

# STAYING SHARP

*with  
AngularJS*



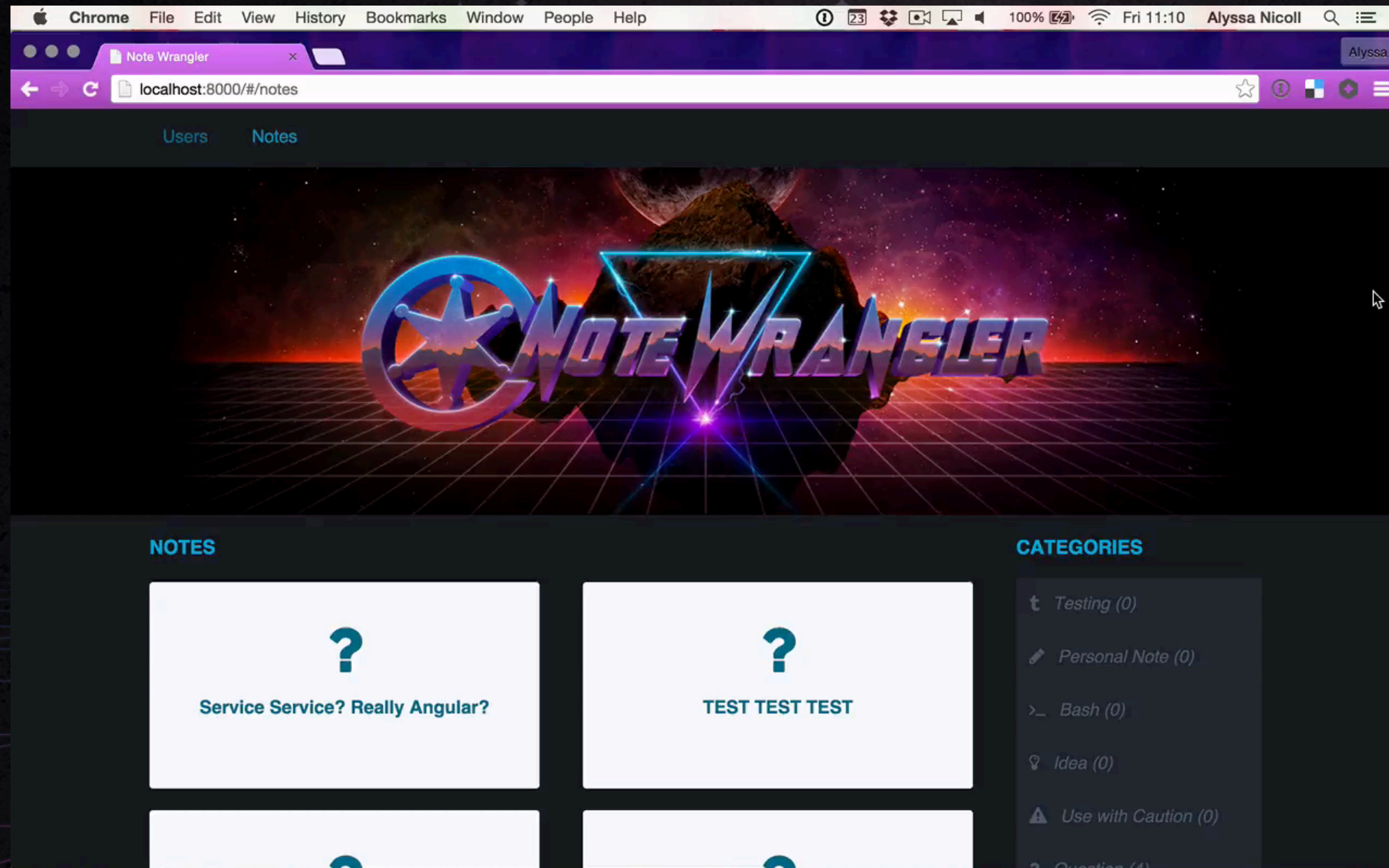
# Reusable Directives Level 4

## Filters Section 2



# Using a Filter to Finish the Job

Now that we have our `nwCategorySelect` directive, and active classes are working properly, we need to filter the displayed notes based on the `activeCategory`.





