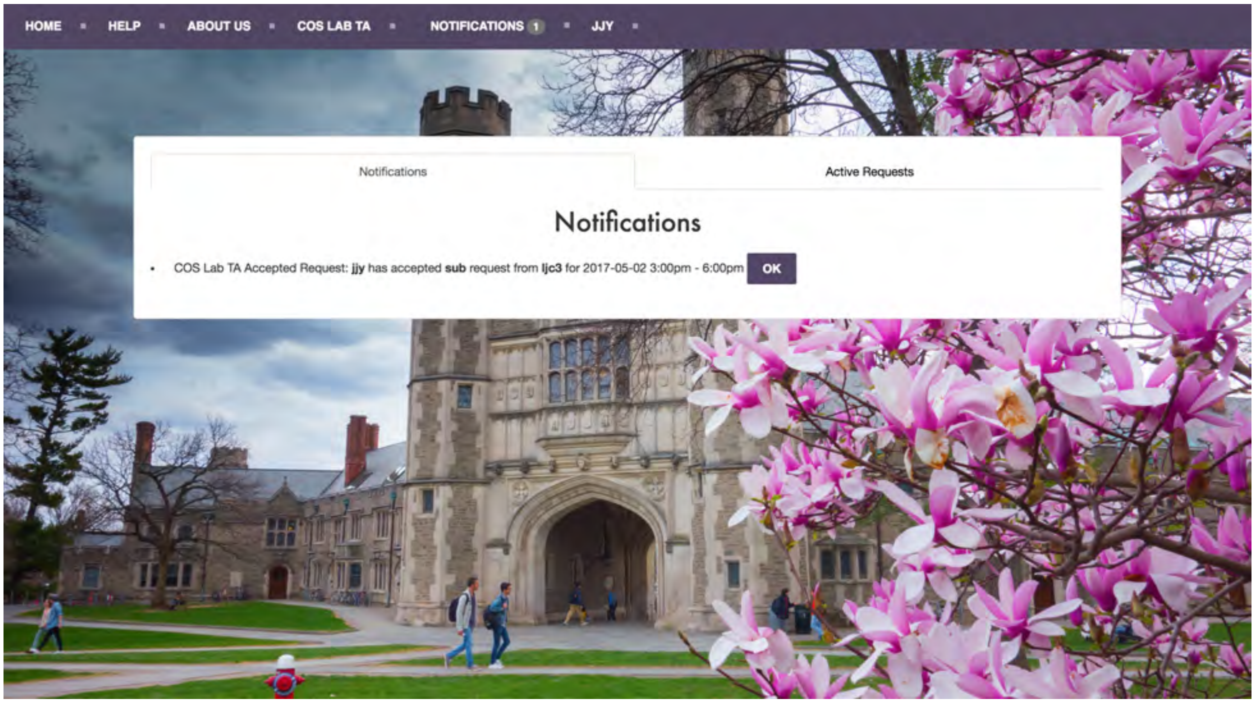
Figure 29: Cancel Sub Request



Notifications

- COS Lab TA Request: **sub** for 2017-05-02 3:00pm - 6:00pm

Figure 30: Accept Sub Request



Notifications

- COS Lab TA Accepted Request: **jjy** has accepted **sub** request from **ljc3** for 2017-05-02 3:00pm - 6:00pm

