**A.1 SOURCE CODE**

**A.1.1 Ino Code**

#include <Servo.h>

#include <ESP8266WiFi.h>

#include <FirebaseESP8266.h>

#define Firebase\_Host "<https://iot-feeder-1d737-default-rtdb.asia-southeast1.firebasedatabase.app/>"

#define Firebase\_Auth "E3ZIcq4fdZaPuihuSBX0yBLxNlpnLW5BNGwfZBYf"

const char\* ssid = "kirthik";

const char\* password = "987654321";

const int trigPin = D6;

const int echoPin = D7;

long duration;

int distance=0;

const int sensorPin = A0;

int ledPin = 4;

// Variables for storing the sensor data

int waterLevel = 0;

FirebaseData firebaseData;

FirebaseJson json;

Servo servo;

WiFiServer server(80);

void setup()

{

Serial.begin(9600);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

pinMode(sensorPin, INPUT);

pinMode(ledPin, OUTPUT);

digitalWrite(ledPin, LOW);

servo.attach(5); //D1

servo.write(0);

delay(2000);

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED)

{

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

server.begin();

Serial.println("Server started");

// Print the IP address

Serial.println(WiFi.localIP());

Firebase.begin(Firebase\_Host, Firebase\_Auth);

Firebase.reconnectWiFi(true);

}

void loop()

{

delay(1000);

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// measure the duration of the echo pulse in microseconds

duration = pulseIn(echoPin, HIGH);

// calculate the distance in centimeters

distance = duration\*0.034/2;

if(1){

Serial.print("Distance: ");

Serial.print(distance);

Serial.println(" cm ");

Serial.println("Food is very LOW");

}

waterLevel = analogRead(sensorPin)-5;

if(waterLevel<100){

Serial.print("Water Level: ");

Serial.println(waterLevel);

}

delay(1000);

Firebase.setString(firebaseData, "ultra value/us", distance);

Firebase.setString(firebaseData, "ultra value/lvl", waterLevel);

if(Firebase.getString(firebaseData, "/ultra value/user\_name"))

{

String ledstatus = firebaseData.stringData();

if(ledstatus.toInt() == 0){

servo.write(180);

delay(2000);

servo.write(0);

delay(1000);

}

if(Firebase.getString(firebaseData, "/ultra value/Water\_pump"))

{

String waterstatus = firebaseData.stringData();

if(waterstatus.toInt() == 0){

digitalWrite(ledPin, LOW);

}

else{

digitalWrite(ledPin, HIGH);

}

}

}

}

**A.1.2 Login Page**

<html>

<head>

<style>

/\*@import url(https://fonts.googleapis.com/css?family=Roboto:400,300,500);\*/

\*:focus {

outline: none;

}

body {

margin: 0;

padding: 0;

background: #DDD;

font-size: 16px;

color: #222;

font-family: 'Roboto', sans-serif;

font-weight: 300;

}

#login-box {

position: relative;

margin: 5% auto;

width: 600px;

height: 400px;

background: #FFF;

border-radius: 2px;

box-shadow: 0 2px 4px rgba(0, 0, 0, 0.4);

}

.left {

position: absolute;

top: 0;

left: 0;

box-sizing: border-box;

padding: 40px;

width: 300px;

height: 400px;

}

h1 {

margin: 0 0 20px 0;

font-weight: 300;

font-size: 28px;

}

input[type="text"],

input[type="password"] {

display: block;

box-sizing: border-box;

margin-bottom: 20px;

padding: 4px;

width: 220px;

height: 32px;

border: none;

border-bottom: 1px solid #AAA;

font-family: 'Roboto', sans-serif;

font-weight: 400;

font-size: 15px;

transition: 0.2s ease;

}

input[type="text"]:focus,

input[type="password"]:focus {

border-bottom: 2px solid #16a085;

color: #16a085;

transition: 0.2s ease;

