Java Help Desk Layout Schematic Version 1.0

Everything is subject to change

Front Page:

- Top:
    - Title like UNO Help Desk assistant
    - UNO icon
    - Navigation bar:
        - Instructor sign-up
            - Brings the user to a sign-in page that collects data about the user and creates an account for them;
            - Takes in the instructors name, uno ID, college level in text fields
            - Check accuracy of field inputs and only except correct inputs
            - Create username and password
                - Encryption and security here
        - Instructor log-in
            - Brings the user to a log-in page
            - Send user to instructor/admin page
- Center:
    - Info on helpdesk
        - Classes that the help desk user can help with
        - Hours of operation
        - Help desk worker's schedule
        - Other info as needed
- Bottom:
    - Student sign-in button
        - Pop-up window
            - Asks name: text field
            - Asks UNO id: text field
            - Asks problem: dropdown menu
                - Homework
                - Lab
                - Technical (bash problem, installation..ect)
                - Lecture renforcement
                - Other
            - Asks which class: dropdown menu
                - 1583: Java 1 lecture
                - 1581: Java 1 lab

  - 2120: Java 2 lecture
  - 2121: Java 2 lab
  - 2125: Data Structures
  - Assembly (forgot the number)
  - 2467 Systems
  - Other
- Insert data into database
- Add to queue

- Queue (This can be an optional feature)
  - Dynamic
  - Lists the top N number of of students waiting for the help desk
  - Students get removed from the queue when helped

Instructor Page:

- Top:
  - Navigation bar
    - Link to front page
    - Log-out button
      - Logs user out and returns to the front page
  - Start Session Button
    - Large, easy to see (Green maybe)
      - Starts timestamp for session and insert it into database
      - Insert student key into database
        - Pull data using queue
      - Insert day, month, year, semester
  - End Session Button
    - Large, easy to see (Red maybe)
      - Insert end timestamp into database
- Center:
  - Session Counter (maybe)
- Bottom
  - Buttons/menu (radio, traditional, dropdown menu..ect)
    - Monitor the successfulness of the tutoring session per the instruct
      - Felt like the student has a much better understands
      - Felt like the not much impact was made
      - Felt like the student has little or no understanding
    - Optional comment field
      - Added info from the tutor

- Submit button
  - Submits all gathered data to the database
  - Can only be clicked if the session is over
  - A new session cannot be started until the old session’s info has been submitted

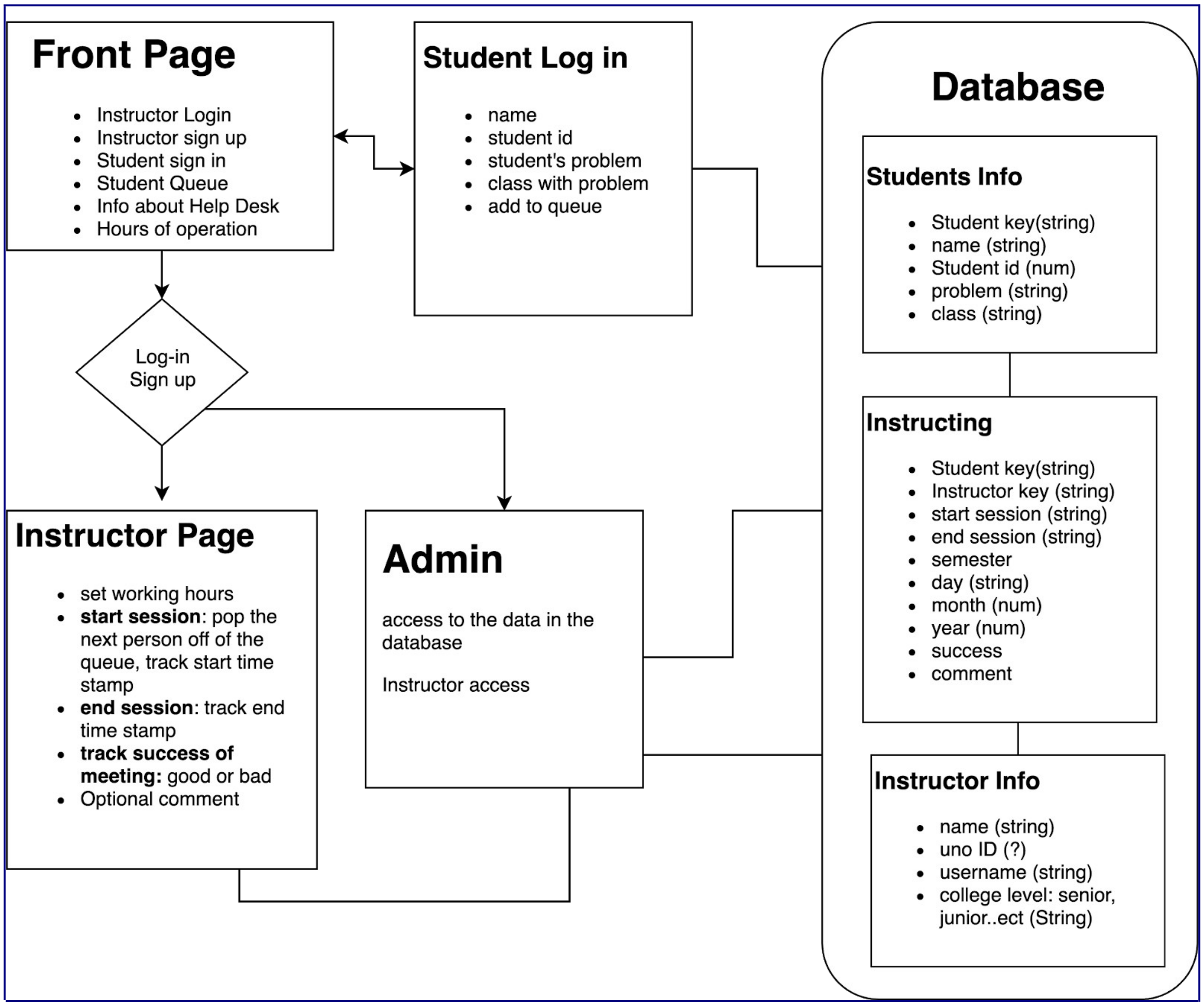
Admin Page:

