**Front end fixes**

Steps:

* git clone https://github.com/orionkepler/augur
* cd augur/
* git checkout frontend-dev
* cd frontend/

Branches:

* frontend-dev (Main frontend development branch)

Rewriting the install script:

1. Don't confuse users when installing Augur.
2. The user should be able to clearly differentiate between required inputs
3. The default options chosen must be obvious
4. As users expect, it should conform to Unix standards
5. No files are needed to illustrate the difference between inputs
6. The user should be able to distinguish between each request
   * + 1. For example, SQL credentials
   1. There should have more clear explanation to explain which choice is for what function and what the argument or condition they are expected.

Optional (If there is enough time to implement):

Frontend Components (Requirements):

1. Home Page / Show All Repo Groups
2. Display all Repo groups in the database.
3. Each group should be clickable
4. Display information about each group
5. Repos List
6. Drop down menu
7. Each repo groups name
8. All
9. Display list of repos
10. Each repo should be clickable
11. Display information about each repo
12. Metrics For A Repo
13. Display graphs for different metrics
14. Total Forks
15. Total Contributors
16. New Issues / week
17. New Contributors / week
18. Commits / week
19. Reviews / week
20. Header Bar
21. Title
22. Repo Groups Button
23. Repos Button
24. Search Bar (angular material)
25. Dropdown for autocomplete

Goals:

Optional Goals (If there is enough time to implement):

* Metrics For A Repo
* Add routing to the metrics page. URL Path: /metrics
* Display charts on the page for presentation
* Home Page / Show All Repo Groups
* Routing to the other pages (/pageName)
* There is a basic container for the repo group
* Repos List
* Design the user interface of the repo list component
* Header Bar
* Design the header bar for repo so that it can be easy for users to search

**Currently Finished:**

* Installation script rewrite
* Make the input more detailed
* Clearly differentiate between similar-looking inputs
* Provide more feedback on possible options
* Reduce the amount of manual configuration post-install
* Clearly define the back-end/front-end running commands
* Clearly define stop conditions
* Make the default decisions Unix compliant
* Default options are CAPITALIZED
  + - * + Current options are reversed