}

input[type="submit"] {

margin-top: 28px;

width: 120px;

height: 32px;

background: #16a085;

border: none;

border-radius: 2px;

color: #FFF;

font-family: 'Roboto', sans-serif;

font-weight: 500;

text-transform: uppercase;

transition: 0.1s ease;

cursor: pointer;

}

input[type="submit"]:hover,

input[type="submit"]:focus {

opacity: 0.8;

box-shadow: 0 2px 4px rgba(0, 0, 0, 0.4);

transition: 0.1s ease;

}

input[type="submit"]:active {

opacity: 1;

box-shadow: 0 1px 2px rgba(0, 0, 0, 0.4);

transition: 0.1s ease;

}

.or {

position: absolute;

top: 180px;

left: 280px;

width: 40px;

height: 40px;

background: #DDD;

border-radius: 50%;

box-shadow: 0 2px 4px rgba(0, 0, 0, 0.4);

line-height: 40px;

text-align: center;

}

.right {

position: absolute;

top: 0;

right: 0;

box-sizing: border-box;

padding: 40px;

width: 300px;

height: 400px;

background: url('https://goo.gl/YbktSj');

background-size: cover;

background-position: center;

border-radius: 0 2px 2px 0;

}

.right .loginwith {

display: block;

margin-bottom: 40px;

font-size: 28px;

color: #FFF;

text-align: center;

}

button.social-signin {

margin-bottom: 20px;

width: 220px;

height: 36px;

border: none;

border-radius: 2px;

color: #FFF;

font-family: 'Roboto', sans-serif;

font-weight: 500;

transition: 0.2s ease;

cursor: pointer;

}

button.social-signin:hover,

button.social-signin:focus {

box-shadow: 0 2px 4px rgba(0, 0, 0, 0.4);

transition: 0.2s ease;

}

button.social-signin:active {

box-shadow: 0 1px 2px rgba(0, 0, 0, 0.4);

transition: 0.2s ease;

}

button.social-signin.facebook {

background: #32508E;

}

button.social-signin.twitter {

background: #55ACEE;

}

button.social-signin.google {

background: #DD4B39;

}

</style>

<title>Firebase Authentication</title>

<link rel="stylesheet" href="cssFile.css">

</head>

<body>

<div id="login-box">

<div class="left">

<h1 id='status'>login</h1>

<input type="text" id="email" name="email" placeholder="E-mail" />

<input type="password" id="password" name="password" placeholder="Password" />

<input type="submit" id='submitData' name="submitData" value="Login" />

</div>

<div class="right">

<span class="loginwith">Sign in with<br />social network</span>

<button class="social-signin facebook">Log in with facebook</button>

<button class="social-signin twitter">Log in with Twitter</button>

<button class="social-signin google">Log in with Google+</button>

</div>

<div class="or">OR</div>

</div>

</body>

<script type="module">

// Import the functions you need from the SDKs you need

import {initializeApp} from "<https://www.gstatic.com/firebasejs/9.6.10/firebase-app.js>";

import {

getAuth,

createUserWithEmailAndPassword,

signInWithEmailAndPassword,

signOut

} from "<https://www.gstatic.com/firebasejs/9.6.10/firebase-auth.js>";

import {getDatabase, set, ref, update} from "<https://www.gstatic.com/firebasejs/9.6.10/firebase-database.js>";

// Your web app's Firebase configuration

const firebaseConfig = {

apiKey: "AIzaSyAleW3S2ytbwk4aqKE95P1-JFMUdcZgswc",

authDomain: "login-cd940.firebaseapp.com",

databaseURL: "[https://login-cd940-default-rtdb.asia-southeast1.firebasedatabase.app](https://login-cd940-default-rtdb.asia-southeast1.firebasedatabase.app/)",

projectId: "login-cd940",

storageBucket: "login-cd940.appspot.com",

messagingSenderId: "736086173042",

appId: "1:736086173042:web:72fce179065f3d0bb729eb",

measurementId: "G-EER8D7YTH0"

};

// Initialize Firebase

const app = initializeApp(firebaseConfig);

const auth = getAuth();

const database = getDatabase(app);

submitData.addEventListener('click', (e) => {

var email = document.getElementById('email').value;

var password = document.getElementById('password').value;

