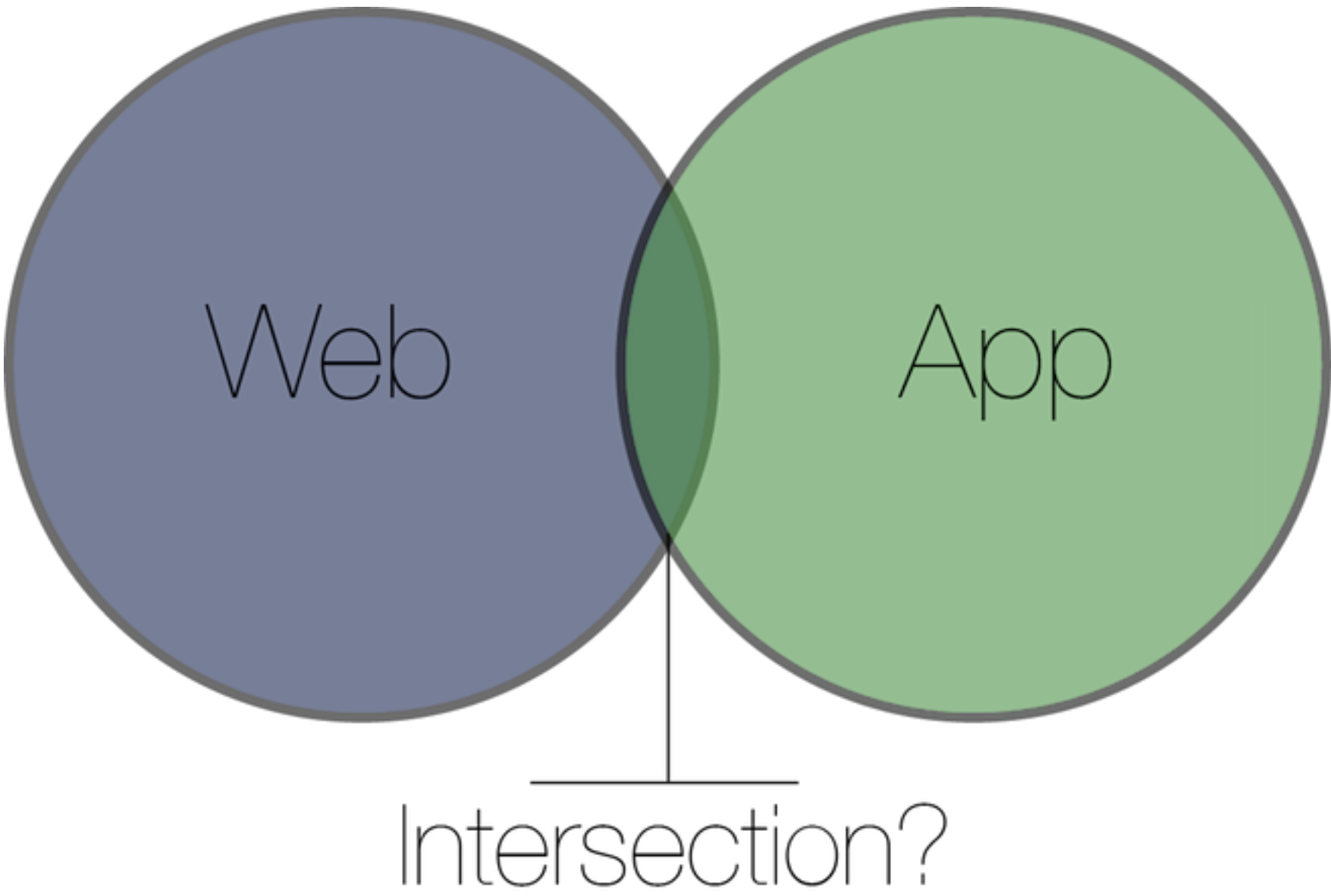


pas·sion

/ˈpaSHən/ 

noun

- 1. strong and barely controllable emotion.



