Home Opioid Patient-Controlled Analgesia (PCA) Box

Team 11 Recovery Improvement Interactive Technologies (RIIT)
Ian Russell, Finn Thompson, Ali Morgan, Michael Beach
Sponsor Seattle Children's Hospital & The University of Washington
Dr. Lance Patak, Dr. Stuart Solomon, Dr. Jacob Gross

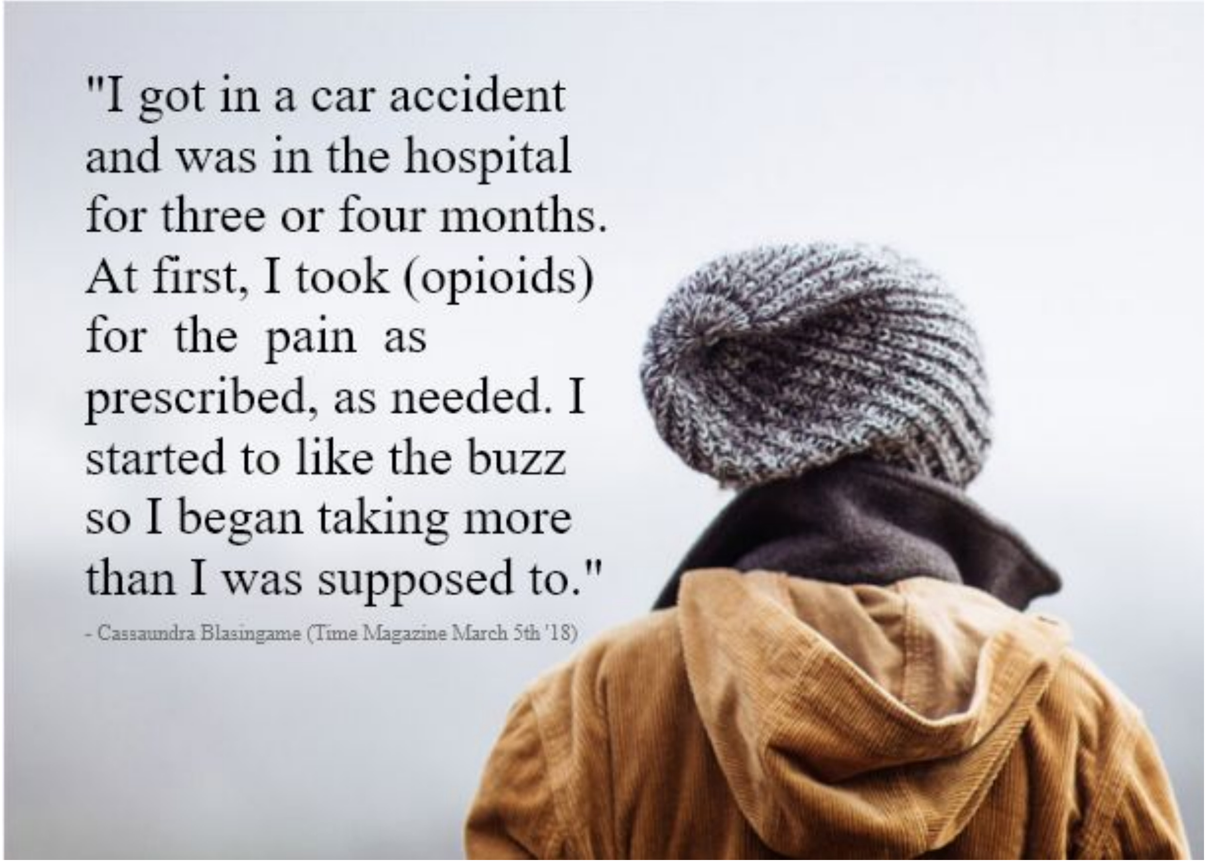
Milestone 2 **Prototype**

W
HUMAN CENTERED
DESIGN & ENGINEERING
UNIVERSITY of WASHINGTON

Project Overview

Opioids provide necessary pain relief to postoperative patients. However, opioids can be addictive, dangerous, and are often subject to misuse. Our project, an in-home oral Patient-Controlled Analgesia (PCA) box, will address these issues by providing patients with the guidance they need to manage their pain effectively during their postoperative recovery.

Our team will be researching, designing, prototyping, evaluating, and iterating a Patient-Controlled Analgesia (PCA) device and companion app that monitors and manages opioid prescriptions while connecting patients to doctors throughout the postoperative recovery phase. Through prototyping, we will explore various solutions to the problem as supported by our initial research and design sessions.



Design Process

Milestone 1
Design

Pill Box
Companion App

Milestone 2
Prototype

Physical Computing
Interactive Demo

Milestone 3
Evaluation

Usability Testing
Data Analysis

Milestone 4
Iteration

Pill Box 2.0
Updated Companion App

Milestone 2

Prototyping

Project Manager: Ian Russell

Milestone 2 Overview

The prototyping phase is an essential part of the process where conceptual ideas become concrete artifacts and systems to be tested in the next phase. We created interactive prototypes, a physical device and a companion app following our designs from Milestone One.

- Elements Used From Phase 1
- Interactive Interface
  - Phone App
  - Device
- 3D Modeling
  - Influences from Phase 1
  - Design Evolution
  - Laser Cut
  - 3D Model Version 1
  - 3D Model Version 2
  - 3D Modeling Process
- Physical Computing
  - Arduino
  - Code
- Moving Forward
  - Complications
  - A Look Ahead
  - The Next Phase