// // log in user

signInWithEmailAndPassword(auth, email, password)

.then((userCredential) => {

// Signed in

const user = userCredential.user;

var lgDate = new Date();

update(ref(database, 'users/' + user.uid), {

last\_login: lgDate,

})

.then(() => {

// Data saved successfully!

window.location.href = "index.html";

})

.catch((error) => {

// The write failed...

alert(error);

});

})

.catch((error) => {

const errorCode = error.code;

const errorMessage = error.message;

alert(errorMessage);

});

});

</script>

</html>

**A.1.3 Javascript**

import { initializeApp } from "<https://www.gstatic.com/firebasejs/9.1.0/firebase-app.js>";

import { getDatabase, ref, set, child, update, remove, onValue } from "<https://www.gstatic.com/firebasejs/9.1.0/firebase-database.js>";

/\*\*\*\*\* Firebase config \*\*\*\*\*/

const firebaseConfig = {

    apiKey: "AIzaSyBuAT3UJ9zZEPVsR2Oml0V1oFMHeT8CgXw",

    authDomain: "iot-feeder-1d737.firebaseapp.com",

    databaseURL: "[https://iot-feeder-1d737-default-rtdb.asia-southeast1.firebasedatabase.app](https://iot-feeder-1d737-default-rtdb.asia-southeast1.firebasedatabase.app/)",

    projectId: "iot-feeder-1d737",

    storageBucket: "iot-feeder-1d737.appspot.com",

    messagingSenderId: "801152065411",

    appId: "1:801152065411:web:85d50665c53af8c265e0b6",

    measurementId: "G-4B4BSRMM91"

  };

/\*\*\*\*\* Initialize Firebase \*\*\*\*\*/

const app = initializeApp(firebaseConfig);

//\*\*\*\* write data to firebase

function write\_db() {

    console.log("DEBUG: Write function");

    var db = getDatabase();

    var create\_db\_table = ref(db, 'ultra value/');

    var user\_name = document.getElementById("user\_name").value;

    var waterpump = document.getElementById("water").value;

    if( user\_name == '' || waterpump=='' ){

        alert("Make sure, must be non-empty data is required!!!");

        console.log("Make sure, must be non-empty data is required!!!");

        throw "Make sure, must be non-empty data is required!!!";

    }

    set(ref(db, 'ultra value/'), {

      user\_name: user\_name,

      Water\_pump: waterpump

    }).then((res) => {

        console.log();

    })

    .catch((err) => {

        alert(err.message);

        console.log(err.code);

        console.log(err.message);

    })

}

/\*\*\*\*\* read data from firebase \*\*\*\*\*/

function read\_db() {

    var db = getDatabase();

    var connect\_db = ref(db, 'ultra value/');

    var retrieve\_data='';

    console.log("DEBUG: Read function");

    onValue(connect\_db, (snapshot) => {

        retrieve\_data = snapshot.val();

        //console.log("user\_name: " + retrieve\_data.user\_name);

        //console.log("user\_name\_password: " + retrieve\_data.user\_name\_password);

        call\_loop\_print(retrieve\_data);

        document.getElementById("display\_read\_data").innerHTML =  "<pre>" + "Distance: " + retrieve\_data.us + "\n"+"level:"+retrieve\_data.lvl+ "</pre>";

        })

    function call\_loop\_print(retrieve\_data){

        for (var r=0;r<Object.entries(retrieve\_data).length;r++){

            var key = Object.keys(retrieve\_data)[r];

            var value = retrieve\_data[key];

            console.log("Key\_" + r + ': ' + key + " Value\_:" + r + ': ' + value );

           }

 }

}

/\*\*\*\*\* call write data function \*\*\*\*\*/

var write\_data\_to\_firebase = document.getElementById("write\_data\_to\_firebase");

write\_data\_to\_firebase.addEventListener('click', write\_db);

