Team A1

Pour-Over-and-Over ...and-over ...and-over

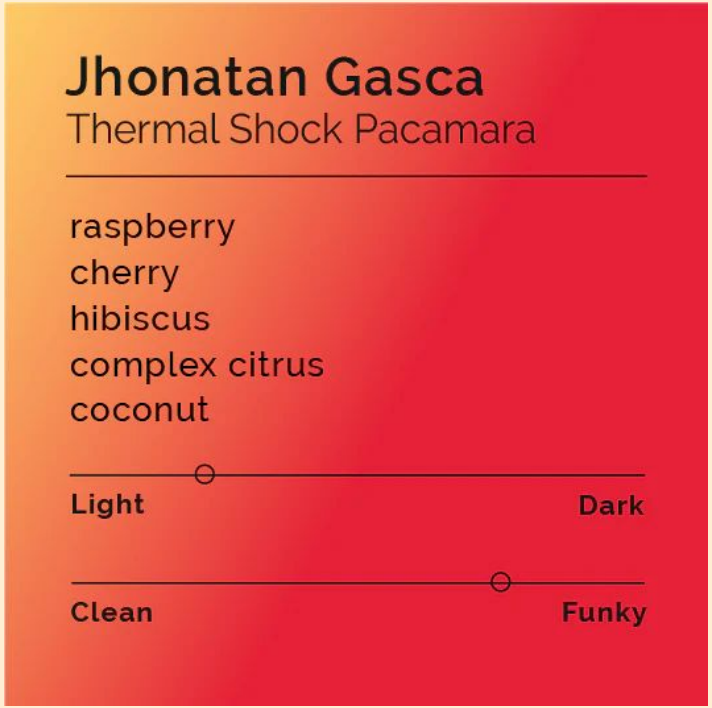
Automatic Coffee Machine

By:
Rio Pacheco
Elijah Knupp
Corrado Govea

Why pour-over coffee?

- Allows for **higher extraction** and more **differentiable flavors** compared to other brewing methods
  - Bean tasting notes become much more apparent allowing them to be **experienced more fully**, think ‘blueberry’ or ‘citrus fruit’
- Highly customizable to taste! Variables such as flow rate, pour pattern, water temperature, and bed agitation can be adjusted to affect the taste of the coffee

- ***TL;DR**: Pour-over coffee delivers a premium, barista-level cup of coffee at home!*



Pictured above: An example offering of coffee beans from Black & White Roasters

Pour-over coffee technique

1) **Grind coffee beans:** Grind 20 grams of coffee beans to a medium-fine consistency
2) **Boil water:** Let it cool for a few moments until it reaches a temperature of around 95°C.
3) **Pre-wet the filter:** Place a paper filter in your coffee dripper and pour some hot water over it to pre-wet it. This helps to eliminate any papery taste and also heats up the dripper.
4) **Bloom Pour:** Place your dripper and pre-wet filter on your coffee cup or carafe. Add the ground coffee to the filter and level it out. Start the timer and pour 50g of water, starting from the center and moving outward in a circular motion. The water should be poured very slowly and continuously for 15 seconds.
5) **Second pour:** Wait for 30 seconds and then pour 70g of water in the same manner as the first.
6) **Third pour:** At about the 1:30 mark, pour another 60g of water.
7) **Fourth pour:** At 2:15 add another 60g of water.
8) **Final pour:** at 3:00 add another 60g of water and cut the brew off at 3:30.

Source: https://samjamescoffeebar.com/blogs/recipes/what-is-the-4-6-method

Pour-over vs other brewing options

- Keurig - pod based coffee machine
  - Produces weak coffee due to small coffee:water ratios
  - Uses plastic pods that produce more environmental waste
- Nespresso - pod based espresso-like machine
  - Limited to espresso sized portions (2-4oz)
  - Pods are expensive
  - Pod coffee is sourced from underpaid and overworked farmers

Pour over

Pros

- Can **use any ground coffee**, allowing people to **support local** coffee shops
- **Low environmental impact** - brewing devices are **reusable** and consumable items are compostable (paper filters and coffee grounds)
- One coffee can be made to taste in a variety of different ways, giving users **flexibility** and **specificity** to brew just **how they want**

Cons

- The skill ceiling is very high, meaning many people may get discouraged if their first few tries aren't good
- Pour-over is not accessible, making it difficult for individuals that struggle with fine motor skills

What currently exists

- Xbloom - Pod-based automated pour over machine
  - Issues: Closed source, expensive(!), **rigid** and **vague** parameters

