STEM ARDUINO PBL — PROJECT 3 | TIER 1: FOUNDATION

Motion Detection Alarm

Full PBL Lesson Plan, Assessment Rubric & Teaching Resources

Grades 7–9 • 2 Weeks • SDG 11, SDG 16

PROJECT SNAPSHOT	
Driving Question	"How can communities with limited financial resources use low-cost electronics to build effective home security systems that protect their families?"
Real-World Angle	Low-cost home security for underserved communities — addressing genuine safety needs where commercial alarm systems are financially out of reach
Grade Level	Grades 7–9 (builds on Projects 1 and 2: digital output, threshold logic, Serial Monitor — introduces PIR sensing and timing concepts)
Duration	2 weeks (10 × 45-minute lessons, or equivalent block periods)
SDG Connections	SDG 11 — Sustainable Cities & Communities | SDG 16 — Peace, Justice & Strong Institutions
Key Components	Arduino UNO · HC-SR501 PIR motion sensor · Buzzer (active) · Red LED · Green LED · Jumper wires · Breadboard
Core Skills	Digital input reading · PIR sensor physics and calibration · Boolean logic · Timing with millis() · System states (armed/disarmed) · False alarm reduction strategies
21st C. Skills	Community-centred design · Cost-benefit thinking · Ethical reasoning (surveillance implications) · Communication to a community audience · Systems reliability thinking
Public Audience	Community member, parent, local neighbourhood association representative, or school security staff
Entry Products	Community safety needs analysis + system requirements document
Final Products	Working motion detection alarm prototype + cost analysis + installation guide + community presentation

CURRICULUM STANDARDS COVERAGE	
Science / Physics	Infrared radiation and the electromagnetic spectrum · Passive detection (PIR detects emitted IR, not reflected) · Pyroelectric effect in sensor crystals · Fresnel lens optics for field-of-view shaping
Technology / ICT	Digital input reading with digitalRead() · Boolean state logic (armed/disarmed/triggered) · Time-based logic using millis() vs delay() · System design with multiple states · False positive reduction techniques

Mathematics	Geometry of detection zones (angle, distance, field-of-view calculations) · Reliability statistics (false positive rate, detection rate) · Cost analysis with real component prices · Simple probability (chance of detection given sensor placement)
Social Studies / Ethics	Community safety as a public good · Surveillance ethics and privacy rights · Equity in access to security technology · The role of community-designed solutions vs commercial products
Design Technology	User-centred design for a specific community context · Reliability and failure mode analysis · Installation design (placement, mounting, power) · Cost engineering — achieving requirements at minimum cost
Literacy	Community needs analysis writing · Cost analysis document · Installation guide for a non-technical audience · Persuasive presentation to community stakeholders

How This Project Completes the Tier 1 Foundation

Project 3 is the final project of Tier 1 and the **most socially complex** of the three foundation projects. Where Project 1 introduced empathy through accessibility and Project 2 introduced evidence through data, Project 3 introduces **community equity and ethical reasoning** as legitimate engineering considerations. Students who reach Tier 2 without these habits will build impressive systems that serve no one in particular. The PIR sensor is teachable in one lesson. The question of who deserves security and why it should be affordable is the project's real curriculum.

PBL ELEMENT	HOW IT APPEARS IN THIS PROJECT
Centrality	The community security challenge organises the full two weeks. Boolean logic, millis() timing, PIR calibration, and state machines are learned because students need them to answer the driving question — not as abstract exercises.
Driving Question	The question names both the constraint (limited financial resources) and the community (families). Students must grapple with equity from the start — not just build something that works, but build something that a low-income family could actually afford and use.
Constructive Inquiry	How far does the PIR detect? What causes false alarms? What detection range makes a door-entry alarm useful vs an annoyance? None of these questions can be answered without systematic testing — students must investigate to design.
Student Autonomy	Teams decide: what scenario are they protecting? What arm delay is appropriate? What alert behaviour fits the community context? What cost is acceptable? The teacher constrains the technology; students define the security system.
Realism / Authenticity	The cost constraint is real. Commercial alarm systems cost £80–500+. This project can be built for £6–12. Whether that price difference matters is a question about community equity, not just engineering — and the public audience (a community member or parent) makes that concrete.

