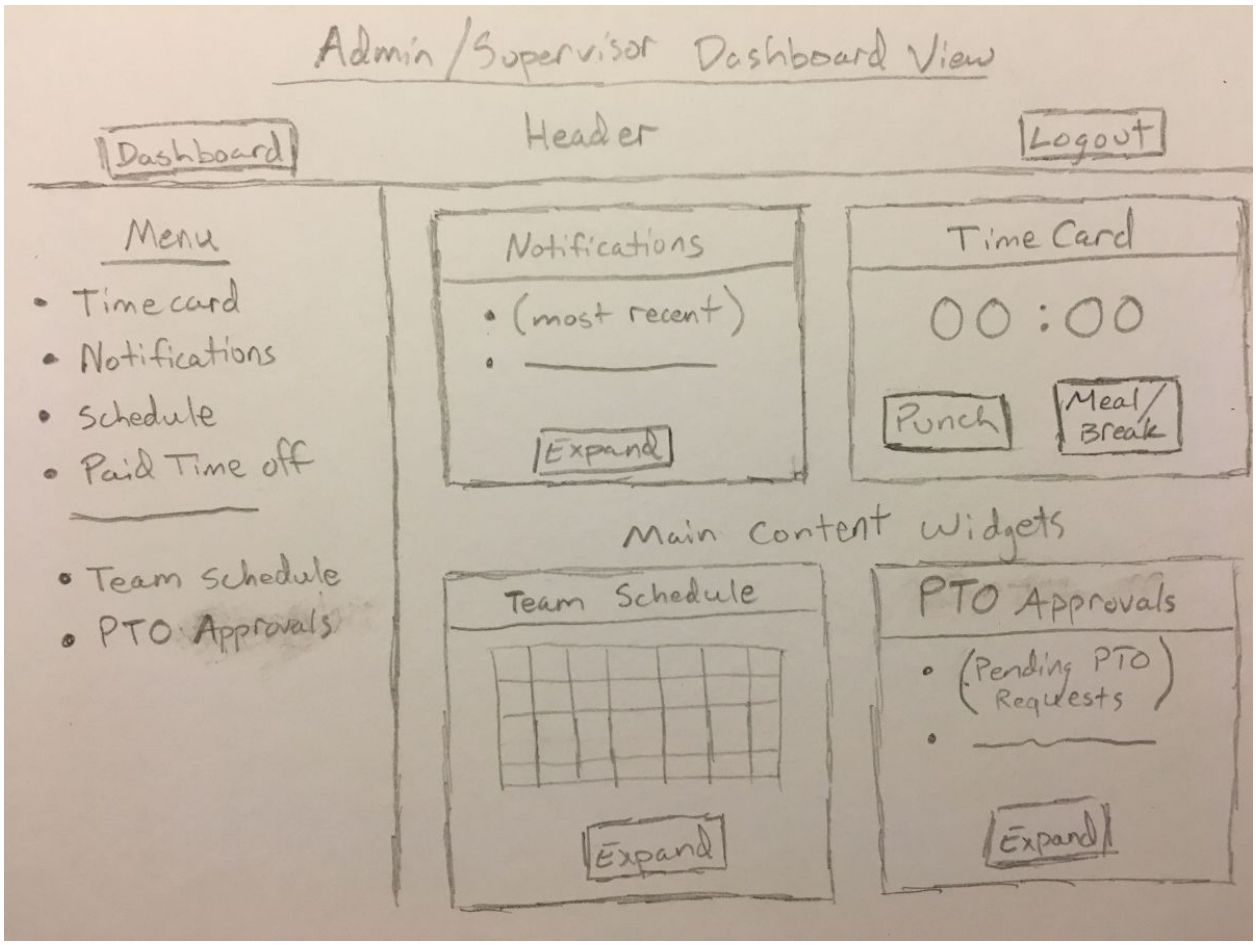
Scheduling and Time Management Application - Requirements

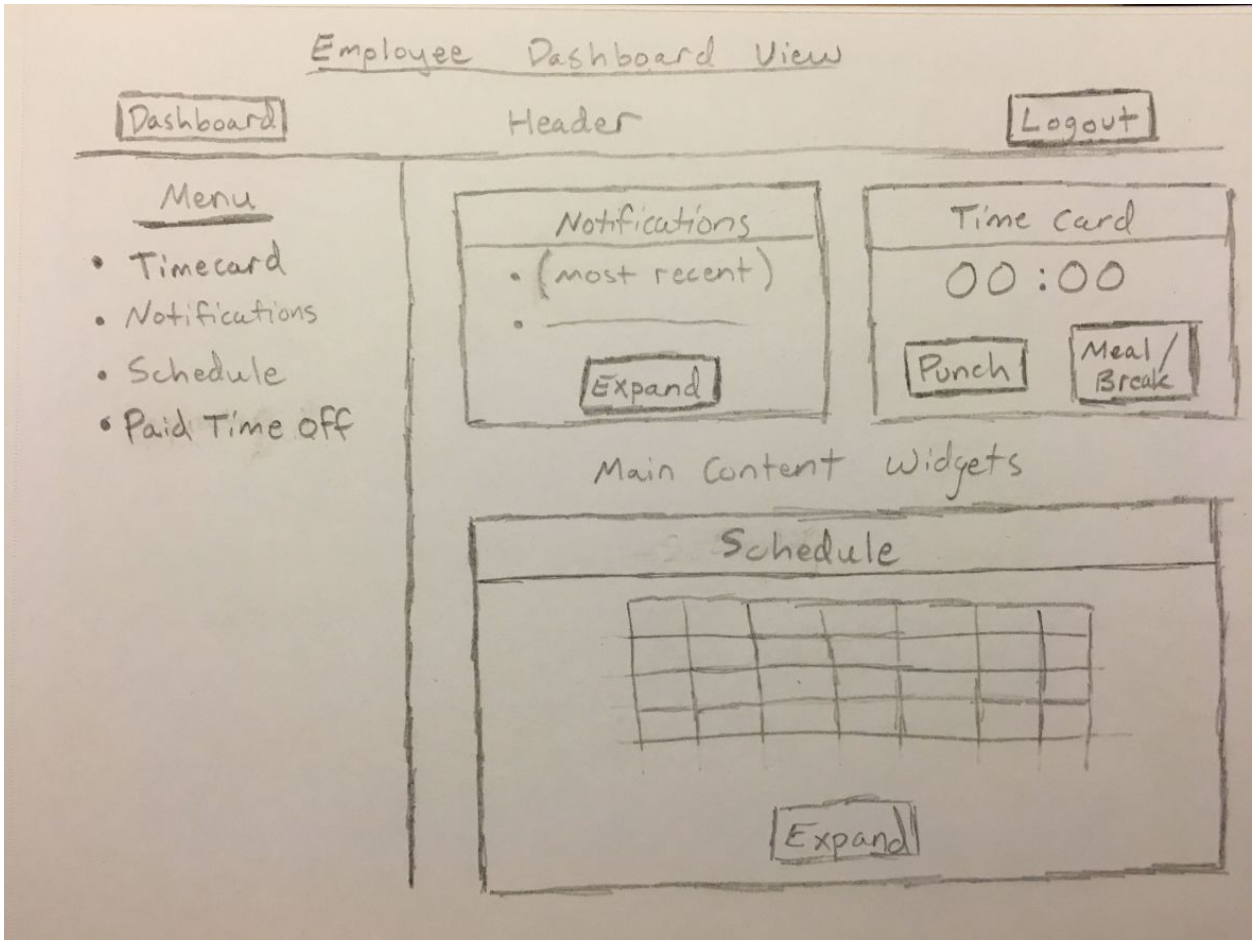
Group 20:
Sunghoon Cho
Michael Czapary
Christopher Elliott
Alex Kolstad
Zachary Wetekamm

