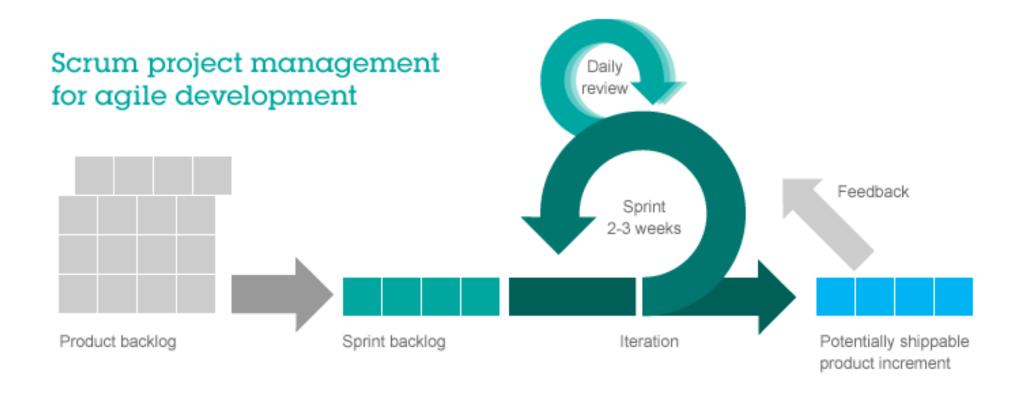


Let's say you have designed a product. How would you go and build it?

## Agile Manifesto

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan



http://www.ibm.com/smarterplanet/us/en/business\_agility/article/agiledevelopment.html

5 Points about Agile Development

We don't know everything... yet

Iteration over Specification

Potentially shippable releases

Definition of Done

Concise Feature Definitions (i.e. User Stories)



What's good and what's bad about Agile?

## Release Planning

Easier to more accurately estimate development effort Helps identify dependencies Keeps priorities in check