- Links to instructor page
  - Admin has the option to tutor
- Option to view the data in the Database
  - View in table format
    - View based on the relation’s attributes
      - Semester, class, problem type,

Other notes:

- Keep a record of wait times
  - Start when student logs on the queue
  - End when mentor starts session with the student
- Include 2467 in dropdown menus
- Should not remove person from the queue until the mentor submits their input

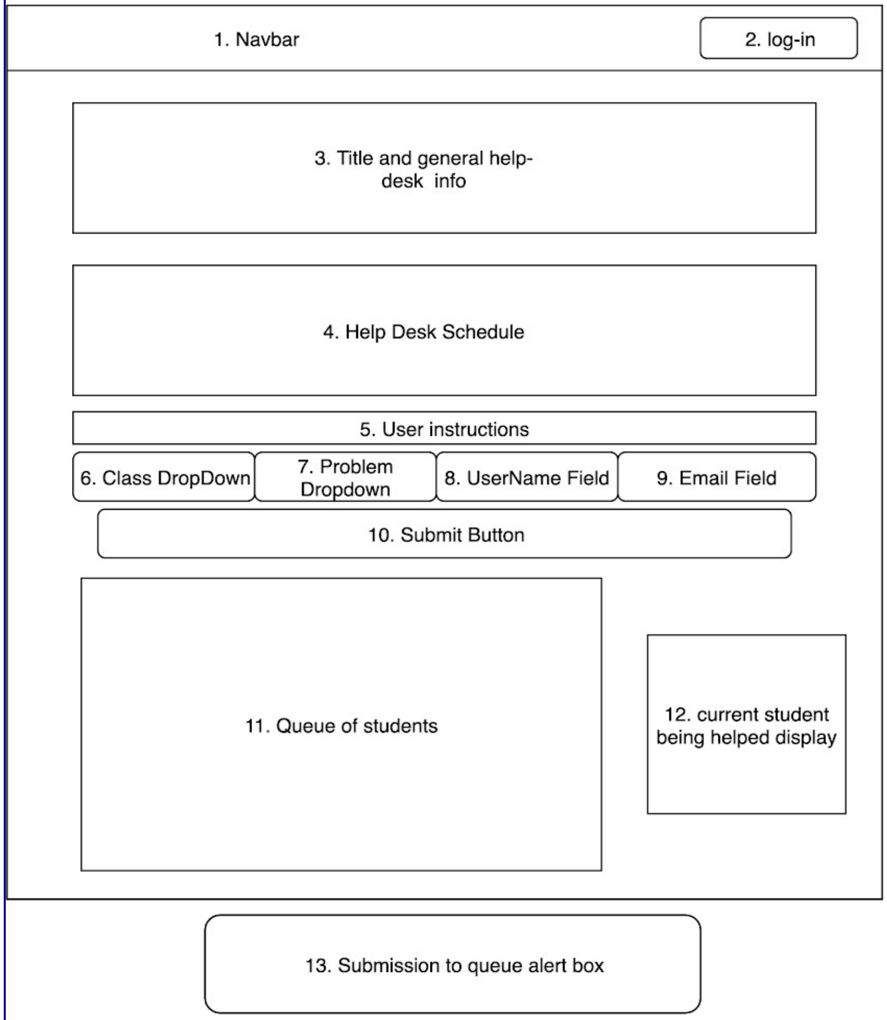
Chart



Future Features:

- Dynamic update of the queue to the client
    - Student alert when it is their turn
        - Browser supported notifications
- **React and redux**

Front Page Layout and component Listing

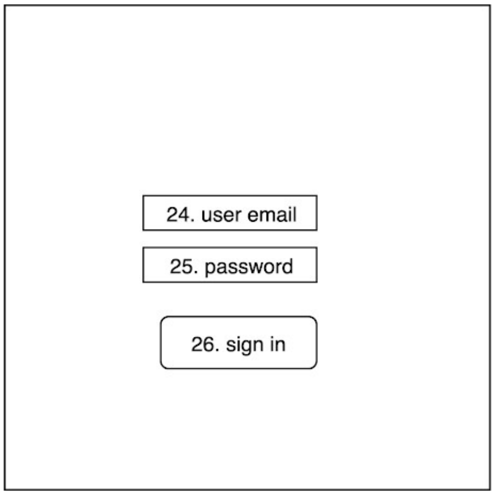


Component Descriptions

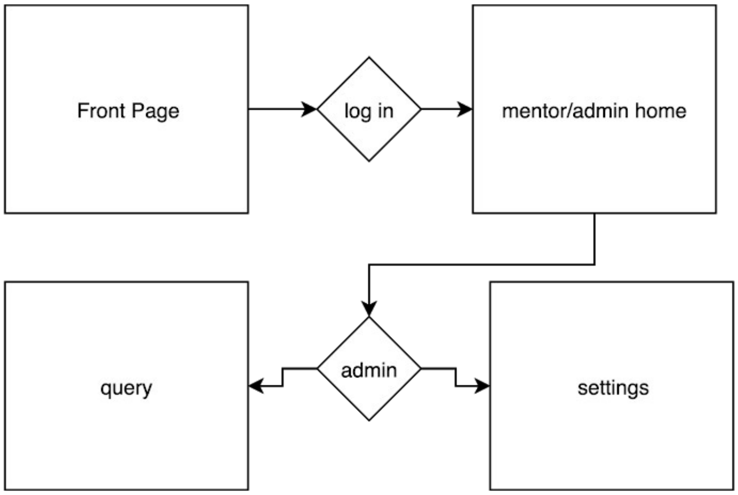
1. The nave bar component should be reused for every page. what is displayed on it should depend on what page is showing and who is logged in.
2. Log-in button is used for mentors or admins only. The text "log-in" should display on the front page only. when a person is logged in the button should read "log out". It should function according to the displayed text
3. Static info section should contain general information about the help desk. (Will not change often)
4. The help desk schedule should be set by an admin. It must contain the current semester's help desk workers name and hours of operation
5. User instructions block should briefly describe what this website is for and how to be added to the queue.
6. Class drop down menus should list the course that the help-desk services. This is the course that the student requires help in. The courses are as followed: 1581, 1583, 2120, 2121, 2125, 2460,2467.

7. Problem drop down menu should contain the type of problem that the student is having issues with. The Problem listing should be as followed: Lecture, Homework, lab, technical (bash,software),other
8. The user name is a text field that the user inputs there uno user name.
9. the email field is a text field that the user inputs there uno user email
10. The submit button inserts all of the data from components 6 through 9 into the database. It should be disabled until the user has provided all of the information from components 6 through 9. The submission should be rejected if the user info doesn't match the current enrolled C.S. students (no database insertions). It should also reject any submission that already exists on the queue. With a successful submission, the students info should be added to the queue, the database should be updated and the submission alert box should be deployed.
11. The queue component should display who is in the queue in the numbered order that each will be helped. The queue components content should depend on which page is displaying it. For the front page, only the number order and the user's username should populate each field. In the admin/mentor page, the user's name, class with the issue and issue type should populate the queue field. As students are added to the queue, they should populate from the bottom. When students are removed from the queue they should pop of from the top. The right side of each field should have an option to instantly remove the field from the queue. This is reserved for the admin only
12. Currently helped window should display who is being helped, by whom and for how long(perhaps a starting time)
13. Submission alert box should briefly pop up when the user has successfully been added to the queue and inform them of the success.