Paper Prototypes

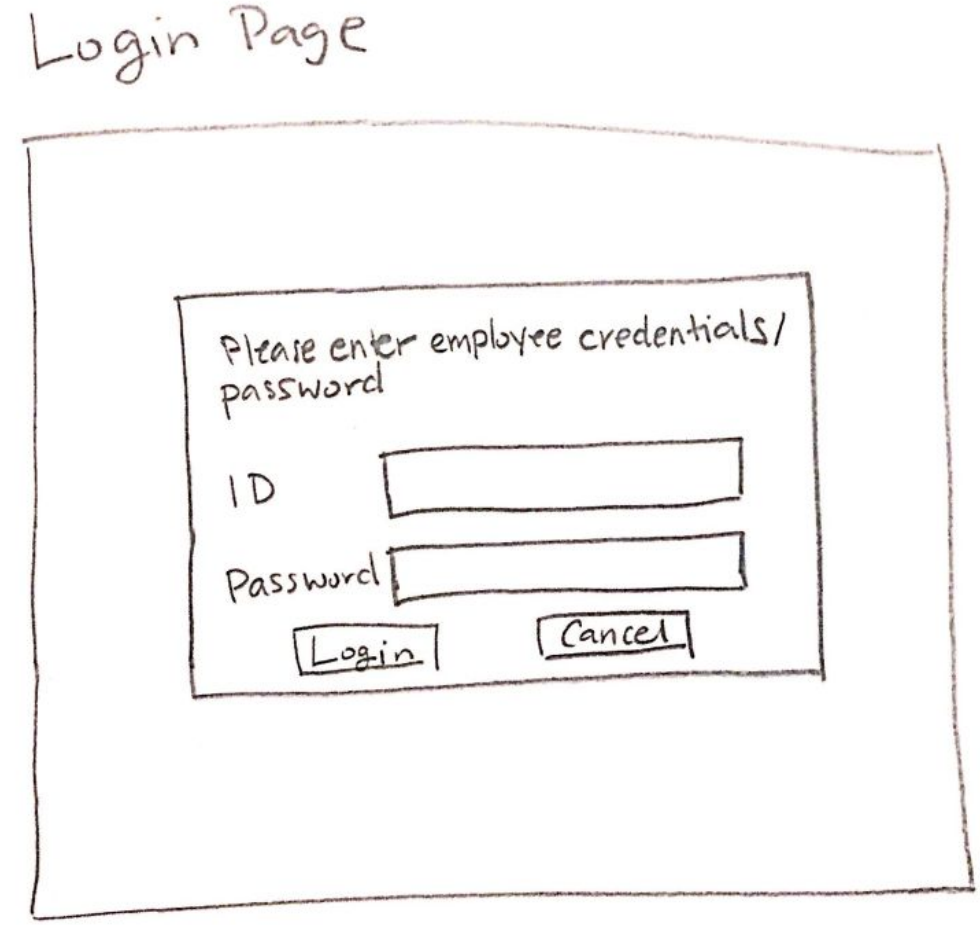
Admin/ Supervisor Dashboard:

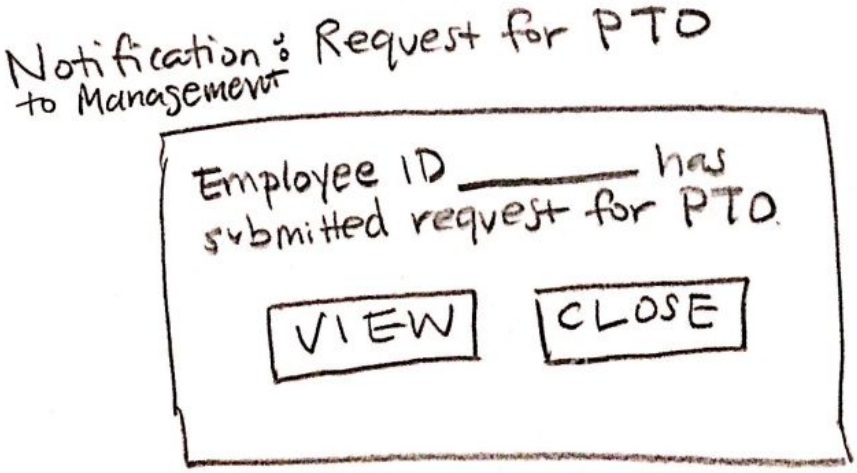


Employee Dashboard:



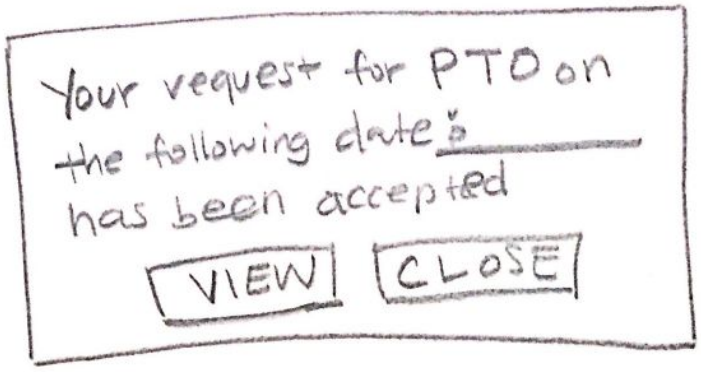
Login Screen and Notifications:



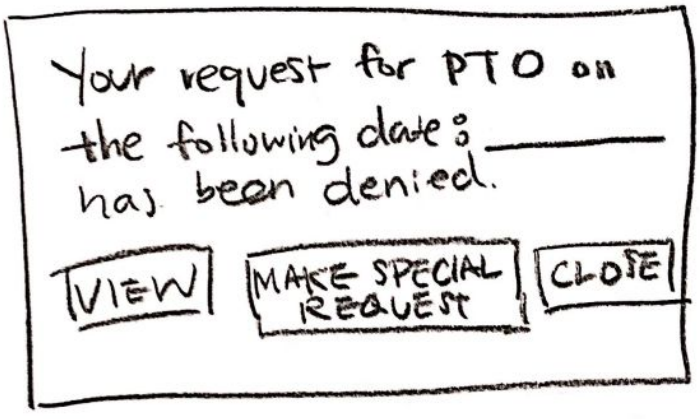


Notifications cont.

