CSCI3055 Final Project

Steven Avery

Git Repository:

- https://github.com/stevenAvery/CSCI3055_Final_Project.git

Try Online:

- http://159.203.45.249:8080/chat/roomName

Problem Statement:

- Modern communication software provide barriers to entry, and unintuitive functionality that provide better provide better long term experience, but can be cumbersome for short-term use.

Problem:

- For example:
- managing friends lists
- Slack teams

Languages Used:

Clojure:

- *Compojure*
- *Ring*
- *Http-kit*
- *Hiccup*

Clojurescript:

- *Domina*

Languages Used:

Clojure:

- *Compojure*: generates Ring request/response handlers with added routing capabilities.
- *Ring*: handles requests and response for web development.
- *Http-kit*: a minimal http server.
- *Hiccup*: allows representation of HTML in Clojure.

Clojurescript:

- *Domina*: provides DOM manipulation through the Google Closure library.

Alternatives:

Clojure:

- many alternative for server development, including:
  - Python with Django
  - Ruby on Rails
  - Server-Side Java
  - Node.js

Clojurescript:

- javascript

Alternatives:

Clojure:

- *Compojure*: Stray
- *Ring*: Aleph
- *Http-kit*: Webbit
- *Hiccup*: Enlive

Clojurescript:

- *Domina*: Dommy, Reagent

Build Tools:

Leiningen:

- Leiningen automatically builds clojure projects

```
lein cljsbuild once
lein run
```

Code Walk:

type systems type inference lexical scoping and closure functions as data, coroutines list comprehension