Figure 31: Accepted Sub Request

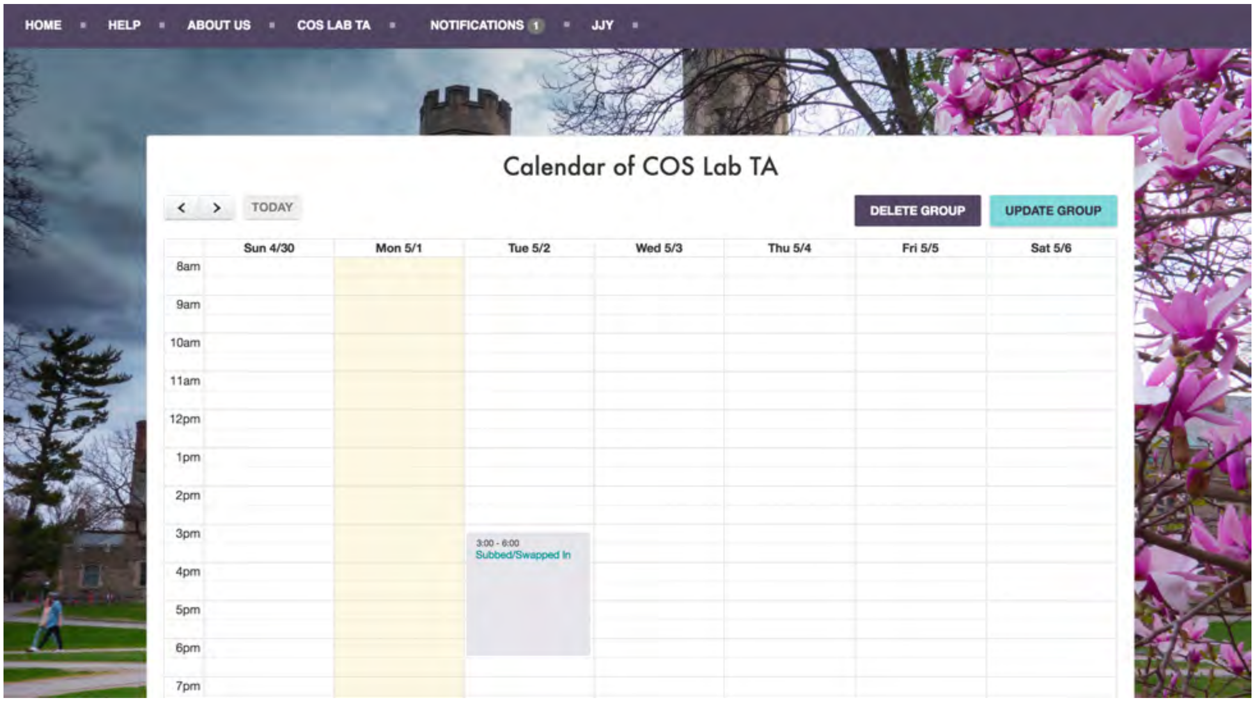