- Poursteady - Professional multi-cup pour over machine
  - Issues: costs $11,000, geared towards cafe environment

ECE area coverage

- Software
  - UI/UX
  - App development
- Hardware
  - Sensor integration
  - Hardware-Software interface
  - Thermally stable design
- Circuits
  - Analog/Digital conversion
  - PCB design
  - High voltage circuitry

Use-Case Requirements

Goals: precise, controllable, and repeatable

- Make pour over accessible for everyone
  - Using presets on the machine, the user will be able to get a delicious pour-over cup of coffee through **three simple tasks**
    - Pour room-temperature water into the water tank
    - Add coffee grounds
    - Select preset
- What our machine will do
  - Heat the water to the desired temperature (**∓5°F**)
  - Pour a precise amount of water over the coffee grounds (**∓5 - 10 grams**)
  - Have a consistent, highly controlled flow rate throughout the entire pour (**0 g/s - 12 g/s**)
  - Provide an intuitive, straightforward, and accessible experience (**>= 5 preloaded presets)** for a user of any experience-level that can be precisely tuned to the users taste (**temp, pour rate, coffee ground amount, and pour pattern will all be customizable**)
  - Deliver a repeatable, highly consistent cup of coffee between pours (**∓0.5% TDS**)

Technical Challenges

Water control

- The temperature reported to the user must match the water temperature being poured
  - Keeping temperature stability between our heating chamber and the pouring mechanism
- The water flow rate should be consistent throughout the making of the coffee
- Ensuring electrical components are insulated and not exposed to other components such as liquid or heating elements