Interactive Interfaces

Device Interface

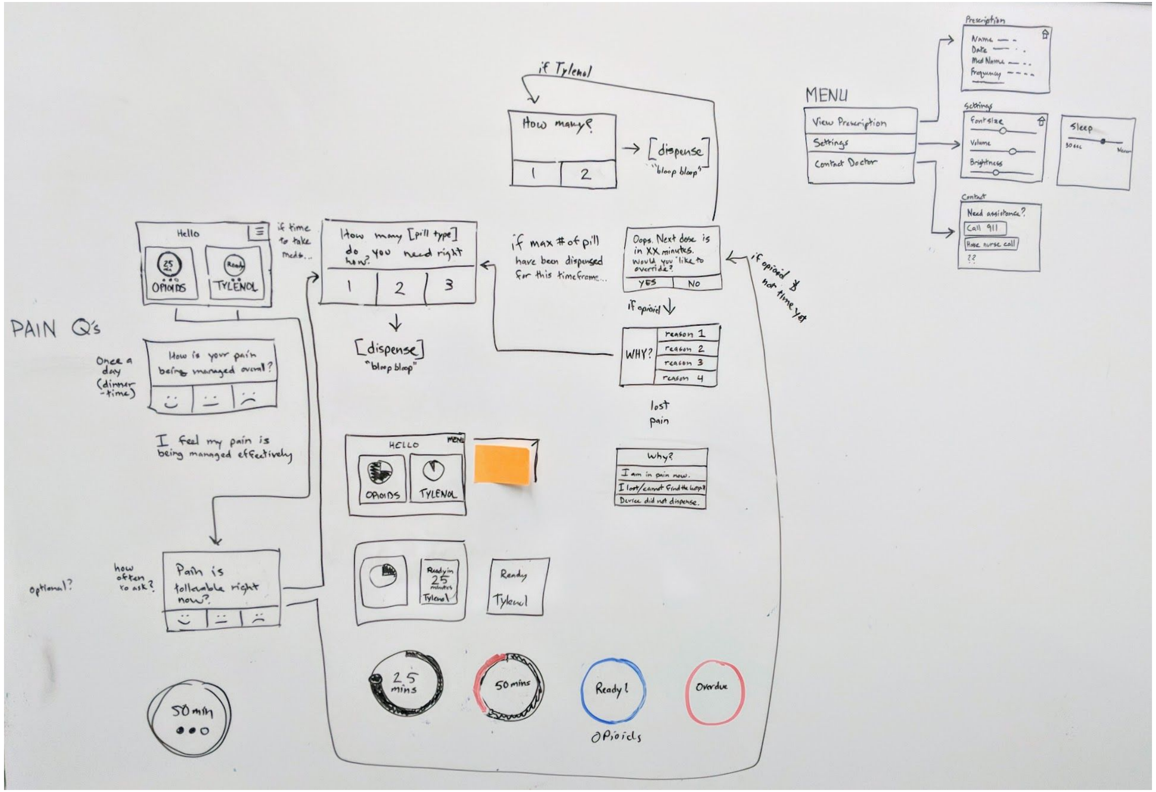
Purpose

Create an interface that guides patients through their pain management process.

Result

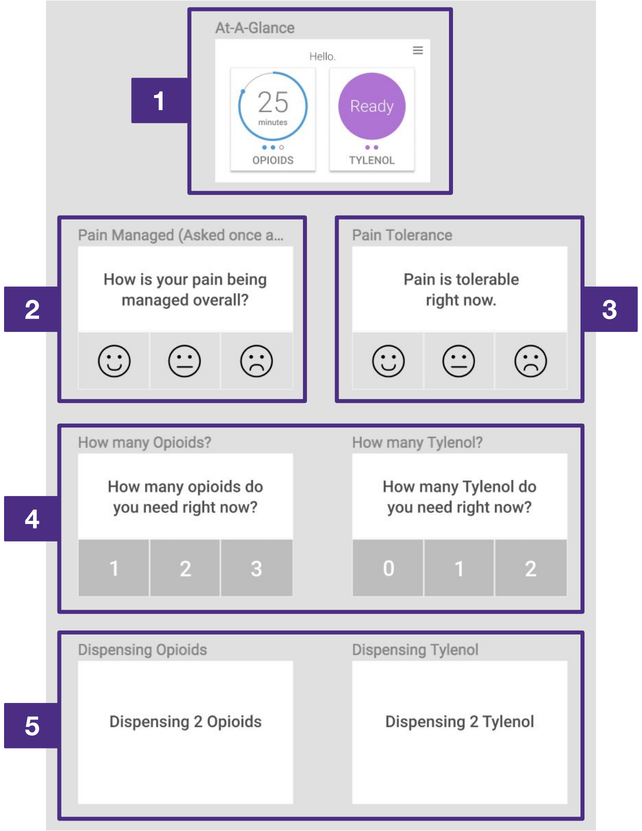
A patient-controlled interface that is used to control the device.

Elements used from Phase 1



These device screen sketches from Phase 1 informed our design for the interactive prototypes.

Interactive Interface: Device



We built these Figma using the sketches from our Design Phase and created an interactive demo using InvisionApp to demonstrate how our device screens would ideally appear.

Interactive Demo: https://invis.io/75HAGSBRSVX#/291898316_Home__No_Opioids_Yet

1. Patients are greeted with an at-a-glance screen that shows the number of Opioids and Tylenol that are ready for use, and how long until the next ones are available.
2. Once a day, patients will be asked how their pain is being managed overall.
3. After a patient clicks on one of the buttons to dispense opioids or Tylenol, they will be asked about their pain tolerance.
4. The patient is asked how many pills they need.
5. This screen is shown while pills are dispensed.

Interactive Interface: Device

If patients request doses beyond their prescription, the device presents them with the option to override the suggested dosage.

1. Patients are greeted with an at-a-glance screen that shows the number of Opioids and Tylenol that are ready for use, and how long until the next ones are available.

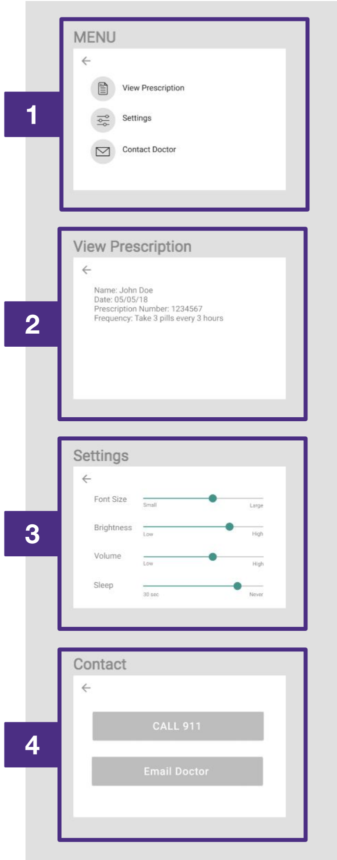
2. When no Opioids are available according to the prescription time, the patient can request an override.

