

# How to Impress Girls (or Boys) with HTML5 and AngularJS

DevNexus 2014

# Code Samples Online

- Code Items available on GitHub:
  - <https://github.com/verlin-inbloom/DevNexus-2014>



**You're not gonna amount to  
JACK SQUAT!**  
— Matt Foley

## **The Thinking Man's Chris Farley**

Now with a beard.

Works for dating as well as business.

**FAIL FAST, THEN ITERATE.**



**“I made you a mix tape”**

Late 1980s – mid 1990s



## **Taking it back**

HTML 4.01 / Static Web Page

# What's Wrong With This Picture?

- It's ugly.
  - We could fix that with CSS. But...
- It doesn't actually play the music!
  - Sure, we could fix that with Flash...
  - ...And forget about mobile.



**“I made you a CD”**

Late 1990s – 2003



*I've seen the*  
**FUTURE**  
*It's in my*  
**BROWSER**



**The future is 2006**

HTML5 / Static Web Page

# Make It Better With HTML5

- Add HTML5 elements for media playback
  - `<video>`, `<audio>`, `<canvas>`
  - Obviously, we're using `<audio>`
- We're already getting better, but this is still pretty lame...
  - Separate player for each file.
  - No easy way to move between tracks.
  - It's still ugly.



## **A first look at AngularJS**

HTML5 / AngularJS View

# Look Ma! I'm CODING!

- Let's turn our static web page into a real application with AngularJS directives
  - ng-app
    - Flags the element containing an Angular application
    - Single page can contain multiple Angular applications!
  - ng-repeat
    - DRY out the code, less copypasta
    - Does not fix the multiple players issue, though.
- Data Binding
  - AngularJS features two-way data binding
    - If you change it in the view, it's changed in the model, and vice-versa.
  - We're hardcoding our data here, but it gets better...



**“Um...I made you a web application?”**

2012 – present



**Everything is awesome!**

HTML5 / AngularJS View and Controller

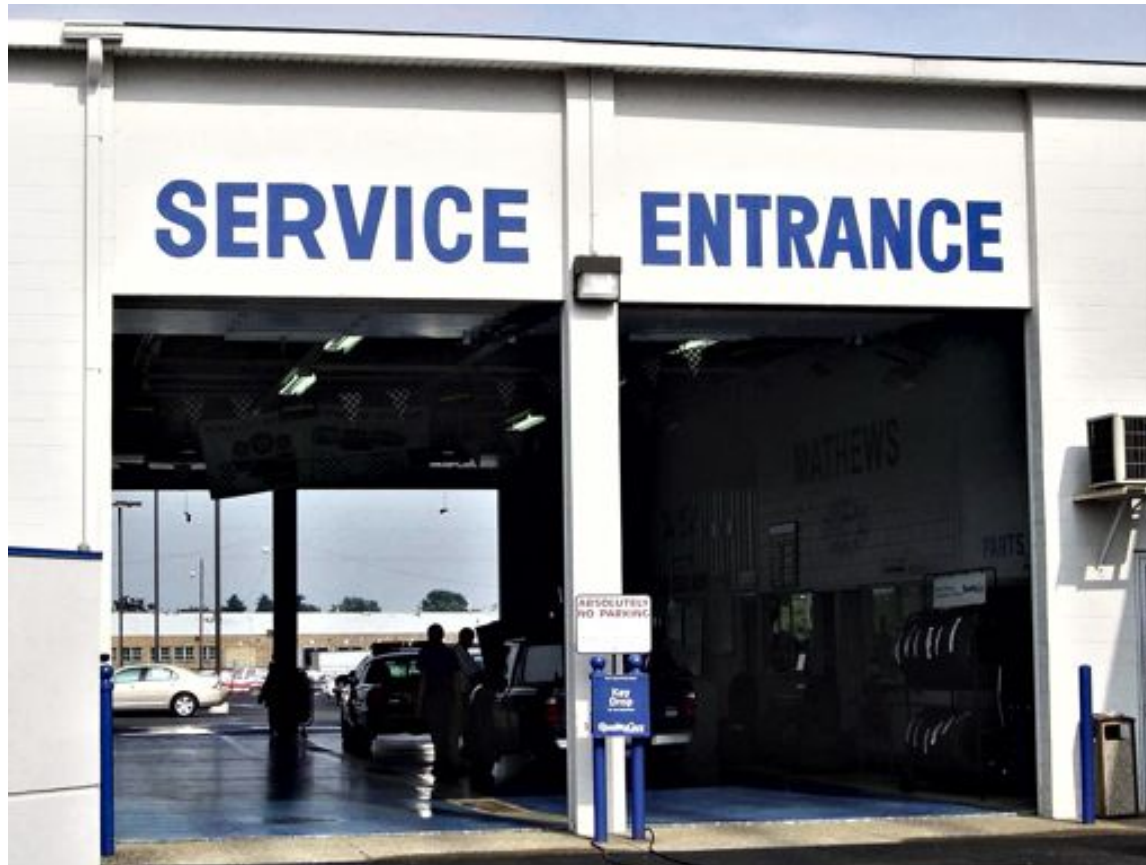
# Make Mine Modular

- We'll add a simple controller to provide the track listing to the view.
  - By doing this, we're separating our presentation from our data model and business logic.