Week-by-Week Lesson Calendar

WEEK 1: ENTRY EVENT & INQUIRY

Lesson	PBL Phase	Learning Activities	Teacher Facilitation Notes
L1	**Entry Event**	• Launch: Show UN-Habitat or Amnesty data — what percentage of urban residents in low-income countries live without any formal security system? • Show the cost of a commercial home alarm system (ADT, Ring, Yale) vs the monthly income of a typical informal-settlement household • Pose the Driving Question — discuss: is security a luxury? Who decides who deserves it? • Record 'Need to Know' questions on class board — must include community needs AND technical questions	→ This entry event must provoke moral discomfort before technical interest — sequence matters → Do NOT produce an Arduino yet — let the equity problem drive the inquiry → If students immediately say 'we'll just use an Arduino' — redirect: 'How will a family without technical knowledge install and maintain it?'
L2	**Entry Event**	• Community needs analysis: research a specific community context — what security risks do they face? What has failed for them before? • Requirements workshop: what must the alarm DO? (detect intruders) NOT do? (false alarm at 3am) COST? (under £15 materials) • Stakeholder mapping: who installs it? who uses it? who maintains it? who could misuse it? • Entry product: system requirements document (functional, non-functional, cost, ethical constraints)	→ Push specificity: 'You wrote they need security. Security from what, specifically? At what time of day? At which entry point?' → Introduce the cost constraint as a hard design requirement — not a nice-to-have → Formative: requirements document — is it specific to a real scenario, or generic?
L3	**Inquiry — PIR Physics**	• Mini-lesson: what does PIR stand for? How does a pyroelectric crystal detect infrared radiation? • Demo: hold hand near PIR from different angles and distances — observe Serial Monitor • Investigate: detection range, detection angle, warmup time, sensitivity potentiometer effect • Record findings in a PIR characterisation table — what are the limits of this sensor?	→ Ask: 'Why is it called PASSIVE infrared? What would an ACTIVE infrared detector need?' → Key discovery: PIR needs movement, not just presence. Ask: 'What would happen if a burglar moved very slowly?' → Formative: PIR characterisation table — range, angle, warmup time, false trigger sources recorded
L4	**Inquiry — Digital Input & Boolean Logic**	• Mini-lesson: digitalRead() vs analogRead() — why is the PIR output digital (HIGH/LOW)? • Write code: read PIR state → print to Serial Monitor → observe HIGH/LOW transitions • Add LED response: HIGH = red LED on; LOW = green LED on • Introduce Boolean variables: bool motionDetected = false — why use a variable rather than reading the pin directly in every if statement?	→ Ask: 'If you read the PIR pin twice in one millisecond, could you get different values? Why?' → Common confusion: pin reads HIGH for entire duration of motion window, not just on initial detection → Formative: annotated code — every variable named and commented with its purpose
L5	**Inquiry — Timing & States**	• Problem: why is delay() bad for an alarm system? (blocks all other code — can't arm/disarm during delay) • Introduce millis() timing pattern — non-blocking delay technique • Design the alarm state machine: DISARMED → ARMED (after 10-second arm delay) → TRIGGERED → back to ARMED	→ The delay() vs millis() distinction is one of the most important intermediate Arduino concepts — worth taking time here → State machine diagram on paper first is mandatory —

		• Students draw the state diagram on paper before writing any code	students who skip this write unworkable code → Ask: 'Why does a real burglar alarm have an entry delay? What happens if someone enters while the alarm is arming?'

WEEK 2: BUILD, TEST & COMMUNITY SHOWCASE

Lesson	PBL Phase	Learning Activities	Teacher Facilitation Notes
L6	**Build & Iterate**	• Full system build: PIR + green LED + red LED + buzzer + Arduino in a complete circuit • Implement basic alarm: motion detected → red LED + buzzer for 5 seconds → return to green LED • First reliability test: 10 trials — how many correct detections? How many false positives? • Document in test log: what triggered false positives? (heat sources, air movement, sensor settling)	→ Expected false positives: sensor not warmed up, placed near radiator or window with direct sunlight → Ask: 'If this alarm had a 30% false positive rate in a family home, would they use it? What would they do?' → Formative: 10-trial test log with false positive count and identified causes
L7	**Build & Iterate**	• Implement the full state machine: arm delay → armed → triggered → reset cycle • Test: walk through the detection zone immediately after startup — does the arm delay prevent false trigger? • Calibrate PIR sensitivity and time-delay potentiometers for the chosen deployment scenario • Peer test: one team member leaves room; another arms system; first member re-enters — does it work as designed?	→ The peer test is the first human-behaviour test — students discover that real people move differently from deliberate test motions → Ask: 'How do you set the alarm before you leave? What if you forget and walk back in?' This is the UX problem of every alarm system → Formative: state machine test evidence — each state transition tested and documented
L8	**Upgrade Challenge**	• Introduce differentiated upgrade options: • Upgrade A: Alarm pattern — code a distinctive buzzer tone pattern (e.g. 3 short beeps + pause) using millis() rather than delay() • Upgrade B: Entry delay — add a 10-second entry delay between motion detection and alarm trigger, indicated by fast LED blink, allowing the resident to disarm • Upgrade C: Zone-based detection — add a second PIR sensor for a second entry point; implement independent zone logic (either zone triggers alarm) • Teams plan upgrade on paper first, justify choice with reference to the community scenario	→ Upgrade B introduces the human interaction model of real alarm systems — the most practically important upgrade → Upgrade C requires two PIR modules; introduces multi-sensor logic that bridges to Tier 2 integration projects → Formative: upgrade plan with rationale — must connect to community need, not just technical interest
L9	**Prepare Showcase**	• Calculate total system cost with real component prices; compare to 3 commercial alarm systems • Draft installation guide (see Handout 4) — tested with a non-technical person • Prepare presentation: community context first, cost comparison second, demo third	→ Cost comparison must use real prices — students research online (Amazon, eBay, local electronics shops) → The ethics question is not optional — teams who cannot

