



FastFood

Review

The Browser

- HTML
- CSS
- JavaScript
- jQuery
- Bootstrap
- SEO

API Interaction

- APIs (Consuming)
- JSON
- AJAX
- Real Time Cloud Database via Firebase

Cutting Edge Development

- React.js

Dev Tools

- Heroku
- Git
- GitHub

Databases

- MySQL
- MongoDB

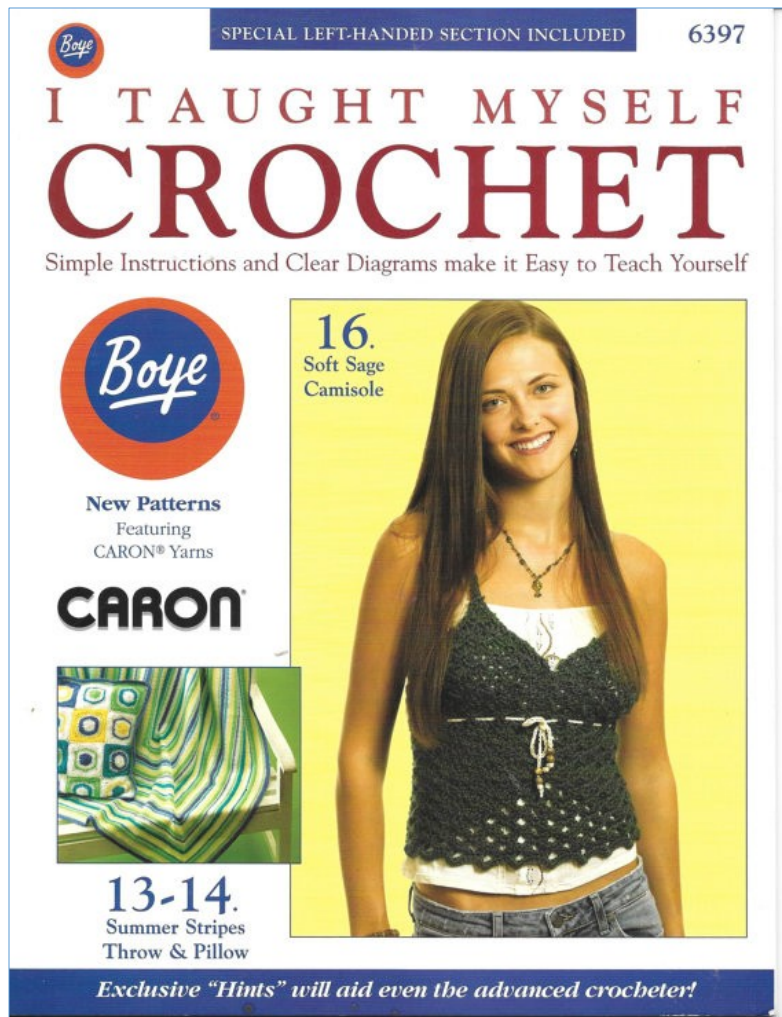
Server Side

- Templating Engines
- Sessions
- Writing tests
- Node.js
- Express.js
- Creating APIs
- MVC
- User Authentication
- ORM (Sequelize)
- Laravel

CS Fundamentals

- Algorithms
- Design Patterns

Never Stop Learning!



- The key to hireability and relevance is to constantly be learning.
- The field is always changing and there is never a point where you can call it quits.
- Stick with it! It's a **craft** that you continue to hone with time.

Final Projects

Your Challenge

Your Challenge

You have plenty of time to work on this.

You have ALL the skills you need.

You have a team of skilled developers.

You have experts willing to help.

No excuses. Time to prove yourself.

Expectations

Expectations

1. We expect whatever you build to have utility
2. We expect you to have market or real-world research that evidences your idea has REAL value to people.
3. We expect you to have done research on other web / mobile applications in your domain.
4. We expect you to put serious time and thought into this.
5. We expect you to report problems you are facing along the way.
6. We expect you to utilize some form of project management system.
7. We expect you to dig deep into documentation and external resources to learn what you need.

Requirements

Coding Requirements (but not really requirements)

- Must use ReactJS or Angular in *some* way (even if minimal)
- Must use a Node and Express Web Server
- Must be backed by a MySQL or MongoDB Database with a Sequelize or Mongoose ORM
- Must have both GET and POST routes for retrieving and adding new data
- Must be deployed using Heroku (with Data)
- Must utilize at least two libraries, packages, or technologies that we haven't discussed
- Must allow for or involve the authentication of users in some way
- Must have a polished frontend / UI
- Must have folder structure that meets MVC Paradigm
- Must meet good quality coding standards (indentation, scoping, naming)

But Honestly...

