IOT based Smart Plant Care System

Team #6

Team Members

ISE:

824 DEUJA RITIK (듀야리틱)
306 KHOSHIMOV ASADBEK FAYZULLA UGLI (아삳벡)
311 AKHMADOV AZIZBEK AKHMAD UGLI(아지즈벡)

IBT:

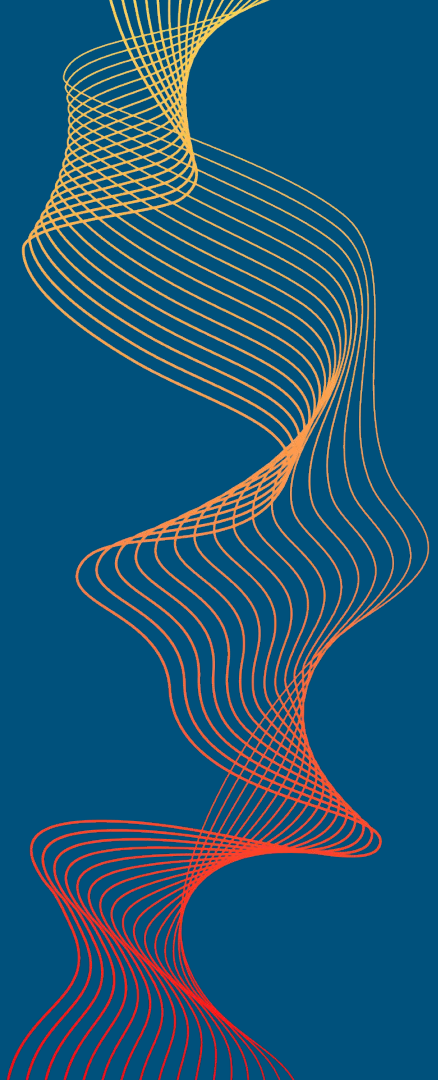
938 TAMANG BABINA (타망바비나)
341 Shokhrukh Khayitov
927 TULKINHUJAEV JASURBEK JOBIRBEK UGLI (자수르벡)
750 ORIPOV ULUGBEK (어르포브울룩베크)
334 ABDULLAJONOV JALOLIDDIN OYBEK UGLI(잘럴도)

Content:

1. Research Method
2. Process
3. Technology
4. Management
    (1) Production
    (2) Service
5. Distribution and logistics
6. Price
7. Promotion: Marketing
8. Technology innovation
9. Product innovation and legal aspect

Research Method for Smart Plant Care System

An outline of the research method for the development of a smart plant care system using Raspberry Pi and other hardware components.



Literature Review

Review of current smart greenhouse technologies and Raspberry Pi-based solutions

Focus on features like people counting, plant monitoring, face detection, and disease detection in plants

Identify gaps in existing solutions and opportunities for improvement

Requirements Analysis

01 Consider desired functionalities and necessary hardware components

02 Analyze the requirements for the proposed smart plant care system

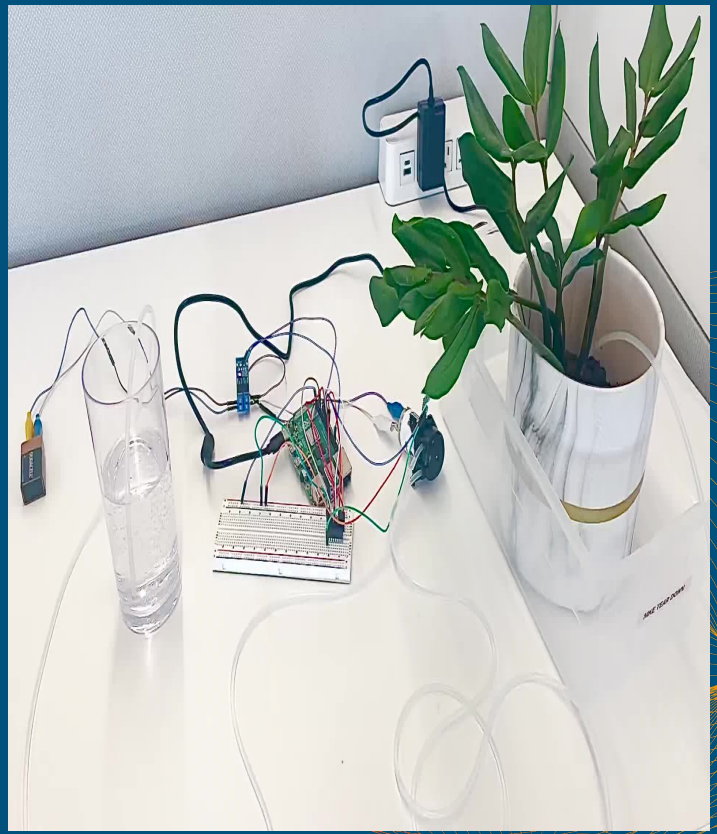
System Design and Architecture

- Collaboratively design a modular system, including sensor selection, Raspberry Pi and camera module integration, communication protocols, data processing algorithms, and smartphone application development

