John Marks’ AngularJS Capstone Project

# Project Name: SomeCode

Okay, so it’s a cheesy name, but it’s what stuck. If this project goes somewhere, the name will certainly be improved.

# Purpose: SomeCode is an online code snippet utility providing a clean intuitive user experience that helps coders quickly store, search, share, and retrieve their hard-learned coding tips and tricks.

# Feature Set:

Version 1 = **Green**

Version 2 = **Red**

1. Coders can **store away key** **chunks of code** for future use.
2. **Notes** pertaining to code snippets can be saved along with the code to provide further context and insight.
3. Snippets can be written and viewed in their **natural language syntax and formatting**.
4. Snippets can be saved away in **topic buckets** for easy access and organization.
5. For those that don’t like or need a hierarchical organization, snippets can be tagged with **hashtags**. The hashtags can then be used for searching and retrieval.
6. Snippets can be set to a “**public**” state so other coders can search and view and even “**Snip**” the snippet for their own use. User’s snippets can have a **different color coding** from public snippets created by other users.
7. Snippets can be **viewed in multiple sort orders**. They can be displayed in a custom order to allow multiple snippets to be grouped in a logical ordering that as a whole provide even more information.
8. Snippets can be **searched** with full-text database searches, using complex queries and globbing. Snippets can be searched according to their hashtags. Users can **search their own created snippets** (both public or private), or **search** **all public SomeCode snippets**. **Total public / private snippet counts** are shown in the search field.
9. Snippets can be **filtered** according to language type.
10. All public snippets are **searchable by non-logged in users**.
11. Snippets can be **viewed in multiple layouts**.
    1. The **horizontal layout** is useful for large code chunks or notes, where the note would be displayed at the full width of the page, and then the code associated with the note is displayed.
    2. The **vertical layout** is useful when multiple small chunks of code are to be displayed on the page along with their notes. This allows for a higher snippet density on the page, to provide more context for a quicker scan of desired information.
    3. The **text only layout** will display only the snippet titles. This is useful to get a much snippet density on the page as possible. Once the desired snippet is located, a quick hovering of the mouse over the snippet selector and the full snippet text of that single snippet will be shown.
12. Snippet data is stored in the cloud and is accessible through a **RESTful API**. Web and mobile clients can be written to provide full access to private and public snippet data, anywhere, anytime.

Enhancements:

1. Topics can be created in a hierarchical structure for precise organization. This would be reflected in the database to support more complex topic CRUD.
2. Users can share snippet topics. Snippet topics contain multiple snippets organized according to a theme or topic.
3. Create mobile apps for Android, iPhone/iPad.
4. Implement in No-SQL and test for search performance
5. Web snippet clipping. Recognize chunks of code in webpages and be able to quickly create a snippet in the user’s account. It would offer the same clean snippet-add functionality where the user could set the language, topic, public or private state, etc.
6. Add WebSocket functionality to snippet statistics. Update the number of public snippets asynchronous to any user page refreshed.

# Why SomeCode – another snippet tool?

1. To implement a somewhat complex project to gain the following skills:
   * Creating backend RESTful API
   * Implementing backend persistence, using an SQL DBs along with an ORM for OO DB access.
   * Backend framework: Python Flask
   * Python language
   * Front end development: HTML, CSS, intermediate JavaScript
   * Improved CSS page-layout abilities
   * Front end frameworks: AngularJS (Dependency Injection, URL Routing, MVC Data-binding, Templates, Extended HTML through Directives, Filters, Unit & E2E Testing), jQuery, Bootstrap. AngularJS really brings the structure and testability needed for larger projects. jQuery/JavaScript-only projects are very difficult to build and maintain. Bootstrap is helpful in providing nice default CSS style and useful class based components. Bootstrap UI provides many of the base Bootstrap components in AngularJS directives. So nice!
   * WebDev project build management: Gulp, NPM, Bower
   * Web testing: BDD, Jasmine, Karma, E2E Testing w/ Protractor on the Selenium Web-Driver.
2. There are many snippet tools out there. Some, like GIST, are web-base and provide many useful services. Many are native apps with local-only storage. Some are natively built for specific platforms. What is needed is a platform-independent application that can be supported by web, native, and mobile apps. The backend API should be a simple, clear RESTful API for easy client access anywhere, anytime.
3. Many of the snippet tools allow you to search and filter your snippets. The problem is that the tools seem to put the emphasis on the tool itself. When snippet searches are loaded, only a few snippets are displayed on the page. Users need more context, with the ability to view many snippets at once to provide for rapid viewing and access.
4. Many of the snippet tools provide the ability to create and edit snippets. The problem again is that the focus seems to be on the tool, and not on the job at hand, and that is creating and referencing snippets. The tools will pop up a large modal dialog encompassing the entire page, graying out the background and putting focus on creating the new snippet. Providing focus is good, but not at the expense of the other snippets on the page. Many times a snippet is somehow related to other snippets on the page, such that multiple snippets together provide a “topic” that needs to be explained through a number of snippets. If the ancillary snippets are greyed out, then it makes it difficult to create the new “follow-on” related snippet. We need to let the user focus on creating chunks of code that might be independent, but are most likely related to other code chunks. Allowing the entire page to stay in focus and stay in context is necessary.
5. It would be nice to be able to share snippets as well as snippet topics. This needs to be done in a Pinterest like fashion, where snippets can be organized into Snippet boards. The snippets in the “Snippet Board” could be from user-created snippets, or “Snipped” from other snippets created by other users. We need to ability to easily share and explore other’s efforts and quality code. Just like Google is so useful for finding help to many coding problems, the SomeCode snippet tool should be useful for sharing and finding answers to specific coding issues. Adding the ability to search, filter, and pin into boards or topics would be quite useful.
6. In general, as mentioned above, the focus on many of the current snippet tools seem to be the actual tool itself and not the users problem at hand. Users want a clean, fast tool for quickly creating, editing, searching, and organizing chunks of code. To accomplish this the UI needs to be focused on the user. The UI needs to get out of the way and not take up too much precious space. When searches are performed, the page needs to be filled with snippets that pertain to the search criteria. They need to be able to view the snippets in multiple layouts, each of which can be changed with the simple click or press of a layout button. The topic panel and search panel need to get out of the way to make room for and allow the user to focus on the code. Much design and creativity needs to go into creating a webpage UI/UX that meets these needs.
7. The concept of the “Snippet Board” as mentioned above would be a useful paradigm for code snippets. But rather than storing pictures, the user is storing and organizing code snippets. The snippet board is presented in the above design as a Topic. One of the weaknesses to Pinterest boards, as fascinating as they are for showing and displaying visually in one location certain topics of interested, is their inability to place the pins on the board where the user wants them placed. To organize the board in the order the user desires is messy and difficult. With this in mind, it is important that snippet boards, or topics, need to be able to be sorted in the order a user desires. A custom sort order is therefore important.