Do what you want. Talk to us first. Get our input.

*But if you have an idea and want to run with it,
we're not going to stop you.*

Remember:

No excuses. Time to prove yourself.

Deliverables

Project Updates

Instead of homework, you will be required to submit **4 Project Deliverables**.

These will be due each Saturday from now until Presentation Day.

Deliverables #1 (2/23) – Plan, Design, and Research

Deliverables #2 (3/3) – Ready MVP (Working App!), Mini-Presentation

Deliverables #3 (3/10) – Project Polish, Functionality Push, Final Plan

Final Presentation (3/17) – Presentation Time!

Deliverable #1

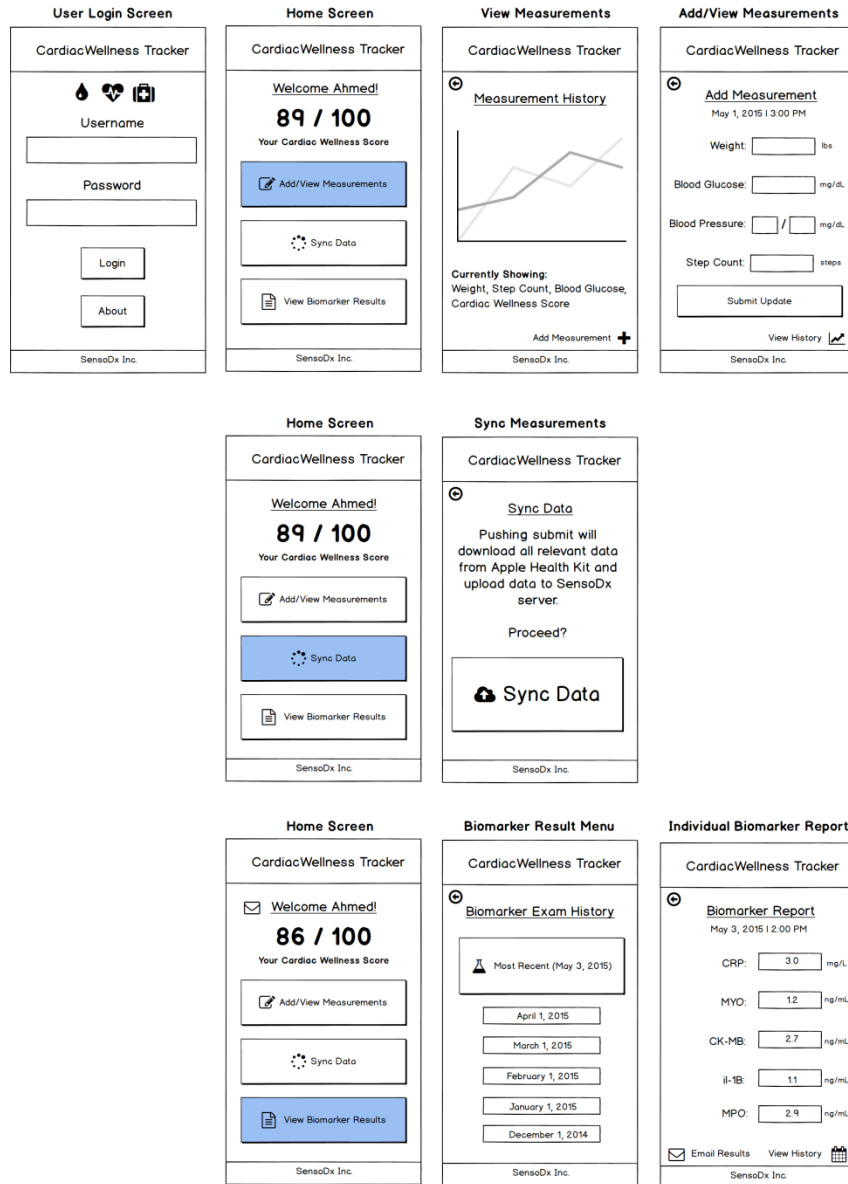
Deliverable #1

By July 2, 2016 you must submit a detailed plan of action for your project.

