



# Cross-Cutting **CONCERNS** for Angular Apps

# hi!

## I am Matt Vaughn



Developer, Speaker, Consultant, PodCaster, Musician, Owned by Lukka



@angularlicious



github.com/angularlicious



<http://www.angularlicio.us> **OR** [www.angularlicious.com](http://www.angularlicious.com)

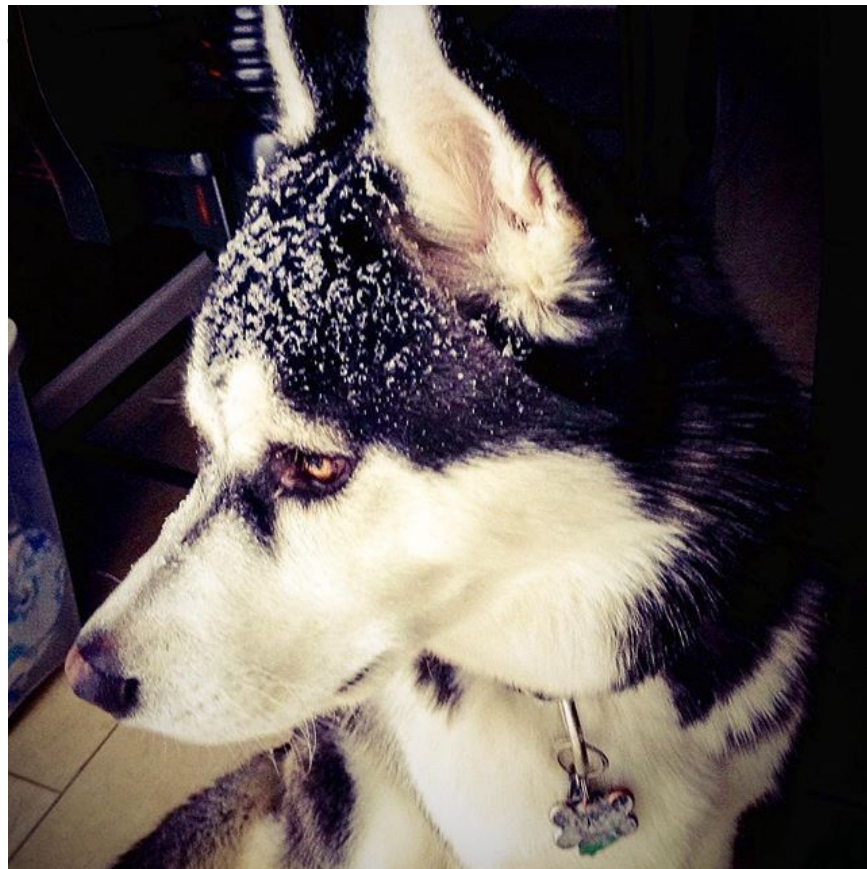


angularlicious

# Lukka **THE** Husky

Profile:

- 4 years old
- Instagram: **lukka\_the\_husky**





**Real-life  
cross-cutting  
concerns?**

# Home KITCHEN



- Lights
- Gas Stove/Range
- Hot Water
- Cold Water
- Heat/Air Conditioning
- Appliances





# What are **cross-cutting** concerns?

Things you need...







# Cross-Cutting Concerns

They are concerns that other parts of the application rely on or use - and are generally not related to the domain of the system or application.

- solve inter-dependency communication
- used by multiple layers and/or verticals of a system
- typically have a single-responsibility
- utility types
- allows reuse of the concern from a single-source (library)





# Types of **cross-cutting...**

Things you need...



# Types of Cross-Cutting Concerns

- Configuration
- Logging
- Exception Handling
- Exception Logging
- Notifications
- Security
  - Authentication
  - Authorization
- Data Validation
- Business Rules
- HTTP/Data Access
- Caching
- Data Persistence and Storage



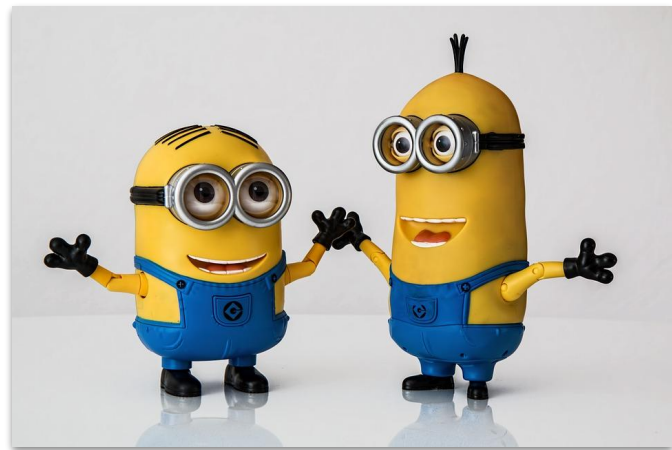
# BENEFITS?

WHY SHOULD I USE THEM, WHAT'S THE BENEFIT?



# WHAT are the BENEFITS?

- Solve common problems
- Single source of truth
- Consistent
- Reusable
- Maintainable
- Separate from domain code





# Implementation CONCERNS

What do I need to know about implementation?



# Implementation Goals

- Easily accessible
- Available early to application
- @Injectable
  - Use Angular DI
  - Service with API
  - Consumer provides the service
- Contained in its own @NgModule
- Should be configurable

# DEMO Time

Let's build a library that can be published to npm...







# **Final** THOUGHTS

The future is so bright...



# REMEMBER WHY.

- Don't Repeat Yourself (DRY) Principle
- Separation of Concerns
- Single Responsibility
- Encapsulation
- Interface-Driven Design
- Versioning
- ...

“

*BE SO GOOD  
THEY CANNOT  
IGNORE  
YOU.*

# Presentation **RESOURCES**



- [Cross-Cutting Concern Definition](#)
- [Aspects and Aspect-Oriented-Programming](#)



# Thanks!

Any questions?

You can find me at @angularlicious & matt@angularlicio.us