Coding and Solution

Team ID	PNT2022TMID38427
Project Name	Real-time river water quality monitoring and control system

Utilization Of Algorithms

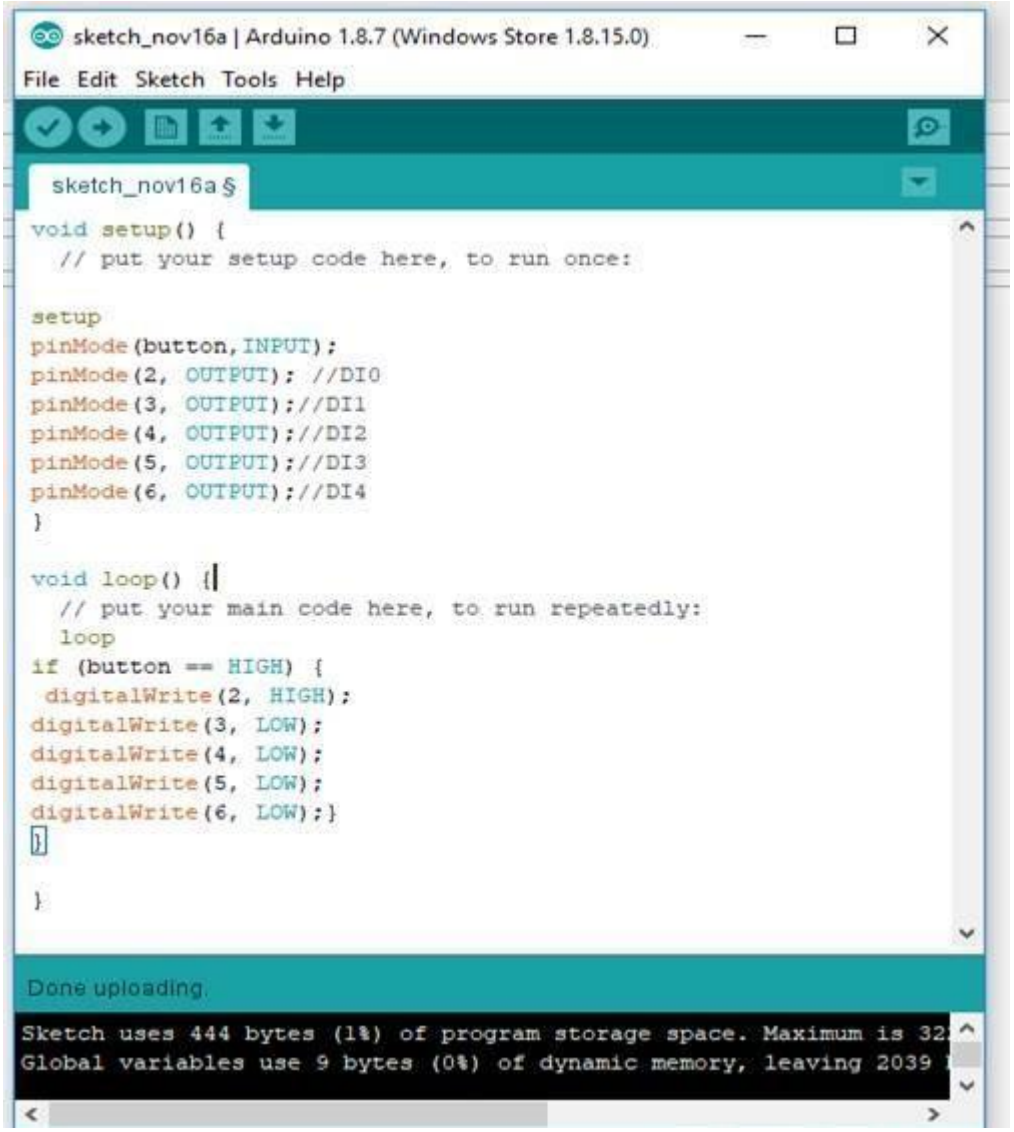
1. We are collecting the data from the sensor nodes.
2. We have to setup the IBM cloud connection configuration in Node-RED platform
3. Then it can connect the IBM Watson IoT with Node-RED platform
4. Then the data are transferred to IBM Watson IoT platform
5. We have to design and develop the app for our needed works.
6. And connect the app with Node-RED
7. So, it can easily show the real time water's pH and Turbidity values in our mobile app
8. If we want to close the particular dam, we needed motor controller.
9. So, we made a motor controller in our own mobile app.
10. The controller's results are shown in Node-RED