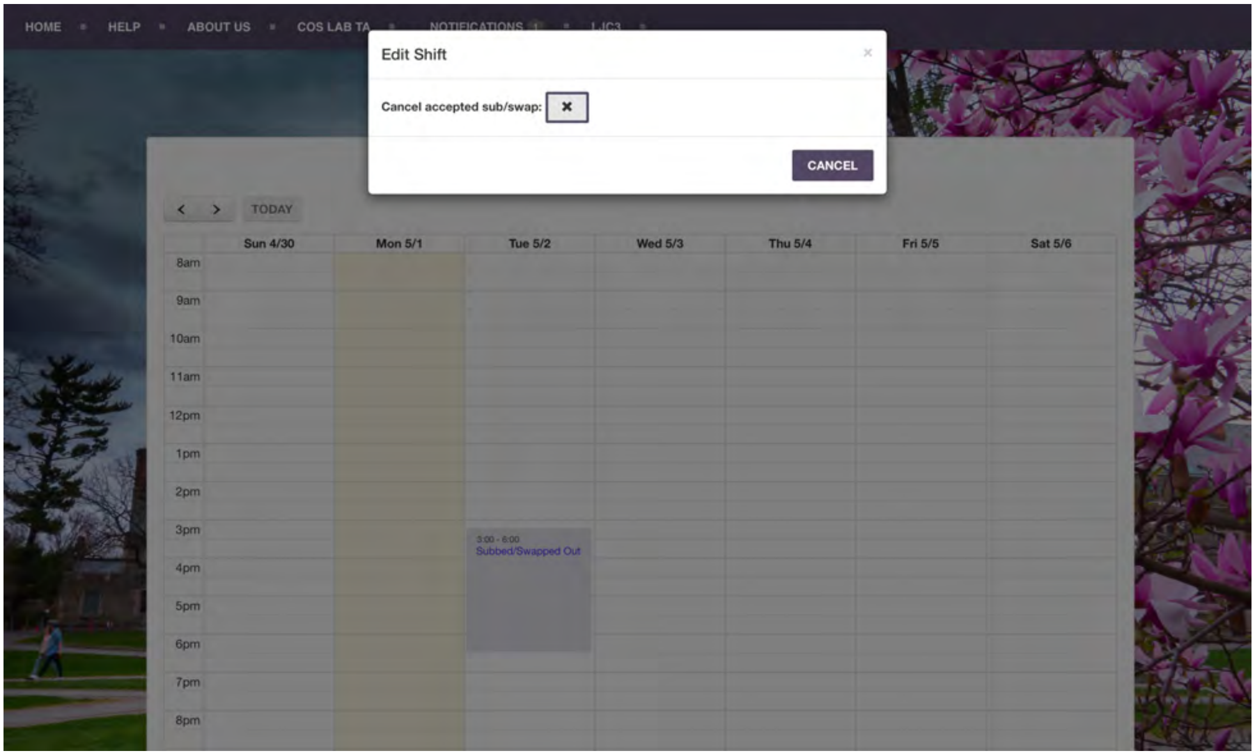
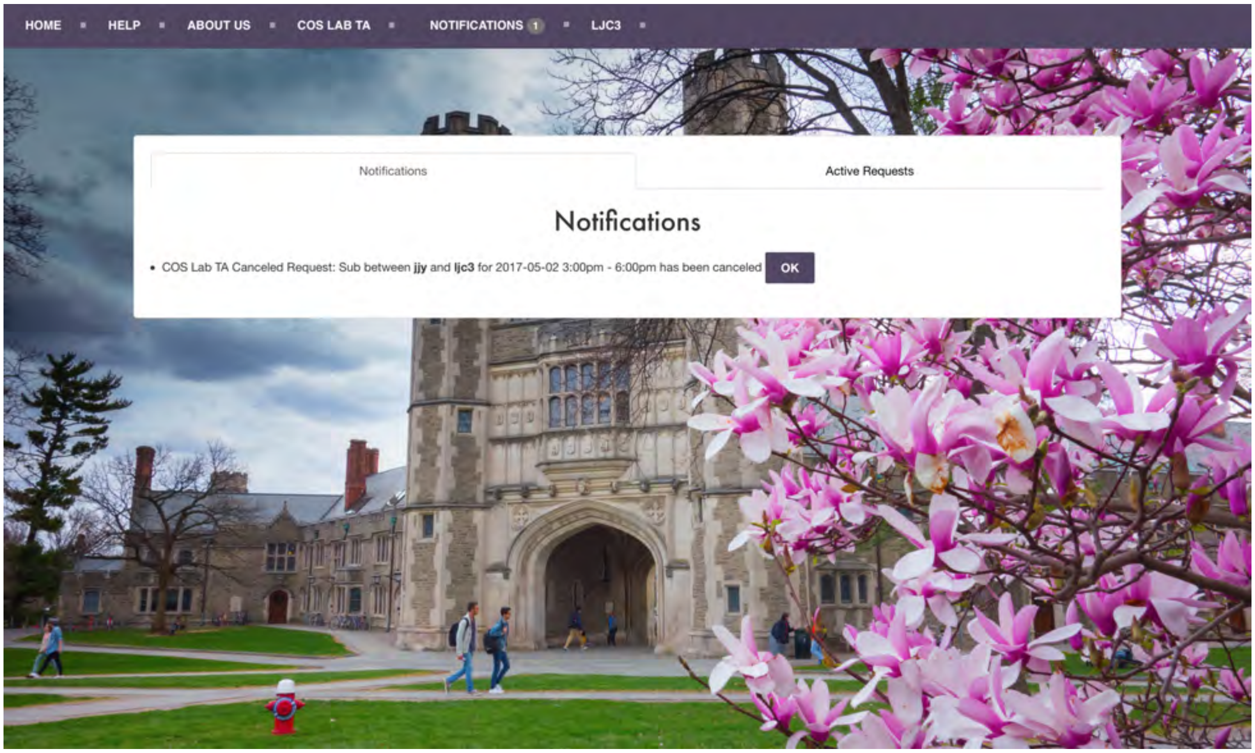