		• Ethics discussion: prepare answers to 'Could this alarm be used to control or harm residents?'	address it have an incomplete project → Formative: installation guide usability test — can a non-technical tester follow it without help?
L10	**Community Showcase**	• Showcase: teams present to community audience (parent, community member, school security staff) • Each team: 4-minute presentation + 2-minute live demo + Q&A • Audience completes feedback form (see Handout 5) • Individual reflection in final 10 minutes (see Handout 6)	→ Invite someone who genuinely cares about community safety — not just a sympathetic teacher → Teacher role: facilitator only — do not intervene during Q&A → Summative: presentation rubric (criteria 4 + 5) + written documents (criterion 6)

Detailed PBL Phase Plans

Phase 1 — Entry Event: Security as an Equity Issue

SUB-PHASE / TASK	TEACHER ACTIONS	STUDENT ACTIONS + PRODUCTS
Entry Event Launch (Lesson 1)	Present the cost data and the equity framing before any technology appears. Ask: 'Is it fair that security is only available to people who can afford £200+? What does that mean for communities with lower incomes?' Let moral reasoning precede engineering.	Engage with the equity problem first. Discuss: what do people currently do for security when they cannot afford commercial systems? What has failed? What has worked? Generate Need to Know questions covering both community needs and technical questions. Product: Need to Know board.
Community Needs Analysis (Lesson 2)	Facilitate the requirements workshop with a hard cost constraint: materials must cost under £15 per system. Ask: 'Given this constraint, what does your design have to do and what can it not do?' Introduce the concept of design under constraint.	Research a specific community context (can be their own neighbourhood, a case study, or a researched community). Define: what are they protecting? From whom? At what time? At what entry point? Product: System Requirements Document with hard cost constraint acknowledged.
Entry Assessment	Check: is the requirements document specific to a real scenario or generic? Does it include an ethical constraint — not just 'must detect intruders' but 'must not be used to monitor residents without consent'?	Self-check: 'If I handed this requirements document to an engineer, could they build exactly what my community needs? What is still ambiguous?'

Phase 2 — Inquiry: Understanding the Sensor and the Logic

SUB-PHASE / TASK	TEACHER ACTIONS	STUDENT ACTIONS + PRODUCTS

PIR Physics (Lesson 3)	Facilitate the detection investigation — do NOT tell students the detection range or angle. Ask: 'Design a test that would tell you the exact detection boundary of this sensor in this room.' They must design the experiment before they run it.	Design and execute PIR characterisation experiment: measure range (walk toward sensor until trigger), angle (walk parallel at fixed distance), warmup time (power on → first reliable trigger), false trigger sources. Product: PIR Characterisation Table with methodology note.
Digital Input Logic (Lesson 4)	Ask: 'What does HIGH mean in volts? Measure it with a multimeter or reason from what you know about Arduino.' Connect digital input to digital output reasoning from Project 1 — but now the information flows the other way.	Write and test PIR → LED response code. Introduce Boolean variables with named purpose. Test: can you distinguish a burglar-speed movement from a resident walking slowly? Document: what movement speed reliably triggers the sensor? Product: Annotated detection code with Boolean variable documentation.
State Machine Design (Lesson 5)	Draw a state machine for a real burglar alarm on the board: disarmed, exit delay, armed, entry delay, triggered, alarm. Ask: 'Which of these states does your design need? Which can you omit given your community constraints?' Let students decide the scope.	Draw their own state machine on paper — each state, each transition, each trigger condition. Write code from the diagram, not from guesswork. Test each state transition in isolation before testing the full sequence. Product: State Machine Diagram + Milestone Code.
Inquiry Assessment	Update Need to Know board. Ask: 'Why is millis() better than delay() for an alarm system? Answer in terms of what the alarm needs to do during the delay period.' If students cannot answer — this is the week 2 focus.	Peer quiz: explain the state machine to your partner using only the diagram. No looking at the code. If you need the code to explain it — the diagram is incomplete.

Phase 3 — Build & Iterate: Reliability is the Standard

SUB-PHASE / TASK	TEACHER ACTIONS	STUDENT ACTIONS + PRODUCTS
First Build (Lesson 6)	Circulate with one question: 'What is your false positive rate? What causes it?' False positives are not a sign of failure — they are the data that drives design improvement. Treat them as findings, not bugs.	Build full circuit. Run 10-trial reliability test. Document: detection count, false positive count, identified false trigger sources. Peer circuit review: safety check + detection zone coverage assessment. Product: First Prototype + 10-Trial Reliability Test Log.
State Machine Build (Lesson 7)	After the state machine is implemented, test each transition systematically rather than running the full sequence repeatedly. Ask: 'How do you know the arm delay is exactly 10 seconds? Measure it.' Precision in timing is a security requirement.	Implement full state machine. Test each transition independently with millis() timing verified. Run peer human test: team member leaves, arms system, re-enters — does it behave as designed? Product: State

		Machine Implementation + Transition Test Evidence.
Reliability Improvement (Lesson 7/8)	After peer testing: ask 'What failure modes did your peers find that your own testing missed?' The peer test almost always reveals at least one edge case (too-fast disarming, double-trigger, sensor blind spot from certain angles).	Address at least two identified failure modes. Document: what failed, why it failed, what was changed, what the new test result was. Revision log required. Product: Revised Prototype + Revision Log.
Build Assessment	Assess: is the reliability improvement driven by systematic analysis (failure mode → hypothesis → fix → test) or by random code changes? The engineering process matters as much as the outcome.	Self-assess: 'What is my system's false positive rate after improvements? Is that acceptable for my community scenario? Would a family actually use this system with this false positive rate?'