3. After a patient requests an override, we ask for a reason for the override.

4. Next, the patient can choose how many opioids they need.

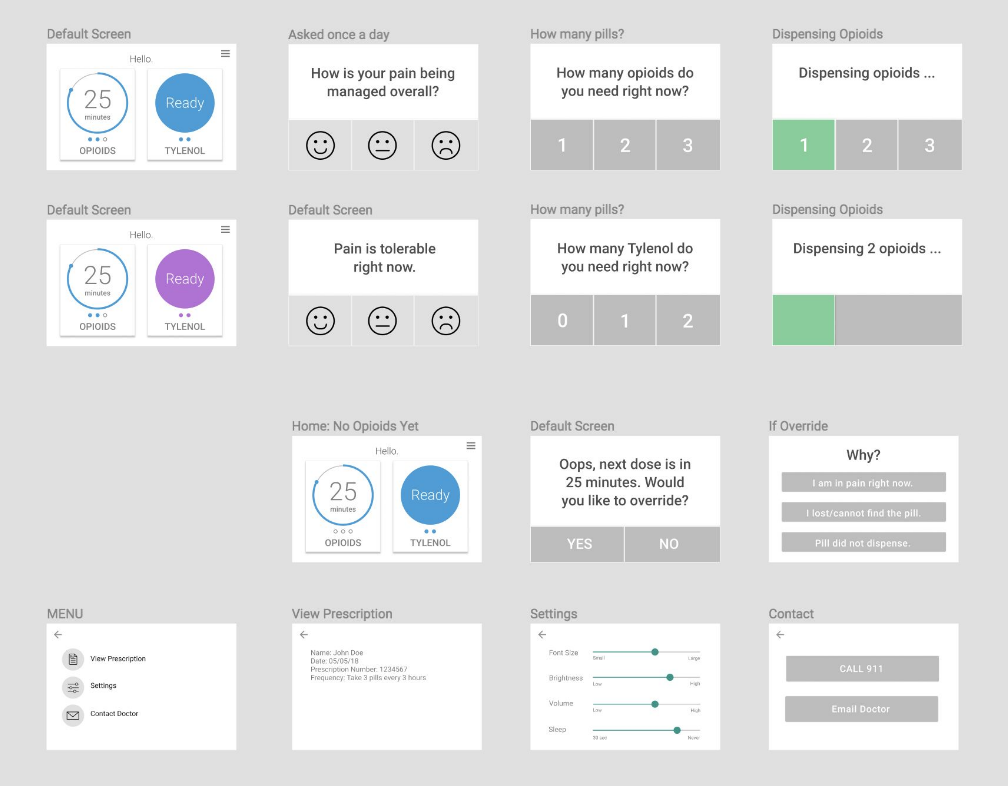
5. This screen is shown while pills are dispensing.

Interactive Interface: Device



1. The Main Menu allows users to navigate to: prescription information, settings, and contact options.
2. The Prescription screen shows prescriptions information.
3. The Settings screen allows users to adjust: font size, screen brightness, volume, and screen sleep time.
4. The Contact screen allows users to contact 911 or their doctor.

Interactive Interface: Device



Our full set of device interactive prototype screens.

App Interface

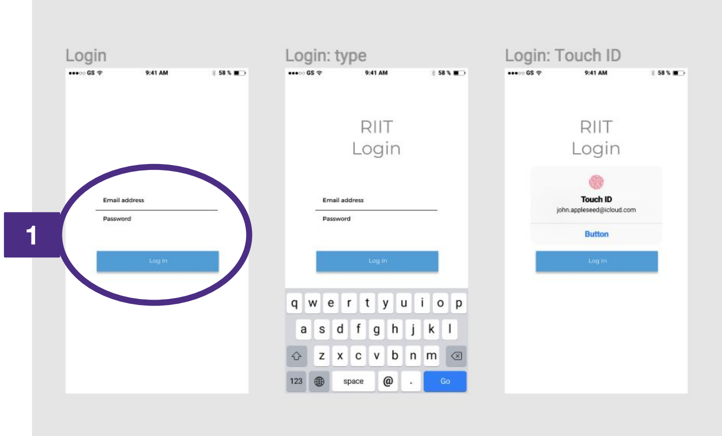
Purpose

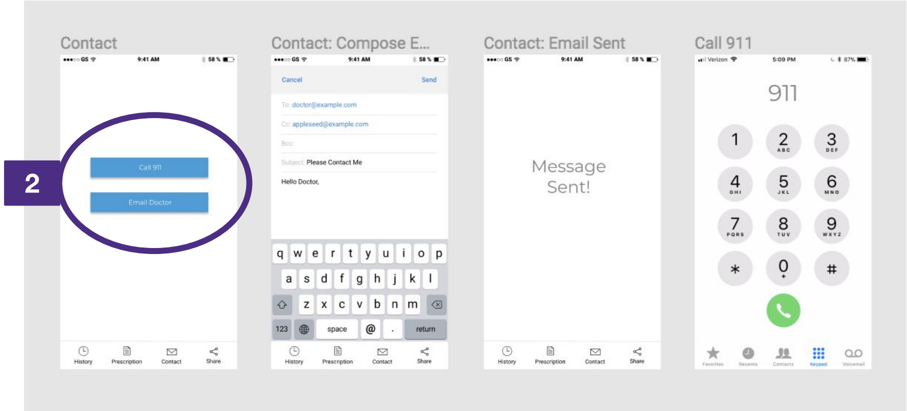
Create an interface to view personal data and share information with family and friends.

Result

An interactive prototype of an app that gives a patient the support they need to stay on track.

Interactive Interface: Phone App





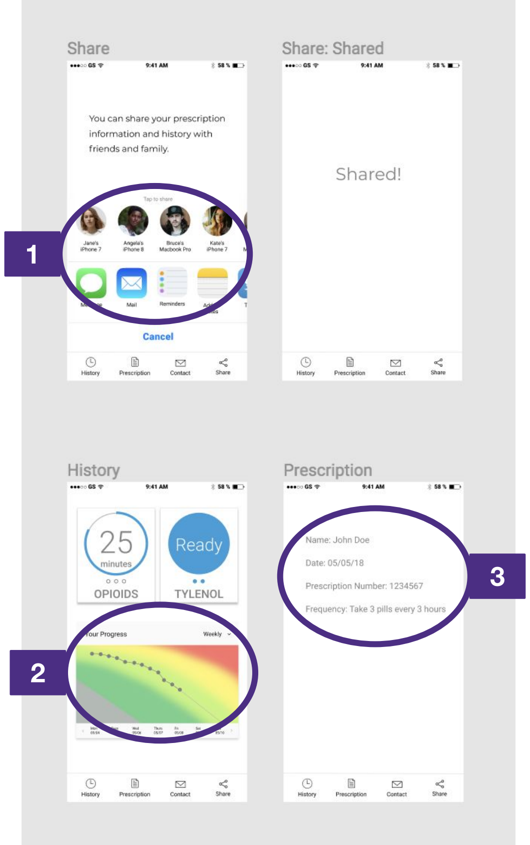
We used Figma and InvisionApp again to build the app interactive prototype, where users can view their historical dosage and prescription data, contact their doctor, or share access to their data with friends and family.

