*ABC*s to fight bullying

https://github.com/EthanZeigler/ABC

Issue

Bullying is a massive national problem. The idea of "snitches get stitches" is problematic for educators and administration. The idea that others will figure out who said something discorages them from getting help, and new laws such as HIB cannot be effective if people aren't willing to speak out about problems.

Solution (Option 2)

Creating a customizable *webapp* for reporting bullying incidents anonomyously (or named if one wants to) that allows educators to have a better idea of bullying incidents going on at their institution, even if a student does not want them to act on a specific incident.

I will call this system the **ABC**, or the **Anti-Bullying Connection**.

This allows administrators to be more aware of bullying and bias incidents in a technological age. We are not aware of an open implementation of such a system.

Implementation

Tools

- RoR
- Linux
- HTML/CSS
- Typescript/Javascript if necessary
- Bootstrap

Algorithms

- Event clustering, detecting reports that appear to be of the same type and highlighting them as a more serious situation
- Spam detection, detecting when a single user is submitting a lot of reports that are the same or vastly different, but not in between.
- Login system with proper security measures

Data Structures

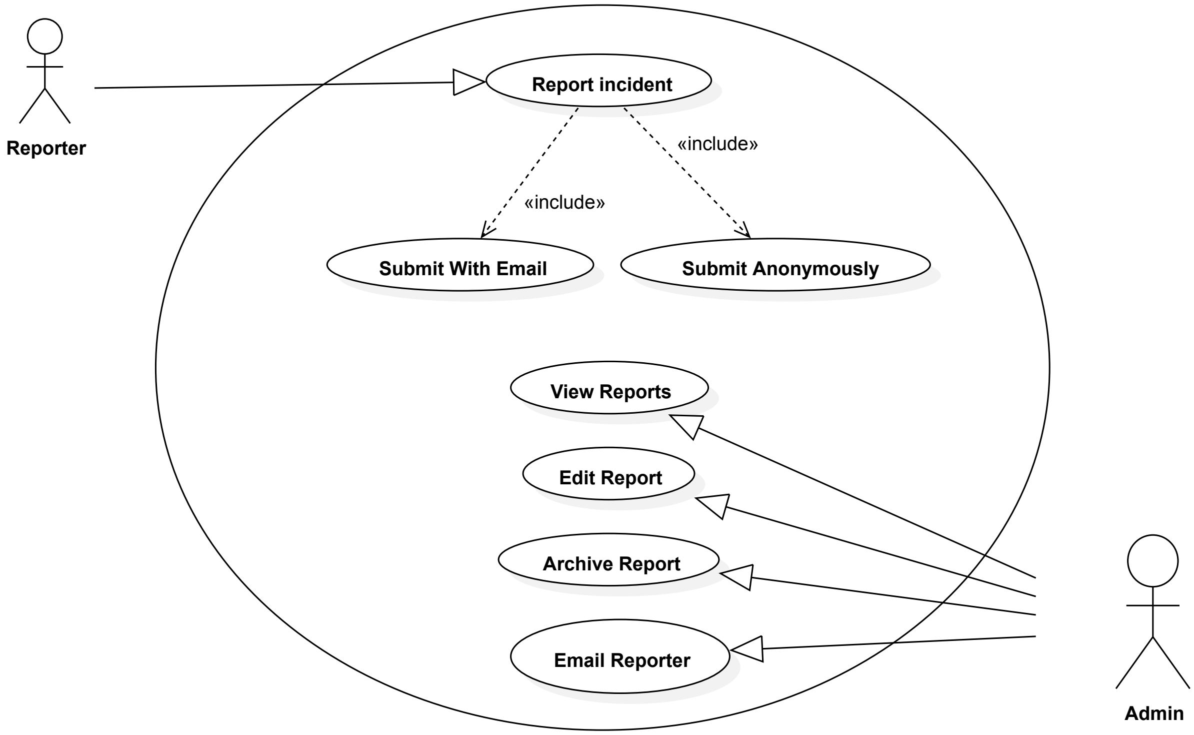
- Database (Postgres) for storage of reports

- Correlations done by field similarity (so list-like implementations, possibly priority list)

Software Engineering Concepts

- MVC model
- DBMS
- Security between student and admin sides

Use case diagram



Timeline

- Mid March
    - Able to submit reports
    - No distinguishing between sides just yet
    - Basic front-end design
    - Rea through the ruby on rails book and gain a basic understanding
    - Look at existing rails applications to
- End of March
    - Distinguishing between students and administration. Studetns unable to see other reports
    - Basic back-end design
    - Database schema design/implementation complete
    - Read through the ruby on rails book and gain an intermediate (enough to complete the project) understanding.
        - From now on, learning about the platform is on a need-case basis.

- Mid April
    - Fully designed interface (template-wise)
    - All information stored into database
- End of April
    - Partially functional organization of reports
    - Additional notes about reports can be attached
    - Reports can be edited and deleted
- Mid May
    - Implement basic spam detection
    - Bug fixes and overall system rubustness testing

Additional implementation details

The goal is to be able to have the system be a sort of 'ticket manager' for these reports. Notes will be added in a ongoing basis as things are handled. Reports can be sent to the reporter if they decided to not be anonymous.

If that can be finished quickly...

Enhance to have student logins as well to manage status of things they have reported.