Dynamic Program

```
void setup()
{
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
  wifiConnect();
  mqttConnect();
}

void loop() {
  bool isNearby = dist < 100;
  digitalWrite(led, isNearby);

  publishData();
  delay(500);

  if (!client.loop()) {
```

```
mqttConnect();
```

```
  }
}
```

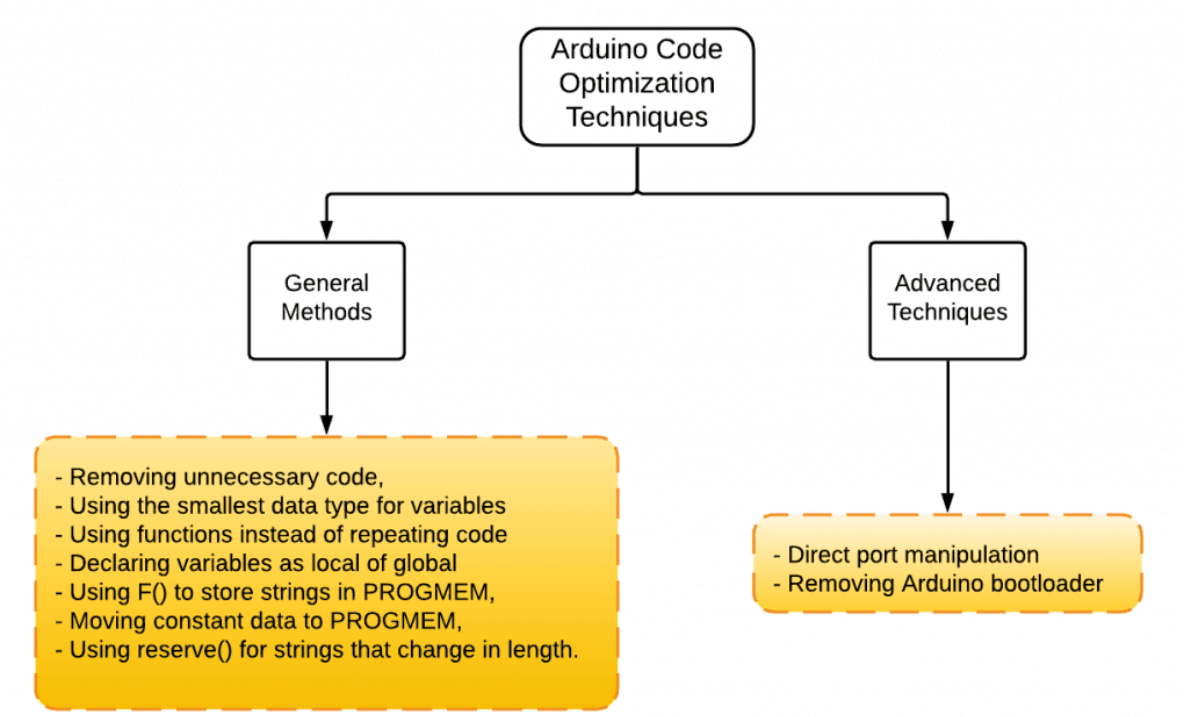


Optimisation

```
void mqttConnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting MQTT client to "); Serial.println(server);
    while (!client.connect(clientId, authMethod, token)) {
    Serial.print(".");
      delay(500);
    }
    initManagedDevice();
    Serial.println();
```

```
  }
}

void initManagedDevice() {
  if (client.subscribe(topic)) {
    // Serial.println(client.subscribe(topic));
    Serial.println("IBM subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}
void publishData()
{
  digitalWrite(trigpin,LOW);
  digitalWrite(trigpin,HIGH);
  delayMicroseconds(10);
  digitalWrite(trigpin,LOW);
  duration=pulseIn(echopin,HIGH);
  dist=duration*speed/2;
  if(dist<100){
    String payload = "{\"Alert Distance is\":";
    payload += dist;
    payload += "}";

    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
      if(client.publish(publishTopic, (char*) payload.c_str())) {
      Serial.println("Warning crosses 110cm -- it automaticaly of the loop");
      digitalWrite(led,HIGH);
    }

  }
    if(dist>101 && dist<111){
    String payload = "{\"Normal Distance\":";
    payload += dist;
    payload += "}";

    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);

    }
```

}


Arduino Code Optimization Techniques
General Methods
Advanced Techniques
- Removing unnecessary code,
- Using the smallest data type for variables
- Using functions instead of repeating code
- Declaring variables as local of global
- Using F() to store strings in PROGMEM,
- Moving constant data to PROGMEM,
- Using reserve() for strings that change in length.
- Direct port manipulation
- Removing Arduino bootloader