Interactive Demo:
https://invis.io/N2H6Z3EGUJ6#/292485173_Login

1. The Login page allows new users to log into the application. This screen is shown to patient’s who are just getting started, or family, friends, or care providers that have had the account shared.

2. The Contact page allows users to call 911 or easily send an email to the doctor on file to request a check in.

Interactive Interface: Phone App



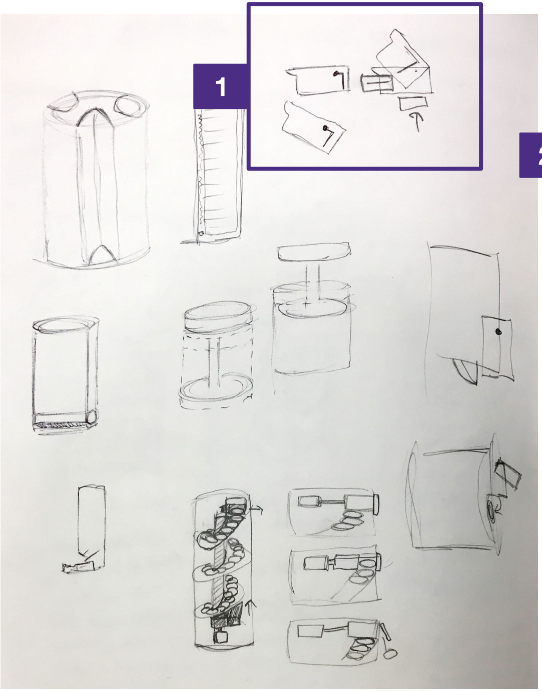
1. The Share page would allow patients to share their prescription information and history with their friends and family. This will helps members who are offering support to see the patient's progress.

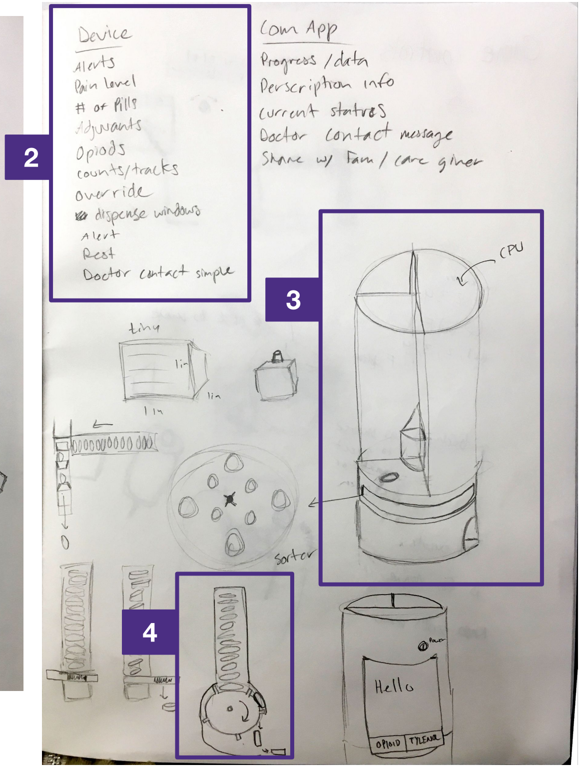
2. The History page shows stats of the patient's opioid use over time. This multicolor graph shown here would help the patient see their weaning progress. If the use path does not follow the green trajectory and moved into the orange or red areas, a medical team would contact the patient for a check in.

3. The Prescription page would show important details of the prescription that is loaded into the PCA device.

3D Modeling

3D Modeling: Influences from Phase 1





In Phase 1 we focused primarily on the devices interface and the companion application, leaving ambiguity to how the physical device would look and function.

1. We brainstormed many concepts for controlled dispensing ranging from pez dispensers to gumball machine styles.

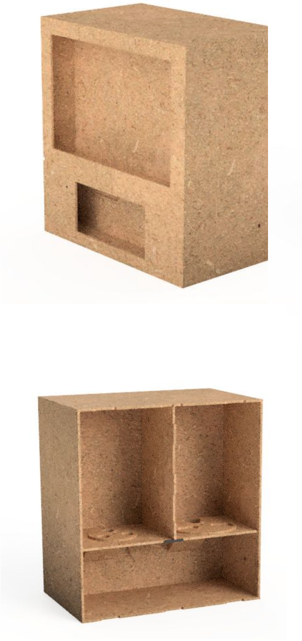
2. Device requirements were determined in phase one.

3. Inspiration for the overall look, feel and size of the device was gleaned from these design sessions.

4. Mechanisms much like our final iteration were conceptualised in phase one.

3D Modeling: Design Evolution

Laser Cut



3D V1

3D V2

Low Fidelity Laser Cut

High speed, low effort; with this we created a physical model to realize size and functionality requirements and measurements

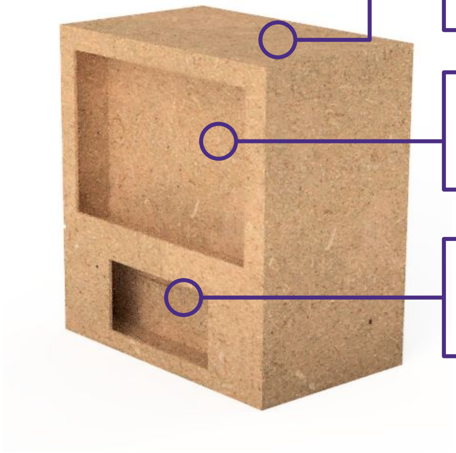
3D Model Version 1

In the first round of 3D modeling, we incorporated size adjustments as we explored new possibilities with physical design and space-efficient functionality.

3D Model Version 2

Our first 3D print guided further iteration, changing overall design shape, dispensing mechanism, dispensal area, cartridge design, and microcontroller area.

3D Modeling: Laser Cut



Square design allowed for swift laser cut and assembly. We determined dimensions from measuring our microcontroller components.

Flat screen angle due to limitations on swift laser cut prototype; depth of space designed to suit screen specifications.

Pill portal located on the same side as the screen for user convenience, large size to comfortably fit various hand sizes.