Phase 4 — Upgrade Challenge: Practical Security Features

SUB-PHASE / TASK	TEACHER ACTIONS	STUDENT ACTIONS + PRODUCTS
Upgrade A — Alarm Pattern (Core+)	Ask: 'Why do commercial alarms use a specific sound pattern rather than a constant tone? Think about who hears it — what do they need to understand from the sound?' Connect sound design to communication design.	Implement a distinctive buzzer pattern using millis() timing (not delay — the state machine must keep running during the tone). Test: can a person who has never heard the alarm before identify it as an alarm rather than a system beep? Product: Alarm pattern demo with non-technical listener test.
Upgrade B — Entry Delay (Advanced)	Introduce the entry delay concept: 'A resident comes home and opens the door — the alarm should give them time to disarm before it triggers. How long? What feedback do they get during the delay? What happens if they don't disarm in time?'	Implement entry delay: motion detection → fast LED blink for 10 seconds → if not disarmed (button press or specific PIR pattern) → full alarm triggers. Test: can a new user understand the entry delay behaviour without reading a manual? Product: Entry delay demo with user comprehension test.
Upgrade C — Dual Zone (Extended)	Introduce multi-sensor logic: 'Your community scenario has two entry points — a front door and a back window. Either can be breached. How do you write code that handles two independent detection zones with one alarm?'	Wire and code second PIR sensor on a separate digital pin. Implement: either zone triggers alarm, both zones shown independently on Serial Monitor. Test: can you identify which zone was triggered from the Serial output? Product: Dual-zone alarm with zone identification output.
Upgrade Assessment	Upgrade A assessed on tone pattern distinctiveness and millis() implementation (no delay()). Upgrade B on entry delay timing accuracy and	Design decision log: upgrade chosen, community rationale, how tested, what was revised, one failure during

	user comprehension without documentation. Upgrade C on independent zone logic and sensor independence test.	development and what was learned from it.

Phase 5 — Showcase: Community Voice and Real Accountability

SUB-PHASE / TASK	TEACHER ACTIONS	STUDENT ACTIONS + PRODUCTS
Cost Analysis (Lesson 9)	Teach the cost comparison structure: 'Your number only means something in comparison. Compare to: (1) the cheapest commercial alarm, (2) the most popular commercial alarm, (3) doing nothing (cost of a burglary).' The third comparison is often the most powerful.	Research 3 commercial alarms with real prices. Calculate: materials cost, time to build (estimate labour), annual maintenance. Calculate: average cost of a residential burglary (research crime compensation data). Present: 'Our system costs X. A commercial equivalent costs Y. The cost of doing nothing, if a burglary occurs, is Z.' Product: Cost Analysis Document.
Installation Guide (Lesson 9)	Coach on audience: 'Your installation guide reader has never heard of an Arduino. They are a parent or community member who wants their family to be safer. Write for them.' Test it before submission.	Write step-by-step installation guide. Test with a non-technical person. Revise based on their confusion points. Minimum required: placement guidance, wiring steps, arm/disarm instructions, troubleshooting. Product: Tested and revised installation guide.
Showcase Prep (Lesson 9)	Frame the presentation order: community story → cost problem → cost solution → demo → ethical considerations. The ethics section is not optional — teams who skip it have an incomplete presentation.	Prepare 4-minute presentation. Practice the ethics Q&A questions in Handout 3. Prepare demo contingency. Product: Presentation outline.
Community Showcase (Lesson 10)	Introduce the audience — a real community member, parent, or security professional. Do not intervene during Q&A. Note which questions teams struggled to answer — these inform the reflection and the next project.	Present to community audience. Lead with the equity problem. Demonstrate live system. Answer ethics questions with a prepared position. Note two audience reactions that were not anticipated. Product: Showcase delivery + feedback forms.
Reflection (Lesson 10)	10 quiet minutes for individual written reflection. No prompting — let the community encounter settle before analysis.	Complete Handout 6. Required: one technical failure and what it taught; one community need the project addresses; one ethical question the project raises that the team did not fully resolve; connection to Project 1 and Project 2 skills. Product: Individual Reflection.

Project 3 Assessment Rubric

Distribute on Lesson 1. *Project 3 introduces a rubric criterion not present in Projects 1 or 2: Ethical Reasoning on Surveillance (Criterion 4). This is deliberately weighted at 15% — not as a token gesture, but because a security system designed without ethical thinking is not a complete engineering product.*