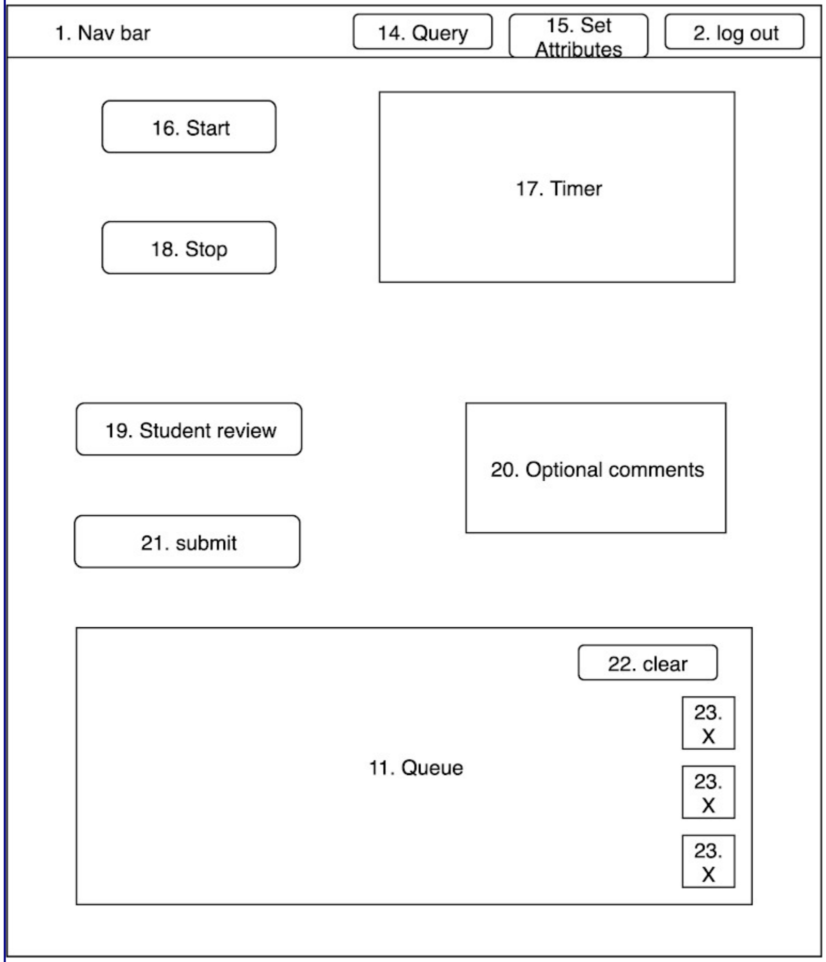
mentor/admin sign in page



Page Flow



Admin/Mentor page



14. Query button component should only populate when an admin is logged in. Used to navigate to the Query page

15. Set attribute button should only populate when an admin is logged in. Used to navigate to the setting page

16. Start button starts the help desk session. Initially is should be active. It will trigger the timer for the session and pops off the next student from the queue. Once pressed it should remain disabled until the mentor/admin presses the submit button.

17. The timer counts down from the time limit set in settings by the admin. When it reaches zero the session automatically ends.

18. stop button ends the time session.

19. Student review dropdown menu contains good, neutral and bad: student understanding is good, student understanding is so-so, student understanding is bad