Figure 32: Subbed In



Cancel accepted sub/swap: ☒

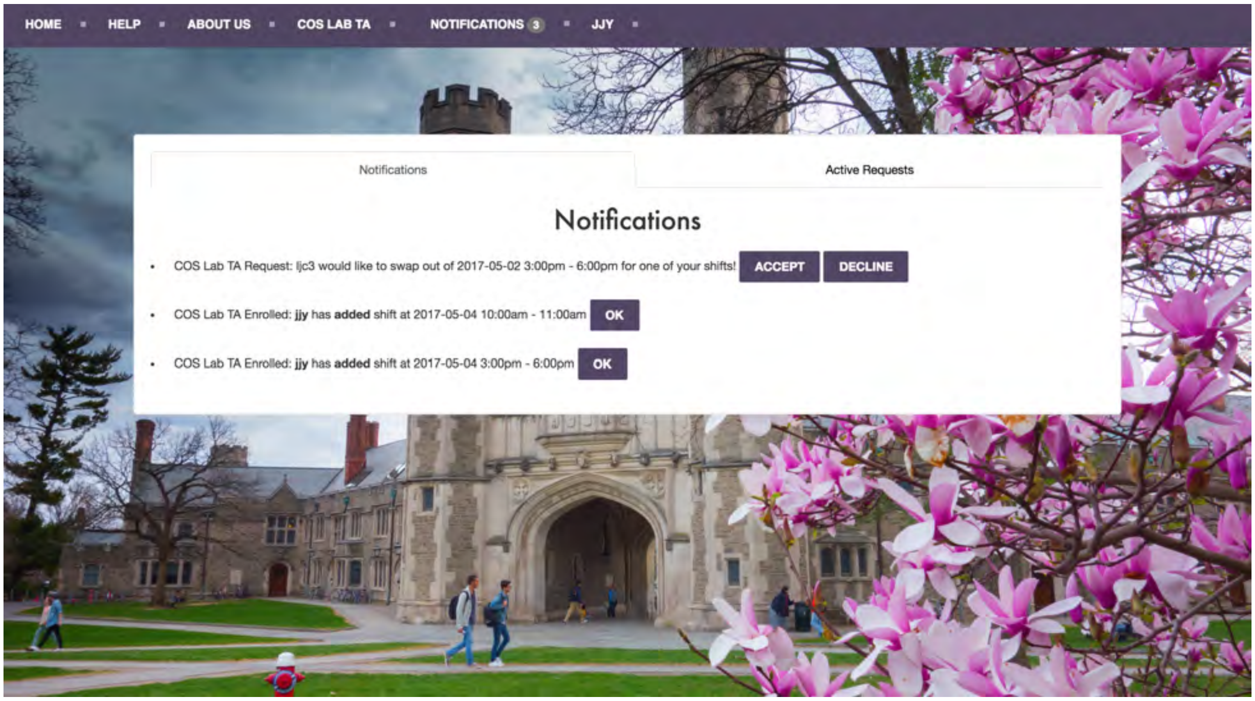
Figure 33: Cancel Request



Notifications

- COS Lab TA Canceled Request: Sub between **jjy** and **ljc3** for 2017-05-02 3:00pm - 6:00pm has been canceled

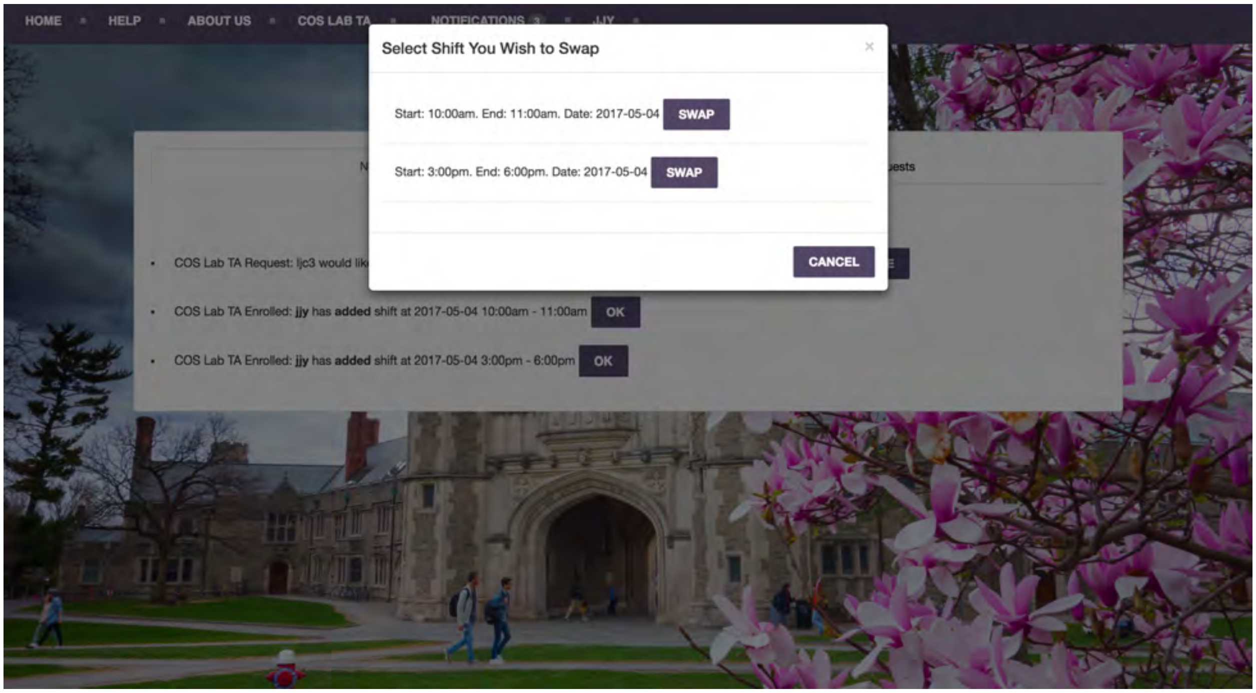
Figure 34: Notification for Canceled Request



Notifications

- COS Lab TA Request: ljc3 would like to swap out of 2017-05-02 3:00pm - 6:00pm for one of your shifts!
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 10:00am - 11:00am
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 3:00pm - 6:00pm