If you think about it, web apps and native apps seem to go about things in completely opposite ways. The web server runs remotely and serves up pages; the native app runs locally and pulls down data. One is written in Ruby, HTML, CSS and Javascript while the other is written in Swift or Java.

Different architectures, different languages, yet it's the same app. These days we must support both Web and Mobile. What can be done? My passion project will implement each of the required technologies and then proceed to experiment with one or more ways to take them mobile or make it easy to do so.

Mike Farr

michaelfarr@gmail.com
mikefarr@mac.com

DBC sf-dragonflies-2015

User Requirements:

- I want to use my apps via the web, my phone, my tablet and often on my PC/Mac.
- I don't want a lowest common denominator experience, each should exploit the best of it's platform
- I want my tablet and phone apps to feel like native apps
- I want my web experience to feel like a web experience when appropriate: if what I am doing is browsing and a page metaphore is appropriate, don't give me a fixed frame app. When I resize a page of information, I want it to reflow.
- If the Internet is not available I should still have an app experience on my device using persistant data if appropriate.

Developer Requirements:

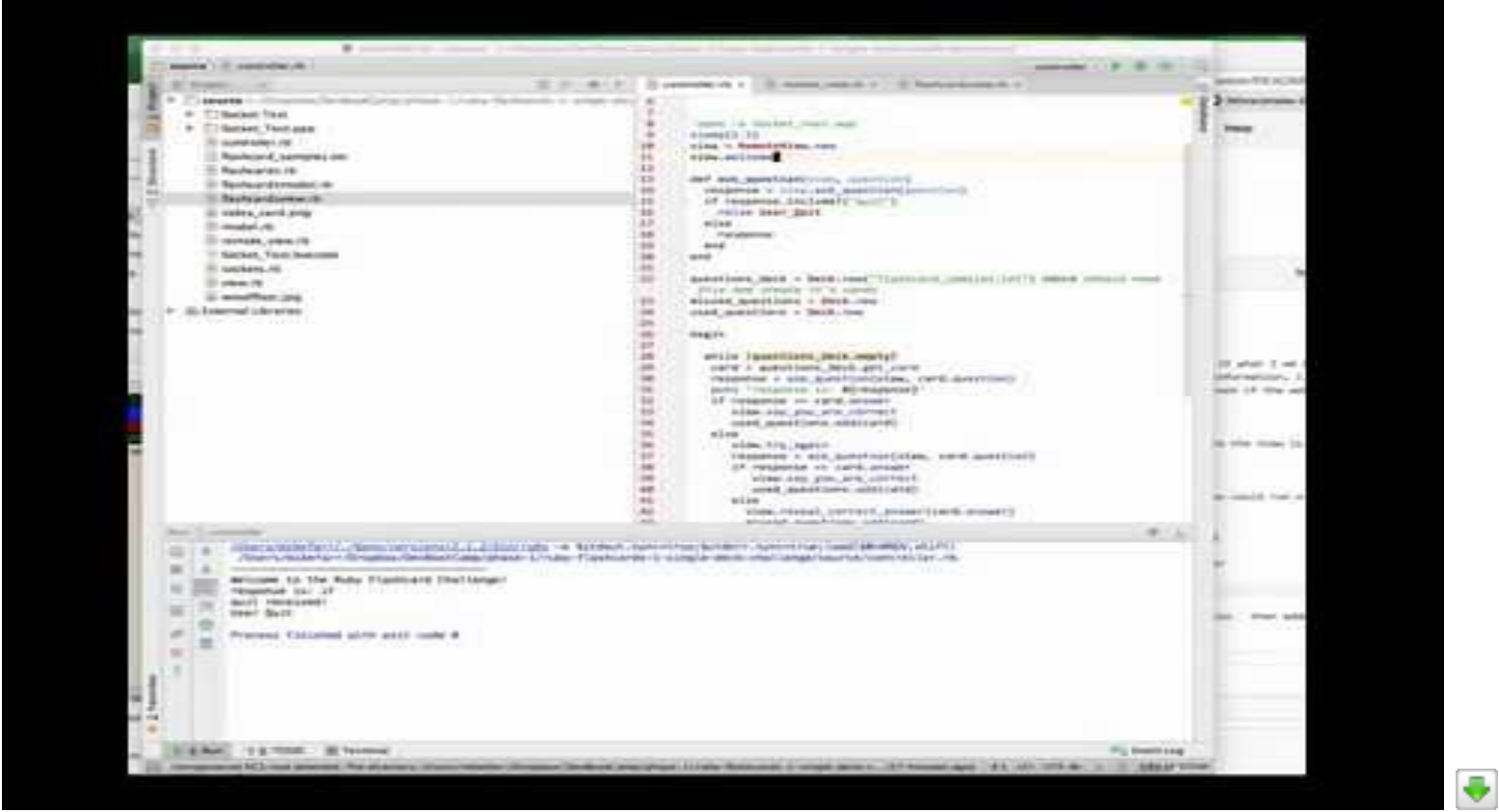
- I don't want to have to write multiple separate apps
- I want to leverage MVC to keep the model and controller logic common while the view is free to change with output target
- I want to reduce the number of languages and frameworks I have to learn

