STEM ARDUINO PBL RESOURCE PACKAGE

# 10 Project-Based Learning Units

Grades 7–12 • Foundation to Portfolio Level • SDG-Aligned

*Based on BIE PBL Framework (Thomas, 2000) & Buck Institute Essential Elements*

## PBL Framework Foundation

All ten projects in this package are built on the Buck Institute for Education (BIE) model of Project Based Learning. Each project incorporates the Eight Essential Elements of PBL and Thomas's (2000) Five Defining Features.

| PBL ELEMENT | HOW IT APPEARS IN THESE PROJECTS |
|---|---|
| **CENTRALITY** | The Arduino challenge IS the curriculum unit — not an add-on activity. |
| **DRIVING QUESTION** | Each project opens with an open-ended, authentic question students must answer. |
| **CONSTRUCTIVE INVESTIGATION** | Students build knowledge through inquiry, experimentation, and iteration. |
| **STUDENT AUTONOMY** | Learners make design decisions; teacher facilitates rather than directs. |
| **REALISM / AUTHENTICITY** | All projects connect to real-world problems with genuine stakeholders. |
| **21ST CENTURY SKILLS** | Critical thinking, collaboration, communication, and creativity are explicitly assessed. |
| **REVISION & REFLECTION** | Upgrade challenges and peer feedback drive iterative improvement. |
| **PUBLIC AUDIENCE** | Every project culminates in a presentation beyond the classroom. |

## Project Progression Overview

| # | Project | Tier | Grades | Duration | Key Skills |
|---|---|---|---|---|---|
| 1 | Smart LED Control System | **1 – Foundation** | 7–9 | 2 wks | Input→Processing→Output, IR control |
| 2 | Digital Temperature Monitor | **1 – Foundation** | 7–9 | 2 wks | Sensors, threshold logic, LCD |
| 3 | Motion Detection Alarm | **1 – Foundation** | 7–9 | 2 wks | PIR sensor, Boolean logic |
| 4 | Smart Home Mini System | **2 – Integration** | 8–10 | 3 wks | Relay, multi-input, state machines |
| 5 | Smart Irrigation System | **2 – Integration** | 8–10 | 3 wks | Agriculture tech, automation |
| 6 | Obstacle Avoidance System | **2 – Integration** | 7–10 | 2–3 wks | Ultrasonic sensing, zones |
| 7 | Bluetooth-Controlled Device | **2 – Integration** | 8–10 | 2–3 wks | UART, mobile integration |
| 8 | Smart Classroom Environment | **3 – Portfolio** | 9–11 | 4 wks | Multi-sensor, data analysis |

| | | | | | |
|---|---|---|---|---|---|
| 9 | Multi-Sensor Data Dashboard | **3 – Portfolio** | 10–12 | 4–5 wks | Systems thinking, data fusion |
| 10 | RFID Smart Access System | **3 – Portfolio** | 9–12 | 3–4 wks | SPI, security system design |

# TIER 1: FOUNDATION PROJECTS (Projects 1–3)

*Foundation projects teach the Input → Processing → Output model. Students learn individual sensors and components before combining them. Focus: circuit building, basic coding, and clear real-world purpose.*

TIER 1 — FOUNDATION

## 1. Smart LED Control System

DRIVING QUESTION

***"How can we use programmable lighting to create an accessible home environment for elderly and disabled users?"***

**Real-World Angle:** Remote-controlled lighting for accessibility (elderly/disabled users)

**COMPONENTS**

- Arduino UNO / Nano
- IR receiver + IR remote
- LEDs (red, green, blue)
- 220Ω resistors
- Breadboard + jumper wires

**SKILLS & CONCEPTS**

- Input → Processing → Output model
- Digital & PWM signals
- IR signal decoding
- Conditional logic (if/else)
- Brightness control with analogWrite()

### PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Show video of smart homes for elderly users; present challenge brief | Brainstorm questions: Who needs this? What makes lighting accessible? | KWL chart – what do students know about light/accessibility? |
| **Inquiry** | Mini-lesson: circuits, IR signals, PWM; guided code exploration | Research: how IR remotes work; test basic LED circuits | Circuit diagram sketch + labeled components |
| **Build & Iterate** | Facilitate prototyping; ask probing questions about failure points | Build circuit, write code, test IR input → LED response loop | Prototype checklist; peer feedback on function |
| **Upgrade Challenge** | Introduce brightness levels / colour-mode challenge | Add modes: dim / medium / bright; colour-change via remote | Code review: logic structure, comments, efficiency |
| **Public Showcase** | Organise presentation to school community / parents | Demo device; explain design choices; present accessibility impact | Rubric: functionality, presentation, real-world connection |