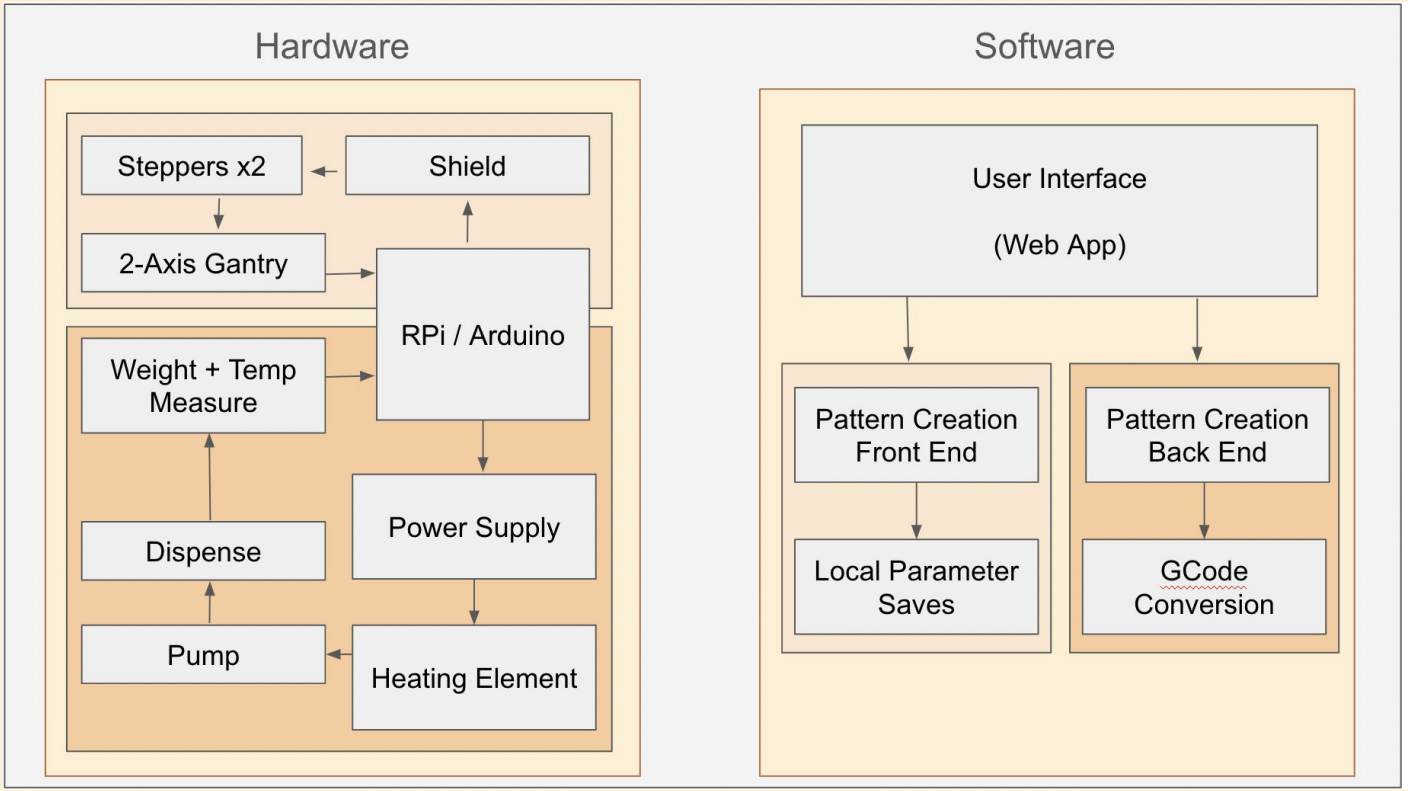
Pouring mechanism control

- Making a robotic system that moves in a precise and steady 2-D manner
- There should be physical fail safes and boundaries to ensure user safety
- The mechanism should be big enough to allow users to use different pour-over devices

User Interface / User Experience

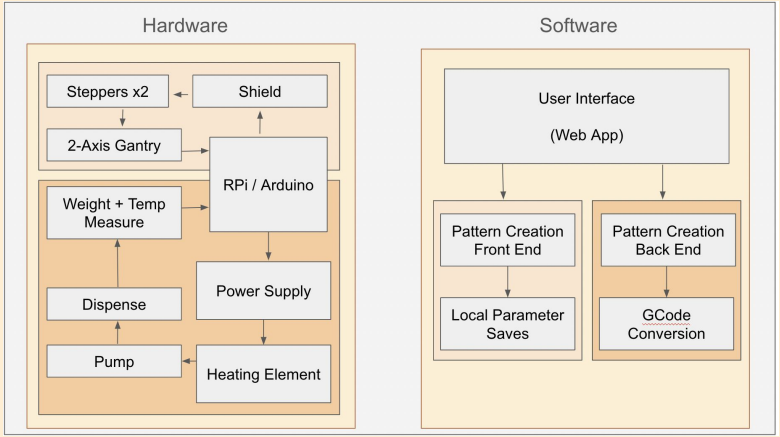
- The design must strike a balance between ease of use and customizability to cater to both audiences
- The user should be able to store customized presets that they previously liked, allowing them to reuse them in the future
- There should be minimal to no user input during the making of the coffee

Solution Approach



Solution Approach (cont.)

- The hardware used will be similar components to those found in 3D printers, stepper motors, belt-driven pulley systems, and guide rails
- Built in sensors will allow us to collect data to test and fine tune our machine
- Data collection and power circuit packaged in PCB.
- All of the electronics will be powered by a Raspberry Pi & Arduino pairing, allowing us to do web app hosting and motor control at the same time



Through the combination of **familiar** hardware and **simple** software, users will have a seamless experience brewing their own pour-over coffee!

- Our software will be a web application to allow users to customize their presets and preferences
- Pattern(s) will be spliced into G-Code for ease of readability by the mechanical system
- The web app will allow users to create and save custom presets, making it easy to repeat previously made presets

