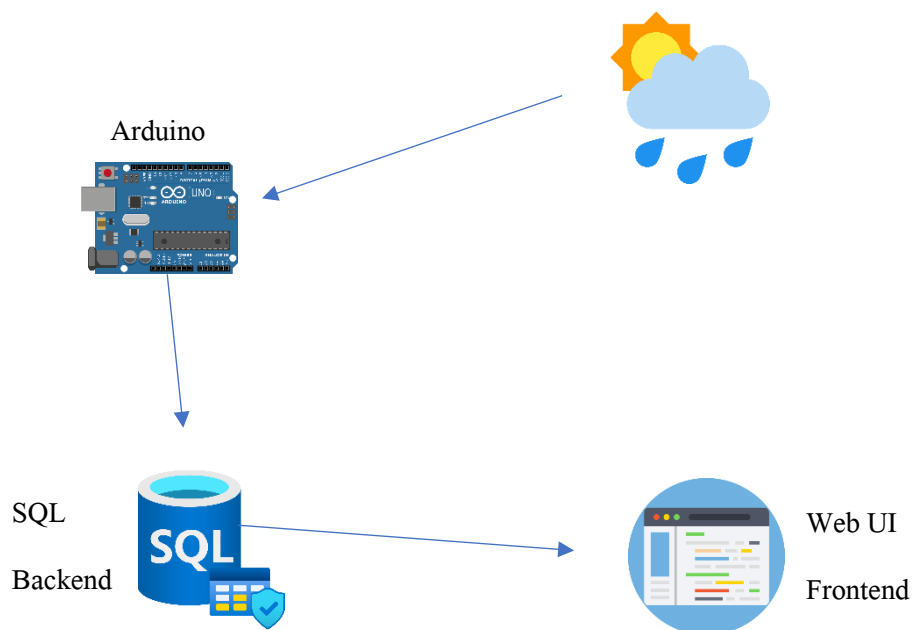# Project Plan

for

Embedded Systems "Rain 1"

Distribution:

Tampere University of Applied Sciences

Members:

Prithviraj Kalburgi (Project Manager)

Innokentti Kozlov (Arduino Coder)

Ali Abdelhamed (Technician)

Contents:

1. Overview

This project is an essential part of the Embedded Systems course which is required to fulfil in order to successfully pass the course. The motivation behind this project is to implement our knowledge that we have gained throughout this course to build a functional circuit board which is connected to a rain meter that measures the amount of rainfall. The circuit board is programmed through the Arduino interface which consists of C++ language which helps us derive the results. The duration of this course is about 4 months while we get to work on the project for nearly 2 months. This course is conducted by Tampere University of Applied Sciences and therefore all required materials and assistance for this project is provided by them. This project is tied up to our Web Development course as the results fetched from this project will be used as a backend for the Web UI that will be created for the Web Development project.

2. Goals and Scope

2.1 Project Goals

The goal of this project is to implement our software and hardware skills in a real embedded environment. The skills will be used to create a functional weather station system. Our group was assigned a specific weather situation which is "Rainfall". The main goal for our group was to create a functional circuit board which would then be connected to a MQTT server that would initially give us rainfall measurements.

Project Goal	Priority (1 – 3)	Comment/Description
Functional Goals	3	
Wiring		Making sure the wiring is functioning properly without problems
Hooking up LCD to circuit board		Making sure LCD is functioning properly by printing the right code
Adding a LED, resistor and button to the board		Key Components to the board which should function properly
Functional code which measures the voltage frequency		Making sure code is programmed properly to display the accurate voltage frequency
Technological Goals	3	
Working Code		Code which works efficiently without any setbacks
Hardware Components		Components which work properly without any problems
Quality Goals	2	
Clean Board		Wiring looks clean and good
Clean Code		Code looks clean and efficient
Communication/Team Goals	2	
Role Responsibility		Everyone is responsible for their roles and works efficiently as a team
Means of communication		Team members are communicating well with each other and updating about their progress
Constraints		
Wiring		If wiring is not done properly then circuit board can't function
Applications used		Arduino, Raspberry PI, MQTT, PostgreSQL

2.2 Project Scope

2.2.1.   Included

As a team we will be working towards a delivering a functional circuit board which will help us collect information regarding rainfall. All our technical work will be completed on our laptops where we will be programming the code using Arduino interface, and we will be using different reading machines such as a voltmeter and ammeter. Our hardware components contain, a wooden board (base), breadboard, Arduino mega, Arduino mini, LCD display, electric wires, LED light, resistor and a button. The final product will be connected to the MQTT server where the code will be uploaded in order to fetch the results from the existing database. The data we receive will then be the measurements for our specific assigned weather situation which is "Rainfall".