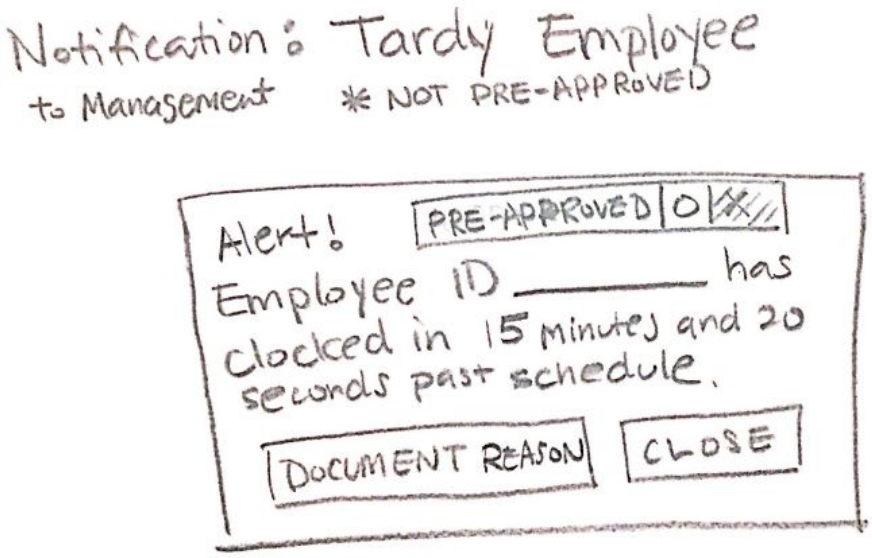
Notification: Acceptance of PTO Request to Employee

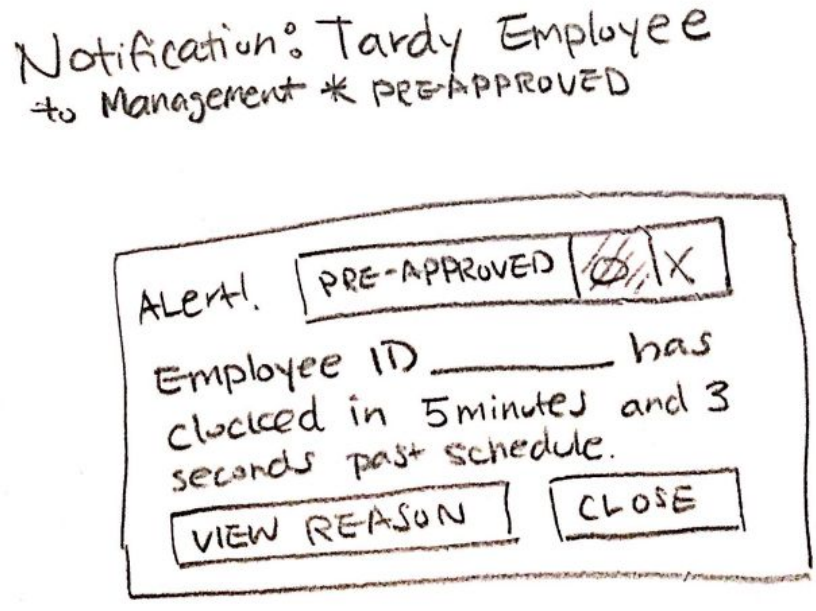


Notification: Denial of PTO Request to Employee

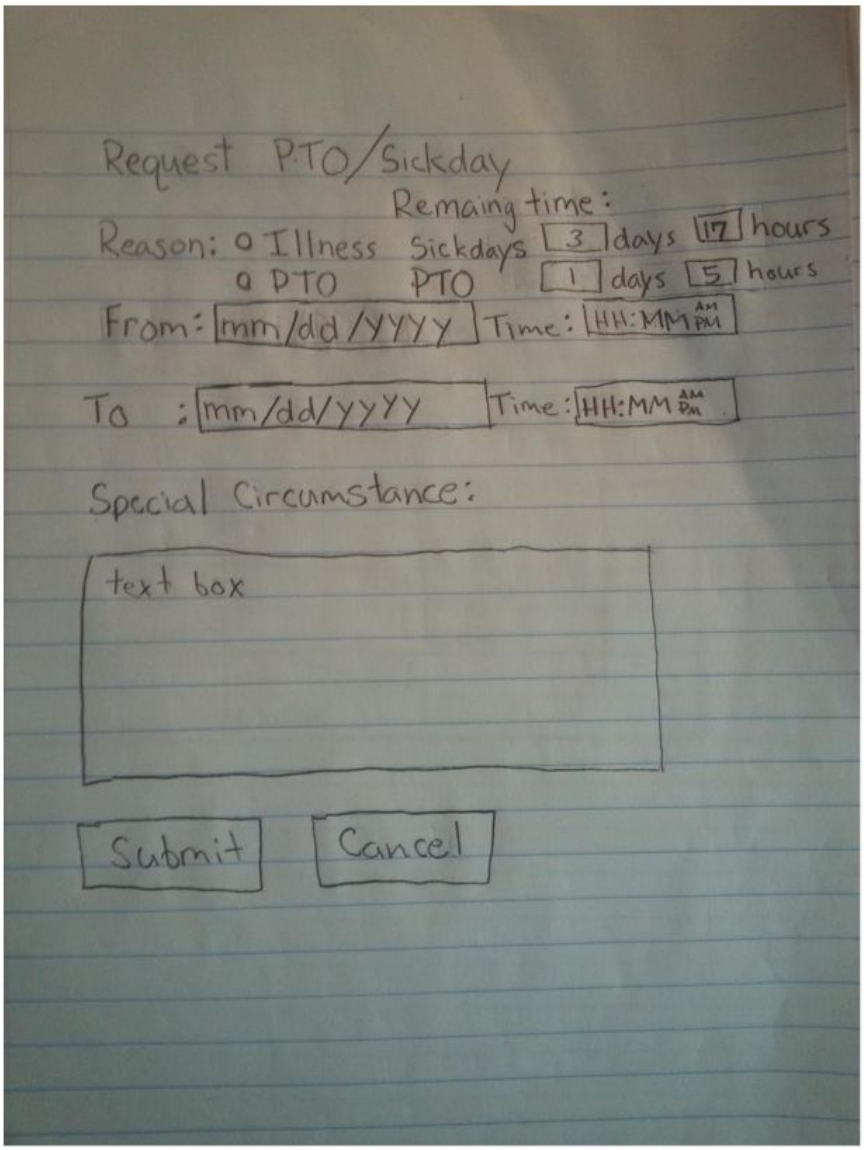


Notifications cont.