# Remember Filters?

A filter is an easy way to format our content before it gets printed in our template.

**date** : Formats date to a string based on the requested format.

**lowercase** : Converts string to lowercase.

**uppercase** : Converts string to uppercase.

**limitTo** : Creates a new array or string containing only a specified number of elements.

**orderBy** : Orders a specified array by the expression predicate (numerical, alphabetical).

```
{{ date_expression | date : format }}
```

```
{{ 603606171000 | date : "MM/dd/yyyy" }}
```

 → 02/16/1989



# The filter Filter

Selects a subset of items from an array and returns it as a new array.

The filter method takes a string, object, or function:

```
{{ ["Alyssa", "Nicoll", "loves", "Angular", "to", "pieces"] | filter:"a" }}
```

In this case, it selects all the words containing 'a':

`["Alyssa", "Angular"]`

We're using it to search our array



# Filtering an Array of Objects

```
var notesArray =  
[  
  {  
    "UserId": user.id,  
    "category": {"name": "Question",  
                 "icon": "question"},  
    "description": "Define Service",  
    "title": "What is a Service",  
    "content": "Service: Angular...",  
  },  
  {  
    "UserId": user.id,  
    "category": {"name": "Best Practice",  
                 "icon": "thumbs up"},  
    "description": "NgModel Best Practice",  
    "title": "NgModel BP",  
    "content": "Always use dot synt..."  
  }  
]
```

If we only want notes that have a category.name of "Question", we would filter like this:

```
filter:{category: {name: 'Question' }}
```



# Filtering our Notes Array

index.html

```
...  
<div class="note-wrapper">  
  <a ng-repeat="note in notes | filter:{category: {name: 'Question' }}"  
    ng-href="#/notes/{{note.id}}" class="card-notes" >  
  
    <nw-card ... ></nw-card>  
  </a>  
</div>  
  
<nw-category-select></nw-category-select>
```



# Filter Returns an Array

Filter returns a new notesArray which contains the proper category name.

```
var notesArray =  
[  
  {  
    "UserId": user.id,  
    "category": {"name": "Question",  
                 "icon": "question"},  
    "link" : "",  
    "description" : "Define Service",  
    "title" : "What is a Service",  
    "content": "Service: Angular...",  
  },  
  {  
    "UserId": user.id,  
    "category": {"name": "Best Practice",  
                 "icon": "thumbs up"},  
    "link" : "https://www.youtu...",  
    "description": "NgModel Best Practice",  
    "title" : "NgModel BP",  
    "content" : "Always use dot synt..."  
  }  
]
```

```
var notesArray =  
[  
  {  
    "UserId": user.id,  
    "category": {"name": "Question",  
                 "icon": "question"},  
    "link" : "",  
    "description" : "Define Service",  
    "title" : "What is a Service",  
    "content": "Service: Angular...",  
  }  
]
```

## NOTES



What is a Service

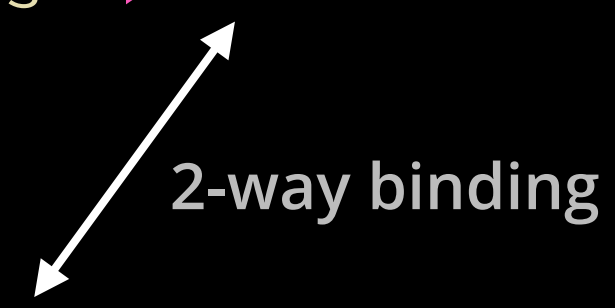


# Using filter to Sort Notes

We need to filter notes dynamically, so we use the `activeCategory` to designate which category is currently selected.

index.html

```
...  
<div class="note-wrapper">  
  <a ng-repeat="note in notes | filter: {category: {name: activeCategory}}"  
    ng-href="#/notes/{{note.id}}" class="card-notes" >  
    <nw-card ... ></nw-card>  
  </a>  
</div>  
  
<nw-category-select active-category="activeCategory"></nw-category-select>
```



A white double-headed arrow points from the `activeCategory` property in the `filter` object of the `ng-repeat` directive to the `active-category` attribute of the `nw-category-select` component. The text "2-way binding" is written next to the arrow.