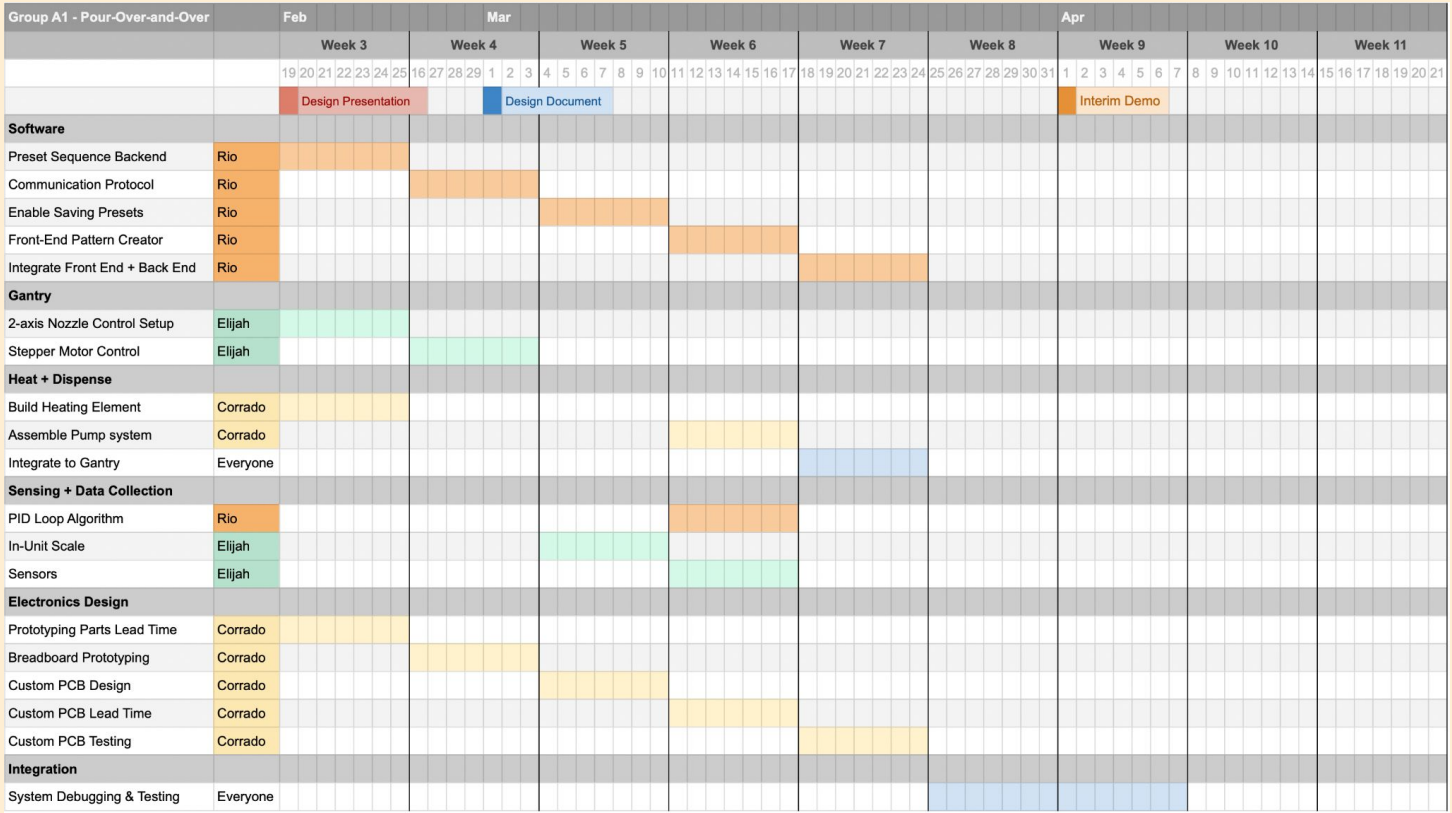
Testing, Verification and Metrics:

Use Case Requirements	**Testing Procedure**
Stable water temperature and flow rate	Use temperature probes and record data during brews to ensure stability
Accurate weight measurements	Compare our built in scale measurements to third-party scales
Consistent extraction yields	Measure each brew with a Brix refractometer to ensure consistent TDS yields

Tasks and Division of Labor:

- Software - Rio
  - Create UI for making presets / sequences
  - Enable saving presets
  - Communication protocol to send GCode
- Gantry (Algorithm focused) - Elijah
  - 2-Axis nozzle control setup
  - Stepper motor control
- Heat + Dispense
  - Heating Element - Corrado
  - Pump system - Corrado
- Sensing + Data Collection
  - PID Loop algorithm - Rio
  - In-unit scale - Elijah
  - Sensors - Elijah
- Electronics Design - Corrado
  - Custom PCB for data collection

Schedule


Group A1 - Pour-Over-and-Over
Feb
Mar
Apr
Week 3
Week 4
Week 5
Week 6
Week 7
Week 8
Week 9
Week 10
Week 11
19 20 21 22 23 24 25 16 27 28 29 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
Design Presentation
Design Document
Interim Demo
Software
Preset Sequence Backend Rio
Communication Protocol Rio
Enable Saving Presets Rio
Front-End Pattern Creator Rio
Integrate Front End + Back End Rio
Gantry
2-axis Nozzle Control Setup Elijah
Stepper Motor Control Elijah
Heat + Dispense
Build Heating Element Corrado
Assemble Pump system Corrado
Integrate to Gantry Everyone
Sensing + Data Collection
PID Loop Algorithm Rio
In-Unit Scale Elijah
Sensors Elijah
Electronics Design
Prototyping Parts Lead Time Corrado
Breadboard Prototyping Corrado
Custom PCB Design Corrado
Custom PCB Lead Time Corrado
Custom PCB Testing Corrado
Integration
System Debugging & Testing Everyone