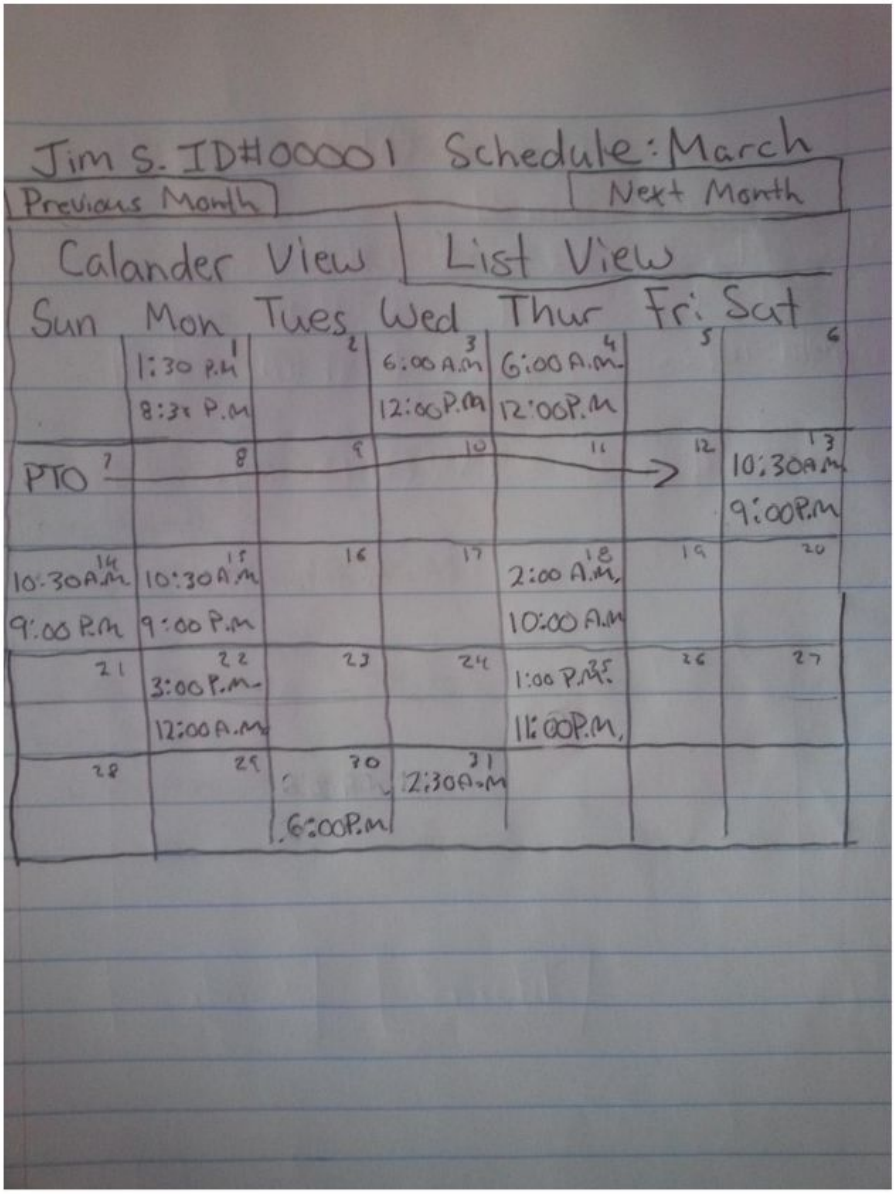




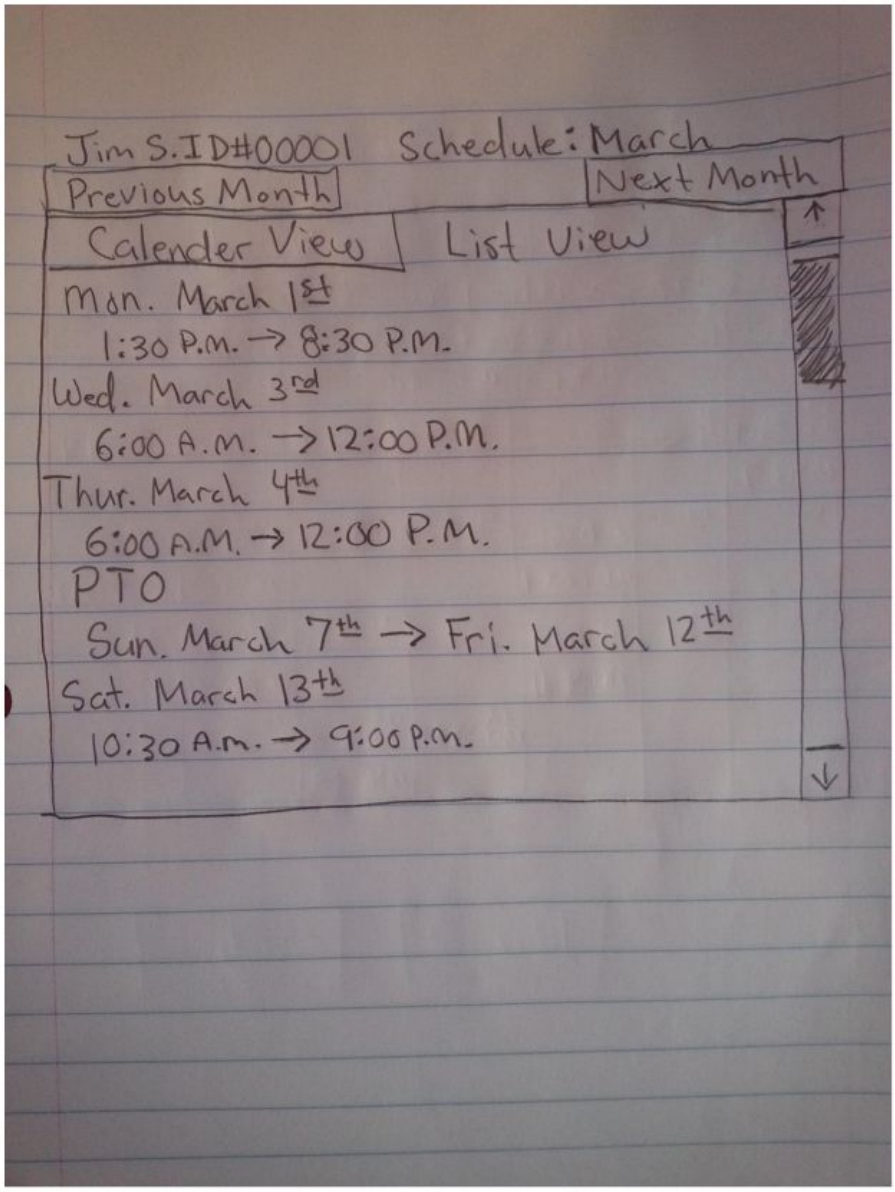
Request PTO/STO screen



Schedule Calendar View:



Schedule List View



Requirements Definition (HW 2)