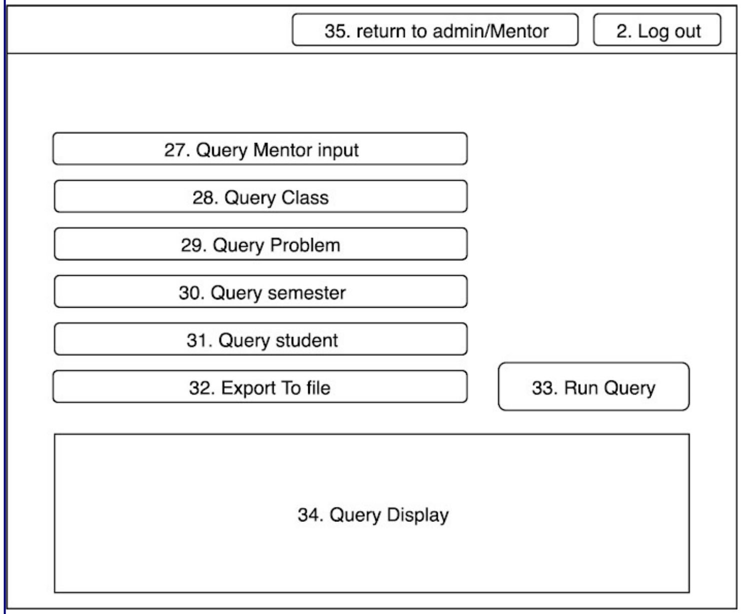
20. Optional comment text field if where the mentor can can write a comment on the session just given.

21. submit button should only be active when the session is over and the mentor/admin has selected a choice from component 19. The submission should insert the data from component 19 and 20 into the database

22. Clear button should only be displayed when an admin is logged in. Function should clear the entire queue

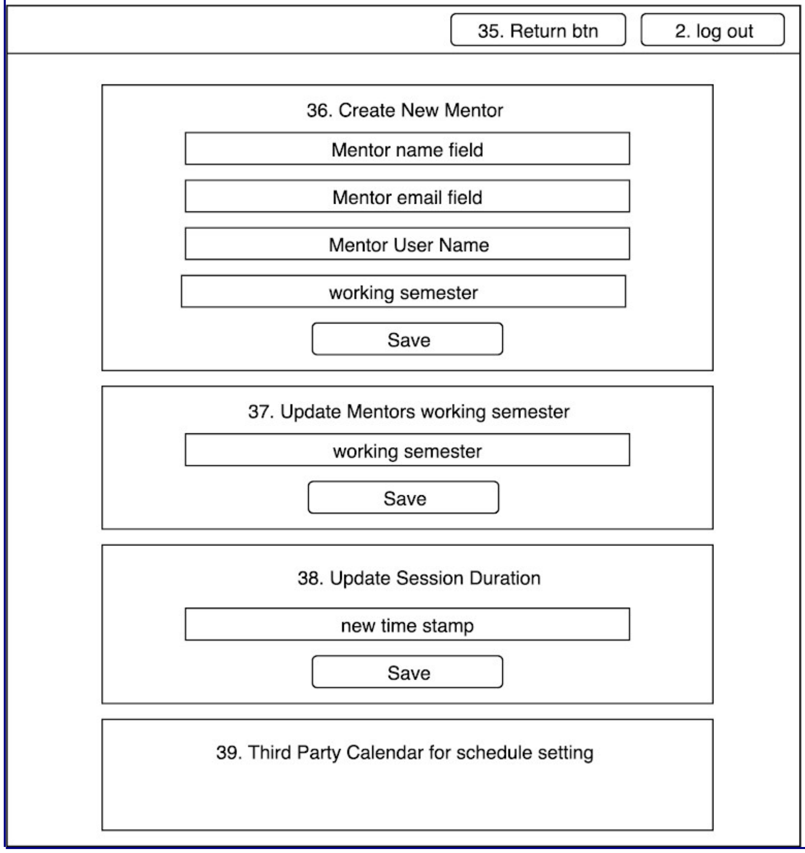
23. x button should allow the mentor to delete a row of the queue. (maybe just for admin)

Query Page



27. Query by mentor input text field should take in a mentor's input for query argument
28. Query by class drop down menu should select a class for query argument
29. Query by problem drop down menu should select a problem type for query argument
30. Query by semester text field should take in a semester (fall_2019) for query argument
31. Query by student text field should take in a student's email for query argument
32. Export ran query into a file
33. Run query button should generate a query based on the arguments given by the parameters
34. Query display should display the query in rows and columns
35. return button should return the admin back to the admin/mentor home page

Admin Settings Page



36. Create New Mentor field should take in the appropriate data for a mentor and insert a new entry into the database. The data should include, but not limited to name, email, user name and working semester. There should be a save button that is responsible for inserting this data into the database
37. Mentor semester update block should take in a new semester of work for a mentor that already has an entry in the database. It should over write the previous working semester with the new entry. There should be a save button that executes the process
38. Update session time limit block should take in a new time stamp in a text field. There should be a save button that sets this time as the new session duration
39. Set Schedule of mentor via 3rd party application