### Sample Arduino Code

```
// IR Receiver - LED Control
#include <IRremote.h>
const int RECV_PIN = 11;
IRrecv irrecv(RECV_PIN);
decode_results results;
void setup() {
  irrecv.enableIRIn();
  pinMode(9, OUTPUT);
}
void loop() {
  if (irrecv.decode(&results)) {
    if (results.value == 0xFFA25D) analogWrite(9, 255); // Full bright
    if (results.value == 0xFF629D) analogWrite(9, 128); // Half bright
    if (results.value == 0xFFE21D) analogWrite(9, 0);   // Off
    irrecv.resume();
  }
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 3 – Good Health & Well-Being \| SDG 10 – Reduced Inequalities | **Duration:** 2 weeks | **Level:** Foundation |

TIER 1 — FOUNDATION

# 2. Digital Temperature Monitor

DRIVING QUESTION

***"How can we design a smart classroom monitor that alerts teachers and students when environmental conditions become unsafe?"***

**Real-World Angle:** Environmental health monitoring for schools and clinics

**COMPONENTS**

- Arduino UNO
- DHT11 / DHT22 sensor
- 16×2 LCD (I2C)
- Red/Green LEDs
- Buzzer
- 10kΩ resistor

**SKILLS & CONCEPTS**

- Sensor data acquisition
- I2C communication protocol
- Threshold logic
- Real-time display
- Warning system design

## PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Share WHO guidelines on classroom air temperature; pose: 'Could our classroom be unhealthy right now?' | Measure room temperature manually; compare against guidelines; generate questions | Entry ticket: student hypothesis about ideal classroom temperature |
| **Inquiry** | Teach DHT11 wiring, LCD I2C setup, threshold concept | Research: health effects of temperature extremes on learning | Concept map: sensor → data → decision → alert |
| **Build & Iterate** | Guide debugging of LCD display issues; challenge students to refine thresholds | Build circuit; code temperature/humidity display; add LED/buzzer warning | Functionality test: does warning trigger at correct threshold? |
| **Upgrade Challenge** | Introduce data logging concept; suggest trend display | Add cool/warm/danger indicators; customise thresholds per scenario | Code commenting quality; threshold justification document |
| **Public Showcase** | Connect to school health/facilities team for authentic audience | Present monitor to school nurse or facilities team with recommendations | Written report: problem, design, data, recommendations |

## Sample Arduino Code

```
#include <DHT.h>
#include <LiquidCrystal_I2C.h>
#define DHTPIN 2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x27,16,2);
void setup() { dht.begin(); lcd.begin(); lcd.backlight(); }
void loop() {
  float t = dht.readTemperature();
```

```
  float h = dht.readHumidity();
  lcd.setCursor(0,0); lcd.print("Temp: "); lcd.print(t); lcd.print("C");
  lcd.setCursor(0,1); lcd.print("Hum:  "); lcd.print(h); lcd.print("%");
  if (t > 30) { digitalWrite(8, HIGH); tone(7, 1000, 500); }
  else digitalWrite(8, LOW);
  delay(2000);
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 3 – Good Health & Well-Being \| SDG 4 – Quality Education | **Duration:** 2 weeks | **Level:** Foundation |

TIER 1 — FOUNDATION

# 3. Motion Detection Alarm

DRIVING QUESTION

***"How can communities with limited resources use simple electronics to build effective home security systems?"***

**Real-World Angle:** Low-cost home security for underserved communities

**COMPONENTS**

- Arduino UNO
- PIR motion sensor (HC-SR501)
- Buzzer
- Red LED
- Green LED
- Jumper wires

**SKILLS & CONCEPTS**

- Digital input reading
- PIR sensor calibration
- Boolean logic
- Timing functions (millis)
- System states (armed/disarmed)

## PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Show statistics on home break-ins; commercial alarm costs vs community income data | Research affordable security solutions; identify stakeholder needs | Problem statement: who needs this, what does success look like? |
| **Inquiry** | Explain PIR sensor physics; demo sensitivity adjustment | Test PIR response to different movements; measure detection range/angle | Data table: PIR test results (distance, angle, material) |
| **Build & Iterate** | Scaffold arming/disarming logic; ask: 'How do you prevent false alarms?' | Build and code: green LED = safe, red LED + buzzer = alarm | Peer test: does system trigger reliably? false alarm rate? |
| **Upgrade Challenge** | Introduce delay timer concept for arm/disarm | Add 10-second arming delay; add tone pattern for alarm | User test with 'intruder scenario'; reliability score |
| **Public Showcase** | Invite community partner or parent to evaluate prototype | Present cost analysis: their system vs commercial options; demo functionality | Cost-benefit analysis document; presentation rubric |

## Sample Arduino Code

```
const int PIR = 3, BUZZ = 8, RED = 7, GRN = 6;
bool armed = false;
void setup() {
  pinMode(PIR, INPUT); pinMode(BUZZ, OUTPUT);
  pinMode(RED, OUTPUT); pinMode(GRN, OUTPUT);
  delay(30000); // PIR warmup
  armed = true;
  digitalWrite(GRN, HIGH);
}
void loop() {
```

```
  if (armed && digitalRead(PIR) == HIGH) {
    digitalWrite(GRN, LOW); digitalWrite(RED, HIGH);
    tone(BUZZ, 2000); delay(3000);
    noTone(BUZZ); digitalWrite(RED, LOW);
    digitalWrite(GRN, HIGH);
  }
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 11 – Sustainable Cities & Communities \| SDG 16 – Peace, Justice & Strong Institutions | **Duration:** 2 weeks | **Level:** Foundation |

# TIER 2: INTEGRATION PROJECTS (Projects 4–7)

*Integration projects combine multiple sensors and actuators into cohesive systems. Students develop systems thinking, multi-input logic, and more complex code architecture. This is where students distinguish themselves.*

TIER 2 — INTEGRATION

## 4. Smart Home Mini System

DRIVING QUESTION

***"How can we design an integrated home automation system that improves energy efficiency and convenience for a real household?"***

**Real-World Angle:** Intro to IoT and automation systems for K-12 learners

**COMPONENTS**

- Arduino UNO
- IR remote + receiver
- Relay module (5V)
- PIR motion sensor
- 16×2 LCD (I2C)
- LEDs + resistors

**SKILLS & CONCEPTS**

- Systems integration
- Relay control (HIGH/LOW logic)
- Multi-input logic
- State machines
- LCD status display

### PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Show smart home product videos; ask: 'What makes a home smart? Could we build one?' | Map current home inefficiencies; identify automation opportunities | System requirements list: must-have vs nice-to-have features |
| **Inquiry** | Teach relay module safety; IR + PIR combined input logic; LCD status | Research: how do commercial smart home systems work? What are relay risks? | System block diagram: input → controller → output pathways |
| **Build & Iterate** | Guide integration of multiple subsystems; introduce debugging strategy | Build incrementally: IR first, then PIR, then relay, then LCD | Integration checklist; peer code review |
| **Upgrade Challenge** | Challenge: add energy-saving mode (auto-off after inactivity) | Code auto-off timer; add LCD status messages for each state | Demo video: all features working simultaneously |
| **Public Showcase** | Arrange demo to school facilities team or parents as 'client' | Present: system demo, energy savings estimate, installation guide | Technical report + live demo rubric |

### Sample Arduino Code

```
// Smart Home: PIR auto-lights + IR remote override
// Relay on pin 4; PIR on pin 3; LCD on I2C
bool autoMode = true;
void loop() {
  // IR remote: toggle auto/manual
  if (irrecv.decode(&results)) {
    if (results.value == 0xFFA25D) autoMode = !autoMode;
    irrecv.resume();
  }
  // Auto mode: lights follow motion
  if (autoMode) {
    bool motion = digitalRead(PIR);
    digitalWrite(RELAY, motion ? HIGH : LOW);
    lcd.print(motion ? "Motion: ON " : "Motion: OFF");
  }
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 7 – Affordable & Clean Energy \| SDG 9 – Industry, Innovation & Infrastructure | **Duration:** 3 weeks | **Level:** Intermediate |

TIER 2 — INTEGRATION

# 5. Smart Irrigation System

DRIVING QUESTION

***"How can Kenyan farmers and developing-world communities use low-cost technology to conserve water and improve crop yields?"***