Functional Requirements

- The system shall authenticate the employee's credentials before logging them in
- The system shall allow employees to check their schedules and work history from anywhere
- The system shall allow employees to use sick days and vacation days
- The system shall allow employees to communicate with management to apply for paid time off, and will notify the administration of the day and time of the days being taken off
- The system shall notify employees that a request is under review if it is not granted automatically
- The system shall notify an employee if a request is denied with a reason why it was denied (not enough PTO available, etc.)
- If a request is denied the system shall allow an employee to make a special request to their supervisor to explain special circumstances
- The system shall keep track of available and used sick days and PTO for employees
- The system shall adjust the remaining PTO and sick days available to an employee when a request for sick days or PTO is granted
- The system shall, if allowed by management, allow employees to create their own schedule
- The system shall be universal for all jobs in the workplace
- The system shall track employee's lateness
- The system shall identify employees who are working more than a standard number of hours per week
- The system shall track when people clock in and out and keep a record of people's tardiness, and to make that information available to management
- The system shall not allow employees to clock in from any location other than the work site
- The system shall allow employees to clock in and out, clock in and out to take their lunch, and clock in and out for their break time
- The system shall notify management when someone is late
- The system shall make daily, weekly, and monthly reports of when employees clock in late
- The system shall allow management to adjust when these notifications are sent (i.e. after 5 mins. Late, 30 mins. Late etc.)
- The system shall allow management two options for notifications: Email and Text

- The system shall allow administrators to edit recorded timestamps
- The system shall keep a record of edited timestamps
- The system shall not allow an administrator to edit their own timestamps

Non-Functional Requirements

- The system shall log users in within 20 seconds
- The system shall log the user out after 5 minutes of inactivity
- The system shall notify management within 5 minutes of the employee clocking in
- The system shall notify management within 5 minutes of a sick day or PTO request being made
- The system shall notify and employee within 5 minutes of a sick day or PTO request being granted or denied by management
- The system shall send message between management and employees in less than 5 minutes
- When a sick day or PTO is used the system shall update the remaining time available within 20 seconds
- The system shall update employee schedules at least once per day if changes are made

Requirements Specification (HW 2)

Functional Requirements

- User details with be validated with their login information stored in the database
- The system will store work schedules in a database
- The system will keep track of employee sick days and PTO in a database
- The system will use email to allow employees to request sick days or PTO
- The system will use email to allow employees and management to communicate with each other
- The system shall track changes to timestamps by administrators in a database. The record of these changes should be uneditable.
- The system shall use information on employee tardiness stored in the database to create reports in MS excel.
- The system shall compare login information with employee information to prevent an employee from editing their own timestamp
- The system will keep a record of the days late and by how much time an employee is late in a database
- The system will check login IP address to ensure employees are only logging in from the work site

Non-Functional Requirements

- The system shall query the employee database and verify the employee's login credentials within 20 seconds
- The system shall update stored work schedules in the database within 1 minute of a change being made
- The system shall deliver messages between employees and management within 20 seconds of a message being sent
- The system will query the database to present updated sick days and PTO
- When a change to a timestamp is submitted a copy of the original timestamp should be automatically saved in a database with the ID of the manager making the change. This database entry should be read-only
- The system shall query the database and automatically fill the information into an MS excel spreadsheet to generate reports on employee tardiness

Customer Meeting:

The group reached out to meet with the client. A response was not immediate. The response that was received requested a meeting later in the week. However, while times were made available for the client to confirm a meeting, no final response was provided granting a meeting with the client this week.

Requirements Changes:

While specific feedback from the client would have been ideal, there was some feedback provided by the TA which allowed us to make immediate adjustments. A variety of changes were discussed within the group also and Michael was able to make client facing commentary as he continued his role as client as he is the client for another group as well.

During the first meeting with the client, it was mentioned multiple times that the system would be used to help troubleshoot issues within the culture of the workplace. We therefore wrote this in our first list of requirement definitions. The TA commented that this was not clear. We agreed that this was not clear how the system was meant to accomplish this from speaking with the client. She may have intended this purpose to be more of a use of the tool rather than something innate to the tool itself. It was asked if this would be something used by an HR team but were told that it would not. Despite the use of a client surrogate, after lengthy discussion, the group made the decision that this was a complicated feature that would almost certainly require HR involvement for any methodology we could imagine. For the time being, as we were unable to receive specific feedback on this matter, we have removed it from our list of definitions. We may revisit this issue at a later date if more clarification can be provided.

We added multiple requirement definition list entries concerning the notifications of lateness to an administrator or supervisor. Previously, we had mentioned it in non-functional requirements but were not specific. We have added that they exist and management ability to select how, ie. email or text. Other options may become available as the system progresses but as the requirements are to be settled up front in the Waterfall method, these two options were solidified. Option settings had been placed in the second Use Case but we had not applied them to the requirements. These new settings have also been included in the revised ERD.

In addition, we have included timestamp editing, recording of each of these edits and by whom, and stated that self-edits will not be allowed. These features had been placed in the first Use Case but we had neglected them in the requirements. This feature has been included in the revised ERD as well.

One of our more overlooked sets of requirements was in that of the requests for paid time off and sick time off. Again, these conditions were stated in Use Case 3 but we were not specific in the requirements list. We have added a definition specifying number of days. We have added a notification of approval or denial to the employee. We have also added that the system will track the number of available days to be taken for vacation and sick time which by updating said time when it becomes available and when it is taken by request. In addition, the requestor may now add reasons for the requests available for review by the administrator or supervisor.

The group had a mutual blind spot differentiating between requirements definitions and requirements specifications. The TA had commented that the list had simply been repeated. After lengthy discussion, we have essentially rewritten this section entirely.

We clarified that this would be a database driven project. Other methods could be used such as JSON files but there seemed no need to deviate from the tried and true for this application. The user information is stored here for login purposes. The schedule is accessible to view via the database. This is where available time off is recorded as well as updated when requests are made and approved. Each login is timestamped and recorded in the database as well as any edits to these final timestamps along with the identity of the editor. This last portion allows for later audits of use and compilations of trends over time. Perhaps these aspects can contribute to the culture of workplace oversight if that is revisited specifically but it does offer insight into trends for use as a tool in this process if not.

The use of software outside of the system is mentioned. We had previously mentioned the creation of reports but have settled on the use of MS Excel compatible formatting for use in this process. This software is likely the most ubiquitous in professional settings and will provide the greatest ability for immediate adoption of the system. This may also allow for a bit more functionality by utilizing a particular external software application rather than choosing a more universal format as the more advanced features of MS Excel can be accessed directly.