The results from this project will be used as data for our Web Development course where we will make a web UI browser to visualise and analyse our measurements.

3. Organization

We formed a team before the project started as we knew each other's capabilities and skills from the past. This made it easier for us to work efficiently as a group and sort out each other's roles and responsibilities. As a team we made sure each group member was comfortable with their roles and knew how to handle their own specific tasks. We never faced any issues of any sort in our group as everything went smoothly and our team communication was very vital which helped us to succeed. Our formal roles were the following:

Project Manager – Prithviraj Kalburgi
I was the project manager for our team and my main role was to make sure the tasks were being completed within our planned schedule for this course. I took care of the major documentation part which consisted of recording our observations and analysations from the weekly exercises.

Technical Expert – Ali Abdelhamed
Ali was the technician for our team. He took the lead to assemble the board and made sure that the team wouldn't encounter any technical problems. His technical skills and knowledge really helped the team to progress efficiently with the weekly tasks, so we were able to complete them in time.

Programmer – Innokentti Kozlov
Innokentii was our team's lead programmer. He mainly worked on programming the code for all the weekly tasks and due to his previous experience with Arduino, the team was able to progress efficiently with all the tasks and didn't require much assistance from the teachers.

Even though each of us had different roles, we assisted each other if any of us encountered problems while working on individual tasks.
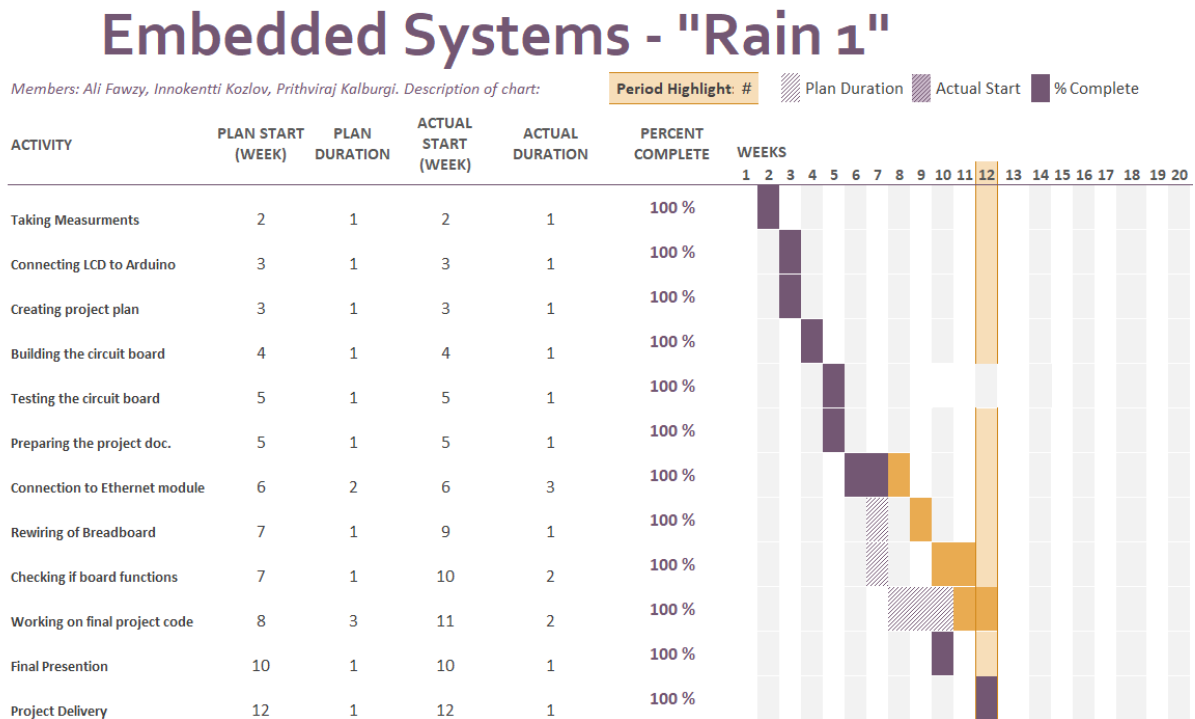
3.1 Resource Owners

All our components and materials were provided by Tampere University of Applied Sciences. Our teachers made sure that we would have full access to these materials so we could successfully complete the project by using any material of choice required. We used external resources to help us resolve our problems with the code such as Arduino forum and Stack Overflow.