/\*\*\*\*\* call read data function \*\*\*\*\*/

var read\_data\_from\_firebase = document.getElementById("read\_data\_from\_firebase");

read\_data\_from\_firebase.addEventListener('click', read\_db);

**A.1.4 Signup Page**

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

body {

font-family: Arial, Helvetica, sans-serif;

background-color: black;

}

\* {

box-sizing: border-box;

}

/\* Add padding to containers \*/

.container {

padding: 16px;

background-color: white;

}

/\* Full-width input fields \*/

input[type=text],

input[type=password] {

width: 100%;

padding: 15px;

margin: 5px 0 22px 0;

display: list-block;

border: none;

background: #f1f1f1;

}

input[type=text]:focus,

input[type=password]:focus {

background-color: #ddd;

outline: none;

}

/\* Overwrite default styles of hr \*/

hr {

border: 1px solid #f1f1f1;

margin-bottom: 25px;

}

/\* Set a style for the submit button \*/

.registerbtn {

background-color: #04AA6D;

color: white;

padding: 16px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 50%;

opacity: 0.9;

}

.registerbtn:hover {

opacity: 1;

}

/\* Add a blue text color to links \*/

a {

color: dodgerblue;

}

/\* Set a grey background color and center the text of the "sign in" section \*/

.signin {

background-color: #f1f1f1;

text-align: center;

}

</style>

</head>

<body>

<form>

<div class="container">

<h1>Register</h1>

<p>Please fill in this form to create an account.</p>

<hr>

<label for="email"><b>Email</b></label>

<input type="text" placeholder="Enter Email" name="email" id="email" required>

<label><b>Password</b></label>

<input type="password" placeholder="Password" name="psw" id="psw" required>

<hr>

<p>By creating an account you agree to our <a href="#">Terms & Privacy</a>.</p>

<button type="button" id="submitData" name="submitData" class="registerbtn">Register</button>

</div>

<div class="container signin">

<p>Already have an account? <a href="log.html">Sign in</a>.</p>

</div>

</form>

</body>

</html>

<script type="module">

// Import the functions you need from the SDKs you need

import {initializeApp} from "<https://www.gstatic.com/firebasejs/9.6.10/firebase-app.js>";

import {

getAuth,

createUserWithEmailAndPassword,

signInWithEmailAndPassword,

signOut

} from "<https://www.gstatic.com/firebasejs/9.6.10/firebase-auth.js>";

import {getDatabase, set, ref, update} from "<https://www.gstatic.com/firebasejs/9.6.10/firebase-database.js>";

// TODO: Add SDKs for Firebase products that you want to use

// <https://firebase.google.com/docs/web/setup#available-libraries>

// Your web app's Firebase configuration

const firebaseConfig = {

apiKey: "AIzaSyAleW3S2ytbwk4aqKE95P1-JFMUdcZgswc",

authDomain: "login-cd940.firebaseapp.com",

databaseURL: "[https://login-cd940-default-rtdb.asia-southeast1.firebasedatabase.app](https://login-cd940-default-rtdb.asia-southeast1.firebasedatabase.app/)",

projectId: "login-cd940",

storageBucket: "login-cd940.appspot.com",

messagingSenderId: "736086173042",

appId: "1:736086173042:web:72fce179065f3d0bb729eb",

measurementId: "G-EER8D7YTH0"

};

// Initialize Firebase

const app = initializeApp(firebaseConfig);

const auth = getAuth();

const database = getDatabase(app);

submitData.addEventListener('click', (e) => {

var email = document.getElementById('email').value;

var password = document.getElementById('psw').value;

//sign up user

createUserWithEmailAndPassword(auth, email, password)

.then((userCredential) => {

// Signed in

const user = userCredential.user;

// ... user.uid

set(ref(database, 'users/' + user.uid), {

email: email,

password: password

})

.then(() => {

// Data saved successfully!

alert('user created successfully');

})

.catch((error) => {

// The write failed...

alert("here error");

});

})

.catch((error) => {

const errorCode = error.code;

const errorMessage = error.message;

// ..

alert(errorMessage);

});

});

</script>