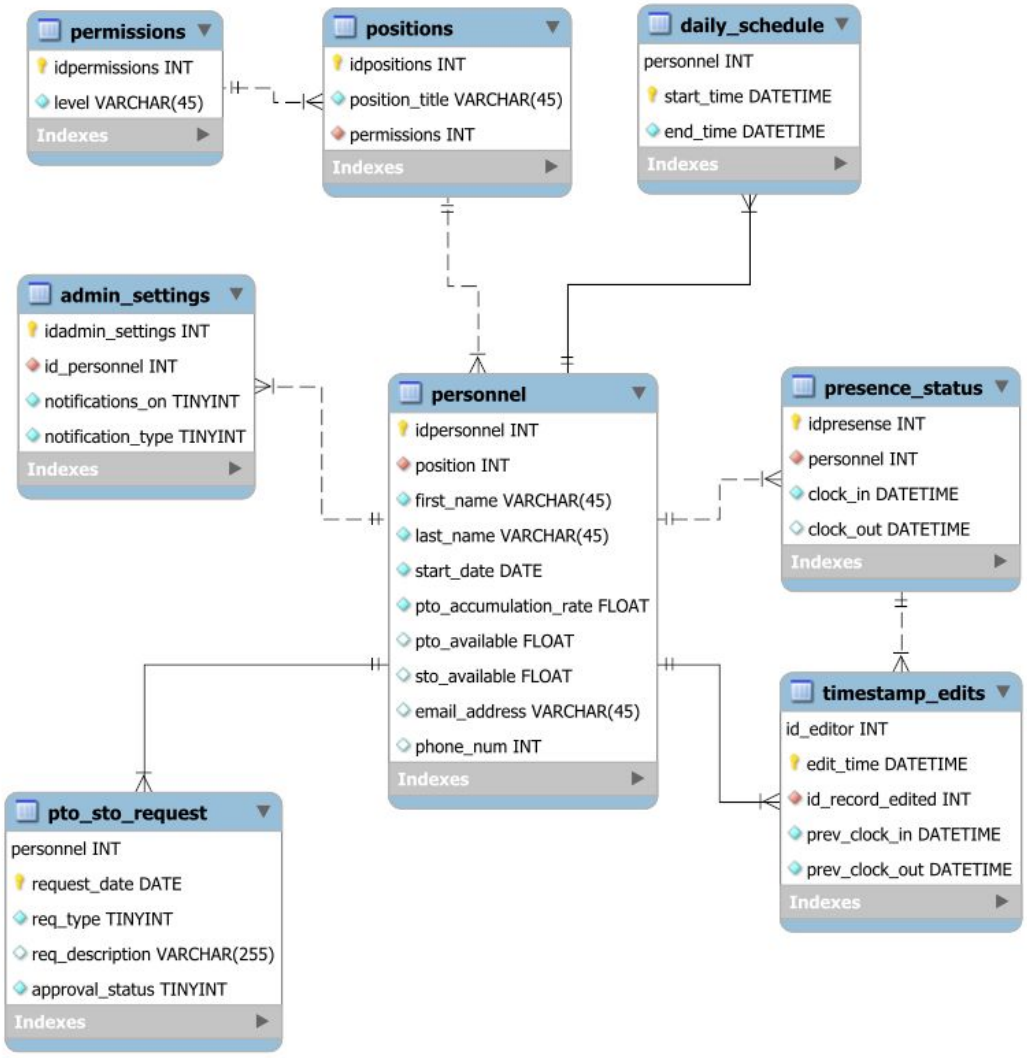
We have added in the specifications to check the IP address of any login. This will basically provide for the differentiation of a correct clock in as opposed to an employee simply attempting to review the current schedule or their own past attendance record.

Team Contributions (HW 2):

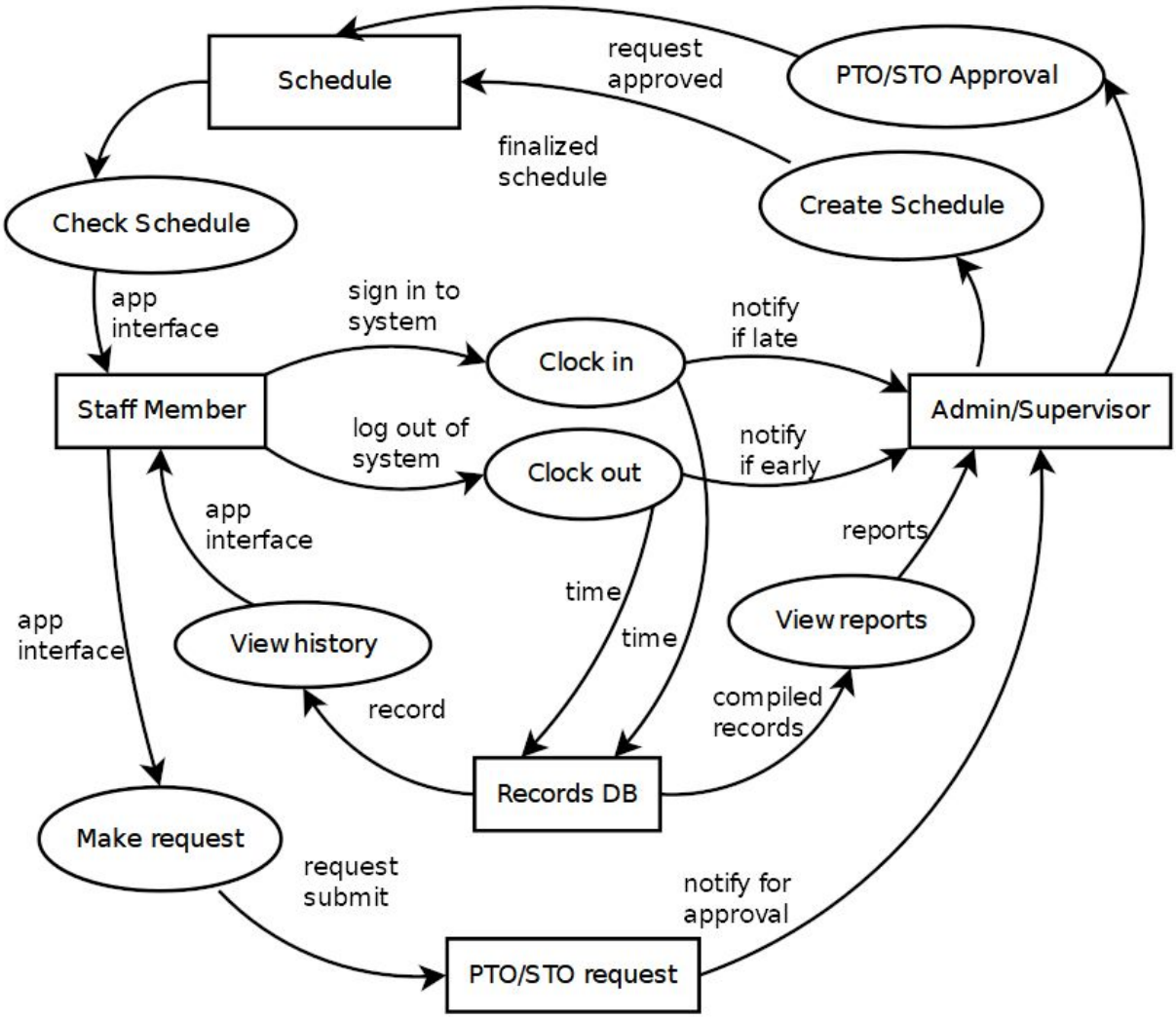
All team members met Thursday, 10/24/19, and discussed project goals and scope. Specific to Homework 2, work was divided across different tasks:

- Alex Kolstad: Revised Requirements definition (functional/non-functional requirements) and revised Requirements specification (functional/non-functional requirements)
- Zachary Wetekamm: Paper prototypes - Admin Dashboard and Employee Dashboard
- Sunghoon Cho: Paper prototypes - Log in Screen and all Notifications
- Michael Czapary: Request PTO/STO screen and Schedule Views
- Christopher Elliott: Requirements changes summary and revised diagrams

Revised Entity Relationship Diagram (HW 2):



Revised Data Flow Diagram (HW 2):



Use Case 1: Employee Timestamp

Actors:

- Non-admin Employees
- Admin Employees

Preconditions:

- User is onsite at company office.
- User is registered as an authorized company user.
- User is registered to the application system.
- User logs into the system using username and password credentials.