# Now What?

- It's still ugly, but we've laid a foundation here to get better quickly.
- The next order of business is to get rid of the multiple players!





## **Inject yourself before you wreck yourself**

HTML5 / AngularJS View, Controller, and Services

# Services in AngularJS

- Instead of building giant controllers, AngularJS allows you to build modular services and inject them into controllers.
- Of course, you can still build giant controllers if you want to.
- Don't do that.
- No, seriously, don't do that.

# Adding Services

- We're going to add two services to our app
  - audio
    - Provides an invisible <audio> control with play and pause functions
    - We have to implement the player interface in the view!
  - tracklist
    - Provides files information about our playlist
    - Functions to return the current file and navigate forwards and backwards

# Win Some, Lose Some

- We've cleaned up the view
  - Only one player on the page
  - Familiar media player look and feel
- But we lost some functionality, too
  - What happened to the default controls?
  - Did I mention the tracks are still hardcoded?



## **Not enough tags for you?**

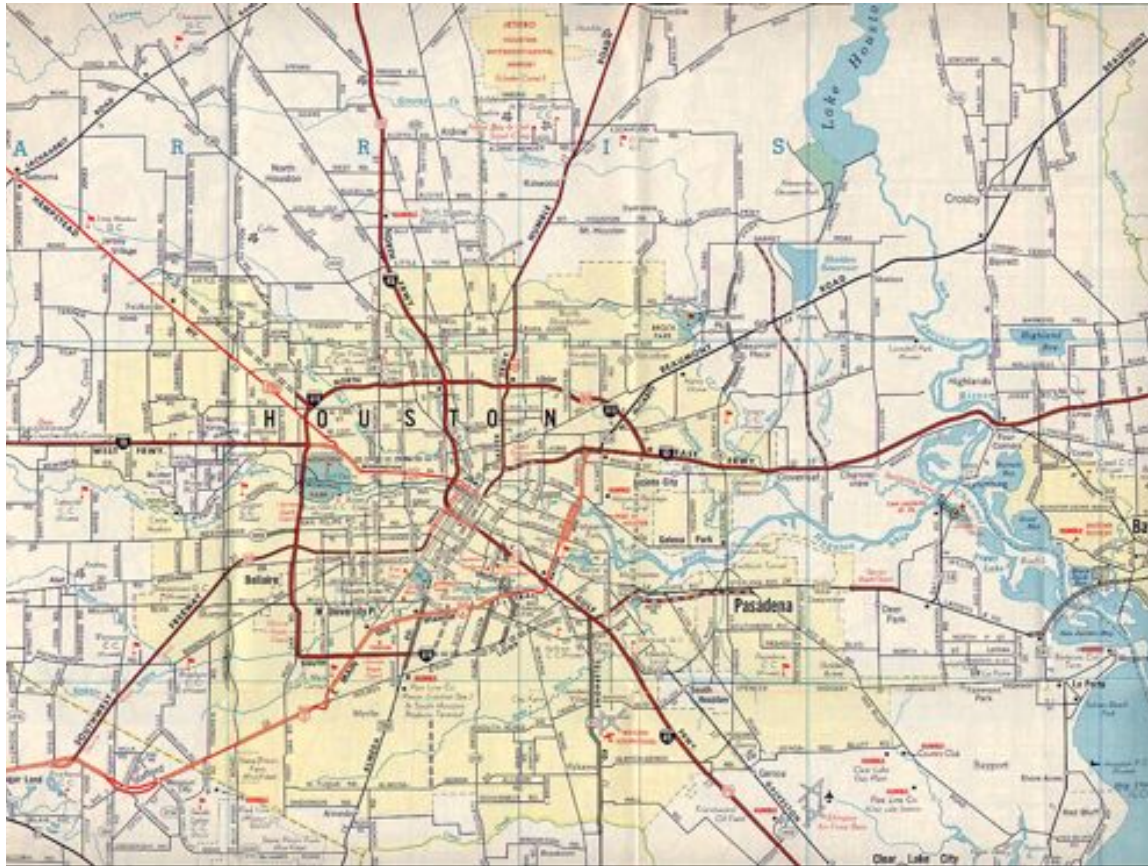
HTML5 / AngularJS View, Controller, Services and Directives

# AngularJS Directives

- Sometimes, the HTML you already know isn't enough.
- Angular lets you create your own directives to do some heavy lifting.
- We're going to create a simple directive that just writes to the console log.

# Some Notes on Directives

- The 'restrict' option specifies how the directive will be used
  - 'A' - Attribute
  - 'E' - Element
  - 'C' - Class
  - 'M' – Comment
- Use Element Directives with caution – older browsers will throw up!



**Where are we? And where do we go from here?**

Next Steps



# What Did We Learn?

- We laughed, we cried, we wrote a tiny app.
- HTML5 <audio>
- AngularJS Views
- AngularJS Controllers
- AngularJS Services
- AngularJS Directives

# What's Next?

- Wire up an HTTP backend
  - AngularJS supports GET, POST, and JSONP
  - Watch out for CORS
- Testing
  - Although I completely ignored it today, AngularJS provides great support for unit testing