We need to pass the `activeCategory` into the `nwCategorySelect`, so the value is shared through 2-way binding.



# Adding 2-way Binding for activeCategory

We need to create an isolate scope and give it access to the activeCategory.

📄 nw-category-select.html

```
<nw-category-select active-category="activeCategory"></nw-category-select>
```

📄 nw-category-select.js

```
angular.module('NoteWrangler')
.directive('nwCategorySelect', function(Category) {
  return {
    ...
    scope: {
      activeCategory: "="
    }
  };
});
```



ChromeFileEditViewHistoryBookmarksWindowPeopleHelp

localhost:8000/#/notes

Alyssa

UsersNotes

Filter is Working!

NOTEWRANGLER

NOTES

?

Service Service? Really Angular?

?

TEST TEST TEST

CATEGORIES

Testing (0)

Personal Note (0)

Bash (0)

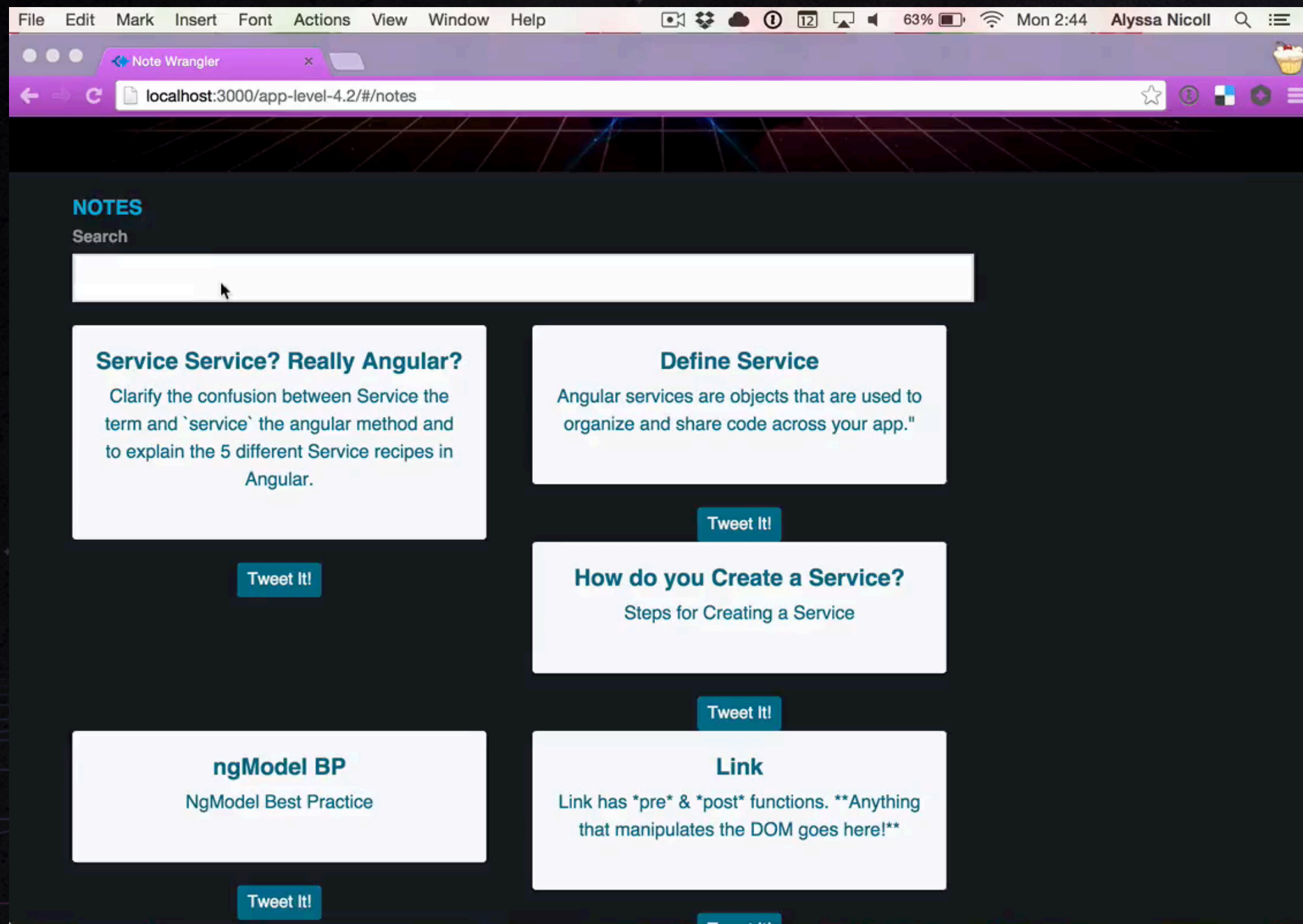
Idea (0)