Postconditions:

- User has successfully recorded timestamp for either:
    - Clock In/Out
    - Break
    - Lunch
- User receives a success or failure notification for any timestamp query.
- Appropriate Admins are notified if an employee is tardy or absent.
    - Frequency of notifications are determined by user preferences.
- User can access a view of their recorded timestamps from a customizable date range.
    - Admins can edit recorded timestamps.
        - Any edited timestamp is associated with the admin that edits that timestamp.
        - Admins cannot edit their own timestamps.
    - Non-admins cannot edit recorded timestamps.

Flow of Events:

- User is onsite and logs into an authorized company computer.
- User logs into the application with username and password credentials.
- User records a timestamp for the appropriate condition:
    - Clock In/Out
    - Break
    - Lunch
- If user is Admin, user may edit an existing timestamp for an employee.
- User receives confirmation that timestamp was recorded.

Use Case 2: Notification of Tardy or Absent Employees

Actors:

- Non-admin Employees
- Admin Employees

Preconditions:

- A registered and authorized user logs into the system using appropriate credentials.
- User has successfully records appropriate timestamps
- System uses employee time schedule and check-in status to determine employee presence
- Admin employees have opted to receive notification for tardy or absent employees

Postconditions:

- Admin employees are notified of tardy or absent employees based on their settings
    - Notification type and frequency can be set based on preference
- Tardy or absent employees are notified of their status at check-in or at set times (e.g., 10 minutes past starting time) based on preferences set by respective admin employees
- Admin/supervisor or employee can choose to document reason for tardiness or absence
- Admin employees can check whether tardiness or absence was pre-approved or notified in advance
- Separate notifications for persistent tardiness can be sent (conditions for persistence determined by admin employee, e.g., based on repeated reasons)

Flow of Events:

- System records timestamp by employee
- System sends employee and admin notification when timestamp is recorded based on previously set preference
    - Notifications can be sent only at late check-in
    - Notifications sent after set time period past beginning of shift
- User or admin/supervisor can choose to document reason for tardiness or absence
- Based on preferences, system sends notification to admin/supervisor for persistent tardiness

Use Case 3: Request Paid Time Off

Actors:

- All employees up through management should have access to this use case

Preconditions:

- The employee is registered in the system
- The employee is logged into the system (from anywhere)
- The employee has sufficient vacation or sick days remaining to cover request
- The employee has given sufficient time in advance of request so that their shifts can be covered
    - If a request comes after the predetermined advanced notice time then a notification should be sent to the on duty supervisor for approval

Postconditions:

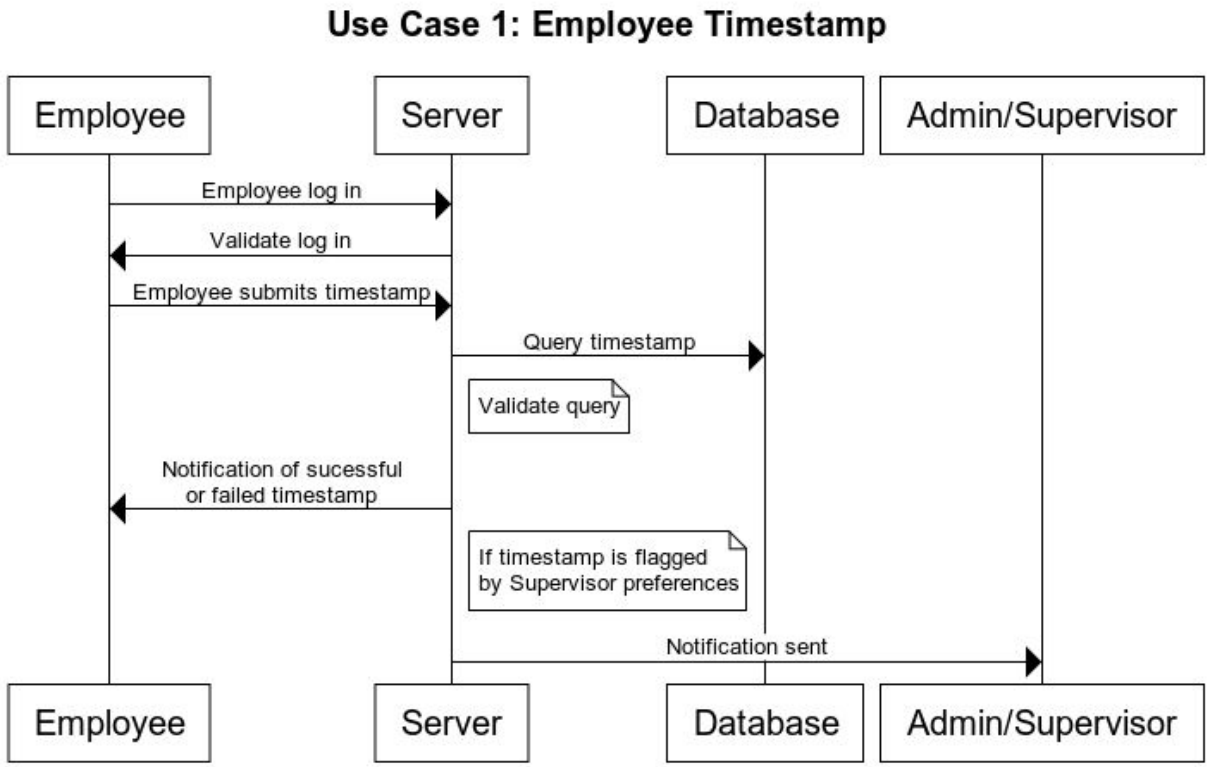
- *If the request is granted then the scheduler is notified of the days and time the employee is unavailable
- Remaining sick/vacation/personal time off is reduced by the amount being used
- An approval message is sent to the employee so they know the request was granted.
- *If the request come later then the predetermined advanced notice time then the employee will get a message informing them that their request is being reviewed by a supervisor. After if is accepted or denied it will follow the regular postcondition route.
- *If a request is denied a message will be sent to the employee informing them why (not enough advanced notice, not enough remaining vacation days etc.) The employee will then be given the option to make a special request to their supervisor if they have special circumstances to be considered.