Figure 35: Notification for Swap Request



Select Shift You Wish to Swap

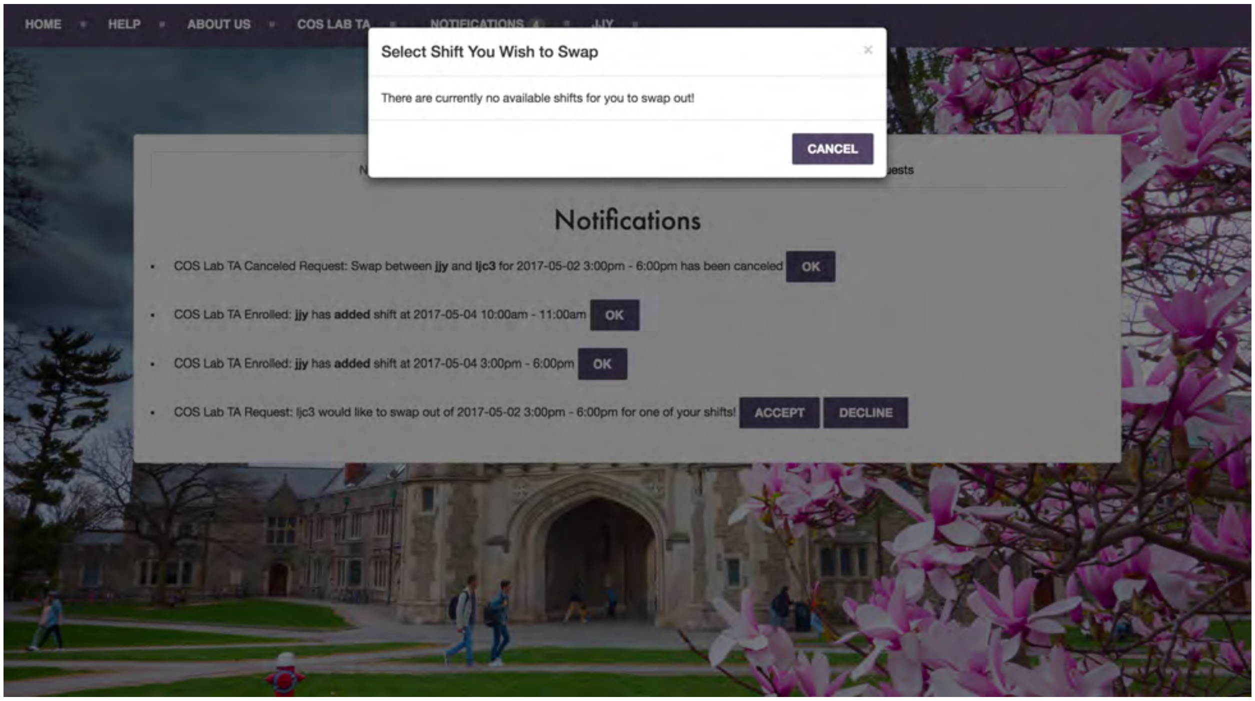
Start: 10:00am. End: 11:00am. Date: 2017-05-04 SWAP

Start: 3:00pm. End: 6:00pm. Date: 2017-05-04 SWAP

CANCEL

- COS Lab TA Request: ljc3 would lik
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 10:00am - 11:00am OK
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 3:00pm - 6:00pm OK

Figure 36: Select Swap Notification for Swap Request



Select Shift You Wish to Swap

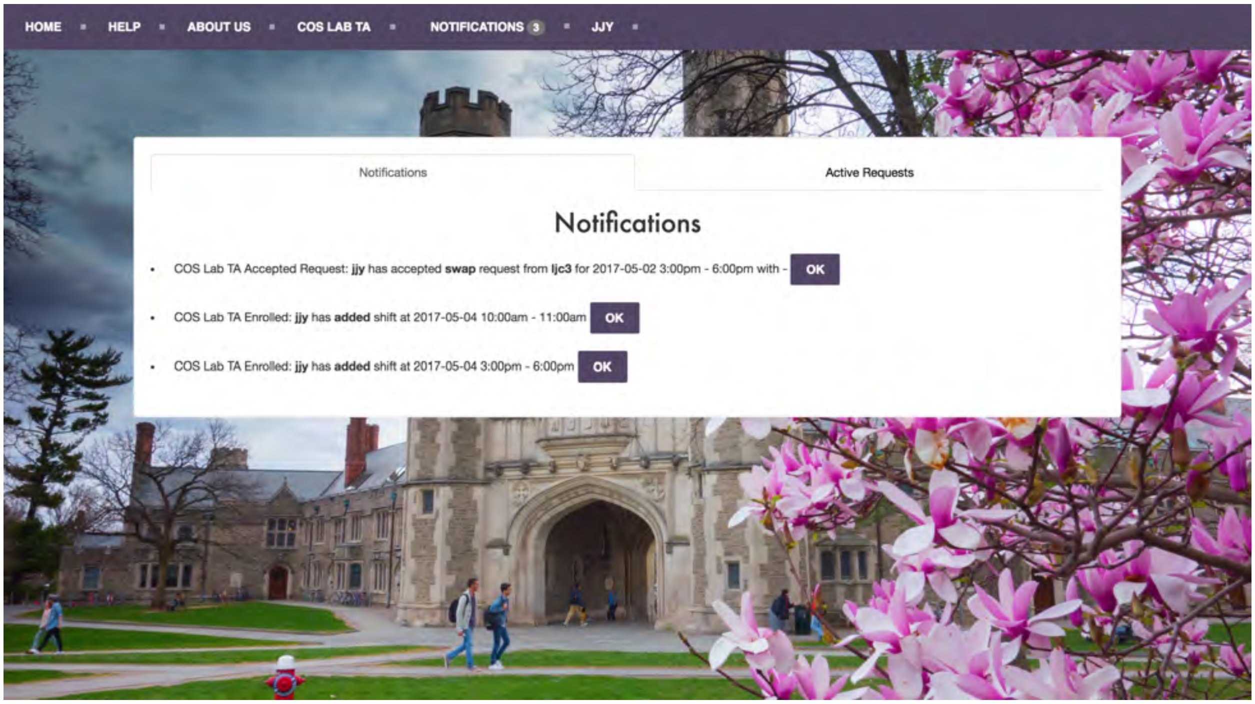
There are currently no available shifts for you to swap out!

