6.170 Project 2 Phase 2 Design Document

Michael Wee – October 7, 2013

**Feature descriptions**

* User accounts and associated notes
* Sessions and cookies for persistent user login
* Create, show, edit, and delete notes
* Restricted access to user’s own notes
* Note feed on each user’s profile and home page sorted by last creation time
* View all users on the system
* Change user account settings
* Option to import Gravatar account information
* Secure password encryption

**Design Challenges**

* Create notes and users
  + Both have controllers, models, and views. Database schemas are created with generate and updated as necessary with migrations.
* Edit notes and users
  + Done similarly with updates, shows (for notes), in the with controller actions updating the model, then routed to the view
* Associate users and notes
  + Has\_many and belongs\_to relationship between users and notes in the rails models classes as well as a user\_id field to associate them
* Sessions and persistent login
  + Implemented with a sessions controller and helper that keeps track of the current user after signin and persists it in a cookie.
  + Secured with SecureRandom module to generate cookie encryption
  + Take advantage of has\_many and belongs\_to relationships as well as current\_user implementation to retrieve associated notes
* Secure logins
  + Uses has\_secure\_password module in User model
* Display repetitive HTML elements over multiple pages
  + Options:
    - Copy and paste duplicate code in different views
    - Use partials to render and stitch together different page elements
  + Solution:
    - Use partials to piece together HTML pieces with embedded Ruby code to avoid repetition
* Resolve and generate dynamic link structure that grows with the number of posts and users
  + Use rails routing and the RESTful design paradigm as an intuitive and dynamic link structure
* Restrict user to only view own notes
  + In users controller, make sure before any actions are taken that the logged in user is viewing the requested users’ notes
  + In notes controller make sure that before showing the note, make sure the signed in user is the note’s creator