Prototype Development and Implementation

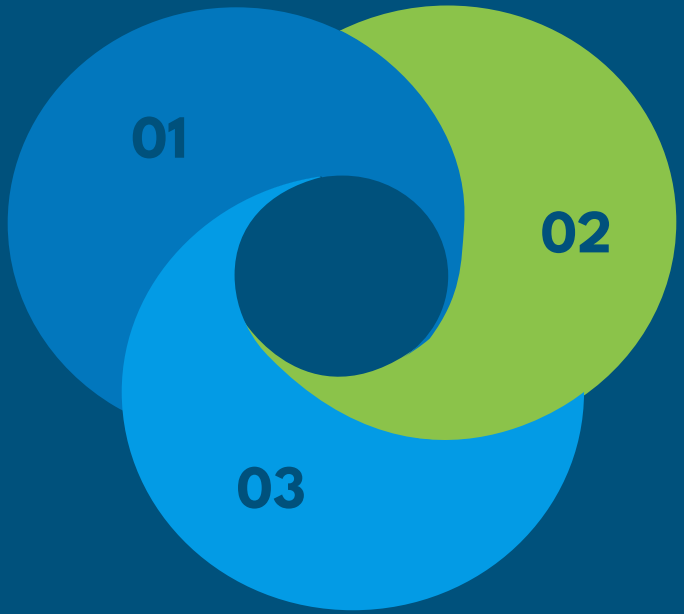
01 Implement the prototype in a greenhouse setting for initial testing and evaluation

02 Develop a prototype of the smart plant care system, encompassing hardware components and software elements



Data Collection and Analysis

Collect data to assess the system's performance



Covering people counting, plant monitoring, face detection, and disease detection in plants

Analyze the data to evaluate the system's accuracy, efficiency, and user-friendliness

System Evaluation and Comparison

01 Assess its potential for adoption and integration into current greenhouse management practices

02 Evaluate the proposed system in relation to existing Raspberry Pi-based solutions

03 Identify the strengths and weaknesses of the proposed system

Opportunities and Market Analysis

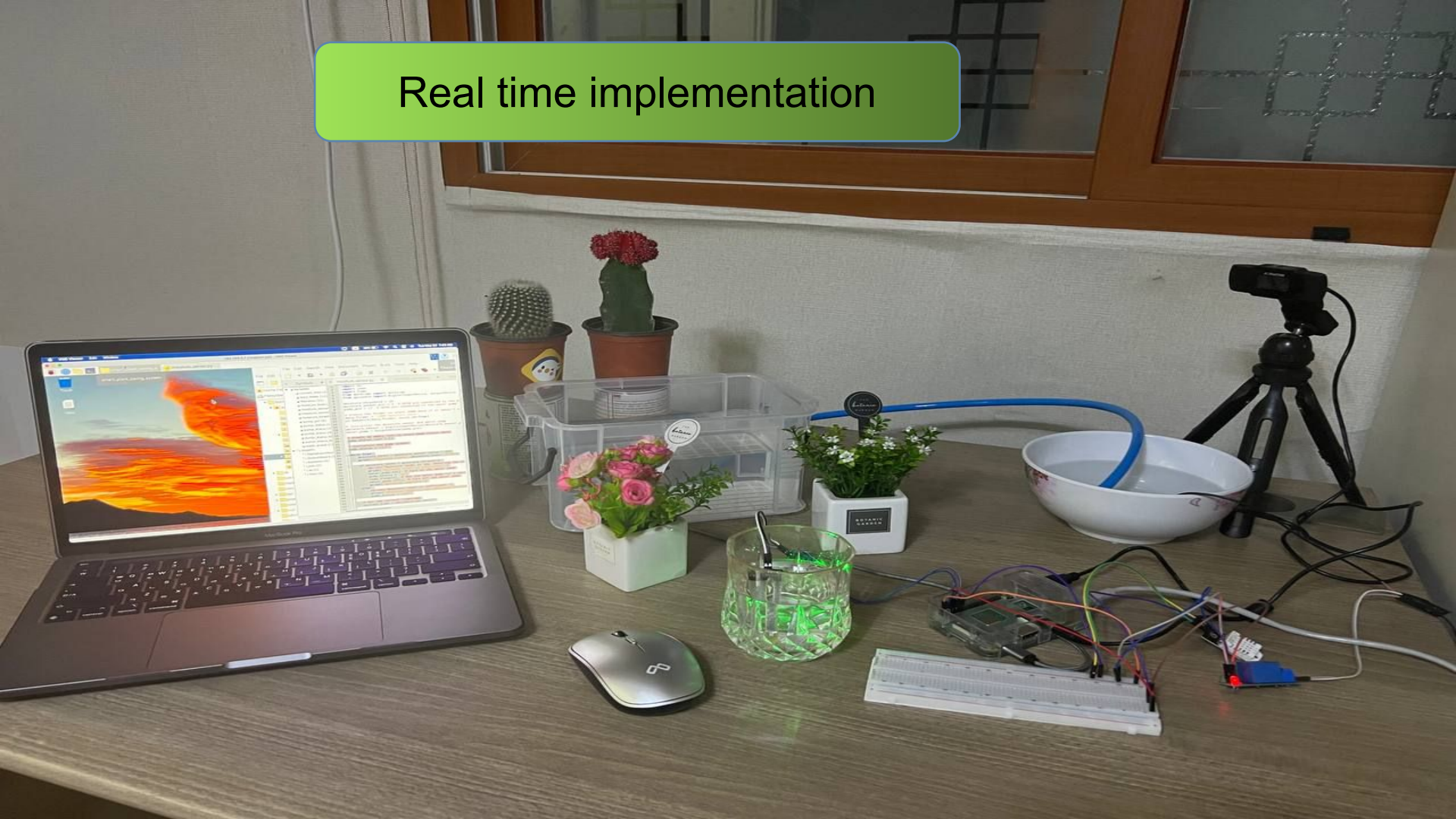
- Investigate the market opportunities for the developed Raspberry Pi-based smart plant care system
- Consider its unique features, benefits, and potential applications
- Explore potential collaborations with industry stakeholders
- Devise a marketing strategy for promoting the system