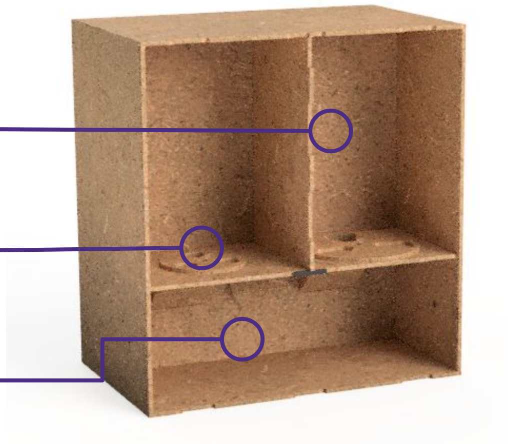
Lessons Learned

- Vertical orientation made the screen hard to see and use.
- Screen spacing did not allow space for cables.
- Cartridges need to be funneled towards holes.
- Need a compartment to insure one pill is dispensed at a time.

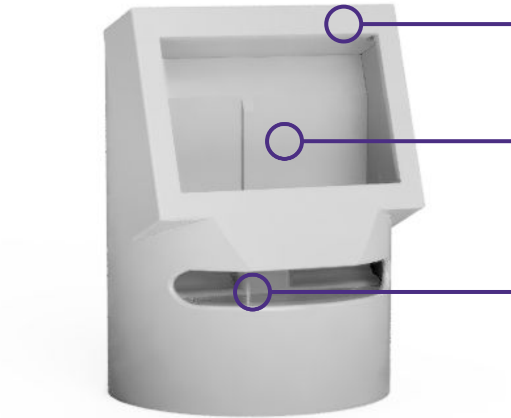
Large pill containers to hold over 30 pills at a time, loaded by removing top cover.

Two size pill dispenser wheels to signify ability to dispense different pill types.

Large undercarriage to allow for two motors, microcontroller, speaker, and necessary wiring, placed in the back two-thirds to allow for pill flow from cartridges above and space for the pill portal.



3D Model Version 1



Round device shape emulates pills bottles and helped maximize space while keeping the device footprint small.

Tilted screen casing makes it easier for users to interact with the device.

Opening to dispensed pills raised to increase space for electronics.

Lessons Learned

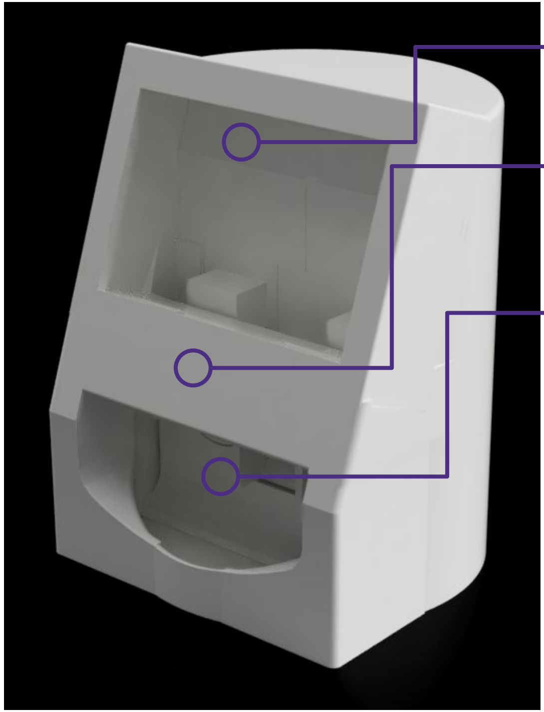
- Device’s fully round shape contrasted unpleasantly with the screen case
- Pill portal size was too small for fingers to reach into comfortably
- Revolving door mechanism did not function properly
- Electronics had three times the amount of space they needed.

Dispensing mechanism design that works similar to a revolving door. This design reduced the space needed under the cartridges and functioned with our original gear motors.

Slot for screen wiring. Designed to separate electronics from pills and users.

Most electronic components are stored in the base of the device. These components also provide weight to the device to make it more stable.

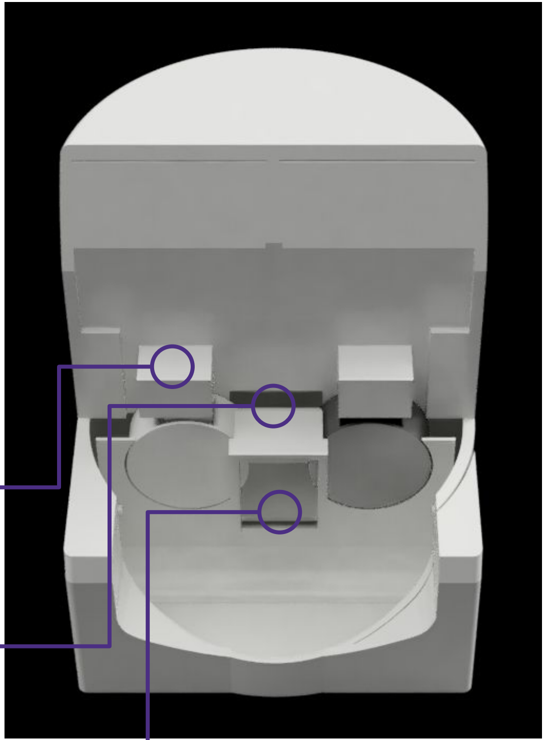
3D Model Version 2



Increased screen spacing and tolerants to make installation with wiring easier.

Adjusted front face of device to aesthetically unite screen face with overall device.

Pill portal size dramatically increased for ease of access; outer wall size increased to hold descending pills inside container until user extracts them.

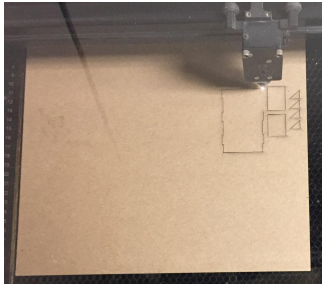


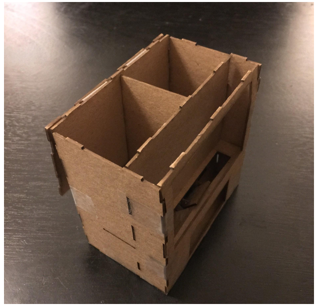
Mechanism and cartridge discharge style adjusted to accommodate servo motors and new dimensional constraints on electrical compartment.