4. Scheduling

4.1 Schedule and Milestone

As the project manager, I was in charge of setting our periodic goals and making sure we would achieve them within our planned schedule. In order to make our schedule, I used Microsoft Excel to create a gantt chart where I separated the tasks to each separate week. This chart allowed the team to stay on schedule and make sure we wouldn't fall behind.

Gantt Chart:

Embedded Systems - "Rain 1"

Members: Ali Fawzy, Innokentti Kozlov, Prithviraj Kalburgi. Description of chart:

Period Highlight: #   Plan Duration   Actual Start   % Complete

ACTIVITY	PLAN START (WEEK)	PLAN DURATION	ACTUAL START (WEEK)	ACTUAL DURATION	PERCENT COMPLETE
Taking Measurments	2	1	2	1	100 %
Connecting LCD to Arduino	3	1	3	1	100 %
Creating project plan	3	1	3	1	100 %
Building the circuit board	4	1	4	1	100 %
Testing the circuit board	5	1	5	1	100 %
Preparing the project doc.	5	1	5	1	100 %
Connection to Ethernet module	6	2	6	3	100 %
Rewiring of Breadboard	7	1	9	1	100 %
Checking if board functions	7	1	10	2	100 %
Working on final project code	8	3	11	2	100 %
Final Presention	10	1	10	1	100 %
Project Delivery	12	1	12	1	100 %

5. Risk Management

During the entire course we were working around open electrical sources and electric materials which could possibly pose a threat to someone's health, therefore the teacher made it very clear in the beginning of what one should do and shouldn't do in the lab.

A few things that we should be aware of is:

- Make sure to work with the equipment on a clear surface distant away from water or any substances that could pose a threat.

- Be careful with the wiring so it doesn't get damaged and make sure it's in good condition. Follow the instructions properly while connecting the wires and if in any doubt, ask for help from the teachers.

- Make sure voltage supplied is within the specified limit and going over the limit can have severe consequences.

- While installing the components on the board with nails and screwdrivers, keep an eye on your hands

- Connect the wires to the right measuring meters as per instructions to get accurate data.

- Record results consistently to make sure the board is working properly without any delays or problems

6. Communication

As the project manager, I made sure the team coordinated well and each team member was fulfilling their roles responsibly. We made sure to update each other every class on our individual progress and helped each other if there were any problems. Thanks to great support from my teammates everything went extremely smooth. We mainly used WhatsApp, Outlook and Teams to communicate and send project files/updates and the code. We made a group Gitlab to upload our documents so that everyone in the team can have access to them at any time.

