**Implementation**

**Step 1:** Setup Telegram Channel

1. Open Telegram application and click on the "Pencil" icon in the bottom-right corner.
2. Click on "New Channel".
3. Enter a Name and Description for your channel.
4. In the next screen, set the channel as "Public" and enter a permanent link for your channel.
5. This link will be used later on to send alerts to this channel.

**Step 2:** Create a Bot

1. Click on the "Magnifying Glass" icon to search for the bot named "BotFather".
2. Click on the verified account of BotFather and type "**/start**" and send the message.
3. Then type "**/newbot**" to create a new bot.
4. Follow the instructions specified in the chat and enter the Bot name and Bot username.
5. After successfully creating a bot, it will provide you with A Token to access the HTTP API. Save it, as it will be used later on the the python code.

**Step 3:** Link the Telegram Channel and the Bot.

1. Open your new Telegram Channel. Click on the channel name and go to Administrators tab.
2. Here, search for your newly created bot and select it.

**NOTE:** You have successfully created a telegram channel and a bot for your temperature monitoring system.

**Step 4:** Setup the Bolt Python Library

1. Open your Ubuntu Server. It can be VMware, Digital Ocean or Virtual Machine Box. You can also open the terminal in Ubuntu Operating System.

**NOTE:** In case you do not have these software or Operating System then you can download them for free on the internet.

2. Type in the following commands.

sudo apt-get -y update #update packages on Ubuntu

sudo apt install python3-pip #install python3 pip3

sudo pip3 install boltiot #install boltiot library using pip

## Reading Sensor Values from Bolt Device

* 1. Create a new directory.

sudo mkdir temp\_alert #new directory

cd temp\_alert #move into directory

* 1. Create new file to hold the configurations of boltiot and Telegram.

sudo nano conf.py #new file

bolt\_api\_key = "XXXX"

# This is your Bolt Cloud API Key

device\_id = "XXXX"

# This is the device ID and will be similar to BOLTXXXX where XXXX is some numbers

telegram\_chat\_id = "@XXXX"

# This is the channel ID of the created Telegram channel. Paste after @ symbol.

telegram\_bot\_id = "botXXXX"

# This is the bot ID of the created Telegram Bot. Paste after bot text.

max\_threshold = 387

min\_threshold = 233

# Threshold beyond which the alert should be sent

**NOTE:** To save and exit from file, press "CTRL+X" and then press "Y". Hit "ENTER".

* 1. Create a new python file.

sudo nano telegram\_alert.py #new file

import requests # for making HTTP requests

import json # library for handling JSON data

import time # module for sleep operation

from boltiot import Bolt # importing Bolt from boltiot module

import conf # config file

mybolt = Bolt(conf.bolt\_api\_key, conf.device\_id)

def get\_sensor\_value\_from\_pin(pin):

"""Returns the sensor value. Returns -999 if request fails"""

try:

response = mybolt.analogRead(pin)

data = json.loads(response)

if data["success"] != 1:

print("Request not successfull")

print("This is the response->", data)

return -999

sensor\_value = int(data["value"])

return sensor\_value

except Exception as e:

print("Something went wrong when returning the sensor value")

print(e)

return -999

## Send Message via Telegram

Append this code to "telegram\_alert.py" to send message via Telegram.

def send\_telegram\_message(message):

"""Sends message via Telegram"""

url = "https://api.telegram.org/" + conf.telegram\_bot\_id + "/sendMessage"

data = {"chat\_id": conf.telegram\_chat\_id, "text": message}

try:

response = requests.request("POST", url, params=data)

print("This is the Telegram URL")

print(url)

print("This is the Telegram response")

print(response.text)

telegram\_data = json.loads(response.text)

return telegram\_data["ok"]

except Exception as e:

print("An error occurred in sending the alert message via Telegram")

print(e)

return False

## Compare Sensor Values with Threshold and Sending Alert Messages

Append this code to "telegram\_alert.py" to check if the temperature is within the range. If not, sends alert.

while True:

# Step 1

sensor\_value = get\_sensor\_value\_from\_pin("A0")

print("The current sensor value is: ", sensor\_value)

sensor\_value\_temp=(100\*sensor\_value)/1024

print("The current temperature is: ",sensor\_value\_temp)

# Step 2

if sensor\_value == -999:

print("Request was unsuccessfull. Skipping.")

time.sleep(10)

continue

# Step 3

if sensor\_value >= conf.max\_threshold:

print("Sensor value has exceeded the maximum threshold")

message = "Alert! Sensor value has exceeded the maximum threshold of " + str(conf.max\_threshold) + \

". The current value is " + str(sensor\_value)+\

". Your butterflies are at a temperature of

"+str(sensor\_value\_temp)+\". Save them!"

telegram\_status = send\_telegram\_message(message)

print("This is the Telegram status:", telegram\_status)

if sensor\_value >= conf.min\_threshold:

print("Sensor value has gone below the minimum threshold")

message = "Alert! Sensor value has gone below the minimum threshold of " + str(conf.min\_threshold) + \

". The current value is " + str(sensor\_value)+\

". Your butterflies are at a temperature of

"+str(sensor\_value\_temp)+\". Save them!"

telegram\_status = send\_telegram\_message(message)

print("This is the Telegram status:", telegram\_status)

# Step 4

time.sleep(10)

**NOTE:** Save and exit by pressing "CTRL+X" and then "Y". Hit "ENTER".

**Step 5:** Run the code.

sudo python3 telegram\_alert.py