CRITERION	1 — Beginning	2 — Developing	3 — Proficient ✓	4 — Exemplary
Technical Functionality & Reliability (25%)	System does not detect motion; PIR not triggering; buzzer or LED not responding; major wiring or code errors prevent operation.	System detects some motion but is unreliable — frequent false positives (triggers without motion), false negatives (misses obvious movement), or alert does not clear after trigger period ends.	System reliably detects motion within its detection zone and triggers the alert correctly; alert clears after the programmed duration; false positive rate is documented and acceptably low in test conditions; green LED shows safe status, red LED and buzzer activate on detection.	System exceeds brief: armed/disarmed state machine fully implemented; arm delay prevents self-triggering on startup; alert tone is distinct and purposeful; false positive rate quantified from 20+ trials; system tested in realistic deployment conditions (low light, movement at distance, pet-size movement excluded).
PIR Sensor Science Understanding (20%)	No understanding of PIR physics; student cannot explain what 'passive' means or how the sensor detects movement rather than presence.	Surface understanding: student knows the PIR detects movement but cannot explain the pyroelectric effect, why warm-blooded animals trigger it while cold objects may not, or why there is a warmup time.	Clear understanding: student explains that the PIR is passive (detects emitted IR, not reflected), that it senses changes in IR radiation across two sensing elements, that the Fresnel lens shapes the detection field, and that the warmup period allows the sensor to calibrate to ambient IR levels.	Deep understanding: student explains the pyroelectric crystal mechanism, the differential detection principle (why motion is needed, not just presence), the role of sensitivity and time-delay potentiometers, connects PIR principles to real applications (automatic doors, occupancy sensors, wildlife cameras), and discusses limitations (can be fooled by heat sources, fails in very cold or very hot environments where human-ambient contrast is low).
Community-Centred Design & Cost Analysis (20%)	No connection to the community context; system built as a circuit exercise with no reference to cost, community need, or real deployment conditions.	Mentions low-cost or community context but does not demonstrate specific design decisions driven by those constraints; cost not calculated or grossly inaccurate.	Design decisions explicitly connected to community affordability and need; total system cost calculated with real component prices; cost compared to a commercial alarm system; installation	Deep community-centred analysis: student researched actual burglary statistics or security needs in a specific low-income community context; cost-per-installation calculated for a multi-

			requirements considered (power source, mounting, weatherproofing for outdoor use); one specific community deployment scenario described.	room or multi-entry-point deployment; maintenance cost over 1 year estimated; barriers to adoption identified (technical literacy, spare parts availability, power reliability) and design adapted to address them.
Ethical Reasoning on Surveillance (15%)	No engagement with surveillance ethics; project treated purely as a technical challenge with no consideration of who is monitored, who controls the system, or potential for misuse.	Acknowledges that surveillance raises privacy concerns but does not analyse specific implications or propose any design response.	Identifies key ethical tensions: who has access to alarm data? Can the system be weaponised against residents rather than protecting them? What consent is needed to install in shared spaces? Proposes at least one design or policy response to these tensions.	Sophisticated ethical analysis: examines power dynamics (who owns the system, who can arm/disarm it, who receives alerts), considers specific harms (racial profiling, domestic abuse scenarios), draws on a rights framework (privacy as a human right, SDG 16 principles), and proposes a community governance structure for the system's use.
Communication & Community Presentation (20%)	Presentation absent, unclear, or directed at a technical audience; no cost information; no connection to community need.	Presentation covers the project technically but fails to connect to community affordability or safety needs; cost mentioned but not contextualised; audience not considered in language or framing.	Presentation clearly tailored to a community audience; cost comparison with commercial alternatives provided; specific community deployment scenario described; installation guide presented accessibly; ethical considerations addressed honestly.	Compelling community-first presentation: leads with the safety problem and community context, not the circuit; cost analysis includes multi-year TCO; addresses practical community questions ('Who fixes it if it breaks?' 'What if someone trips the alarm accidentally?'); handles questions with evidence and empathy; installation guide could be handed to a non-technical community member and followed immediately.
Written Installation Guide & Cost Analysis (20%)	Documents absent or cover only components; no instructions a non-technical community	Documents present but written for a technical audience; installation steps assume Arduino knowledge; cost listed	Installation guide is step-by-step and usable by a non-technical community member; includes placement guidance,	Installation guide is community-publication quality: includes diagrams or photo descriptions, placement

	member could follow; no costed analysis.	but not compared to alternatives.	power requirements, and a troubleshooting section; cost analysis compares system cost to commercial alternatives and calculates cost per protected entry point.	recommendations based on detection zone physics, maintenance checklist; cost analysis includes 3-year TCO, compares 3 commercial alternatives, and calculates breakeven time vs doing nothing; written at accessible reading level and tested with a non-technical reader.

CRITERION	WEIGHTING	ASSESSED WHEN
Technical Functionality & Reliability	25%	Lessons 6–8 (test logs) + Lesson 10 (summative demo)
PIR Sensor Science Understanding	20%	Lesson 3 (characterisation table) + Lesson 10 (Q&A)
Community-Centred Design & Cost Analysis	20%	Lesson 2 (requirements doc) + Lesson 9 (cost analysis)
Ethical Reasoning on Surveillance	15%	Lesson 2 (requirements) + Lesson 9 prep + Lesson 10 Q&A
Communication & Community Presentation	20%	Lesson 9 (practice) + Lesson 10 (summative)
Written Installation Guide & Cost Analysis	20% (see note)	Lesson 10 final submission
TOTAL	**120% → normalised to 100%**	**Portfolio submitted Lesson 10**

Note: Weightings sum to 120%. Divide raw total by 1.2 to get a percentage score out of 100.

Student Handouts

Handout 1 — 'Need to Know' Starter Questions

WHAT DO I NEED TO KNOW TO ANSWER THE DRIVING QUESTION?