ISE Capstone Project

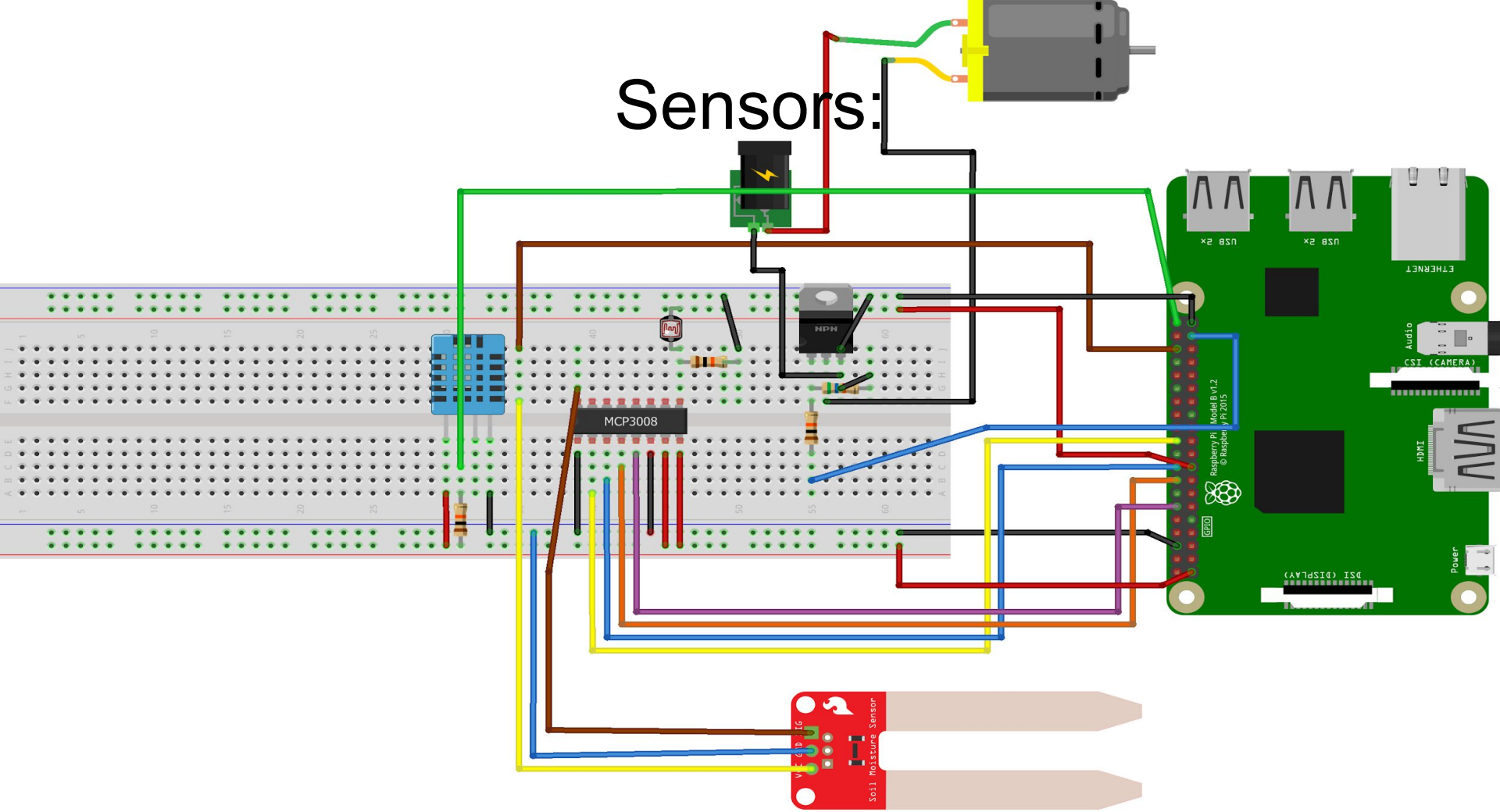
Smart Plant Caring System

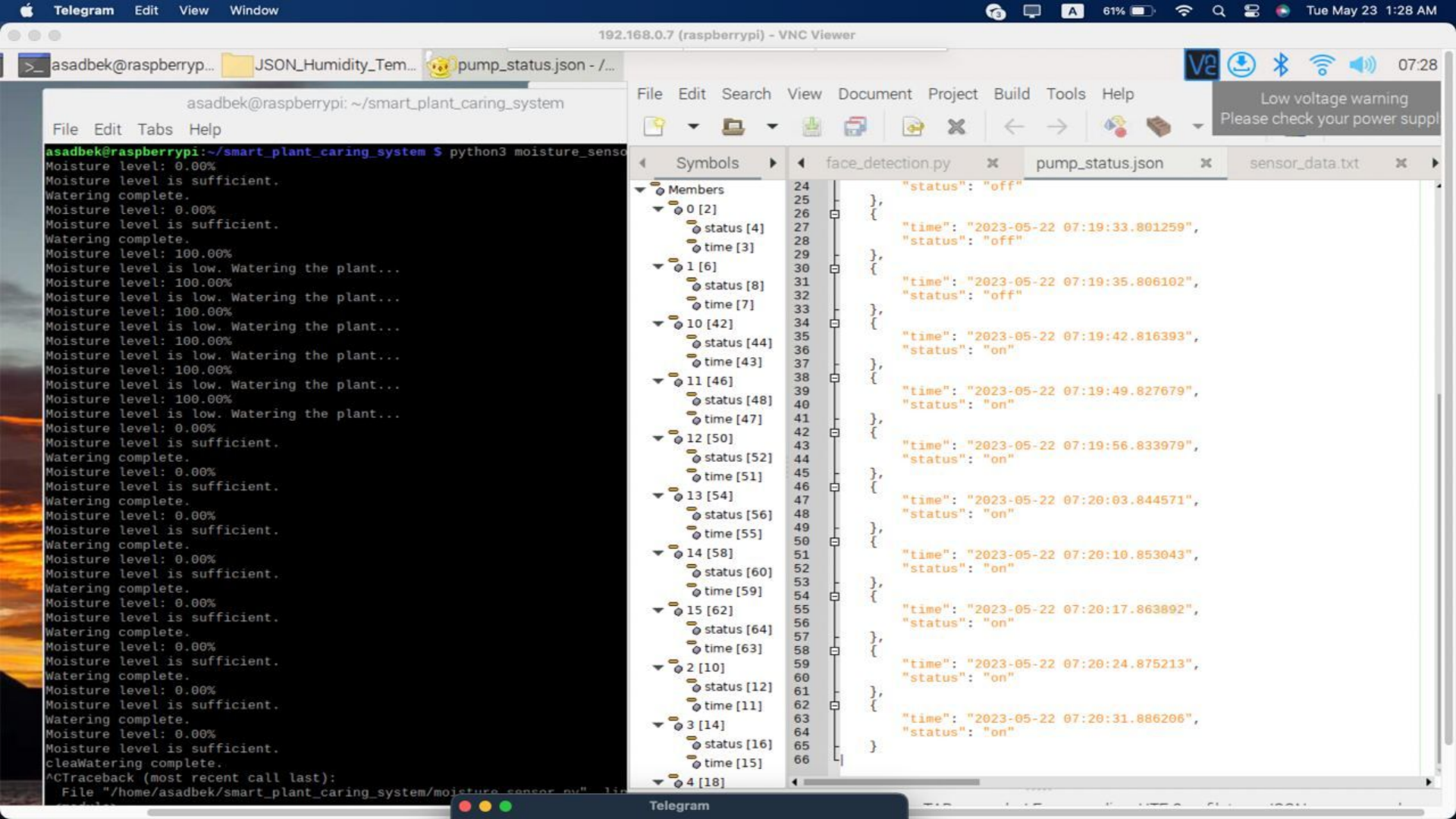
Name: Khoshimov Asadbek
ID: 12200306

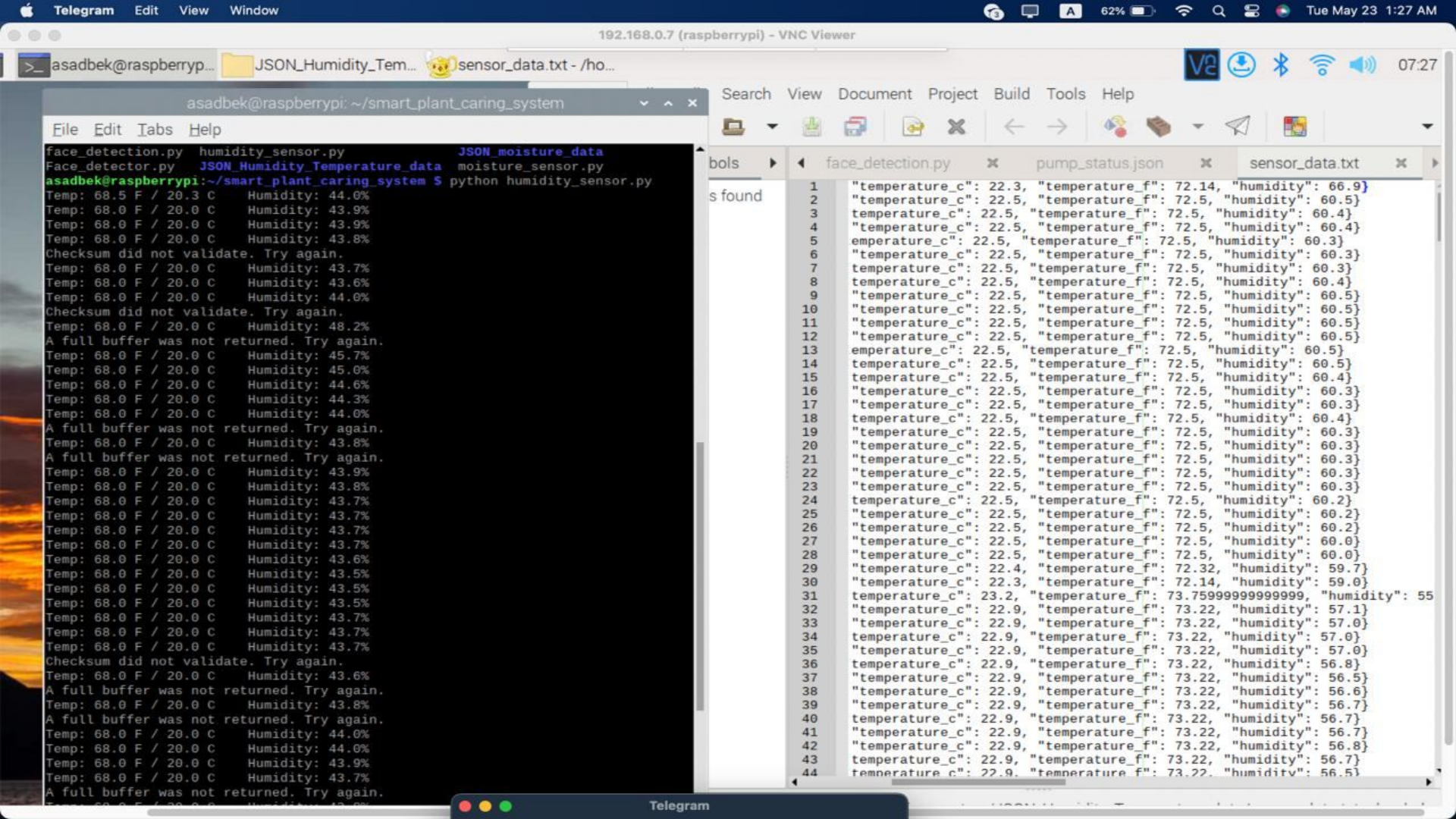
Real time implementation

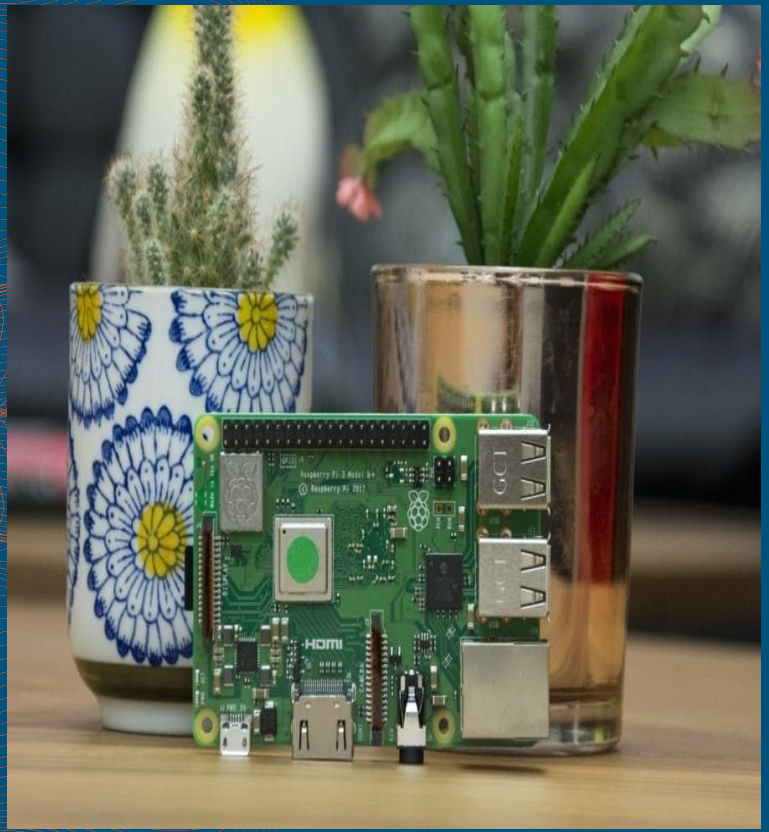


Sensors:


MCP3008
NPN
Soil Moisture Sensor
USB 2x
USB 2x
ETHERNET
Audio
CSI (CAMERA)
HDMI
DSI (DISPLAY)
Power
GPIO
Raspberry Pi Model B v1.2
© Raspberry Pi 2015
fritzing


Telegram Edit View Window
Tue May 23 1:28 AM
192.168.0.7 (raspberrypi) - VNC Viewer
asadbek@raspberryp...
JSON_Humidity_Tem...
pump_status.json - /...
07:28
asadbek@raspberrypi: ~/smart_plant_caring_system
File Edit Tabs Help
asadbek@raspberrypi:~/smart_plant_caring_system $ python3 moisture_senso
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 100.00%
Moisture level is low. Watering the plant...
Moisture level: 100.00%