CANCEL

Notifications

- COS Lab TA Canceled Request: Swap between **jjy** and **ljc3** for 2017-05-02 3:00pm - 6:00pm has been canceled OK
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 10:00am - 11:00am OK
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 3:00pm - 6:00pm OK
- COS Lab TA Request: ljc3 would like to swap out of 2017-05-02 3:00pm - 6:00pm for one of your shifts! ACCEPT DECLINE

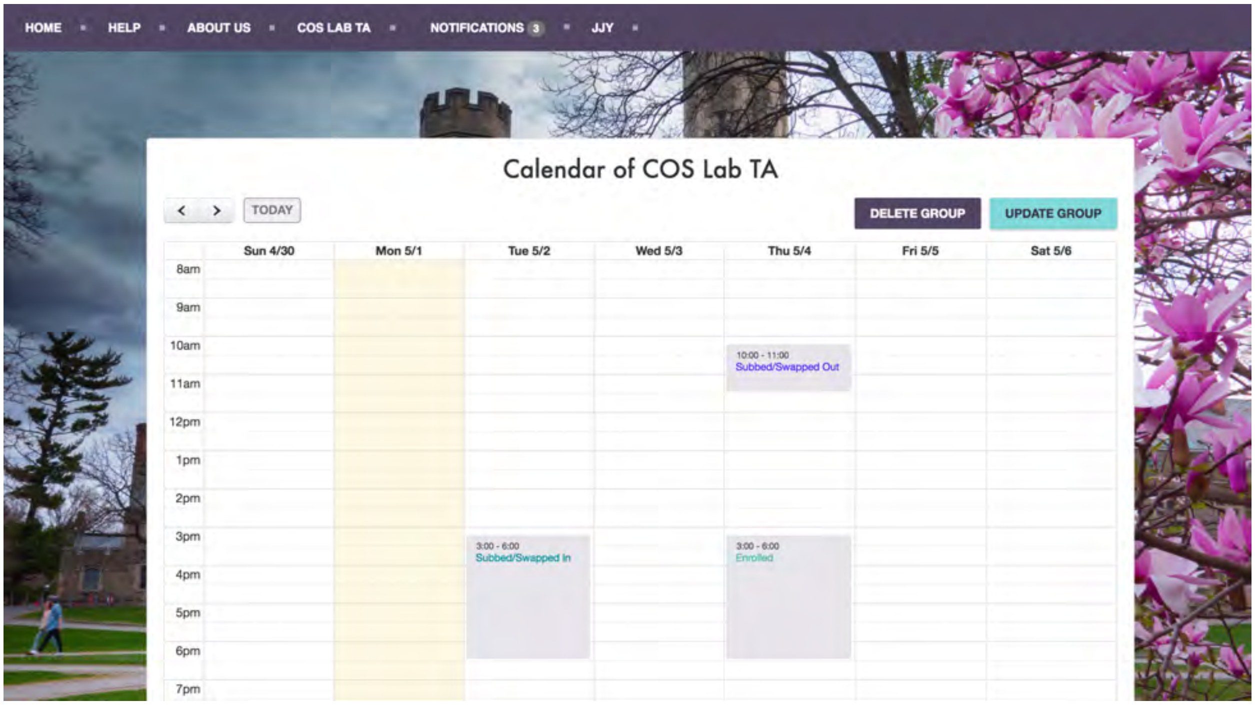
Figure 37: No Swap Notification for Swap Request



Notifications

- COS Lab TA Accepted Request: **jjy** has accepted **swap** request from **ljc3** for 2017-05-02 3:00pm - 6:00pm with - OK
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 10:00am - 11:00am OK
- COS Lab TA Enrolled: **jjy** has **added** shift at 2017-05-04 3:00pm - 6:00pm OK