In this plan, you will include:

1. An overview of the intended application and WHY you feel it's valuable.
2. A set of DETAILED screen-by-screen design layouts with annotations describing all UI/UX components and all data relevant to the screen.
3. A breakdown of roles by group member.
4. A schedule for completion of various tasks.
5. A screenshot of your Jira, Trello, or Project Management Board that shows breakdown of tasks – assigned to group members with a schedule.

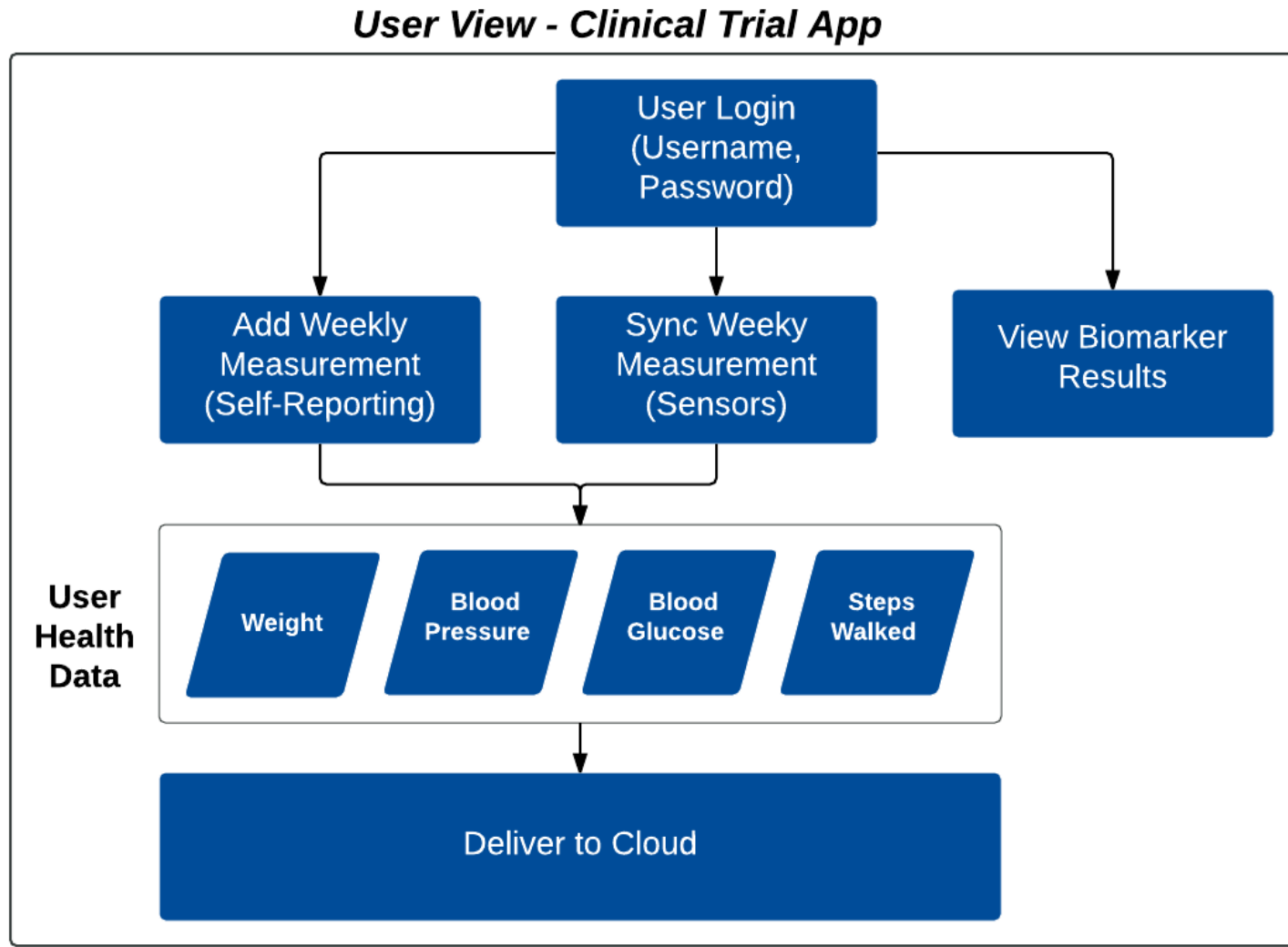
Example UI/UX Flow



Screen Map

This is intended to show you the order in which users would see screens. The three branches correlates with the three button options from the main menu.

Example Data Flow



Questions