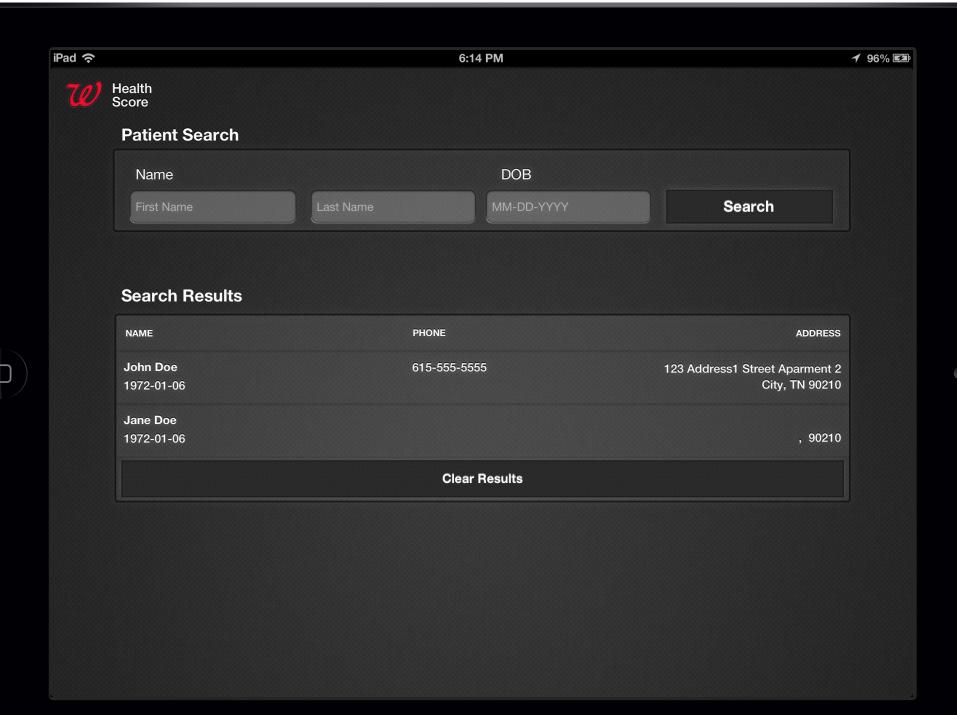
# A Few Techniques

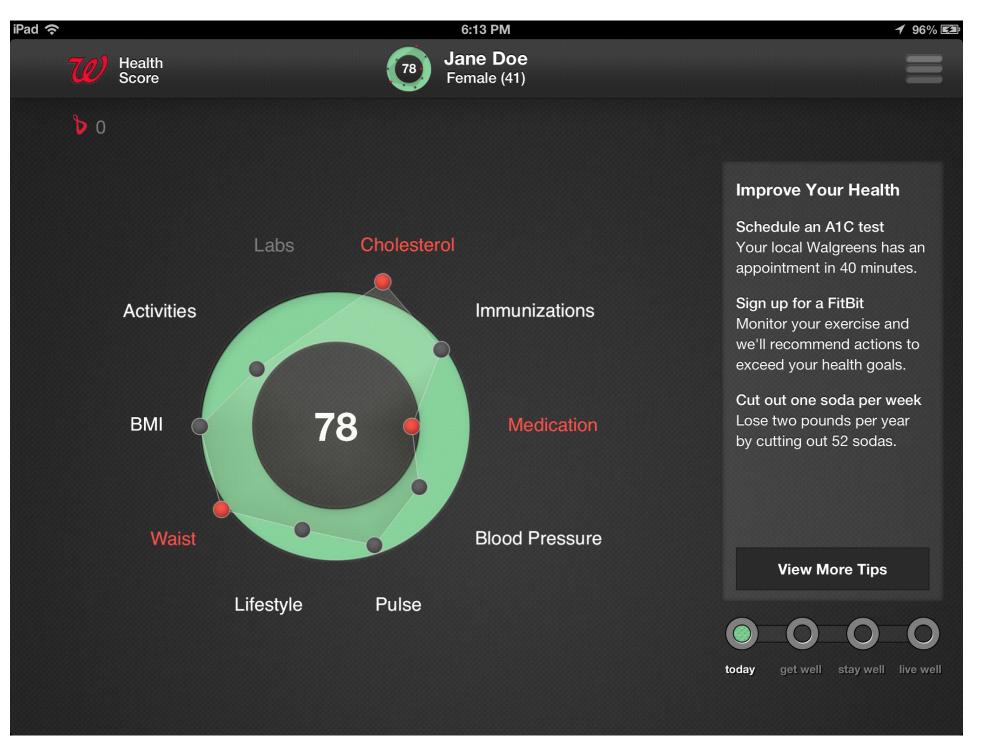
Breadth First Functionality vs Depth First Functionality By Functional Chunk Screen by Screen Walkthrough Based on a protocol (i.e. IMAP)

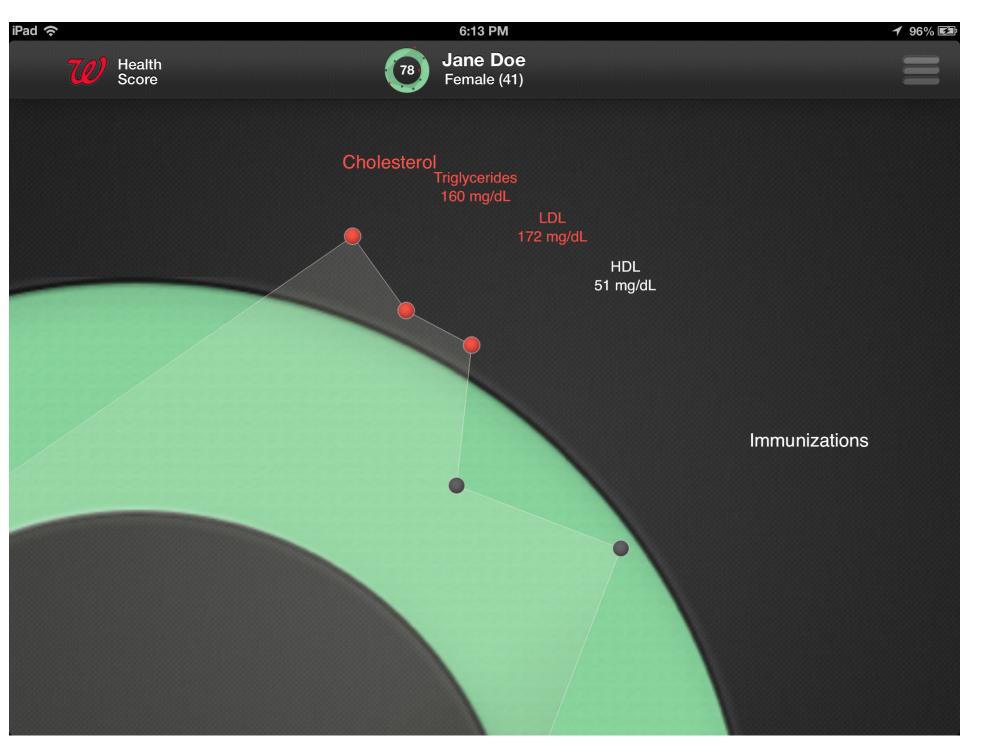
How does this relate to the concept of an MVP?