Use with Caution (0)

Question (4)



# Let's Add a Search Box





# Searching With filter Filter

We can use filter Filter to create a quick and easy search!

index.html

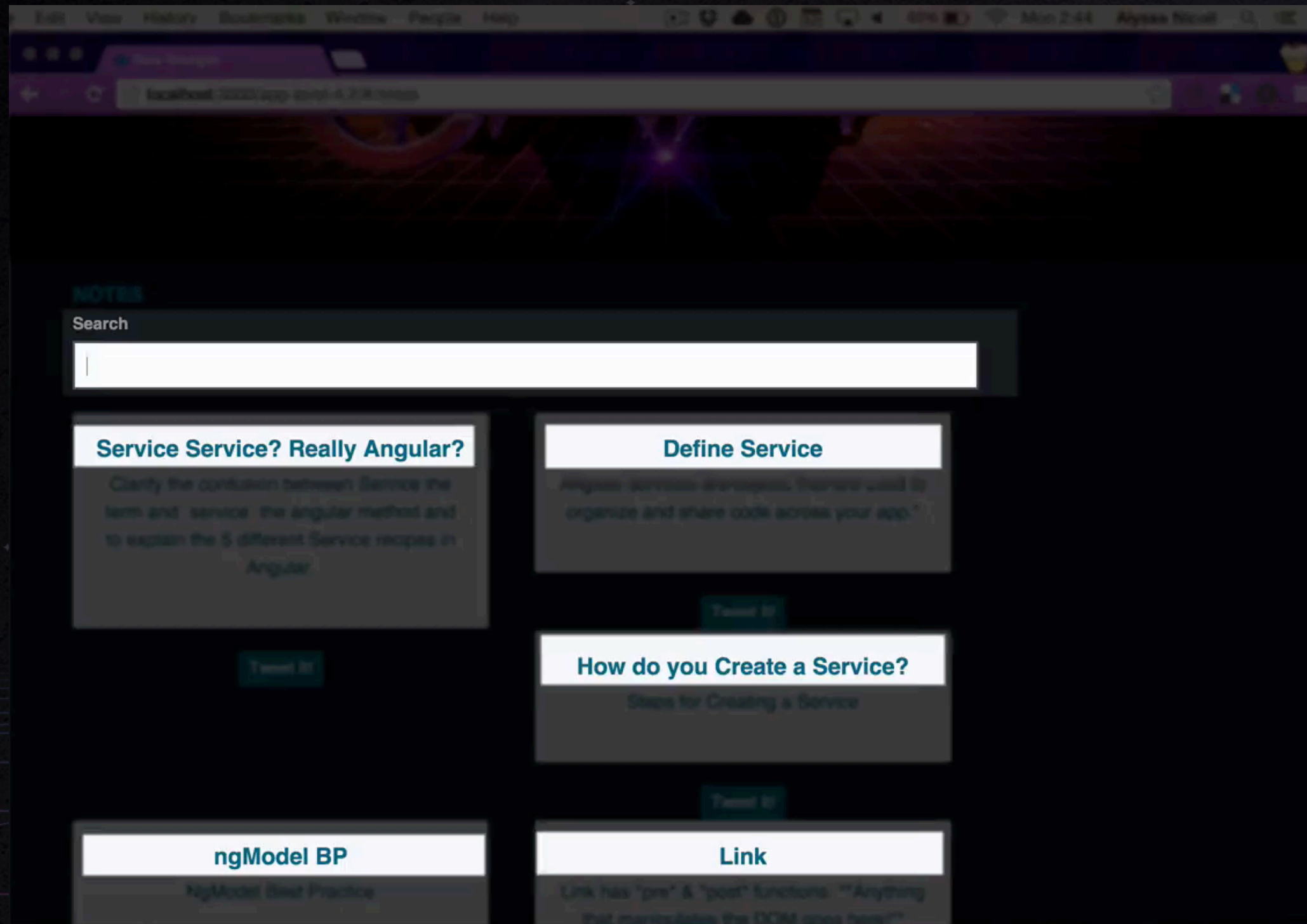
```
<input ng-model="search.title"/>
...
<div class="note-wrapper">
  <a class="card-notes" ng-repeat="note in notes | filter:search"
    ng-href="#/notes/{{note.id}}">
    <nw-card ... ></nw-card>
  </a>
</div>

<nw-category-select active-category="currentCategory" notes="notes">
</nw-category-select>
```