You cannot design an affordable, community-appropriate security system without understanding both the community need and the technology. Work through both columns before touching the hardware.

About the Community Need:

- What are the most common types of home intrusion in low-income urban communities? (research a specific context)
- How much does a basic commercial home alarm system cost to buy and install? (research 3 real products)
- What percentage of households in the community you are designing for could realistically afford a commercial alarm?
- What has happened when communities have tried to use low-cost DIY security solutions before? What failed?
- Who in the community is most at risk and why? (single parents, elderly residents, small business owners?)

About the Technology:

- What does PIR stand for? What is infrared radiation? How does the pyroelectric effect enable detection?
- Why is the sensor called PASSIVE? What would an active IR sensor need that a passive one does not?
- What is the typical detection range and angle of the HC-SR501 PIR sensor? (look up the datasheet)
- What is the warmup time of the PIR sensor and why does it exist?
- What is the difference between delay() and millis() in Arduino code? Why does it matter for an alarm system?
- What is a state machine? Can you find a real-world example of one? (traffic lights, vending machines, elevator)

About the Ethics:

- What is the difference between security and surveillance? When does one become the other?
- If a landlord installs this alarm in a rented property, who controls it — the landlord or the tenant? What should the answer be?
- Could this alarm be used to monitor and control residents rather than protect them? How would you prevent that?

Add your own questions:

1. ____
2. ____
3. ____

Handout 2 — PIR Sensor Characterisation Investigation

PIR CHARACTERISATION — design your test, then run it

Before you test, write your hypothesis: I predict the PIR will detect motion up to ___ metres and within ___ degrees of centre.

Test 1: Detection Range

Walk directly toward the PIR sensor from 5 metres, stopping every 0.5 metres. Record when it first triggers.

Distance (m)	PIR Output (HIGH/LOW)	Triggered? (Y/N)	Notes
5.0			
4.5			
4.0			
3.5			
3.0			
2.5			
2.0			
1.5			
1.0			
0.5			

Test 2: Detection Angle

Stand 2 metres from the sensor. Walk parallel to the sensor face at different lateral positions. Record the angle at which detection fails.

Left edge of detection: ___ degrees from centre   Right edge: ___ degrees   Total detection cone: ___ degrees

Test 3: Warmup Time

Power on the sensor. Record the time at which it first gives a stable reading with no false triggers. Warmup time: ___ seconds

Test 4: False Trigger Sources

Test each of the following — does it trigger the PIR without human movement?

Potential False Trigger Source	Does it trigger? (Y/N)	Notes
Hot air from a radiator or heater		

Direct sunlight moving across the sensor face		
A fan blowing air across the detection zone		
A cold drink placed in the detection zone		
A small animal (toy or object of similar size)		
Rapid temperature change (open window)		

Conclusion: Based on your tests, how should you position the PIR sensor in a real home installation to maximise detection and minimise false alarms?

____

____

Handout 3 — Ethics Preparation & Showcase Q&A

PREPARE THESE ANSWERS BEFORE THE SHOWCASE — your audience will ask them

Your community audience will ask practical AND ethical questions. You need prepared, specific answers for both. Vague answers like 'it depends' or 'we would need to think about that' are not acceptable at the showcase.

Ethics Questions:

'Could a landlord use this to spy on tenants?'

Preparation hint: *Answer requires: who controls the system (arm/disarm)? Is there a log of triggers? Who can access that log? What physical design prevents covert installation? Propose a minimum: the resident must be the one who arms/disarms.*

Your prepared answer: ____

____

'What if the alarm goes off at 3am and frightens children or vulnerable people?'

Preparation hint: *Address your false positive rate from your test data. Address your alarm volume design. Propose: a 30-second entry delay before full alarm. State your actual false positive rate from testing.*

Your prepared answer: ____

____

'Could someone disable this easily?'

Preparation hint: *Be honest. A determined intruder with time could disable most DIY systems. Your system's value is deterrence and early warning, not guaranteed prevention. State this clearly — overclaiming will destroy credibility.*

Your prepared answer: ____

____

'Is this appropriate for shared housing where multiple people live together?'

Preparation hint: *Address: who has access to the arm/disarm mechanism? What happens if one resident arms it and another enters? Propose: a shared disarm code or button accessible to all legitimate residents.*

Your prepared answer: ____

____

Practical Questions:

'How much does this cost to build?'

Preparation hint: *State your exact component cost. State time to build (honest estimate). State comparison to the cheapest commercial alternative.*

Your prepared answer: ____

'What happens when it breaks? Who fixes it?'

Preparation hint: *Address: what components are most likely to fail? Where can replacements be bought locally? Is the installation guide clear enough for self-repair?*

Your prepared answer: ____

Handout 4 — Installation Guide Template

YOUR INSTALLATION GUIDE — write for a community member, not a technician

Test this guide with a non-technical person before submission. If they cannot follow it, revise it.

1. What this system does (3 sentences, plain English)

- What does it detect? What does it do when it detects something? What does safe look like (green LED)?