7. Delivery Plan

We finished all our weekly tasks within our planned schedule which granted us enough time to work on our final project. We had our final presentation on the 8th of April and by then we were 90% completed with our project with few tweaks needed to be done to the circuit board and code. Once that was finished, we worked on the project document as a team and planned to deliver the entire project completed successfully.
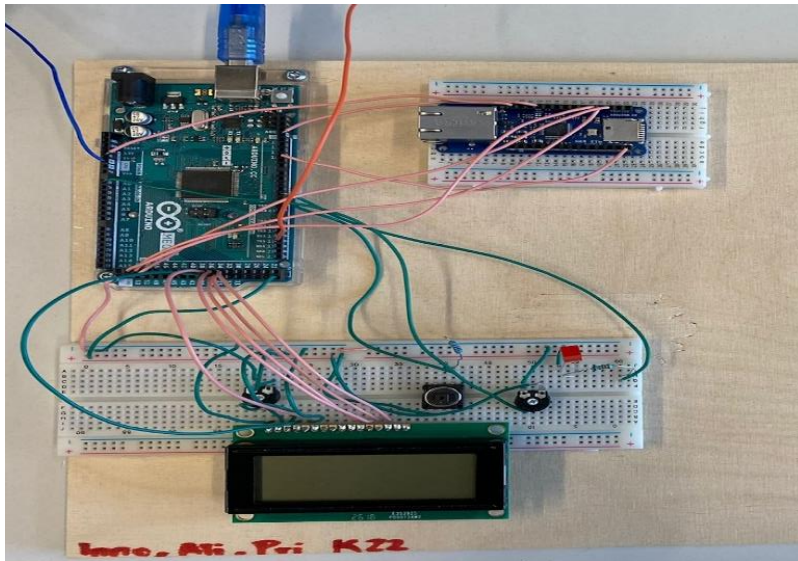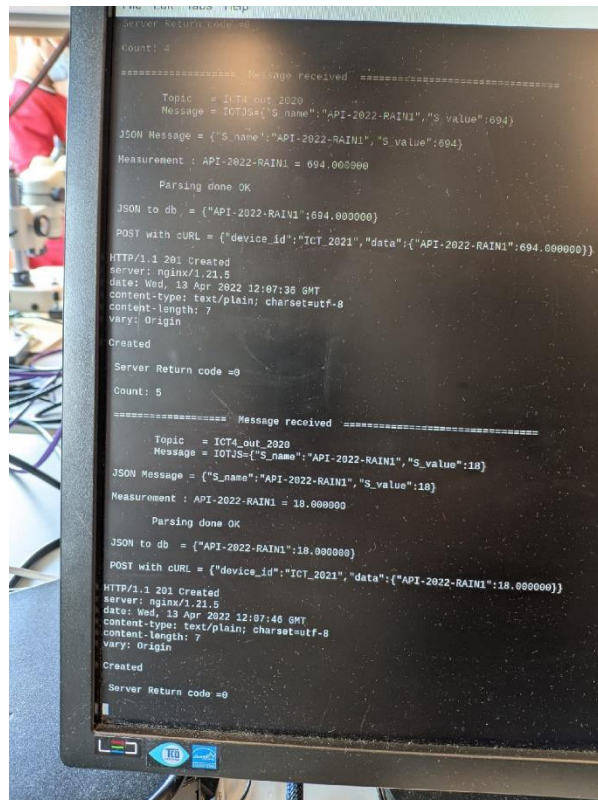
Image of our Circuit Board:

Image of data collected MQTT server:

Final project code:

```cpp
#include <LiquidCrystal.h>                          // include LCD library
                            // include Ethernet libarty W5500 library
#include <Ethernet.h>                               // incluide Ethernet library W5100

#include <PubSubClient.h>                           // include MQTT library
#include <TimerOne.h>                               // include timer library

#define rain 18

int pulses_counter=0;
int maxValue;
boolean is_rain = false;

LiquidCrystal lcd(37, 36, 35, 34, 33, 32);          // LCD pin wiring settings for MEGA

EthernetClient ethClient;

static uint8_t mymac[6] = { 0x44,0x76,0x58,0x10,0x00,0x73 };

unsigned int Port = 1883;                           //  MQTT port number

byte server[] = { 10,6,0,21 };              // TAMK IP

char* deviceId     = "API-2022-RAIN1";                  // * set your device id (will be the MQTT client username)
char* clientId     = "223API";                      // * set a random string (max 23 chars, will be the MQTT client id)
char* deviceSecret = "tamk";

void callback(char* topic, byte* payload, unsigned int length); // subscription callback for received MQTTT messages

PubSubClient client(server, Port, callback, ethClient);    // mqtt client

#define inTopic    "ICT4_in_2020"                   // * MQTT channel where data are received
#define outTopic   "ICT4_out_2020"

void setup() {
  pinMode(rain, INPUT);
  attachInterrupt(digitalPinToInterrupt(rain), isr_rain, CHANGE);
  Serial.begin(9600);
  lcd.begin(20,4);
  lcd.setCursor(0,0);
  lcd.print("30.3.2020 Alyk jatk ");
  Serial.println("Start 11.3.2022");
  delay(500);

  Timer1.initialize(10000000);
  Timer1.attachInterrupt(Timer_isr);

  fetch_IP();
  Connect_MQTT_server();
}

void loop(){
  while(true)
  {
    lcd.setCursor(0,3);
    lcd.print("Send message:  ");
    lcd.print(pulses_counter);
    send_MQTT_message(pulses_counter);
    delay(10000);
  }
}

void fetch_IP(void)
{
  byte rev=1;
  lcd.setCursor(0,0);
  lcd.print("     Waiting IP     ");
  rev=Ethernet.begin( mymac);
  Serial.print( F("\nW5100 Revision ") );
  if ( rev == 0){
    Serial.println( F( "Failed to access Ethernet controller" ) );
    lcd.setCursor(0,0); lcd.print(" Ethernet failed   ");
  }

  Serial.println( F( "Setting up DHCP" ));
  Serial.print("Connected with IP: ");
  Serial.println(Ethernet.localIP());
  lcd.setCursor(0,0);
  lcd.print("                    ");

  lcd.setCursor(0,0);
  lcd.print("myIP=");
  lcd.print(Ethernet.localIP());
  delay(1500);

}


void send_MQTT_message(int num){
  char bufa[100];
  sprintf(bufa,"IOTJS={\"S_name\":\"%s\",\"S_value\":%d}",deviceId,num);
  Serial.println( bufa );

  if (client.connected()) {
    client.publish(outTopic,bufa);
  }
  else {
    delay(500);
```

```
          lcd.setCursor(0,1);
          lcd.print("   RE Connecting    ");
          Serial.println(" RE Connecting" );
          client.connect(clientId, deviceId, deviceSecret);
          delay(1000);
   }
}

void isr_rain() {
  pulses_counter++;
  is_rain = !is_rain;
}

void Timer_isr() {
  pulses_counter = 0;
}


 void Connect_MQTT_server()                            //  MQTT server connection
 {

          Serial.println(" Connecting to MQTT" );

          Serial.print(server[0]); Serial.print(".");     // Print MQTT server IP number to Serial monitor
          Serial.print(server[1]); Serial.print(".");
          Serial.print(server[2]); Serial.print(".");
          Serial.println(server[3]);


          lcd.setCursor(0,1);
          //        012345678901234567890
          lcd.print("                    ");

          lcd.setCursor(0,1);
          lcd.print("MQTT=");
          lcd.print(server[0]);lcd.print(".");              // Print MQTT server IP number to LCD
          lcd.print(server[1]);lcd.print(".");
          lcd.print(server[2]);lcd.print(".");
          lcd.print(server[3]);

          delay(500);

   if (!client.connected())                           // check if allready connected
     {
                                                        // connection to MQTT server
      if (client.connect(clientId, deviceId, deviceSecret))
        {
          lcd.setCursor(0,1);
          lcd.print("Conn");                            // Connection is OK

           Serial.println(" Connected OK " );

           client.subscribe(inTopic);                   // subscript to in topic
        }
      else
        {
          lcd.setCursor(0,1);
          //        01234567890123456789
          lcd.print("  MQTT Error       ");                             // error in connection

          Serial.println(" MQTT Connection ERROR " );

        }
     }

 }
                                                // Receive incoming MQTT message

 void callback(char* topic, byte* payload, unsigned int length)
 {
                                               // copu the payload content into a char*
   char* receiv_string;
   receiv_string = (char*) malloc(length + 1);
   memcpy(receiv_string, payload, length);        // copy received message to receiv_string
   receiv_string[length] = '\0';

       lcd.setCursor(0,0);
       //        01234567890123456789
       lcd.print("Mess=            ");

       lcd.setCursor(5,0);
       lcd.print(receiv_string);                 // print reveived message to LCD

       Serial.println( receiv_string );

   free(receiv_string);
 }
```

Conclusion:

As a group we would like to thank our teachers, Esa, Sonja and the assistant student teacher for their support throughout the course and project. They gave us the opportunity to implement our coding skills in a real time embedded environment. The project was very fun and interesting. The whole experience has been top-notch, informative and educational. It was a very fun to work with fellow classmates and collaborate as a team to create an exceptional product.

Team Cheesecake

Prithviraj Kalburgi- Project Manager (prithviraj.kalburgi@tuni.fi)

Ali Abdelhamed- Technician (ali.abdelhamed@tuni.fi)

Innokentti Kozlov - Coder (innokentti.kozlov@tuni.fi)