Figure 38: Accepted Swap Notification for Swap Request



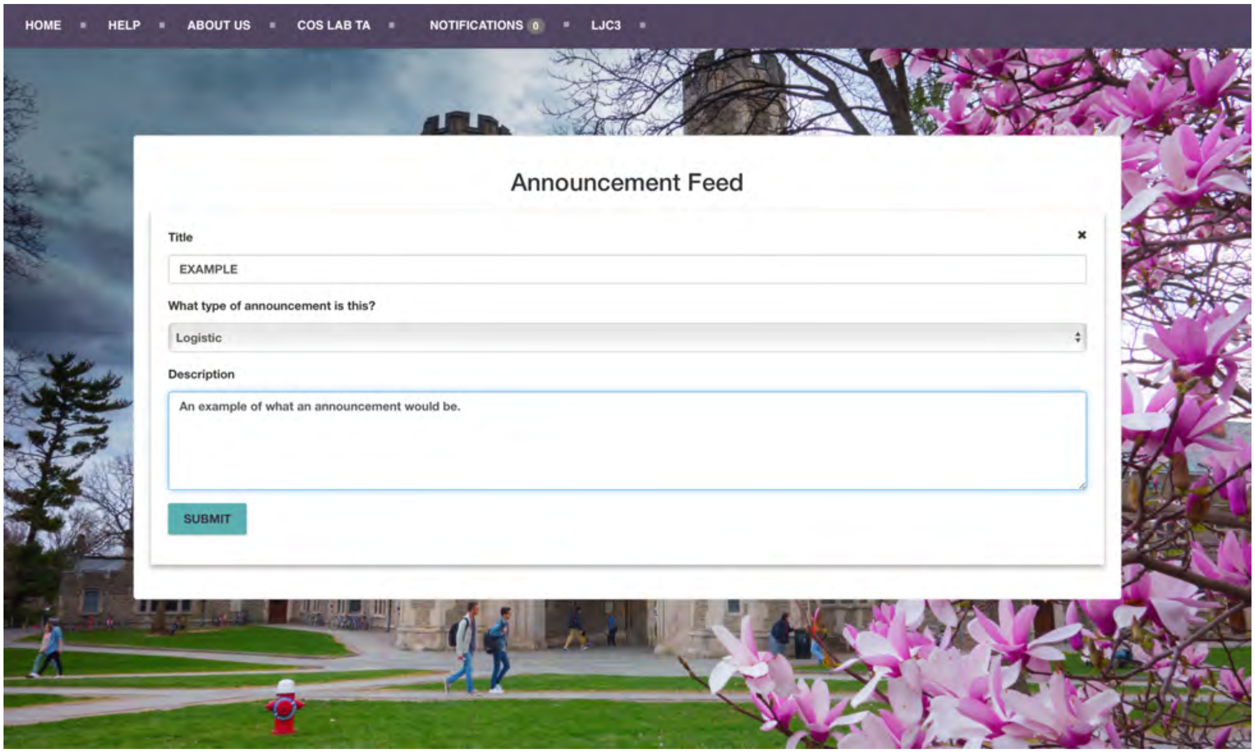
Calendar of COS Lab TA

10:00 - 11:00
Subbed/Swapped Out

3:00 - 6:00
Subbed/Swapped In

3:00 - 6:00
Enrolled

Figure 39: Calendar after Swap Request



Announcement Feed

Title

EXAMPLE

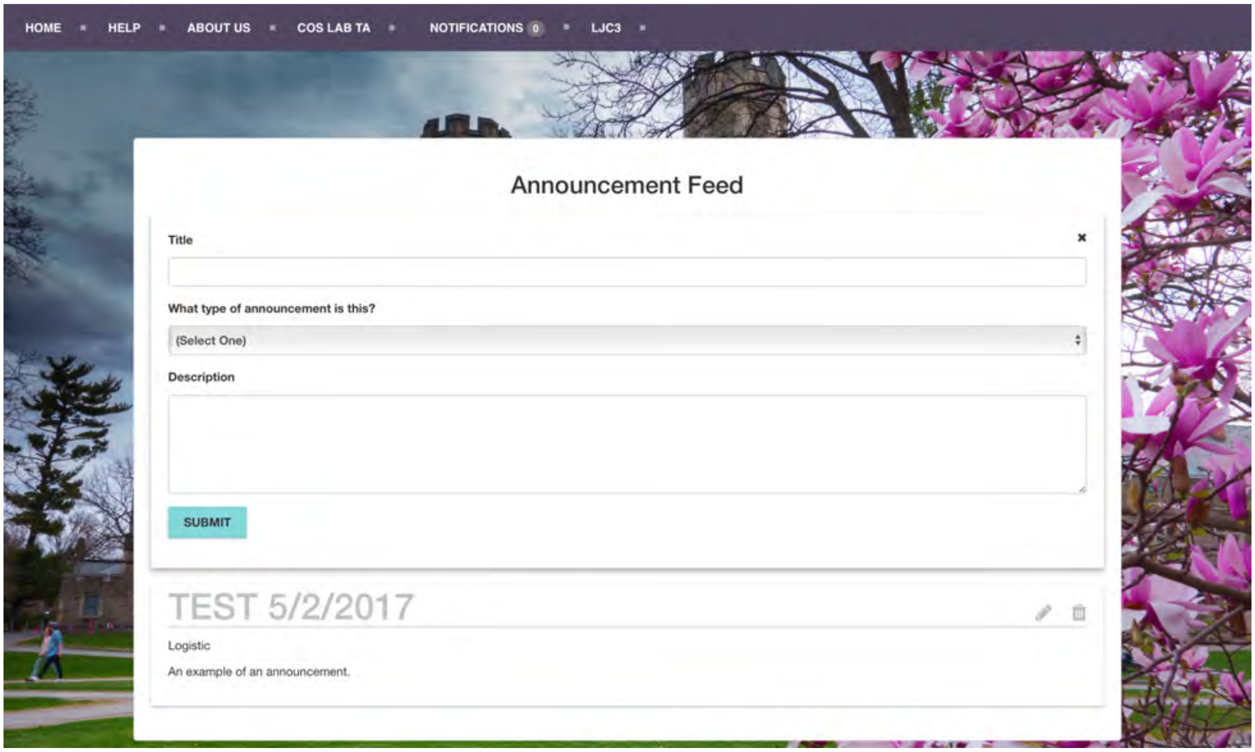
What type of announcement is this?