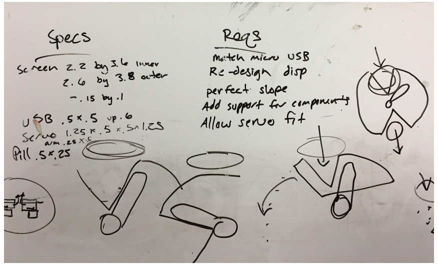
Wiring portals moved and increased in size as well as introducing two more compartments for wires to go through without allowing contact from user or pills.

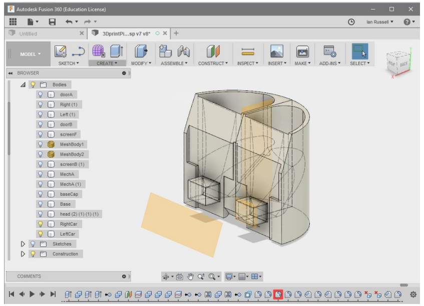
Container to guide pills to fall down the middle regardless of which chamber they are released from while hiding electronics.

3D Model Process







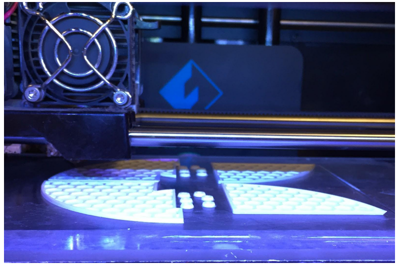


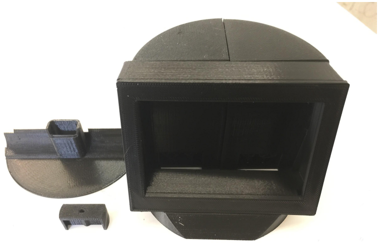
Laser cut creation and assembly

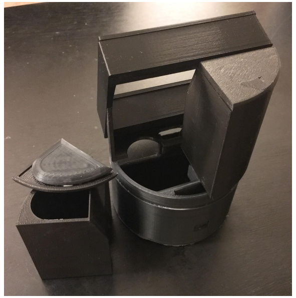
Test for physical constraints

Measure, sketch and design as requirements evolve

3D modeling the device in Autodesk Fusion 360







3D printing using Flashforge Creator Pro

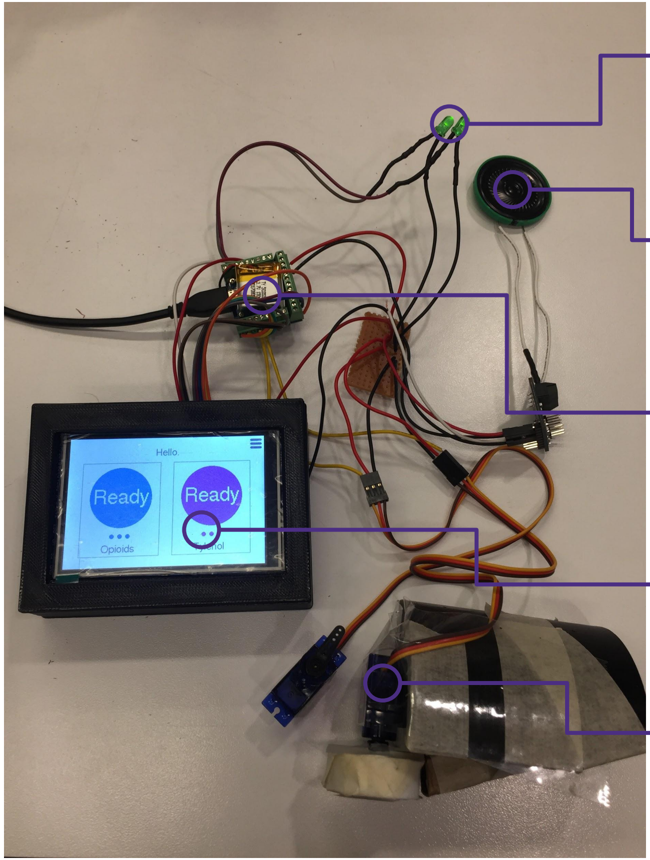
Inspection and post processing

Assembly and design analysis

Testing electronic components fit with the body and its usability

Physical Computing

Physical Computing



A pair of LEDs, intended to be placed on either side of the area where the pills are dispense, display which medication is ready without requiring the screen to be on.

A speaker, connected to and powered by an amplifier, provides the ability to play audio feedback when the user chooses to dispense a pill or perform another action.

The main board, a TinyDuino, with proto board and terminal blocks is a compact Arduino with easily accessible pinouts that provides a simple battery module for mobile operation while charging when plugged in.

The device display screen, a 3.5” thin-film transistor LCD with a touchscreen, acting as the primary mode of both input and output for the user.

Two servo motors, enabling the device to spin the pill dispensing mechanism.

Physical Computing

```
#ifndef Component_h
#define Component_h

#include <Adafruit_GFX.h>
#include "States.h"

class Component {
  bool needsClear = false;
  bool needsRepaint = true;

protected:
  int x;
  int y;
  int w;
  int h;

public:
  Component(int x, int y, int w, int h);
  int getX();
  int getY();
  int getWidth();
  int getHeight();
  virtual bool isValid(State state);
  virtual void onRepaint(Adafruit_GFX &g);
  virtual void onPress(int x, int y);
  virtual void onClick(int x, int y);
  bool contains(int x, int y);
  int dx(int x); // translate x
  int dy(int y); // translate y
  void paint(Adafruit_GFX &g);
  void repaint();
  void clear(Adafruit_GFX &g);
  void requestClear();
};

#endif
```

Once we finished assembling the microcontroller, we programmed our device to function for the purpose of our prototype. While Arduino offers basic built-in libraries and Adafruit supplies a simple graphics drawing library for their touchscreen, neither allowed us to easily and seamlessly create interactive graphical components to realize our designs from Phase 1.

To address this issue, we wrote a basic component library that tracks position and size while providing inheritable functions that allow custom components to handle drawing and touch events seamlessly. The Component header class is pictured here.

We also wrote the custom graphical code to draw arcs to create the partial circles indicating time until next dose, as the built-in graphics library only supported drawing simple shapes like circles, rectangles, triangles, and so on. We originally hoped to directly draw images from Figma onto the device screen off of an SD card. However, we found that the Arduino could not handle this action at a reasonable speed (10 seconds to draw an image the size of the screen), and the Adafruit graphics library could not handle smaller images correctly once the screen was set to a rotated state.