2. What you will need

- List all components with plain names (not 'Arduino UNO' — 'the controller box')
- List power requirements: USB cable + 5V adapter, or battery pack

3. Where to install the sensor (with diagram)

- Recommended height: ___ cm from floor (based on your detection range tests)
- Recommended angle: point toward the entry point, not toward a heat source

- Avoid: direct sunlight, near heaters, near air conditioning vents
- Sketch a top-view diagram of a room showing optimal sensor placement

4. How to arm the system (numbered steps)

- Step 1: Plug in the USB power cable.
- Step 2: Wait for the green light — this means the system is warming up (about 30 seconds).
- Step 3: Press the arm button / leave the room within the arm delay period.
- Continue with your specific implementation steps...

5. What happens when the alarm triggers

- What does it look like? What does it sound like?
- If you are the resident returning home: [your entry delay procedure]
- If the alarm is unexpected: [your recommended response]

6. How to disarm (numbered steps)

- Your specific disarm procedure

7. Troubleshooting

- Green light flashes but no alarm when I walk past: sensor may need warmup — wait 60 seconds after power-on
- False alarms with no visible movement: check for heat sources within 2 metres; reposition sensor
- System does not respond to movement: check USB power; press reset button

8. Cost to build this system

- List all components with individual prices
- Total: £___
- Compared to the cheapest commercial alternative (name the product): £___

Usability test: Tester name: ___________ What confused them: ___________ What was changed: ___________

Handout 5 — Community Audience Feedback Form

COMMUNITY AUDIENCE FEEDBACK FORM (for invited guests)

Team name / number: ______________________________________ Date: _______________

1. Does this system address a real community need?

Did the team understand the security problem in the community they designed for? Was the context specific and credible?

Rating: 1 — Needs work 2 — Adequate 3 — Good 4 — Excellent

Comments: ____

2. Is it genuinely affordable?

Based on the cost analysis presented, could a family with a limited income realistically build or buy this? Is the cost comparison to commercial systems accurate and fair?

Rating: 1 — Needs work 2 — Adequate 3 — Good 4 — Excellent

Comments: ____

3. Would you trust this system?

Based on the test data presented, is the false positive rate acceptable? Would you feel comfortable with this alarm in your home?

Rating: 1 — Needs work 2 — Adequate 3 — Good 4 — Excellent

Comments: ____

4. Could you install and use this without technical help?

Looking at the installation guide: could you follow it? Could you arm and disarm the system without assistance?

Rating: 1 — Needs work 2 — Adequate 3 — Good 4 — Excellent

Comments: ____

5. Were the ethical concerns handled honestly?

Did the team identify how this system could be misused? Did they propose credible safeguards? Were they honest about limitations?

Rating: 1 — Needs work 2 — Adequate 3 — Good 4 — Excellent

Comments: ____

Overall: Would you recommend this system to someone in your community? Yes / No / With changes

If 'With changes': what specifically? ____

Handout 6 — Individual Reflection Prompts

REFLECTION — reference specific moments and data from your project

Complete individually in the final 10 minutes of the showcase. Be specific — general statements will not receive credit.

Technical learning:

Describe the moment you most clearly understood why millis() is better than delay() for an alarm system. What was the problem you were trying to solve when the understanding arrived?

____

____

____

False positive investigation:

What caused your most stubborn false positive during testing? How did you identify the cause? What did you change, and did it work? What would you do differently in a permanent installation?

____

____

____

Community design:

Give one specific example of a design decision you made differently because of the community context — not because of technical convenience. What would you have done if you were just building the most technically impressive system?

____

____

____

Ethical reasoning:

Describe one way this alarm system could be used to harm or control a resident rather than protect them. What single design or policy change would most reduce that risk?

____

____

____

Tier 1 synthesis:

This is your third Arduino project. What is one skill or habit from Project 1 or Project 2 that you used in Project 3 without thinking about it — that has now become automatic? What does that tell you about how your learning has changed?

____

____

____

Looking to Tier 2:

The next projects combine multiple sensors and components into integrated systems. What is the one thing about this project — technically or in terms of design thinking — that you most want to get stronger at before you start that work?

____

____

____

Teacher Reference: Common Misconceptions & Facilitation Moves

Project 3 surfaces a different category of misconception than Projects 1 and 2 — alongside the expected technical errors, students frequently misunderstand what a reliable system actually requires and underestimate the engineering complexity of a 30-second arm delay.