Background

In phase-1, we created several command line applications. Taking advantage of the MVC paradigm one could swap out a vanilla command line view with something more interesting. Both my Sudoku solver and my Ruby Racer use this approach. Here is RubyRacer (click to view video).



That's all well and good, but how far can this go? In the next video I replace the command-line View Module of our flashcard assignment with a RemoteView leaving the model and controller untouched. The remote view manages a UI server I built as a native Macintosh application. The UI server and the Ruby Flashcard app talk over sockets. The UIServer is much like a browser except instead of HTML, it's Ruby objects that are indirectly being rendered. This is done through a RemoteView class leaving the rest of the program untouched.



The next step would be to look at creating an ActiveRecord/Sinatra app, then as a native app reusing the model and controller.

The Web App I'd like to do revolves around a sheet music database for my Men's chorus at the St. Francis Yacht Club. We have around 1,000 pieces of music with data on composer, lyricist, season (i.e. holiday), style, parts (i.e Tenor, Bass). We'd want to search on all those fields. Additionally I think that storing the programs we sing gives the required many to many relation: Each song is in potentially many programs (think holiday favorites), each program contains many songs.

User Stories

- I want to list music titles
- I want to search for music based on genre, season, arranger, composer, parts.
- I want to select music for a program
- I want to add new music to the database
- I want to delete music from the database
- I want to be able to view the PDF of the music
- I'd like to print music.
- I'd like to "print to iPad"
- I'd like to see a calendar of upcoming events
- I'd like to see a detail of those events
- I'd like to see/hear past performances
- I'd like to search past performances by date, title or season
- I'd like to be able to add photos or recordings to the record of a performance
- I'd like to be able to add mp3 to a record
- I need to be able to log in securely
- I need to be able to add or remove users
- I need to be able to retrieve email addresses of users.
- I'd like to be able to store meeting minutes, Create assignments
- I'd like to be able to store the book and organizational records of skits or theatrical productions
- I'd like to be able to store the financial results of our performances
- I'd like to be able to see a preview of music (large thumbnail?)

MVP

- Viewing music.
- Searching the library by Composer, Arranger, Title, Season
- View sheet music PDFs
- Add/remove users and provide authentication.

Wireframes

1. login screen
- 2.

The club loaned me an intern who has around 800 pieces of sheet music scanned and we have meta data collected for about 500 of those.

More screen designs to come.

People

- id
- name
- password
- email
- phone
- status

Songs

- Composer
- Lyricist
- Publisher
- Title
- Parts
- URLs of scanned sheet music PDF
- URLs to recordings

Events

- id
- description
- location
- call time
- performance time
- contact_id

Officials

- Title
- person_id
- term