Moisture level is low. Watering the plant...
Moisture level: 100.00%
Moisture level is low. Watering the plant...
Moisture level: 100.00%
Moisture level is low. Watering the plant...
Moisture level: 100.00%
Moisture level is low. Watering the plant...
Moisture level: 100.00%
Moisture level is low. Watering the plant...
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 0.00%
Moisture level is sufficient.
Watering complete.
Moisture level: 0.00%
Moisture level is sufficient.
cleaWatering complete.
^CTraceback (most recent call last):
File "/home/asadbek/smart_plant_caring_system/moisture_sensor.py"
File Edit Search View Document Project Build Tools Help
Low voltage warning
Please check your power suppl
Symbols
face_detection.py
pump_status.json
sensor_data.txt
Members
0 [2]
status [4]
time [3]
1 [6]
status [8]
time [7]
10 [42]
status [44]
time [43]
11 [46]
status [48]
time [47]
12 [50]
status [52]
time [51]
13 [54]
status [56]
time [55]
14 [58]
status [60]
time [59]
15 [62]
status [64]
time [63]
2 [10]
status [12]
time [11]
3 [14]
status [16]
time [15]
4 [18]
24 "status": "off"
25 },
26 {
27 "time": "2023-05-22 07:19:33.801259",
28 "status": "off"
29 },
30 {
31 "time": "2023-05-22 07:19:35.806102",
32 "status": "off"
33 },
34 {
35 "time": "2023-05-22 07:19:42.816393",
36 "status": "on"
37 },
38 {
39 "time": "2023-05-22 07:19:49.827679",
40 "status": "on"
41 },
42 {
43 "time": "2023-05-22 07:19:56.833979",
44 "status": "on"
45 },
46 {
47 "time": "2023-05-22 07:20:03.844571",
48 "status": "on"
49 },
50 {
51 "time": "2023-05-22 07:20:10.853043",
52 "status": "on"
53 },
54 {
55 "time": "2023-05-22 07:20:17.863892",
56 "status": "on"
57 },
58 {
59 "time": "2023-05-22 07:20:24.875213",
60 "status": "on"
61 },
62 {
63 "time": "2023-05-22 07:20:31.886206",
64 "status": "on"
65 }
66
Telegram