**Real-World Angle:** Agriculture tech for food security and water conservation

COMPONENTS

- Arduino UNO
- Capacitive soil moisture sensor
- 5V relay module
- Mini water pump
- Silicone tubing
- 9V battery / USB power

SKILLS & CONCEPTS

- Analog sensor reading
- Threshold-based automation
- Relay/pump control
- Agricultural data interpretation
- Sustainability thinking

## PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Share data: 40% of crops in sub-Saharan Africa lost to drought; show traditional vs drip irrigation | Interview (or research): What do farmers need? What barriers exist to technology? | Stakeholder empathy map |
| **Inquiry** | Teach soil moisture sensor calibration; relay + pump circuit | Experiment: moisture readings for dry/moist/waterlogged soil; calibrate thresholds | Calibration data table with recommended threshold values |
| **Build & Iterate** | Guide full circuit assembly; prompt: 'How do you prevent overwatering?' | Build and test: sensor reads → relay closes → pump waters → relay opens | System test with actual soil: accuracy, response time |
| **Upgrade Challenge** | Introduce LCD display and manual override concept | Add: LCD moisture % display, manual water button, low-water warning | Field test with plant pot; water usage log over 3 days |
| **Public Showcase** | Connect to school garden or local farm as authentic context | Present to school groundskeeper or agriculture teacher; propose real installation | Cost analysis; water savings projection; stakeholder feedback |

## Sample Arduino Code

```
const int SENSOR = A0, RELAY = 4;
const int DRY = 800, WET = 400;
void setup() {
  pinMode(RELAY, OUTPUT);
  Serial.begin(9600);
}
void loop() {
  int val = analogRead(SENSOR);
```

```
  int pct = map(val, DRY, WET, 0, 100);
  Serial.print("Moisture: "); Serial.print(pct); Serial.println("%");
  if (pct < 30) {
    digitalWrite(RELAY, HIGH); // pump ON
    delay(5000);               // water for 5 seconds
    digitalWrite(RELAY, LOW);  // pump OFF
  }
  delay(10000); // check every 10 seconds
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 2 – Zero Hunger \| SDG 6 – Clean Water & Sanitation \| SDG 13 – Climate Action | **Duration:** 3 weeks | **Level:** Intermediate |

TIER 2 — INTEGRATION

# 6. Obstacle Avoidance System

DRIVING QUESTION

***"How can we use ultrasonic sensing to help robots and assistive devices navigate safely in complex environments?"***

**Real-World Angle:** Assistive navigation technology for mobility-impaired users; autonomous vehicle foundations

**COMPONENTS**

- Arduino UNO
- HC-SR04 ultrasonic sensor
- Buzzer
- Red/Yellow/Green LEDs
- Servo motor (upgrade)
- Motor driver (robot upgrade)

**SKILLS & CONCEPTS**

- Ultrasonic distance measurement
- Speed of sound calculations
- Proximity zones (danger/warning/safe)
- PWM motor control
- Servo actuation

## PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Demo commercial parking sensor; show clips of assistive navigation tech for blind users | Brainstorm: where is distance sensing useful? Who benefits most? | Use-case list ranked by social impact |
| **Inquiry** | Teach ultrasonic pulse/echo physics; distance formula derivation | Experiments: measure objects at varying distances; test detection angle/limits | Distance accuracy test: sensor vs ruler (% error) |
| **Build & Iterate** | Guide traffic-light warning system build; ask: 'What's a safe reaction distance?' | Build: green LED >50cm, yellow 20–50cm, red <20cm + buzzer | Calibration test with standardised objects |
| **Upgrade Challenge** | Introduce servo motor for steering or robot car chassis (optional) | Add servo: auto-turn when obstacle detected; build robot chassis | Obstacle course test: completion time, number of collisions |
| **Public Showcase** | Set up obstacle course challenge event | Race/challenge demo; present design evolution and physics behind it | Engineering design journal; demo performance; peer vote |

## Sample Arduino Code

```
#define TRIG 9
#define ECHO 10
long duration; int dist;
void setup() {
  pinMode(TRIG, OUTPUT); pinMode(ECHO, INPUT);
  Serial.begin(9600);
}
void loop() {
  digitalWrite(TRIG, LOW); delayMicroseconds(2);
```