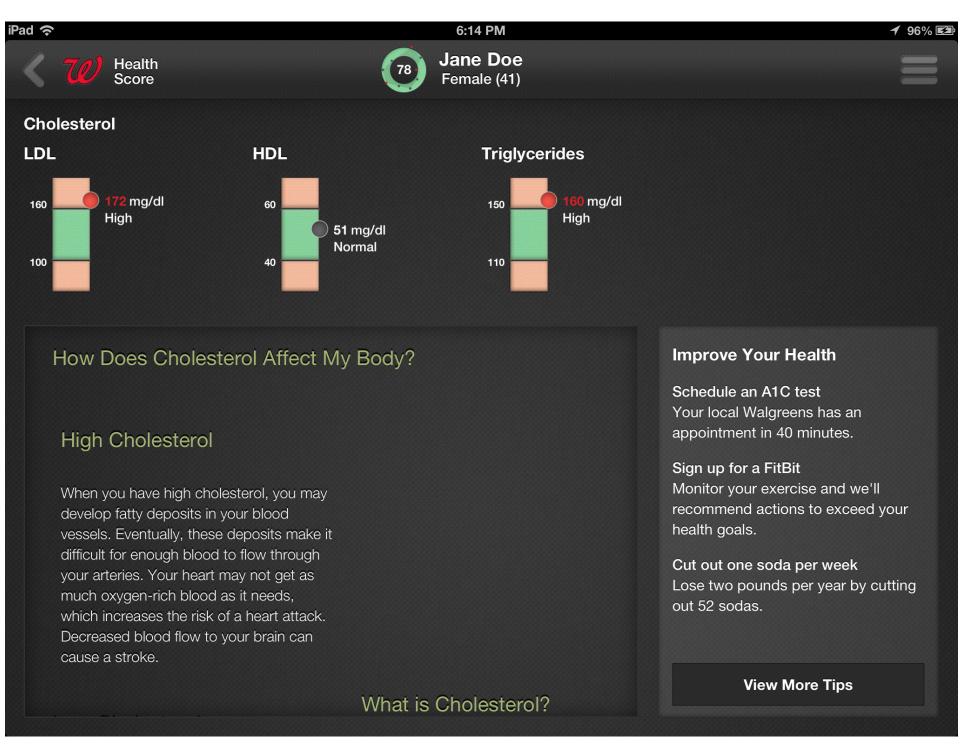
A Case Study.

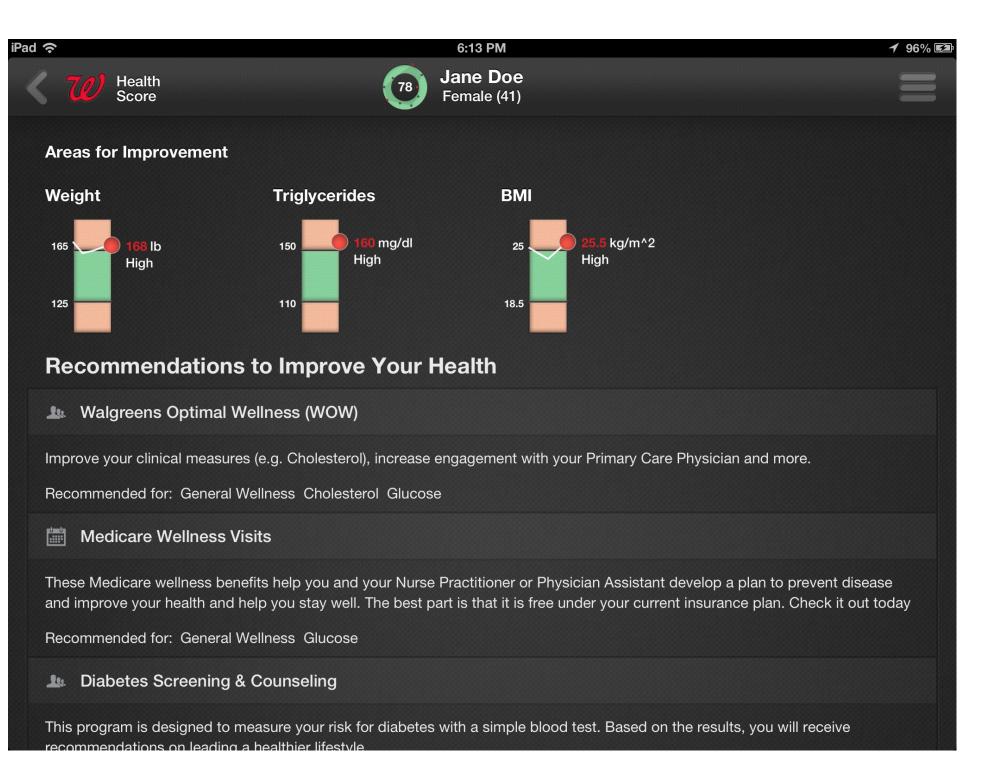
How would you break this design down?











#### Screens

Search (with and without results)
Single Patient's hGraph (zoomed in/out, what if)
Metric Detail
Top Recommendations
Patient Demographics

# Other Functionality

Health Risk Assessment Print PDF Authentication

# As a Group

How would you build a webmail app (i.e. Gmail). Start out by listing out the functionality of your app. Then organize the functionality into releases.

# Some Big Questions

What drives the release cycle?

Are we making incremental releases to the users?

How do we prioritize what to build first?

How do you balance user facing functionality with infrastructure tasks?