Telegram Edit View Window
62% Tue May 23 1:27 AM
192.168.0.7 (raspberrypi) - VNC Viewer
asadbek@raspberryp...
JSON_Humidity_Tem...
sensor_data.txt - /ho...
07:27
asadbek@raspberrypi: ~/smart_plant_caring_system
File Edit Tabs Help
face_detection.py humidity_sensor.py JSON_moisture_data
Face_detector.py JSON_Humidity_Temperature_data moisture_sensor.py
asadbek@raspberrypi:~/smart_plant_caring_system $ python humidity_sensor.py
Temp: 68.5 F / 20.3 C Humidity: 44.0%
Temp: 68.0 F / 20.0 C Humidity: 43.9%
Temp: 68.0 F / 20.0 C Humidity: 43.9%
Temp: 68.0 F / 20.0 C Humidity: 43.8%
Checksum did not validate. Try again.
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Temp: 68.0 F / 20.0 C Humidity: 43.6%
Temp: 68.0 F / 20.0 C Humidity: 44.0%
Checksum did not validate. Try again.
Temp: 68.0 F / 20.0 C Humidity: 48.2%
A full buffer was not returned. Try again.
Temp: 68.0 F / 20.0 C Humidity: 45.7%
Temp: 68.0 F / 20.0 C Humidity: 45.0%
Temp: 68.0 F / 20.0 C Humidity: 44.6%
Temp: 68.0 F / 20.0 C Humidity: 44.3%
Temp: 68.0 F / 20.0 C Humidity: 44.0%
A full buffer was not returned. Try again.
Temp: 68.0 F / 20.0 C Humidity: 43.8%
A full buffer was not returned. Try again.
Temp: 68.0 F / 20.0 C Humidity: 43.9%
Temp: 68.0 F / 20.0 C Humidity: 43.8%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Temp: 68.0 F / 20.0 C Humidity: 43.6%
Temp: 68.0 F / 20.0 C Humidity: 43.5%
Temp: 68.0 F / 20.0 C Humidity: 43.5%
Temp: 68.0 F / 20.0 C Humidity: 43.5%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
Checksum did not validate. Try again.
Temp: 68.0 F / 20.0 C Humidity: 43.6%
A full buffer was not returned. Try again.
Temp: 68.0 F / 20.0 C Humidity: 43.8%
A full buffer was not returned. Try again.
Temp: 68.0 F / 20.0 C Humidity: 44.0%
Temp: 68.0 F / 20.0 C Humidity: 44.0%
Temp: 68.0 F / 20.0 C Humidity: 43.9%
Temp: 68.0 F / 20.0 C Humidity: 43.7%
A full buffer was not returned. Try again.
Search View Document Project Build Tools Help
bols
s found
face_detection.py
pump_status.json
sensor_data.txt
1 "temperature_c": 22.3, "temperature_f": 72.14, "humidity": 66.9}
2 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.5}
3 temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.4}
4 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.4}
5 emperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
6 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
7 temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
8 temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.4}
9 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.5}
10 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.5}
11 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.5}
12 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.5}
13 emperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.5}
14 temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.5}
15 temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.4}
16 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
17 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
18 temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.4}
19 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
20 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
21 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
22 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
23 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.3}
24 temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.2}
25 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.2}
26 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.2}
27 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.0}
28 "temperature_c": 22.5, "temperature_f": 72.5, "humidity": 60.0}
29 "temperature_c": 22.4, "temperature_f": 72.32, "humidity": 59.7}
30 "temperature_c": 22.3, "temperature_f": 72.14, "humidity": 59.0}
31 temperature_c": 23.2, "temperature_f": 73.75999999999999, "humidity": 55
32 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 57.1}
33 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 57.0}
34 temperature_c": 22.9, "temperature_f": 73.22, "humidity": 57.0}
35 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 57.0}
36 temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.8}
37 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.5}
38 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.6}
39 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.7}
40 temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.7}
41 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.7}
42 "temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.8}
43 temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.7}
44 temperature_c": 22.9, "temperature_f": 73.22, "humidity": 56.5}
Telegram



Technology

Sensors
Motors
Hardware Architecture

FUNCTIONS OF SENSORS

AZIZBEK AKHMADOV

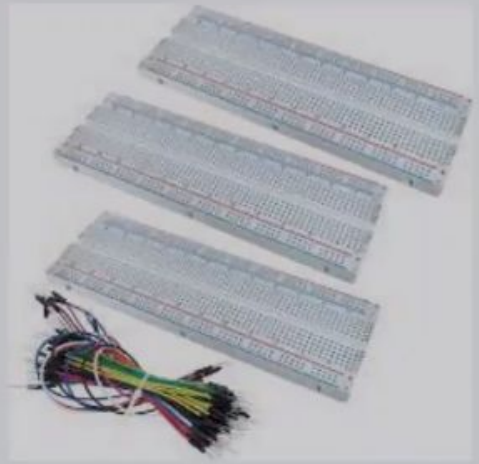
ARDUINO RELAY 1 CHANNEL B63:

- THE ARDUINO RELAY IS AN ELECTROMECHANICAL SWITCH THAT ALLOWS YOU TO CONTROL HIGH-POWER DEVICES USING A LOW-POWER SIGNAL FROM AN ARDUINO OR ANY MICROCONTROLLER.
- IT PROVIDES A SINGLE CHANNEL FOR CONTROLLING A CONNECTED DEVICE, SUCH AS LIGHTS, MOTORS, OR OTHER ELECTRICAL APPLIANCES.
- YOU CAN USE IT TO SWITCH ON OR OFF THE CONNECTED DEVICE BASED ON INPUT SIGNALS FROM THE ARDUINO.

