# **Execution Plan**

| **End Date** | **Cindy** | **Mark** | **Zeeshan** | **Status** |
| --- | --- | --- | --- | --- |
| 03/02/2022 | - Develop database  - Construct sample data  - Create a mock system to gain familiarity | - Order Initial Parts  -Finish IR Camera/ Emitter system to begin ML data collection | - Full application UI prototyped | Done |
| 3/02/2022 | Midterm Presentation | | | Done |
| 03/09/2022 | - Verify dataset has sufficient amount  - Create base python script that adjusts temperature | -Finish water sprinkler system | - Home page functional. Ability to select roast types, cup sizes and begin brewing process | Done |
| 03/23/2022 | - Create a base machine learning algorithm | -Finish heating system  - Finish calibration of sprinkler system | - History page functional. Displays a list of user’s previous brews.  - Settings page functional. Users are able to modify the appearance and adjust profile and notification settings. | Done |
| 03/30/2022 | - Run tests on prediction to create neural network | -Finish calibration of heating system to make stable temperatures | - Entire GUI Tested. Users can move throughout the app without bugs and are able to virtually start the brewing process. | Done |
| 3/30/2022 | Status Update Presentation | | | Done |
| 04/06/2022 | - Run unit tests and check edge cases | - Finish power system | - App is able to connect to the database and retrieve sample data. | Done |
| 04/13/2022 | - Connect ML with Database  - Run final tests | -Write python script to follow values given by database | - App is able to connect to the Raspberry Pi and turn on an LED | Done |
| 04/20/2022 | - Revise any issues with neural network and algorithm | -Testing of a full brew cycle  -Install and test fault devices | - Final testing completed  - App fully functional | Done |
| 04/30/2022 | Final Presentation, Demo, and Report | | | Done |

# **Validation Plan**

| **Validation Plan** | | | | |
| --- | --- | --- | --- | --- |
| **Test Name** | **Success Criteria** | **Methodology** | **Status** | **Responsible Engineer(s)** |
| Functional User Interface | User is able to navigate throughout all pages within the application without bugs, crashes, or need for outside intervention. | Open the application and handle it as an end user. | Passed | Zeeshan Virani |
| Android Application Storage | The Android application of the system shall not exceed 15 megabytes of storage. | Monitor frontend design images and functions size. | Passed | Zeeshan Virani |
| Application Connection With Raspberry Pi | The Raspberry Pi turns on an LED when a connection is established between the android application and itself. | Visually Verify connection has been established. | Passed | Zeeshan Virani |
| Database Connection With Application | The application is able to send and receive data from the database. | Send data to the database and then query to see if that data has been stored. | Passed | Zeeshan Virani |
| Brew Temperature Prediction | The machine learning algorithm is able to receive feedback from the user and can adjust the temperature up or down if the feedback is negative. | Data will have linear relationship between user feedback ratings and temperature to each user. If user has positive feedback, the temperature will stay constant. If user has negative feedback on too hot or too cold, the temperature will adjust in the direction that generates a positive outcome for the user. | Passed | Cindy Ho |
| Database Storage Amount | The database shall not exceed 65,535 bytes or 8.61 GB (Raspberry Pi maximum storage). | Return error output if the user tries to exceed allocated storage space. | Passed | Cindy Ho |
| Database Storage Savings | The database is able to save user feedback ratings, temperature, and water saturation requirements. | Running the training data and checking if the database saves all of the data with no null responses. | Passed | Cindy Ho |
| Water Saturation Prediction Accuracy | The machine learning determines the percentage of ground bean saturation within +/- 10% range. | Run initial training data and two subsequent train data tests through (train/test splits). Have a confusion matrix to compare results. | Passed | Cindy Ho |
| Input Voltage (Peripherals) | The input voltage level for the converter used for the Raspberry Pi, Sensor Array, and Water Distribution Valve will convert 120VAC to 5 VDC at at least 5 Amps. | Use a multimeter and a test load to validate input and output voltage levels. | Passed | Mark Golla |
| Mass | Mass of all hardware shall not exceed 15 lbs. | Measure the entire system utilizing a scale. | Untested | Mark Golla |
| Volume | The size of the full system shall not exceed 48x48x48 inches. | Measure dimensions of system | Untested | Mark Golla |
| Basic Sensor Readings | Test the temperature sensor against a known working thermometer at 200 F and ensure CO2 value increases when breath or co2 from a duster is introduced | Test the CO2 and temperature sensors by connecting to the pi and writing scripts to receive their values | Passed | Mark Golla |
| Multispectral Camera | Successful capture of an image at each NIR wavelength and its compilation into a multidimensional image within less than one second. | The camera should be able to cycle through wavelengths and capture a full image of a white background as well as coffee grounds every second for 2 minutes. The images should be consistent and the light intensity should not saturate the camera | Passed | Mark Golla |
| Water Heating System | The temperature of water in the reservoir should be able to be held at a constant temperature anywhere in the range of 185 degrees F to 205 degrees F for 30 seconds without fluctuating more than 2 degrees | The temperature values will be swept in increments of one from 185 to 205 F in 5 F intervals and held at each step for 30 seconds | Passed | Mark Golla |
| Fault System | The system should not start brewing or stop actively brewing if there is a fault present | Each fault condition will be implemented individually and in parallel both while the system is running or about to be run and observe whether it attempts to brew | Passed | Mark Golla |
| Brew Quantity | The amount of water dispensed should be within 10% of what was requested. | Run the hardware with pre defined quantities and measure the amount of water dispensed to verify the dispensing mechanism | Failed | Mark Golla |
| Full System Demo | A user of the system can select brew type and cup size according to their needs and make a cup of coffee. | The system is able to complete when the camera reads the grounds to have reached the required saturation levels, the temperature sensor reads all of the temperature for water maintained, and CO2 levels are matched according to specifications. | Untested | Everyone |