Team Name: BrevityFinesse
Members: Henry Zhang, Julian Atkin, Yuki Chen, Billy Wong

Roles:
Henry: Project manager
Julian: Javascript / Front-end
Yuki: Database management
Billy: Back-end management

Mission of project:

Ticketing system for the tech services department. Teachers call Angela for some issue. Angela makes a log and contacts Sydney or Joe. It would be nice to have a system that teachers can fill out a request online and Sydney and Joe gets a message on their cell phone. Or they still call Angela and she logs in the request in the same system and it automatically contacts Sydney/Joe. Sydney and Joe should have a feature to respond to teachers (tasked completed, time, schedule a time frame like the cable guy, deferred until tomorrow, etc).

Features:

- Four levels of accounts -- superadmin, admin, teacher, tech
  - Superadmin (only one)
    - All admin privileges
    - Ability to grant/revoke admin privileges
  - Admin
    - Complete access to teacher and tech functionality
    - Ability to create accounts (techs and teachers)
    - Can toggle guest functionality to prevent harmful activity
  - Teachers can create requests, and see past & pending requests and their status
  - Techs can accept requests, rate each request's urgency, update on the progress
  - Guests can submit requests when guest functionality is on
- Email/text techs when new request arrives
- Email/text teachers when request is fulfilled or when status changes

Database Schema (SQL):

Requests: teacher side; tech side

INT primary key	TEXT teacher_name	TEXT date of request	INT room_num	TEXT req_subject	TEXT req_body	TEXT tech_name	INT urgency	INT status

Notes:
- Req_subject is the type of trouble to be chosen from a dropdown by the requester
- Req_body is the comment that the requester has to make
- Teacher_name will be filled automatically if logged in with teacher account

Accounts

INT primary key	TEXT username	TEXT email	TEXT password	TEXT salt	TEXT type	TEXT phone_num

Notes:
- Salt is the unique hash provided to each user's password
- 4 types of accounts: superadmin, admin, tech, teacher

Replies (is it necessary?):

INT Primary key	TEXT reply

Component Map:
Python
- auth_manager.py
- db_manager.py
- request_manager.py
- message.py
- admin_tasks.py

JavaScript
- Form validation, stylish responsiveness (AJAX to load content instead of refreshing the page)

HTML
- Guest Page to make a request if enabled
- Login Pages (4 different types of accounts)
- Superadmin:

- All of the pages that admins can view
- Grant/revoke admin privileges
- <u>Admin</u>:
    - Add new account
    - View requests (modify request information)
    - Make requests
    - Turn on/off guest functionality
- <u>Teacher</u>:
    - Make request
    - Pending requests
    - Past requests
- <u>Tech</u>:
    - New requests (not accepted yet)
    - Pending requests (responded to, order of urgency)
    - Past requests

Site Map:
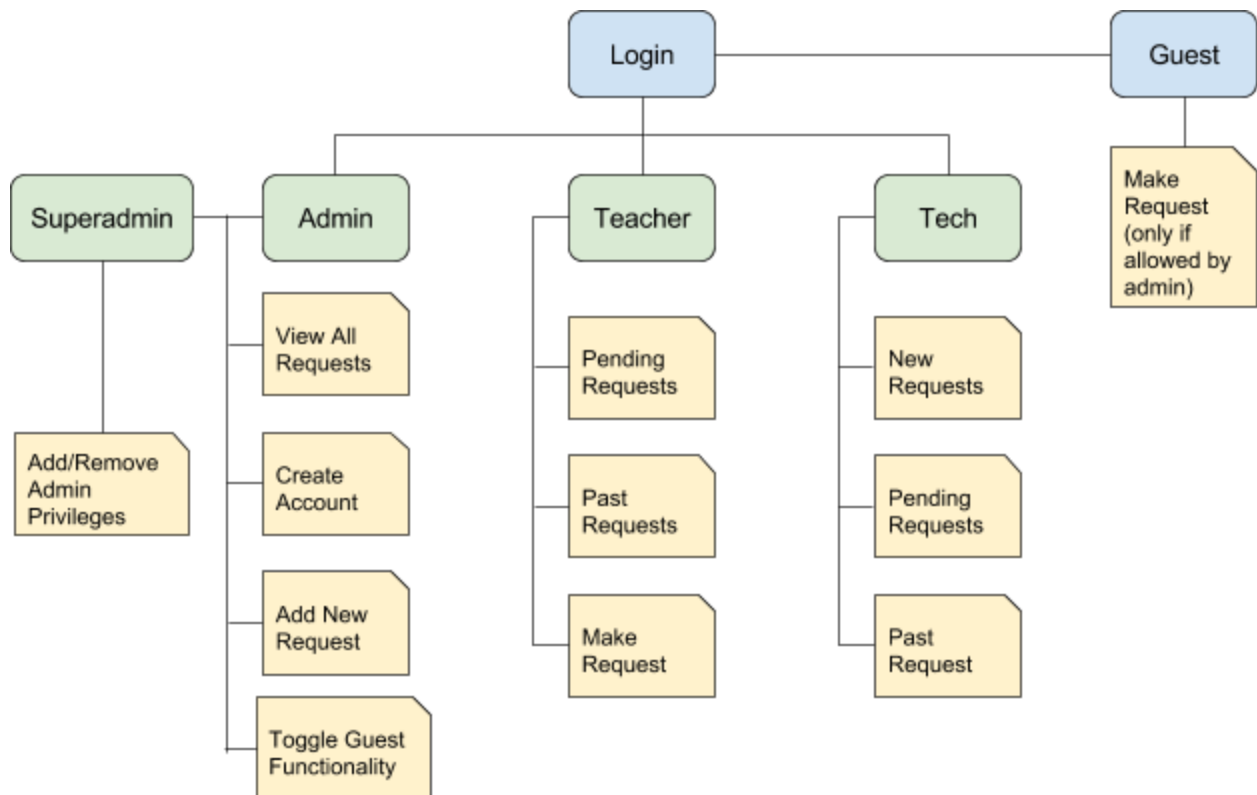
STUY issue tracking system

Smart bard machine broke
posted 4/20/69 by Bill Boericke

yes it did broke ithe screem is not work no mattter press

Priority:

Top Priority:
- Basic ticketing support
- Reply system for continuous discussion
- Secure accounts, shouldn't allow to inconvenience techs
- Workable, easy to understand UI -- fanciness should be a goal but not before the program is functional

Med. Priority:
- SMS/email notification
- Timeframe indicator
- Urgency
- Slick stuff