ARDUINO BREADBOARD ELECTRONIC BREADBOARD 830 HOLES (MB-102):

THE ARDUINO BREADBOARD, ALSO KNOWN AS AN ELECTRONIC BREADBOARD, IS A PROTOTYPING TOOL USED TO BUILD AND TEST ELECTRONIC CIRCUITS.

- IT PROVIDES A GRID OF INTERCONNECTED HOLES, ALLOWING YOU TO INSERT AND CONNECT VARIOUS ELECTRONIC COMPONENTS WITHOUT SOLDERING.
- THE 830 HOLES ON THE BREADBOARD PROVIDE AMPLE SPACE FOR PLACING AND CONNECTING COMPONENTS SUCH AS RESISTORS, CAPACITORS, LEDS, AND MORE.
- IT IS COMMONLY USED WITH ARDUINO BOARDS TO CREATE AND TEST CIRCUITS BEFORE PERMANENT SOLDERING ON A PCB (PRINTED CIRCUIT BOARD).



ARDUINO MINI WATER PUMP MOTOR DM151:

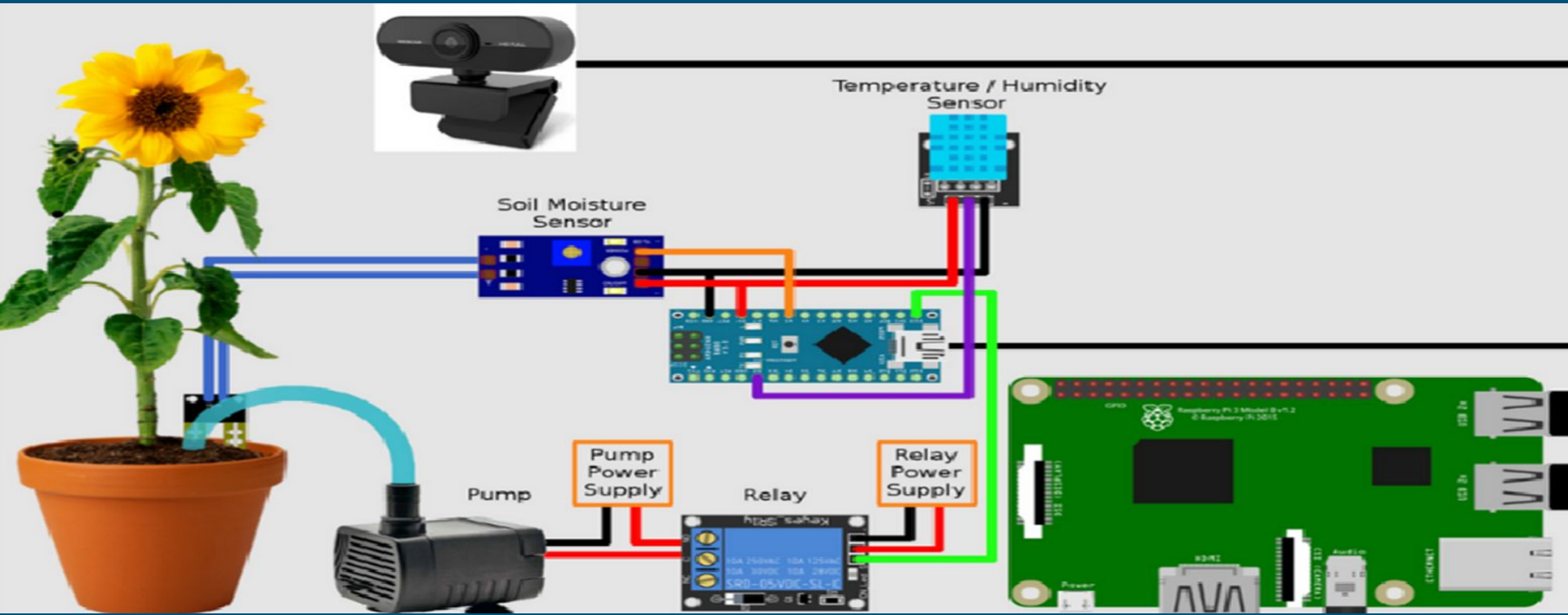
THE ARDUINO MINI WATER PUMP MOTOR DM151 IS A SMALL WATER PUMP DESIGNED FOR USE WITH ARDUINO AND OTHER MICROCONTROLLER PLATFORMS.

- IT IS TYPICALLY USED FOR PROJECTS INVOLVING WATER CIRCULATION, WATER COOLING, OR WATER-BASED EXPERIMENTS.
- THE MOTOR CAN BE CONTROLLED USING DIGITAL OUTPUT PINS OF THE ARDUINO BOARD TO TURN IT ON OR OFF, CONTROLLING THE WATER FLOW.