Full code can be found at https://github.com/FThompson/SmartPCA-Capstone

Physical Computing

```
#include "Prescription.h"
#include "Arduino.h"

Prescription::Prescription(char* label, int maxDose, long doseWindow, bool showOverride)
    : label(label), maxDose(maxDose), doseWindow(doseWindow), showOverride(showOverride) {

}

long Prescription::getTimeUntilNextDose() {
  if (lastDoses[0] > 0) {
    for (int i = 0; i < maxDose; i++) {
      long timeSinceLastDose = millis() - lastDoses[i];
      long time = doseWindow * (i + 1) - timeSinceLastDose;
      if (time >= 0) {
        return time;
      }
    }
  }
  return 0;
}

int Prescription::getAvailableDoses() {
  int available = 0;
  unsigned long ms = millis();
  for (int i = 0; i < maxDose; i++) {
    if (lastDoses[i] == 0) {
      available++;
    } else {
      long curWindow = doseWindow * (i + 1);
      if ((ms - lastDoses[i]) > curWindow) {
        available++;
      }
    }
  }
  return available;
}

void Prescription::use(int count) {
  for (int i = maxDose - 1; i > 0; i--) {
    lastDoses[i] = lastDoses[i - 1]; // push back previous dose times
  }
  unsigned long ms = millis();
  for (int i = 0; i < count; i++) {
    lastDoses[i] = ms; // record current dose times
  }
}
```

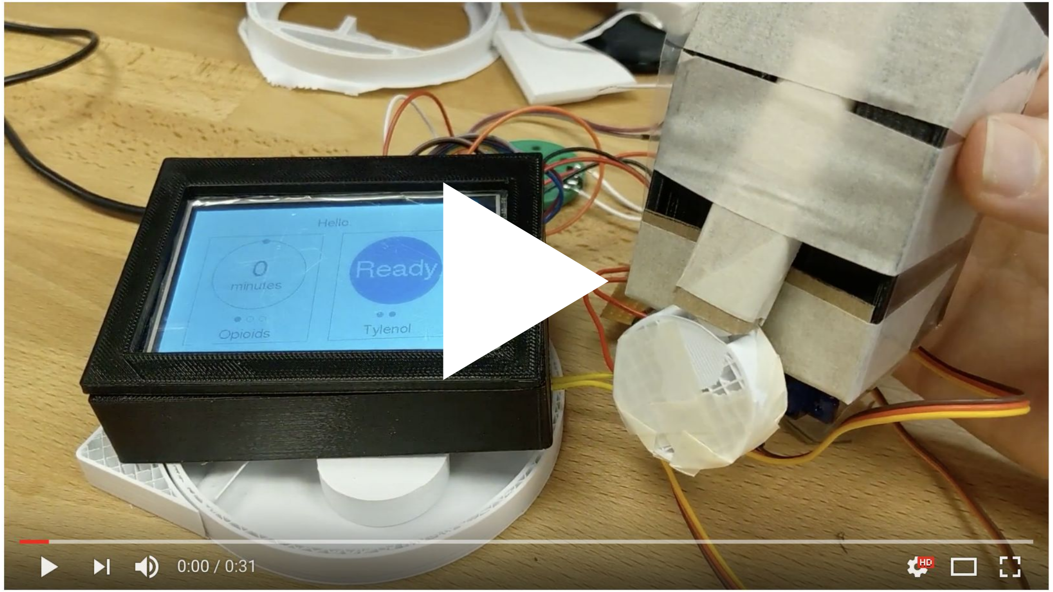
In addition to programming the user interface and interactive flow of our device, we wrote classes to control the physical components of our device and code to handle prescriptions of any dose and dose window.

A typical opioid prescription might direct a patient to take 2-3 pills every 3-4 hours. Our device prototype supports withdrawing without question up to the maximum dose (3 pills every 4 hours). If a user tries to withdraw beyond that point, they may be shown the override screen, if a prescription is set to do so. For example, requesting opioids beyond the prescribed dosage brings up a screen that asks the user why they need extra. Tylenol and other OTC medications are never limited.

The Prescription class on the left tracks how many doses are available and the time, allowing for these pieces of information to be easily retrieved elsewhere and displayed on the at-a-glance home display showing dosage status.

User Flow

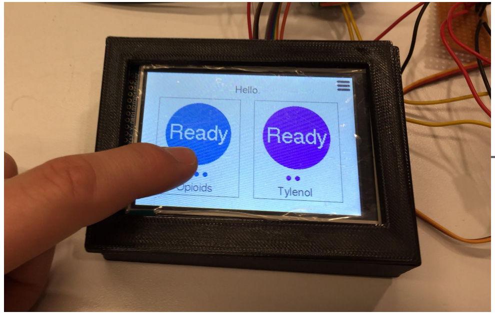
Prototype 1

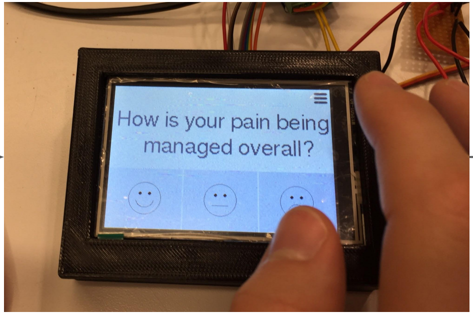


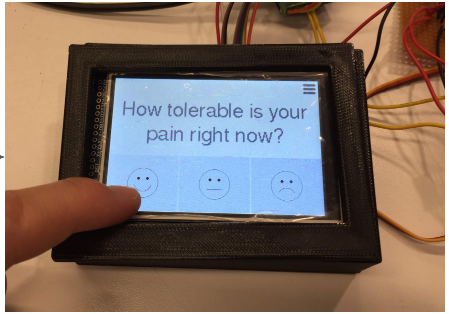
We combined the physical prototype with a low fidelity modifications to our first 3D print using chipboard and tape.