Logistic

Description

An example of what an announcement would be.

Figure 40: New Announcement



Announcement Feed

Title

What type of announcement is this?

(Select One)

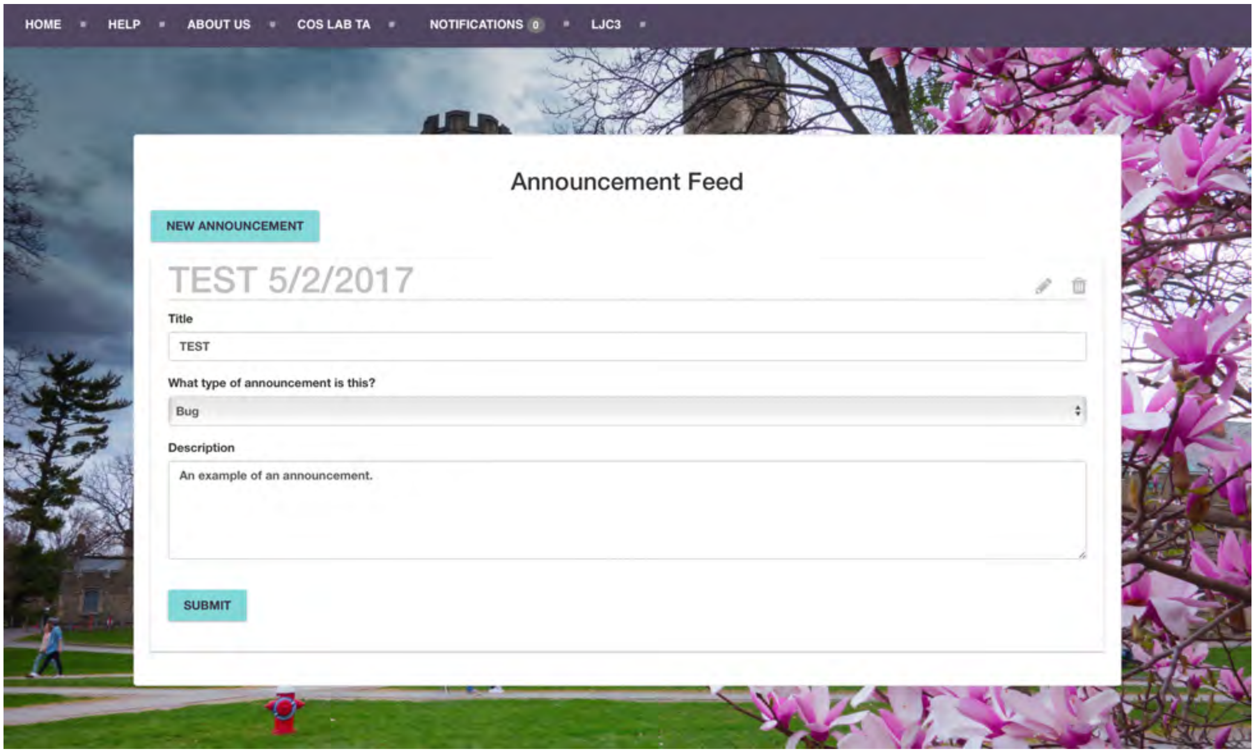
Description

TEST 5/2/2017

Logistic

An example of an announcement.

Figure 41: Announcement Posted



Announcement Feed

TEST 5/2/2017

Title

TEST

What type of announcement is this?

Bug

Description

An example of an announcement.

Figure 42: Editing Announcement