SUMP WATER PUMP:

THE PRIMARY FUNCTION OF A SUMP PUMP MOTOR IS TO POWER THE SUMP PUMP, WHICH IS A DEVICE DESIGNED TO REMOVE WATER FROM BASEMENTS, CRAWL SPACES, AND OTHER AREAS PRONE TO FLOODING. THE MOTOR IS RESPONSIBLE FOR DRIVING THE PUMP'S IMPELLER, WHICH CREATES A CENTRIFUGAL FORCE THAT PUSHES WATER OUT OF THE SUMP PIT AND AWAY FROM THE BUILDING.

Overall Process Design



Management

Production

Smart plant care system is built with hardware components such as Raspberry Pi, Camera modules, sensors and controllers.The initial cost are cheaper compared to other applications.

Service

-Smart Irrigation
-Human Detection
- Integrated Monitoring
-Security

Distribution Channel

Indirect Distribution Channel

Direct Distribution Channel

-Internet selling

-Mail ordering

HYBRID Distribution Channel

Logistics

Outsourced Logistics Strategy

Third- party Logistics,(3PL)
-Transportation
-Warehousing
-Order fulfillment
-Inventory management

Team 6

Pricing Model

Shokhrukh Khayitov 12200341

Value-based Pricing

Value-based pricing is often a suitable strategy for innovative products like a Smart Plant Care System.

Identify Key Value Drivers:

1) Real-time plant monitoring and alerts
2) Automated watering and fertilization
3) Disease and pest detection
4) Mobile app control and insights

Upselling and Add-ons:

Offering **additional products or services** that complement the Smart Plant Care System to increase the transaction value. For example, **plant nutrition supplements, premium customer support, or consulting services for advanced gardening techniques**.

Pricing Levels:

$19.9	$29.9	$39.9
BASIC	STANDARD	PREMIUM

Pricing Tiers:

- Basic Tier: Includes essential monitoring and basic control features.
- Standard Tier: Offers advanced plant care features and customization options.
- Premium Tier: Provides premium features, personalized support, and exclusive benefits.

Continuous Evaluation and Adjustment:

Monitoring customer feedback, market trends, and competitors' offerings. Adjust the pricing tiers, features, or pricing levels as needed to align with customer preferences and market dynamics.

Price

Based on market analysis, a reasonable profit margin for this product would be

35%

Target price:

Variable cost per product = $188

Desired profit margin as a decimal = 0.35

Target price = $290 per unit

Consider fixed costs:

Fixed costs:

Storage: $500 per month x 12 months = $6,000

Insurance: $1,000 per year

Accountant: $2,000 per year

Break-even point (units):

Fixed costs = $9,000

Sales price per unit = $290

Variable cost per unit = $188

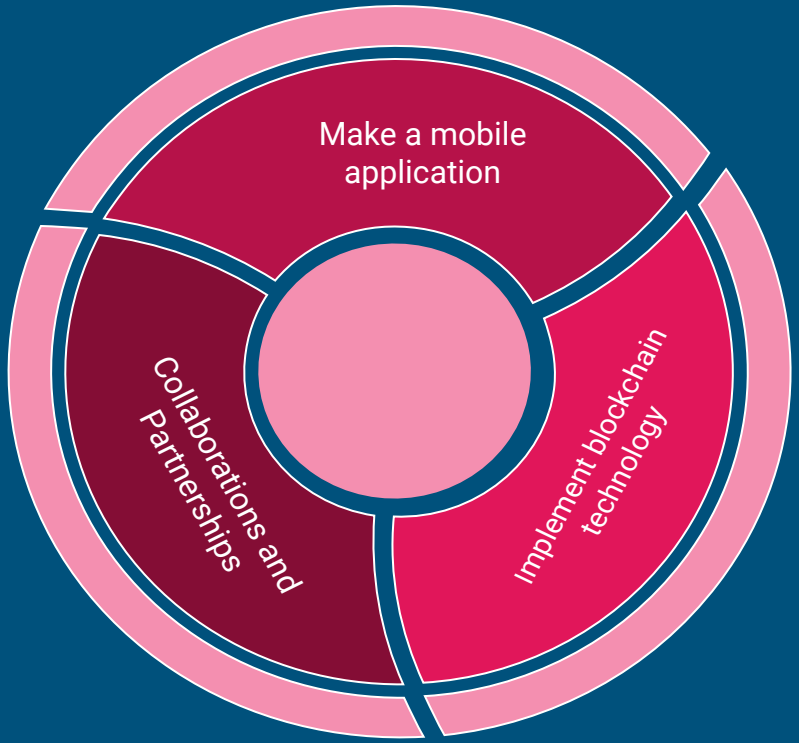
Promotion for Smart Plant Care System

Collaboration

Accessibility

01	**Starter kits**	• Beginners • Advanced Systems
02	Discounts	• Reduction cost • Special Promotions
03	**Bundled deals**	• Gardeners • Early adopters

Remaining steps



Technology Innovation

Raspberry Pi smart plant care system

Some suggestions for how to advance this technology:

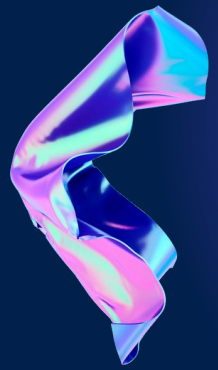
- **Increase the number of plant species supported**
- **Bring AI capabilities together**
- **Enable remote control and access**

Jaloliddin
12200334

Remaining steps

Make a mobile application

Implement blockchain technology



Collaborations and Partnerships:

Product innovation and legal aspect

Legal Aspect

Legal aspects

International Property Rights

Environmental Regulations

Health & Safety Regulations

Liability

Thank you for your time and attention 🙂