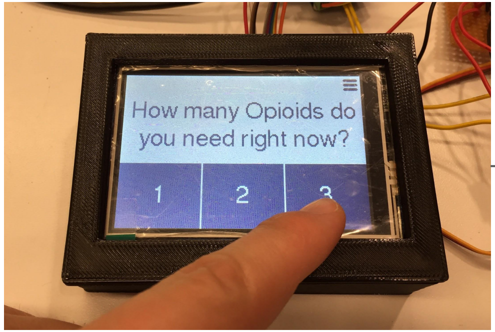
https://youtu.be/ZwmD_5nV8Iw

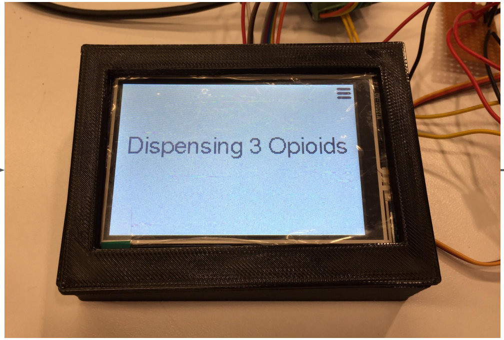
User Flow: Requesting Opioids



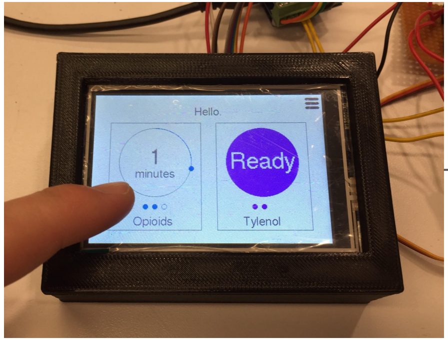


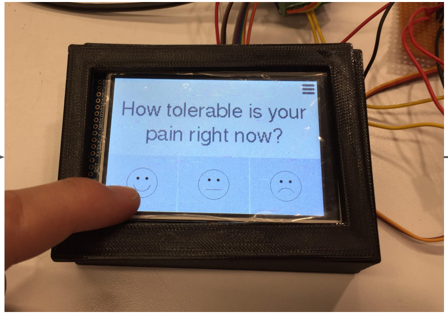


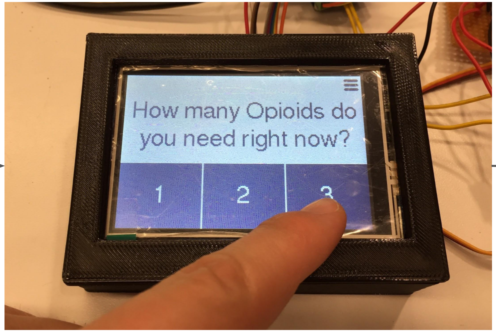


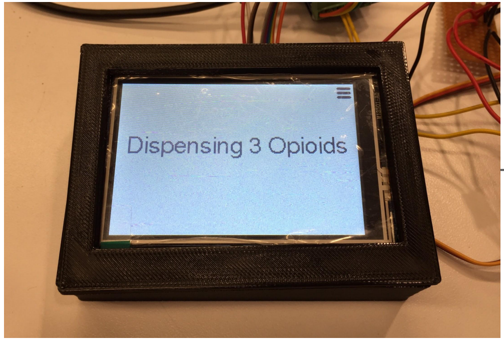


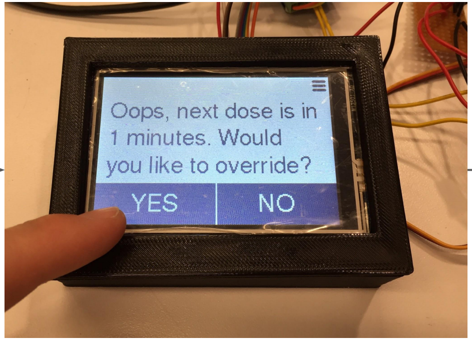
User Flow: Requesting an Override

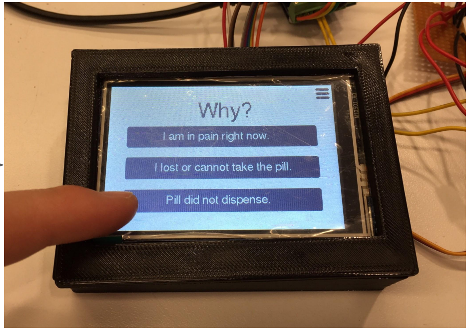






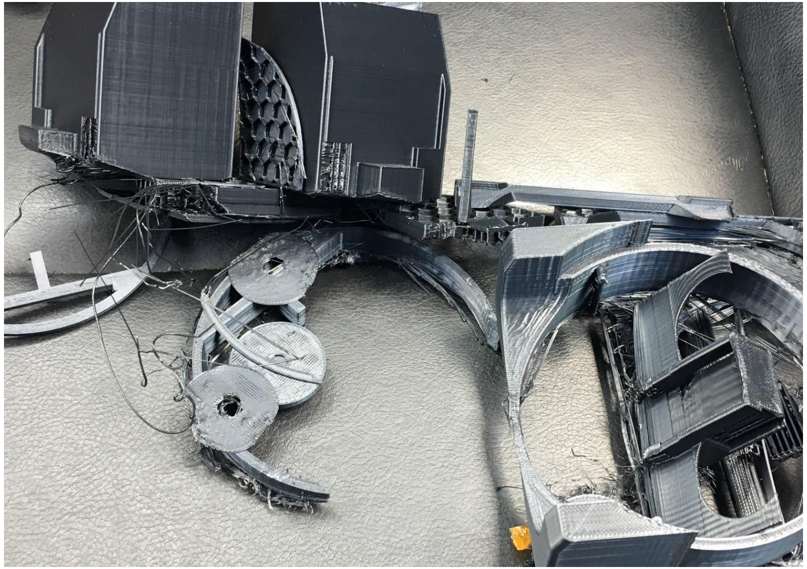






Moving Forward

Complications



Over the course of the Prototyping Phase, there were a few complications. Many of the 3D prints failed and there were a few component changes in the physical computing portion of the prototype.

Unfortunately, due to the time constraints brought on by the failed prints, we do not have a fully functional prototype.

We are printing the needed components and will be able to complete a functional prototype with no delay to our next phase.

A Look Ahead

In this Design phase we turned conceptual ideas into concrete artifacts so systems can be built and tested.

In Phase 2, we created a 3D and 2D Prototype for the PCA device, the device interface, and the companion app using 3D printing, physical computing, and digital interactive wireframes.

In Phase 3, we will Evaluate and Analyze our PCA device through usability evaluations with users.

In Phase 4, we will Iterate on our design by utilizing what we have learned from all previous steps in this process.

The Next Phase

Milestone 1
Design

Pill Box
Companion App

Milestone 2
Prototype

Physical Computing
Interactive Demo

Milestone 3
Evaluation

Usability Testing
Data Analysis

Milestone 4
Iteration

Pill Box 2.0
Updated Companion App