```
  digitalWrite(TRIG, HIGH); delayMicroseconds(10);
  digitalWrite(TRIG, LOW);
  duration = pulseIn(ECHO, HIGH);
  dist = duration * 0.034 / 2;
  Serial.print("Distance: "); Serial.print(dist); Serial.println("cm");
  if (dist < 20) { tone(8, 2000); digitalWrite(7, HIGH); } // DANGER
  else if (dist < 50) { noTone(8); digitalWrite(6, HIGH); } // WARNING
  else { noTone(8); digitalWrite(5, HIGH); } // SAFE
  delay(100);
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 3 – Good Health & Well-Being \| SDG 9 – Industry, Innovation & Infrastructure | **Duration:** 2–3 weeks | **Level:** Intermediate |

TIER 2 — INTEGRATION

# 7. Bluetooth-Controlled Device

DRIVING QUESTION

***"How can wireless mobile technology bridge the gap between smartphones and physical devices in low-resource community settings?"***

**Real-World Angle:** Mobile + hardware integration for community tech projects

**COMPONENTS**

- Arduino UNO
- HC-05 Bluetooth module
- LEDs + resistors
- DC motor + transistor (upgrade)
- Smartphone with Serial Bluetooth Terminal app

**SKILLS & CONCEPTS**

- Serial communication (UART)
- Bluetooth pairing & AT commands
- Mobile app interfacing
- Character-based command parsing
- Wireless control systems

## PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
| --- | --- | --- | --- |
| **Entry Event** | Show mobile-controlled devices: apps, RC toys, smart plugs; ask: 'What could you control with your phone?' | Design brief: propose a device to control wirelessly; justify real-world need | Design proposal: problem, user, solution sketch |
| **Inquiry** | Teach UART serial protocol; HC-05 AT command setup; Bluetooth pairing | Research: how Bluetooth works; test HC-05 with AT commands | AT command log: baud rate, name, PIN set successfully? |
| **Build & Iterate** | Guide HC-05 wiring (voltage divider); scaffold command parsing code | Build: phone sends '1' → LED ON; phone sends '0' → LED OFF | Command response test: 10 commands sent, 10 received correctly? |
| **Upgrade Challenge** | Challenge: add multiple devices; design a custom app interface | Control 3+ LEDs independently; optionally add motor for physical actuation | User experience test: can a non-technical user operate the system? |
| **Public Showcase** | Invite younger students or community members to try the device | Demo to novice users; collect usability feedback; iterate on interface | User feedback form analysis; final demo rubric |

## Sample Arduino Code

```
#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11); // RX, TX
char cmd;
void setup() {
  BT.begin(9600);
  pinMode(13, OUTPUT); // LED
}
void loop() {
  if (BT.available()) {
    cmd = BT.read();
```

```
    if (cmd == '1') digitalWrite(13, HIGH);
    if (cmd == '0') digitalWrite(13, LOW);
    if (cmd == 'B') tone(8, 1500, 300); // buzzer
  }
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 9 – Industry, Innovation & Infrastructure \| SDG 10 – Reduced Inequalities | **Duration:** 2–3 weeks | **Level:** Intermediate |

# TIER 3: PORTFOLIO PROJECTS (Projects 8–10)

*Portfolio projects are conference-ready and LinkedIn-worthy. They tackle real institutional problems with data-driven solutions. Students demonstrate systems architecture, stakeholder communication, and professional documentation.*

TIER 3 — PORTFOLIO

## 8. Smart Classroom Environment System

DRIVING QUESTION

***"How can we use sensor networks to monitor and optimise learning environments, and what data-driven recommendations should schools act on?"***

**Real-World Angle:** Improving learning outcomes through environmental health monitoring

**COMPONENTS**

- Arduino UNO / Mega
- DHT22 temperature/humidity sensor
- MQ-135 air quality sensor
- 16×2 or 20×4 LCD (I2C)
- RGB LED or 3 LEDs
- Active buzzer

**SKILLS & CONCEPTS**

- Multi-sensor data fusion
- Air quality index calculation
- Data-driven decision making
- Alert system design
- Environmental science integration

### PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Share research: poor air quality reduces student performance by 20%; challenge: 'Is our classroom safe right now?' | Baseline measurement campaign: collect temp/humidity data over one week manually | Data collection protocol; baseline data table |
| **Inquiry** | Teach MQ-135 calibration; compare sensor readings to WHO/EPA standards | Research: what chemicals does MQ-135 detect? What are safe CO2/VOC levels? | Literature review summary: 3 sources, key thresholds identified |
| **Build & Iterate** | Guide multi-sensor integration; scaffold alert logic hierarchy | Build: DHT22 + MQ-135 + LCD + LED/buzzer; code tiered alert system | Alert accuracy test: system triggers at correct thresholds |
| **Upgrade Challenge** | Introduce data logging to SD card or Serial plotter for trend analysis | Log data every 5 minutes; analyse: when are conditions worst? Why? | Data analysis report: trends, peaks, recommendations |
| **Public Showcase** | Arrange presentation to school principal and facilities manager | Present: data, findings, recommendations for ventilation improvements; propose permanent installation | Policy brief (1-page); presentation to decision-makers; implementation proposal |

## Sample Arduino Code