This will properly search on just the title values.



# Searching With filter Filter





# Searching With a Wildcard

We can use \$ to search across all properties on our notes.

index.html

```
<input ng-model="search.$"/>
...
<div class="note-wrapper">
  <a class="card-notes" ng-repeat="note in notes | filter:search"
    ng-href="#/notes/{{note.id}}">
    <nw-card ... ></nw-card>
  </a>
</div>

<nw-category-select active-category="currentCategory" notes="notes">
</nw-category-select>
```



# Searching With filter Filter

The screenshot shows a web browser window with the address bar displaying `localhost:3000/app-level-4.2/#/notes`. The application has a dark theme and a sidebar on the left with the title "NOTES" and a "Search" input field. The main content area displays a grid of note cards. Each card has a title, a description, and a "Tweet It!" button. The notes are:

- Service Service? Really Angular?**  
Clarify the confusion between Service the term and `service` the angular method and to explain the 5 different Service recipes in Angular.
- Define Service**  
Angular services are objects that are used to organize and share code across your app."
- How do you Create a Service?**  
Steps for Creating a Service
- ngModel BP**  
NgModel Best Practice
- Link**  
Link has `*pre*` & `*post*` functions. **Anything that manipulates the DOM goes here!**



# Chaining Filters

What if we want to use two different filters on the same ngRepeat?

index.html

```
<input ng-model="search.$"/>
...
<div class="note-wrapper">
  <a class="card-notes"
    ng-repeat="note in notes | filter:activeCategory | filter:search"
    ng-href="#/notes/{{note.id}}">
    <nw-card ... ></nw-card>
  </a>
</div>

<nw-category-select active-category="activeCategory" notes="notes">
</nw-category-select>
```

Using another pipe to  
separate filters



# Chaining filters

Chrome File Edit View History Bookmarks Window People Help 97% Mon 4:00 Alyssa Nicoll

Note Wrangler

localhost:3000/app-level-4.2/#/notes

## NOTE WRANGLER

### NOTES

Search

**Service Service? Really Angular?**

Clarify the confusion between Service the term and `service` the angular method and to explain the 5 different Service recipes in Angular.

**Define Service**

Angular services are objects that are used to organize and share code across your app."

**How do you Create a Service?**

Steps for Creating a Service

### CATEGORIES

- Testing
- Personal Note
- Bash
- Idea
- Use with Caution
- Question
- Best Practice
- Code Snippet

**STAYING SHARP** with Angular.js