STUDENT MISCONCEPTION	CORRECT UNDERSTANDING	TEACHER FACILITATION MOVE
'The PIR detects body heat'	The PIR does not detect static body heat — it detects changes in infrared radiation across two sensing elements. A person standing perfectly still in the detection zone will eventually stop triggering the sensor as the differential signal reduces to zero. The sensor detects movement across a thermal gradient, not presence.	*Demonstrate: stand in the detection zone and hold perfectly still. After 20–30 seconds, the output drops LOW. Ask: 'Did you leave? Did you cool down? What changed?' Then move slowly — it triggers again. This is the most memorable demonstration in the project.*
'false alarms mean something is wrong with my circuit'	False positives are a fundamental characteristic of PIR sensors, not circuit faults. Heat sources, direct sunlight, HVAC air movement, and nearby reflective surfaces all cause legitimate false triggers. Managing false alarms through placement, calibration, and code logic is the design challenge — not eliminating them by finding a wiring error.	*Ask: 'What specific conditions caused your false alarms? Record them.' Then: 'If you were installing this in a real room, how would you position the sensor to minimise exposure to those conditions?' Reframe false alarm reduction as environmental design, not debugging.*
'I can use delay() for the alarm duration — it still works'	Using delay(5000) for a 5-second alarm duration freezes all code execution for 5 seconds — including any code that checks for disarming input. During that delay, the Arduino cannot read a button press, process a second sensor, or update the display. For a basic demonstration it works; for a usable alarm system, it is a design failure that will surface during user testing.	*Ask: 'What if someone walks in during your alarm delay? Can they disarm it?' Test it. Students almost always discover the problem when they try to disarm during an active alarm. The moment they cannot disarm is the moment they understand why millis() matters.*
'30 seconds is too long for a warmup delay — I'll skip it'	The PIR warmup period (typically 30–60 seconds) allows the sensor to establish a baseline reading of the ambient IR environment. Without it, the sensor will false-trigger on startup as it adjusts — which means the alarm fires every time the system is powered on. Skipping the warmup produces the most common and most frustrating student false positive.	*Ask: 'What happens when you power on your system and immediately walk in front of it?' Then: 'What happens if you wait 60 seconds first?' Let students discover the warmup effect empirically. They remember this because they experienced the false alarm, not because they read about it.*

'Our system is cheap so it must be good for communities'	Cost is a necessary but not sufficient condition for community appropriateness. A £10 system that requires an Arduino IDE to add authorised users, needs a computer to reprogram, fails in high temperatures, or triggers false alarms at 2am is not community-appropriate regardless of its component cost. Community design also requires ease of use, reliability, maintainability, and cultural fit.	*Ask: 'If a family with no technical background and no computer bought your system — could they install it, use it, and maintain it? What would break first? Who would they call?' This question surfaces the difference between being cheap and being useful.*
'Ethics questions are about feelings, not engineering'	Ethical constraints are engineering requirements. A security system that can be weaponised against residents, generates surveillance data without consent, or excludes people with disabilities from armed/disarmed control is a system with design failures — not just moral concerns. The ethics rubric criterion (15%) exists because professional engineers are accountable for the uses of what they build.	*When students dismiss ethics questions as subjective: 'Let me reframe that as an engineering requirement. Write this in your requirements document: the system must not enable monitoring of residents without their explicit consent. Now: does your current design meet that requirement? How would you test it?'*

Differentiation Strategies

LEARNER PROFILE	SUPPORT SCAFFOLDS	EXTENSION CHALLENGES
Learners needing additional support	Pre-drawn state machine diagram with blanks to fill in · PIR wiring guide (3 pins labelled: VCC, GND, OUT) · Code template with state variable and transition stubs pre-written · Community scenario provided (no independent research required) · Cost comparison table template with 2 products pre-filled · Ethics discussion with sentence starters	Basic alarm only: motion → buzzer + red LED for 5 seconds → green LED · Fixed 30-second warmup in code · 5-trial reliability test · Installation guide with 4 sections only · Cost comparison to one commercial product · Ethics: identify one misuse risk in writing
On-track learners	Standard project brief · BIE phase check-ins · All 6 rubric criteria targeted · Peer test in Lesson 7	Full state machine (arm delay + armed + triggered + reset) · One upgrade chosen · 10-trial reliability test with FAR documented · Full 8-section installation guide tested with non-technical reader · Full cost analysis with 3 commercial comparisons · Ethics addressed in both requirements doc and presentation
Advanced learners	Minimal scaffolding · Self-directed community research · Design own state machine from real alarm system analysis · Research pyroelectric effect from a physics source	All three upgrades (tone pattern + entry delay + dual zone) · 20+ trial reliability test with statistical false positive rate · Community context based on real research (named community, cited statistics) · 3-year TCO cost analysis · Ethics framework based on a named

		rights document (UDHR, SDG 16) · Presentation to a real community organisation or neighbourhood watch group

Materials List & Approximate Costs

COMPONENT	QTY (per team)	UNIT COST (USD approx.)	NOTES
Arduino UNO (or Nano)	1	$4–8	Nano is cheaper and easier to mount in a compact enclosure for the deployment scenario
HC-SR501 PIR motion sensor	1	$1–3	Includes sensitivity and time-delay potentiometers; allow 60-second warmup in all tests
Active buzzer (5V)	1	$0.30–0.80	Active = buzzes on HIGH without tone(); simpler for alarm use; passive if students want tone patterns (Upgrade A)
Passive buzzer (Upgrade A)	1	$0.30–0.80	Required for Upgrade A tone patterns; use tone() and noTone()
Red LED (5mm)	1	$0.10–0.20	For alarm/triggered state indicator
Green LED (5mm)	1	$0.10–0.20	For safe/armed state indicator
220Ω resistors	2	$0.05 each	One per LED
Push button (momentary) — Upgrades B/C	1	$0.20–0.50	Optional; for disarm implementation in Upgrade B
Second HC-SR501 PIR sensor — Upgrade C	1	$1–3	Optional; for dual-zone implementation in Upgrade C
Breadboard (half-size)	1	$1–2	One per team
Jumper wires (assorted)	15–20	$0.50 pack	Male-to-male; include long wires for sensor positioning tests
TOTAL per team (core kit)	—	≈ $8–15	One of the most affordable full-function security systems available; use this price point in the cost comparison presentation