```
// Multi-sensor: DHT22 + MQ135 + LCD
#include <DHT.h>
#include <LiquidCrystal_I2C.h>
DHT dht(2, DHT22);
LiquidCrystal_I2C lcd(0x27, 16, 2);
void loop() {
  float t = dht.readTemperature();
  float h = dht.readHumidity();
  int aq = analogRead(A0); // MQ135
  lcd.clear();
  lcd.setCursor(0,0); lcd.print("T:"); lcd.print(t,1);
  lcd.print(" H:"); lcd.print(h,0); lcd.print("%");
  lcd.setCursor(0,1);
  if (aq > 700) { lcd.print("AIR: POOR"); tone(8, 1500); }
  else if (aq > 400) { lcd.print("AIR: FAIR"); }
  else { lcd.print("AIR: GOOD"); noTone(8); }
  delay(5000);
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 3 – Good Health & Well-Being \| SDG 4 – Quality Education \| SDG 13 – Climate Action | **Duration:** 4 weeks | **Level:** Advanced |

TIER 3 — PORTFOLIO

# 9. Multi-Sensor Data Dashboard

DRIVING QUESTION

***"How can we design an integrated environmental monitoring system that communicates complex data clearly to non-technical decision makers?"***

**Real-World Angle:** Systems thinking and data integration for community environmental monitoring

**COMPONENTS**

- Arduino Mega (recommended)
- DHT22 sensor
- MQ-135 air quality sensor
- PIR motion sensor
- Soil moisture sensor
- 20×4 LCD or TFT display
- HC-05 Bluetooth (wireless dashboard)

**SKILLS & CONCEPTS**

- Systems architecture design
- Multi-sensor data fusion
- Data visualisation principles
- Serial/Bluetooth data streaming
- User interface design

## PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Show professional environmental dashboards (NASA, WHO); ask: 'What makes data readable? Who is the audience?' | Analyse 3 existing dashboards: what works, what doesn't? Apply to their own design | Dashboard critique document with design principles list |
| **Inquiry** | Review all sensors from previous projects; teach data normalisation | Map all sensor data types; research: what does each number mean in real life? | Data dictionary: sensor, unit, range, healthy threshold, danger threshold |
| **Build & Iterate** | Guide system architecture; scaffold Bluetooth streaming to phone | Integrate all sensors; design LCD layout; stream data via Bluetooth to phone app | System architecture diagram; data flow validation test |
| **Upgrade Challenge** | Challenge: create a 'health score' algorithm combining all sensors | Design weighted algorithm; display overall 'Environment Score' 0–100 | Algorithm justification document; usability test with 5 non-technical users |
| **Public Showcase** | Organise a STEM fair or school showcase event | Present as a 'product launch': live demo, user manual, impact statement | STEM fair judging rubric; peer voting; written reflection |