Flow of Events:

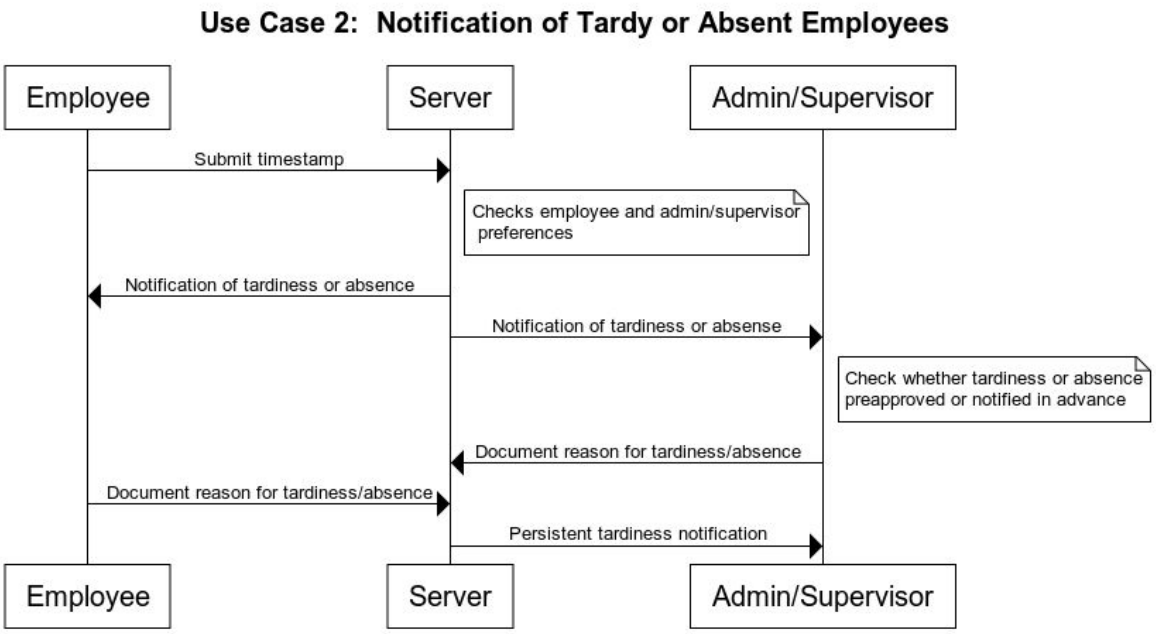
- Employee logs into system with their username and password from anywhere
- User fills out and submits request form
- *If necessary, message is immediately sent to supervisor for review.
- A notification is sent the employee either confirming request was granted, informing that it is being reviewed, or informing the employee why it was denied and allowing them the option to contact an active supervisor if they have special circumstances.
- *if the request was granted then the employee's remaining sick days, vacation days or PTO is immediately reduced.
- *if the request was granted a notification is sent to scheduling so that they know when the employee will be unavailable

Message Sequence Charts:

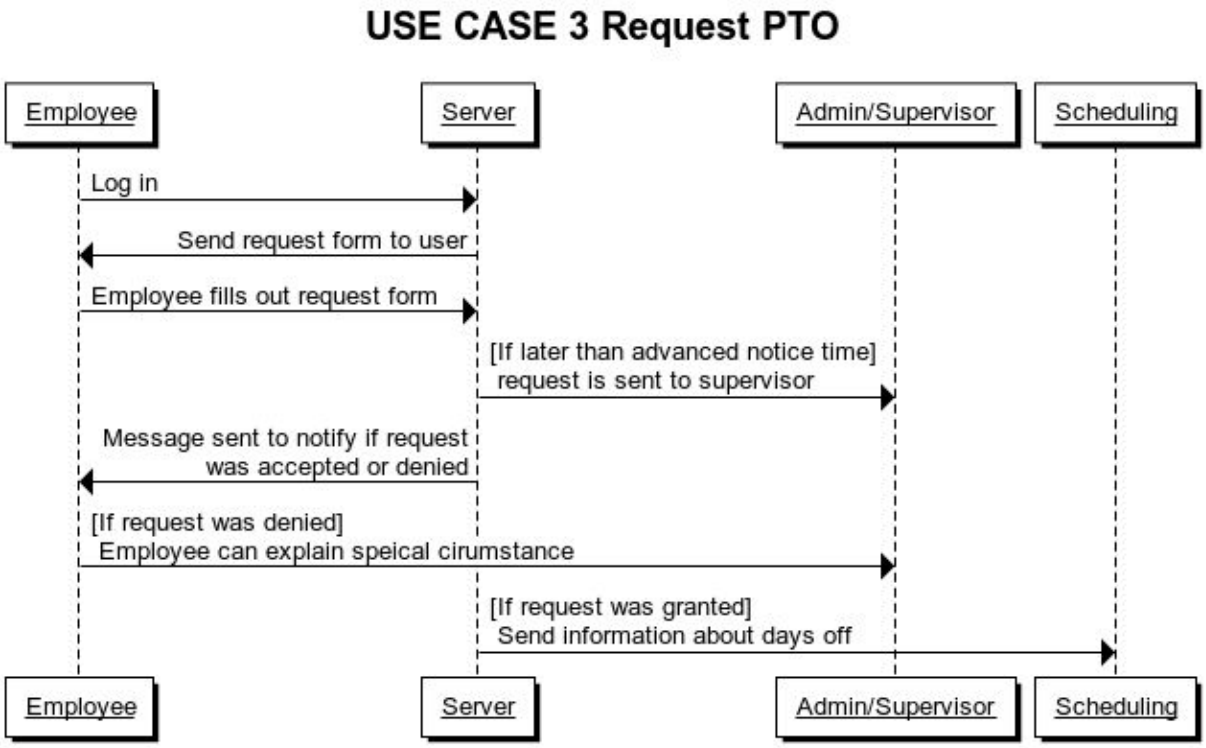
Use Case 1:



Use Case 2:


Use Case 2: Notification of Tardy or Absent Employees
Employee
Server
Admin/Supervisor
Submit timestamp
Checks employee and admin/supervisor preferences
Notification of tardiness or absence
Notification of tardiness or absense
Check whether tardiness or absence preapproved or notified in advance
Document reason for tardiness/absence
Document reason for tardiness/absence
Persistent tardiness notification
Employee
Server
Admin/Supervisor

Use Case 3:


USE CASE 3 Request PTO
Employee
Server
Admin/Supervisor
Scheduling
Log in
Send request form to user
Employee fills out request form
[If later than advanced notice time]
request is sent to supervisor
Message sent to notify if request
was accepted or denied
[If request was denied]
Employee can explain speical cirumstance
[If request was granted]
Send information about days off
Employee
Server
Admin/Supervisor
Scheduling