## Sample Arduino Code

```
// Dashboard: stream all sensor data via Bluetooth
#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11);
void loop() {
  float temp = dht.readTemperature();
  float hum  = dht.readHumidity();
  int   air  = analogRead(A0); // MQ135
  int   soil = analogRead(A1); // Moisture
```

```
  bool  pir  = digitalRead(3); // Motion
  // Stream JSON to phone
  BT.print("{T:");
  BT.print(temp,1); BT.print(",H:"); BT.print(hum,0);
  BT.print(",A:"); BT.print(air);
  BT.print(",S:"); BT.print(soil);
  BT.print(",M:"); BT.print(pir); BT.println("}");
  delay(3000);
}
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 3, 4, 6, 13, 15 – Multiple Goals via Data-Driven Action | **Duration:** 4–5 weeks | **Level:** Advanced |

TIER 3 — PORTFOLIO

# 10. RFID Smart Access System

DRIVING QUESTION

***"How can low-cost RFID technology be deployed to create secure, equitable access control systems for schools and community spaces?"***

**Real-World Angle:** Low-cost smart security and access control for schools/institutions

COMPONENTS

- Arduino UNO
- MFRC522 RFID module
- SG90 servo motor
- 16×2 LCD (I2C)
- Green/Red LEDs
- Buzzer
- RFID cards/tags

SKILLS & CONCEPTS

- SPI communication protocol
- RFID card reading & UID matching
- Servo motor control (0°/90°)
- Access control logic
- Security system design

## PBL Phases — Teacher & Student Roadmap

| PBL PHASE | TEACHER ACTIONS | STUDENT ACTIONS | ASSESSMENT |
|---|---|---|---|
| **Entry Event** | Show school access problem: unauthorised entry, lost keys, expensive commercial systems; challenge: 'Design a better solution' | Stakeholder interviews: what do teachers, students, security staff need? | User requirements document: functional + non-functional requirements |
| **Inquiry** | Teach SPI protocol; RFID physics; servo control; security principles | Research: how does RFID work? What are privacy/security concerns? | Technology analysis: RFID vs Bluetooth vs keypad – trade-offs |
| **Build & Iterate** | Guide MFRC522 wiring and UID reading; scaffold access logic | Build: scan card → check UID → unlock (servo 90°) or deny (buzzer) | Access test: 10 valid + 10 invalid cards; false acceptance/rejection rate |
| **Upgrade Challenge** | Challenge: add LCD for user messages; log access attempts to Serial | Add: LCD 'Welcome [Name]' / 'Access Denied'; multi-card system; access log | Security audit: can they bypass the system? How would they improve it? |
| **Public Showcase** | Propose to school administration as a real installation pilot | Present business case: cost vs commercial system; security analysis; demo | Cost-benefit analysis; security assessment report; admin feedback |

## Sample Arduino Code

```
#include <SPI.h>
#include <MFRC522.h>
#include <Servo.h>
MFRC522 rfid(10, 9); // SS, RST
Servo door;
byte authorised[] = {0xAB, 0xCD, 0xEF, 0x12}; // your card UID
void setup() { SPI.begin(); rfid.PCD_Init(); door.attach(6); }
void loop() {
```

```
    if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {
      bool ok = memcmp(rfid.uid.uidByte, authorised, 4) == 0;
      if (ok) { door.write(90); digitalWrite(GREEN, HIGH); delay(3000); door.write(0); }
      else { tone(BUZZ, 500, 1000); digitalWrite(RED, HIGH); delay(1000); }
      rfid.PICC_HaltA();
    }
  }
```

| | | |
|---|---|---|
| **SDG Connections:** SDG 9 – Industry, Innovation & Infrastructure \| SDG 16 – Peace, Justice & Strong Institutions | **Duration:** 3–4 weeks | **Level:** Advanced |

# Appendix A — Universal PBL Assessment Rubric

*Use this rubric across all 10 projects. Adjust weightings per tier and learning context.*

| CRITERION | 1 — Beginning | 2 — Developing | 3 — Proficient | 4 — Exemplary |
|---|---|---|---|---|
| **Technical Functionality** | System does not function; major wiring/code errors | System partially functions; some errors present | System functions correctly; all components respond as designed | System exceeds brief; includes enhancements or error handling |
| **Real-World Connection** | No connection to real-world context evident | Mentions real-world context but does not engage with it | Clearly connects project to real stakeholder need; justified design choices | Deep stakeholder analysis; design decisions evidence-based and data-supported |
| **PBL Process (Inquiry)** | Minimal inquiry; followed instructions only | Some inquiry; tested basic functions | Systematic inquiry; experimented, iterated, and documented process | Extended inquiry; generated original questions; tested multiple design alternatives |
| **Collaboration & Communication** | Limited participation; minimal contribution to team | Contributed to team; needed prompting | Consistent contribution; communicated ideas clearly within team | Led team effectively; facilitated peer feedback; resolved disagreements constructively |
| **Presentation & Audience** | Presentation unclear; audience not considered | Presentation covers project but lacks clarity or audience awareness | Clear, well-structured presentation tailored to audience; includes demo | Professional presentation; compelling narrative; handles questions confidently; proposes next steps |